

UNIVERSIDAD TÉCNICA DE AMBATO



FACULTAD DE TECNOLOGÍAS DE LA INFORMACIÓN, TELECOMUNICACIONES E INDUSTRIAL

MAESTRÍA EN AUTOMATIZACIÓN Y SISTEMAS DE CONTROL

Tema: “INDUSTRIA 4.0 PARA LA MONITORIZACIÓN DE UN PROCESO INDUSTRIAL”

Trabajo de Investigación: Previo a la obtención del Grado Académico de Magister en Automatización y Sistemas de Control

AUTOR: Ingeniero Mauricio Xavier López Flores

DIRECTOR: Ingeniero Marcelo Vladimir García Sánchez, PhD.

Ambato – Ecuador

Junio – 2019

A la Unidad Académica de Titulación de la Facultad de Tecnologías de la Información, Telecomunicaciones e Industrial.

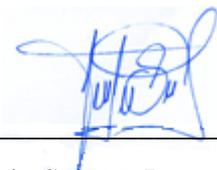
El Tribunal receptor del Trabajo de Investigación presidido por la Ingeniera Elsa Pilar Urrutia Urrutia Mg., e integrado por los señores: Ingeniero Germán Patricio Encalada Ruiz Mg., Ingeniero Franklin Wilfrido Salazar Logroño Mg. e Ingeniero Carlos Diego Gordón Gallegos Mg., designados por la Dirección de Postgrado de la Universidad Técnica de Ambato, para receptor el Trabajo de Investigación con el tema: “**INDUSTRIA 4.0 PARA LA MONITORIZACIÓN DE UN PROCESO INDUSTRIAL**”, elaborado y presentado por el Ingeniero Mauricio Xavier López Flores, para optar por el Grado Académico de Magister en Automatización y Sistemas de Control; una vez escuchada la defensa oral del Trabajo de Investigación el Tribunal aprueba y remite el trabajo para uso y custodia en las bibliotecas de la UTA.



Ing. Elsa Pilar Urrutia Urrutia Mg.
Presidente del Tribunal



Ing. Germán Patricio Encalada Ruiz Mg.
Miembro del Tribunal



Ing. Franklin Wilfrido Salazar Logroño Mg.
Miembro del Tribunal



Ing. Carlos Diego Gordón Gallegos Mg.
Miembro del Tribunal

AUTORÍA DEL TRABAJO DE INVESTIGACIÓN

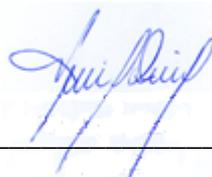
La responsabilidad de las opiniones, comentarios y críticas emitidas en el Trabajo de Investigación presentado con el tema: “**INDUSTRIA 4.0 PARA LA MONITORIZACIÓN DE UN PROCESO INDUSTRIAL**”, le corresponde exclusivamente al Ingeniero Mauricio Xavier López Flores, Autor, bajo la dirección del Ingeniero Marcelo Vladimir García Sánchez PhD., Director del Trabajo de Investigación; y el patrimonio intelectual a la Universidad Técnica de Ambato.



Ingeniero Mauricio Xavier López Flores

C.C. 1804062915

AUTOR



Ingeniero Marcelo Vladimir García Sánchez, PhD.

C.C. 1803543055

DIRECTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que el Trabajo de Investigación, sirva como documento disponible para su lectura, consulta y procesos de investigación, según las normas de la institución.

Cedo los derechos de mi Trabajo, con fines de difusión pública, además apruebo la reproducción de este, dentro de las regulaciones de la Universidad.



Ingeniero Mauricio Xavier López Flores

C.C. 1804062915

AUTOR

ÍNDICE GENERAL

Portada.....	i
A la Unidad Académica de Titulación	ii
AUTORÍA DEL TRABAJO DE INVESTIGACIÓN	iii
DERECHOS DE AUTOR	iv
ÍNDICE GENERAL.....	v
ÍNDICE DE FIGURAS	ix
ÍNDICE DE TABLAS.....	xi
ÍNDICE DE ECUACIONES	xii
AGRADECIMIENTO	xiii
DEDICATORIA.....	xiv
RESUMEN EJECUTIVO.....	xv
EXECUTIVE SUMMARY	xvi
INTRODUCCIÓN.....	1
CAPITULO I.....	2
EL PROBLEMA DE INVESTIGACIÓN	2
1.1 Tema de investigación	2
1.2 Planteamiento del problema.....	2
1.2.1 Contextualización	2
1.2.2 Análisis crítico.....	4
1.2.3 Prognosis	4
1.2.4 Formulación del problema.....	5
1.2.5 Preguntas directrices.....	5
1.2.6 Delimitación de la investigación	5
1.3 Justificación	6
1.4 Objetivos	6
1.4.1 Objetivo general	6
1.4.2 Objetivos específicos.....	6
CAPÍTULO II.....	8
MARCO TEÓRICO	8
2.1 Antecedentes investigativos.....	8
2.2 Fundamentación filosófica.....	10
2.3 Fundamentación legal	10
2.4 Categorías fundamentales	11

2.4.1	Gráficos de inclusión interrelacionados	11
2.4.2	Constelación de ideas de la variable independiente	12
2.4.3	Constelación de ideas de la variable dependiente	13
2.5	Fundamentación teórica	14
2.5.1	Revolución industrial.....	14
2.5.2	Industria 4.0	16
2.5.3	Internet de las Cosas (IoT).....	16
2.5.4	Sistemas Ciber-físicos (CPS) en la industria.....	17
2.5.5	Interfaz Hombre Máquina (HMI)	18
2.5.6	Controlador Lógico Programable (PLC)	19
2.5.7	Open source	19
2.5.8	Comunicación Máquina a Máquina (M2M).....	19
2.5.9	JavaScript.....	20
2.5.10	Control de procesos	20
2.5.11	Plataforma Internet de las Cosas (IoT)	22
2.5.12	Protocolo OPC UA	22
2.5.13	Protocolo AMQP	23
2.5.14	Protocolo CoAP.....	24
2.5.15	Protocolo MQTT	24
2.6	Hipótesis	25
2.7	Señalamiento de variables de la hipótesis.....	25
2.7.1	Variable independiente	25
2.7.2	Variable dependiente	25
CAPÍTULO III		26
METODOLOGÍA.....		26
3.1	Enfoque.....	26
3.2	Modalidad básica de la información	26
3.3	Nivel o tipo de investigación	27
3.4	Población y muestra.....	27
3.5	Operacionalización de variables	28
3.5.1	Variable independiente	28
3.5.2	Variable dependiente	29
3.6	Recolección de información	30
3.7	Procesamiento y análisis	31

3.8	Desarrollo del proyecto.....	31
CAPÍTULO IV		32
ANÁLISIS E INTERPRETACIÓN DE RESULTADOS		32
4.1	Desarrollo del proyecto.....	32
CAPÍTULO V.....		34
CONCLUSIONES Y RECOMENDACIONES		34
5.1	Conclusiones.....	34
5.2	Recomendaciones	35
CAPÍTULO VI.....		36
PROPUESTA DE INVESTIGACIÓN		36
6.1	Tema de la propuesta	36
6.2	Datos informativos.....	36
6.2.1	Ejecutada por:.....	36
6.2.2	Beneficiarios:	36
6.2.3	Ubicación:	36
6.2.4	Equipo responsable:	36
6.2.5	Financiamiento:.....	36
6.3	Antecedentes de la propuesta.....	37
6.4	Justificación	41
6.5	Objetivos	42
6.5.1	Objetivo General	42
6.5.2	Objetivos Específicos.....	42
6.6	Análisis de Factibilidad.....	42
6.6.1	Factibilidad técnica	42
6.6.2	Factibilidad operativa.....	42
6.6.3	Factibilidad económica	43
6.7	Fundamentación Científico Técnica	43
6.7.1	Comparación y selección de protocolo	43
6.7.2	MQTT.....	44
6.7.3	Broker Mosquitto	45
6.7.4	Servidor HTTP Apache.....	45
6.7.5	PHPMyAdmin.....	45
6.7.6	Python.....	46
6.7.7	Controlador Lógico Programable (PLC) Siemens S7 1200.....	47

6.7.8	Regulador por Control Difuso.....	48
6.7.9	Factory I / O	49
6.7.10	MySQL.....	49
6.7.11	Wireshark	50
6.8	Desarrollo de la propuesta	51
6.8.1	Arquitectura de la propuesta	51
6.8.2	Selección de equipos y programas utilizados.....	51
6.8.3	Programación PLC y FACTORY IO	54
6.8.3.1	Control mediante Lógica Difusa (Fuzzy Logic).....	57
6.8.3.2	Control mediante Regulador PID	60
6.8.4	PLC y RASPBERRY SNAP 7	62
6.8.5	Aplicación del Protocolo MQTT.....	63
6.8.6	Topología	67
6.8.7	Topics.....	67
6.8.8	Comunicación, publicación y suscripción.....	68
6.8.9	Calidad del Servicio (QoS)	68
6.8.10	Seguridad y Puerto	69
6.8.11	WebSockets.....	70
6.8.12	Desarrollo de Página Web.....	72
6.8.13	Análisis de Tráfico de Datos	78
6.8.14	Tiempos de Respuesta y Carga del Servidor (SBC)	82
6.8.15	Comprobación de la Hipótesis de la propuesta	85
6.9	Conclusiones y Recomendaciones	88
6.9.1	Conclusiones	88
6.9.2	Recomendaciones.....	89
BIBLIOGRAFÍA		91
ANEXOS		97
ANEXO 1: Modelado matemático del sistema.....		97
ANEXO 2: Tablas y gráficas de reglas de inferencia y funciones de membresía		99
ANEXO 3: Cálculos de sintonización del controlador PID.....		104
ANEXO 4: Tabla de variables y Topics del sistema		108
ANEXO 5: Detalle configuración IP pública		110
ANEXO 6: Código de Programa		111

ÍNDICE DE FIGURAS

Figura 1: Árbol del Problema.....	3
Figura 2: Categorías Fundamentales	11
Figura 3: Constelación de ideas, variable independiente	12
Figura 4: Constelación de ideas: variable dependiente	13
Figura 5: Diagrama revoluciones industriales sucesivas.....	14
Figura 6: Esquema referencial Interfaz Hombre Máquina (HMI)	18
Figura 7: Niveles de un sistema de control de procesos	21
Figura 8: Estructura general Protocolo AMQP	24
Figura 9: Arquitectura MQTT	44
Figura 10: Controlador Lógico Programable (PLC) S7 1200 1214 AC/DC/Rly	47
Figura 11: Control proceso Factory I/O – TIA Portal	49
Figura 12: Arquitectura de la plataforma	51
Figura 13: Interconexión Factory I/O y Controlador	54
Figura 14: Bloque de programación principal.....	55
Figura 15: Tabla de variables de programación	55
Figura 16: Layout planta en Factory IO	56
Figura 17: Configuración de entradas salidas PLC – Factory IO	56
Figura 18: Modelo matemático del sistema	57
Figura 19: Modelo difuso Fuzzy Logic Toolbox	58
Figura 20: Visualización de evaluación de reglas	59
Figura 21: Gráfica superficie de decisión	59
Figura 22: Programación PID_Compact en TIA V13.....	60
Figura 23: Estructura e interconexión de la plataforma	64
Figura 24: Esquema protocolo MQTT – TCP/IP	65

Figura 25: Confirmación librerías MQTT para lenguaje en C.....	66
Figura 26: Puesta en servicio de bróker MQTT Mosquitto	66
Figura 27: Topología de comunicación protocolo MQTT	67
Figura 28: Arquitectura protocolo MQTT sobre WebSockets.....	70
Figura 29: Configuración WebSockets config.mk	71
Figura 30: Estado de configuración final bróker.....	72
Figura 31: Función botón START página Web	72
Figura 32: Función movimiento caja – Página Web	73
Figura 33: Página de ingreso a Panel de Control	73
Figura 34: Panel de control operador	75
Figura 35: Panel de control Ingeniero	75
Figura 36: Estructura del análisis de tráfico de datos.....	78
Figura 37: Visualización tráfico de datos al iniciar bróker	78
Figura 38: Visualización tráfico de datos Wireshark con el proceso en marcha	79
Figura 39: Interacciones de red en Wireshark, detalle de paquete 2078 Wi-Fi	79
Figura 40: Ancho de Banda (bits/seg) del protocolo de comunicación	80
Figura 41: Detalle de paquetes en transacción publicación/suscripción	81
Figura 42: Tiempo de procesamiento paquetes.....	81
Figura 43: Tiempo desde solicitud, Wireshark	83

ÍNDICE DE TABLAS

Tabla 1: Características de las revoluciones industriales	15
Tabla 2: Operacionalización de la variable independiente	28
Tabla 3: Operacionalización de la variable dependiente	29
Tabla 4: Plan de recolección de información	30
Tabla 5: Contraste entre posibles controladores del proceso	52
Tabla 6: Comparativa entre SBC y Arduino	52
Tabla 7: Detalle equipos y programas utilizados	53
Tabla 8: Detalle direcciones IP de equipos	53
Tabla 9: Contraste control difuso y PID	61
Tabla 10: Características de norma ISA-SP101 y UNE-EN ISO 9241	74
Tabla 11: Detalle acceso y restricción Panel operador	76
Tabla 12: Detalle acceso y restricción Panel ingeniero	77
Tabla 13: Detalle cuantitativo análisis de tráfico de Red	82
Tabla 14: Tiempos de respuesta de la comunicación	83
Tabla 15: Información prueba de funcionamiento Raspberry Cliente	84
Tabla 16: Información prueba de funcionamiento Raspberry Servidor	85
Tabla 17: Detalle frecuencias observadas del sistema	86
Tabla 18: Detalle frecuencias esperadas del sistema	87

ÍNDICE DE ECUACIONES

Ecuación 1: Modelo matemático del tanque	57
Ecuación 2: Chi-cuadrado	86
Ecuación 3: Frecuencias esperadas	86
Ecuación 4: Grados de libertad.....	86
Ecuación 5: Parámetro p	86

AGRADECIMIENTO

Mi agradecimiento al Altísimo Todo Poderoso por darme la vida y la salud, por bendecirme diariamente y guiarme en cada paso que doy.

A mis padres por su apoyo perdurable, sus consejos, experiencia y ejemplo fueron una guía para cumplir mis objetivos planteados; enseñándome que el éxito se alcanza con sacrificio.

Al Ing. Marcelo García PhD. su guía en el tutorial, me han permitido efectuar exitosamente el proyecto, su paciencia y apoyo fueron imprescindibles.

Xavier López Flores

DEDICATORIA

A Dios, fuente de sabiduría.

A mi familia, por su amor y apoyo incondicional dedicándome tiempo, paciencia y comprensión, inculcando en mí valores que permitieron mi desarrollo personal.

A mi novia Carolina por su cariño, dulzura y lealtad durante el postgrado.

Xavier López Flores

UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE TECNOLOGÍAS DE LA INFORMACIÓN,
TELECOMUNICACIONES E INDUSTRIAL**

MAESTRÍA EN AUTOMATIZACIÓN Y SISTEMAS DE CONTROL

TEMA:

**INDUSTRIA 4.0 PARA LA MONITORIZACIÓN DE UN PROCESO
INDUSTRIAL**

AUTOR: Ingeniero Mauricio Xavier López Flores

TUTOR: Ingeniero Marcelo Vladimir García Sánchez PhD.

FECHA: 26 de abril, 2019

RESUMEN EJECUTIVO

El presente trabajo de investigación centra su estudio en actualizar el modelo de monitorización de procesos, desarrollando como propuesta de solución, una plataforma basada en Industria 4,0 para la monitorización de un proceso industrial, mediante la aplicación de Internet de las Cosas (IoT) con el protocolo de comunicación MQTT. El protocolo aplicado opera mediante una arquitectura cliente-servidor, utiliza comunicación sobre WebSockets y el transporte de mensajería se filtra por eventos de publicación/suscripción

El proceso industrial es controlado mediante un Controlador Lógico Programable (PLC) y simulado en el software Factory I/O. La plataforma de comunicación consta de un bróker y dos clientes y su funcionamiento es demostrado mediante pruebas de funcionamiento y el análisis de tráfico de datos mediante la herramienta Wireshark.

La plataforma de comunicación desarrollada a través del protocolo MQTT, permite una monitorización de procesos en línea mediante una página web, su utilización y aplicabilidad es sencilla, además posee características de escalabilidad, calidad de entrega, seguridad y sobre todo la comunicación en entornos restringidos.

Descriptores: Protocolo de comunicación, MQTT, PLC, Raspberry Pi, monitorización, Industria 4.0, Internet de las Cosas (IoT), QoS, Websockets, WiFi,

UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE TECNOLOGÍAS DE LA INFORMACIÓN,
TELECOMUNICACIONES E INDUSTRIAL**

MAESTRÍA EN AUTOMATIZACIÓN Y SISTEMAS DE CONTROL

THEME:

**INDUSTRY 4.0 FOR THE MONITORING OF AN INDUSTRIAL
PROCESS**

AUTHOR: Engineer Mauricio Xavier López Flores

DIRECTED BY: Engineer Marcelo Vladimir García Sánchez PhD.

DATE: April 26, 2019

EXECUTIVE SUMMARY

The present research work focuses its study on updating the process monitoring model, as a solution proposal, a platform based on industry 4.0 for the monitoring of an industrial process, through the application of Internet of Things (IoT) with the MQTT communication protocol. The applied protocol operates through client-server architecture, uses communication over WebSockets, and messaging transport is filtered by publish / subscribe events

The industrial process is controlled by a Programmable Logic Controller (PLC) and simulated in the Factory IO software. The communication platform consists of a broker and two clients and its operation is through operational tests and the analysis of data traffic using the Wireshark tool.

The communication platform developed through the MQTT protocol allows monitoring of online processes through a web page, its use and applicability is simple, it also has characteristics of scalability, quality of delivery, security and above all communication in restricted network.

Descriptors: Communication protocol, MQTT, PLC, Raspberry Pi, monitoring, Industry 4.0, Internet of Things (IoT), QoS, WebSockets, WiFi.

INTRODUCCIÓN

El presente trabajo de investigación tiene como tema: “INDUSTRIA 4.0 PARA LA MONITORIZACIÓN DE UN PROCESO INDUSTRIAL”. El mismo analiza la aplicación del protocolo de comunicación MQTT y el desarrollo de su plataforma para mejorar el control y visualización de procesos industriales, mediante la aplicación del enfoque Industria 4,0 o Internet de las Cosas (IoT). El contenido del trabajo consta de los siguientes capítulos:

Capítulo I, El Problema, se detalla el objeto de la investigación, mediante su planteamiento, contextualización, análisis crítico, delimitación y justificación. Al igual se determinan los objetivos necesarios para su posterior estudio.

Capítulo II, Marco Teórico, se desarrolla el análisis de información bibliográfica y contenidos teóricos referentes a la investigación y que respalden su aplicación e implementación. El capítulo además consta de: Antecedentes investigativos, fundamentación, formulación de hipótesis y su respectivo señalamiento de variables.

Capítulo III, Metodología, describe la sistemática y técnicas a cumplir para el desarrollo del proyecto, su disposición está conformada por el detalle de: enfoque investigativo; modalidad, nivel y/o tipo de investigación; técnicas e instrumentos necesarios; plan de recolección y procesamiento de información, entre otros.

Capítulo IV, Análisis e Interpretación de la Información, en este capítulo se analiza, resuelve y determina acciones y posibles consecuencias de la investigación, temática necesaria para el planteamiento de deducciones y posibles propuestas de solución.

Capítulo V, Conclusiones y Recomendaciones, resultado del análisis completo de capítulos anteriores y obtenidos en base a los objetivos planteados, éstas permitirán establecer lineamientos y metas para el desarrollo de la propuesta de solución necesaria para el proyecto.

Capítulo VI, La Propuesta, capítulo en el que se define la solución a desarrollar para la problemática presentada inicialmente; planteando y describiendo objetivos, teoría, justificación, así como el desarrollo completo de propuesta, pruebas de funcionamiento, conclusiones y recomendaciones complementarias.

CAPITULO I

EL PROBLEMA DE INVESTIGACIÓN

1.1 Tema de investigación

Plataforma basada en el concepto de Industria 4.0 para la monitorización de un proceso industrial.

1.2 Planteamiento del problema

1.2.1 Contextualización

En el entorno actual de competencia global, desarrollo tecnológico e innovación, las empresas, sobre todo de manufactura, se ven forzadas a reconfigurar sus procesos. La Industria 4.0 y la manufactura inteligente son parte de esta transformación, en la que tecnologías de fabricación e información se han integrado para crear innovadores sistemas de manufactura, gestión y formas de hacer negocios, que permiten optimizar los procesos de fabricación, alcanzar una mayor flexibilidad, eficiencia y generar una propuesta de valor para sus clientes, así como responder de forma oportuna a las necesidades de su mercado (Ynzunza, Izar, Bocarando, Aguilar, & Larios, 2017).

Países como Dinamarca, Estados Unidos, Reino Unido y Corea del Sur disponen de estrategias para situar sus industrias en un nivel competitivo global, mediante el desarrollo de sus herramientas de fabricación inteligente. Destacando entre todos estos la iniciativa alemana denominada y actualmente conocida como “Industria 4.0”. Si bien es cierto, ningún país de Latinoamérica consta dentro de las grandes potencias mencionadas, pero la actualización y transformación industrial no sólo esta influenciada por el marco económico interno del país, la investigación, aplicación de nuevas tecnologías, iniciativa manufacturera y comercial son algunas de las herramientas que podemos tomar y aplicar para solventar las cambiantes y muy exigentes necesidades del cliente. Los nuevos mercados se establecen en la personalización y la creación de nuevos productos y servicios innovadores. Exigiendo calidad en los productos, pagando por la experiencia o el servicio más que por el producto en sí. Cualidades que ésta cuarta revolución industrial provoca debido al impacto de las tecnologías digitales aplicadas a los modelos de negocio de la industria (Val Román & Val Román, 2017).

Latinoamérica y Ecuador, sin embargo, cuentan con la oportunidad de aprovechar las posibilidades que internet y la digitalización brindan para evolucionar y crecer. La investigación, colaboración de todos los agentes implicados y la aplicación de nuevas tecnologías permiten realmente un cambio en el modelo productivo basado en internet. El desafío es considerable, pero la innovación totalmente necesaria.

De acuerdo a lo mencionado anteriormente, y con el afán de proporcionar una herramienta adecuada e innovadora para la monitorización de procesos, planificación y toma de decisiones, además de la investigación y aplicación de tecnología vanguardista en la industria sectorial y nacional, el problema central de la investigación se centra en el: “deficiente proceso de monitorización de procesos industriales”.

