

**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN**

TEMA:

**Diseño de un sistema automatizado de permisivos aplicados en
fábricas de balanceado acuícola.**

AUTOR:

Rivera Camacho, Nahomy Sughey

**Trabajo de Titulación previo a la obtención del título de
INGENIERA EN ELECTRÓNICA Y AUTOMATIZACIÓN**

TUTOR:

Ing. Bonilla Sánchez, Ronnie Alexander, MSc.

Guayaquil, Ecuador

4 de septiembre del 2025



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

**FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN**

CERTIFICACIÓN

Certifico que el presente trabajo de realizado en su totalidad por la Sra. Rivera Camacho, Nahomy Sughey como requerimiento para la obtención del título de INGENIERA EN ELECTRÓNICA Y AUTOMATIZACIÓN.

TUTOR

f. _____

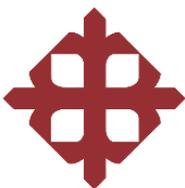
Ing. Bonilla Sánchez, Ronnie Alexander, MSc.

DIRECTOR DE CARRERA

f. _____

Ing. Bohórquez Escobar Celso Bayardo, PhD.

Guayaquil, a los 4 días del mes de septiembre del año 2025



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

**FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN**

DECLARACIÓN DE RESPONSABILIDAD

Yo, Rivera Camacho, Nahomy Sughey

DECLARO QUE:

El Trabajo de Titulación **Diseño de un sistema automatizado de permisivos aplicados en fábricas de balanceado acuícola**, previo a la obtención del título de **Ingeniera en Electrónica y automatización**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 4 días del mes de septiembre del año 2025

LA AUTORA

f. _____

Rivera Camacho, Nahomy Sughey



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

**FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN**

AUTORIZACIÓN

Yo, Rivera Camacho, Nahomy Sughey

Autorizo a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución del Trabajo de Titulación: **Diseño de un sistema automatizado de permisivos aplicados en fábricas de balanceado acuícola**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

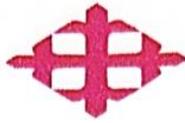
Guayaquil, a los 4 días del mes de septiembre del año 2025

LA AUTORA

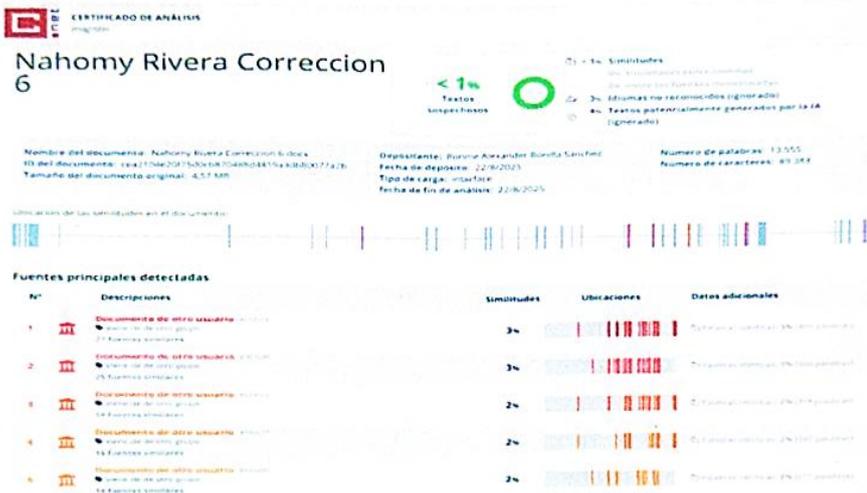
f. _____

Rivera Camacho, Nahomy Sughey

REPORTE COMPILATIO



CARRERA DE INGENIERÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN FACULTAD PARA LA EDUCACIÓN TÉCNICA PARA EL DESARROLLO



Se revisó el Trabajo de Titulación, **Diseño de un sistema automatizado de permisos aplicados en fábricas de balanceado acuícola** presentado por el estudiante Rivera Camacho, Nahomy Sughey de la carrera de Ingeniería en Electrónica y automatización, donde obtuvo del programa COMPILATIO, el valor de $< 1\%$ de coincidencias, considerando ser aprobada por esta dirección.

Certifica.

Tutor:

f.

Ing. Bonilla Sánchez, Ronnie Alexander

AGRADECIMIENTO

Hoy culmina una etapa muy importante de mi vida y no puedo dejar pasar la oportunidad de agradecer a quienes hicieron posible este logro.

A Dios, por darme salud, fortaleza y la oportunidad de aprender; y por sostenerme en los momentos en que no creí continuar.

Con mucho cariño a mis padres, Luis y Celia, que con su amor y ejemplo me enseñaron lo valioso que es el esfuerzo, la perseverancia, recordándome siempre que todos los sacrificios habrán valido la pena.

A mis hermanos, Diego, Joe, Jennifer, Matías y Luis, gracias por la paciencia y por creer en mí incluso en los días más trabajosos. En especial a mi hermana Jennifer, por siempre brindarme sus consejos y no dejarme sola en ninguna circunstancia.

A mis compañeros, gracias por todas las risas y el apoyo mutuo en este camino universitario, y por demostrarme que el camino es más llevadero cuando es en equipo.

De manera especial, a David, por su apoyo constante, por acompañarme en cada paso y por esperarme tantas noches hasta que terminara mis clases, dándome la tranquilidad de saber que no estaba sola.

A mis docentes, por su dedicación y por inspirarme a ir más allá de lo que imaginaba posible.

A cada uno, gracias de corazón.

LA AUTORA

Rivera Camacho, Nahomy Sugely.

DEDICATORIA

A mis padres, Luis y Celia, por ser ejemplo de fortaleza y la inspiración que me impulsa a seguir adelante.

A mis hermanos, con quienes comparto cada logro y aprendizaje de este camino, y en especial a Luis, porque me ha demostrado que, a pesar de las dificultades, siempre es posible salir adelante y brillar con luz propia.

Dedico también este logro, con especial cariño, a mi sobrina Nadia, que se ha convertido en esperanza y en un motivo más para luchar y dar lo mejor de mí.

LA AUTORA

Rivera Camacho, Nahomy Suguey.



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN

TRIBUNAL DE SUSTENTACIÓN

f. _____

ING. BOHÓRQUEZ ESCOBAR CELSO BAYARDO, PhD.

DIRECTOR DE CARRERA

f. _____

ING. UBILLA GONZÁLEZ RICARDO XAVIER, MSc.

COORDINADOR DEL ÁREA DE LA CARRERA

f. _____

ING. HERAS SANCHEZ, MIGUEL ARMANDO, MSc.

OPONENTE



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN

CALIFICACIÓN

10/10

f. _____

ING. BONILLA SÁNCHEZ RONNIE ALEXANDER, MSC.

TUTOR

Índice General

RESUMEN.....	XIV
ABSTRACT.....	XV
CAPÍTULO 1: DESCRIPCIÓN GENERAL DEL TRABAJO DE TITULACIÓN	2
1.1. Introducción	2
1.2. Definición del problema.....	4
1.3. Justificación del problema	5
1.4. Objetivos del problema.....	5
1.4.1. Objetivo general	5
1.4.2. Objetivos específicos	6
1.5. Hipótesis.....	6
1.6. Metodología de la investigación	6
CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA.....	7
2.1. Automatización industrial.....	7
2.1.1. Conceptos de automatización.....	7
2.1.2. Niveles de automatización.....	9
2.1.3. Importancia en procesos industriales	10
2.2. Controladores lógicos programables (plc).....	11
2.2.1. Definición de plc	11
2.2.2. Arquitectura básica y funcionamiento.....	12
2.2.3. Ciclo de scan del plc.....	13
2.2.4. Señales y módulos de expansión	14
2.2.5. Lenguajes de programación.....	15
2.2.6. Aplicaciones.....	18
2.2.7. Software tia portal como entorno de automatización.....	19
2.3. Sistema scada	20
2.3.1. Definición y componentes.....	21
2.3.2. Funciones principales	22
2.3.3. Protocolo de comunicación	22
2.3.4. Ignition.....	25
2.4. Permisivos de secuencia.....	27
2.4.1. Definición y tipos	27
2.4.2. Lógica de funcionamiento	29
2.5. Instrumentación y señales de campo.....	31
2.5.1. Instrumentación	31
2.5.2. Módulos de expansión de señales de campo	33
CAPÍTULO 3: ESTUDIO DE PARÁMETROS DEL PROCESO EN LA PRODUCCIÓN DE ALIMENTO ACUÍCOLA	36
3.1. Secuencia de producción	36
3.1.1. Intake	36
3.1.2. Dosificación y mezclado.....	38
3.1.3. Molienda y pulverizado.....	39
3.1.4. Extrusión	40
3.1.5. Revestimiento y empaque	41
3.2. Permisivos.....	42
3.2.1. Secuencia plc	42
3.2.2. Seguridad plc	45
3.2.3. Arranque plc.....	48
3.2.4. Etapas del proceso de popup ignition.....	50
3.2.5. Script y binding en ignition	54
3.2.6. Visualización	57
CAPÍTULO 4: DISEÑO DEL SISTEMA DE CONTROL Y LÓGICA DE PERMISIVOS PARA LA LÍNEA DE PRODUCCIÓN	59

4.1.	Virtualización.....	59
4.2.	Red ethernet/ip.....	60
4.3.	Plcsim	63
4.4.	Conexión de ignition con plcsim	64
4.5.	Cliente del scada.....	65
CAPÍTULO 5: ANÁLISIS DE RESULTADOS Y PRESUPUESTO.....		69
5.1.	Análisis técnico de los resultados obtenidos	69
5.2.	Eficiencia operativa.....	70
5.3.	Presupuesto referencial	70
Conclusiones		74
Recomendaciones		75
Referencias		76
Anexos		80

Índice de Figuras

Figura 1. Estructura de un sistema automatizado.....	8
Figura 2. Pirámide de automatización.....	9
Figura 3. PLC siemens S7-1200.....	12
Figura 4. Arquitectura de un PLC.....	13
Figura 5. Ciclo de operación de un PLC.....	14
Figura 6. Ejemplo de programación Ladder.....	16
Figura 7. Ejemplo de programación SFC.....	18
Figura 8. SoftwareTia portal.....	20
Figura 9. Arquitectura Modbus.....	23
Figura 10. Arquitectura OPC.....	24
Figura 11. Arquitectura Profinet siemens.....	24
Figura 12. Arquitectura Ethernet/IP.....	25
Figura 13. Árbol de proyecto Ignition.....	26
Figura 14. Navegador de tags.....	28
Figura 15. Estructura lógica de encendido de un motor.....	28
Figura 16. Bloque de programa.....	29
Figura 17. Estructura del permisivo de proceso.....	30
Figura 18. Pantalla de secuencia intake.....	30
Figura 19. Estructura del permisivo de seguridad.....	31
Figura 20. Lenguaje SCL de siemens para señales de entrada de campo.....	32
Figura 21. Imagen de entrada analógica.....	33
Figura 22. Módulo de expansión ET-200.....	34
Figura 23. Red de PLC con módulos de expansión.....	35
Figura 24. Área de intake.....	37
Figura 25. Área de dosificación.....	38
Figura 26. Área de molienda y pulverización.....	39
Figura 27. Área de extrusión.....	41
Figura 28. Área de revestimiento y empaque.....	42
Figura 29. Logica LADDER de permisivo de proceso.....	43
Figura 30. Diagrama de secuencia en sistema SCADA.....	43
Figura 31. Condición del bit OutOpen.....	44
Figura 32. Lógica LADDER de permisivo de seguridad.....	45
Figura 33. Lógica por falta de seccionador de campo.....	46
Figura 34. Lógica por falta de guardamotor.....	47
Figura 35. Esquema de permisos de arranque en pulverizador.....	48
Figura 36. Timer de permisivo de arranque.....	49
Figura 37. Árbol de proyecto del popup de interloks.....	51
Figura 38. Popup de interloks.....	52
Figura 39. Árbol del proyecto de t�mplate para interloks.....	52
Figura 40. T�mplate de Interloks.....	53
Figura 41. Binding de stilos.....	54
Figura 42. Instancias con propiedad binding.....	55
Figura 43. Binding de permisos.....	55
Figura 44. Script del binding.....	56
Figura 45. Informaci�n en el tag.....	57
Figura 46. Permisivos de seguridad en SCADA.....	57
Figura 47. Permisivos de proceso en SCADA.....	58
Figura 48. Permisivos de arranque en SCADA.....	58

Figura 49. Ide VMWare.	59
Figura 50. Configuración de red VMWare.	60
Figura 51. Comando IP por cmd en sistema virtualizado.	60
Figura 52. Comando IP por cmd en sistema anfitrión.	61
Figura 53. Ping desde el servidor anfitrión al virtualizado.	61
Figura 54. Gateway SCADA por dirección IP virtualizada.	62
Figura 55. Simulador de PLC Siemnes.	63
Figura 56. Declaración de PLC OPC en Ignition via Ethernet/IP.	64
Figura 57. Lista de PLC declarados en Ignition.	64
Figura 58. Áreas de trabajo en la línea de producción.	65
Figura 59. Pantalla de Intake.	66
Figura 60. Pantalla de dosificación.	66
Figura 61. Pantalla de Molienda y pulverizado.	67
Figura 62. Pantalla de Extrusión.	67
Figura 63. Pantalla de revestimiento y empaque.	68
Figura A1. Pantalla de inicio del sistema SCADA.	80
Figura A2. Pantalla de intake.	80
Figura A3. Pantalla de Dosing y Mixing.	81
Figura A4. Pantalla de Pulverizado y Molienda.	81
Figura A5. Pantalla de extrusión.	82
Figura A6. Pantalla de Coating y Packing.	82

Índice de Tablas

Tabla 1. Presupuesto de horas hombres.	71
Tabla 2. Puesta en marcha por línea.	71
Tabla 3. Equipos requeridos.	72
Tabla 4. Inversión total estimada.	73

Resumen

El objetivo de este proyecto de titulación se determinó como el desarrollo de diseñar un sistema de permisos de proceso, seguridad y arranque en una planta automatizada de producción de alimento acuícola. Para esto se utilizó un controlador lógico programable (PLC S7-1500) y el software de SCADA Ignition, lo que permitió probar y verificar la lógica de control utilizando PLCSIM, sin ninguna alteración en el proceso. El sistema desarrollado estableció condiciones de proceso, interbloqueos de seguridad y restricciones de arranque, garantizando que los equipos no deben funcionar a menos que se cumplan las condiciones establecidas. De este modo, se redujeron los riesgos eléctricos, se evitó la mala operación de los equipos y se mejoró la seguridad de los operadores. A su vez, se diseñó una interfaz gráfica para el operador con ventanas eventuales tipo popup, lo que ayudó a la supervisión de estados y las condiciones en tiempo real, llevando a una mejor interacción entre los operadores y el sistema. Los resultados obtenidos indicaron una reducción significativa en las fallas de línea, sin necesidad de una prueba de arranque y que el tiempo para iniciar la línea estaba asegurado, siendo una alternativa técnica y económicamente viable.

Palabras clave: permisos de proceso, seguridad industrial, arranque automático, automatización, SCADA, PLC

Abstract

The objective of this degree project was to develop a system for process permits, safety, and startup in an automated aquaculture feed production plant. A programmable logic controller (PLC S7-1500) and Ignition SCADA software were used to develop this system. This enabled the control logic to be tested and verified using PLCSIM, without any process disruption. The system established process conditions, safety interlocks, and startup restrictions, ensuring that equipment cannot operate unless the established conditions are met. This reduced electrical risks, prevented equipment malfunctions, and improved operator safety. A graphical operator interface with pop-up windows was also designed, which facilitated real-time status and condition monitoring, leading to improved interaction between operators and the system. The results obtained indicated a significant reduction in line failures, without the need for a start-up test, and that the time to start the line was guaranteed, making it a technically and economically viable alternative.

Keywords: process permissives, industrial safety, automatic startup, automation, SCADA and PLC

CAPÍTULO 1: DESCRIPCIÓN GENERAL DEL TRABAJO DE TITULACIÓN

1.1. Introducción

En la actualidad el desarrollo tecnológico y la automatización industrial han transformado profundamente los procesos de producción en diversos sectores tales como petrolíferos, acuícolas, textiles, etc. Las plantas productoras de alimento animal requieren de sistemas altamente eficientes y seguros que garanticen la continuidad de sus procesos sin comprometer la integridad de los equipos, la calidad del producto, accidentes industriales, economización energética y de materia, y seguridad del personal. En este contexto la correcta implementación de permisivos de procesos, seguridad y arranque se vuelve necesaria para el funcionamiento óptimo de las líneas de producción automatizadas.

Un permisivo es definido como una condición lógica o física que debe cumplirse para habilitar el encendido de uno o más motores, aperturas y cierres de válvulas, los cuales serán definidos ahora como módulos de control debido a que su acción de proceso es controlada. Los permisivos de procesos garantizan el orden lógico de las operaciones dentro del sistema de producción. Por ejemplo, para el encendido de un transportador que moviliza materia es necesario que se evalúe sus condiciones como los sensores de nivel de la tolva al que el producto va a ingresar, en caso de existir más de una tolva también se debe de analizar por el nivel de las tolvas, la disponibilidad de esta y según esto se realice la apertura de una válvula que permita la descarga de un material a la tolva a través del transportador. Los permisivos de seguridad, por su parte esta relacionados con sensores y

dispositivos que detectan situaciones anómalas, como velocidades excesivas o atascamientos en el transportador, alarmas de sobre corriente y sobrecarga en caso de motores, los cuales pueden generar fallas mecánicas o accidentes, también en caso de equipos grandes como mezcladores, pulverizadores, texturizadores y extrusores los cuales poseen una cámara donde fácilmente una persona puede realizar algún trabajo mecánico o de mantenimiento es requerimiento tener sensores de que las compuertas de esas cámaras se encuentren cerradas, los cuales deben ser tomados como permisivos de seguridad. Finalmente, los permisivos de arranque tienen como objetivo controlar el encendido secuencial de equipos, particularmente en entornos con múltiples líneas de producción, donde el arranque simultáneo de motores puede causar picos de consumo eléctrico que pueden desestabilizar el sistema.

Sin embargo, en muchas industrias no se dispone de un sistema integrado que considere estas tres categorías de permisivos, lo que puede dar lugar a tiempos de inactividad no programados, daños en los equipos y riesgos para la seguridad. Por lo tanto, de ahí surge la necesidad de diseñar y analizar un sistema que integre estos factores de manera estructurada y flexible a diferentes situaciones de producción.

El presente trabajo tiene como finalidad diseñar y analizar un sistema de permisivos de proceso, seguridad y arranque, enfocándose en su aplicación en módulos de control dentro de fábricas e industrias automatizadas productoras de balanceado acuícola. Esta propuesta busca no solo mejorar la eficiencia del proceso, sino también incrementar la confiabilidad de la planta y la protección de los activos industriales al integrar

principios de ingeniería de control y gestión energética.

1.2. Definición del Problema

Las plantas industriales automatizadas cuyos procesos operativos involucran el uso de módulos de control los cuales deben de responder a una lógica de control estricta muy aparte de su funcionamiento autónomo es requerimiento que se incluya una serie de condiciones de proceso, seguridad y arranque, sin embargo aún no hay plantas las cuales carecen de un sistema estructurado de permisos que integre adecuadamente las condiciones necesarias para el arranque, la ejecución de procesos y la protección de los equipos y operadores.

Se suelen presentar situaciones en las que los módulos de control son accionados sin una verificación previa de los estados de físicos, lógicos u otros elementos de proceso, lo que puede derivar en pérdidas de producto, obstrucciones, contaminación entre líneas de producción y fallas mecánicas. Así mismo la ausencia de permisos de seguridad incrementa el riesgo de daños a los equipos y personal operativo. Otro problema frecuente es el arranque de motores de gran potencia, al arrancar varias líneas de producción de forma simultánea estos producen picos muy altos de corriente para pasar la inercia los cuales afectan la estabilidad del sistema eléctrico y aumentan los costos energéticos.

Esta ausencia de control y monitoreo organizado no solo compromete la integridad del proceso, si no también impacta fuertemente en los costos operativos, la eficiencia eléctrica y continuidad de la producción. Igualmente, limita la posibilidad de evolucionar hacia esquemas de fabricación más complejos, ya que se dificulta las diversas integraciones sin un control sólido.

También limita la capacidad de diagnosticar y prevenir de fallos, al no contar con una lógica central para verificar las condiciones antes de ejecutar una acción.

En consecuencia, surge la necesidad de diseñar un sistema de permisos que integre de forma lógica y automatizada las condiciones de procesos, seguridad y arranque, de forma interna en la programación del PLC y de forma visual en un sistema SCADA, dando así información exacta de estas condiciones al personal operativo, eléctrico y de mantenimiento dentro de una planta automatizada.

1.3. Justificación del Problema

El presente proyecto de titulación se realizará principalmente por la necesidad de valorar condiciones industriales para mejorar la continuidad de la producción, seguridad operativa, ahorro energético.

Así surge la idea del desarrollar una estructura de condiciones lógicas con referencia a señales de instrumentos físicos industriales la cual corresponderá a programación en PLC y un sistema visual donde se puedan visualizar estas condiciones requeridas para la manipulación de módulos de control.

1.4. Objetivos del Problema

1.4.1. Objetivo General

Se posee como desarrollar un sistema de control automatizado que integre programación PLC y supervisión mediante SCADA para una línea de producción de alimento acuícola, incorporando un esquema de permisos que asegura el correcto funcionamiento en secuencia operativa, la protección de equipos y la eficiencia energética del proceso.

1.4.2. Objetivos Específicos

- Identificar los parámetros del proceso de la línea de producción para fabricar alimento acuícola.
- Elaboración de lógica de control de proceso de producción y permisos aplicados y mostrados en un sistema SCADA.
- Análisis de resultados y presupuesto.

1.5. Hipótesis

Al implementar el sistema de permisos expuestos en el presente trabajo permitirá mejorar la secuencia operativa, reducir riesgos y fallas por diversos factores y ahorro energético al evitar arranques simultáneos en una línea de producción industrial.

1.6. Metodología de la investigación

El presente trabajo de titulación tendrá pertenece al tipo de metodología aplica debido a que busca resolver un problema práctico el cual es la necesidad de un sistema SCADA con lógica de permisos en una planta industrial automatizada, aplicando conocimientos teóricos y técnicos en una solución funcional.

En adición también puede interpretarse con la metodología de tipo proyectiva debido a que abarca el desarrollo de un sistema técnico estructurado debido a que no solo se analizara la solución si no que se desarrollara el software requerido para este proyecto.

CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA

2.1. Automatización industrial

En las últimas décadas ha sido tendencia la automatización de procesos de manera progresiva en las industrias. Esta tendencia ha permitido el desarrollo tecnológico y la disminución de costes en componentes electrónicos. Siendo la automatización de los procesos de producción aquella que persigue los objetivos indicados a continuación:

- Mejorar los procesos y mantener un nivel de calidad uniforme.
- Preservar de manera continua la producción del producto.
- Mejorar la calidad reducir costes.
- Hacer más flexible el sistema productivo.

Estos objetivos se han convertido de hecho en requisitos indispensables para mantener la competitividad, por lo que el aumento del nivel de automatización de los procesos es simplemente una necesidad para sobrevivir en el mercado actual. (Ajiga, 2024)

2.1.1. Conceptos de automatización

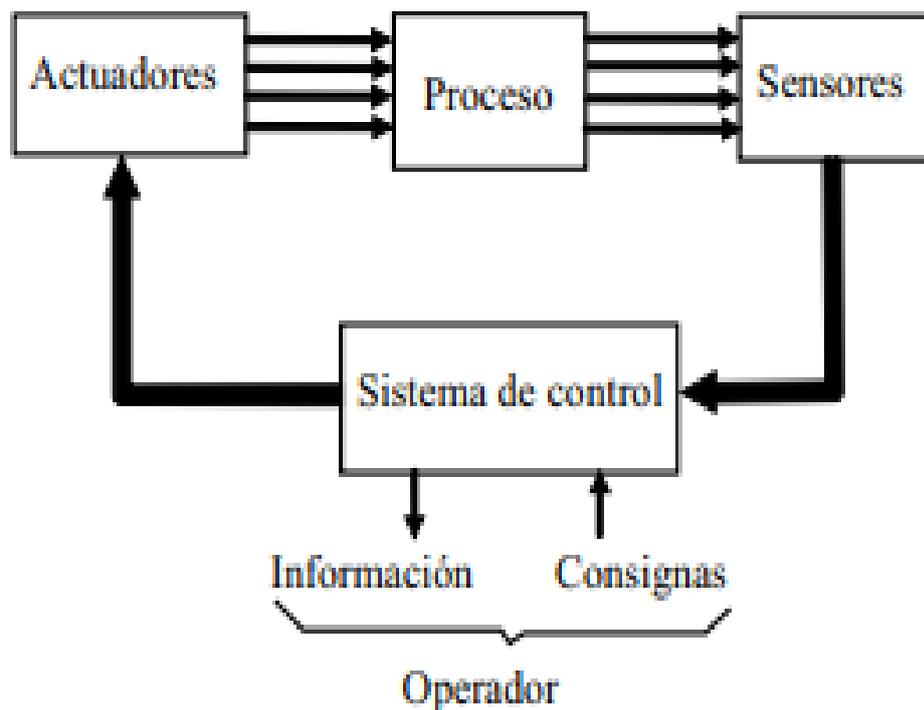
Se define un sistema automatizado como aquel capaz de reaccionar de forma automática ante los cambios que se producen en el mismo, desarrollando las actividades apropiadas para lograr el objetivo para lo que ha sido diseñado. La automatización consiste en un bucle cerrado, donde la información sobre los cambios del proceso captada por los sensores es procesada dando lugar a las acciones necesarias, que se implementan físicamente sobre el proceso por medio de los actuadores. Este sistema de control se comunica eventualmente con el operador, recibiendo de éste consignas de funcionamiento, tales como marcha, paro, cambio de

características de producción y comunicándole información sobre el estado del proceso para la supervisión del correcto funcionamiento. (Ajiga et al., 2024)

La Figura 1 muestra la estructura típica de un sistema automatizado.