Árbol del problema

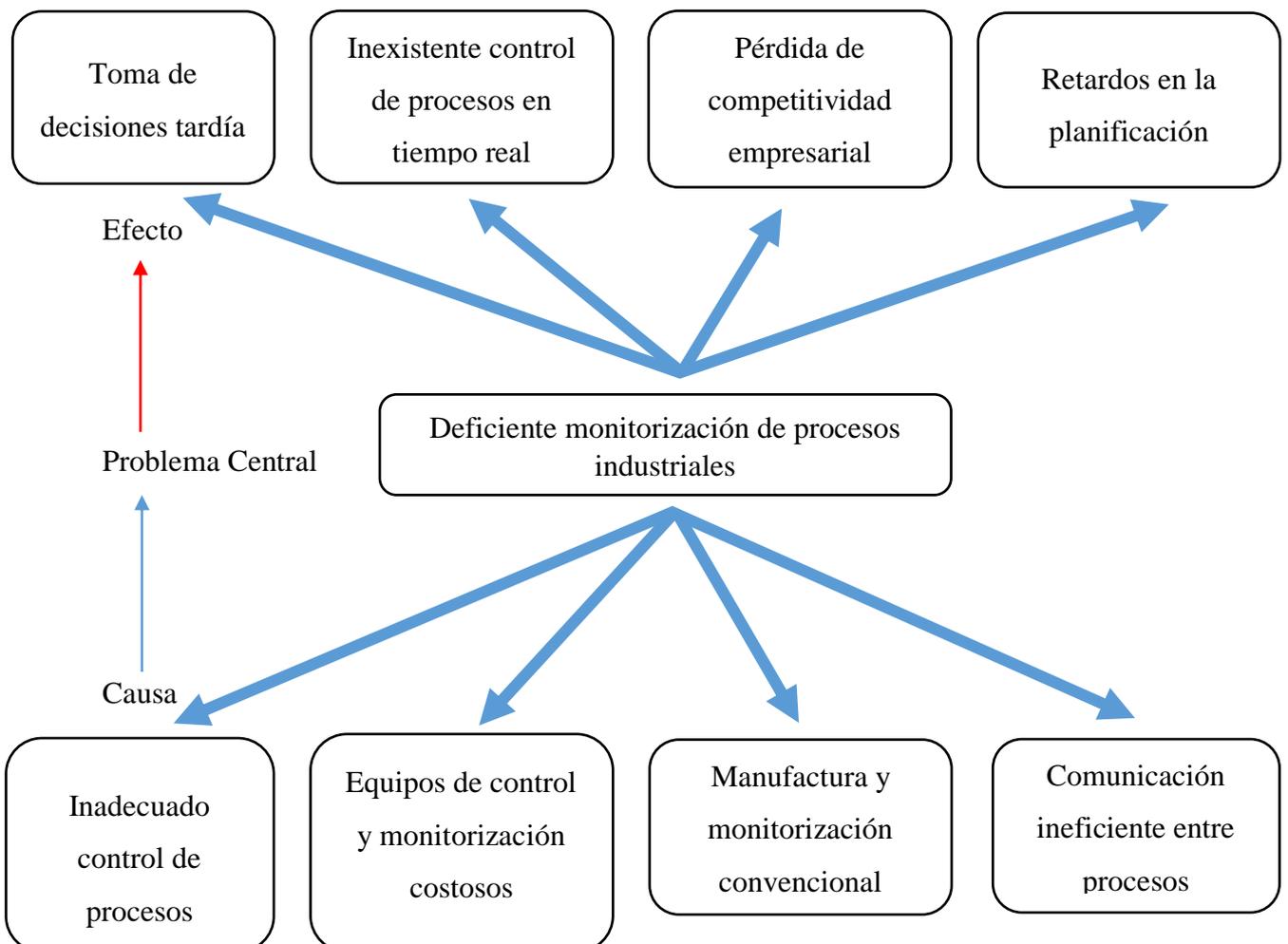


Figura 1: Árbol del Problema
Elaborado por: El Investigador

1.2.2 Análisis crítico

El desarrollo de la relación causa efecto del árbol de problema descrito en el Gráfico N. 1, se presenta a continuación:

La industria ecuatoriana en general mantiene un inadecuado control de procesos, situación que provoca tomar tardías decisiones y muchas de estas desenfocadas de la realidad de los procesos, evitando un control adecuado que permita el desarrollo manufacturero y comercial.

Por otra parte, el elevado costo de equipos especializados en comunicación y monitorización no permite mantener esta característica indispensable en los diferentes procesos industriales, causando fallas y retrasos en entregas por falta de información entre niveles de operación.

El incremento de la demanda de productos a nivel general permite la prematura adquisición de maquinaria y el aumento de monina de trabajadores, para cumplir con la oferta requerida, pero los métodos de manufactura y monitorización convencional no permiten mantener un índice competitivo empresarial, característica necesaria en el cambiante y exigente mercado actual.

Uno de los errores mayormente cometidos en los procesos de producción en general es, la escasa e ineficiente comunicación entre procesos productivos y estos a su vez con el nivel administrativo, circunstancia que evita una adecuada planificación de procesos y gestión para alcanzar los objetivos empresariales y en general el concepto tan anhelado de mejora continua.

1.2.3 Prognosis

De mantenerse el actual sistema de monitorización de procesos industriales, se ahondará en limitaciones de fabricación, toma de decisiones y planificación, efectos que cada vez serán más evidentes y convergerán ineludiblemente en pérdidas económicas y de clientes, impidiendo que los procesos en general y las empresas se mantengan competitivas en el mercado nacional y más aún de ingresar competencia internacional, sectores que siempre están a la vanguardia en las nuevas tecnologías de fabricación e innovación manufacturera.

1.2.4 Formulación del problema

¿Cómo incide el enfoque de Industria 4,0 para optimar la monitorización de un proceso industrial?

1.2.5 Preguntas directrices

¿Qué estructura, protocolo o software es aplicable al enfoque de Industria 4,0?

¿Qué características posee y como opera el actual método de monitorización de los procesos industriales?

¿Existen plataformas validadas de diseños en Industria 4,0 para la monitorización de procesos industriales?

1.2.6 Delimitación de la investigación

Área: Ingeniería

Línea de investigación: Manufactura y automatización

Programa de investigación: Sistemas de fabricación y producción automática

Delimitación espacial

La investigación tendrá lugar en el Laboratorio de PLC's de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato.

Delimitación temporal

La investigación se efectuará en los 6 (seis) meses posteriores a la aprobación del proyecto por parte del Honorable Consejo de Postgrado de la Facultad.

Unidades de observación

Leguajes de programación; recolección y registro de información; además de pruebas, validación y aprobación mediante caso de estudio de la plataforma a diseñar.

1.3 Justificación

El **interés** por investigar el tema propuesto nace con el estudio y potencial aplicación del concepto de Industria 4.0, técnica poco y nada aplicada en los sectores productivos industriales ecuatorianos, la misma utiliza conceptos básicos, aplicándolos a procesos pequeños, medianos y grandes a fin de que estos posean características que permitan comunicación, monitorización y control de procesos, permitiendo un nivel de flexibilidad que se refleje en índices favorables de productividad, eficiencia y calidad.

El proyecto de investigación es de **importancia**, ya que busca brindar una herramienta logística y técnica que permita mejorar en general los procesos de monitorización en las PyMES, explorar el entorno tecnológico, así como sus implicaciones y perspectivas futuras. Iniciativas sobre Industria 4.0 con el objetivo que la empresa o empresas que empleen correctamente la metodología, alcancen un sobresaliente nivel competitivo en el complejo mercado industrial a nivel provincial, nacional e internacional.

Existe **factibilidad** para el pleno desarrollo de este proyecto, gracias al conocimiento del tema adquirido durante la carrera estudiantil de pregrado y posgrado, así también de la aplicación técnica de programas informáticos afines al contexto. Además de disposición e interés por indagar métodos y propuestas de desarrollo, acceso a información bibliográfica, recursos tecnológicos y económicos necesarios para el cumplimiento del proyecto. Finalmente, los **beneficiarios** del proyecto serán en general las industrias pequeñas, medianas o grandes que pretendan o presenten la intención de implementar este proyecto a escala industrial, quienes contarán con la guía y métodos adecuados que permitan una mejor monitorización y control de procesos.

1.4 Objetivos

1.4.1 Objetivo general

Diseño de plataforma basada en el concepto de Industria 4.0 para la monitorización de un proceso industrial.

1.4.2 Objetivos específicos

- Investigar el Estado del Arte sobre estructuras, protocolos y software aplicable a Industria 4.0

- Diseñar e implementar una plataforma de Industria 4.0 para procesos industriales
- Validar la plataforma de Industria 4.0 mediante pruebas y caso de estudio

CAPÍTULO II

MARCO TEÓRICO

2.1 Antecedentes investigativos

Efectuada la indagación en artículos, trabajos y proyectos afines al tema planteado, a continuación, se presentan los más relevantes:

El concepto de Industria 4.0, hace referencia a una política económica gubernamental basada en estrategias de alta tecnología; caracterizada por la automatización, la digitalización de los procesos y el uso de las tecnologías de la electrónica y de la información en la manufactura. Igualmente, por la personalización de la producción, la prestación de servicios, la creación de negocios de valor agregado, y por las capacidades de interacción y el intercambio de información entre humanos y máquinas (Villanueva, 2016) (Val Román & Val Román, 2017) (Ynzunza, Izar, Bocarando, Aguilar, & Larios, 2017) (Campero Núñez, 2016) (González Páramo, 2017) (Oliván, 2016)

La Industria 4.0 está sustentada en el desarrollo de sistemas, el Internet de las Cosas (IoT), el internet de la gente y de los servicios; además de otras tecnologías como la fabricación aditiva, la impresión 3D, la ingeniería inversa, el Big Data (datos masivos), inteligencia artificial, etc., las que al trabajar de forma conjunta, están generado cambios trascendentales no sólo en la industria de la manufactura sino también en el comportamiento del consumidor y en la manera de hacer negocios. Así, al mismo tiempo, favorecen la construcción de capacidades que permiten a las empresas adaptarse a los cambios del mercado (Val Román & Val Román, 2017) (Sanmartín , Ávila , Virola, & Jabba, 2016) (Sabalza, 2016).

Dentro de las tecnologías que sustentan la Industria 4.0 y la manufactura inteligente, se presentan la simulación, fabricación aditiva, los sistemas de integración horizontal y vertical, la ciber-seguridad, la realidad aumentada, el cómputo en la nube, los robots autónomos, el Internet Industrial de las Cosas (IIoT) y el Big Data (datos masivos), y la analítica avanzada. También, las tecnologías sociales, los Sistemas Ciber-físicos (CPS) y de colaboración abierta; los dispositivos móviles (tabletas, teléfonos), las plataformas

y aplicaciones tecnológicas, así como, la inteligencia artificial y tecnologías de la información (Val Román & Val Román, 2017) (Escudero, 2018) (Esmerio, 2018) (Félix & Blesa, 2015).

A través del Internet de las Cosas (IoT), los sistemas pueden interactuar entre sí y con los humanos en tiempo real. El Internet de los Servicios (IoS) es el medio mediante el cual es posible ofertar y acceder a éstos. Mientras que, el Big Data (datos masivos), el cómputo en la nube y la inteligencia artificial son facilitadores de la Industria 4.0 y junto con la automatización industrial están cambiando la forma en la que los productos se fabrican. Contribuyendo al mejoramiento de la manufactura y a que las empresas cuenten con procesos totalmente automatizados e interconectados, que faciliten el flujo de información, la descentralización de la manufactura, la creación de nuevos procesos, la toma de decisiones y un enfoque al desarrollo de competencias que agreguen valor a las organizaciones de forma tal, que se genere no solo mayor innovación de productos y procesos; sino también fábricas inteligentes y otros modelos de negocio. Se mejore la cadena de suministro; estrategia clave para incrementar la posición competitiva y la rentabilidad de las organizaciones (Val Román & Val Román, 2017) (Félix & Blesa, 2015) (González Páramo, 2017) (Oliván, 2016) (Sabalza, 2016).

De ahí, que la integración de todas las actividades de la empresa junto con aquellos que interactúan en la cadena de suministro, proveedores, clientes y socios, dentro de amplias redes de trabajo, sea una actividad medular en las organizaciones, y, la tecnología sea el mejor medio para diseñar, crear e implementar tales ambientes que faciliten el intercambio de información, productos y servicios, el aprovechamiento de las oportunidades y la creación de ventajas competitivas (Ortiz, 2016).

La Industria 4.0, hace referencia a tecnologías y conceptos de la organización de la cadena de valor en fábricas inteligentes, que cuentan en su estructura con Sistemas Ciber-físicos (CPS) capaces de efectuar la monitorización de los procesos físicos, crear una copia virtual del mundo real y hacer decisiones descentralizadas. Por lo cual, se espera que los CPS brinden soluciones que permitan transformar la operación y el papel de muchos de los sistemas industriales existentes. En la visión de la Industria 4.0, las máquinas serán capaces de comunicarse entre sí para recibir o transmitir información y ejecutar acciones; los productos serán inteligentes, ya que con el Internet de las Cosas

(IoT), muchos sensores en red podrán estar embebidos en dispositivos y máquinas, pudiendo desplegar enormes cantidades de datos de diferentes tipos, y el cómputo en la nube proveerá las soluciones para el almacenamiento y procesamiento de estos grandes volúmenes de datos (Val Román & Val Román, 2017) (Campero Núñez, 2016) (González Páramo, 2017) (Oliván, 2016).

Por tanto, y de conformidad con lo expuesto como antecedentes investigativos, el tema propuesto, su análisis, estudio y propuesta de solución, responden a la necesidad de formular, implementar y poseer herramientas y recursos que permitan la innovación de la actual industria.

2.2 Fundamentación filosófica

La presente investigación será desarrollada acogiéndose al fundamento filosófico del paradigma positivista, conceptualizado por Carmen Rivoy como: “el paradigma cuantitativo, empírico-analítico, racionalista, sistemático gerencial y científico tecnológico”. Por tanto, este paradigma sustentará la investigación, que tiene como objetivo desarrollar la propuesta planteada mediante la aplicación de parámetros determinísticos de una variable. Además, mencionado método cuantitativo se caracteriza por ser racional, objetivo, basándose en lo observable, manipulable y verificable. Este fundamento filosófico es válido debido a que se podrá aplicar métodos experimentales, manipulando de forma intencionada variables en diversos niveles de experimentación (Rivoy, 2006)

2.3 Fundamentación legal

La presente investigación se fundamenta en el Plan Nacional de Desarrollo 2017 – 2021 – Toda una Vida, el mismo que establece: “es necesario aumentar las posibilidades reales de transformación estructural, generando nuevos procesos y fortaleciendo el sistema productivo basado en eficiencia, generación de valor agregado e innovación, para reducir la vulnerabilidad externa. El desafío es alcanzar la transformación de la matriz productiva e incentivar nuevas producciones, para superar la estructura primario-exportadora”. (Secretaría Nacional de Planificación y Desarrollo - Senplades 2017, 2017).

2.4 Categorías fundamentales

2.4.1 Gráficos de inclusión interrelacionados

En la Figura 2, se muestra el detalle de la inclusión interrelacionada entre variable dependiente e independiente.

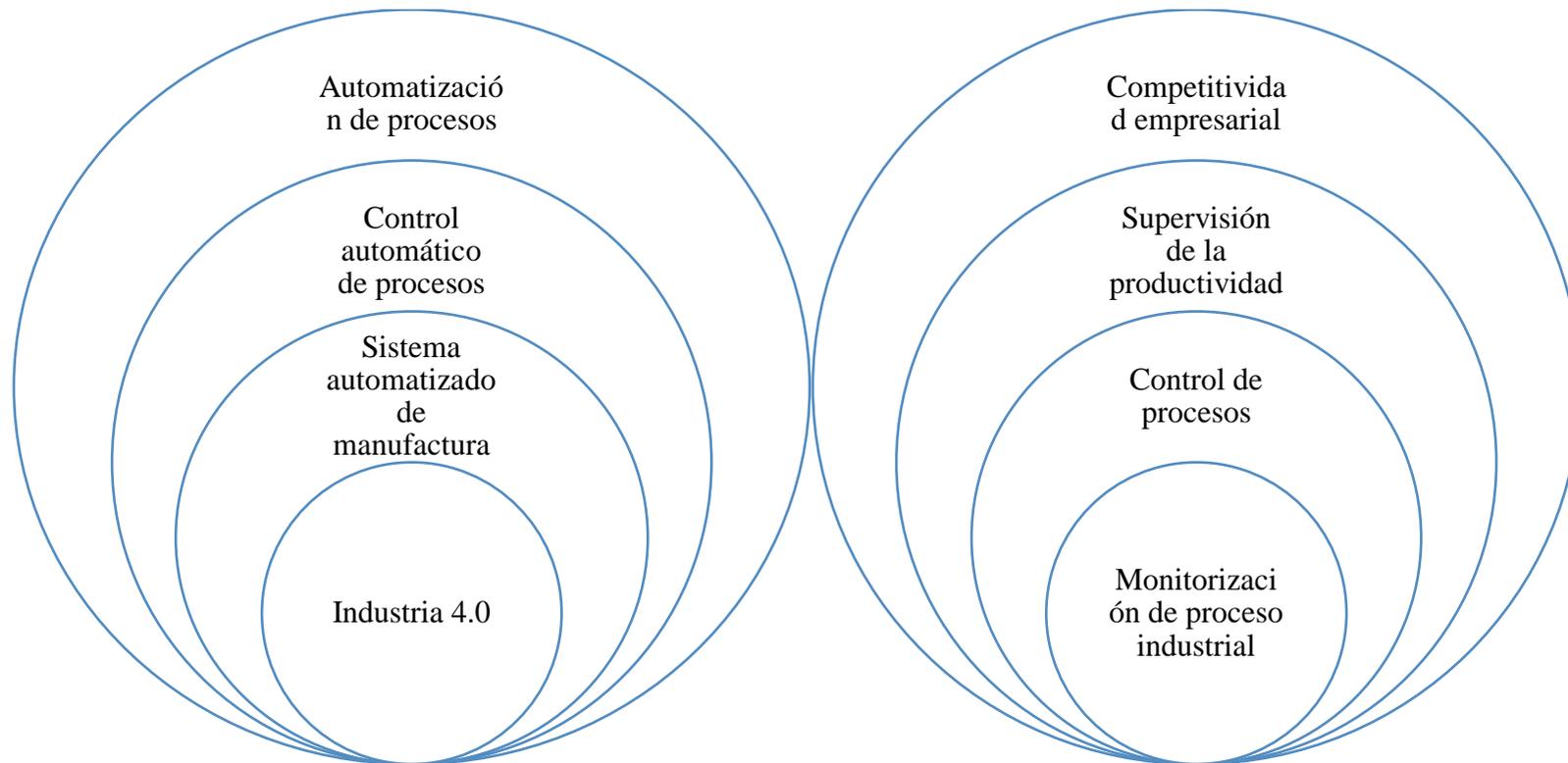


Figura 2: Categorías Fundamentales
Elaborado por: El Investigador

2.4.2 Constelación de ideas de la variable independiente

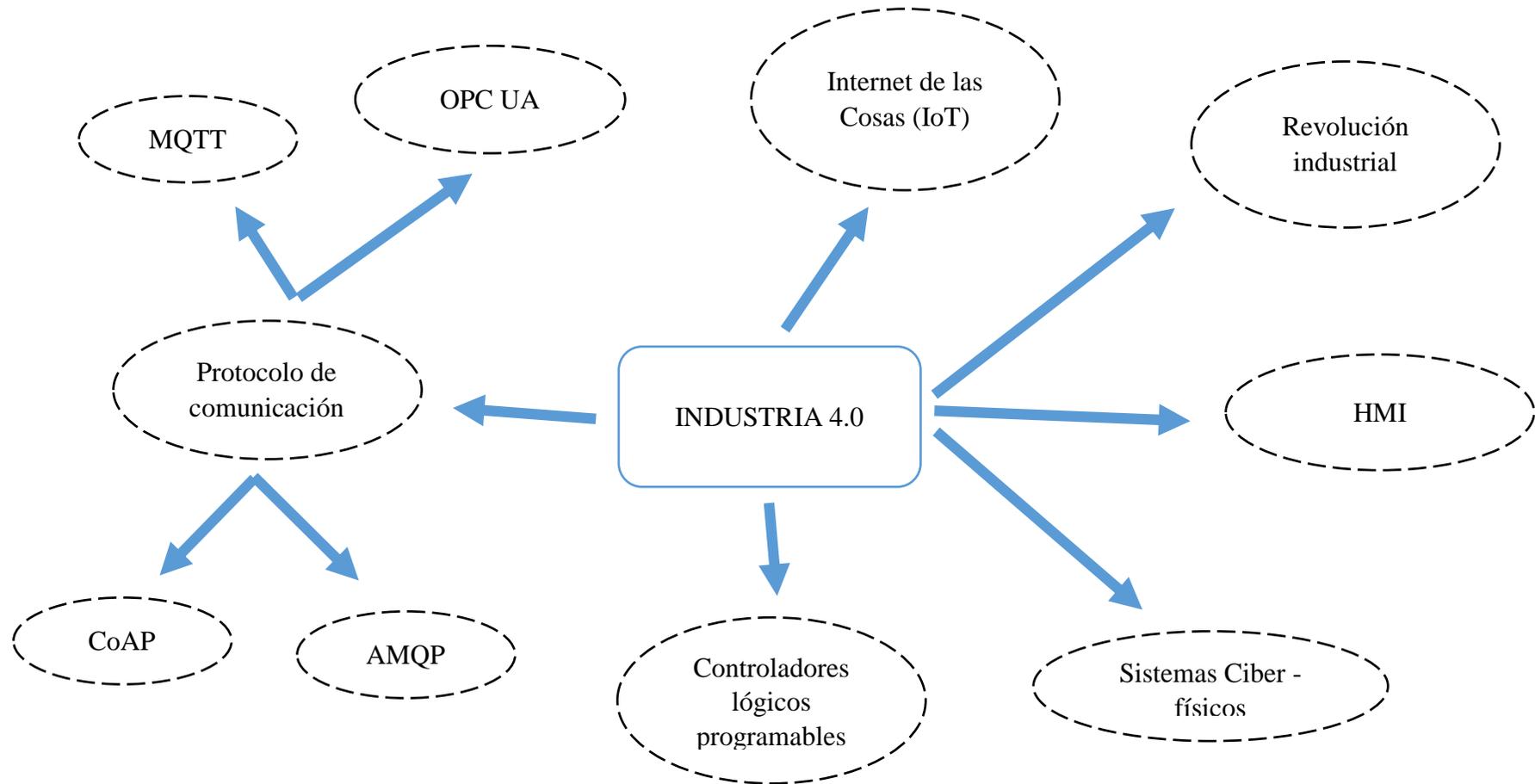


Figura 3: Constelación de ideas, variable independiente
Elaborado por: El Investigador

2.4.3 Constelación de ideas de la variable dependiente

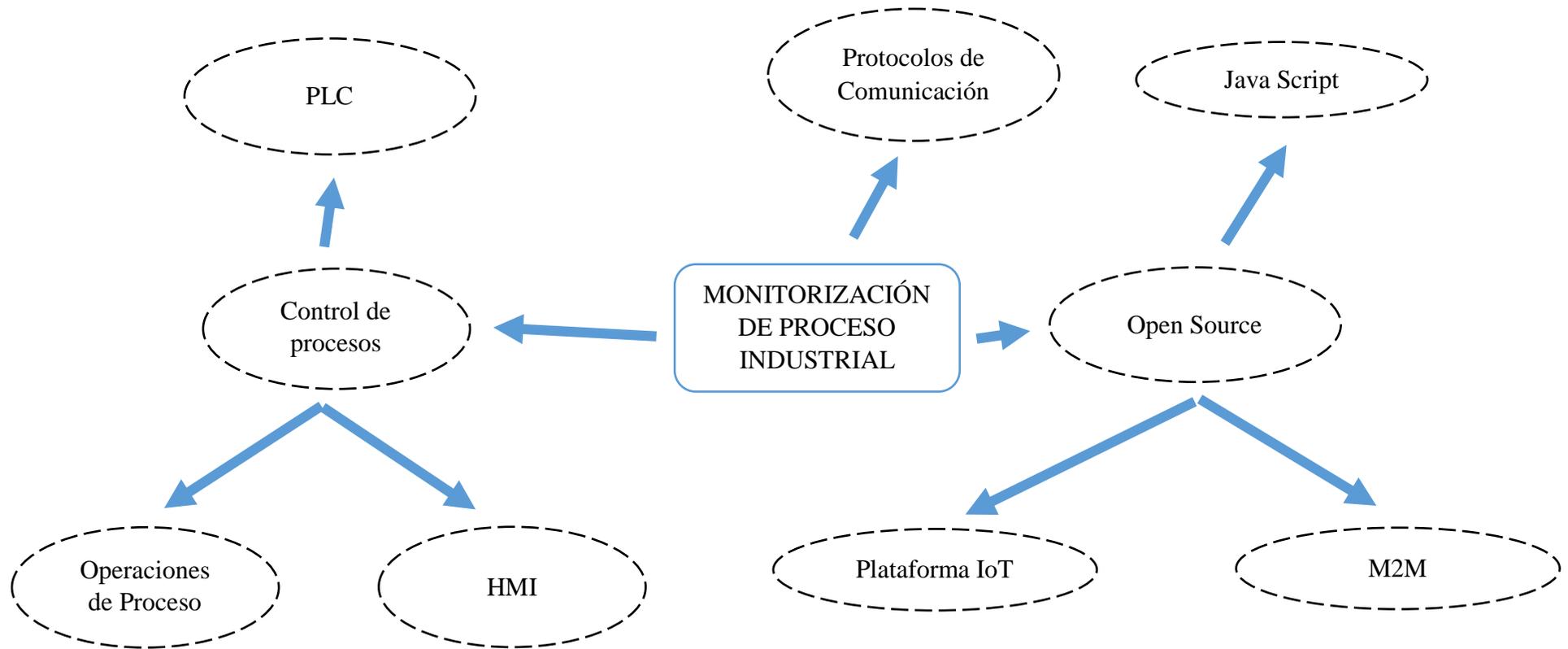


Figura 4: Constelación de ideas: variable dependiente
Elaborado por: El Investigador

2.5 Fundamentación teórica

2.5.1 Revolución industrial

El concepto Industria 4.0, Cuarta Revolución Industrial, Industria Inteligente o Ciber-industria del futuro; corresponde a una nueva manera de organizar los medios de producción.