Figura 1.

Estructura de un sistema automatizado.



Nota. Elaborado por Ajiga, 2024.

Se denomina automatismo al sistema completo, aunque con este término suele hacerse referencia fundamentalmente al sistema de control, ya que es este el que produce de forma automática las acciones sobre el proceso a partir de la información mediante las señales de entradas y de salidas, estas señales pueden ser de cualquier tipo, sin embargo el concepto tradicional de automatismo se utiliza para sistemas de eventos discretos también llamados sistemas secuenciales en los que las señales son binarias (Davydov, Larionov, & Nagul, 2024).

2.1.2. Niveles de automatización

La automatización industrial se organiza en distintos niveles jerárquicos que permiten estructurar y gestionar las funciones del sistema de forma eficiente. Esta organización facilita el diseño, control, monitoreo y optimización de los procesos productivos. Los niveles principales en una arquitectura de automatización típica son el nivel de campo, control y supervisión, cada uno con funciones, tecnologías y responsabilidades específicas, tal como muestra la Figura 2 (SICMA21, 2024).

Figura 2.

Pirámide de automatización.



Nota. Elaborado por INCIBE, 2023.

- **Campo:** El nivel de campo es donde yacen los instrumentos o dispositivos físicos que interactúan directamente con el proceso industrial. Aquí se incluyen sensores, actuadores, interruptores, transductores y válvulas, los cuales son responsables de detectar variables físicas y ejecutar acciones sobre el proceso. (SMC International Training, 2023)

- **Control:** Este nivel es responsable de toma decisiones automáticas basadas en la información proveniente del nivel de campo, en este nivel se encuentran los controladores lógicos programables (PLC) y otros dispositivos inteligentes. Su función principal es procesar las señales de entradas provenientes de los sensores y mediante una lógica programada general señales de salida que controlan a los actuadores. (Rangel & Vega, 2020)
- **Supervisión:** Este está orientado al monitoreo, visualización y adquisición de datos y gestión de alarmas, está conformado por sistemas SCADA, interfaces HMI que permiten a los operadores humanos interactuar con el proceso de manera gráfica en tiempo real. (Lliguicota & Jiménez, 2024)

2.1.3. Importancia en procesos industriales

La automatización industrial desempeña un papel fundamental en la modernización y optimización de los procesos productivos, permitiendo alcanzar altos niveles de eficiencia, precisión y seguridad. Su implementación en entornos industriales facilita el control continuo de variables críticas como la velocidad, la presión, la temperatura y el caudal, garantizando la calidad del producto final y reduciendo la intervención humana en tareas repetitivas o de alto riesgo. Además, permite una supervisión centralizada, el registro histórico de datos operativos y la detección temprana de fallas, lo cual se traduce en una reducción significativa de tiempos muertos, desperdicios y costos operativos. En sectores como la industria alimentaria, farmacéutica, petroquímica o acuícola, donde la trazabilidad y la continuidad del proceso son esenciales, la automatización se convierte en una herramienta estratégica para aumentar la competitividad y cumplir con normativas técnicas y de

seguridad.

2.2. Controladores lógicos programables (PLC)

Un autómata programable industrial (API) es un equipo eléctrico que puede programarse mediante lenguajes no informáticos, pensado para tener la finalidad de controlar de forma secuencial y en tiempo real dentro de un entorno industrial.

2.2.1. Definición de PLC

El autómata programable, comúnmente conocido como PLC, como se muestra en la Figura 3, se trata de un computador especializado tanto en el software como en el hardware (Petruzella, 2020). En lo que respecta al software, este se desarrolla en un lenguaje que ha sido diseñado específicamente para facilitar la elaboración del programa que implemente los algoritmos de control para procesos secuenciales. Dicho algoritmo de control es programado y ejecutado de forma periódica en distintos intervalos suficientemente cortos como para permitir controlar los procesos en tiempo real. (Nuñez, Zabala, Sánchez, & Chamba, 2025) Adicionalmente, estos programas son flexibles y pueden adaptarse fácilmente en la producción. En el hardware, porque utiliza componentes robustos que soportan condiciones de trabajo adversas, como las que se dan en ambientes industriales refiriéndose a polvo, temperatura, vibraciones, etc. Estos equipos fueron elaborados con el propósito de garantizar gran fiabilidad y resistencia, asegurando una operación constante y estable en entornos industriales exigentes. Gracias a estas características, los PLC son una pieza fundamental en la cuarta revolución industrial posibilitar la digitalización de procesos, la mejora de recursos y el incremento del rendimiento operativo en las plantas

de producción.

Figura 3.

PLC siemens S7-1200.



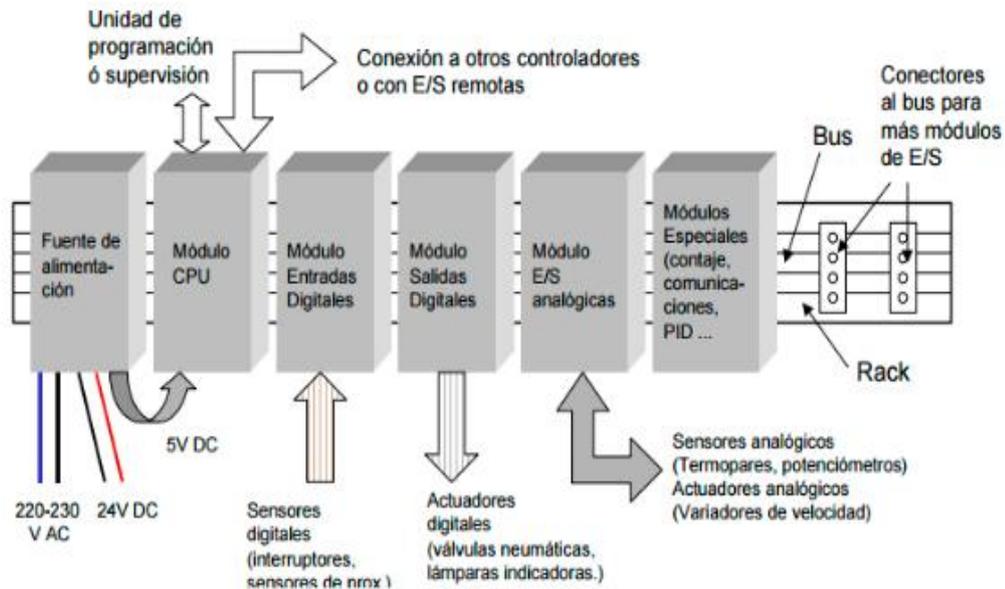
Nota. Elaborado por Carbajal Morán, H., 2021.

2.2.2. Arquitectura básica y funcionamiento

Internamente el PLC tiene una arquitectura del cual consta de procesador que es el que ejecuta el programa almacenado en la memoria de programa. La memoria de programa y la de datos esta físicamente separadas, constituyendo una arquitectura tipo Harvard. Además, la memoria de datos esta separa en dos tipos, que en la Figura 4 se denomina memoria de datos y memoria interna. Esta última se utiliza para almacenar los valores de las señales de entrada y salida, por lo que están conectadas con módulos de expansión y embebidos de entradas y salidas, que son los elementos de interfaz donde se conectan los sensores y actuadores del proceso. También dispone de periféricos para comunicar con otros dispositivos, como pantallas táctiles, ordenadores u otros autómatas. (Ajiga, 2024).

Figura 4.

Arquitectura de un PLC.



Nota. Elaborado por César Jimenez Jorquera, 2021.

2.2.3. Ciclo de Scan del PLC.

Como se ilustra en la Figura 5, el ciclo de escaneo del PLC es la actividad continua mediante el cual el controlador ejecuta su programa de control. Este ciclo consta de varias fases: primero, el PLC lee el estado de las entradas (sensores, botones o cualquier señal analógica) donde la lógica ejecutada en el CPU refleja las condiciones de las entradas externas para ejecutar la programación definido por el usuario. Luego, actualiza los dispositivos de salida (analógicos o digitales) basados en el estado lógico. Y Finalmente, el PLC realiza tareas como el mantenimiento de diagnósticos internos y comunican al CPU el estado de programación como encendido o apagado (Domínguez, 2022). Este ciclo se repite de forma de bucle cerrado y a muy alta velocidad, generalmente el tiempo de respuesta en milisegundos, lo que da paso a una respuesta en tiempo real a los cambios del proceso. El

tiempo del ciclo de escaneo es importante en aplicaciones donde se requiere alta precisión o sincronización entre dispositivos, y su duración puede depender por la complejidad del programa o el número de señales que se están procesando.

Figura 5.

Ciclo de operación de un PLC.



Nota. Elaborado por Carlos Abner, 2023.

2.2.4. Señales y módulos de expansión

En los sistemas de automatización basados en PLC Siemens, la correcta selección de los tipos de señales y módulos de expansión es esencial para garantizar una integración eficiente con los dispositivos de campo. Las señales pueden clasificarse principalmente en digitales (todo/nada) y analógicas (continuas), las primeras utilizadas comúnmente para detectar estados de presencia, posición o encendido/apagado, y las segundas para medir variables físicas como temperatura, presión o nivel. Los controladores Siemens de las familias S7-1200 y S7-1500, en particular, contienen módulos

de entrada y salida (I/O) integrados como modulares, que se diseñan opcionalmente para que pueden ampliarse mediante módulos de expansión que se conectan lateralmente. Al extender estos dispositivos, pueden adaptarse al sistema conjunto con los requerimientos del proyecto sin necesidad de cambiar el CPU, pudiendo añadir canales digitales, canales analógicos, módulos de comunicación o incluso módulos de seguridad. Esta capacidad de expansión es una característica que nos proporciona el escaler del sistema, de modo que permanece confiable y eficiente dentro del entorno industrial. (Siemens, 2022)

2.2.5. Lenguajes de programación

Los controladores poseen diferentes lenguajes de programación, para este trabajo de titulación se utilizarán dos de los más usados, Ladder por su facilidad y función de gráficos secuencial por sus siglas en ingles SFC debido a la secuencia de batch (Ordoñez Lara & Yépez Mendez, 2023).

- **Ladder (ESCALERA)**

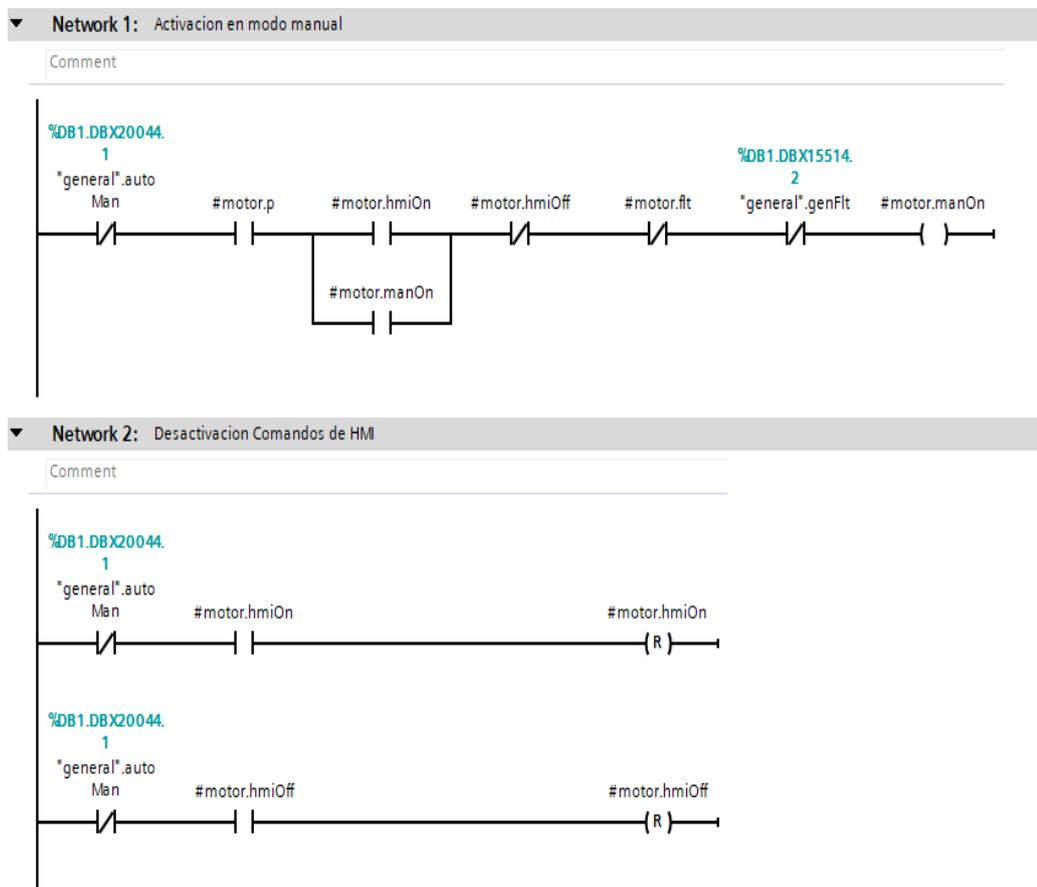
El lenguaje Ladder, también conocido como diagrama de escalera, es un lenguaje de programación gráfico utilizado ampliamente en la automatización industrial para programar PLC (Sosa, Su estructura se asemeja a una escalera, donde las secciones horizontales representan operaciones lógicas o "peldaños", y los límites verticales simulan los rieles de una red eléctrica convencional. (Albarracín Parra, 2023). Cada nivel contiene una condición de entrada y una acción de salida, siguiendo una lógica que se basa en los esquemas de control por relés electromecánicos. Este lenguaje fue utilizado como una herramienta de transición para facilitar al personal técnico acostumbrado a pensar en la lógica de contactores, relés y

temporizadores físicos. Por esta razón, sus símbolos —como contactos normalmente abiertos o cerrados, bobinas, temporizadores o contadores— imitan los dispositivos de control eléctrico tradicionales.

El lenguaje Ladder, ilustrado en la Figura 6, se encuentra estandarizado en la norma IEC 61131-3, que regula los lenguajes de programación de PLC. Su uso es común en tareas de control secuencial, arranques de motores, sistemas de enclavamiento (interlocks), activación de alarmas y control de procesos discretos. Gracias a su visualización intuitiva, permite una fácil comprensión del flujo de control, facilitando tanto el desarrollo como el mantenimiento del sistema. (Lliguicota & Jiménez, 2024)

Figura 6.

Ejemplo de programación Ladder.



Nota. Adaptado de Jacob Biedulski, 2024.

- **SFC (Sequential Function Chart)**

El lenguaje SFC o Diagrama de Funciones Secuenciales, es un lenguaje de programación gráfico utilizado en la automatización industrial para representar procesos que deben ejecutarse de forma ordenada y secuencial. Forma parte de los cinco lenguajes estandarizados por la norma IEC 61131-3, y es especialmente útil para describir máquinas de estado, secuencias de producción, líneas por etapas y procesos cíclicos. Su aplicación es común en sistemas donde la seguridad y la secuencialidad son esenciales para el adecuado funcionamiento de la planta (Ayala, Rivera, & Valle, 2021) .

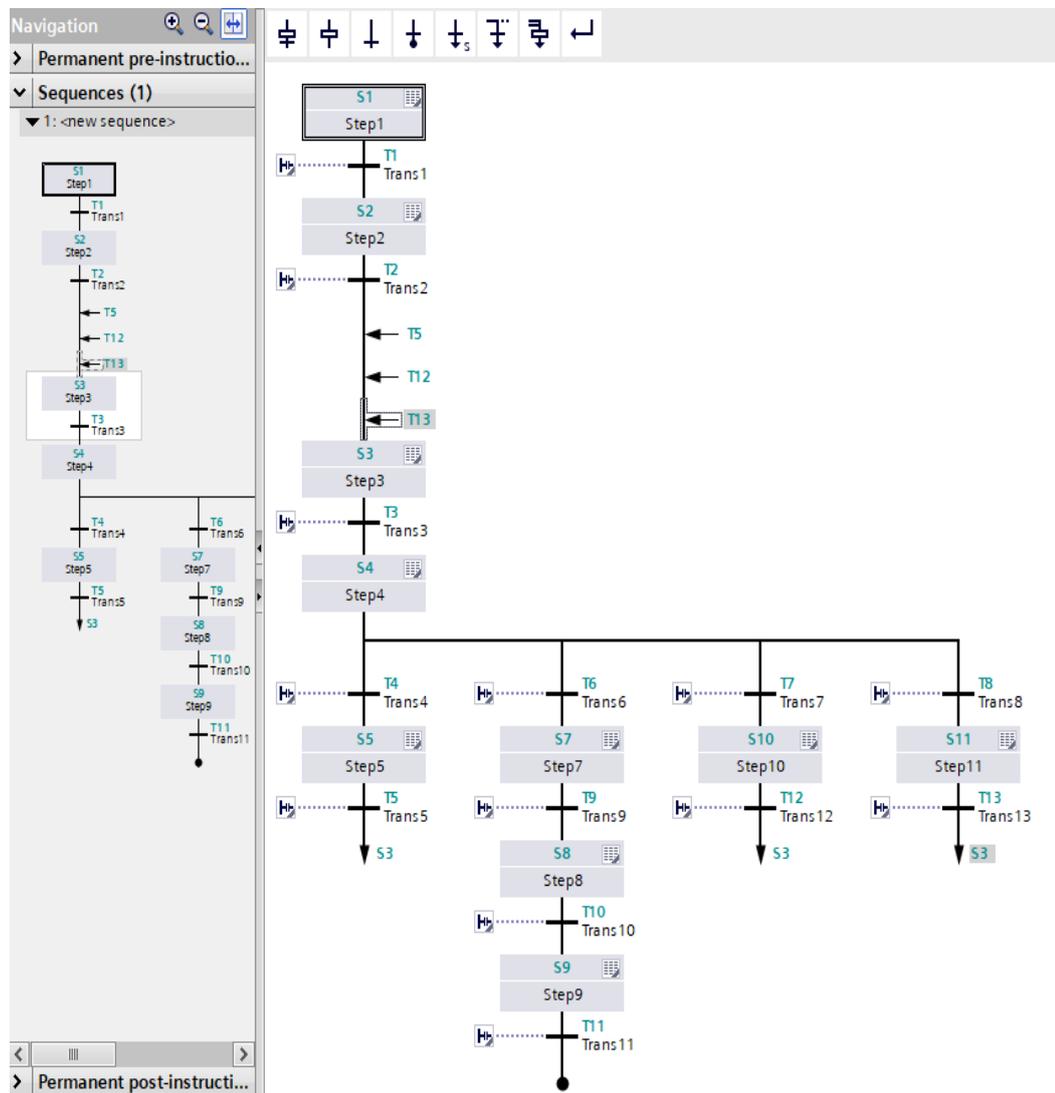
- SFC se basa en la división del programa en pasos (steps) y transiciones.
- Cada paso representa un estado o fase del proceso, y puede contener acciones asociadas que se ejecutan mientras dicho paso esté activo.
- Las transiciones son condiciones lógicas que determinan el paso de un estado al siguiente.

Este razonamiento hace posible la comprensión, diseño y estructuración de procesos complejos que requieren varias etapas y condiciones definidas para el avance, como el llenado de tanques, la carga o descarga de productos, o el arranque progresivo de los equipos.

Además, el lenguaje SFC, mostrado en la Figura 7, permite estructurar, claridad y fácil mantenimiento, ya que el flujo del proceso se representa como un diagrama. Se utiliza típicamente en conjunto con otros lenguajes como Ladder, Diagrama de bloques o Texto Estructurado para especificar las acciones internas de cada paso. De este modo, se obtiene un diseño de programación más robusto que integra la representación gráfica del SFC con la precisión de otros lenguajes (Baldeón Casquete, 2021).

Figura 7.

Ejemplo de programación SFC.



Nota. Adaptado de Siemens, 2021.

2.2.6. Aplicaciones

Los controladores lógicos programables tienen una amplia aplicación a nivel industrial, debido a su robustez para soportar condiciones adversas y a su funcionalidad de recibir y procesar señales físicas de campo y el respectivo control del mismo proceso de producción, también por su facilidad de conexión con diferentes tipos de sistemas a través de comunicación Ethernet, por lo cual se puede realizar un control muy completo con estos dispositivos.

2.2.7. Software Tia portal como entorno de automatización

El software TIA Portal (Totally Integrated Automation Portal) es una plataforma de software desarrollada por Siemens para la programación, la configuración, la simulación y el monitoreo de sistemas de automatización industrial. Este entorno unifica en una sola interfaz todas las herramientas necesarias para diseñar y poner en marcha soluciones de control, ofreciendo integración total entre los diferentes dispositivos Siemens, al igual que el (S7-1200 o el S7-1500), HMI (Human Machine Interface), variadores de velocidad, redes de comunicación y sistemas SCADA (Pin López & Criollo Zárata, 2024).

TIA Portal se basa en el concepto de automatización integrada, lo que significa que desde un único proyecto se pueden gestionar todos los elementos involucrados en el proceso de control, eliminando la necesidad de utilizar múltiples aplicaciones independientes. Este enfoque reduce los errores de configuración, simplifica el flujo de trabajo del ingeniero y mejora la eficiencia en el desarrollo de proyectos.

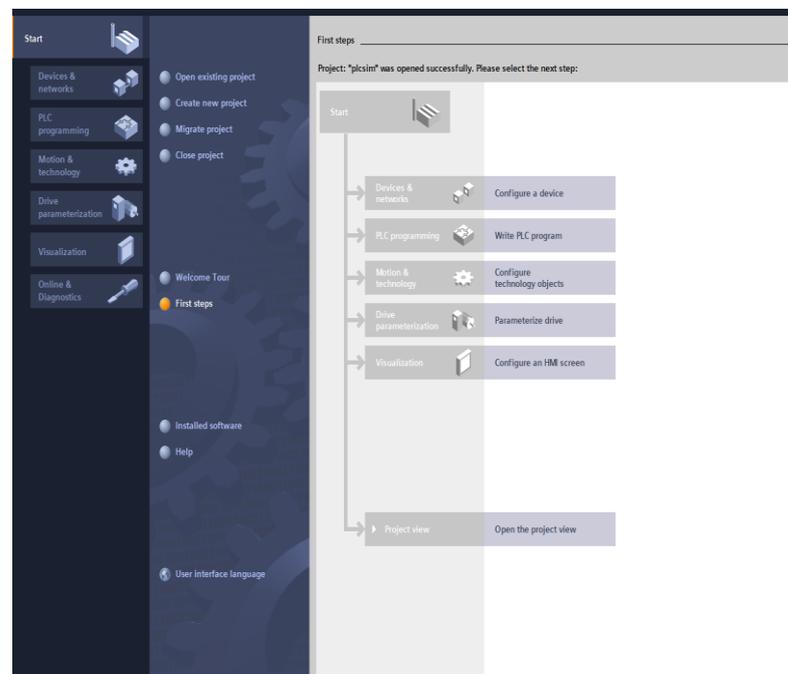
Tiene numerosas características, entre ellas destaca una interfaz gráfica intuitiva de fácil entendimiento, que no requiere de hardware físico, cientos de funciones integradas, simulación de la lógica del programa sin necesidad de un PLC y posibilidad de trabajar con la mayoría de los lenguajes de programación definidos por el estándar IEC 61131-3, como el Diagrama Ladder (LD), Texto Estructurado (ST) y Diagrama de Bloques de Función.

Asimismo, TIA Portal permite la creación de alarmas, el manejo de variables globales, el diseño de paneles HMI personalizados y la configuración de enlaces de comunicación industrial como PROFINET, Profibus o Modbus TCP (Siemens, 2022) .

El uso de TIA Portal, presentado en la Figura 8, es especialmente relevante en proyectos que requieren altos niveles de confiabilidad, trazabilidad y estandarización, como es el caso de las líneas de producción automatizadas en la industria acuícola. En este tipo de entornos, su integración con sistemas SCADA y la posibilidad de aplicar lógicas de permisivos, enclavamientos y secuencias de arranque de manera gráfica lo convierten en una herramienta poderosa para ingenieros de control.

Figura 8.

Software TIA portal.



Nota. Elaborado por Hosseinzadeh, 2024.

2.3. Sistema SCADA

Se denomina al sistema SCADA (Supervisory Control and Data Acquisition) como herramientas fundamentales en la automatización moderna, ya que permiten supervisar, controlar y analizar procesos industriales en tiempo real. Estos sistemas están diseñados para gestionar una gran cantidad de variables distribuidas geográficamente o dentro de una

planta, proporcionando una interfaz visual amigable para los operadores y facilitando la toma de decisiones operativas y estratégicas. (Andrade & Mendez, 2022)

2.3.1. Definición y componentes

Un sistema SCADA es un conjunto de herramientas de hardware y software que permite la supervisión, control remoto, adquisición de datos y análisis de variables de proceso en sistemas industriales. Su objetivo principal es proporcionar una plataforma central desde la cual los operadores puedan monitorear el estado de los dispositivos, visualizar tendencias, recibir alertas y enviar comandos de control. (Vásquez & Herrera, 2022)

Los componentes principales de un sistema SCADA incluyen:

- Estaciones de operación (HMI/Interfaz Hombre-Máquina): Son los terminales gráficos que permiten al operador visualizar el estado del sistema, cambiar parámetros, reconocer alarmas y tomar decisiones en tiempo real.
- PLC o RTU (Unidades Terminales Remotas): Son dispositivos electrónicos ubicados en campo que recogen datos de sensores y ejecutan acciones sobre actuadores. Los PLC actuales se utilizan tanto para control como para la comunicación con el SCADA.
- Servidor SCADA: Es un protocolo de interconexión para toda la colección de equipos TI Y OT, para comunicar la información desde el campo a los clientes ubicados en la consola central, donde los operadores de planta logran el control operativo del proceso SCADA.
- Red de comunicación: Conecta todos los componentes del sistema; por un lado, el campo y los controladores, y por el otro lado, el centro de monitoreo

(sitio de trabajo).

2.3.2. Funciones principales

El sistema SCADA cumple diversas funciones que se convierten en herramientas esenciales dentro de un entorno industrial automatizado, entre las que destacan:

- **Adquisición de datos:** Recolecta señales de sensores y dispositivos en campo, ya sean analógicas o digitales, y las registra en tiempo real.
- **Supervisión y visualización:** Presenta gráficamente el estado de los equipos y procesos a través de pantallas HMI, donde se representan tanques, motores, válvulas, alarmas, etc.
- **Control remoto:** El sistema SCADA brinda la posibilidad de que los operadores gestionen la planta desde una sala de control central, lo que facilita enviar órdenes y ejecutar maniobras necesarias para la operación diaria sin tener que actuar directamente sobre cada equipo.
- **Gestión de alarmas:** El sistema tiene la capacidad de detectar condiciones anormales y avisar al personal mediante señales sonoras o visuales. Cada alerta queda registrada, lo que permite llevar un control de toda la planta,
- **Registro de datos históricos:** El SCADA guarda y registra información de los procesos a lo largo del tiempo (Gálvez & Ricaurte, 2022).
- **Generación de informes:** Automatiza la creación de reportes periódicos que resumen el desempeño del sistema, consumos, fallos o eficiencias.

2.3.3. Protocolo de comunicación

Los sistemas SCADA se comunican con los controladores PLC y otros dispositivos de campo mediante protocolos de comunicación industrial, que definen cómo se transmite la información entre dispositivos. Estos protocolos

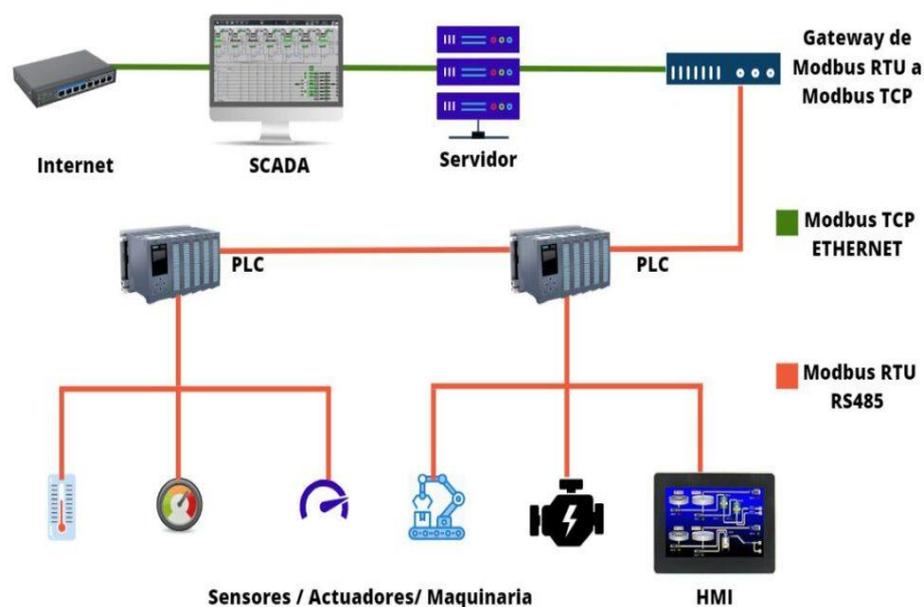
pueden ser propietarios o abiertos, y su elección depende del fabricante, la velocidad requerida, el nivel de seguridad y la compatibilidad del sistema.

Algunos de los protocolos más utilizados en entornos SCADA son:

- Modbus: Es uno de los protocolos más comunes, abierto, simple y ampliamente soportado. Actúa respecto al RS-485 modbus RTU o Ethernet Modbus TCP/IP, y su diagrama de transferencia de información como se muestra en la Figura 9 (Logicbus, 2024).

Figura 9.

Arquitectura Modbus.

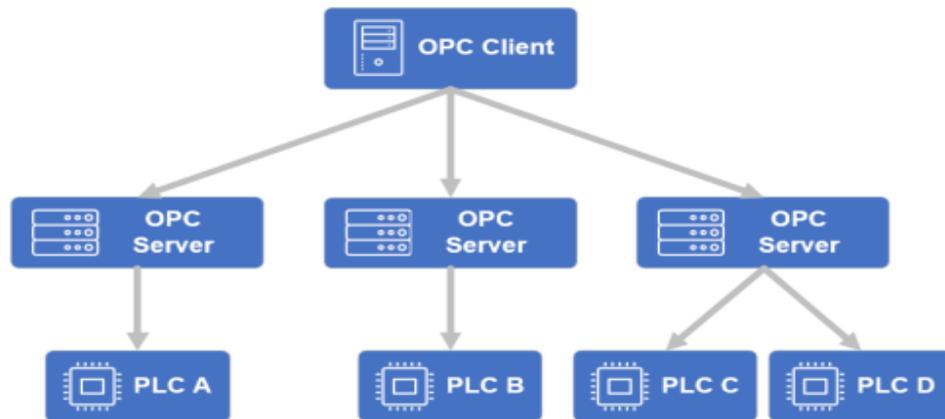


Nota. Elaborado por Martínez y Martínez, (2024).

- OPC (OLE for Process Control): Actúa como un puente estándar entre el software SCADA y los distintos dispositivos de automatización. Existen variantes modernas como OPC UA, que es independiente de plataforma, además permite una comunicación segura y ampliable entre equipos de diferentes fabricantes en el entorno industrial, tal como se aprecia en la Figura 10 (Rives Sempere, 2021).

Figura 10.

Arquitectura OPC.

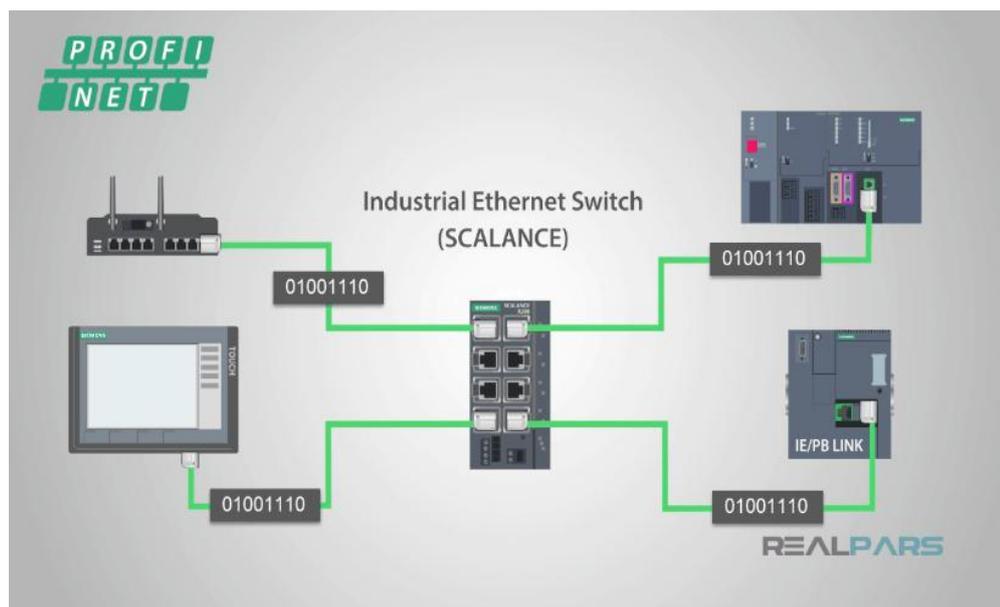


Nota. Elaborado por Kirvesoja, (2021).

- Profibus / Profinet: Protocolos desarrollados por Siemens, donde Profibus opera sobre buses seriales y Profinet sobre redes Ethernet, ofreciendo mayor velocidad y capacidades de diagnóstico avanzadas, cuyo esquema se aprecia en la Figura 11 (Naranjo Ninasunta & Salavarría Moreno, 2022).

Figura 11.

Arquitectura Profinet siemens.

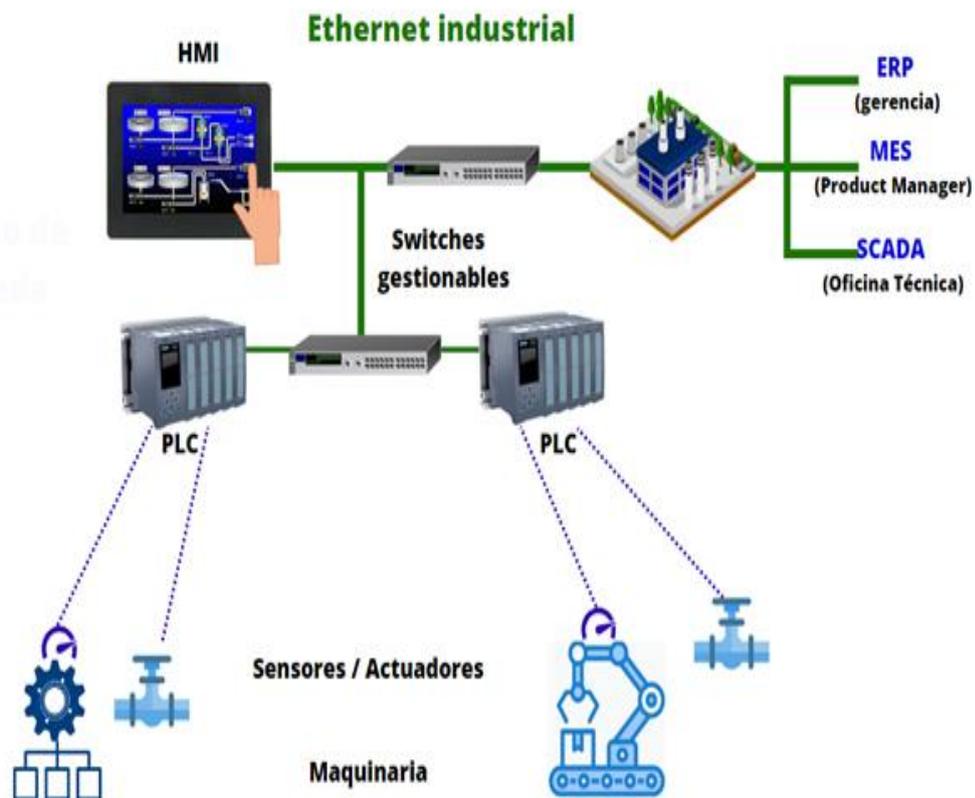


Nota. Elaborado por Vandieukhien, 2022.

- Ethernet/IP: Este protocolo empleado principalmente en equipos Rockwell Automation, posibilita el intercambio de datos sobre redes Ethernet convencionales mediante el modelo CIP (Common Industrial Protocol), cuyo funcionamiento se presenta en la Figura 12.

Figura 12.

Arquitectura Ethernet/IP.



Nota. Elaborado por Martínez, 2024.

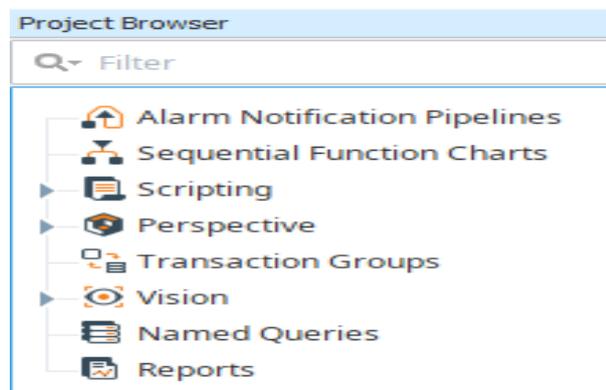
2.3.4. Ignition

Ignition es una plataforma de desarrollo de sistemas tanto industriales como de gestión, perteneciente a inductive automation. Esta plataforma permite una gran facilidad al establecer comunicación con equipos de campo de control PLC, en el cual se puede comunicar vía Ethernet IP, OPC o Modbus, adicional posee una infraestructura por módulos en los cuales se tiene la

flexibilidad de desarrollar pantallas de sistemas nativas en desktop y web con sus módulos de desarrollo Windows y perspective, para este proyecto se utilizará el módulo de Perspective. También posee módulos de script basadas en Python en el cual aporta en gran parte al desarrollo de funciones y tratamiento de información.

Figura 13.

Árbol de proyecto Ignition



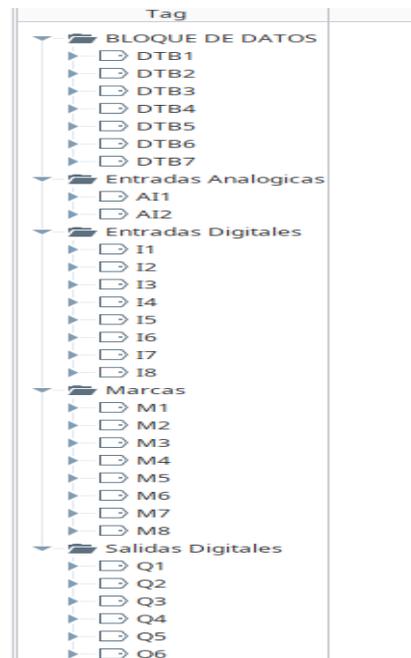
Nota. Elaboración propia.

En la Figura 13 se aprecia los módulos de desarrollo de Ignition, en el cual constan el módulo de scripting donde se pueden desarrollar funciones y librerías en código Python, el módulo de perspective consta en el desarrollo de sistemas basadas en plataforma web ya que su motor visual es html, el módulo visión al igual que perspective trata del desarrollo de sistemas con la diferencia de que este es nativo o embebido en desktop multiplataforma, y named queries donde se efectuaran las consultas a las bases de datos declaradas.

En la Figura 14 se aprecia la lista de tags. Un tag es una instancia de donde se puede almacenar información la cual puede tener diferentes tipos de orígenes, una de ellas es por comunicación OPC desde un PLC, es decir se pueden crear tags que referencien a información atreves del PLC.

Figura 14.

Navegador de tags.



Nota. Elaboración propia.

2.4. Permisivos de secuencia

2.4.1. Definición y tipos

El control de procesos aplicado en industrias se debe de aplicar una normativa o estructura para el encendido o activación de equipos eléctricos como motores, pistones, válvulas, ect.

La Figura 15 network 3 muestra una estructura programada en Ladder en el cual se condiciona la activación de un bit denominado .out perteneciente a un tipo de dato que hace referencia a un motor controlado por un variador de frecuencia. La activación debe cumplir con que no esté activo un bit de falla general y falla propia del motor *#genFault* y *#motor.flt*. Luego se evalúa el bit *#motor.ps* el cual hace referencia al permisivo de seguridad, al estar estos tres bits en serie son una condición necesaria para la activación del equipo.

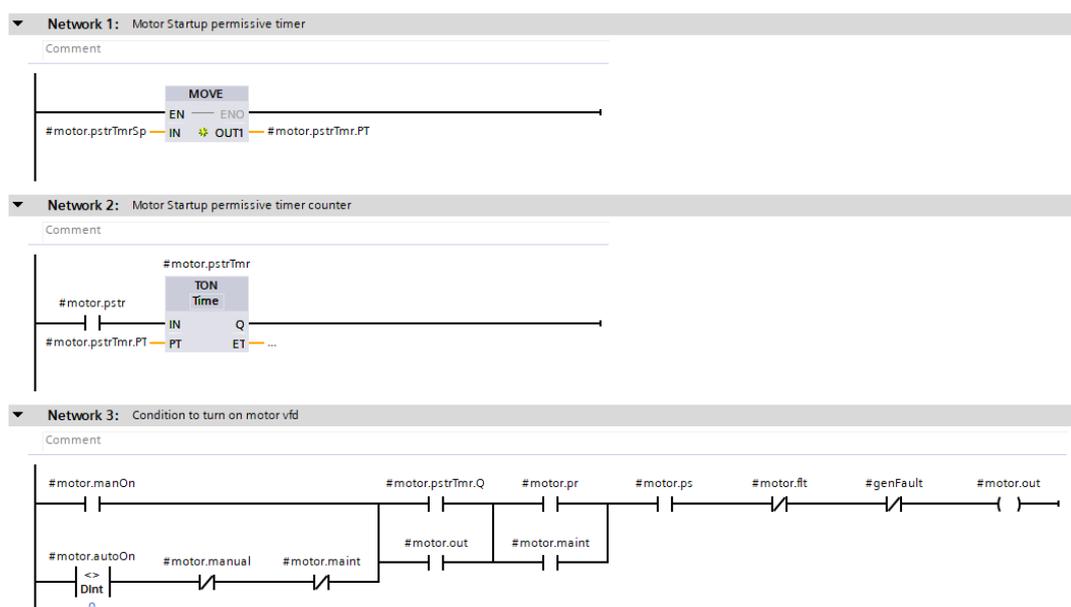
Posterior tenemos dos bits en paralelo *#motor.pr* y *motor.maint* el

cual indica que si el motor está en modo de operación de mantenimiento este debe de hacer caso omiso al bit del permisivo de proceso, caso opuesto el permisivo de proceso debe de cumplirse para poder encender el equipo. Luego se aprecian dos bits en paralelo el cual *#motor.out* cumple la función de enclavamiento de la estructura lógica, y *#motor.pstrTmr.Q* el cual consiste en un tiempo que debe cumplirse después de haberse cumplido que el bit de *#motor.pstr* se haya energizado, por lo cual se aprecia esta condición en el network 2.

Finalmente se visualiza una estructura lógica en el cual *#motor.manOn* hace referencia a la señal de encendido manual, la acción de que un usuario mande a encender mediante el SCADA el equipo. En paralelo al bit anterior se encuentra una estructura de encendido automático donde evalúa que el equipo no esté en modo de operación manual ni mantenimiento, previo a eso evalúa que el bit de encendido automático *#motor.autoOn* sea diferente a 0.

Figura 15.

Estructura lógica de encendido de un motor.



Nota. Elaboración propia.

En la estructura se aprecia la función que cumple los tres tipos de permisos de procesos (.pr), seguridad (.ps) y arranque (.pstr) en el cual se propone como estructura para implementaciones en industrias automatizadas, las lógicas de activación de estos bits deben de ser evaluadas según secuencias de producción y normas de seguridad para cada caso.

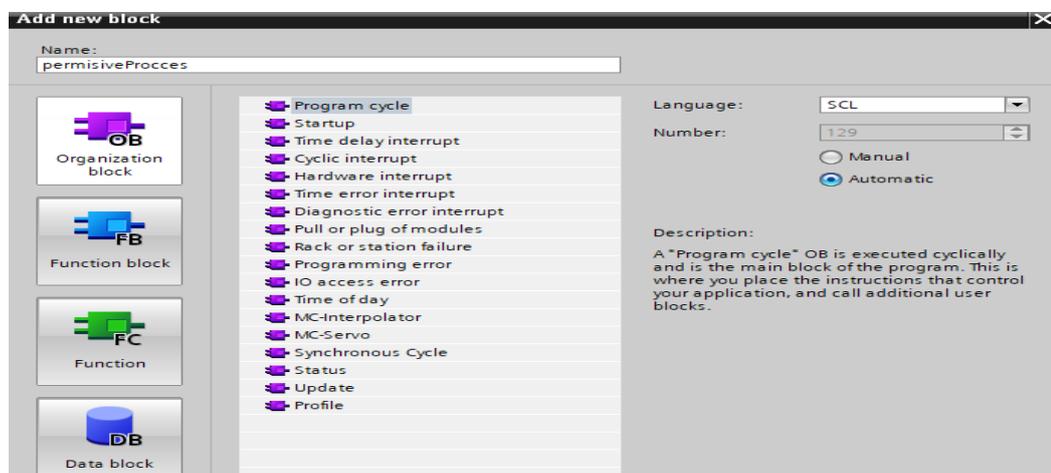
2.4.2. Lógica de funcionamiento

La estructura por desarrollar en la propuesta consta de dos secciones, en PLC y en SCADA, en PLC debido a que es el dispositivo que comanda los instrumentos y actuadores de campo, en SCADA por la visualización e información para el usuario.

En PLC debido a que se utilizara el entorno de siemens, se debe de utilizar bloques organizacionales de tipo ciclo de programa en lenguaje Ladder como se aprecia en la Figura 15, ahí se incluída la estructura compuesta por contactos sean abiertos o cerrados como condición para energizar una bobina, esta estructura debe de ser evaluada según la secuencia de producción de la planta para los permisos de proceso la Figura 16 y 17.

Figura 16.

Bloque de programa.



Nota. Elaboración propia.

Figura 17.

Estructura del permisivo de proceso.

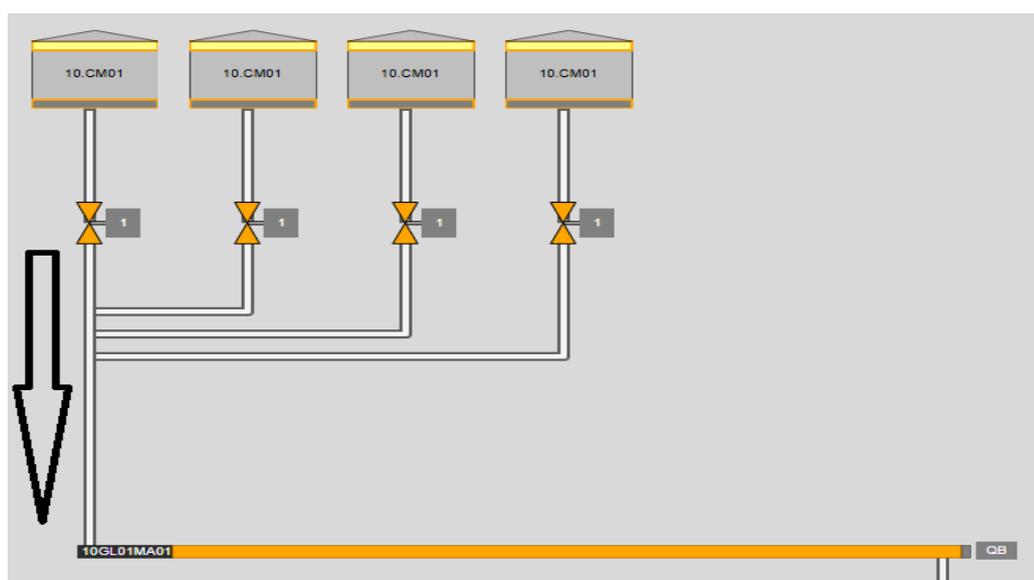


Nota. Elaboración propia.

En la Figura 16 se muestra que el tag en la bobina con nombre *10Qm01.pr* representa el permisivo de proceso de una válvula, esta se encuentra condicionada por el estado de encendido denominado *running* del transportador en un contactor abierto con tag *10GL01Ma01.running* debido a que cuando se abra la válvula el producto empezara a circular por gravedad, es requerimiento que el transportador por debajo de la válvula este encendido, de lo contrario el producto no circula como se aprecia en la Figura 17.

Figura 18.

Pantalla de secuencia intake.

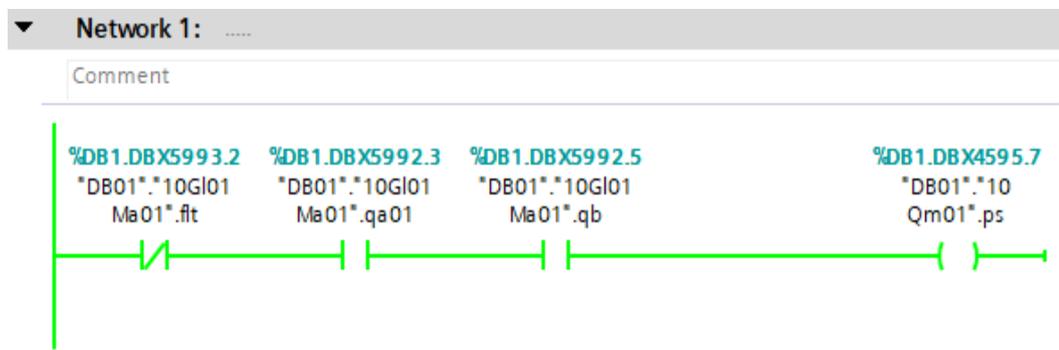


Nota. Elaboración propia.

Para el mismo caso, pero con permisivo de seguridad se contemplará que los equipos involucrados en secuencia tengan el seccionador de campo y guardamotor activos los cuales se trataran con los tags de nombre *.qb* y *.qa01* respectivamente, adicional de que el equipo no esté en falla con el tag *.flt*, la Figura 19 muestra esta estructura utilizando los tags del proceso anterior.

Figura 19.

Estructura del permisivo de seguridad.



Nota. Elaboración propia.