El objetivo que pretende alcanzarse es la puesta en marcha de un gran número de fábricas inteligentes capaces de una mayor adaptabilidad a las necesidades y a los procesos de producción, así como a una asignación más eficiente de los recursos, abriendo así la vía a una nueva revolución industrial o Revolución industrial 4.0.

Las bases tecnológicas en que se apoya esta orientación, entre otras son: Internet de las Cosas (IoT); Sistemas Ciber-físicos (CPS), Cultura Maker y Fábrica 4.0. Sin embargo, la Industria 4.0 no se reduce exclusivamente a los cuatro puntos recién citados, pues es mucho más que eso. La Industria 4.0, llamada Cuarta Revolución Industrial, enfatiza y acentúa la idea de una creciente y adecuada digitalización y coordinación cooperativa en todas las unidades productivas de la economía (González Páramo, 2017) (Oliván, 2016) (Ortiz, 2016).

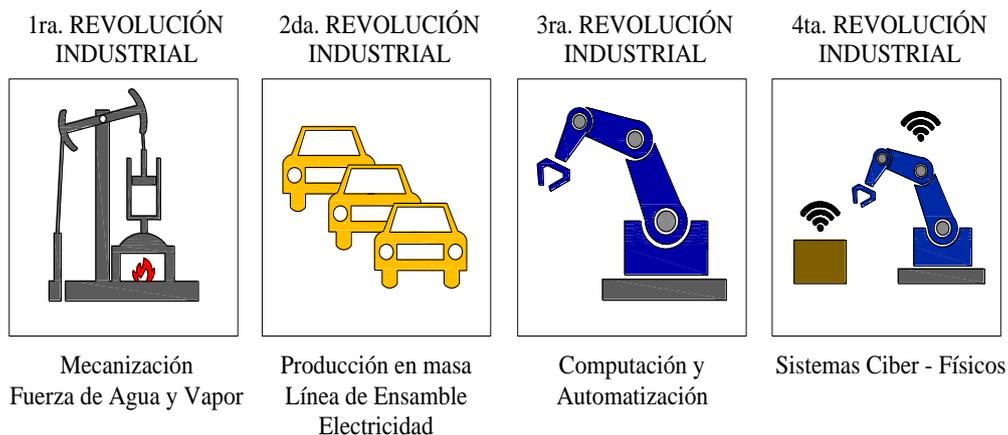


Figura 5: Diagrama revoluciones industriales sucesivas
Elaborado por: El Investigador

Esta revolución, también llamada digital, está marcada por avances tecnológicos emergentes en varios campos, que incluyen: robótica, inteligencia artificial, nanotecnología, computación cuántica, biotecnología, Internet de las cosas (IoT), impresión 3D y vehículos autónomos. La Tabla 1, detalla las características de los periodos históricos conocidos.

Fundamentalmente la Revolución 4.0, es diferente de las tres revoluciones anteriores, por los avances en tecnología y potencial para continuar conectando a miles de millones de personas a la web, mejorar drásticamente la eficiencia de las empresas, organizaciones y ayudar a regenerar el entorno natural a través de una mejor gestión de activos (Ortiz, 2016).

Tabla 1: Características de las revoluciones industriales
Fuente: (González Páramo, 2017) (Oliván, 2016) (Ortiz, 2016)

Descripción	Fuerza motriz	Maquinaria	Industria y Aplicación	Comunicación e Interconexión	Motores de combustión	Máquinas herramientas	Potencia eléctrica	Seguridad laboral	Automatización	Robótica	Inteligencia Artificial	IoT / IIoT
1ra. Revolución Industrial	Carbón / Vapor	Mecanización	Siderúrgica / Textil	Ninguna	●	●						
2da. Revolución Industrial	Electricidad / Petróleo	Motores de explosión, dinamo. Cinematógrafo y teléfono	Industria química, adelantos en agricultura y medicina	Básica	●	●	●					
3ra. Revolución Industrial	Energía alternativa (natural y atómica)	Automatización y robotización	Modernización de industrias e investigación aplicada, aparición de la aeronáutica.	Avanzada	●	●	●	●	●	●		
4ta. Revolución Industrial	Energía limpia y renovable	Robótica e inteligencia artificial	Industria digital y nanotecnología	Interconexión total por web	●	●	●	●	●	●	●	●

2.5.2 Industria 4.0

La Industria 4.0, también llamada cuarta revolución industrial se puede resumir en el concepto de crear fábricas más inteligentes. Una nueva manera de organizar los medios productivos para conseguir una mayor eficiencia.

Así también, podemos definir la Industria 4.0 como la digitalización de la industria y todos los servicios que conlleva, debido a la necesidad de globalización en un proceso de desarrollo que se acelera a pasos agigantados en los últimos tiempos. Es decir, el mundo virtual y real se fusionan en las fábricas, aplicando nuevas tecnologías de la información en todos los procesos productivos (Bonilla, Tavison, Morales, Guajardo, & Laines, 2016) (Aldakin, 2017).

Inmerso en esta nueva y cuarta revolución industrial, con la adecuada aplicación del enfoque Industria 4.0, se obtienen: procesos más depurados, repetitivos, sin errores ni alteraciones; optimización de niveles de calidad, mayor eficiencia y ahorro en costes; además de producción mucho más flexible y con tiempos reducidos. El flujo de datos es eficiente, con una competitividad empresarial más elevada y un nivel alto de respuesta a las necesidades de los mercados. Finalmente se asegura un gran potencial para conectar a millones de personas por medio de las redes digitales (Aldakin, 2017).

Sin embargo, no todas las organizaciones pueden adaptarse al ritmo de los nuevos métodos, con cambios cada día más veloces, muchas industrias pueden quedar desactualizadas en poco tiempo. La Industria 4.0 tiene una enorme dependencia tecnológica, del gobierno local y sus legislaciones. Además, el personal necesario en los nuevos procesos es mucho más especializado y los avances tecnológicos e industriales permiten desigualdades y cierta fragmentación social (Aldakin, 2017).

2.5.3 Internet de las Cosas (IoT)

Internet de las Cosas (en inglés, Internet of Things, IoT), es un concepto que se refiere a la interconexión digital de objetos cotidianos con Internet.

La revolución de Internet ha llevado a la interconexión entre las personas por medio de aplicaciones en dispositivos inteligentes, correo electrónico, redes sociales, etc. Ahora

estamos en la era de la interconexión con las cosas u objetos, para crear y promover un ambiente totalmente informado y confortable para una mejor toma de decisiones y calidad de vida.

La primera consecuencia directa del Internet de las Cosas (IoT), es la generación de grandes cantidades de datos, donde cada objeto físico o virtual conectado al IoT puede tener un doble digital en la nube, que podría generar actualizaciones periódicas. Como resultado, el volumen de mensajería relacionada con IoT podría alcanzar entre 1.000 y 10.000 mensajes por persona y día. No hace falta decir, que esto es una ingente cantidad de mensajes que es necesario gestionar (Bonilla, Tavison, Morales, Guajardo, & Laines, 2016) (Terán , Espinosa, Hernández, & Flores, 2017).

Hay tres componentes o elementos básicos que interactúan entre sí: a) el hardware, como sensores, actuadores (dispositivos que controlan los sistemas) y otros dispositivos de comunicación alojados en los objetos; b) la plataforma de middleware, que es el software que permite el intercambio de información entre las aplicaciones, así como las herramientas computacionales que permitan el análisis de datos; y c) las herramientas que en forma fácil permitan la visualización e interpretación de la información y que deben ser diseñadas para ser utilizadas por diferentes aplicaciones y dispositivos (Bonilla, Tavison, Morales, Guajardo, & Laines, 2016).

2.5.4 Sistemas Ciber-físicos (CPS) en la industria

Según la fundación americana NSF (National Science Foundation), describe los CPS (Cyber-Physical System) como: Sistemas construidos a partir de la integración transparente de componentes físicos y computacionales, que permitirán superar a los simples sistemas integrados actuales en cuanto a capacidad, adaptabilidad, escalabilidad, resiliencia, seguridad y usabilidad. Una evolución del Internet de las Cosas (IoT), una nueva generación de sistemas que permitirá un grado de conectividad múltiple que todavía hoy no se ha desarrollado en el mercado (Cámara Valencia, 2016) (García, Irísarri, Pérez, Estévez , & Marcos, 2018).

La llegada de los Sistemas Ciber-físicos (CPS) a diferentes disciplinas de la tecnología, empresa o de la sociedad se identificará especialmente con algunas áreas tecnológicas,

como: Software, Servicios, Nube y Big Data (datos masivos) (García, Irísarri, Pérez, Estévez , & Marcos, 2018).

2.5.5 Interfaz Hombre Máquina (HMI)

El Interfaz Hombre-Máquina (HMI) es el interfaz entre el proceso y los operarios; se trata básicamente de un panel de instrumentos del operario. Es la principal herramienta utilizada por operarios y supervisores de línea para coordinar y controlar procesos industriales y de fabricación. El HMI traduce variables de procesos complejos en información útil y procesable.

La función de los HMI consiste en mostrar información operativa en tiempo real y casi en tiempo real. Proporcionan gráficos de procesos visuales que aportan significado y contexto al estado del motor y de la válvula, niveles de depósitos y otros parámetros del proceso. Suministran información operativa al proceso, permiten el control y la optimización al regular los objetivos de producción y de proceso, como se detalla en el esquema presentado en la Figura 6 (Wonderware, 2016).

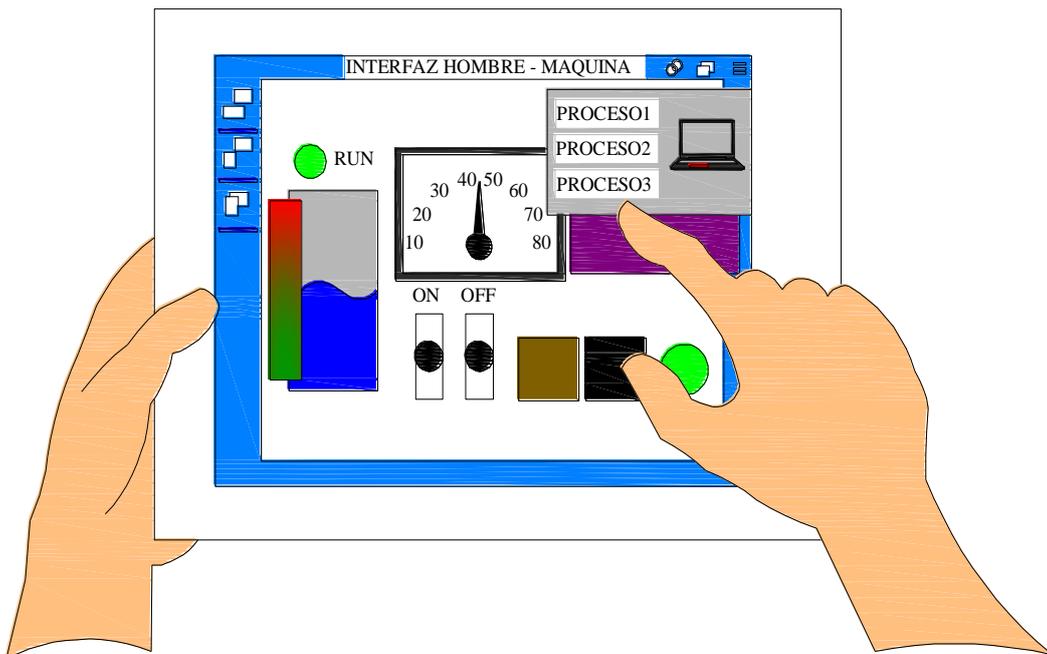


Figura 6: Esquema referencial Interfaz Hombre Máquina (HMI)
Elaborado por: El Investigador

2.5.6 Controlador Lógico Programable (PLC)

Un PLC o Controlador Lógico Programable (Programmable Logic Controller), en sí es un sistema de control. Son dispositivos electrónicos o computadoras digitales de tipo industrial que permiten la automatización, especialmente de procesos de la industria, debido a que controlan tiempos de ejecución y regulan secuencias de acciones.

De acuerdo con la definición de la NEMA (National Electrical Manufacturers Association) un PLC es: “Un aparato electrónico operado digitalmente, que usa una memoria programable para el almacenamiento interno de instrucciones para implementar funciones específicas, tales como lógica, secuenciación, registro y control de tiempos, conteo y operaciones aritméticas para controlar, a través de módulos de entrada/salida digitales (ON/OFF) o analógicos (1-5 VDC, 4-20 mA, etc.), varios tipos de máquinas o procesos” (Páez, Zamora, & Bohórquez, 2015).

2.5.7 Open source

Open source, es una expresión de la lengua inglesa que pertenece al ámbito de la informática. Aunque puede traducirse como “fuente abierta”, suele emplearse en nuestro idioma directamente en su versión original, sin su traducción correspondiente. Se califica como open source, por lo tanto, a los programas informáticos que permiten el acceso a su código de programación, lo que facilita modificaciones por parte de otros programadores ajenos a los creadores originales del software en cuestión.

Es importante distinguir entre el software open source, que dispone de la mencionada característica de presentar su código abierto, y el software libre (que puede descargarse y distribuirse de manera gratuita). Existe software libre que no brinda acceso al código (y que, por lo tanto, no puede considerarse como open source), y programas open source que se distribuyen de manera comercial o que requieren de una autorización para ser modificados (Valverde, 2005).

2.5.8 Comunicación Máquina a Máquina (M2M)

La Comunicación Máquina a Máquina (M2M), es la tecnología que permite establecer comunicaciones inteligentes entre los objetos, las cosas y los sistemas que los siguen y controlan, básicamente por vía inalámbrica. En el universo de sensores, redes y análisis

de datos aparecen la Comunicación Máquina a Máquina (M2M), como base del sistema de comunicación del Internet de las Cosas (IoT) y como red nerviosa del nuevo mundo digital que combina personas y objetos (Weyrich, Schmidt, & Ebert, 2014).

Un primer e importante campo de aplicación de las comunicaciones M2M es en el mundo de la industria. La conexión inteligente y la comunicación entre todas las máquinas y dispositivos de una fábrica hacen posible controlar, automatizar y mejorar los procesos; expandir los negocios con seguridad y costes reducidos. La fabricación y producción de bienes se ve potenciada por las posibilidades que traen estas nuevas tecnologías (Weyrich, Schmidt, & Ebert, 2014).

2.5.9 JavaScript

JavaScript (abreviado comúnmente JS) es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (cliente - side), implementado como parte de un navegador web, permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. Existe también una forma de JavaScript del lado del servidor (Server – side, JavaScript o SSJS). Su uso es también significativo en aplicaciones externas a la web, por ejemplo, en documentos PDF, aplicaciones de escritorio (widgets), entre otros.

Tradicionalmente se utiliza en páginas web HTML para efectuar operaciones en el marco de la aplicación cliente, sin mayor acceso a funciones del servidor. Actualmente es ampliamente utilizado para enviar y recibir información del servidor junto con ayuda de otras tecnologías como AJAX. JavaScript se interpreta en el agente de usuario al mismo tiempo que las sentencias van descargándose junto con el código HTML (Wagner, 2016).

2.5.10 Control de procesos

El control de procesos consiste en dos funciones claramente diferenciadas: la adquisición de datos y el control. Si tratamos de establecer el nexo con el mantenimiento concluiremos rápidamente que la adquisición de datos contribuye con la

información para el mantenimiento y las acciones de control con la implantación de las acciones confines tanto operativos como de mantenimiento. En la Figura 7, se muestra los niveles de un sistema de control de procesos.

El control automático de procesos es parte del progreso industrial desarrollado durante lo que ahora se conoce como la segunda revolución industrial. El uso intensivo de la ciencia de control automático es producto de una evolución que es consecuencia del uso difundido de las técnicas de medición y control. Su estudio intensivo ha contribuido al reconocimiento universal de sus ventajas.

El control automático de procesos se usa fundamentalmente porque reduce el costo de los procesos industriales, lo que compensa con creces la inversión en equipo de control. Además, hay muchas ganancias intangibles, como por ejemplo la eliminación de mano de obra pasiva, la cual provoca una demanda equivalente de trabajo especializado. La eliminación de errores es otra contribución positiva del uso del control automático (Arbildo, 2011) (Alcuri, 2013).



Figura 7: Niveles de un sistema de control de procesos
Elaborado por: El Investigador

2.5.11 Plataforma Internet de las Cosas (IoT)

Una plataforma IoT es la base para que dispositivos estén interconectados y se genere un ecosistema tecnológico propio. Dicho de otra forma, una plataforma web integrada al Internet de las Cosas (IoT), es el software que conecta hardware, puntos de acceso y redes de datos a lo que generalmente suele ser la aplicación de la que disfruta el usuario (Cárdenas, 2016).

Como características de plataformas del Internet de las Cosas (IoT), podemos describir las que se centran en la conectividad de los dispositivos a través de redes de telecomunicaciones, servicios que proporciona alojamiento y potencia de procesamiento para aplicaciones y servicios, plataformas específicas según su hardware, además de extensiones/ampliaciones de software como integración de dispositivos para el consumo masivo en requerimientos empresariales. Se puede detallar según sus características: plataformas de conectividad M2M, Beckend IaaS, entre otras (Cárdenas, 2016).

Toda plataforma de Internet de las Cosas (IoT) debe poseer propiedades y características adecuadas de software, desarrollo de aplicaciones y análisis de variables. Tipologías que permitan garantizar la precisa transmisión de datos y la interacción adecuada con todos los dispositivos conectados. Además de almacenamiento escalable, reglas de ejecución y toma de decisiones inteligentes, análisis complejos de datos tanto del proceso como del sistema para el continuo aprendizaje, visualización normalizada de información, y finalmente herramientas e interfaces estrenas adicionales. Todo lo detallado anteriormente se puede resumir en parámetros de: conectividad y normalización, gestión de dispositivos, base de datos, procesamiento y gestión de la acción, características analíticas, de visualización, y herramientas e interfaces adicionales. (Cárdenas, 2016).

2.5.12 Protocolo OPC UA

El Protocolo de Comunicación de Arquitectura Unificada (OPC UA), representa la evolución de normas anteriores como OPC-DA, OPC-A&E y OPC HDA con el fin de proporcionar una arquitectura abierta e independiente de proveedor, con el fin de solucionar los problemas de portabilidad de las normas anteriores. El servidor OPC UA emplea una pila de comunicación que puede ser utilizada directamente en los sistemas

de automatización y aplicación industrial, debido a su comunicación fiable, robusta y de alto rendimiento. Además, puede ser integrado en dispositivos empotrados como los que implementan CPPSs.

OPC UA incluye un mecanismo de modelado capaz de soportar los posibles modelos para una planta industrial, ofrece acceso a la información a través de modelos de planta y permite una descripción completa de los datos del sistema, independientemente de su complejidad. Este enfoque introduce el concepto de modelo de proceso para una mejor adaptación a las necesidades de las aplicaciones industriales modernas.

Con los servidores OPC UA es posible una representación basada en el modelo del proceso de planta que consiste en un conjunto de objetos que el servidor pone a disposición de los clientes. Estos objetos representan los datos de proceso subyacentes en tiempo real (García, Irisarri, & Pérez, 2017)

2.5.13 Protocolo AMQP

AMQP (Advanced Message Queuing Protocol), protocolo de estándar abierto en la capa de aplicaciones de un sistema de comunicación. Lo definen la orientación a mensajes, encolamiento, enrutamiento (tanto punto a punto como publicación-subscripción), exactitud y seguridad. Su estructura general se muestra en la Figura 8.

AMQP es un protocolo a nivel de cable, estipula el comportamiento tanto del servidor que provee los mensajes como del cliente de la mensajería. Un protocolo a nivel de cable (wire level protocol) es una descripción del formato de los datos que son enviados a través de la red como un flujo de octetos. Cualquier programa, aplicación o cliente que pueda crear e interpretar mensajes conforme a este formato de datos puede interactuar con otra herramienta que cumpla con este protocolo, independientemente del lenguaje de implementación.

La serie de entidades necesarias según la perspectiva de la interconexión son: corredor de mensajes, un servidor en entorno distribuido al que los clientes se conectan usando el protocolo AMQP; usuario, entidad capaz de conectarse a un corredor; conexión: canal de comunicación ligada a un usuario, ejemplo TCP/IP o SCTP; canal, enlace lógico que está unida a una conexión (Fernandes, Lopez, Rodrigues, & Ullah, 2013).

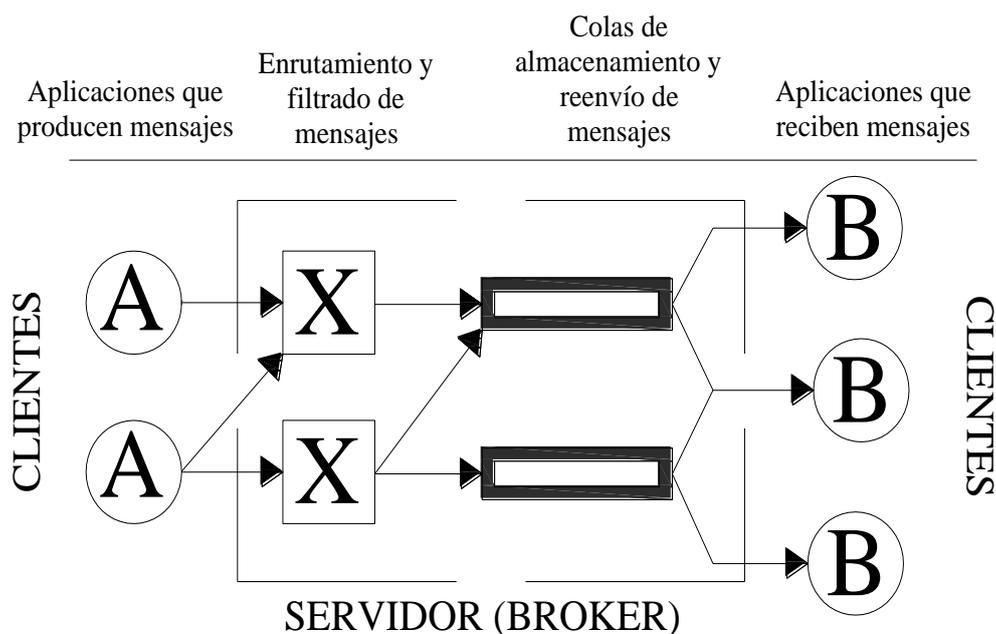


Figura 8: Estructura general Protocolo AMQP
Elaborado por: El Investigador

2.5.14 Protocolo CoAP

El Protocolo de Aplicación Restringida (Constrained Application Protocol, CoAP), es un software a nivel de aplicación usado principalmente en dispositivos electrónicos simples permitiendo que puedan comunicarse sobre Internet.