2.5. Instrumentación y señales de campo

2.5.1. Instrumentación

Todos los instrumentos de medición de campo como sensores de temperatura, velocidad, desbordamiento, pistones de válvulas son señales que las cuales pueden ser leídas por el PLC mediante los puertos de entradas digitales o analógicas dependiendo del tipo de señal que se requiera medir, de igual forma para el control de actuadores se manipulan señales de salidas analógicas o digitales con el objetivo de activar contactores o enviar un valor de velocidad a un variador de frecuencia o apertura de una válvula proporcional las cuales requieren de un valor entero o real los cuales son denominados señales analógicas.

Estas señales deben de ser incluidas en la programación del PLC como imágenes de entradas o salidas, las cuales se desarrollan en lenguaje scl por su facilidad de automatización. Esta metodología posibilita organizar de manera nítida los datos del procedimiento.

Figura 20.

Lenguaje SCL de siemens para señales de entrada de campo.

```
1 // input Image zso
2 IF NOT ("DB01"."10Qm01".sim) AND "I001" THEN
3     "DB01"."10Qm01".zso := TRUE;
4 ELSIF "DB01"."10Qm01".sim AND "DB01"."10Qm01".outOpen THEN
5     "DB01"."10Qm01".zso := TRUE;
6 ELSE
7     "DB01"."10Qm01".zso := FALSE;
8 END_IF;
9
10 // input Image zsc
11 IF NOT ("DB01"."10Qm01".sim) AND "I002" THEN
12     "DB01"."10Qm01".zsc := TRUE;
13 ELSIF "DB01"."10Qm01".sim AND "DB01"."10Qm01".outClose THEN
14     "DB01"."10Qm01".zsc := TRUE;
15 ELSE
16     "DB01"."10Qm01".zsc := FALSE;
17 END_IF;
```

Nota. Elaboración propia.

En la Figura 20 se aprecia que el estado open y close de una válvula denominada 10Qm01 contendrán en los tags .zso y .zsc respectivamente, los

cuales están condicionados a que si el equipo está en simulación el estado de la válvula no sea leído por la señal de campo, pero mientras no este simulada esta será leída mediante la entrada digital 1 y 2 respectivamente.

Figura 21.

Imagen de entrada analógica.

```
// input Image flt speed and current
IF NOT ("DB01"."10Gm01Ma01".sim) THEN
    "DB01"."10Gm01Ma01".speedActual := ("AI001") * (1);
    "DB01"."10Gm01Ma01".outputCurrent := ("AI002") * (10);
ELSE
    "DB01"."10Gm01Ma01".speedActual := 0;
    "DB01"."10Gm01Ma01".outputCurrent := 0;
END_IF;
```

Nota. Elaboración propia.

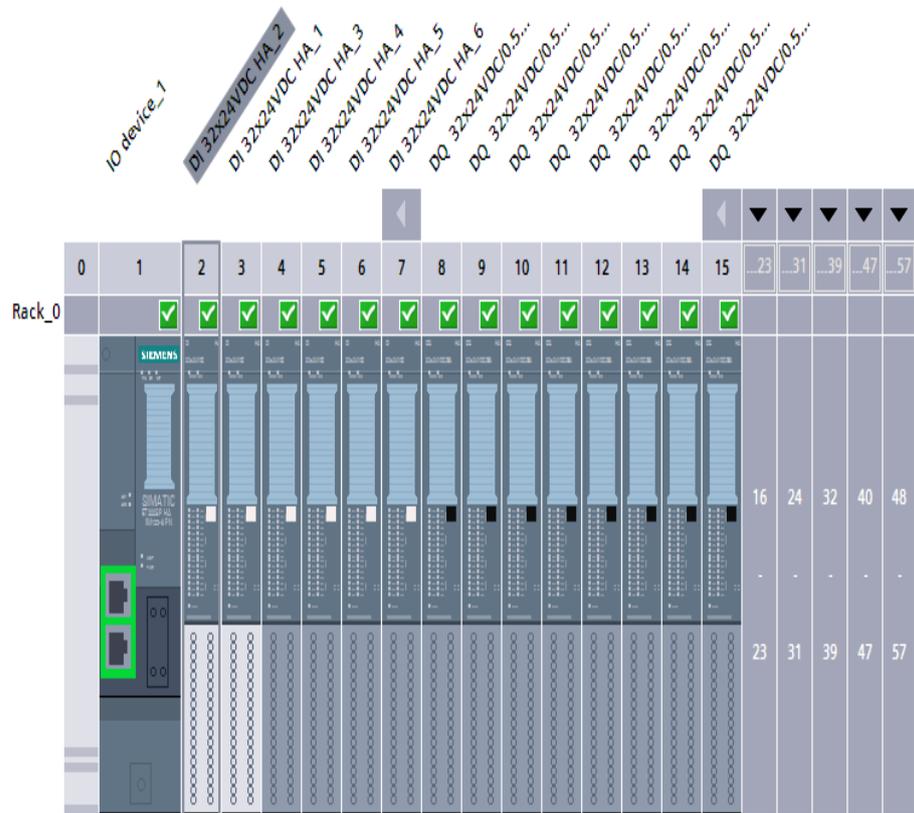
Para las señales analógicas, la estructura de lectura se muestra en la Figura 21, la cual representa la velocidad en la que se encuentra un variador de frecuencia y la corriente del equipo, datos que son recibidas mediante las entradas analógicas 1 y 2 del sistema.

2.5.2. Módulos de expansión de señales de campo

Los PLC vienen con un número limitado de señales de entrada y salida, por lo cual es necesaria la incorporación de módulos de expansión denominados ET-200, estos módulos poseen ranuras donde se instala un dispositivo llamado point I/O el cual físicamente posee un número determinado de borneras para la conexión de equipos de campo, la Figura 22 muestra un ET-200 con 13 points conectados.

Figura 22.

Módulo de expansión ET-200.



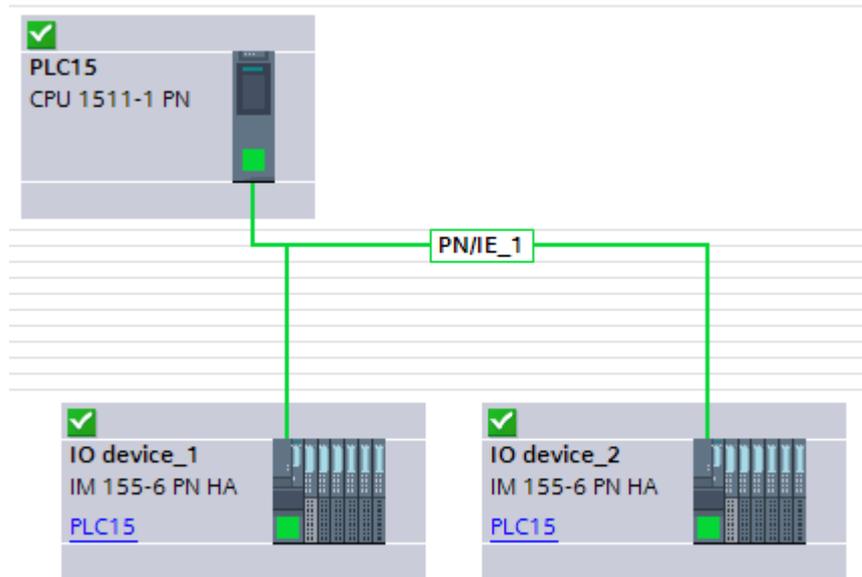
32x24VDC HA_2 [DI32 X24VDC HA]					
General	IO tags	System constants	Texts		
Name	Type	Address	Tag table	Comment	
I001	Bool	%I0.0	et200spHa011		
I002	Bool	%I0.1	et200spHa011		
I003	Bool	%I0.2	et200spHa011		
I004	Bool	%I0.3	et200spHa011		
I005	Bool	%I0.4	et200spHa011		
I006	Bool	%I0.5	et200spHa011		
I007	Bool	%I0.6	et200spHa011		

Nota. Elaboración propia.

El módulo ET-200 puede comunicarse con el PLC mediante profinet o profibus como se aprecia en la Figura 23, luego de establecer la conexión se procede a registrar los tags provenientes de campo como entradas o salidas según el tipo de señal.

Figura 23.

Red de PLC con módulos de expansión.



Nota. Elaboración propia.

CAPÍTULO 3: ESTUDIO DE PARÁMETROS DEL PROCESO EN LA PRODUCCIÓN DE ALIMENTO ACUÍCOLA

3.1. Secuencia de producción

Para llegar a elaborar el grano del balanceado denominado pellet se requiere de un proceso de mezcla y cocción de ingredientes como soya, trigo, aceites, entre otros. Este proceso se divide en varias las cuales son comprendidas desde el almacenamiento de la materia prima, distribución, procesamiento, mezcla, cocción y empaque, las cuales se explican a continuación:

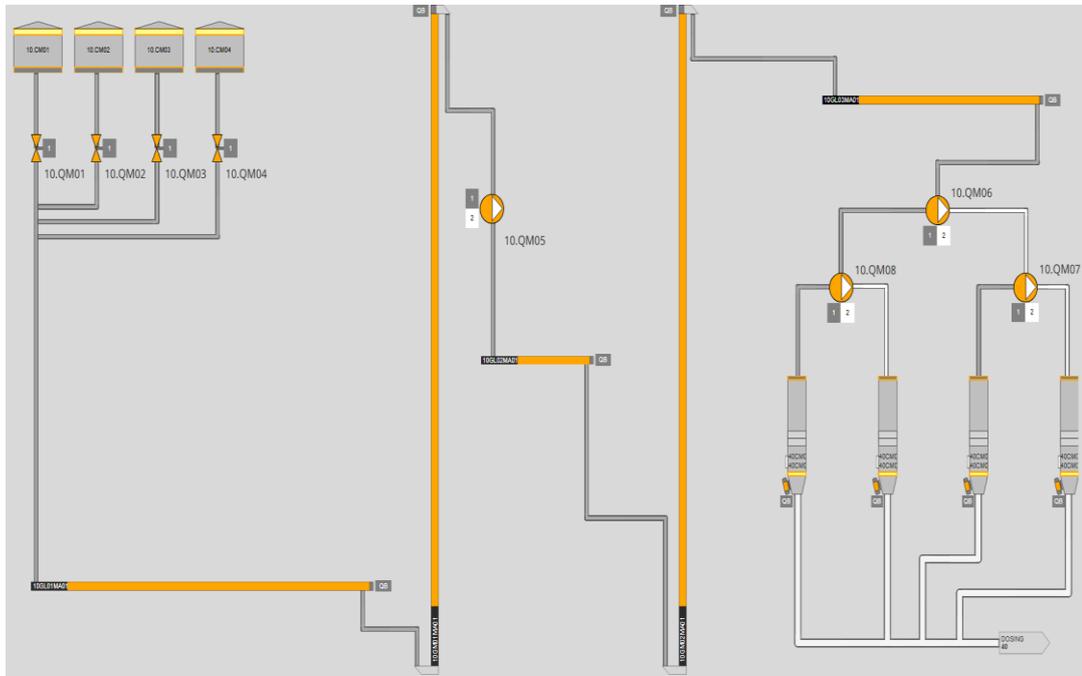
3.1.1. Intake

El proceso de producción del pellet comienza por la recepción y almacenamiento de la materia prima, esta área se denomina intake el cual es un término en inglés. Esta consta de una serie de equipos de campo como basculas para el pesaje de los camiones que traen el material a recibir, posterior al pesaje la materia es almacenada o recibida en bodegas planas en el caso de que el material sea soya o trigo, para el caso de líquidos se almacenan en tanques con resistencias térmicas debido a que los líquidos típicos son aceites en el cual si no se mantiene en movimiento y caliente se vuelve en un material gelatinoso y se echa a perder, una vez que el operador dispone a arrancar la producción el área de intake tiene como objetivo movilizar o entregar las materias solicitadas a las tolvas de dosificación mediante transportadores y elevadores los cuales pueden ser motores controlados mediante variadores de frecuencia o un arranque directo tipo contactor-guardamotor, también se requiere de incluir válvulas y sensores de nivel siendo estos ubicados en la sección superior e inferior de las tolvas y

silos para tener una medición de la materia y producto del cual se va a requerir para el proceso.

Figura 24.

Área de intake.



Nota. Elaboración propia.

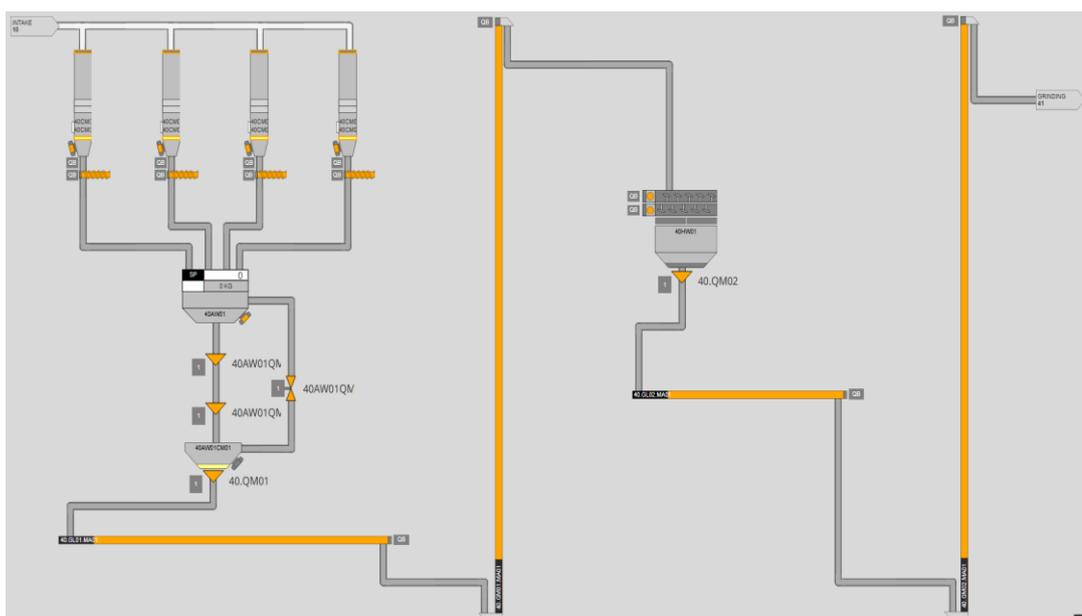
La Figura 24 muestra el diagrama del área de intake, en el cual se pasa a través de una serie de equipos previamente mencionados. Para este ejemplo se constan de cuatro silos de almacenamiento denominados 10CM01, 10CM02, 10CM03, 10CM04, en el cual el numero “10” indica el área al que pertenece, “CM” es el termino que se le da a las tolvas, silos y lugares de almacenamiento, cada silo posee sensores de nivel alto e inferior. Se posee tres transportadores y dos elevadores controlados por sus respectivos variadores de frecuencia, finalmente se observa que hay un juego de cuatro válvulas para permitir el paso de la materia prima desde los silos hasta el transportador inicial, luego un grupo de válvulas en el cual cumplen la función de direccionar el material al silo correspondiente del área de dosificación.

3.1.2. Dosificación y mezclado

El área de dosificación es encargada de proporcionar según una receta la cantidad correspondiente de cada materia prima requerida para la elaboración del pellet en relación a las toneladas que se deseen producir, este proceso inicia con el almacenamiento en los silos de lo que se haya recibido de intake, posterior se presenta una balanza en el cual se realizara el pesaje de cada ingrediente dosificado según la receta, luego de que se haya dosificado en el porcentaje correspondiente cada ingrediente este pasara a una tolva de descarga mediante un juego de válvulas de transporte y compensación de presión, luego los ingredientes serán llevados mediante un transportador y elevador a una mezcladora la cual se encarga de realizar una mezcla correcta de todos los ingredientes, luego del respectivo mezclado el producto es llevado al área de molienda y pulverizado mediante otro juego de transportadores y elevadores, la Figura 25 muestra el área indicada.

Figura 25.

Área de dosificación.



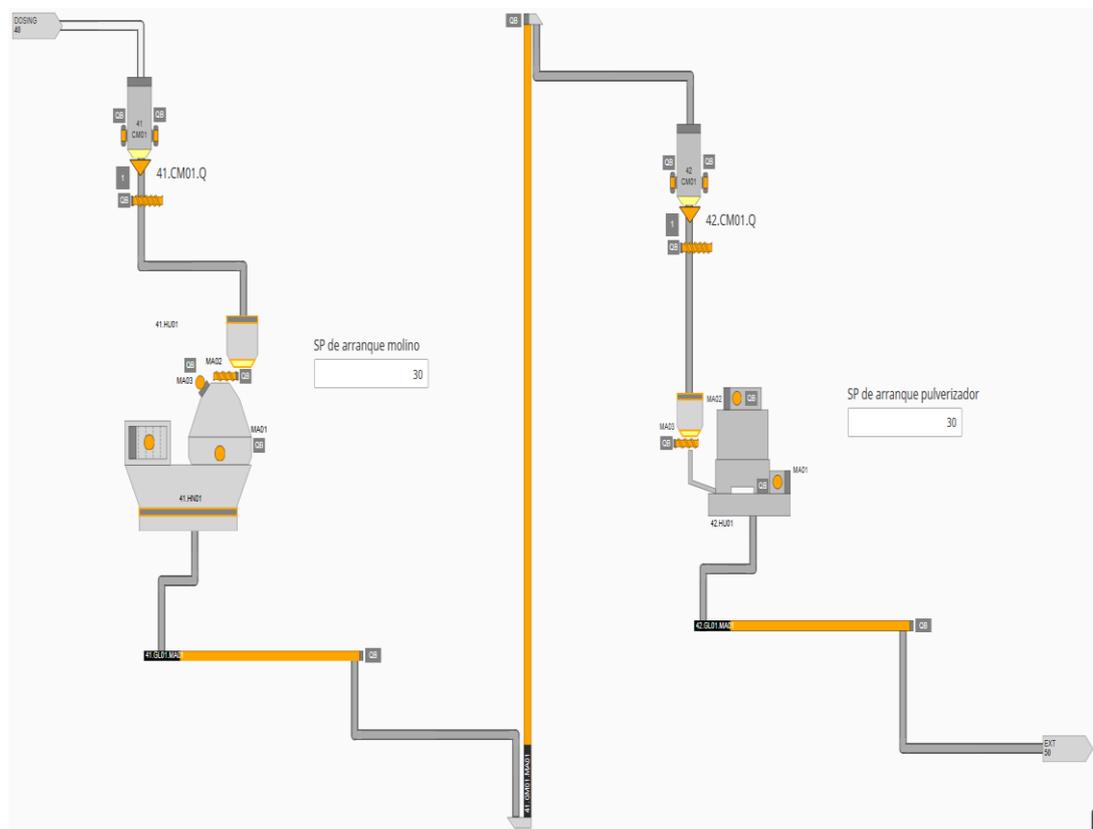
Nota. Elaboración propia.

3.1.3. Molienda y pulverizado

En el área de molienda y pulverizado, como se observa en la Figura 26, se recibe el producto de forma continua y como el nombre lo indica esta área se encarga de moler y pulverizar todo grano con el objetivo de que todo el producto tenga un tamaño similar previo al proceso de extrusión, este proceso es vital debido a que en esta área se encuentran dos de tres de los equipos con motores eléctricos más grandes y potentes, por lo cual en esta área se declaran los primeros permisivos de arranque para salvaguardar la estabilidad de la red eléctrica, evitar multas por el proveedor de energía y evitar un arranque simultaneo de todos los equipos produciendo un alto consumo eléctrico.

Figura 26.

Área de molienda y pulverización.



Nota. Elaboración propia.

3.1.4. Extrusión

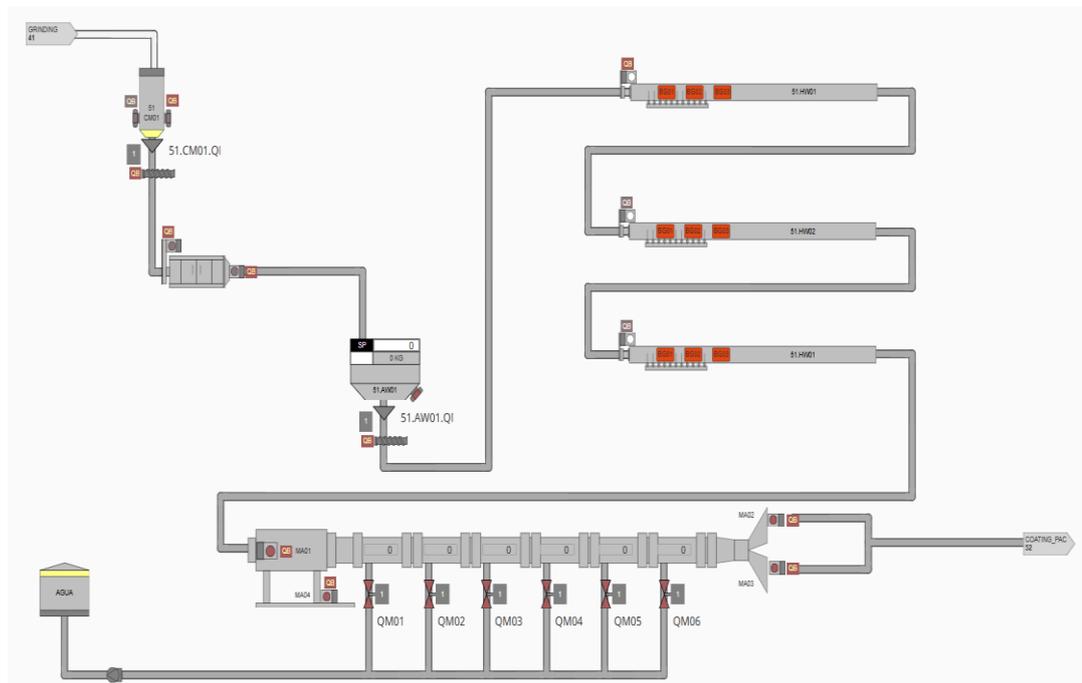
El área de extrusión está compuesta por una diversidad de equipos que cumplen roles específicos, iniciando con la tolva de recepción de producto molido, una vez que la tolva llega a nivel alto se debe de abrir la válvula 51CM01QM01 en conjunto con el encendido del tornillo dosificador con el objetivo de enviar producto a la zaranda, posterior a eso el producto es pesado en una balanza, la cual se debe de mantener en un margen específico de producto ya que su dosificación es continua.

La balanza mantiene el sistema con producto, entregando de forma continua un flujo masico a los acondicionadores los cuales compactaran el pellet, estos acondicionadores poseen unas compuertas las cuales son tomadas como permisivos de seguridad.

Luego del proceso de acondicionamiento el pellet entra en cocción dentro de la extrusora la cual le da la forma final y respectivo cocinado. Finalmente, las cuchillas cortan el grano según el molde al que estén acopladas, la extrusora es un equipo vital el cual requiere de ciertos cuidados para no estropearse, por lo cual debe de estar acompañada de un sistema de enfriamiento por agua, de esta forma las chaquetas que cocinan al pellet pueden enfriarse en caso de sobrepasar su temperatura apropiada. Después del corte, el pellet es descargado hacia el sistema de transporte, donde se dirige al secado para continuar con el proceso productivo, como se muestra en la Figura 27. Posteriormente, el producto avanza hacia el sistema de secado donde inicia la etapa de reducción de humedad para su conservación. Este flujo continuo permite que la extrusión mantenga una operación constante sin interrupciones en la línea.

Figura 27.

Área de extrusión.



Nota. Elaboración propia.

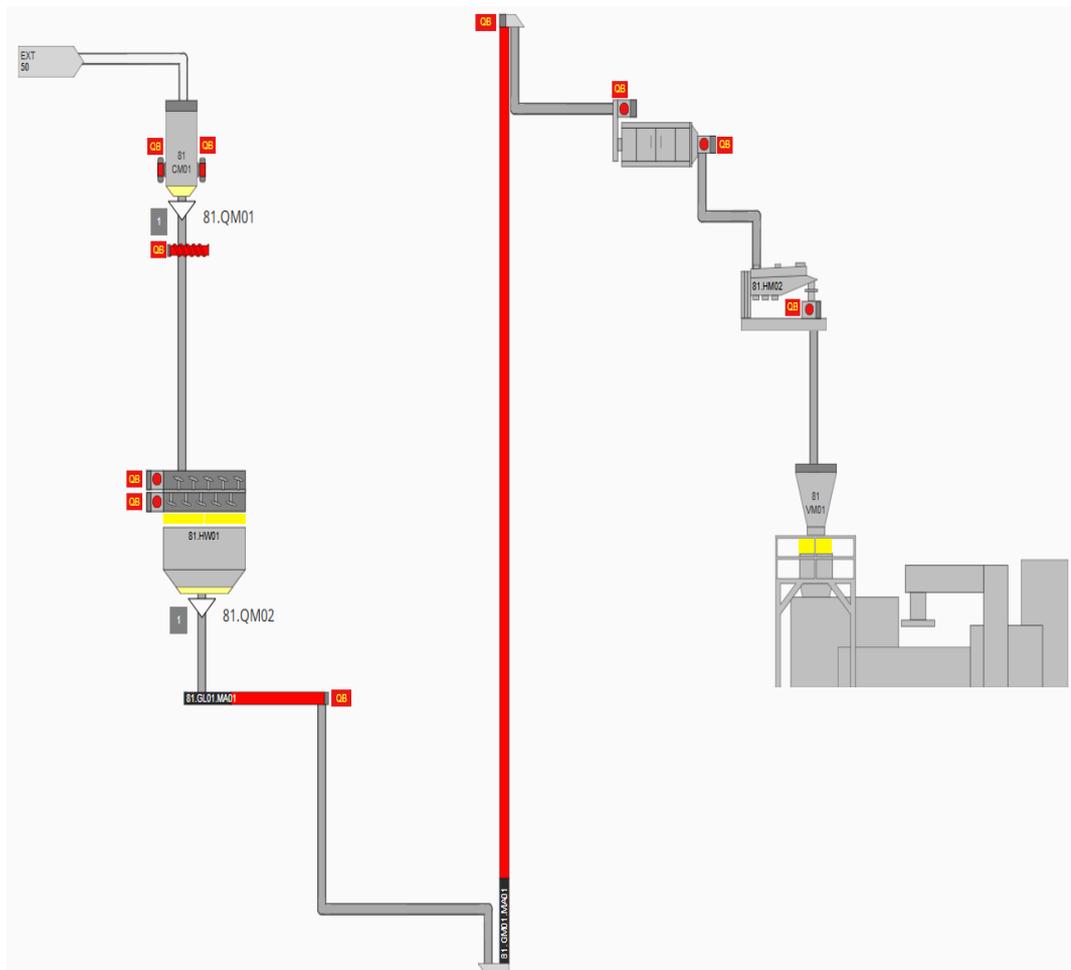
3.1.5. Revestimiento y empaque

Finalmente tenemos el área de revestimiento y empaque en el cual se le da un último mezclado al pellet y su respectivo baño con los líquidos incluidos en la receta de producción, en esta área ya no es un proceso continuo, aquí se maneja un proceso por batch en el cual cada batch es mezclado con los líquidos en un tiempo específico con el objetivo de ser más nutritivo para el camarón, tal como se ilustra en la Figura 28.

Finalmente, después del proceso de mezcla denominado revestimiento se procede a trasladar el producto al área de empaque, donde se realiza un último proceso de zaranda con el objetivo de extraer el polvo o el grano de poca calidad del resto del producto, luego de este proceso el pellet entra en la tolva de empaque donde es almacenado en sacos y empaquetado por brazos robóticos para su venta y entrega.

Figura 28.

Área de revestimiento y empaque.



Nota. Elaboración propia.

3.2. Permisivos

Los permisivos son definidos como condiciones físicas y lógicas a las cuales se deben someter actuadores o equipos de accionamientos como motores y válvulas, esto con el fin de salvaguardar la producción, seguridad industrial y energética.

3.2.1. Secuencia PLC

En el autómatas la estructura de programación será implementada en lenguaje Ladder por su facilidad de legibilidad y estructura lógica, en el cual se diseñara de forma que una secuencia de activación de un bit en una bobina

estará condicionada por una serie de contactos abiertos o cerrados evaluando la secuencia de producción aguas abajo del equipo, es decir que para el accionamiento de apertura de una válvula esta evalué que el transportador/elevador o tornillo esté en funcionamiento con el fin de evitar derrame y desperdicio de producto.

Figura 29.

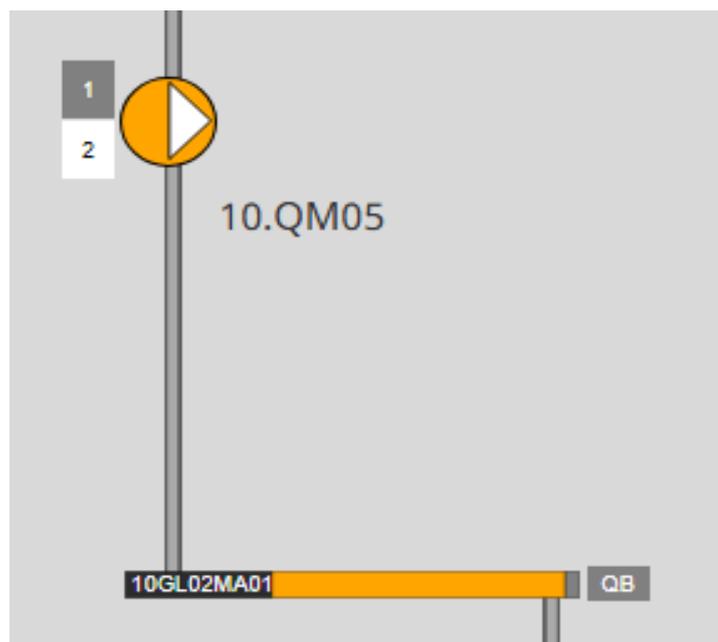
Logica LADDER de permiso de proceso.



Nota. Elaboración propia.

Figura 30.

Diagrama de secuencia en sistema SCADA.

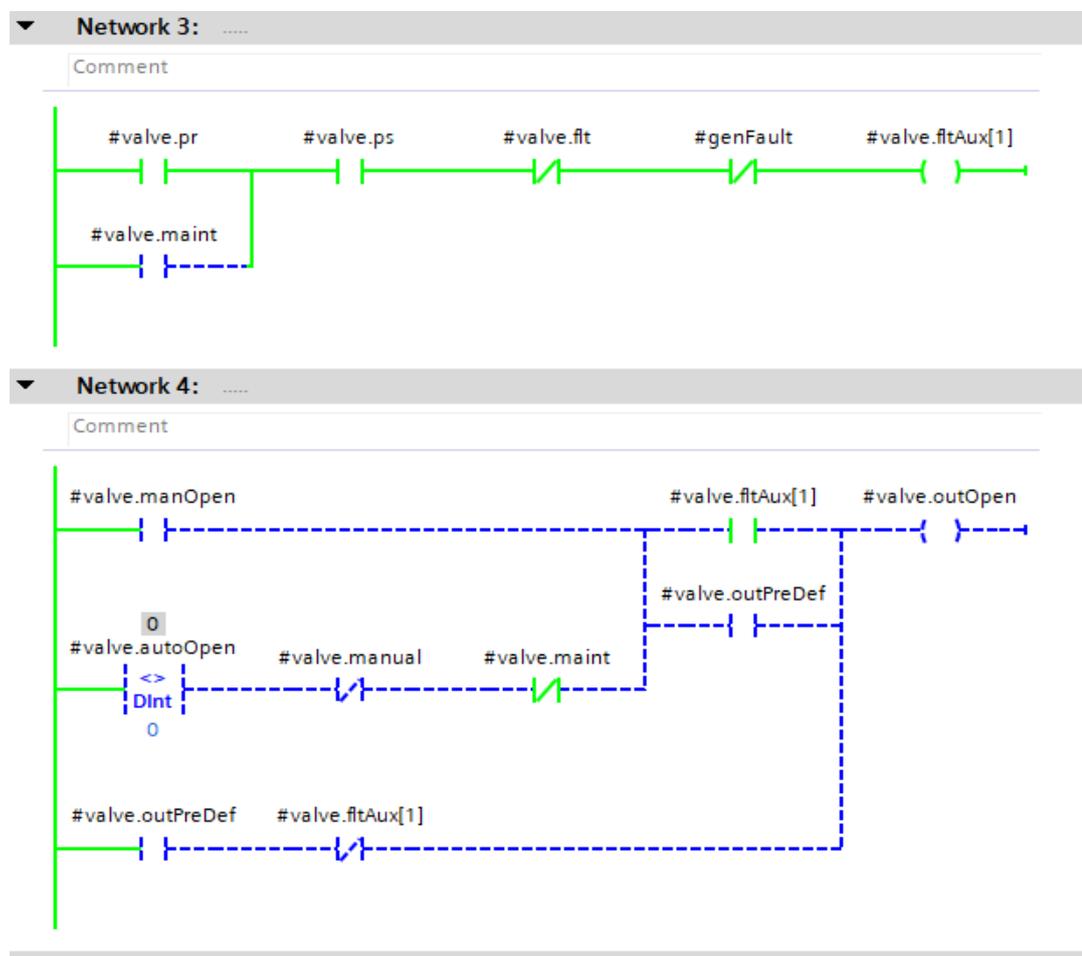


Nota. Elaboración propia.

En la Figura 30 se aprecia el diagrama del sistema SCADA referente a la conexión entre la válvula denominada 10.QM05 y el transportador denominado 10.GL02.MA01. En la Figura 29 se observa que el bit *.pr* (*permisivo de proceso*) de la válvula mencionada está condicionada por la activación de bit *.running* (*corriendo*) del motor.

Figura 31.

Condición del bit OutOpen.



Nota. Elaboración propia.

La Figura 31 muestra dos redes de trabajo donde en la red 3 se evalúa que para la activación de un bit denominado *.fltAux[1]* perteneciente al tag que representa cualquier instancia de una válvula se encuentra condicionada por una serie de bit de los cuales dos son permisivos de proceso y seguridad

.pr y *.ps* respectivamente.

En la red 4 se aprecia que para el enclavamiento del bit *.outOpen* perteneciente a la instancia de una válvula esta de igual forma condicionada por una estructura en la cual se comprende que el bit *.fltAux[1]* expuesto en la red 3 es requerimiento para la activación, por lo cual en caso de perderse uno de los permisos o condiciones de apertura de la válvula esta perderá la señal de apertura y se procederá a cerrar o devolver a su estado base por falta de permisos.

3.2.2. Seguridad PLC

Para la activación de un bit como permiso de seguridad denominado *.ps* al igual que el permiso de proceso será implementado en lenguaje Ladder por su legibilidad, y condicionada con una serie de contactos vinculados a estados de equipos o señales proveniente de campo o condiciones lógicas, en específico para los motores se evalúa que el equipo no esté en falla de encendido, que tenga presente la señal del seccionador de campo y guarda motor tanto para el equipo en secuencia de producción evaluado en el permiso de proceso como para la misma instancia del equipo como seguridad.

Figura 32.

Lógica LADDER de permiso de seguridad.



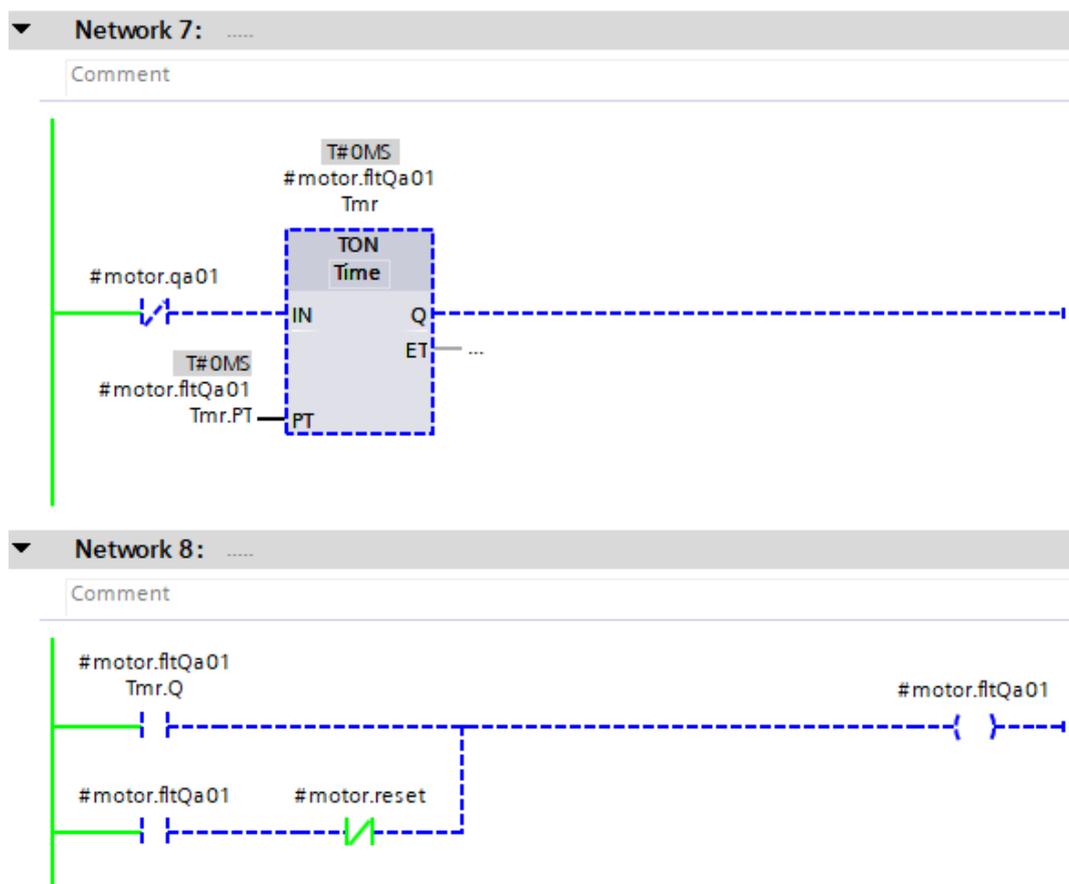
Nota. Elaboración propia.

La Figura 32 muestra el mismo ejemplo de la válvula 10.QM05 donde se condiciona el bit *.ps* con tres contactos instanciado a los estados lógicos de las señales de falla que no este activa, seccionador de campo y guardamotor activas para así tener el bit de permiso de seguridad energizado.

El motor que está en secuencia de producción a la válvula 10.QM05 es el transportador 10.GL02.MA01 al cual también aplican los mismos permisos de proceso, pero para este se instancian de otra forma como se indica en la Figura 33 y 34.

Figura 33.

Lógica por falta de seccionador de campo.

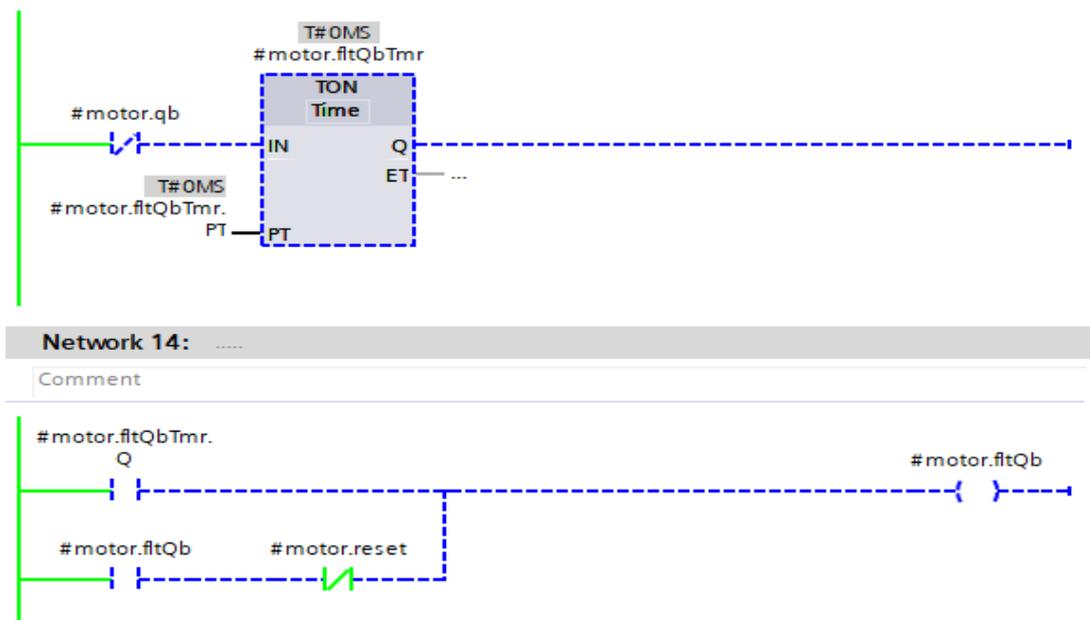


Nota. Elaboración propia.

La Figura 33 muestra dos redes de trabajo donde la 7 indica que si no se tiene presente la señal del seccionador de campo se activara el conteo de un tiempo, una vez que se cumpla ese tiempo se activa un bit que a su vez levanta o energiza una bobina vinculada a la instancia del motor evaluado y activa el bit de falla por seccionador .*fltQa01*.

Figura 34.

Lógica por falta de guardamotor.



Nota. Elaboración propia.

Al igual que el ejemplo que se presenta en la Figura 34 con el seccionador de campo, la misma lógica aplica para el guardamotor, en caso de no tener la señal de campo este procederá a activar un contador, el cual finalizará en el levantamiento de una señal de falla.

Las señales de fallas como norma deben de detener el equipo en cuestión en caso de motores o devolver el equipo a su estado por defecto en el caso de válvulas.

3.2.3. Arranque PLC

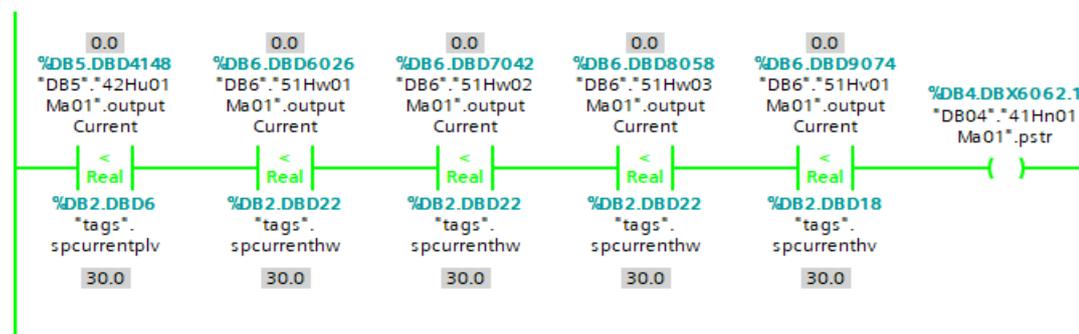
El permisivo de arranque tiene como propósito restringir un arranque simultaneo de los motores más grande en planta, es decir restringir que cuando se arranque un motor de algún equipo como un molino, los equipo como pulverizador y extrusor no puedan ser encendidos, de esta forma se evitan picos de corriente y sobre corriente de varios equipos en las diferentes líneas de transmisión eléctrica alrededor de toda la planta.

Esto ayuda a reducir el estrés en la red y a reducir costos en la facturación por el servicio eléctrico, además de salvaguardar la integridad de los motores más potentes e importantes de las líneas de producción.

Para evaluar el permisivo de arranque los equipos se validarán con un valor referencial de corriente, debido a que cuando un motor arranca este ocupa aproximadamente tres veces su corriente nominal por un breve periodo de tiempo, la lógica a implementar validará que posterior a que la corriente se equilibre permita energizar un bit denominado *.pstr* como permisivo de arranque, y por función se validará un tiempo como filtro para amortiguar el permisivo.

Figura 35.

Esquema de permisivos de arranque en pulverizador.

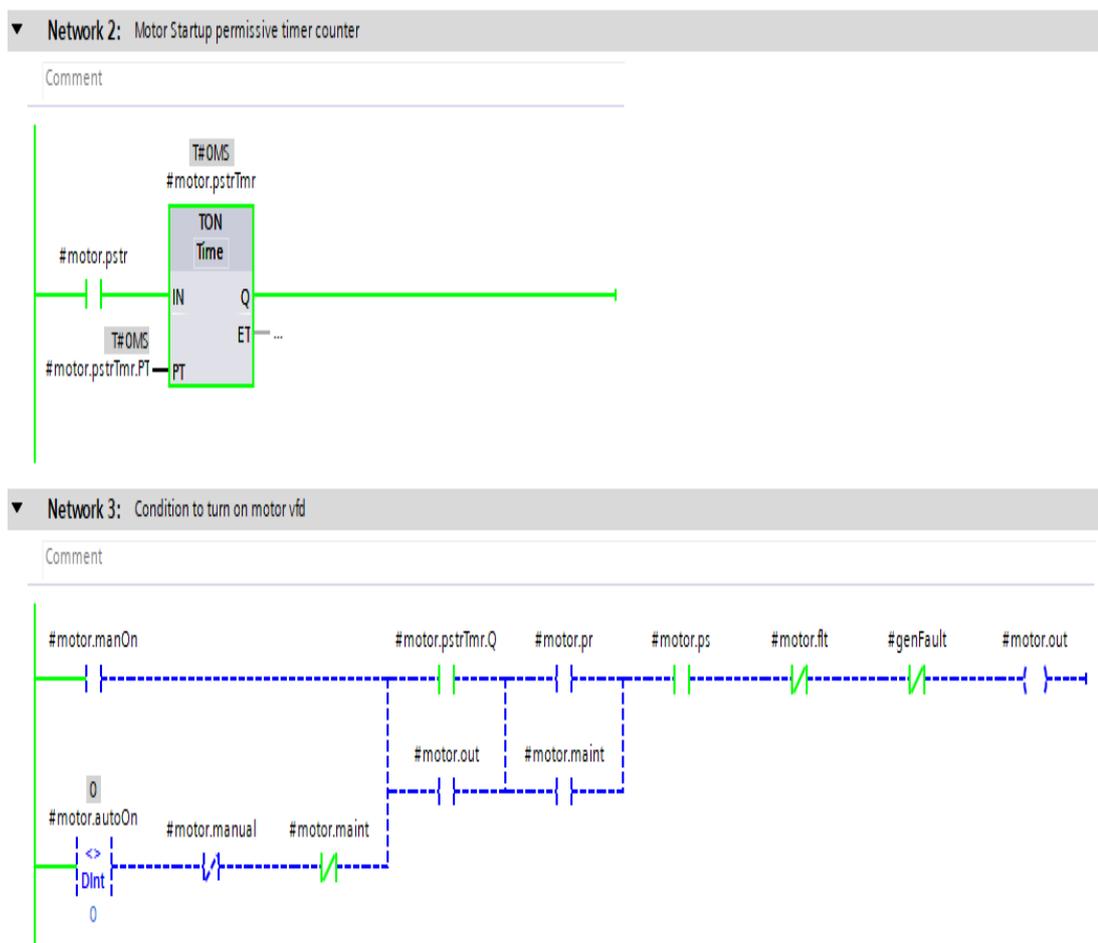


Nota. Elaboración propia.

En la Figura 35 se aprecia el condicionamiento para la activación de bit `.pstr` perteneciente a la instancia del motor 41.HN01MA01 el cual es de un molino, este bit depende de que los motores del pulverizador 42.HU01.MA01, acondicionadores 51HW0#.MA01 y al extrusor 51.HV01.MA01 no sobrepasen la corriente establecida por el operador mediante el sistema SCADA.

Figura 36.

Timer de permisivo de arranque.



Nota. Elaboración propia.

Mediante una función o `addOn` se le pasan las instancias de un motor o el tag de un motor, en el cual dentro de la estructura de programación se aprecia que en la red de trabajo dos, al tener el bit `.pstr` se activa un contador de tiempo denominado `motor.pstrTmr` el cual al completar su conteo activara

un bit que se aprecia en la red de trabajo tres como tag de nombre *motor.pstrTmr.Q*, como se muestra en la Figura 36.

Esta estructura tiene como propósito que, posterior a la activación del bit, este amortigua la señal por un momento antes de permitir el arranque de otros equipos. De este modo, se garantiza una transición secuencial en el inicio de la operación del proceso, evitando así que se produzcan activaciones al mismo tiempo.

3.2.4. Etapas del proceso de popup Ignition

Ignition es una plataforma de desarrollo de sistemas informáticos, esta plataforma está compuesta por diferentes módulos lo cual permite contar con una gran versatilidad de herramientas para trabajar, como por ejemplo comunicaciones OPC directa con PLC de diferentes marcas, librerías para establecer una comunicación con base de datos como SQL server, MySQL, Posgres SQL, etc. De igual manera, su diseño modular posibilita que cada proyecto se adapte y amplie de acuerdo con los requerimientos de la planta, abarcando desde aplicaciones de monitoreo reducida hasta sistemas SCADA integrales.

Adicional de módulos de script basadas en Python y módulos de conexiones remotas por TCP/IP entre servidores de Ignition. Esto permite la unificación de diversas secciones de la fábrica, concentrando los datos en un único espacio visual.

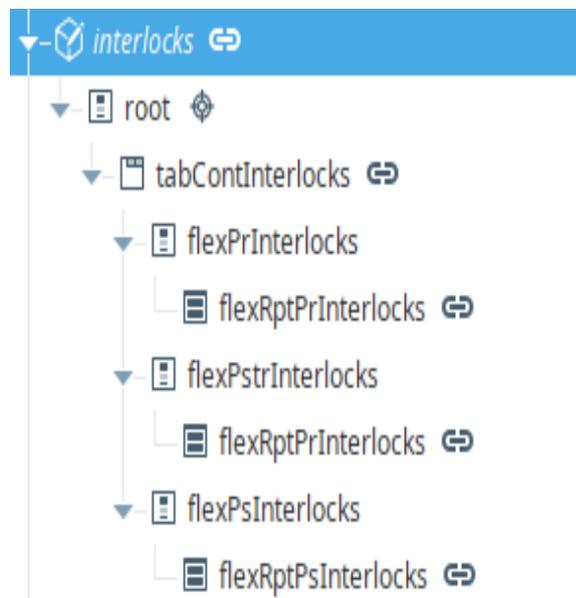
También presenta herramientas de diseño en el cual se pueden crear interfaces gráficas, la Figura 37 muestra la estructura que posee el popup o ventana donde se muestra la lista de permisivos o condiciones requeridas para el accionamiento o activación de un actuador sea este motor o válvula,

proporcionando una rápida comprensión de la situación operativa.

Se aprecia que dentro del menú denominado “root” se contiene otros contenedores de tipo pestañas denominador “tabContInterlocks” y dentro de este se presentan tres contenedores de tipo flexible con nombre “flexPr”, “flexPs” y “flexPstr” en el cual cada uno contiene otro contenedor de tipo repetidor flexible en el cual repite n cantidad de veces un template invocado, al cual se le pasan parámetros de una lista de permisos.

Figura 37.

Árbol de proyecto del popup de interlocks.