CoAP es un protocolo del IETF (Internet Engineering Task Force) un enfoque tradicional de cliente-servidor en comparación al de brokers, opera en la capa de servicio, diseñado para su uso en dispositivos restringidos (nodos de red), como sensores inalámbricos. Posee características de soporte en multidifusión, gastos generales muy bajos y simplicidad además de integración simplificada con la web. Así también: transferencia de documentos cliente/servidor, fácil de traducir a HTTP para la integración web, topología uno a uno con conexiones directas, metadatos para diferenciar clases de documentos, UDP y seguridad a través de DTLS (Khattak, Ruta, & Di Sciascio, 2014), (Pradilla, González, Esteve, & Palau, 2016).

2.5.15 Protocolo MQTT

MQTT (Message Queue Server Telemetry Transport) es una norma ISO (ISO / IEC PRF 20922), protocolo de mensajería basado en publicación-suscripción. Funciona

sobre el protocolo TCP / IP y está diseñado para conexiones con ubicaciones remotas y/o cuando el ancho de banda de la red es limitado. Su sistema consiste en clientes que se comunican con el servidor central, como publicadores de información o suscriptores; existe desacoplamiento total entre editores y suscriptores. La información se organiza en una jerarquía de Topics.

Las principales características del protocolo son: ancho de banda muy bajo, operación sobre TCP/IP, la transferencia de mensajes es por publicaciones/suscripciones, Topología en forma de estrella con un broker central, no se manejan metadatos, existe niveles de calidad del servicio y seguridad en la comunicación (Munner, Mohammed, & Shadi, 2017).

2.6 Hipótesis

La Industria 4.0 incide en la monitorización adecuada de procesos industriales

2.7 Señalamiento de variables de la hipótesis

2.7.1 Variable independiente

Industria 4.0

2.7.2 Variable dependiente

Monitorización de proceso industrial

CAPÍTULO III

METODOLOGÍA

3.1 Enfoque

La investigación asumirá un enfoque cuantitativo debido a que su desarrollo es secuencial y probatorio, cumple con el planteamiento de un problema, hipótesis, desarrollo de la propuesta, implementación, experimentación, análisis de datos y desarrollo de conclusiones. Al mismo tiempo está determinado que la mayor parte de información y datos que se obtengan del proceso sean cuantificables. Sin embargo, para el análisis de información obtenida mediante técnicas de observación, revisión de documentos y afines; se aplicará un enfoque cualitativo, orientado principalmente en una perspectiva interpretativa.

3.2 Modalidad básica de la información

Bibliografía documental

La investigación empleará este tipo de modalidad debido a que podrá ser ampliada y profundizada mediante la búsqueda informativa en revistas, libros, publicaciones, internet, textos y aportes tratados en cursos, ponencias y congresos. Además de fuentes primarias de información obtenidas de experimentación y trabajos relacionados, fundamentos propicios para establecer procedimientos en el desarrollo del proyecto y propuestas de solución.

De campo

La investigación estudia los hechos, características y particularidades de las diferentes pruebas, experimentos y validación de la plataforma, recabando información representativa, real y necesaria para conocer y describir la problemática, indagación que fomentará una mejor propuesta de solución.

3.3 Nivel o tipo de investigación

Exploratorio

Se estipula para la investigación un nivel exploratorio, debido a que se realizará un acercamiento inicial, reconociendo procesos, métodos y técnicas de monitorización actuales, permitiendo familiarizarse con la problemática y obtener información inicial para continuar con una investigación más rigurosa.

Descriptivo

Se aplica este tipo de investigación con el objetivo de plantear las características más relevantes de los procesos industriales a evaluarse, definiendo su análisis y los métodos que involucran al mismo, mediante una técnica adecuada de recolección de datos y fuentes a consultar.

Asociación de Variables

Se fija el nivel de asociación de variables precisando la importancia de establecer la relación entre la variable independiente y dependiente planteadas, mediante el análisis, experimentación y variación de datos que puedan aseverar la hipótesis propuesta.

3.4 Población y muestra

La investigación analizará el estado del arte y antecedentes investigativos, además de probar el funcionamiento de la plataforma mediante la aplicación de proceso industrial básico como caso de estudio, tomando en cuenta opiniones, revisiones y recomendaciones. En todo caso la población será menor a 100, por tanto, la población será la muestra.

3.5 Operacionalización de variables

3.5.1 Variable independiente

Tabla 2: Operacionalización de la variable independiente
Elaborado por: El Investigador

Conceptualización	Dimensiones	Indicadores	Ítems básicos	Técnicas e instrumentos
Industria 4.0, actual revolución industrial, consiste en la digitalización de los procesos industriales por medio de la interacción de la inteligencia artificial con las máquinas y la optimización de recursos enfocada en la creación de efectivas metodologías comerciales.	Cliente	Gráfica de datos	¿Qué tipo de gráfico estadístico es necesario?	Estadísticas de productividad y proceso
		HMI	¿Se dispone de un HMI adecuado de proceso?	Página Web
		Lenguaje de programación	¿Qué tipo de lenguaje de programación se utiliza?	C, Java Script
	Comunicación	Protocolo de comunicación	¿Qué protocolo de comunicación es el adecuado?	MQTT, Internet de las Cosas (IoT)
		Servidor y bróker	¿Se levantó información del servidor y bróker?	Mosquitto
		Session Border Controller (SBC)	¿Se ha seleccionado un SBC?	Raspberry Pi
	Proceso	Controladores lógicos programables	¿Se dispone un contralor robusto y adecuado?	PLC
		Sensores y actuadores	¿Qué tipo de sensores y actuadores se utiliza?	Sensor de presencia y nivel (digital/analógico)

3.5.2 Variable dependiente

Tabla 3: Operacionalización de la variable dependiente
Elaborado por: El Investigador

Conceptualización	Dimensiones	Indicadores	Ítems básicos	Técnicas e instrumentos
Equipos y sistemas encaminados a permitir el control y monitorización de plantas, tareas y/o procesos industriales de una empresa o fábrica.	HMI	Interfaz hombre, máquina	¿Se posee visualización Scada del proceso?	HMI, HTML, ordenador
		Seguridad del control	¿Conoce las características de seguridad de la comunicación?	MQTT safety, protocolos de cifrado
	Controladores lógicos programables	Layout	¿Se posee detalle del layout del proceso?	C++, Java Script
		Lenguaje de programación	¿Qué tipo de lenguaje se va a utilizar?	Open source
		Protocolo de comunicación	¿Existe un registro del tipo de comunicación adecuado?	Ethernet, TCP/IP, Python, Snap 7
		Planta y/o proceso	Proceso digital/analógico	¿Características del proceso industrial?
Sensores y actuadores	¿Se conoce las condiciones actuales del Proceso?		Sensores de nivel, de presencia Simulación del proceso Evaluación de tiempos y variables	

3.6 Recolección de información

El plan que se manejará para recoger información necesaria en el desarrollo del presente proyecto, se describe en la Tabla 4.

Tabla 4: Plan de recolección de información
Elaborado por: El Investigador

Preguntas básicas	Explicación
1. ¿Para qué?	Para alcanzar los objetivos de la investigación: <ul style="list-style-type: none">• Investigar el Estado del Arte sobre estructuras, protocolos y software aplicable a Industria 4.0• Diseñar e implementar una plataforma de Industria 4.0 para procesos industriales• Validar la plataforma de Industria 4.0 mediante pruebas y caso de estudio
2. ¿De qué personas u objetos?	Conceptos, estado del arte y pruebas de funcionamiento
3. ¿Sobre qué aspectos?	<ul style="list-style-type: none">• Protocolos de comunicación• Sensores y actuadores• Internet de las Cosas (IoT)
4. ¿Quién, quienes?	Investigador
5. ¿Cuándo?	Periodo de seis meses durante el 2019 y después de la aprobación del proyecto
6. ¿Dónde?	Laboratorio de PLC's de la FISEI
7. ¿Cuántas veces?	Las necesarias para recoger y validar los datos
8. ¿Qué técnicas de recolección?	<ul style="list-style-type: none">• Análisis de datos• Pruebas y validación• Demostración estadística

9. ¿Con que?	Lista de verificación, herramientas de análisis de tráfico de red, observación directa y medición de datos
10. ¿En qué situación?	Durante la validación de la plataforma y las pruebas de funcionamiento

3.7 Procesamiento y análisis

El procesamiento y análisis de datos se alcanzará conforme los procesos detallados a continuación.

- Revisión de la información
- Transcripción y ordenamiento de la información
- Integración de datos
- Estudio estadístico, elaboración de gráficas y demás operaciones apropiadas.
- Resaltar cualquier resultado significativo
- Interpretación
- Presentación de resultados

3.8 Desarrollo del proyecto

Los pasos a seguir para la ejecución ordenada del proyecto presentado, son los siguientes:

- Estudio de protocolos de comunicación
- Selección de plataforma, protocolos y software para el proyecto
- Diseño de la plataforma y monitorización de variables
- Diseño de arquitectura y modelo de protocolo
- Programar controladores y síntesis de comunicación
- Realizar pruebas y validar plataforma
- Redacción de informe final, correcciones y mejoras

CAPÍTULO IV

ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

4.1 Desarrollo del proyecto

Pregunta directriz:

¿Qué estructura, protocolo o software es aplicable al enfoque de Industria 4,0?

La Industria 4.0 se puede definir como la digitalización de la industria y todos los servicios que conlleva, caracterizada como la automatización y el uso de tecnología de la electrónica y de la información en la manufactura (Ynzunza, Izar, Bocarando, Aguilar, & Larios, 2017)

Los protocolos de datos que pueden ser utilizados para el Internet de las Cosas (IoT), son los siguientes: MQTT, CoAP, AMQP y OPC UA.

La aplicación de cada uno de los protocolos dependerá específicamente de las características del proceso, comunicación recursos y demandas del sistema a implementar (Postscapes, 2017).

Pregunta directriz:

¿Qué características posee y como opera el actual método de monitorización de los procesos industriales?

La comunicación y monitorización actual de procesos, sean estos industriales, de servicios o atención al cliente, suponen uno de los principales problemas en los procesos de fabricación; la existencia de silos de información a diversos niveles, que impiden que esta fluya a lo largo de la cadena de valor. Se trata de un problema de

comunicación vertical, en el que la capa de proceso no se comunica adecuadamente con la capa de empresa. Y horizontal, entre actores de una cadena de valor como puede ser el fabricante de un producto y sus proveedores (Aldakin, 2017) (Atiga, 2017).

La aplicación del paradigma de Internet de las Cosas (IoT) y sus tecnologías dentro de un entorno industrial permite eliminar las barreras de conectividad tanto a nivel vertical como horizontal. Esto posibilita por un lado la obtención de información de valor de múltiples fuentes, lo que permite analizarla en detalle para extraer nuevo conocimiento; por otro lado, se mejora la comunicación entre diversos actores, mejorando la eficiencia de los procesos. Pequeñas, medianas y grandes empresas ofrecerán servicios de mayor calidad, velocidad y adaptabilidad a las necesidades de la sociedad, cada vez más exigentes en términos como la inmediatez, interconexión, comodidad, procesos múltiples o seguridad (Atiga, 2017).

Pregunta directriz:

¿Existen plataformas validadas de diseños en Industria 4,0 para la monitorización de procesos industriales?

Tal y como se describe en el literal 2.6 y 2.6.1 sobre las características de una plataforma de Internet de las Cosas (IoT), la selección y desarrollo de una plataforma validada dependerá del proceso a controlarse y de las características del interfaz de usuario final; enmarcados en las propiedades sistemáticas de una plataforma IoT, como: conectividad y normalización, la gestión de dispositivos, bases de datos, procesamiento y gestión de la acción, propiedades analíticas, visualización, herramientas adicionales e interfaces externas. Se puede añadir que el mercado de las plataformas IoT está en auge y en continua expansión, existe encuestas que hablan de que más 80% de las empresas cree que el campo de Internet de las Cosas (IoT) es el más interesante para sus negocios (Cárdenas, 2016).

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

- Actualmente el avance y desarrollo de tecnologías para la industria ha conllevado a ser partícipes directos de la cuarta revolución industrial o Industria 4.0, obligándonos a adaptarnos al ritmo de nuevos métodos, tecnologías y en general a cambios cada vez más veloces, corriendo el riesgo como industria de continuar con los actuales métodos de control y monitorización; de quedarnos desactualizados en muy poco tiempo.
- El Internet de las Cosas (IoT) presenta una solución robusta para la actualización e inclusión de tecnología en el ámbito industrial, sin importar su aplicación directa; fabricación digital, gestión de instalaciones e inventarios, seguimiento del flujo de producción, seguridad, calidad, logística entre otras; son los usos del IoT para los procesos y demás características industriales.
- Los protocolos de datos que pueden ser utilizados para el IoT, son MQTT, CoAP, AMQP y OPC UA; los que poseen características de comunicación, seguridad y desarrollo conforme las necesidades de usuario y proceso. MQTT (Message Queue Telemetry Transport) por su parte, es un protocolo para la Comunicación Máquina a Máquina (M2M), adecuado e idóneo para la aplicación sobre procesos industriales, particularidad de la presente investigación.
- No existe plataformas validadas específicas para la aplicación de un proceso industrial específico, la misma debe desarrollarse conforme las necesidades y tipologías propias del proceso a controlar, tomando en cuenta las condiciones necesarias para el desarrollo de aplicaciones, software y servicios para la comunicación de personas y “cosas”.

5.2 Recomendaciones

- Desarrollar una plataforma de Internet de las Cosas (IoT) que permita el control y monitorización de un proceso industrial aplicando el concepto de Industria 4.0.
- Determinar las características necesarias de comunicación, topología, conectividad, visualización, seguridad e interfaz necesarias para el desarrollo de la plataforma IoT mediante el protocolo de Comunicación Máquina a Máquina (M2M) en MQTT (Message Queue Telemetry Transport).
- Validar la plataforma IoT desarrollada para Industria 4.0 mediante un caso de estudio y pruebas de funcionamiento, además de realizar un respectivo análisis de tráfico de datos y herramientas necesarias.
- Se recomienda utilizar un entorno de desarrollo Web compatible con el protocolo de comunicación para la interfaz y visualización final de la aplicación y plataforma IoT.

CAPÍTULO VI

PROPUESTA DE INVESTIGACIÓN

6.1 Tema de la propuesta

DESARROLLO DE UNA PLATAFORMA IoT BASADO EN EL CONCEPTO DE INDUSTRIA 4.0, PARA LA MONITORIZACIÓN DE UN PROCESO INDUSTRIAL

6.2 Datos informativos

6.2.1 Ejecutada por:

Ingeniero Industrial Mauricio Xavier López Flores, estudiante egresado de la Maestría de Automatización y Sistemas de Control de la Universidad Técnica de Ambato.

6.2.2 Beneficiarios:

Empresas públicas y privadas que deseen desarrollar e implementar una plataforma de Industria 4.0 para el control y gestión de sus procesos, productos y/o servicios.

6.2.3 Ubicación:

Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato.

6.2.4 Equipo responsable:

- Ing. Ind. Mauricio X. López F. (Investigador)
- Ing. Marcelo García PhD. (Tutor)

6.2.5 Financiamiento:

Con recursos propios del investigador.

6.3 Antecedentes de la propuesta

La tecnología de los próximos años vendrá marcada en gran parte por la eclosión de los Sistemas Ciber-físicos (CPS), los mismos brindarán una mayor interconexión, rapidez, adaptabilidad o seguridad a todo tipo de productos, servicios y procesos. Todo ello será vital para el crecimiento de una empresa, la generación de negocio, empleo y el bienestar de un cada vez más exigente usuario/cliente (Tecnología para los negocios, 2018).

La tecnología avanza a pasos agigantados, y las principales novedades de hoy dan paso a evoluciones en nuevas tecnologías del mañana. El concepto de Internet de las Cosas (IoT) ha tomado forma, se estima que el número de dispositivos conectados crecerá de 10,3 millones en 2014 a más de 29,5 millones en 2020. En la actualidad, el mercado ya cuenta con todo tipo de dispositivos interconectados que satisfacen nuevas necesidades de la sociedad y de la empresa (Universidad de Barcelona, 2016).

Es por ello que, el desarrollo de sistemas y plataformas de comunicación industrial que permitan la adaptación adecuada a procesos y sus respectivas características es de trascendental importancia; objetivo en el que se enmarca la presente propuesta de investigación

Efectuada la indagación en artículos, trabajos y proyectos afines a la propuesta planteada, a continuación, se presentan los más relevantes:

En la investigación titulada “Arquitectura de Automatización basada en Sistemas Ciber-físicos (CPS) para la Fabricación Flexible en la Industria de Petróleo y Gas” (García, Irísarri, Pérez, Estévez , & Marcos, 2018); se presenta el uso de arquitecturas y entornos de programación bajo la norma IEC 61499, para introducir elementos novedosos sobre CPS dentro del paradigma de la Industria 4.0. La arquitectura propuesta proporciona una infraestructura de Comunicación Máquina a Máquina (M2M), para las comunicaciones a nivel de planta y la integración en el proceso productivo de dispositivos de niveles superiores.

El resultado obtenido en la investigación “Implementation of MQTT protocol in Context with Industry 4.0” (Sonawala, Tank, & Patel, 2017); muestra el sistema de

monitorización de información de sensores industriales, de bajo costo, bajo consumo de energía y en tiempo real, basado en el concepto de Industria 4.0 e Internet de las Cosas (IoT) mediante el protocolo de comunicación MQTT. Logrando una monitorización de datos en tiempo real, un control de producción inteligente, así como propósitos de seguridad informática e industrial.

La cuarta revolución industrial (Industria 4.0), encapsula las tendencias futuras de desarrollo de la industria para lograr más procesos de fabricación inteligentes, incluidos los Sistemas Cibernéticos (CPS) y Sistemas de Producción Ciber Físicos (CPPS). El artículo “Industry 4.0: Towards Future Industrial Opportunities and Challenges” introduce aspectos relevantes del tema en cuestión, relacionados específicamente a la planificación estratégica, tecnologías clave, oportunidades y retos. Logrando las tres integraciones anheladas en la industria (integración horizontal, integración vertical e integración de extremo a extremo) (Zhou, Liu, & Zhou, 2015).

Conocer y estudiar la sobrecarga de un mecanismo de comunicación permite al sistema robustez para su óptima utilización y evita la degradación de su rendimiento. En el artículo titulado “Comparison of IoT Data Protocol Overhead “, se construye un modelo teórico para derivar y comparar los gastos generales de los protocolos WebSocket, CoAP y MQTT al enviar un número arbitrario de paquetes de datos en sentido ascendente. Tomando en cuenta que para la lógica empresarial su aplicación está estrechamente unida y dependiente al tipo de conexión específica, los resultados que se obtuvieron son: los tres protocolos aumentan su rendimiento cuando se envían más datos de aplicaciones en un solo envío; el protocolo CoAP proporciona un mecanismo de solicitud / respuesta que lo hace adecuado para casos de uso general que no dependen de una conexión de larga vida; en cuanto a una comunicación asíncrona dúplex completa entre el cliente y el servidor, WebSocket debe ser considerado. Finalmente, cuando se requiere una eficiente distribución masiva de datos a las partes interesadas, características de escalabilidad y conociendo que dispositivos finales/clientes pueden encontrarse en ambientes restringidos, particularidades de la propuesta de investigación planteada, el protocolo MQTT presenta las mejores características (Sarafov, 2017). Además este último posee la capacidad de complementarse mediante enlaces y la comunicación entre el protocolo TCP/IP sobre WebSockets y HTML, característica del agente/broker Mosquitto MQTT, intermediario a ser aplicado en la presente propuesta

de investigación, referenciado en el desarrollado del artículo de investigación titulado “Web Performance Evaluation for Internet of Things Applications” (Babovic & Milutinovic, 2016).

La presente investigación y su propuesta de solución se guían y referencian respecto a seguridad y aplicabilidad en el documento de investigación titulado “A Web-Based IoT Solution for Monitoring Data Using MQTT Protocol”, donde se propone una solución de Internet de las Cosas (IoT) basada en la web para monitorear, rastrear y analizar en tiempo real, datos en el área de la agricultura, utilizando el protocolo de transporte de telemetría de publicación / suscripción MQTT. La arquitectura final utilizada se constituye de tres niveles cliente-servidor independientes, la interfaz gráfica de usuario (capa de presentación), las funciones y la lógica de la aplicación (capa de aplicación) y el almacenamiento de datos de computadora (capa de base de datos) (Grgic, Speh, & Hedi, 2016). Además de las investigaciones “MQTT Based Secured Home Automation System” y “A low cost implementation of MQTT using ESP8266”, documentos donde se implementa sistemas de automatización seguros basados en MQTT protocolo de ancho de banda muy ligero, económico y escalable, mediante el uso de sensores, dispositivos y equipos como: WiFi ESP8266, Raspberry pi B+, así como, complementos: Lista de Control de Acceso (ACL), para proporcionar niveles mayores de seguridad. Finalmente, la monitorización de datos se efectúa desde páginas web en cualquier dispositivo de red. La plataforma presenta características de configuración y mecanismos de funcionamiento con un nivel de eficiencia mayor sobre protocolos de comunicación similares, como COAP, TCP / UDP, HTTP, entre otros. En cuanto a la aplicabilidad, el sistema propuesto se puede usar y desplegar tanto en finalidades domóticas como en aplicaciones industriales (Upadhyay, Borole, & Dilepan, 2016) (Kodali & Mahesh, 2016).

La Redes de Sensores Inalámbricos (WSN) plantean desafíos nuevos en comparación con las redes tradicionales. Para responder a estos desafíos, se plantea el paradigma de comunicación centrada en datos, mediante el uso de MQTT-S, una extensión del protocolo de transporte de telemetría de cola de mensajes por publicación / suscripción (MQTT), diseñado para ejecutarse en dispositivos sensores / actuadores de baja gama, a través de WSN. Los aspectos de implementación y desafíos abiertos se detallan en el trabajo “MQTT-S – A Publish/Subscribe Protocol For Wireless Sensor Networks”

(Hunkeler, Truong, & Stanford-Clark, 2008), mismo que puede ser complementado con el estudio y resultados obtenidos del artículo titulado “Secure MQTT for Internet of Things (IoT)”, “A Security Framework for MQTT” y “The Day After Mirai: A Survey on MQTT Security Solutions After the Largest Cyber-attack Carried Out through an Army of IoT Devices” los mismos muestran una versión segura de los protocolos MQTT y MQTT-SN (SMQTT y SMQTT-SN), AugMQTT y AugPAKE en un proyecto de código abierto MQTT Mosquitto; y Denegación de Servicio Distribuido (DDoS) respectivamente. Brindando un complemento al protocolo MQTT existente, permitiendo el cifrado de la comunicación basado en atributos de clave / texto cifrado (KP / CP-ABE) utilizando criptografía ligera. Finalmente se demuestra la viabilidad de los protocolos SMQTT y SMQTT-SN para diversos requisitos de IoT a través de simulaciones y evaluaciones de desempeño (Singh, Rajan, Shivraj, & Balamuralidhar, 2015) (Shin, Kobara, Chuang, & Huang, 2016) (Perrone, Vecchio, Pecori, & Giaffreda, 2017); estos trabajos e investigaciones serán tomados en cuenta, tanto para el desarrollo de la actual propuesta de solución planteada así también para recomendar posibles complementos y trabajos futuros.