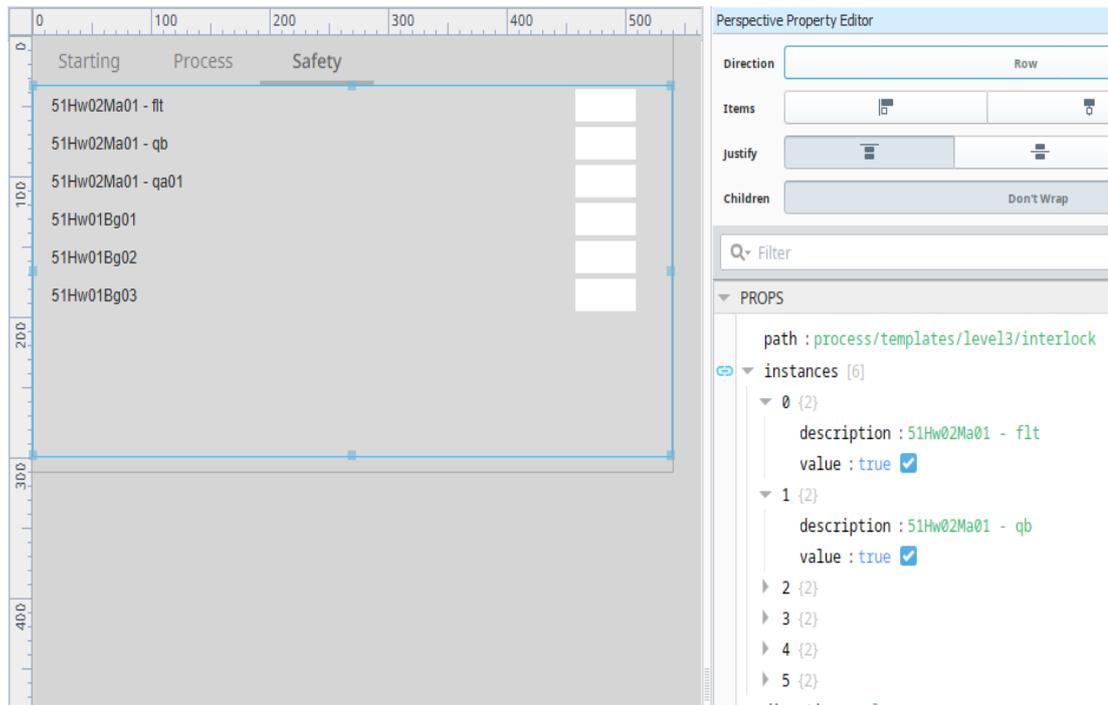
Nota. Elaboración propia.

La Figura 38 muestra cómo se visualizará la ventana de permisos en el sistema SCADA, del lado izquierdo se pueden visualizar las instancias en donde cada instancia se repite mediante el contenedor repetidor y se muestra la data que se le envían como parámetro por niveles en un código json.

La instancia 0 contiene dos datos, “description” el cual contiene el mensaje a mostrar en la ventana y “value” que a su vez contiene un valor de tipo booleana para indicar si cumple o no cumple el permiso descrito.

Figura 38.

Popup de interloks.

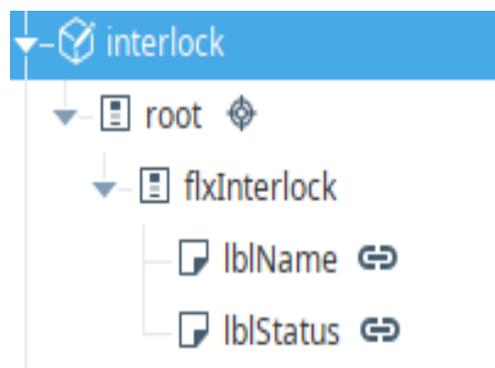


Nota. Elaboración propia.

En la Figura 39 se muestra el árbol de proyecto del template invocado en el popup de permisos, este es compuesto por dos labels, uno donde muestra la descripción que se le envíe como parámetro y el otro objeto muestra el valor del permiso.

Figura 39.

Árbol del proyecto de template para interloks.

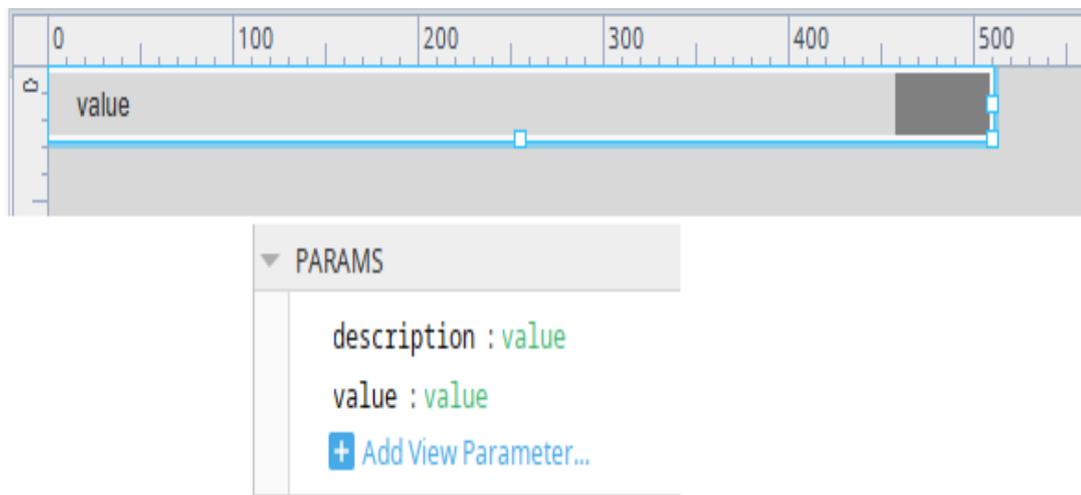


Nota. Elaboración propia.

Los objetos poseen propiedades y la plataforma Ignition a su vez tiene la capacidad de vincular propiedades de objetos con script, otras propiedades, tags y consultas SQL, por lo cual se debe de realizar un vínculo entre la propiedad del label y el parámetro que este recibiendo, esto se puede apreciar en la Figura 40 donde se visualizan los parámetros.

Figura 40.

Témlate de Interloks.



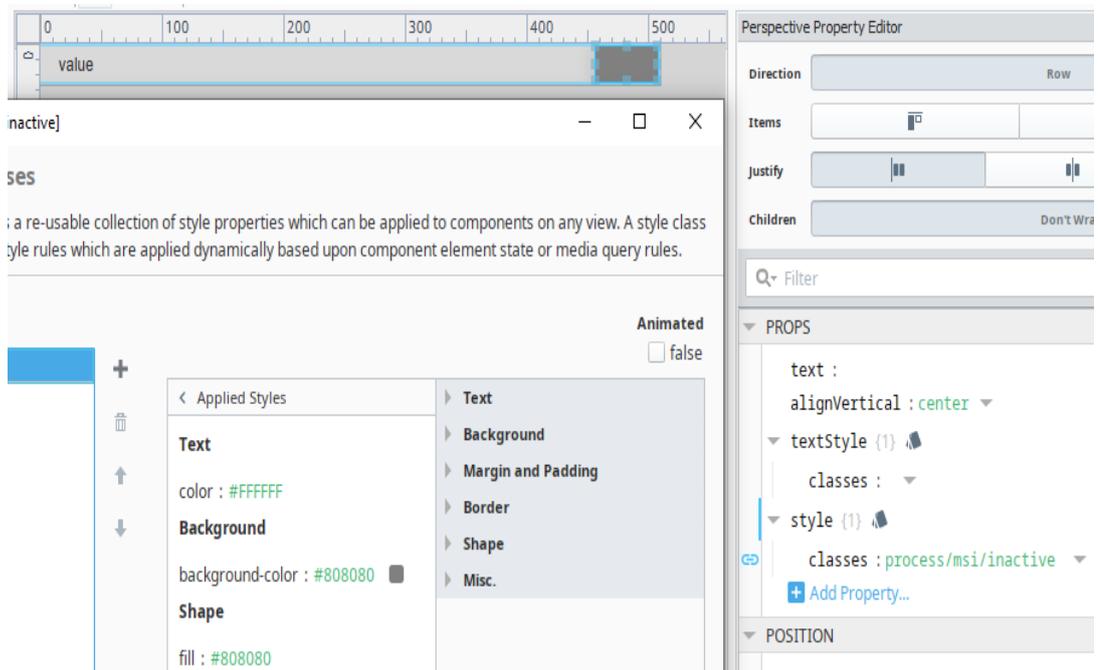
Nota. Elaboración propia.

El label que visualmente es representado como una caja gris en caso de no cumplir con la condición y color blanco en caso de cumplirla, posee un estilo diferente debido a que ese efecto es obtenido mediante estilos el cual es otro módulo de Ignition.

En la sección izquierda inferior se observa que el label tiene un parámetro de estilo en el cual se le indica la dirección de un estilo de colores como se aprecia en la Figura 41. Además, esta forma visual ayuda al operador a reconocer de manera ágil la condición de cada interlock, lo que refuerza la seguridad del proceso. La capacidad de ajustar estilos hace más fácil modificar el diseño sin afectar la operación.

Figura 41.

Binding de estilos.



Nota. Elaboración propia.

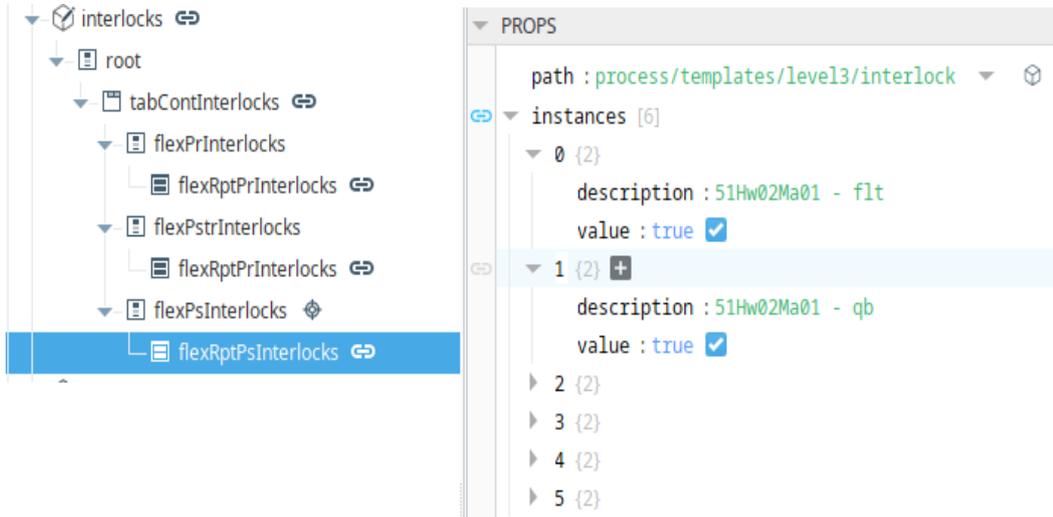
3.2.5. Script y binding en Ignition

Ignition posee dos herramientas muy útiles al momento de diseñar sistemas, los objetos implementados en todos los desarrollos poseen propiedades a las cuales se les pueden realizar vínculos, estos vínculos son denominados bindings.

Los vínculos pueden realizarse entre propiedades, tags, consultas a base de datos, y script para tartar información, por lo cual para el popup de permisos se realizará un binding de la propiedad de instancias dentro del contenedor flex repeater como se aprecia en la Figura 42. Esta funcionalidad posibilita la actualización automática de la información sin necesidad de intervención humana, mejorando la vigilancia en tiempo real. Asimismo, la implementación de bindings facilita la uniformidad de los componentes visuales en la interfaz, previniendo configuraciones repetitivas.

Figura 42.

Instancias con propiedad binding.

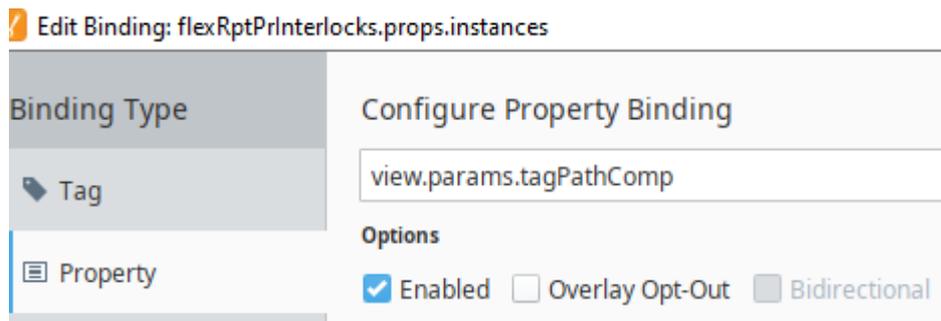


Nota. Elaboración propia.

La Figura 43 muestra la interfaz del vínculo de instancias hacia la propiedad de parámetros denominada “tagPathComp”.

Figura 43.

Binding de permisos.



Nota. Elaboración propia.

La Figura 44 muestra el script en el cual se almacena el valor que se le pase como parámetro, y se arma la dirección donde buscar, es decir que como parámetro se le envía la dirección de un tag, luego con la función readBlocking se accede a la información contenida en el tag.

Figura 44.

Script del binding.

```
Script
1 def transform(self, value, quality, timestamp):
2     param = value
3     pr = '/prPermissives'
4     path = param + pr
5     data = system.tag.readBlocking(path)[0].value
6     dic = []
7     columns = data.getColumnCount()
8     rows = data.getRowCount()
9
10    for i in range(rows):
11        always = data.getValueAt(i,0)
12        tagPath = data.getValueAt(i,1)
13        toEval = data.getValueAt(i,2)
14        aw = tagPath.find("Aw")
15        if aw > -1:
16            spPath = tagPath + '/sp'
17            pvPath = tagPath + '/pv'
18            sp = system.tag.readBlocking(spPath)[0].value
19            pv = system.tag.readBlocking(pvPath)[0].value
20            instance = system.tag.readBlocking(tagPath + '.InstanceName')[0].value
21            descrip = instance + ' - PV: ' + str(pv) + ' < SP: ' + str(sp)
22            if pv < sp:
23                val = bool(1)
24            else:
25                val = bool(0)
26        else:
27            toComp = system.tag.readBlocking(tagPath)[0].value
28            name = system.tag.readBlocking(tagPath + '.Name')[0].value
29            instance = system.tag.readBlocking(tagPath + '.InstanceName')[0].value
30            descrip = instance + ' - ' + name
31            if toComp == toEval:
32                val = bool(1)
33            else:
34                val = bool(0)
35        dictemp = {
36            'description':descrip,
37            'value':val
38        }
39        dic.append(dictemp)
40    return dic
```

Nota. Elaboración propia.

Con un loop de tipo for se itera toda la información almacenada en el tag como se muestra en la Figura 45, dentro de la data del tag se almacenan más direcciones en la cuales el script evalúa su valor y la compara con la columna de condición, en caso de ser iguales o no los valores este almacena un valor booleano verdadero o falso dentro de un diccionario en conjunto con la descripción, luego ese diccionario es devuelto a la propiedad de instancias en el flex.

Figura 45.

Información en el tag.

alwaysRead	tagPath	condition
<input checked="" type="checkbox"/>	[S7-1511]EXT/51Hw02Ma01/fit	<input type="checkbox"/>
<input checked="" type="checkbox"/>	[S7-1511]EXT/51Hw02Ma01/qb	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	[S7-1511]EXT/51Hw02Ma01/qa01	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	[S7-1511]EXT/51Hw01Bg01	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	[S7-1511]EXT/51Hw01Bg02	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	[S7-1511]EXT/51Hw01Bg03	<input checked="" type="checkbox"/>

PARAMS
tagPathComp : [S7-1511]EXT/51Hw01Ma01

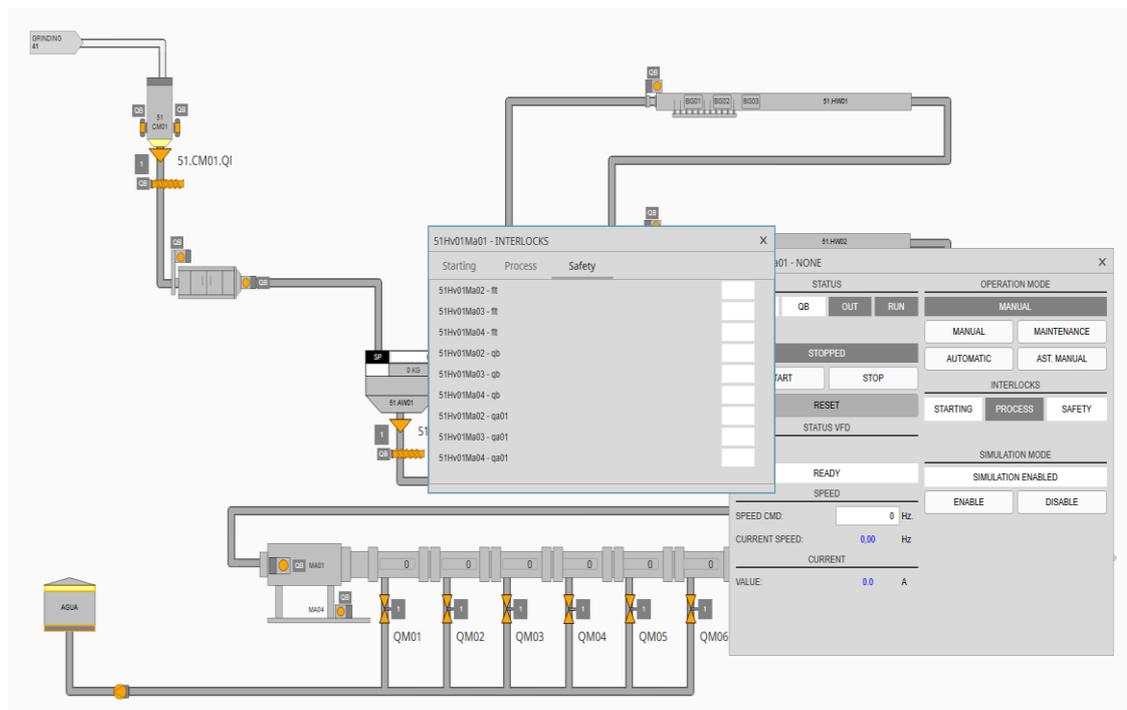
Nota. Elaboración propia.

3.2.6. Visualización

Finalmente, las Figuras 46, 47 y 48 muestran como se aprecia la visualización de los permisos de los tres tipos dentro del sistema SCADA propuesto, en la figura 46 se aprecian los permisos de seguridad del motor principal del extrusor.

Figura 46.

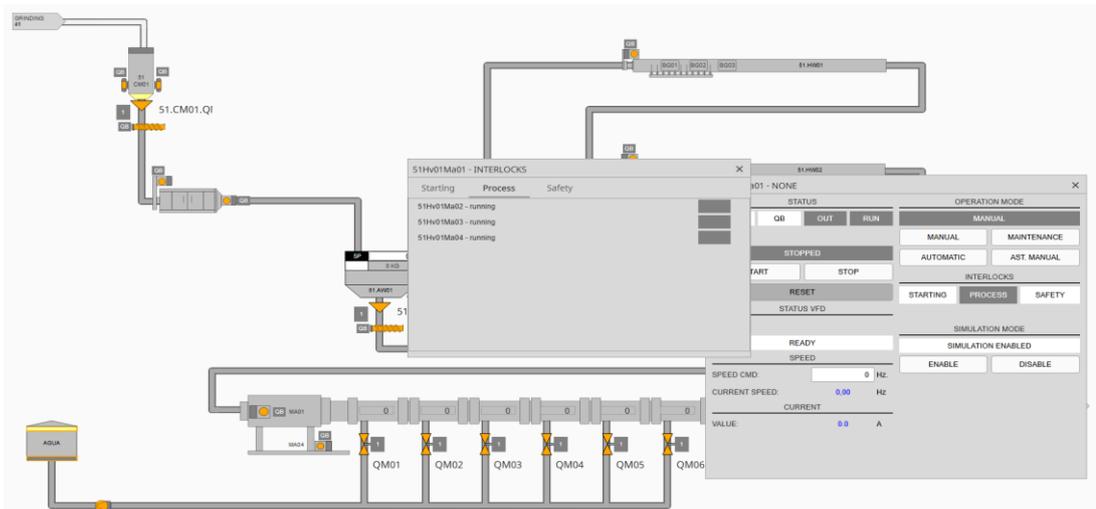
Permisos de seguridad en SCADA.



Nota. Elaboración propia.

Figura 47.

Permisivos de proceso en SCADA.

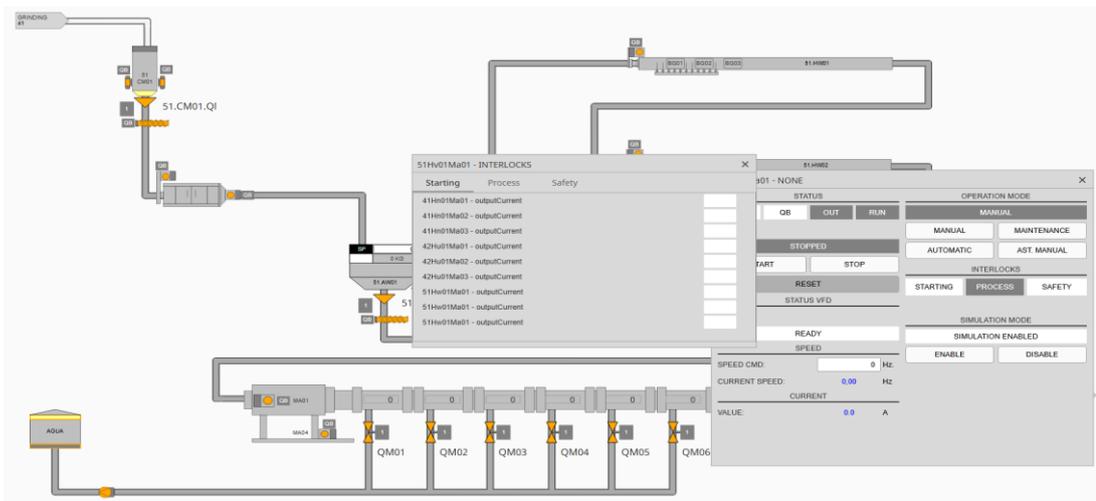


Nota. Elaboración propia.

La Figura 47 por su parte muestra los permisos de proceso del motor del extrusor donde se aprecia que no cumple con las condiciones de que los motores de las cuchillas y el suministro de aceite estén activos.

Figura 48.

Permisivos de arranque en SCADA.



Nota. Elaboración propia.

Finalmente, la Figura 48 muestra las condiciones de arranque del extrusor donde se evalúa el valor de la corriente de los equipos grandes.

CAPÍTULO 4: DISEÑO DEL SISTEMA DE CONTROL Y LÓGICA DE PERMISIVOS PARA LA LÍNEA DE PRODUCCIÓN

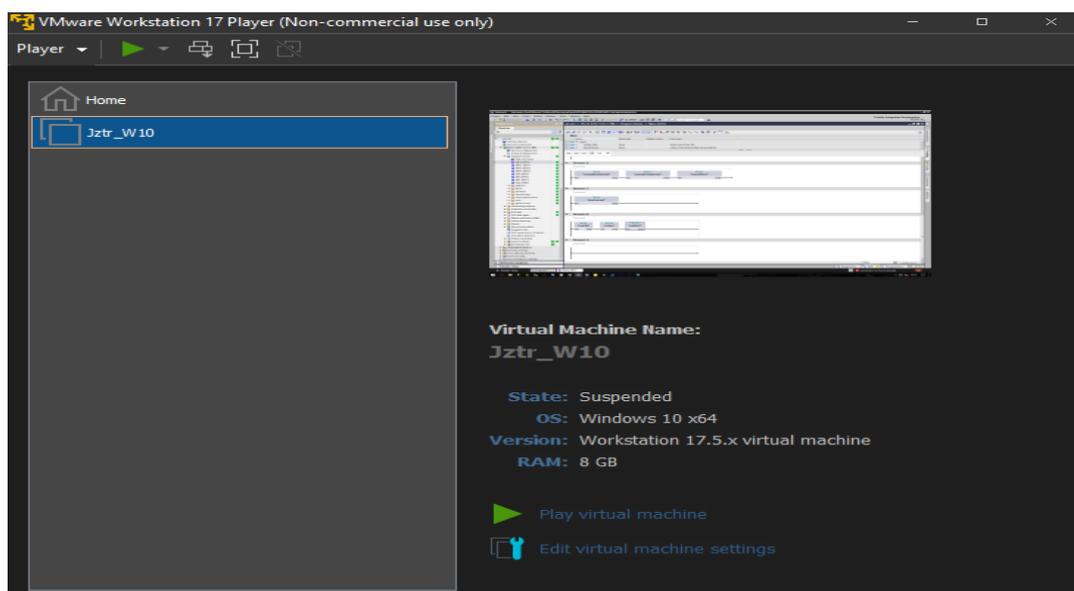
4.1. Virtualización

Para el proceso de contenerización del sistema se realizará mediante el uso de máquinas virtuales donde se puede almacenar un sistema operativo completo, teniendo así la facilidad de instalar el servidor del sistema SCADA, base de datos y otros servicios por separados en diferentes máquinas virtuales con el fin de no sobrecargar un solo servidor sea este físico o virtualizado.

El software que se utilizar para la virtualización es VMware Station, siendo este un programa muy completo al momento de compartir recursos y servicios con el servidor anfitrión, dando así la facilidad de tener servidores físicos y virtualizados en una sola red con el fin de acceder a los equipos de campo mediante comunicación de redes industriales.

Figura 49.

Ide VMWare.



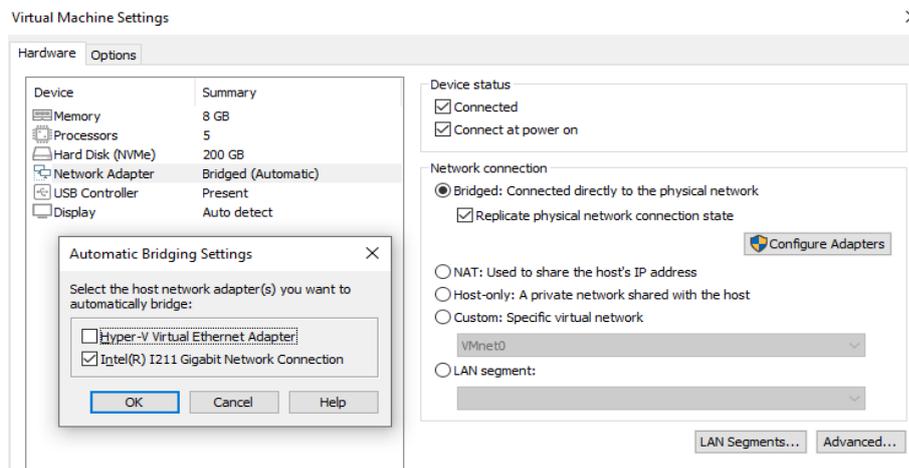
Nota. Elaboración propia.

4.2. Red Ethernet/IP

La plataforma de virtualización de VMWare posee muchas configuraciones de las cuales una de ellas es el compartimiento de la red Ethernet/IP dando así la funcionalidad de que la maquina virtualizada pueda estar dentro del mismo segmento de red en el cual se encuentre el servidor anfitrión y así permitiendo la comunicación con los equipos de campo como el controlador lógico programable PLC.

Figura 50.

Configuración de red VMWare.



Nota. Elaboración propia.

La figura 50 muestra la interfaz de configuración de red dentro VMWare, se comparte solo la tarjeta de red física al servidor virtualizado para estar dentro del mismo segmento de red.

Figura 51.

Comando IP por cmd en sistema virtualizado.

```
Adaptador de Ethernet Ethernet:  
  
Sufijo DNS específico para la conexión. . . :  
Dirección IPv4. . . . . : 192.168.100.7  
Máscara de subred . . . . . : 255.255.255.0  
Puerta de enlace predeterminada . . . . . : 192.168.100.1
```

Nota. Elaboración propia.

La Figura 51 muestra el comando “ipconfig” dentro del servidor virtualizado con el objetivo de visualizar que dirección IP posee posterior a la configuración de red en la interfaz de VMware, garantizando la correcta comunicación estable.

Figura 52.

Comando IP por cmd en sistema anfitrión.

```
Adaptador de Ethernet Ethernet:

Sufijo DNS específico para la conexión. . . :
Dirección IPv6 . . . . . : 2800:bf0:8000:a87:1d77:3d78:bd16:29f8
Dirección IPv6 temporal. . . . . : 2800:bf0:8000:a87:500e:dc66:eea2:4988
Vínculo: dirección IPv6 local. . . : fe80::f671:5602:5667:5e17%18
Dirección IPv4. . . . . : 192.168.100.11
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . . : fe80::1%18
                                           192.168.100.1
```

Nota. Elaboración propia.

La Figura 52 muestra el mismo comando, pero en el servidor anfitrión, esta acción es para constatar que ambos servidores tanto el físico como el virtualizado están dentro del mismo segmento de red.

Figura 53.

Ping desde el servidor anfitrión al virtualizado.

```
Haciendo ping a 192.168.100.7 con 32 bytes de datos:
Respuesta desde 192.168.100.7: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 192.168.100.7:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 0ms, Media = 0ms
```

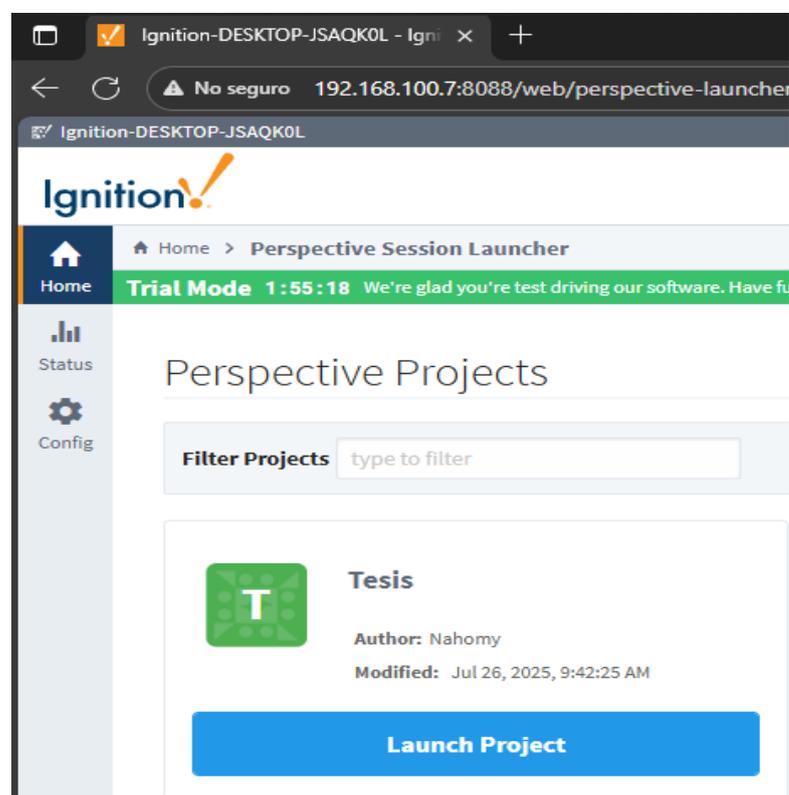
Nota. Elaboración propia.

Finalmente, en la figura 53 muestra la ejecución del comando “ping” con el fin de comprobar que el servidor anfitrión tiene acceso a la dirección IP perteneciente al servidor virtualizado, se evidencia que, de los cuatro bloque de comunicación, los cuatro fueron respondidos correctamente.

Para el acceso al servidor del sistema SCADA se puede ingresar mediante un navegador web utilizando la dirección IP y el puerto de comunicación 8088 perteneciente a Ignition, este puerto se lo establece al momento de instalar la plataforma de desarrollo de Ignition, la figura 54 muestra el acceso al Gateway del proyecto SCADA para este trabajo de titulación. Evidenciando la correcta configuración y disponibilidad del entorno de supervisión.

Figura 54.

Gateway SCADA por dirección IP virtualizada.



Nota. Elaboración propia.

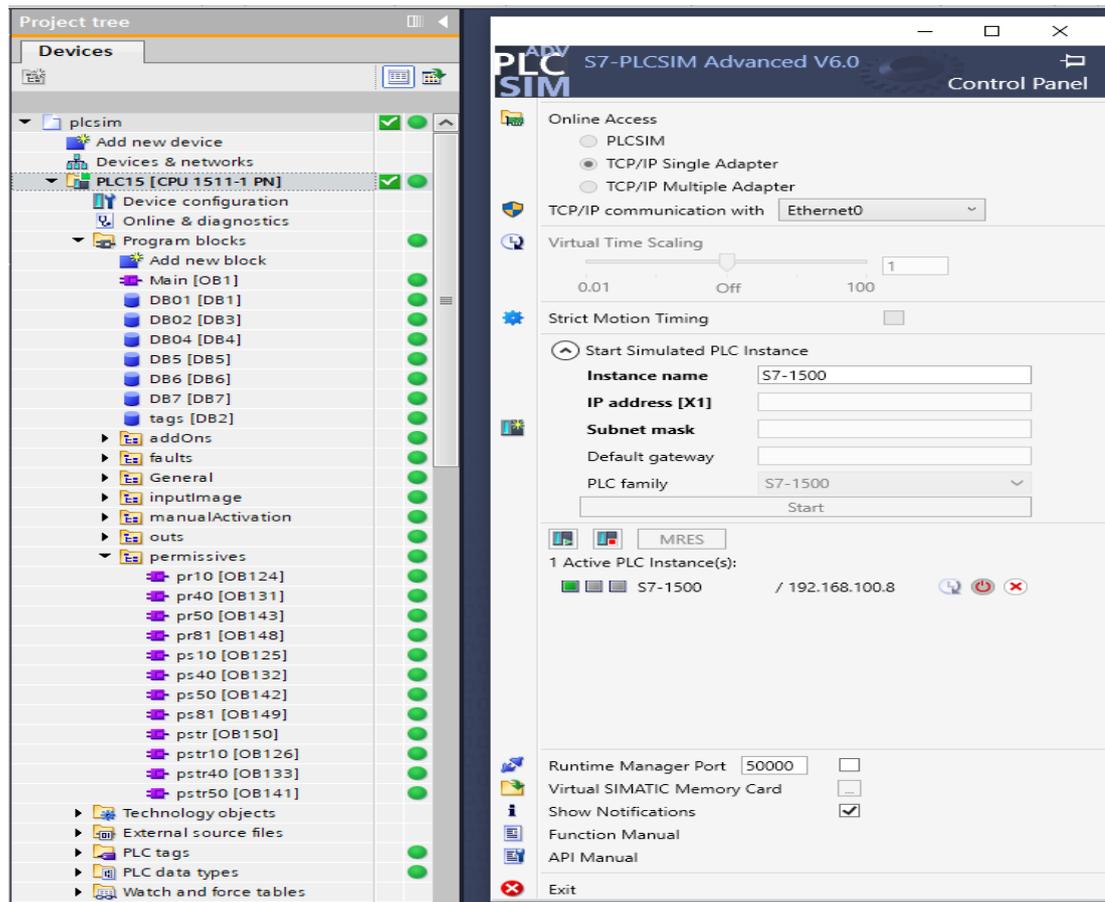
4.3. PLCSIM

Para el proceso de simulación del PLC se usará el programa de Siemens PLCSIM donde se puede configurar un autómata de forma digital al que se le puede cargar la programación implementada en el proyecto y probar lógicas previas a su implementación en campo.

La figura 55 muestra de lado izquierdo todas las funciones, bloques de datos, carpetas y rutinas implementadas en la lógica de programación, esta lógica se le carga al software de simulación antes mencionado, este debe de estar configurado de forma que sea accesible via IP para su posterior conexión con Ignition.

Figura 55.

Simulador de PLC Siemens.



Nota. Elaboración propia.

4.4. Conexión de Ignition con PLCSim

La plataforma de Ignition posee módulos específicos para la comunicación con autómatas vía driver OPC, esta comunicación se establece mediante una dirección Ethernet/IP.

Figura 56.

Declaración de PLC OPC en Ignition via Ethernet/IP.

General	
Name	<input type="text" value="SIMS7-1511"/>
Description	<input type="text"/>
Enabled	<input checked="" type="checkbox"/> (default: true)

Connectivity	
Hostname	<input type="text" value="192.168.100.8"/>
Local Address	<input type="text"/> Address of network adapter to connect from. (default:)
Timeout	<input type="text" value="2000"/> (default: 2.000)

Nota. Elaboración propia.

Figura 57.

Lista de PLC declarados en Ignition.

Name	Type	Description	Enabled	Status		
S7-1212C	Siemens S7-1200		false	Disabled	<input type="button" value="delete"/>	<input type="button" value="edit"/>
SIMS7-1511	Siemens S7-1500		true	Connected	<input type="button" value="delete"/>	<input type="button" value="edit"/>

Nota. Elaboración propia.

4.5. Cliente del SCADA.

Dentro del Gateway de la plataforma Ignition se pueden invocar clientes, sean estos en un navegador web para proyectos realizados en perspective o en un cliente embebido en Windows mediante Ignition Client en caso de realizar un proyecto en visión, para este trabajo de titulación se utilizó el módulo de desarrollo web de Ignition denominado perspectiva, por lo cual los clientes están diseñados en una interfaz web y pueden ser invocados mediante una URL en un navegador web. El sistema cuenta con un total cinco pantallas en las cuales contienen todas las áreas de trabajo de la línea de producción, siendo estas intake, dosificación, molienda, pulverizado, extrusión, revestido y empaque. Estas pantallas están ilustradas en las Figuras 58 a 62, donde se puede observar la representación visual de cada área de la planta y la interfaz creada para la supervisión y funcionamiento de los procesos industriales.

Figura 58.

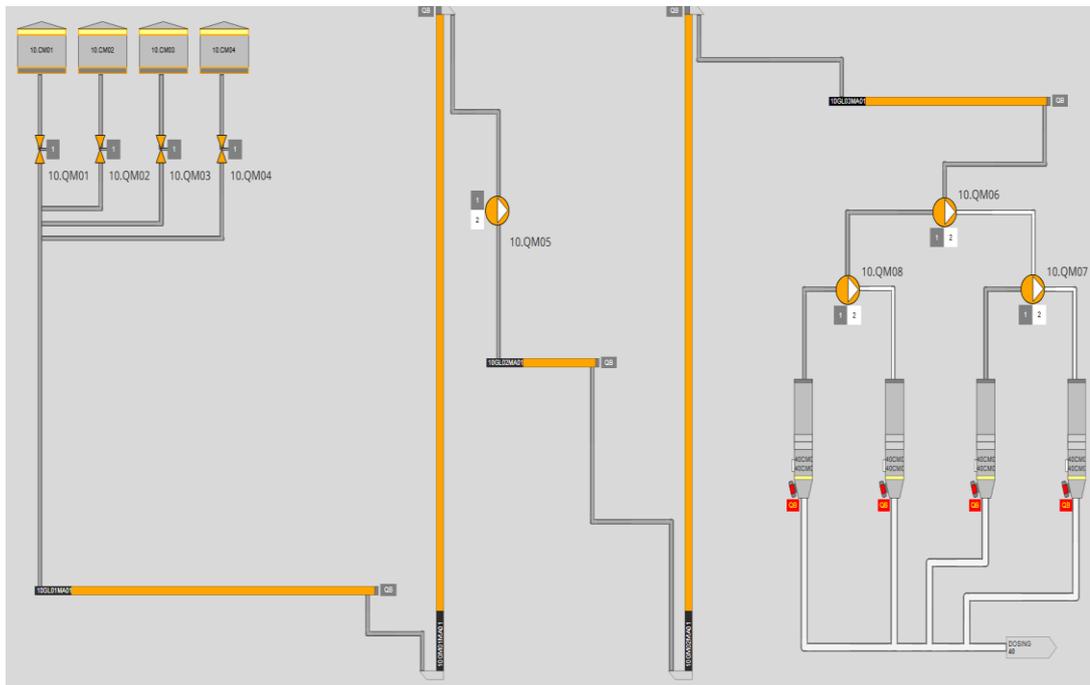
Áreas de trabajo en la línea de producción.



Nota. Elaboración propia.

Figura 59.

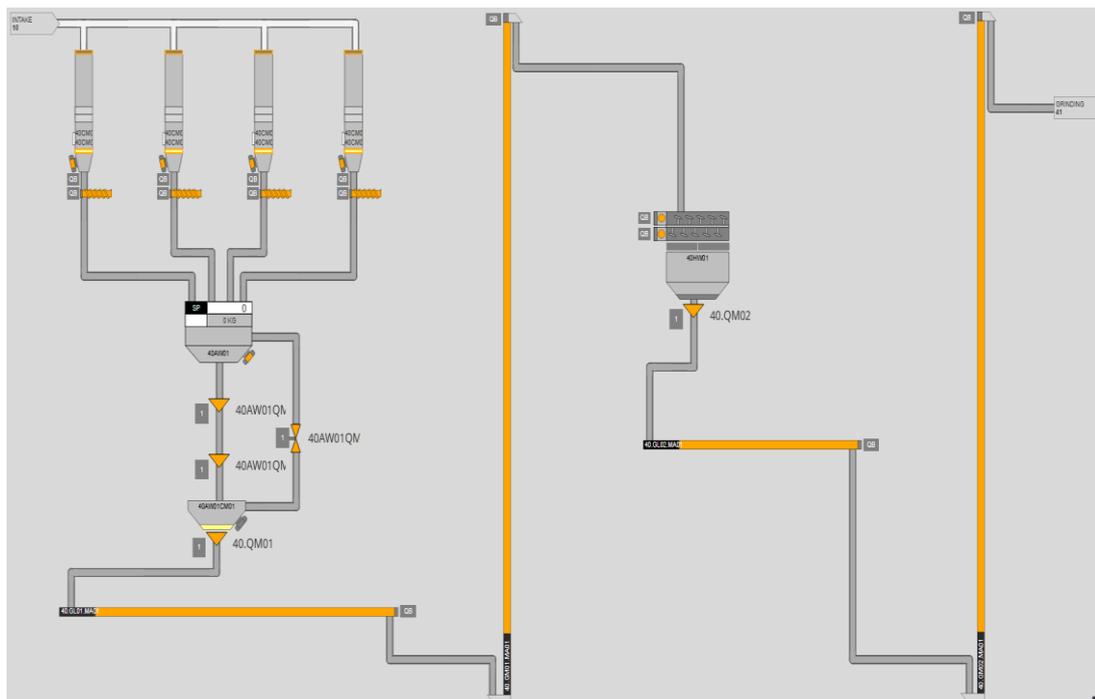
Pantalla de Intake.



Nota. Elaboración propia.

Figura 60.

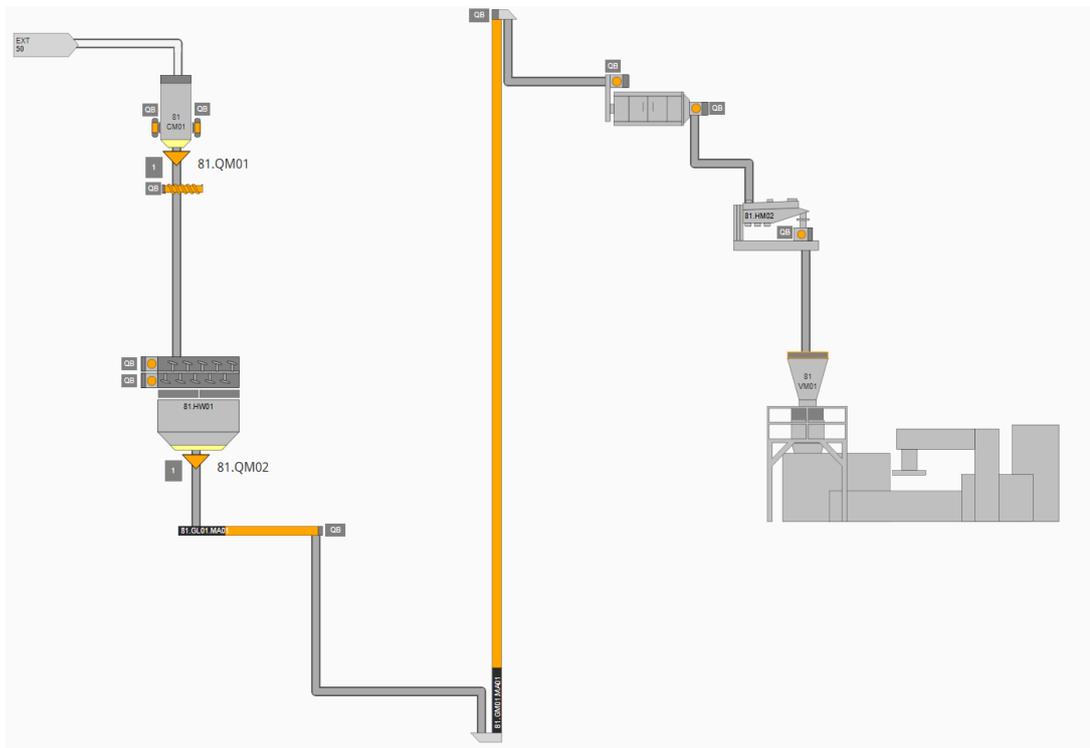
Pantalla de dosificación.



Nota. Elaboración propia.

Figura 63.

Pantalla de revestimiento y empaque.



Nota. Elaboración propia.

CAPÍTULO 5: ANÁLISIS DE RESULTADOS Y PRESUPUESTO.

5.1. Análisis Técnico de los resultados obtenidos

En el presente capítulo se realiza un análisis técnico sobre los beneficios obtenidos con la implementación del sistema de permisos de proceso, seguridad y arranque, en conjunto con la incorporación al sistema SCADA. De igual forma, se incorpora un presupuesto referencial que sustenta la propuesta planteada.

La implementación de la lógica de permisos de proceso, seguridad y arranque en un entorno de simulación PLCSIM e integración con SCADA Ignition permitió validar la funcionalidad propuesta sin intervención en planta real. Las pruebas se realizaron bajo condiciones simuladas que reproducen el comportamiento de la línea de producción de alimento acuícola, desde intake hasta empaque.

Los resultados obtenidos indican lo siguiente, demostrando el rendimiento de la lógica aplicada y su aporte a la mejora del funcionamiento del sistema:

- **Permisivos de proceso:** La secuencia lógica impidió la activación de equipos cuando las condiciones previas no fueron cumplidas (ejemplo: apertura de válvula bloqueada si el transportador de descarga no estaba operativo).
- **Permisivos de seguridad:** Las condiciones de permisos de seguridad que se basan en las señales de campo relacionadas a guardamotors y seccionadores de campo cumplen efectivamente el objetivo de devolver al equipo a su estado inicial o seguro en condiciones anómalas.

- **Permisivos de arranque:** El control de picos de corriente evitó el encendido simultáneo de motores de alta potencia, reduciendo estrés en la red eléctrica y riesgo de multas por demanda máxima.

En el entorno SCADA, la visualización en Ignition permitió:

- Monitoreo en tiempo real de estados de equipos y condiciones de permisivos.
- Interacción intuitiva para operadores mediante ventanas “popup” con estados booleanos de cada condición.
- Flexibilidad de diagnóstico gracias al uso de binding y scripts para consulta directa de tags en el PLC.

5.2. Eficiencia operativa

Tras la validación de la lógica, se estimó que la aplicación del sistema:

- Reduce tiempos de arranque de líneas en un 15% debido a la visualización clara de condiciones previas.
- Reduce en un 90% las fallas operativas ocasionadas por arranques sin verificación previa.
- Minimiza riesgos eléctricos asociados a picos de corriente durante arranques simultáneos.

5.3. Presupuesto referencial

A continuación, se presenta la estimación de costos necesaria para la implementación del proyecto, considerando los recursos, licencias y equipos requeridos. El presente presupuesto se basó considerando valores estimados del mercado y tiene como objetivo garantizar la sostenibilidad económica del proyecto. Los valores expuestos son aproximados y podrían ajustarse en función con la disponibilidad de equipos.in

Tabla 1.*Presupuesto de horas hombre.*

Fase	Horas estimadas	Tarifa (\$/h)	Subtotal USD
Levantamiento de información	16 h	100	1,600
Programación PLC (permisivos)	60 h	100	6,000
Diseño SCADA en Ignition	48 h	100	4,800
Integración PLC–SCADA	20 h	100	2,000
Pruebas y simulación PLCSIM	24 h	100	2,400
Documentación y manuales	12 h	100	1,200
Total Horas Hombre	180 h	—	18,000

Nota. Elaboración propia.

La Tabla 1 detalla las horas necesarias para el desarrollo completo del sistema, desde la recopilación de datos en planta hasta la documentación final. El total de 180 horas hombre representa un trabajo integral que incluye la lógica PLC, interfaz SCADA y validación en simulación.

Tabla 2.*Puesta en marcha por línea.*

Actividad	Tiempo estimado	Tarifa (\$/h)	Costo USD
Instalación de hardware PLC y módulos	8 h	100	800
Configuración y carga de programas	4 h	100	400
Pruebas en campo con producto real	8 h	100	800
Ajustes y capacitación a operarios	4 h	100	400
Total por línea	24 h	—	2,400

Nota. Elaboración propia.

La Tabla 2 establece el tiempo y costo estimado para implementar el sistema en una línea de producción real, incluyendo instalación física, configuración, validación en planta y capacitación a personal operativo.

Tabla 3.

Equipos requeridos.

Equipo / Componente	Cant.	Costo Unit. USD	Subtotal USD
PLC Siemens S7-1200 CPU 1214C	1	1,200	1,200
Módulo de expansión ET-200SP (DI/DO mix)	4	450	1,800
Fuente de alimentación Siemens 24VDC 10A	1	250	250
Licencia Ignition SCADA (8 tags opc)	1	1,500	1,500
Switch Industrial Profinet/Ethernet	1	300	300
Cableado y accesorios (bornes, canaletas)	—	500	500
Total Equipos	—	—	5,550

Nota. Elaboración propia

La Tabla 3 lista los equipos y componentes esenciales para la implementación física del sistema. Incluye el PLC central, módulos de expansión ET-200 para entradas/salidas adicionales, alimentación industrial, licencia de software SCADA y accesorios de instalación. Estos elementos fueron designados teniendo en cuenta su compatibilidad, posibilidad de ampliación y certeza para garantizar un desempeño óptimo del sistema.

Tabla 4.

Inversión total estimada.

Concepto	Costo USD
Horas hombre desarrollo	18,000
Puesta en marcha (1 línea)	2,400
Equipos	5,550
Total Implementación	25,950

Nota. Elaboración propia.

La Tabla 4 consolida el presupuesto total, sumando mano de obra, puesta en marcha y adquisición de equipos. El valor estimado de 25,950 USD representa la inversión necesaria para implementar el sistema en una línea de producción.