Con la indagación y estudio de trabajos, artículos y publicaciones actuales referentes al tema de la propuesta planteada, su desarrollo y plataforma se verá influenciada con los objetivos, resultados y conclusiones obtenidas por el análisis del estado del arte y antecedentes investigativos consultados, teniendo como referentes los trabajos citados a continuación: El concepto de Internet de Vehículos (IoV) desarrollado en el artículo denominado “A MQTT-based Guide and Notification Service System”, estudio que propone una guía basada en el protocolo Message Queue Telemetry Transport (MQTT), estableciendo una arquitectura de cuadrícula jerárquica y geográfica que proporciona servicios basados en la ubicación, información del vehículo, carretera, medio ambiente, entre otras (Dow, Cheng, & Hwang, 2016). Las características de sistemas inteligentes y dispositivos distribuidos, redes de Comunicación Máquina a Máquina (M2M) y capacidad para automatizar entornos industriales y domésticos con gran eficacia; temas abordados y detallados en el trabajo “Machine to Machine Communication for Smart Systems using MQTT” (Ullas, Anush, Roopa, & Govinda, 2014). El trabajo de investigación titulado “muMQ: A Lightweight and Scalable MQTT Broker” estudio efectuado al intermediario/bróker muMQ, un agente ligero y escalable del protocolo de

comunicación MQTT que aprovecha un TCP / IP de nivel de usuario (Pipatsakulroj, Visoottiviseth, & Takano, 2017). Finalmente, el estudio “A Scalable and Low-Cost MQTT Broker Clustering System” del corredor/agente/brokers Mosquitto MQTT para la gestión e intercambio de mensajes entre dispositivos en términos de publicación / suscripción. Se presentan resultados como características de escalabilidad, relativa facilidad de aplicación, seguridad, seguimiento y análisis del rendimiento. Además, la implementación no posee redundancia para datos o mensajes (Jutadhamakorn, Pillavas, & Visoottiviseth, 2017).

6.4 Justificación

En un mercado cada vez más tecnológico y competitivo, la transformación digital se alza como una necesidad empresarial. Una revolución en la manera de realizar los procesos industriales y que conlleva ciertos cambios cuyo impacto afecta a la organización en su totalidad.

El envío, recepción y análisis de cantidades representativas de información a diversos niveles representa uno de los principales problemas en los procesos de fabricación, La aplicación del paradigma de Internet de las Cosas (IoT) y sus tecnologías dentro de un entorno industrial permite eliminar las barreras de conectividad tanto a nivel vertical como horizontal. Y posibilita por un lado la obtención de información de valor de múltiples fuentes, lo que permite analizarla en detalle para extraer nuevo conocimiento; por otro lado, se mejora la comunicación entre diversos actores, mejorando la eficiencia general de los procesos.

El objetivo principal del concepto de Industria 4.0 y su aplicación, además de mejorar la eficiencia, es optimizar los procesos industriales y la utilización de recursos para adaptarse eficazmente a los cambios en la demanda y los mercados, integrando con mayor facilidad en el proceso productivo las necesidades de toda la cadena de valor de la empresa, desde los proveedores hasta los consumidores. A través de: procesos flexibles que se adapten a las necesidades del mercado, ciclos más cortos y rentables en el diseño, fabricación y venta de productos tipificados, trato personalizado al cliente, series de producción más rápidas y eficaces.

6.5 Objetivos

6.5.1 Objetivo General

Desarrollar una plataforma de Internet de las Cosas (IoT) basada en el concepto de Industria 4.0, para la monitorización de un proceso industrial.

6.5.2 Objetivos Específicos

- Establecer las características de conectividad, equipos seguridad y visualización, para la aplicación y puesta en funcionamiento del protocolo de comunicación MQTT
- Desarrollar una Plataforma IoT mediante el protocolo de comunicación M2M MQTT.
- Validar la plataforma de desarrollada mediante pruebas de funcionamiento y respuesta con un proceso industrial básico.

6.6 Análisis de Factibilidad

La presente propuesta posee factibilidad de ejecución debido a que cuenta con disposición de herramientas técnicas, operativas y demás características necesarias para su desarrollo, como son:

6.6.1 Factibilidad técnica

Se posee factibilidad técnica gracias al conocimiento técnico y científico específico del Investigador, así como la existencia de documentación relacionada con el desarrollo y solución de problemas para la implementación de la propuesta planteada.

6.6.2 Factibilidad operativa

El investigador dispone de la experiencia en labores de investigación, así como del entorno de desarrollo necesario y adecuado para la implementación de la propuesta, su puesta en marcha y posteriores pruebas de funcionamiento.

6.6.3 Factibilidad económica

Existe factibilidad económica de la presente propuesta de investigación, debido a la accesibilidad de equipos, materiales y al costo relativamente bajo que supone su desarrollo. Implementación que será autofinanciada por el Investigador.

6.7 Fundamentación Científico Técnica

6.7.1 Comparación y selección de protocolo

El Internet de las cosas se está expandiendo a un ritmo acelerado y cada año dispositivos restringidos que dependen de la intercomunicación se están desplegando. Conocer la sobrecarga que un mecanismo de comunicación agrega al sistema puede ser de gran importancia por su óptima utilización y para evitar la degradación del rendimiento. Con la derivación y comparación general de los protocolos WebSocket, CoAP y MQTT en el trabajo investigativo “Comparison of IoT Data Protocol Overhead”, con características restringidas y de requerimiento masivo, se determinó que: el uso de CoAP conlleva la desventaja de una mayor probabilidad en la pérdida de paquetes, sin embargo, presenta un mecanismo de solicitud / respuesta que lo hace adecuado para casos de uso general que no dependen de una conexión de larga vida. Por otro lado, logrando una comunicación asíncrona dúplex completa entre el cliente y el servidor el uso de WebSocket es imprescindible. En cuando a una eficiente distribución masiva de datos a las partes interesadas, selección simplificada del nivel de calidad del servicio y bajo consumo de recursos, el protocolo MQTT presenta las mejores ventajas (Sarafov, 2017).

Conforme lo detallado y analizado en el párrafo anterior y la fundamentación bibliográfica especificada en los literales del presente documento 2.5.12 Protocolo OPC UA, 2.5.13 Protocolo AMQP, 2.5.14 Protocolo CoAP y 2.5.15 Protocolo MQTT; y en contraste con las características necesarias de la plataforma de Internet de las Cosas (IoT) propuesta en la presente investigación. El protocolo de Transporte de Telemetría por Cola de Mensajes (Message Queue Telemetry Transport, MQTT), presenta las particularidades requeridas para el completo y adecuado desarrollo de la plataforma; por tanto, es el protocolo a utilizar.

6.7.2 MQTT

MQTT (Message Queue Telemetry Transport), es un protocolo usado para la Comunicación Máquina a Máquina (machine-to-machine, M2M) en el Internet de las Cosas (Internet of Things, IoT). Este protocolo está orientado a la comunicación de sensores, debido a que consume muy poco ancho de banda y puede ser utilizado en la mayoría de los dispositivos empotrados con pocos recursos (CPU, RAM). Un ejemplo de uso de este protocolo es la aplicación de Facebook Messenger tanto para Android e Iphone.

La arquitectura de MQTT sigue una topología en estrella, como se muestra en la Figura 9, con un nodo central que hace de servidor o "broker" con una capacidad de hasta 10000 clientes. El broker es el encargado de gestionar la red y de transmitir los mensajes, para mantener activo el canal, los clientes mandan periódicamente un paquete (PINGREQ) y esperan la respuesta del broker (PINGRESP). La comunicación puede ser cifrada entre otras muchas opciones (Munner, Mohammed, & Shadi, 2017).

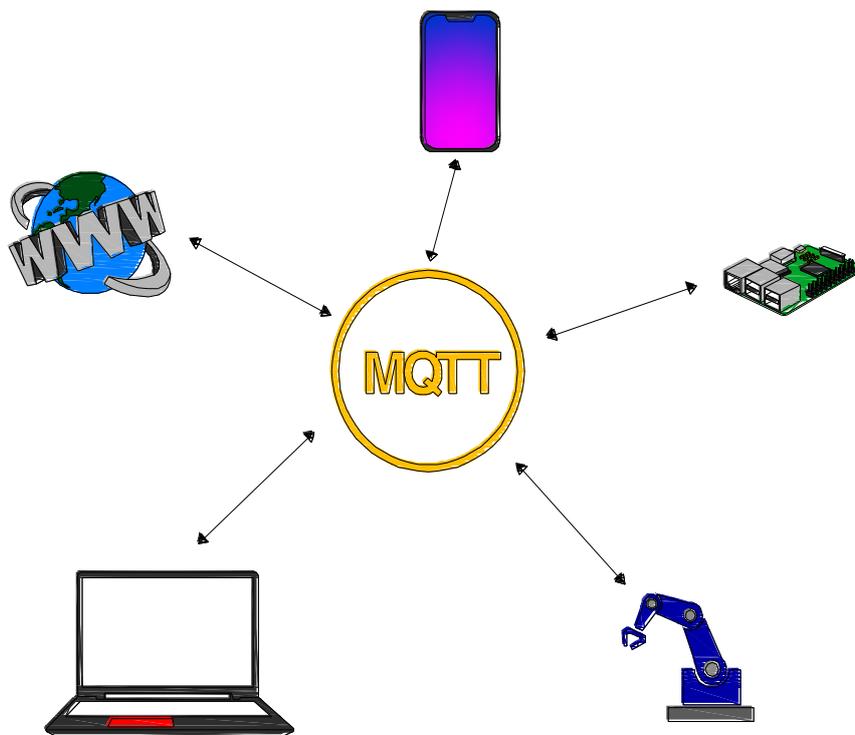


Figura 9: Arquitectura MQTT
Elaborado por: El Investigador

6.7.3 Broker Mosquitto

Eclipse Mosquitto es un Broker MQTT de código abierto, es decir, un intermediario de mensajes (con licencia EPL / EDL). Es liviano y adecuado para su uso en todos los dispositivos, desde computadoras de una sola placa de baja potencia hasta servidores completos.

El protocolo MQTT proporciona un método ligero para llevar a cabo la mensajería utilizando un modelo de publicación/suscripción. Esto lo hace adecuado para la mensajería de Internet de las Cosas (IoT), como sensores de baja potencia o dispositivos móviles como teléfonos, computadoras integradas o microcontroladores.

El proyecto Mosquitto también proporciona una biblioteca de C y Python para implementar servidores y clientes MQTT, y los muy populares clientes de línea de comandos `mosquitto_pub` y `mosquitto_sub` de MQTT. Mosquitto es parte de la Fundación Eclipse y es un proyecto iot.eclipse.org (Eclipse, 2016).

6.7.4 Servidor HTTP Apache

Es un servidor web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux), Microsoft Windows, Macintosh entre otras, que implementa el protocolo HTTP según la normativa RFC 2616.

Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, entre sus ventajas se puede destacar: modular, de código abierto, multi-plataforma, extensible, y Popular (fácil conseguir ayuda / soporte)

Apache es usado principalmente para enviar páginas web estáticas y dinámicas en la World Wide Web. Muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o que utilizarán características propias de este servidor web (Apachecon , 2015).

6.7.5 PHPMyAdmin

PHP, Hypertext Preprocessor (preprocesador de hipertexto), es un lenguaje de programación de propósito general de código del lado del servidor, originalmente

diseñado para el desarrollo web de contenido dinámico. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera el código HTML resultante.

PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes. Puede ser usado en la mayoría de los servidores web al igual que en muchos sistemas operativos y plataformas sin ningún costo.

PHPMYAdmin es una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web, utilizando Internet. Actualmente puede crear y eliminar Bases de Datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en campos, administrar privilegios, exportar datos en varios formatos y está disponible en 72 idiomas. Entre las especificaciones proveídas por el programa incluyen: interfaz web para la gestión gráfica, manejador de base de datos MySQL, MariaDB y Drizzle, importación de datos desde CSV y SQL, exporta datos a varios formatos CSV, SQL, XML, PDF, ISO/IEC 26300, y la administración de múltiples servidores (phpMyAdmin, 2019).

6.7.6 Python

Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, usa tipado dinámico y es multiplataforma.

Es administrado por la Python Software Foundation. Posee una licencia de código abierto, denominada Python Software Foundation License, que es compatible con la Licencia pública general de GNU.

Python es un lenguaje de programación multiparadigma, es decir, más que forzar a los programadores a adoptar un estilo particular de programación, permite varios

estilos: programación orientada a objetos, programación imperativa y programación funcional, además de otros paradigmas soportados mediante el uso de extensiones.

Python usa tipado dinámico, cuando se declara una variable, no es necesario decirle de que tipos son los datos (**int**, **string**, **float**, etc.). La variable se adapta a lo que escribimos cuando se ejecuta el programa; además usa conteo de referencias para la administración de memoria (python, 2019).

6.7.7 Controlador Lógico Programable (PLC) Siemens S7 1200

Es un Controlador Lógico Programable (PLC), modular compacto de la compañía SIEMENS, diseñado para aplicaciones de gama baja de automatización discreta y autónoma, Figura 10.

El PLC a utilizar en el desarrollo de la propuesta es un SIMATIC S7 1200 CPU 1214C AC/DC/Rly, con alimentación de 85-264 VAC, 47-63 Hz, 300 mA. Posee 14 entradas digitales de 24 VDC a 6mA/pt, 10 salidas digitales tipo Rly de 30VDC/250VAC a 2A/pt. El Sistema de comunicación a utilizar es el protocolo Profinet (Profibus Ethernet), definido como un estándar de redes de área local para computadoras con acceso al medio por detección de la onda portadora y con detección de colisiones (Siemens, 2014).

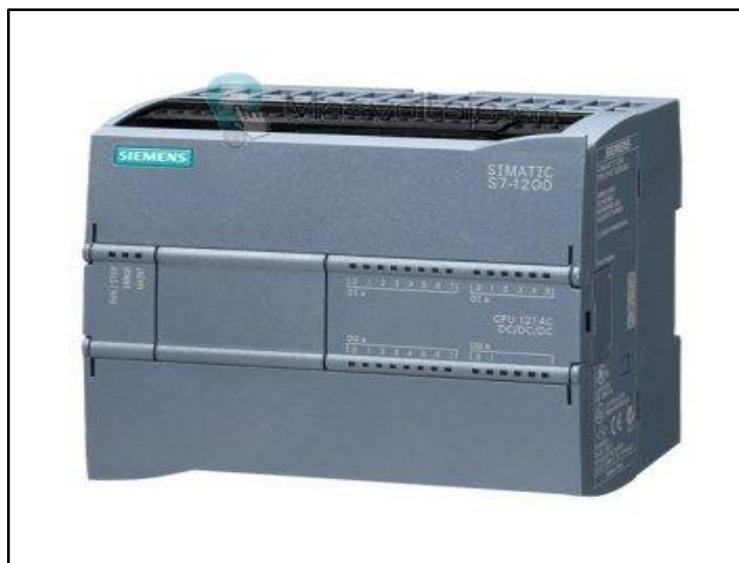


Figura 10: Controlador Lógico Programable (PLC) S7 1200 1214 AC/DC/Rly
Fuente: (Siemens, 2014)

6.7.8 Regulador por Control Difuso

La lógica difusa es la representación de información imprecisa, ambigua o vaga, utilizada para efectuar operaciones en los conceptos que están fuera de las definiciones de la lógica booleana. Se basa en lo relativo de lo observado como posición diferencial, tomando dos valores aleatorios, pero contextualizados y referidos entre sí. Esta lógica se adapta mejor al mundo real en el que vivimos, e incluso puede comprender y funcionar con nuestras expresiones, del tipo "hace mucho calor", "no es muy alto", "el ritmo del corazón está un poco acelerado", entre otras. La clave de esta adaptación, se basa en comprender los cuantificadores de nuestro lenguaje, establecer conjuntos difusos y definir las operaciones necesarias sobre los conjuntos en los que se basa esta lógica. Para cada conjunto difuso, existe asociada una función de pertenencia para sus elementos, que indican en qué medida el elemento forma parte de ese conjunto difuso (Dote, 1995).

La lógica difusa se utiliza cuando la complejidad del proceso en cuestión es muy alta y no existen modelos matemáticos precisos, para procesos altamente no lineales y cuando se envuelven definiciones y conocimiento no estrictamente definido. En cambio, no es una buena idea usarla cuando un modelo matemático ya soluciona eficientemente el problema (especialmente en problemas de modelamiento lineal).

Como principal ventaja, cabe destacar los excelentes resultados que brinda un sistema de control basado en lógica difusa a la respuesta veloz y precisa sobre las salidas de un sistema, disminuyendo así las transiciones de estados fundamentales en el entorno físico que controle; además la capacidad de estabilización y control en general. Como principal inconveniente cabe subrayar su relativa complejidad de implementación, dificultad al elegir una correcta función de pertenencia, determinar conjuntos difusos y especificar el efecto de los cuantificadores de nuestro lenguaje en dicha función. El hecho de que cualquier parámetro o característica ingresado en el sistema estuviese mal especificado, haría fallar, muy probablemente, el sistema completo (Sneha & T.S., 2017) (Dote, 1995).

6.7.9 Factory I / O

FACTORY I / O es una herramienta de simulación de fábrica 3D para aprender tecnologías de automatización. Diseñado para ser fácil de usar, permite construir rápidamente una fábrica virtual utilizando una selección de piezas industriales comunes. Incluye también muchas escenas inspiradas en aplicaciones industriales típicas, con diferentes niveles de dificultad para su control.

FACTORY I / O puede ser utilizado como escenario para plataforma de capacitación de PLC SIEMENS y su software TIA PORTAL, como se muestra en la Figura 11, controladores más comunes que se encuentran en aplicaciones industriales. Sin embargo, también se puede utilizar con microcontroladores, SoftPLC, Modbus, entre muchas otras tecnologías (RealGames, 2019).

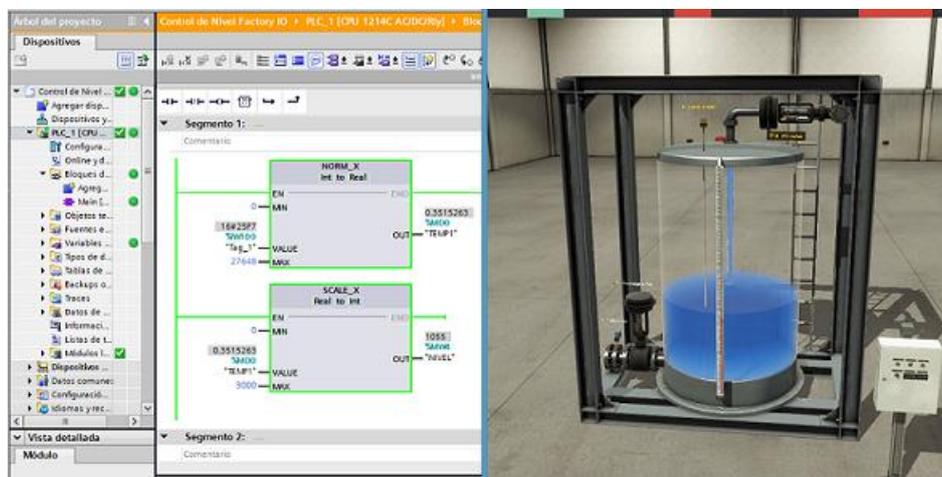


Figura 11: Control proceso Factory I/O – TIA Portal
Elaborado por: El Investigador

6.7.10 MySQL

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general / Licencia comercial por Oracle Corporation y está considerada como la base datos de código abierto más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.

Existen varias interfaces de programación de aplicaciones que permiten, a aplicaciones escritas en diversos lenguajes de programación, acceder a las bases de datos MySQL,

incluyendo C, C++, C#, Pascal, Delphi, Eiffel, Smalltalk, Java, Lisp, Perl, PHP, Python, entre otras, cada uno de estos utiliza una interfaz de programación de aplicaciones específica.

MySQL es una base de datos muy rápida en la lectura, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos, lo que hace a MySQL ideal para este tipo de aplicaciones (MySQL, 2019).

6.7.11 Wireshark

Wireshark es un analizador de protocolos de comunicación gratuito y de código abierto, utilizado para la solución de problemas de red, análisis, software y desarrollo de protocolos de comunicación, además de utilizarse para fines didácticos, educativos y de evaluación.

Originalmente llamado Ethereal, en mayo de 2006 el proyecto fue rebautizado Wireshark debido a problemas de marca registrada. Posee una interfaz gráfica amigable con el usuario, operación sencilla, gráficas estadísticas, opción de organización y filtrado de información; características estándar de un protocolo analizador (similar a la herramienta tcpdump), además de varias particularidades que no se ven en ningún otro producto. Su licencia de código abierto permite a expertos en la comunidad de redes agregar actualizaciones y mejoras. Corre en todas las plataformas de computación populares, incluyendo Unix, Linux y Windows. Los administradores de red, ingenieros de software y desarrolladores lo utilizan para solucionar problemas de red, examinar inconvenientes de seguridad y depurar implementaciones de protocolos.

Entre sus aplicaciones más importantes está el capturar paquetes y analizar su estructura MAC, datagrama IP, segmento de paquete TCP, su contenido y transmisión PDU. Es compatible con alrededor de 500 protocolos de comunicación, incluyendo MQTT, además, su interfaz flexible y gran capacidad de filtrado hacen de Wireshark una herramienta eficaz para las pruebas y el análisis de tráfico de red de la plataforma a desarrollarse. La versión de Wireshark a utilizarse en esta investigación es la 3.0.1. (Wang, Xu, & Yan, 2010).

6.8 Desarrollo de la propuesta

6.8.1 Arquitectura de la propuesta

La arquitectura de la propuesta se presenta en la Figura 12. En la misma se muestra un bróker central que controla la comunicación, un cliente que mantiene comunicación con el controlador del proceso y simulación de la planta; y otro cliente alojado en una página Web desde donde se puede visualizar y controlar parámetros del proceso.

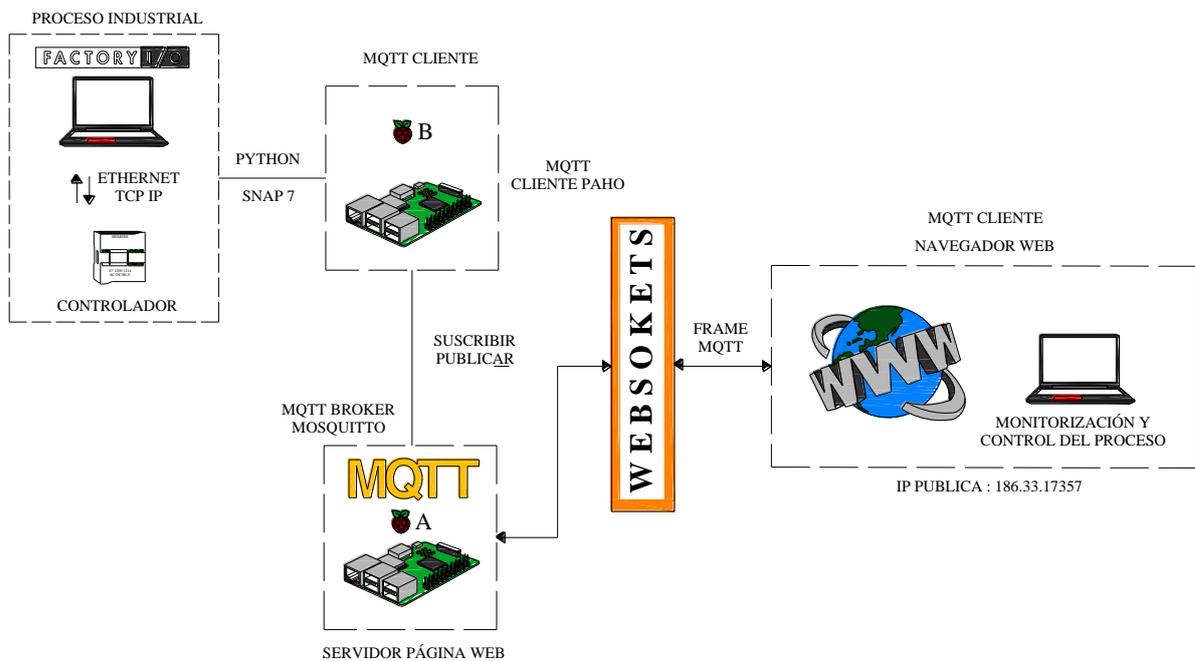


Figura 12: Arquitectura de la plataforma
Elaborado por: El Investigador

6.8.2 Selección de equipos y programas utilizados

Se determina entonces el análisis de características, especificaciones y ventajas de los equipos detallados en la Tabla 5. Conforme el requerimiento de la plataforma a desarrollarse y tomando en cuenta las particularidades del proceso a controlar, como: su simulación virtual en el software Factory I/O (ensamblaje de un producto y el control de nivel en un tanque); la operación con variables de diverso tipo; comunicación entre plataforma, controlador y posterior enlace Cliente/Servidor/Cliente. El controlador a utilizarse es un PLC Siemens S7 1200 1214 AC/DC/Rly, con ventajas como: una equilibrada relación precio/prestaciones, comunicación directa con Factory I/O, posee módulos específicos para el control del proceso a implementarse, conocimiento de su

programación, puesta en marcha y resolución de problemas, además es el controlador más común encontrado en aplicaciones industriales actuales.

Tabla 5: Contraste entre posibles controladores del proceso
Elaborado por: El Investigador

Características		SBC	LOGO	PLC
	Precio	Muy accesible	Accesible	Relativamente alto
	Programación	Código estructurado	Escalera	Escalera
	Robustez	Ninguna	Buena	Excelente
	Escalabilidad	Buena	Buena	Excelente
Aplicabilidad	Industria	Muy poca	Poca	Extendida
	Plataforma	Buena	Buena	Excelente

El ordenador seleccionado a efectuar la comunicación cliente/servidor, es un Computador de Placa Única (SBC, Raspberry Pi 3), equipo que presenta ventajas y características superiores a su competidor directo, Tabla 6. Entre estas: conectividad WiFi y Ethernet integradas, un procesador adecuado para la operación del bróker de comunicación y la gestión completa de mensajería, adecuada accesibilidad multimedia por el continuo uso y enlace con internet, librerías y accesibilidad con el protocolo de comunicación específico MQTT.

Tabla 6: Comparativa entre SBC y Arduino
Elaborado por: El Investigador

Características	Arduino	Raspberry Pi
Precio	Muy accesible	Accesible
Puesta en servicio	Código estructurado	Código estructurado
Procesador	Normal	Más fuerte y veloz
Conectividad (WiFi)	Shield adicionales	Incluida en placa
Aplicación general	Controlador (entradas/salidas)	Varias funcionalidades (servidor)

Luego del análisis de requerimientos y características necesarias para la adecuada implementación de la plataforma propuesta, los equipos y software a utilizar se detallan en la Tabla 7.

Tabla 7: Detalle equipos y programas utilizados
Elaborado por: El Investigador

Ítem	Equipo	Detalle	Descripción	Software	Enlace con:
1	Ordenador	Acceso a internet (Wifi)	Visualiza el proceso	Factory IO	PLC
2	PLC	S7 1200 AC/DC 1214	Controla el Proceso	TIA Portal V13	Factory IO Raspberry Pi 3 con SNAP 7
3	Raspberry	Pi 3	Cliente MQTT	Lenguaje Python	PLC con SNAP 7 Server MQTT
4	Raspberry	Pi 3	Server MQTT	Lenguaje C	Varios Clientes MQTT (WebSockets)
5	Ordenador	Acceso a internet (Wifi)	HMI Cliente MQTT	Página Web	Server MQTT

La definición de las direcciones IP que tendrán cada uno de los dispositivos a utilizarse se designaron conforme el desarrollo y necesidad del proyecto, siendo esto de gran importancia al momento de la configuración final de la plataforma, las direcciones establecidas se muestran en la Tabla 8.

Tabla 8: Detalle direcciones IP de equipos
Elaborado por: El Investigador

Ítem	Dispositivo	Dirección IP
1	Ordenador simulación planta (Factory IO)	192.168.0.1
2	PLC S7 1200	192.168.0.13
3	Raspberry Pi MQTT -cliente	192.168.0.107
4	Raspberry Pi MQTT -servidor (broker)	19.168.0.111
5	IP publica (Red)	186.33.173.57
6	Ordenador de prueba final Página Web	192.168.0.105

6.8.3 Programación PLC y FACTORY IO

Luego del estudio y planteamiento del proceso a controlar, se define que no es necesario utilizar módulos ni dispositivos adicionales en el PLC.

La planta industrial es simulada en el software FACTORY IO el mismo posee conexión directa con el Controlador Lógico Programable (PLC, SIEMENS S7 1200 1214 AC/DC/Rly) utilizado, su interconexión sobre TCP/IP e IP establecidas se detallan en la Figura 13.

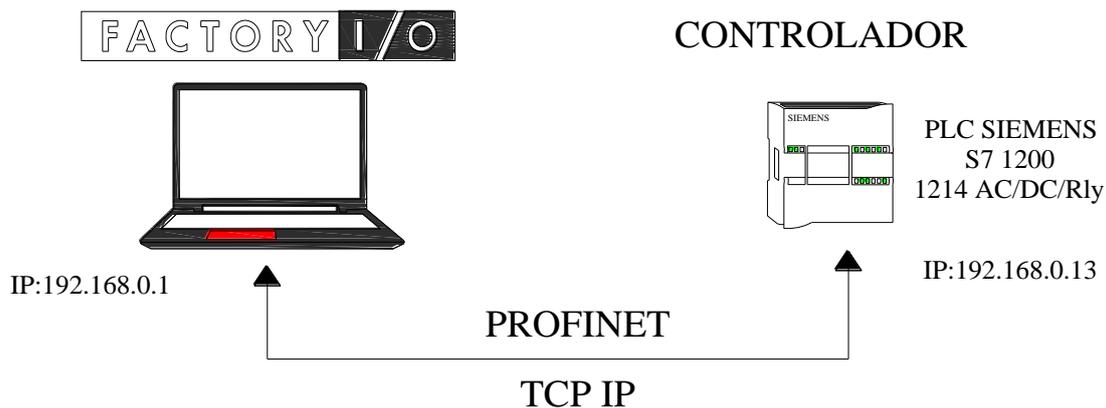


Figura 13: Interconexión Factory I/O y Controlador
Elaborado por: El Investigador

La programación directa del Controlador Lógico Programable (PLC) se realiza en el Software TIA Portal V13, creando, configurando y probando bloques de programación en lenguaje Ladder. Entre las instrucciones más relevantes utilizadas están: NORM_, SCALE_, instrucciones matemáticas, contadores, temporizadores, marcas del sistema, marcas de ciclo, entre otras. La Figura 14 muestra un segmento de la programación final del proceso, la Tabla de variables utilizadas en el PLC para la programación del proyecto se detalla en la Figura 15.

El proceso consta de dos partes, la primera, el ensamble de un material utilizando bandas transportadoras y un brazo con movimiento X y Z; y la segunda, un tanque con control de nivel para su llenado y vaciado. La distribución física de la planta es simulada en el software Factory I/O y se muestra en la Figura 16, mientras la configuración de entradas y salidas para la comunicación con el controlador se expresa en la Figura 17.

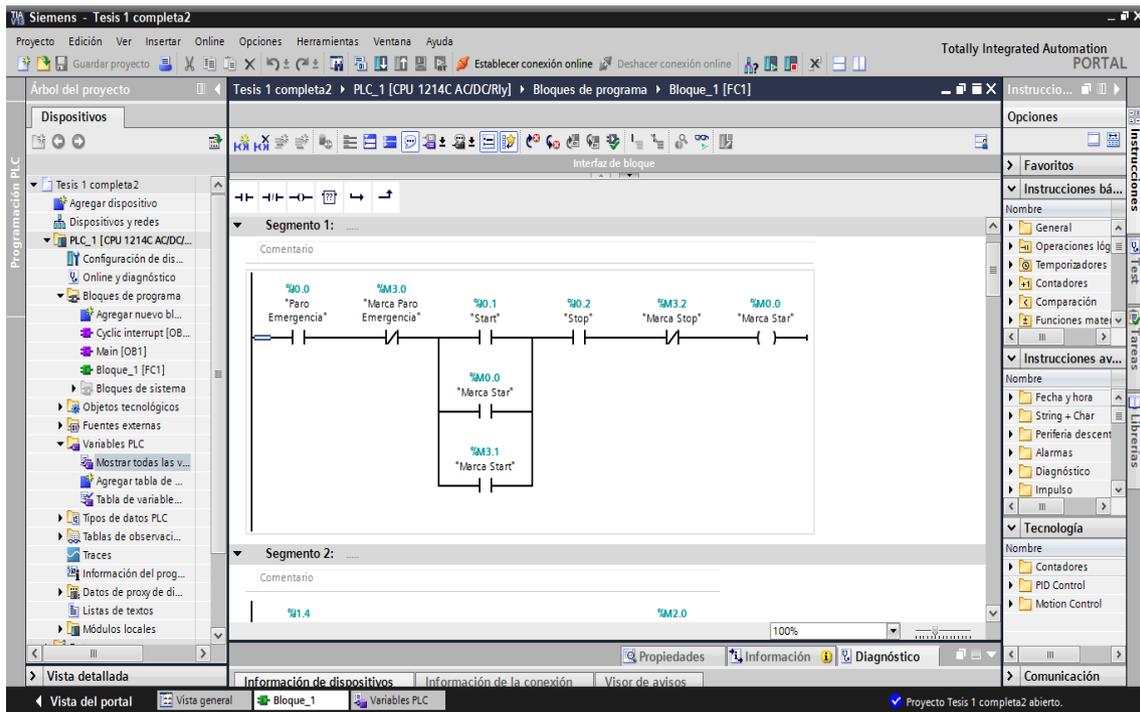


Figura 14: Bloque de programación principal
Elaborado por: El Investigador

Nombre	Tabla de variables	Tipo de datos	Dirección	Rema...	Visibl...	Acces...	Comentario
1 Tag_1	Tabla de variables estándar	Word	%IW102		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	No
2 Tag_12	Tabla de variables estándar	Int	%IW100		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	No
3 Tag_16	Tabla de variables estándar	Bool	%I0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4 Paro Emergencia	Tabla de variables estándar	Bool	%I0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5 Start	Tabla de variables estándar	Bool	%I0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6 Sensor 3	Tabla de variables estándar	Bool	%I0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7 Sensor 4	Tabla de variables estándar	Bool	%I0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8 Stop	Tabla de variables estándar	Bool	%I0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9 Sensor 5	Tabla de variables estándar	Bool	%I0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Si
10 Sensor 6	Tabla de variables estándar	Bool	%I0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11 Sensor 1	Tabla de variables estándar	Bool	%I1.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Si
12 Sensor 2	Tabla de variables estándar	Bool	%I1.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Si
13 Sensor 7	Tabla de variables estándar	Bool	%I1.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	No
14 Reset	Tabla de variables estándar	Bool	%I1.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	No
15 Tag_5	Tabla de variables estándar	Int	%QW100		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	No
16 Valvula Entrada	Tabla de variables estándar	Int	%QW104		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Si
17 Valor Real Tanque	Tabla de variables estándar	Int	%QW102		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	No
18 Valvula Salida	Tabla de variables estándar	Int	%QW106		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	No
19 Banda 1	Tabla de variables estándar	Bool	%Q0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Si
20 Banda 2	Tabla de variables estándar	Bool	%Q0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Si
21 Posicionador 1	Tabla de variables estándar	Bool	%Q0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Si
22 Posicionador 2	Tabla de variables estándar	Bool	%Q0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
23 Brazo Z	Tabla de variables estándar	Bool	%Q0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figura 15: Tabla de variables de programación
Elaborado por: El Investigador

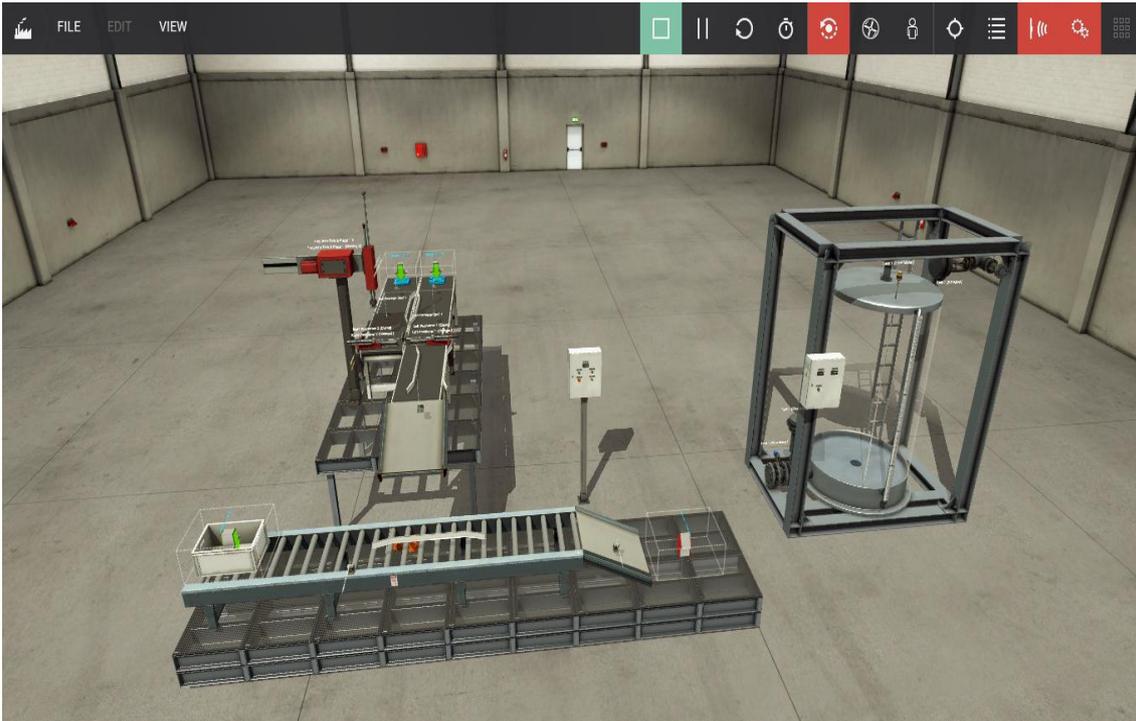


Figura 16: Layout planta en Factory IO
Elaborado por: El Investigador

← DRIVER Siemens S7-1200/1500 [!] CONNECT CONFIGURATION CLEAR

Host: 192.168.0.13

SENSORS		Host: 192.168.0.13		ACTUATORS	
Emergency Stop 1	I0.0	Q0.0	Belt Conveyor (6m) 1	Belt Conveyor (4m) 1	Belt Conveyor (6m) 1
Capacitive Sensor 1	I0.1	Q0.1	Belt Conveyor (4m) 1	Digital Display 1	Digital Display 2
Capacitive Sensor 2	I0.2	Q0.2	Left Positioner 1 (Clamp)	Digital Display 3	Emitter 1 (Emit)
Capacitive Sensor 3	I0.3	Q0.3	Left Positioner 1 (Raise)	Emitter 2 (Emit)	Emitter 3 (Emit)
Capacitive Sensor 4	I0.4	Q0.4	Right Positioner 2 (Clamp)	Left Positioner 1 (Clamp)	Left Positioner 1 (Raise)
Capacitive Sensor 5	I0.5	Q0.5	Two-Axis Pick & Place 1 X	Remover 1 (Remove)	Reset Button 1 (Light)
Capacitive Sensor 6	I0.6	Q0.6	Two-Axis Pick & Place 1 Z	Right Positioner 2 (Clamp)	Right Positioner 2 (Raise)
Capacitive Sensor 7	I0.7	Q0.7	Two-Axis Pick & Place 1 (Grab)	Roller Conveyor (6m) 1	Roller Conveyor (6m) 1
Emergency Stop 1	I1.0	Q1.0	Roller Conveyor (6m) 1	Start Button 1 (Light)	Stop Button 1 (Light)
FACTORY I/O (Paused)	I1.1	Q1.1	Digital Display 1	Stop Button 1 (Light)	Tank 1 (Fill Valve)
FACTORY I/O (Reset)	I1.2	Q1.2	Digital Display 2	Tank 1 (Fill Valve)	Tank 1 (Discharge Valve)
FACTORY I/O (Running)	I1.3	Q1.3	Tank 1 (Fill Valve)	Tank 1 (Discharge Valve)	Two-Axis Pick & Place 1 X
FACTORY I/O (Time Scale)	I1.4	Q1.4	Tank 1 (Discharge Valve)	Two-Axis Pick & Place 1 Z	Two-Axis Pick & Place 1 Z
Left Positioner 1 (Limit)	I1.5	Q1.5	Digital Display 1	Digital Display 2	Two-Axis Pick & Place 1 (Grab)
Left Positioner 1 (Clamped)	I1.6	Q1.6	Digital Display 2	Tank 1 (Fill Valve)	
Potentiometer 1 (V)	I1.7	Q1.7	Tank 1 (Discharge Valve)		
Reset Button 1	IW100 (INT)	(INT) QW100			
Right Positioner 2 (Limit)	IW102 (INT)	(INT) QW102			
Right Positioner 2 (Clamped)	IW104	(INT) QW104			
Start Button 1	IW106	(INT) QW106			
Stop Button 1	IW108	QW108			
Tank 1 (Flow Meter)	IW110	QW110			
Tank 1 (Level Meter)	IW112	QW112			
	IW114	QW114			

Figura 17: Configuración de entradas salidas PLC – Factory IO
Elaborado por: El Investigador

Con el objeto de conseguir una monitorización adecuada del proceso industrial, se desarrolla dos reguladores de control, el primero y comúnmente utilizado Controlador Proporcional, Integral y Derivativo (PID), y el segundo mediante Lógica Difusa (Fuzzy Logic), que permite un control relativo y diferencial de las variables del proceso.

6.8.3.1 Control mediante Lógica Difusa (Fuzzy Logic)

El regulador por lógica difusa se implementa mediante el software Matlab y sus herramientas, Simulink y Fuzzy Logic Toolbox. Se determina entonces el modelo matemático del sistema, tomando en cuenta los parámetros del tanque, como, su forma, variables de entrada/salida, entre otras; obteniendo como resultado la Ecuación 1, que determina la variación del nivel del tanque, cuando este se llena o vacía, modelo matemático para su posterior simulación numérica. El detalle de cálculos y cómputos complementarios del modelo se adjuntan en el Anexo 1.

Ecuación 1: Modelo matemático del tanque

$$\dot{h} = \frac{1}{A} * (f_u - k\sqrt{h}) \quad \text{Ec 1.}$$

Los parámetros necesarios en Matlab – Simulink para el diagrama de Bloques de Control por Lógica Difusa se definen conforme el modelo expuesto en la Figura 18. Donde se observa entradas de señales de referencia, módulos de operaciones y un subsistema “Control de Nivel”, que contiene el modelo numérico del Tanque para simular su volumen y llenado, características del sistema necesarias para ejecutar pruebas de funcionamiento.

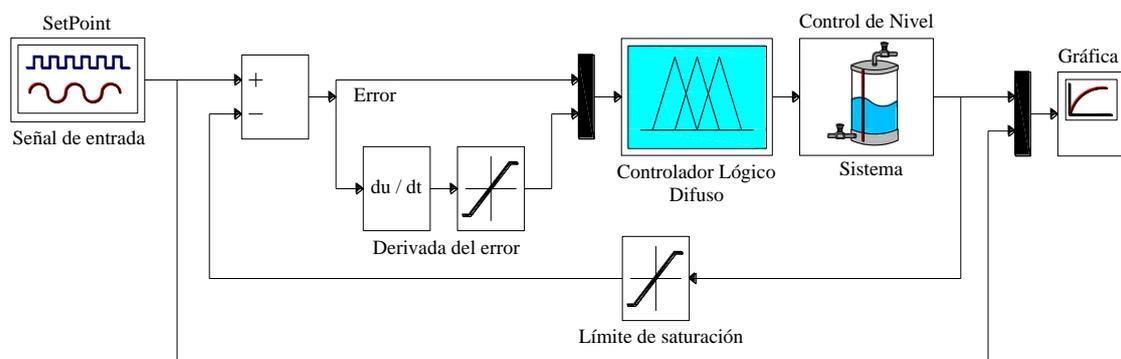


Figura 18: Modelo matemático del sistema
Elaborado por: El Investigador

La simulación numérica del controlador difuso se efectúa en la herramienta Fuzzy Logic Toolbox de Matlab, cuyo motor de inferencia permite establecer las variaciones y señales de salida adecuadas al requerimiento de control del sistema, a través de líneas de código y programación determinadas en el modelo, como: reglas de inferencia y del controlador, variables lingüísticas y funciones de pertenencia. Además del posterior análisis del comportamiento del control difuso.

Las variables utilizadas para el proceso son tres, dos de entrada (Nivel y Error) y una salida (Válvula), con un método de inferencia difusa directo de tipo Mamdani y método de defuzzificación de tipo centroide, como se muestra en la Figura 19. Mientras que los términos lingüísticos utilizados son: Nivel (Alto, Normal y Bajo); Error (Positivo y Negativo); y la salida Válvula (Cerrar_Rapido, Cerrar_Lento, No_Cambiar, Abrir_Lento y Abrir_Rapido) cada una de ellas asociada a la función de pertenencia (Membresía) que rige cada uno de sus elementos. Las entradas Nivel y Error, poseen una función de membresía tipo Gaussiana, con un rango de medición de $[-1, 1]$ y $[-0.1, 0.1]$ respectivamente; la variable salida opera con una función de membresía tipo Triangular con un rango de $[-1, 1]$. El detalle de gráficas y capturas de reglas de inferencia y funciones de membresía de las variables de adjunta en el Anexo 2.

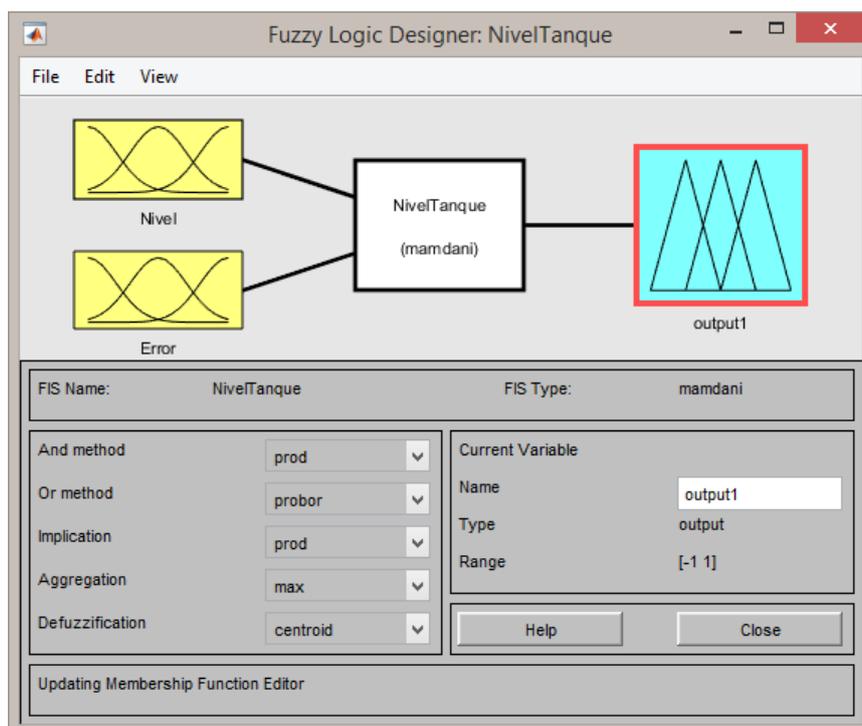


Figura 19: Modelo difuso Fuzzy Logic Toolbox
Elaborado por: El Investigador

El comportamiento del sistema, se establece en función del valor de la entrada Nivel (Error_Nivel), de la velocidad de cambio de la entrada Error (d_Error) y de las reglas de inferencia determinadas, como se muestra en la Figura 20. La gráfica de la superficie de decisión conforme las reglas programadas sobre las premisas de las entradas Nivel y Error y su consiguiente salida se muestran en la Figura 21.