Conclusiones

- La integración de los permisos de proceso, seguridad y arranque demostró ser efectiva para garantizar la continuidad operativa y la protección de equipos y personal.
- La implementación del sistema en Ignition SCADA mejoró significativamente la visibilidad de condiciones operativas, permitiendo un diagnóstico rápido y preciso.
- El control de picos de arranque mediante la lógica de permisos de arranque redujo el riesgo de sobrecargas eléctricas y posibles penalizaciones tarifarias.
- La arquitectura modular propuesta permite la escalabilidad del sistema a múltiples líneas de producción sin modificaciones sustanciales en la lógica.
- La inversión estimada se considera altamente rentable en función de los beneficios en seguridad, eficiencia energética y reducción de tiempos de inactividad. (con un presupuesto ... se pone el precio total)

Recomendaciones

- Implementar el sistema inicialmente en una sola línea de producción, para realizar ajustes necesarios antes de su despliegue por toda la planta. Esta fase ayudaría a identificar posibles fallos, mejorar la programación y minimizar los riesgos.
- Establecer un plan de mantenimiento preventivo del hardware PLC y módulos ET-200, que integra revisiones regulares, limpiezas, actualizaciones y pruebas de desempeño. Es conveniente tener a disponibilidad repuestos críticos y capacitar de manera continua al personal operativo y de mantenimiento en el uso de la interfaz SCADA y procedimientos de arranque seguro.
- Incorporar un sistema de registro histórico de eventos en SCADA para análisis posterior de fallas y optimización de secuencias de operación. También sería beneficioso evaluar la integración futura con sistemas de gestión energética (EMS) para aprovechar el control de arranques y reducir aún más el consumo eléctrico.

Referencias

- Abner, C. (2023). *Ciclo de scan de un PLC*. Carlosabneryt:
<https://carlosabneryt.com/ciclo-de-scan-de-un-plc/>
- Ajiga, D. (2024). *The role of software automation in improving industrial operations and efficiency*. *International Journal of Engineering Research Updates.*, 7(1), 22-35. <https://doi.org/10.53430/ijeru.2024.7.1.0031>
- Albarracín Parra, J. C. (2023). *Implementación de Sistema Batch con Banda*.
<https://virtual.cuautitlan.unam.mx/intar/memoriascimcia/wp-content/uploads/sites/20/2024/12/12-81-85-18.pdf>
- Andrade, L., & Mendez, C. (2022). *Implementación de sistemas SCADA en la supervisión de procesos industriales*.
<https://repositorio.uta.edu.ec/handle/123456789/35500>
- Ayala, K., Rivera, A., & Valle, L. (2021). *Diseño y elaboración de un módulo de entrenamiento PLC utilizando una arquitectura SCADA*. [Tesis de ingeniería, ITCA-FEPADE]. Repositorio Institucional de Redicces.
<http://www.redicces.org.sv/jspui/handle/10972/4507>
- Baldeón Casquete, D. E. (2021). *Migración a Lenguaje SCL y mejoras de programa de planta Festo MPS-500 del laboratorio de fabricación flexible de UPS-G*. Trabajo de Titulación, Guayaquil.
<http://dspace.ups.edu.ec/handle/123456789/21113>
- Biedulski, J. (1 de Septiembre de 2024). *Ladder Logic Basics in Siemens TIA Portal – Conveyor – PLC Programming*. ControlByte:
<https://controlbyte.tech/blog/ladder-logic-basics-conveyor-programming/>
- Carbajal Morán, H. (2021). *Modelo de sistema de recuperación de aguas grises por fotocátalisis solar con nanopartículas de TiO2 para riego de cultivos*.

Universidad Nacional Del Centro Del Perú. Perú: Repositorio Institucional
UNI. <http://hdl.handle.net/20.500.12894/7008>

Davydov, A., Larionov, A., & Nagul, N. (2024). Analysis and Control of Partially Observed Discrete-Event Systems via Positively Constructed Formulas. *Computation*, 12(5), 95. <https://doi.org/10.3390/computation12050095>

Domínguez, A. (7 de Noviembre de 2022). *Programación Multidisciplinar*. <https://www.programacionmultidisciplinar.com/curso-de-tia-portal/ciclo-de-scan-plc-siemens/>

Gálvez, D., & Ricaurte, D. (2022). *Propuesta de Implementación de un Sistema SCADA en una Máquina Empaquetadora Mediante el Uso del Software InduSoft en una Empresa de Alimentos*. Tesis de Maestría en Automatización y Control, Guayaquil. <http://www.dspace.espol.edu.ec/handle/123456789/54268>

Hosseinzadeh, H. (2024). *PLC programming and simulation with TIA Portal software*. Tesis de Maestría, Politecnico di Torino. <https://webthesis.biblio.polito.it/31458/>

INCIBE. (2023). *Configuraciones seguras en dispositivos industriales*. Instituto Nacional de Ciberseguridad: <https://www.incibe.es/incibe-cert/blog/configuraciones-seguras-en-dispositivos-industriales>

Jimenez, C. (2021). *Arquitectura de un PLC*. Colombia: Duoc UC. <https://es.scribd.com/presentation/648482053/Arquitectura-de-Un-PLC>

Kirvesoja, J. (2021). *IT- ja OT-integraatio teollisuusympäristössä [Trabajo de ingeniería, Metropolia Ammattikorkeakoulu, Smart IoT]*. https://www.theseus.fi/bitstream/handle/10024/805422/Kirvesoja_Jonne.pdf;jsessionid=960474F96287A26569EDBF4B8CCCFA46?sequence=2

- Lliguicota, O., & Jiménez, R. (2024). *Diseño e implementación de un sistema SCADA para el monitoreo y supervisión de la producción de camarones en una granja ubicada en la provincia del Guayas-Los Ríos*. Los Ríos: ESPOL. <http://www.dspace.espol.edu.ec/handle/123456789/65798>
- Logicbus. (30 de Noviembre de 2024). *Logicbus Blog* . <https://www.logicbus.com.mx/blog/modbus-tcp-ip/>
- Naranjo Ninasunta, C. V., & Salavarría Moreno, E. J. (2022). *Implementación de un módulo de adquisición de datos para máquinas industriales a través de un plc por conexión ethernet*. Riobamba: Escuela superior politécnica de Chimborazo. <https://dspace.esPOCH.edu.ec/handle/123456789/18981>
- Nuñez, C., Zabala, L., Sánchez, M., & Chamba, J. (2025). *Automatización industrial: estructura, programación y aplicaciones con PLCs*. CIDE. <https://repositorio.cidecuador.org/bitstream/123456789/3167/3/Corregido%20DOI%20Libro%20Automatizacion.pdf>.
- Ordoñez Lara, G., & Yépez Mendez, D. (2023). *Diseño e implementación de un prototipo de PLC*. Trabajo de titulación , Universidad politécnica salesiana, Ecuador. <https://dspace.ups.edu.ec/handle/123456789/28128>
- Petruzella D, F. (2020). *Programmable Logic Controllers (McGraw-Hill (Vol. 6th ed)*. McGraw-Hill Education. <https://doi.org/978-1260363768>
- Pin López, D. A., & Criollo Zárate, F. M. (2024). *Diseño y simulación en interfaz hmi y software tia portal el proceso de producción de aceites lubricantes semi-sintéticos para motocicletas*. Trabajo de titulación , Universidad Politécnica Salesiana , Guayaquil . <http://dspace.ups.edu.ec/handle/123456789/27740>
- Rangel, M., & Vega, D. (2020). *Diseño e implementación de prácticas de Automatización Industrial utilizando el Autómata PLC S7-1200 mediante la*

Planta de Procesos EPC. Guayaquil.
<https://dspace.ups.edu.ec/bitstream/123456789/20092/1/UPS-GT003170.pdf>

Rives Sempere, C. P. (2021). *Programación de una estación robótica*. Trabajo de Fin de Máster , Universidad Miguel Hernández. <http://hdl.handle.net/11000/25460>

SICMA21. (2024). *Pirámide de la Automatización Industrial: Clave para la Industria 4.0*. <https://doi.org/10.1036/0070125821>

Siemens. (2021). GRAPH TIA PORTAL HandsON. SCRIBD.
https://es.scribd.com/document/740089637/EN-GRAPH-TIA-Portal-HandsOn?utm_source=chatgpt.com

Siemens. (2022). *SIMATIC S7-1200/S7-1500: System Manual*. Siemens AG.
https://cache.industry.siemens.com/dl/files/465/36932465/att_106119/v1/s71200_system_manual_en-US_en-US.pdf

SMC International Training. (2023). *Pirámide de Automatización Industrial: Nivel de Campo explicado*. <https://doi.org/10.1036/0070125821>

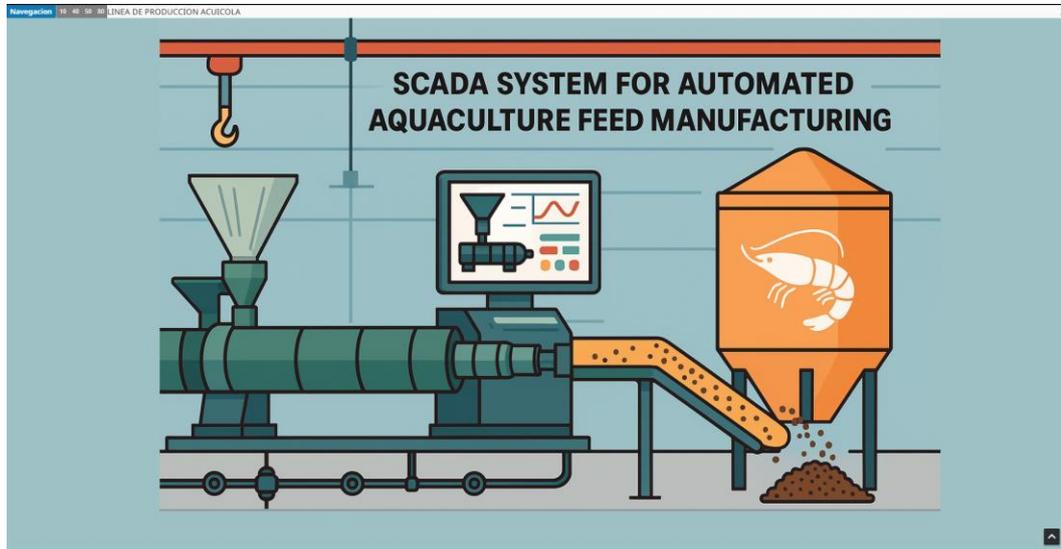
Tabares Ospina, H. (2021). *Texto guía Automatización de procesos industriales*. Medellín, Colombia: Pascual Brazo. ISBN: 978-958-53606-0-0

Vásquez, R., & Herrera, J. ... (2022). *Diseño e implementación de un sistema SCADA para la supervisión de procesos industriales en tiempo real*. Universidad Técnica de Ambato. ISBN: 978-6077073187

ANEXOS

Figura A1

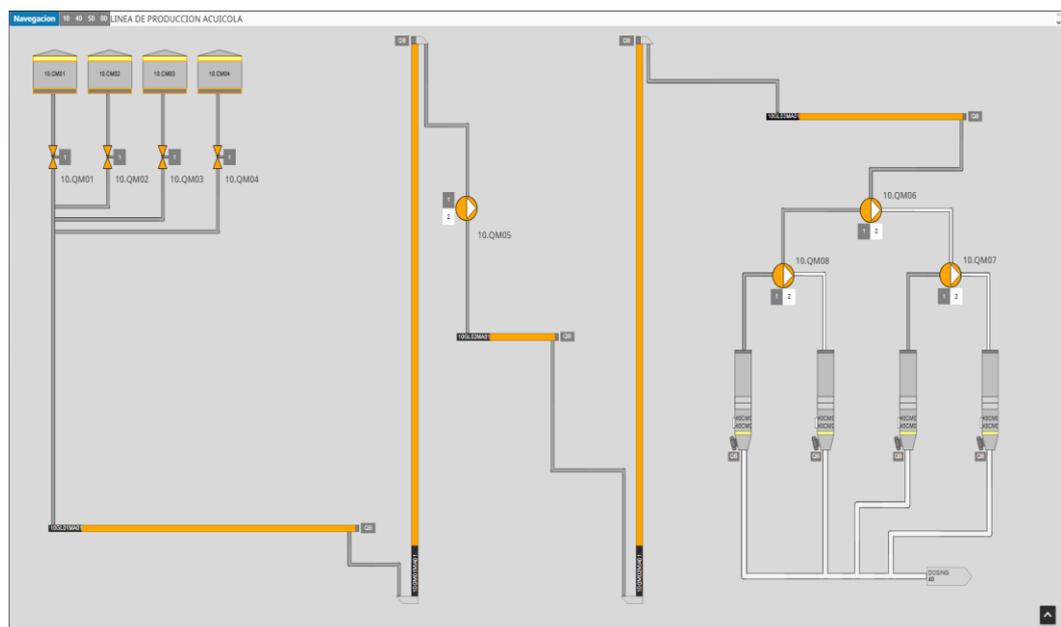
Pantalla de inicio del sistema SCADA



Nota. En esta vista se muestra la interfaz principal del sistema automatizado de la producción de alimento acuícola. Fuente: Elaboración propia.

Figura A2

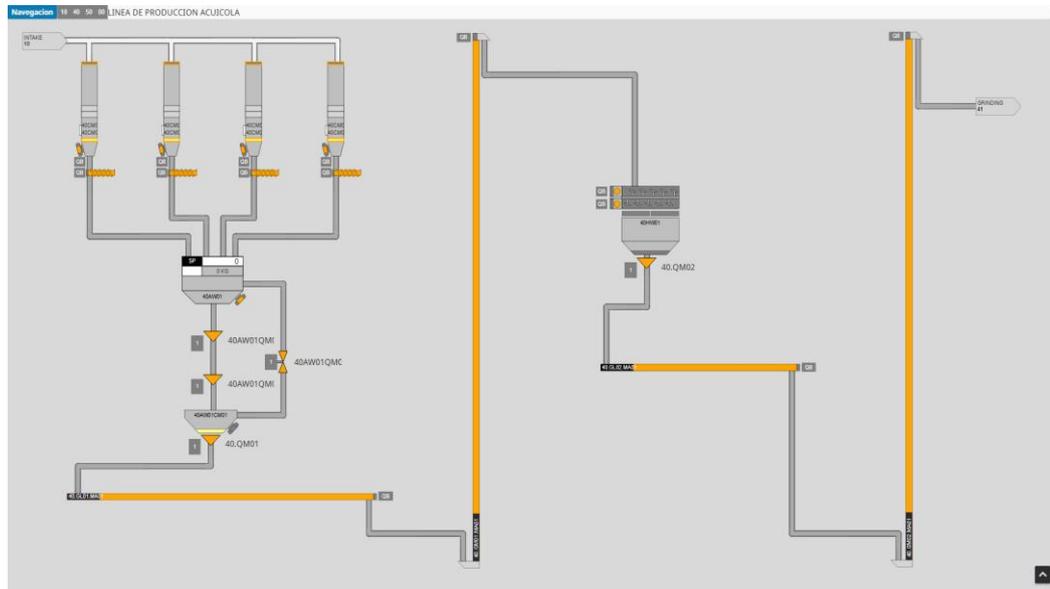
Pantalla de intake



Nota. En esta pantalla se muestra la interfaz del *intake*, correspondiente al inicio del proceso de producción acuícola. Fuente: Elaboración propia.

Figura A3

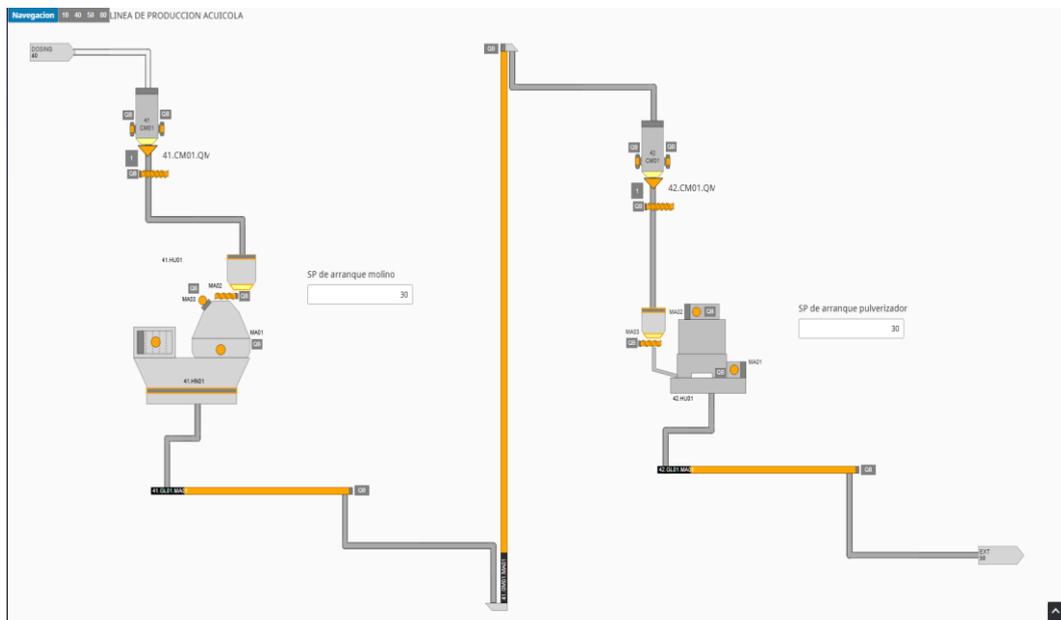
Pantalla de Dosing y Mixing



Nota. En esta pantalla se observa la sección de dosificación y mezclado de la materia prima. Fuente: Elaboración propia.

Figura A4

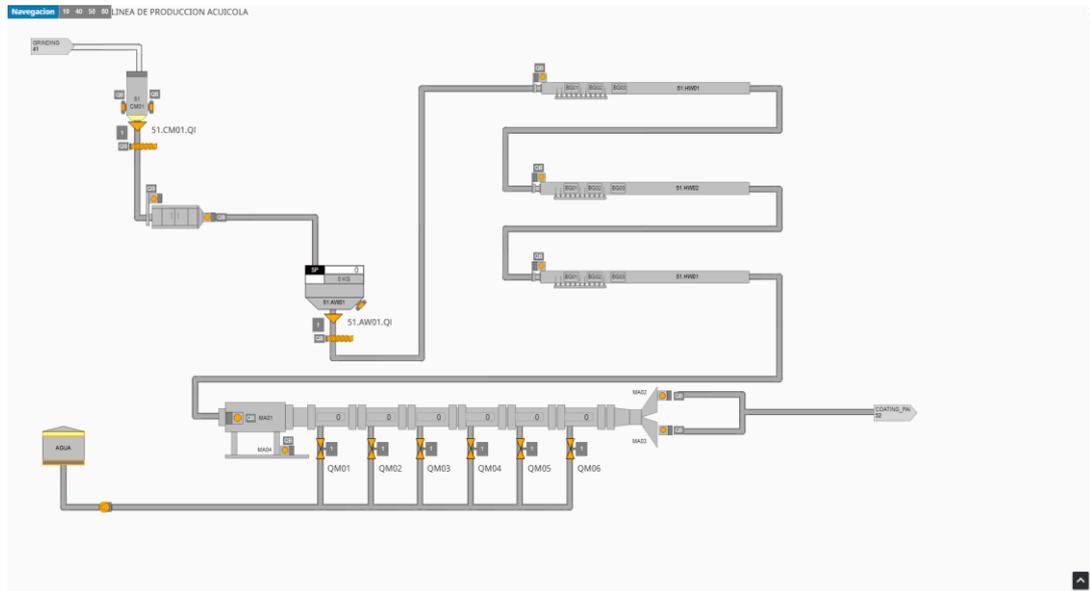
Pantalla de Pulverizado y Molienda



Nota. Esta pantalla demuestra el proceso de pulverizado y molienda, donde la materia prima se reduce hasta obtener el tamaño adecuado dentro del sistema. Fuente: Elaboración Propia.

Figura A5

Pantalla de extrusión

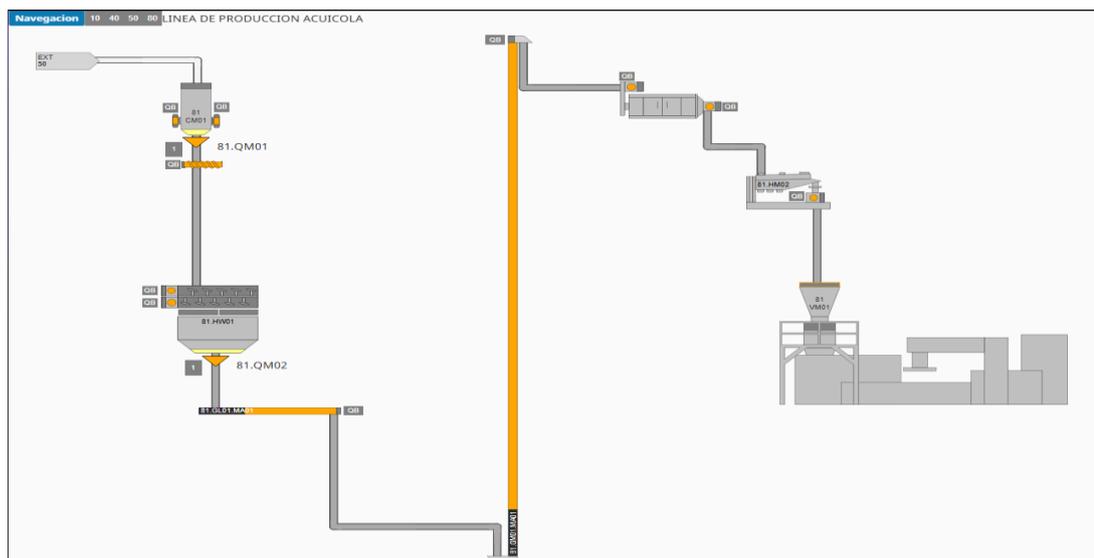


Nota. En esta pantalla se observa la etapa de extrusión, donde la mezcla procesada es transformada en pellets mediante el extrusor en la línea de producción acuícola.

Fuente: Elaboración Propia.

Figura A6

Pantalla de Coating y Packing



Nota. En esta pantalla se muestra la fase de Coating y Empaquetado, donde los pellets son recubiertos y preparados para su almacenamiento o distribución. Fuente:

Elaboración Propia.

DECLARACIÓN Y AUTORIZACIÓN

Yo, **Rivera Camacho, Nahomy Sugey** con C.C: # 0951232008 autora del trabajo de titulación: **Diseño de un sistema automatizado de permisos aplicados en fábricas de balanceado acuícola**, previo a la obtención del título de **Ingeniera en Electrónica y Automatización** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 4 de septiembre de 2025

f. _____

Nombre: **Rivera Camacho, Nahomy Sugey**

CC.: **0951232008**



REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA

FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

TEMA Y SUBTEMA:	Diseño de un sistema automatizado de permisos aplicados en fábricas de balanceado acuícola.		
AUTOR(ES)	Rivera Camacho, Nahomy Sugey		
REVISOR(ES)/TUTOR(ES)	Ing. Bonilla Sánchez, Ronnie Alexander, MSc.		
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil		
FACULTAD:	Facultad de Educación Técnica para el Desarrollo		
CARRERA:	Ingeniería en Electrónica y Automatización		
TÍTULO OBTENIDO:	Ingeniero en Electrónica y Automatización		
FECHA DE PUBLICACIÓN:	4 de septiembre de 2025	No. DE PÁGINAS:	81
ÁREAS TEMÁTICAS:	Fábrica de balanceado acuícola, automatización con sistema de permisos		
PALABRAS CLAVES/ KEYWORDS:	Permisos de proceso, Seguridad industrial, Arranque automático, Automatización, SCADA, PLC		
RESUMEN			
<p>El objetivo de este proyecto de titulación se determinó como el desarrollo de diseñar un sistema de permisos de proceso, seguridad y arranque en una planta automatizada de producción de alimento acuícola. Para esto se utilizó un controlador lógico programable (PLC S7-1500) y el software de SCADA Ignition, lo que permitió probar y verificar la lógica de control utilizando PLCSIM, sin ninguna alteración en el proceso. El sistema desarrollado estableció condiciones de proceso, interbloqueos de seguridad y restricciones de arranque, garantizando que los equipos no deben funcionar a menos que se cumplan las condiciones establecidas. De este modo, se redujeron los riesgos eléctricos, se evitó la mala operación de los equipos y se mejoró la seguridad de los operadores. A su vez, se diseñó una interfaz gráfica para el operador con ventanas eventuales tipo popup, lo que ayudó a la supervisión de estados y las condiciones en tiempo real, llevando a una mejor interacción entre los operadores y el sistema. Los resultados obtenidos indicaron una reducción significativa en las fallas de línea, sin necesidad de una prueba de arranque y que el tiempo para iniciar la línea estaba asegurado, siendo una alternativa técnica y económicamente viable.</p>			
ADJUNTO PDF:	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
CONTACTO CON AUTOR/ES:	Teléfono: +593960347198	E-mail: nahomy.rivera@cu.ucsg.edu.ec	
CONTACTO CON LA INSTITUCIÓN (COORDINADOR DEL PROCESO UTE):	Nombre: Ubilla González Ricardo Xavier		
	Teléfono: +593999528515		
	E-mail: ricardo.ubilla@cu.ucsg.edu.ec		
SECCIÓN PARA USO DE BIBLIOTECA			
Nº. DE REGISTRO (en base a datos):			
Nº. DE CLASIFICACIÓN:			
DIRECCIÓN URL (tesis en la web):			