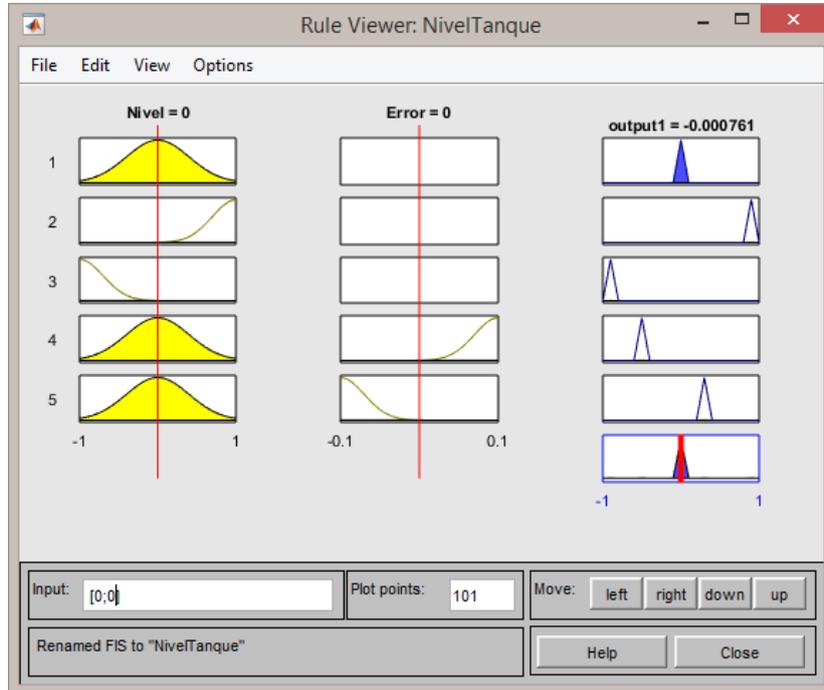


Figura 20: Visualización de evaluación de reglas
Elaborado por: El Investigador

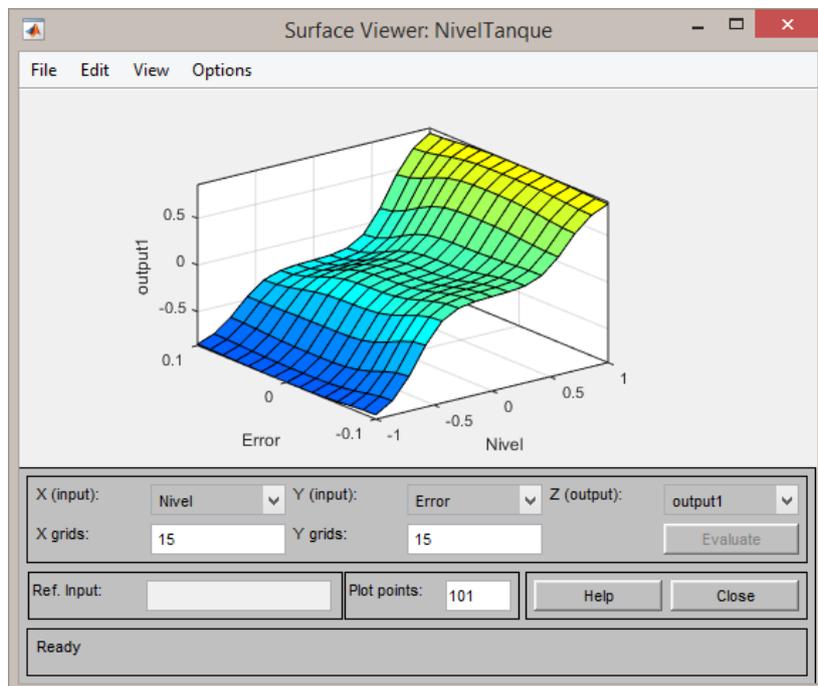


Figura 21: Gráfica superficie de decisión
Elaborado por: El Investigador

Una vez comprobado el correcto funcionamiento del regulador difuso mediante el seguimiento de varias señales de referencia y analizadas sus gráficas de establecimiento, tiempo, operación y ley de control. Se implementa la comunicación entre el Controlador Lógico Programable (PLC, S7 1200) y el software Matlab, mediante la plataforma KEPServerEX y el OPC Quick Client.

6.8.3.2 Control mediante Regulador PID

El control del proceso mediante regulador PID se simplifica respecto al anterior, debido a que el Controlador Lógico Programable (PLC) utilizado, posee una función específica para este tipo de aplicaciones. Mediante el software TIA Portal V13 se realiza la programación y puesta en marcha del bloque PID_Compact, configurando y estableciendo los parámetros necesarios conforme el sistema a controlar. La sintonización se efectúa con la herramienta Autotuning embebida en el controlador, los datos obtenidos se contrastan con los cálculos de sintonización por el método de Ziegler Nichols detallados en el Anexo 3, donde también se expone las pruebas de funcionamiento del controlador realizadas. La Figura 22, muestra la programación para el funcionamiento del controlador PID del PLC Siemens S7 1200, establecida en el software TIA Portal V13.

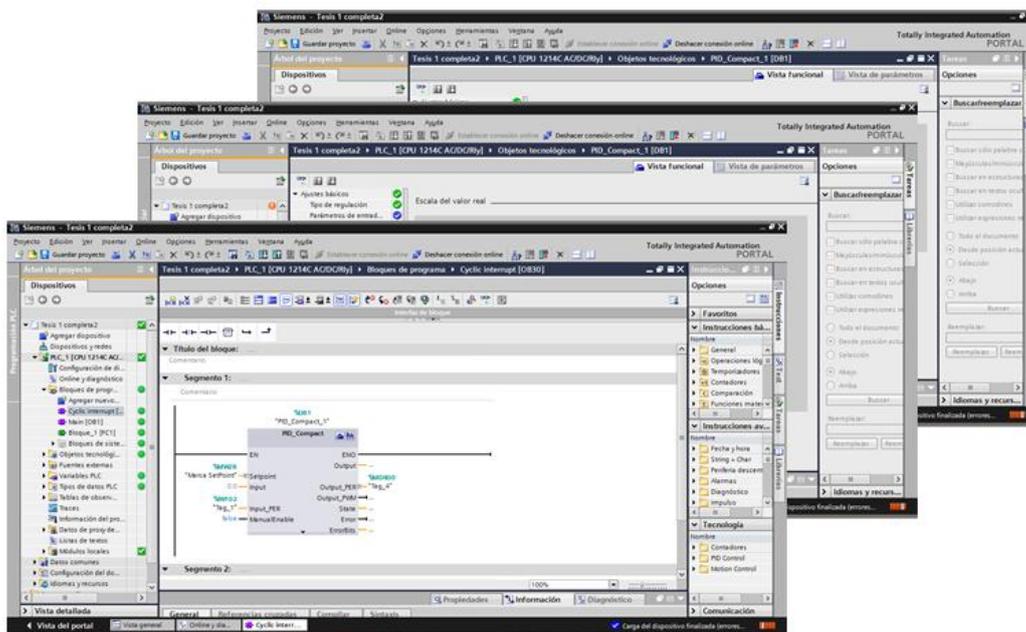


Figura 22: Programación PID_Compact en TIA V13
Elaborado por: El Investigador

Las pruebas de funcionamiento y el posterior análisis de respuestas de control, demuestran que el controlador Lógico Difuso presenta mejor rendimiento que su competidor, aunque su implementación inicial y sintonización son relativamente complejas, sus características de control, respuesta ante perturbaciones, magnitud de sobre impulso, tiempo de establecimiento, escalabilidad y robustez son superiores. Sin embargo, para la implementación específica del presente proyecto, con un proceso a controlar simulado en Factory I/O, con características controladas y conociendo la posterior implementación del controlador como cliente del protocolo de comunicación MQTT; el regulador PID presenta características aptas y resultados de control adecuados al proceso, conforme el contraste detallado en la Tabla 9.

Por tanto, para la implementación de la plataforma de Industria 4.0 para la monitorización de un proceso industrial se utiliza el controlador PID, embebido en el Controlador Lógico Programable (PLC S7 1200).

Tabla 9: Contraste control difuso y PID
Elaborado por: El Investigador

Características / Proceso	Controlador			
	PID		Difuso	
Implementación	Simple	✓	Compleja	
Embebido en PLC	Si	✓	No	
Sintonización	Simple	✓	Compleja	
Tiempo de establecimiento	Bueno	✓	Bueno	✓
Robustez	Normal	✓	Buena	
Aplicabilidad en plataforma	Si	✓	No	
Respuesta ante perturbaciones	N/A		N/A	
Escalabilidad	N/A		N/A	
Magnitud de sobre impulso	N/A		N/A	

6.8.4 PLC y RASPBERRY SNAP 7

La comunicación de datos entre el Controlador Lógico Programable (PLC) y la Raspberry a utilizar como cliente, se establece mediante la herramienta de programación multiplataforma Snap 7 en un archivo de extensión .py, permitiendo obtener los datos del PLC Siemens S7 a través del protocolo Profinet (Profibus Ethernet). Para ello es necesario la instalación, declaración y definición de librerías y variables a utilizar, además de conocer la dirección IP del módulo controlador y dirección de la memoria donde se realizará la lectura y/o escritura.

La comunicación se efectúa mediante el método getMem(), además de la determinación de clases, funciones y estructura necesaria para el correcto funcionamiento del proceso de lectura y escritura en las direcciones de memoria específicas del Controlador Lógico Programable (PLC, S7 1200 1214 AC/DC/Rly), utilizado en este caso como cliente. Según el extracto de sintaxis de código detallado a continuación:

```
class S71200(): # Determinación clase S71200

plc = S71200.S71200 ("192.168.0.13"): # Variable que llama a clase s71200
y dirección IP del controlador
```

```
def getMem(self,mem,returnByte=False): #Método para lectura/escritura
```

Sintaxis para efectuar la escritura en la salida y memoria tipo real del controlador:

```
plc.writeMem('QX0.0',True) # Activar la salida 1 (Q0.0)

plc.writeMem('mw20',1200) # Escribir el valor 1200 en la memoria
mw20
```

Para realizar la lectura de entradas (IX0.0), marcas (MX0.1) y palabras enteras (MW20)

```
print plc.getMem('MX0.1') # Leer el bit de memoria 2 (M0.1)

print plc.getMem('freal100') # Leer el número real almacenado en la
memoria 100
```

Con esta aplicación se efectúa la lectura y/o escritura sobre el estado de las variables de entradas y salidas del autómata programable utilizado y así permitir su control remoto.

Línea de código en la Raspberry Cliente en lenguaje Python utilizado para la definición de variable MW74 del proceso:

```
mw74=plc.getMem('MW74')          # Extracción de información de variable
```

Extracto de código de programación para determinación de condicionales en el movimiento de bandas transportadoras del proceso:

```
if plc.getMem('QX0.7')==True and brazo==0:      # Condicional lectura PLC
```

Código necesario para la publicación de información en la dirección de memoria del controlador MW50 y publicada en el Topic PLC/PLC1/ENSAMBLAJE/PRODUCCION/POSICIONADOR_1.

```
Mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/POSICIONADOR_1",plc.getMem('MW50'))
```

6.8.5 Aplicación del Protocolo MQTT

El bróker aplicado en la plataforma propuesta es Mosquitto MQTT, protocolo de transporte de mensajería de publicación / suscripción de Cliente a Servidor; liviano, abierto y aplicable a la Comunicación Máquina a Máquina (M2M) e Internet de las Cosas (IoT), donde las limitaciones en el ancho de banda de la red no son relevantes. La estructura efectuada en la propuesta, programas, software y conexión de la plataforma se muestra en la Figura 23.

Su implementación inicia con la instalación del Broker en la Raspberry Pi Server, mediante el cumplimiento del código y proceso detallado en el Anexo 6. Tomando en cuenta la aplicación y comunicación posterior con navegadores y páginas web, la disposición necesaria para determinar el enlace del protocolo MQTT sobre WebSockets se detalla en el literal 6.8.11 WebSockets.

Una vez instalado y configurado correctamente el bróker Mosquitto MQTT, se obtendrá el detalle de puertos de comunicación, ejecución de procesos nativos y demás información establecida, por defecto el bróker utilizado opera en el puerto 1883, además se identifica características de nuevas conexiones y desconexiones de clientes, avisos, errores y demás referencias relevantes.

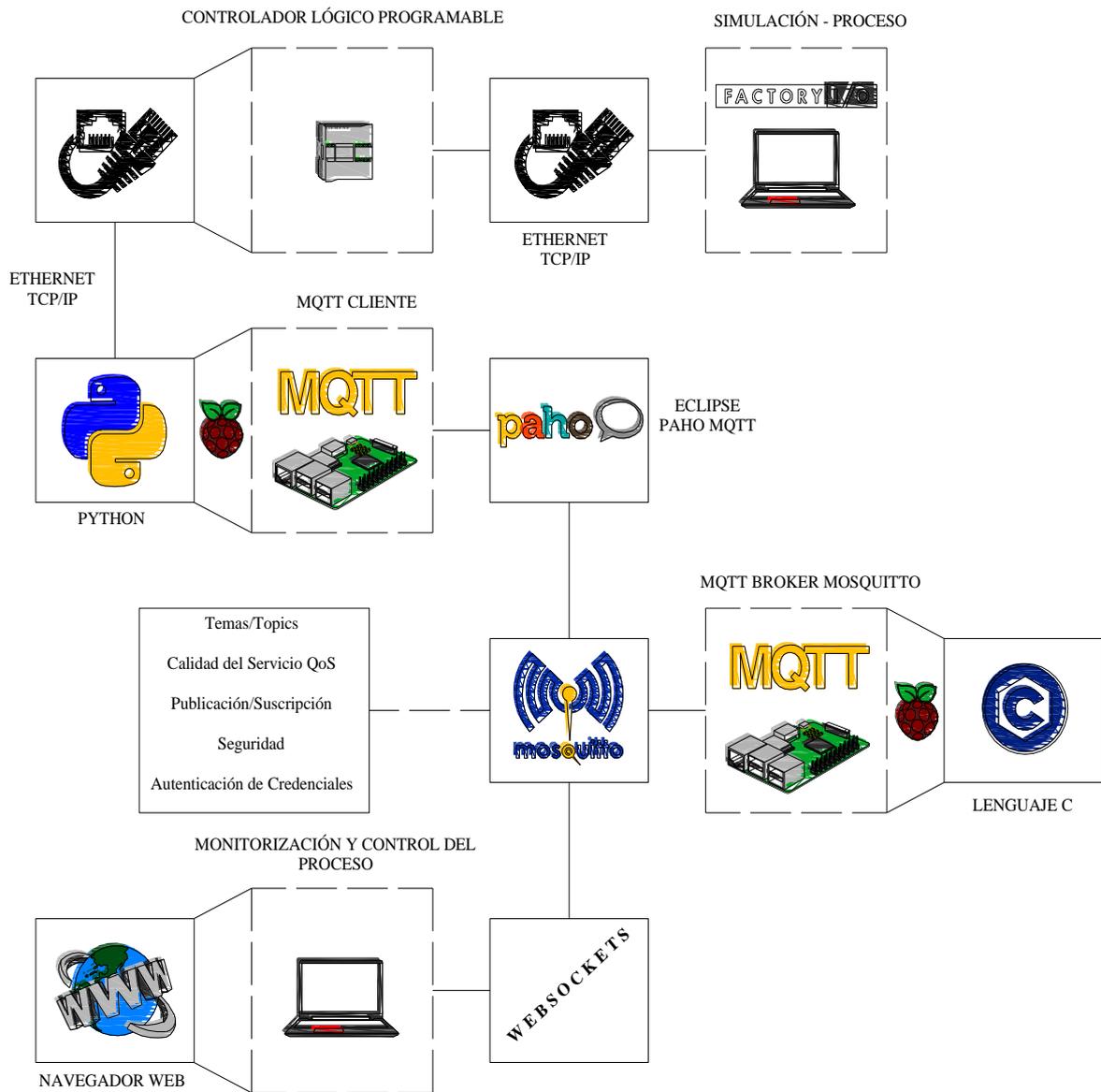


Figura 23: Estructura e interconexión de la plataforma
Elaborado por: El Investigador

La operación con el bróker Mosquitto mediante librerías para uso en lenguaje C (bajo nivel), es con el objetivo de realizar la implementación de clientes MQTT tanto editores como suscriptores sin mayores inconvenientes en la comunicación. Estos clientes pueden presentar características y recursos limitados, sin embargo, al poseer y trabajar bajo una pila TCP/IP tanto en el cliente como en el intermediario, la conexión será viable. Cabe acentuar que el protocolo MQTT se basa en TCP/IP, como muestra la Figura 24.

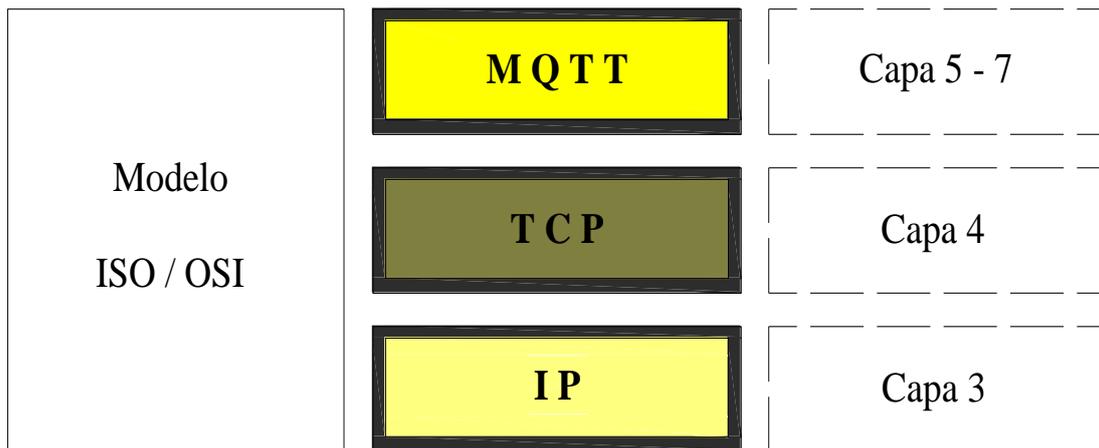


Figura 24: Esquema protocolo MQTT – TCP/IP
Elaborado por: El Investigador

Con la disponibilidad de ejecutar Clientes MQTT en varios lenguajes de programación, su implementación se efectuó de manera relativamente simplificada. La instalación de librerías en C para su operación con el bróker Mosquitto se verifica mediante lo obtenido en la Raspberry. Posteriormente el establecimiento de conexión de las librerías del bróker Mosquitto con C se efectúa y comprueba mediante los códigos y compilación observadas en la Figura 25.

Para la posterior puesta en marcha del bróker Mosquitto MQTT se ejecuta el siguiente código, la Figura 26 muestra el arranque del servicio después de las configuraciones posteriores de Puerto y lectura de mensajes sobre WebSockets.

```
mosquitto -c /etc/mosquitto/mosquitto.conf
```

Finalmente, y adelantando el desarrollo contractual de la plataforma, es relevante aludir la configuración del router mediante su administración local, detallado en el Anexo 5; procedimiento que permite el reenvío de puertos a la dirección IP de la Raspberry bróker 192.168.0.111, con el objetivo de publicar la plataforma Web en línea, mediante la utilización de la IP pública 186.33.173.57 del proveedor de internet local. Con tal aplicación la monitorización y control del proceso industrial de la presente investigación, se podrá realizar desde cualquier dispositivo con acceso a internet a nivel mundial.

```

pi@raspberrypi: ~/Codes
ts
Makefile:273: recipe for target 'install' failed
make: *** [install] Error 1
pi@raspberrypi:~/paho.mqtt.c $ cd /usr/local/lib/
pi@raspberrypi:/usr/local/lib $ ls
libmosquittopp.so      libpaho-mqtt3as.so      libpaho-mqtt3cs.so.1
libmosquittopp.so.1    libpaho-mqtt3as.so.1    libpaho-mqtt3cs.so.1.0
libmosquitto.so        libpaho-mqtt3as.so.1.0  pypy2.7
libmosquitto.so.1      libpaho-mqtt3c.so        python2.7
libpaho-mqtt3a.so       libpaho-mqtt3c.so.1     python3.5
libpaho-mqtt3a.so.1    libpaho-mqtt3c.so.1.0
libpaho-mqtt3a.so.1.0  libpaho-mqtt3cs.so
pi@raspberrypi:/usr/local/lib $ cd
pi@raspberrypi:~ $ cd Codes/
pi@raspberrypi:~/Codes $ ls
a.out  conect_mqtt.py  hola.c  holaPi  mqtt_c.c  tesis_mqtt
pi@raspberrypi:~/Codes $ sudo gcc mqtt_c.c -o holamqttc -lpaho-mqtt3c
pi@raspberrypi:~/Codes $ ls
a.out  conect_mqtt.py  hola.c  holamqttc  holaPi  mqtt_c.c  tesis_mqtt
pi@raspberrypi:~/Codes $ ./holamqttc
Waiting for up to 10 seconds for publication of Hello World!
on topic pi for client with ClientID: SERVER_C
Message with delivery token 1 delivered
pi@raspberrypi:~/Codes $

```

Figura 25: Confirmación librerías MQTT para lenguaje en C
Elaborado por: El Investigador

```

pi@raspberrypi: ~
login as: pi
pi@192.168.0.111's password:
Linux raspberrypi 4.14.70-v7+ #1144 SMP Tue Sep 18 17:34:46 BST 2018 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Mar  9 17:13:19 2019 from 192.168.0.105

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.

pi@raspberrypi:~ $ mosquitto -c /etc/mosquitto/mosquitto.conf
1553953220: mosquitto version 1.4.10 (build date 2018-11-16 16:06:42+0000) starting
1553953220: Config loaded from /etc/mosquitto/mosquitto.conf.
1553953220: Opening websockets listen socket on port 9001.
1553953220: Opening ipv4 listen socket on port 8883.
1553953220: Opening ipv6 listen socket on port 8883.

```

Figura 26: Puesta en servicio de bróker MQTT Mosquitto
Elaborado por: El Investigador

6.8.6 Topología

El protocolo MQTT sigue una topología en estrella, con el nodo central Server Raspberry A como bróker, el que opera a los dos clientes de la aplicación desarrollada, sin embargo, puede trabajar con gran cantidad de clientes. El bróker se encarga de gestionar la red y transmitir los mensajes. La topología de la propuesta se muestra en la Figura 27.

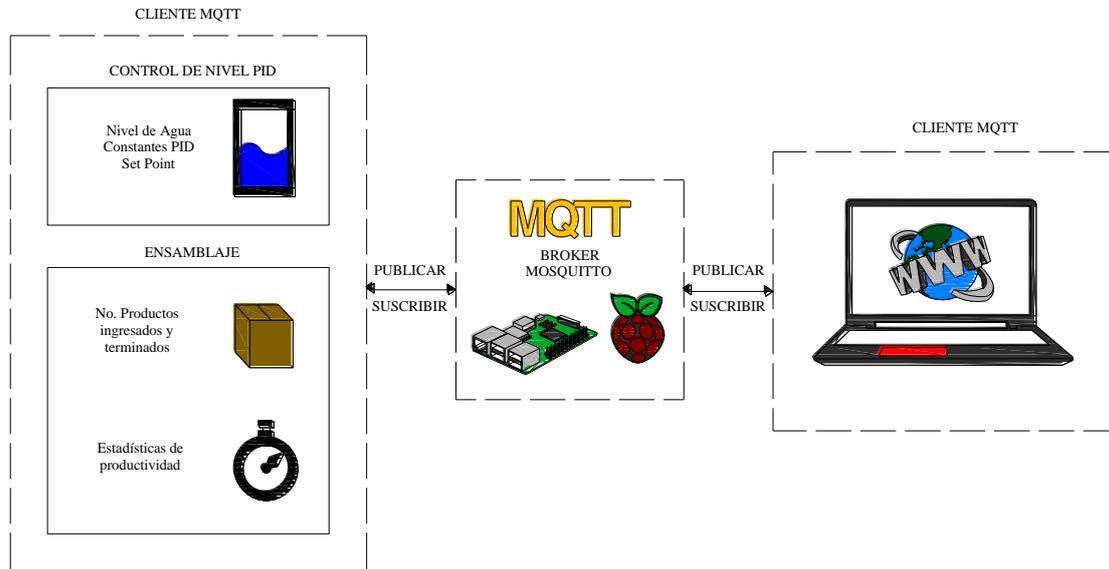


Figura 27: Topología de comunicación protocolo MQTT
Elaborado por: El Investigador

6.8.7 Topics

Ahora es necesario establecer la cadena de caracteres necesaria para que el Broker pueda operar y filtrar adecuadamente cada mensaje recibido. Cada nivel de Topic debe estar separado por una barra diagonal. Los Topics establecidos para cada una de las variables en el Controlador Lógico Programable (PLC) y necesarias para la visualización y control del proceso en la planta se establecen en el Anexo 4. Así, de requerir referirse al valor de la variable del PLC tipo booleana de denominación “Marca Start”, sea esta para publicar o suscribirse el Topic necesario es: PLC/PLC1/ENSAMBLE/CONTROL/START.

Cada Topic utilizado contiene al menos un caracter, tomando en cuenta que el bróker diferencia entre mayúsculas y minúsculas. Se utilizó mayoritariamente el comodín multinivel hash “#”, para suscribirnos y cubrir varios niveles subsiguientes temáticos, como: PLC/PLC1/ENSAMBALE/#

6.8.8 Comunicación, publicación y suscripción

Mediante el esquema publicación/suscripción de MQTT utilizado para el proyecto, se establece una arquitectura simple que permite desacoplar al cliente que envía un mensaje del que lo recibe, la conexión y comunicación controlada por el bróker, filtra los mensajes de los publicadores y los distribuye adecuadamente a los suscriptores. No es necesario crear/declarar el Topic deseado antes de publicarlo o suscribirse, el bróker acepta cada Topic válido sin ninguna inicialización previa.

Comando ejecutado desde el cliente para enviar un mensaje, publicación compuesta de un Topic y el mensaje a enviar.

```
mqttc.publish("TOPIC", "True")
```

El comando ejecutado desde el cliente para suscribirse a un Topic de interés, se compone del tema y un nivel de Calidad del Servicio (QoS), así:

```
mqttc.subscribe("TOPIC", 0)
```

6.8.9 Calidad del Servicio (QoS)

La Calidad del Servicio (QoS) utilizado en el proyecto para definir la garantía de recepción de un mensaje específico, fue siempre de 0, debido a las características de la red, como su nivel de confiabilidad y la lógica de la aplicación implementada. Además, la red propone una comunicación confiable y conexión mayoritariamente estable.

Con un nivel cero "0" de QoS, el servicio del protocolo garantiza una entrega de mejor esfuerzo, sin ninguna garantía de entrega, proporciona la misma garantía que el protocolo TCP subyacente.

Los comandos utilizados en la plataforma del actual proyecto para la conexión, suscripción y determinación de nivel de servicio QoS son:

```
mqttc.username_pw_set("USER_PLC1", "CONTR_PLC1")
```

```
mqttc.connect("186.33.173.57", "8883")
```

```
mqttc.subscribe("PLC/PLC1/CONTROL/#", 0)
```

6.8.10 Seguridad y Puerto

La plataforma realizada mediante programación y configuración permite al bróker MQTT enviar y recibir un usuario y contraseña para la autenticación y autorización del cliente.

El puerto de comunicación estándar en el protocolo MQTT es 1883, el mismo concede una comunicación convencional sin ningún tipo de seguridad excepto la mencionada para la autenticación y autorización de clientes, la operación en este puerto no permite una comunicación cifrada. El puerto utilizado en el proyecto es el 8883, el mismo permite un mayor nivel de seguridad nativo de MQTT en el envío/recepción de sus mensajes, sin opción de configuración alguna. El puerto 8883 estandarizado específicamente para conexiones MQTT seguras, permite la comunicación cifrada de mensajes tanto al publicar como al suscribirse a un Topic específico, mediante SSL (Seguridad de la capa de transporte) y TLS (Capa de sockets seguros). SSL y TLS son protocolos criptográficos para comunicaciones seguras, emitidos por proveedores especializados y confiables. El uso de estos protocolos en MQTT provoca costes en uso de recursos, tanto económicos por la contratación del servicio, uso de CPU y gastos generales en la red de comunicación y ancho de banda utilizada. La presente investigación no utiliza el protocolo MQTT sobre SSL/TLS pero trabaja en el puerto 8883.

La conexión del cliente Raspberry al bróker Mosquitto, su suscripción al topic general para la comunicación y la asignación del nivel de Calidad del Servicio QoS, se detalla el extracto de código a continuación:

```
mqttc.username_pw_set("USER_PLC1","CONTR_PLC1") # nombre de usuario y
contraseña

mqttc.connect("186.33.173.57", "8883") # dirección IP y puerto

mqttc.subscribe("PLC/PLC1/CONTROL/#", 0) # inicio de suscripción al
topic, con QoS nivel 0
```

La conexión de la Raspberry bróker, para la recepción/suscripción de datos a recibir desde el cliente alojado en la Página Web, se detalla el código siguiente:

```

#define ADDRESS      "tcp://192.168.0.111:8883"
#define CLIENTID     "SERVER"
#define TOPIC        "PLC/PLC1/#"
#define QOS          0
#define TIMEOUT      10000L

```

6.8.11 WebSockets

La aplicación y desarrollo de Página Web se soporta en el uso del protocolo MQTT sobre WebSockets, permitiendo en forma general enviar y recibir mensajes directamente desde y en un navegador. Además de poder operar todas las características y funciones de MQTT normalmente.

WebSockets está basado en TCP y como protocolo proporciona una comunicación bidireccional entre un navegador y un servidor web. La Figura 28, muestra la arquitectura utilizada en la comunicación del protocolo MQTT sobre WebSockets.

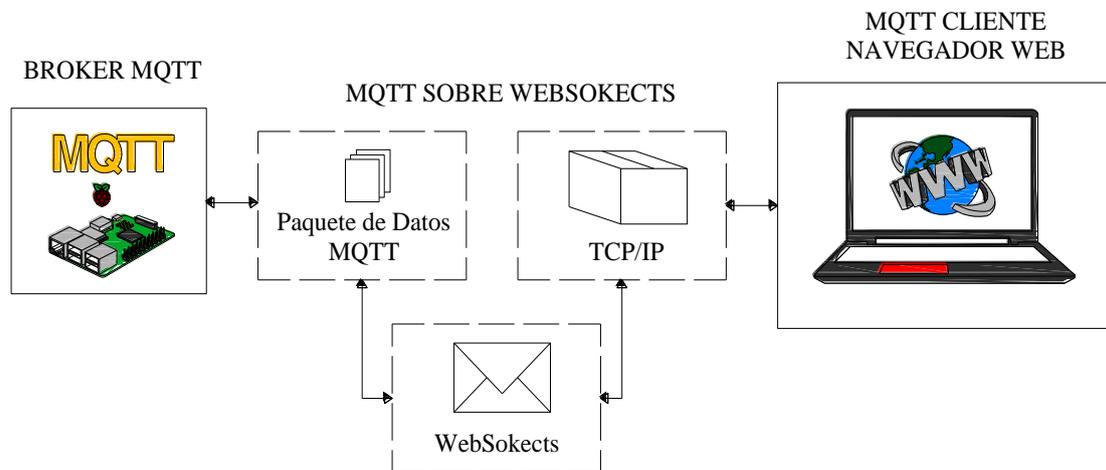


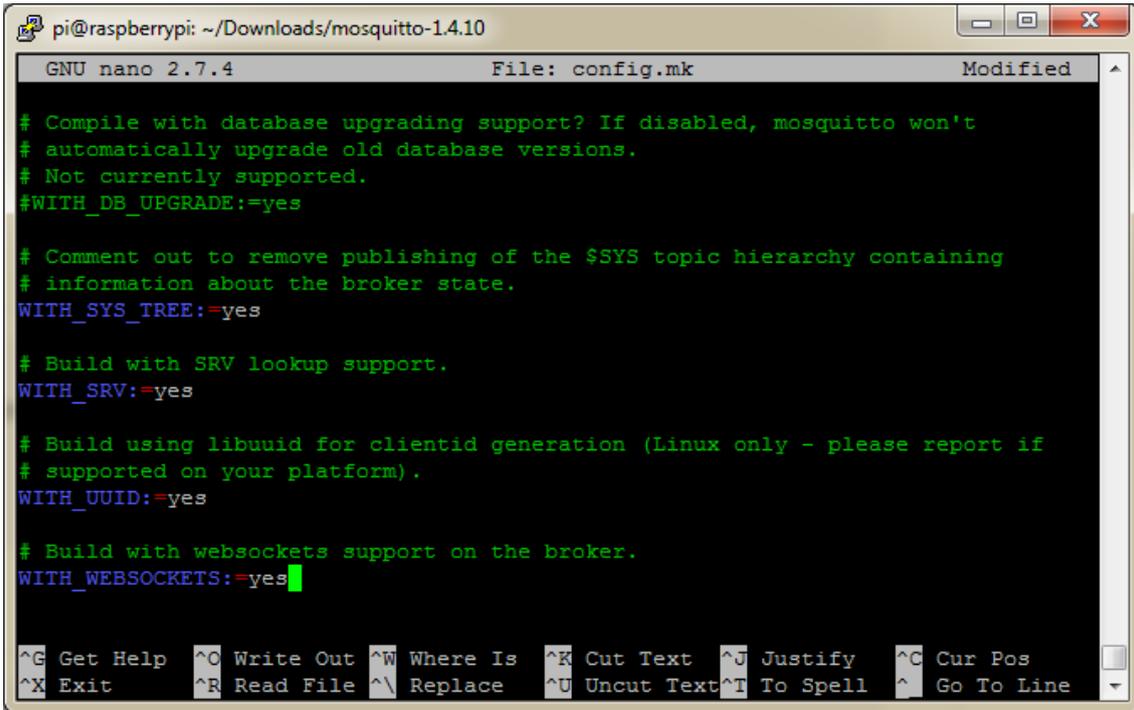
Figura 28: Arquitectura protocolo MQTT sobre WebSockets
Elaborado por: El Investigador

La configuración inicial de WebSockets en el protocolo MQTT, requiere de la activación en la instalación y configuración inicial del bróker Mosquitto. El código necesario a ejecutar en la instalación inicial es:

```
$ sudo apt-get install libwebsockets-dev
```

Es necesario editar config.mk para que la opción WebSockets este configurada en "si". Además de los realizar los requisitos previos y demás configuraciones básicas necesarias, como se muestra en la Figura 29.

WITH_WEBSOCKETS:=yes



```
pi@raspberrypi: ~/Downloads/mosquitto-1.4.10
GNU nano 2.7.4 File: config.mk Modified
# Compile with database upgrading support? If disabled, mosquitto won't
# automatically upgrade old database versions.
# Not currently supported.
#WITH_DB_UPGRADE:=yes

# Comment out to remove publishing of the $SYS topic hierarchy containing
# information about the broker state.
WITH_SYS_TREE:=yes

# Build with SRV lookup support.
WITH_SRV:=yes

# Build using libuuid for clientid generation (Linux only - please report if
# supported on your platform).
WITH_UUID:=yes

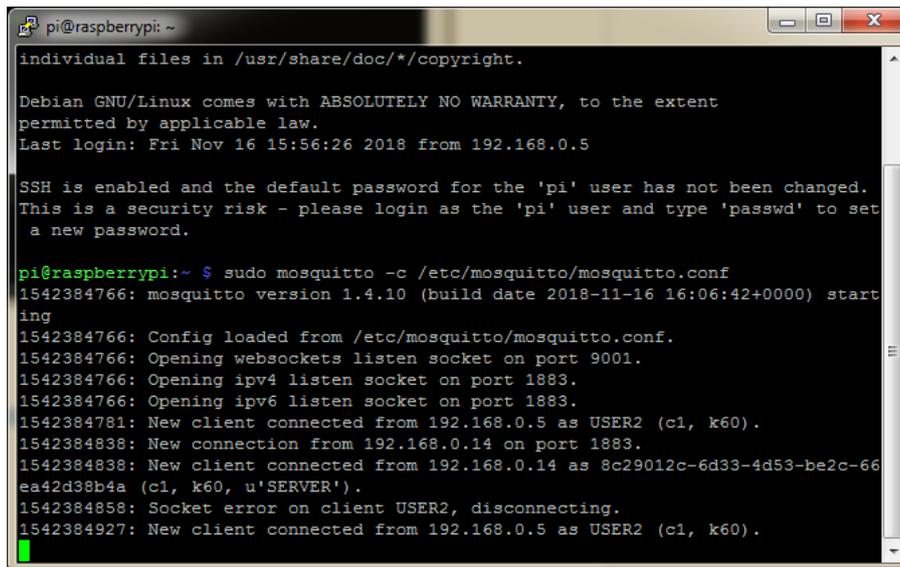
# Build with websockets support on the broker.
WITH_WEBSOCKETS:=yes

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line
```

Figura 29: Configuración WebSockets config.mk
Elaborado por: El Investigador

La configuración realizada nos permite confirmar los puertos abiertos para la comunicación por el bróker Mosquitto de MQTT, para escuchar por WebSockets el puerto determinado es el 9001, además se determinan y actualizan las conexiones, errores, desconexiones y demás avisos de clientes en el protocolo, así como su dirección IP. A continuación, se describe los avisos comunes a recibir realizada la configuración inicial y su detalle se muestra en la Figura 30:

```
Opening websockets listen socket on port 9001
Opening ipv4 listen socket on port 1883
New client connected from 192.168.0.5 as USER
New connection from 192.168.0.14 on port 1883
```



```
pi@raspberrypi ~
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Nov 16 15:56:26 2018 from 192.168.0.5

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.

pi@raspberrypi:~ $ sudo mosquitto -c /etc/mosquitto/mosquitto.conf
1542384766: mosquitto version 1.4.10 (build date 2018-11-16 16:06:42+0000) start
ing
1542384766: Config loaded from /etc/mosquitto/mosquitto.conf.
1542384766: Opening websockets listen socket on port 9001.
1542384766: Opening ipv4 listen socket on port 1883.
1542384766: Opening ipv6 listen socket on port 1883.
1542384781: New client connected from 192.168.0.5 as USER2 (c1, k60).
1542384838: New connection from 192.168.0.14 on port 1883.
1542384838: New client connected from 192.168.0.14 as 8c29012c-6d33-4d53-be2c-66
ea42d38b4a (c1, k60, u'SERVER').
1542384858: Socket error on client USER2, disconnecting.
1542384927: New client connected from 192.168.0.5 as USER2 (c1, k60).
```

Figura 30: Estado de configuración final bróker
Elaborado por: El Investigador

6.8.12 Desarrollo de Página Web

Con la configuración necesaria para la comunicación entre el protocolo MQTT sobre WebSockets, se desarrolla la Página Web adecuada para la visualización y control del proceso industrial. Mediante la instalación, configuración y comprobación del funcionamiento de programas, recursos y demás complementos necesarios para el desarrollo mencionado, se destaca la utilización de: Apache2, PHP, PHP MyAdmin, MySQL. Así como la programación y técnicas de desarrollo ejecutadas en: JavaScript, jQuery, AJAX, entre otras. Utilizando también estas técnicas, sus librerías, funciones y demás características en la creación, ubicación y visualización de gráficos, tablas y demás variables para la distribución visual y estética de la Página Web. Finalmente, la aplicación de funciones complementarias, condiciones, ciclos, bucles, tiempos y demás, así como la función utilizada para el movimiento tanto de productos en las bandas transportadoras, como en el llenado del tanque es setInterval (), utilizado mediante la técnica de desarrollo AJAX. Las Figuras 31 y 32 muestran la ejecución de funciones del botón Start y movimiento de imágenes respectivamente. El código del programa para el desarrollo de la Página Web del presente proyecto, se adjunta en el Anexo 6.

```
function start(){
    var usuario="operario";
    message = new Paho.MQTT.Message("START");
    message.destinationName = "PLC/PLC1/CONTROL/GENERAL/START_WEB";
    mqtt.send(message);
}
```

Figura 31: Función botón START página Web
Elaborado por: El Investigador

```

function frame() { // sube la segunda caja
  if (posy<= ((posicion_banda.height*4)/100)) {
    clearInterval(id);
    id_left = setInterval(frameleft, 10);
    //elem.style.visibility='visible';
  }
  else {
    posy--;
    caja2_ensamblaje.style.top = posy + 'px';
    caja2_ensamblaje.style.visibility='visible';
  }
}
}

```

Figura 32: Función movimiento caja – Página Web
Elaborado por: El Investigador

El sistema después de apuntar a la dirección IP Publica 186.33.173.57; nos permite ingresar a la página de autenticación y autorización, como se muestra en la Figura 33, donde al digitar correctamente el usuario y contraseña se puede acceder a las Páginas de Panel de proceso respectivas. Se desarrollan dos páginas web, las mismas permiten la monitorización diferenciada del proceso desde dos usuarios, operador e ingeniero.

The image shows a login page with the following content:

UNIVERSIDAD TÉCNICA DE AMBATO
 MAESTRÍA EN AUTOMATIZACIÓN Y SISTEMAS DE CONTROL

MONITOREO Y CONTROL

Ingresa sus Datos

Usuario

Clave

Figura 33: Página de ingreso a Panel de Control
Elaborado por: El Investigador

La página del operador e ingeniero poseen acceso a los dos procesos existentes: ensamblaje y control de tanque. Sin embargo, difieren en su accesibilidad por restricciones para el control de estadísticas de producción en el ensamblaje y constantes de control proporcional, integral y derivativo PID del Tanque; gráficas de producción o Layout con movimiento del proceso de ensamblaje. La página también muestra el usuario actual y el botón para el normal cierre de sesión. El detalle de acceso para la monitorización y control del proceso desde la página de operador e ingeniero se muestran en las Tablas 11 y 12 respectivamente.

El proceso de desarrollo y posterior operación del Interfaz Hombre Máquina (HMI), precisa analizar y cumplir las correspondientes normas de estandarización en torno a nuestra Página Web. La normativa aplicada a la monitorización del proceso industrial es, ISA-SP101 y UNE-EN ISO 9241 conforme las características y detalles mostrados en la Tabla 10. En síntesis, se presenta información estrictamente necesaria, comprensible y clara, tanto en tareas y procesos, buscando facilitar la operación intuitiva del proceso indiferentemente del nivel de capacitación del usuario. Las Figuras 34 y 35 detallan la configuración visual y funcional de la Página Web del operador e ingeniero, acorde la normativa correspondiente.

Tabla 10: Características de norma ISA-SP101 y UNE-EN ISO 9241
Elaborado por: El Investigador

Característica	Descripción	Detalle	
Diseño	Simple	Colores de contraste bajo (equilibrado)	
Orden	Lógico	Alineación relativa de objetos	
Simulación	Básica	Productos y nivel del tanque	
Texto	Claro y sencillo	Tres tamaños de letra, mínimo uso de mayúsculas	
Mandos	Botones	Agrupados por procesos (sin simulación)	
Polaridad	Positiva	Equilibrio de luminancia	
Colores y formas	Estándar	Fondo	Gris claro
		Texto	Gris oscuro
		Botones	Encendido (verde); Paro (rojo); Reset (amarillo); Paro Emergencia (rojo, amarillo y negro)
		Información	Azul, negro
		Elementos	Metal (gris oscuro)
		Productos	Azul claro
		Fluido tanque	Azul

Características adicionales:

El depósito posee indicación gráfica del nivel de llenado

Productos en bandas transportadoras con simulación de movimiento

Como resultado de lo establecido en norma y de conformidad con ésta, la Página Web para la monitorización del proceso industrial posee características minimalistas, con un básico desarrollo de simulación y realismo, por tanto, es un interfaz HMI práctico que permite la identificación, deducción y control rápido del proceso.

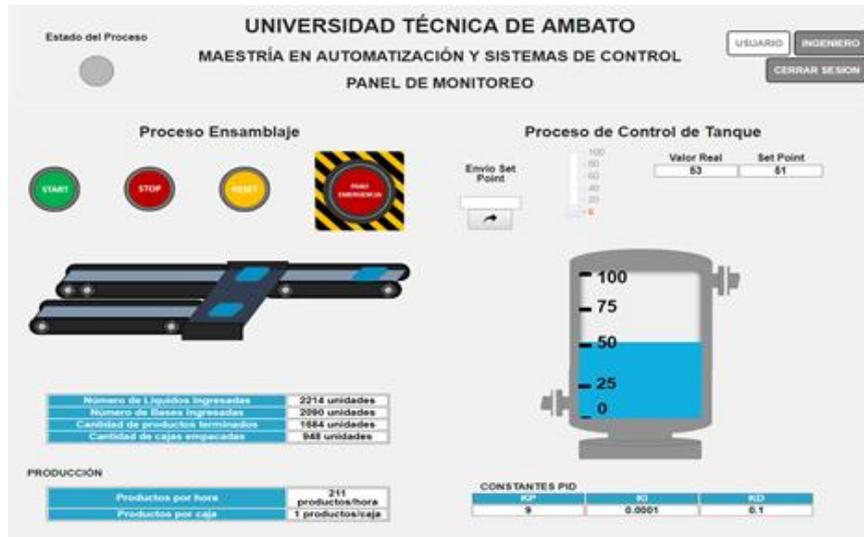


Figura 34: Panel de control operador
Elaborado por: El Investigador

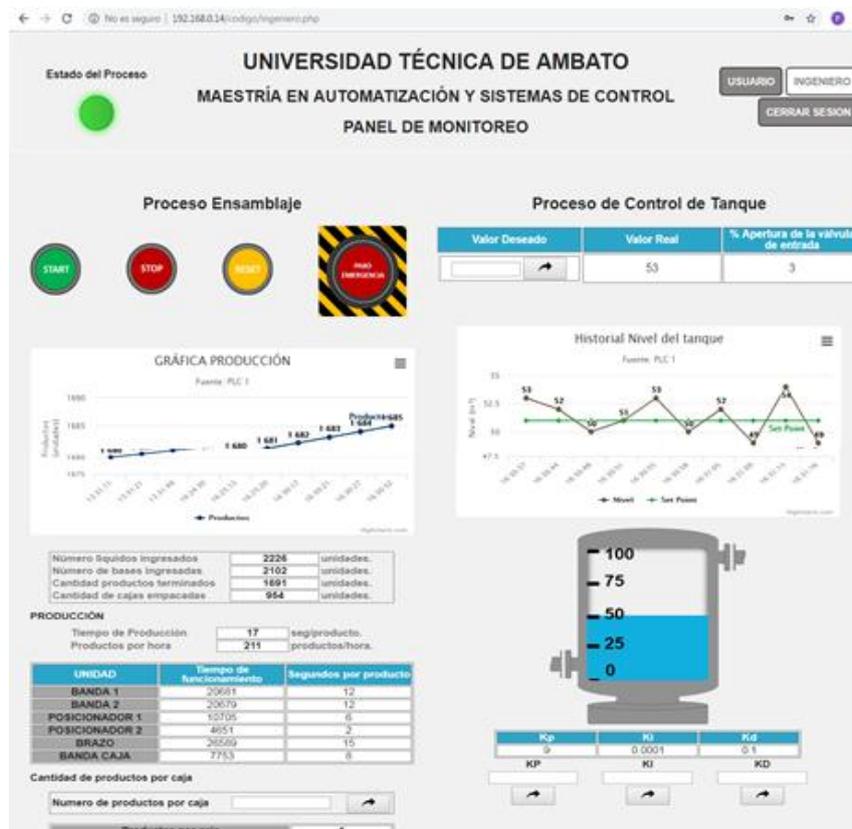


Figura 35: Panel de control Ingeniero
Elaborado por: El Investigador

Tabla 11: Detalle acceso y restricción Panel operador

Elaborado por: El Investigador

Proceso	Layout/Gráfico	Panel de control	Datos del proceso	Información de producción/Datos Adicionales
Ensamblaje	Layout gráfico de bandas transportadoras, movimiento del producto desde su ingreso, ensamble y salida en tiempo real	Botón de Marcha (START)	No. De Lids ingresados	Productos por hora (productos/hora)
		Botón de Paro (STOP)	No. De Bases ingresadas	
		Botón de paro de emergencia (PARO EMERGENCIA)	Cantidad de productos terminados	Visualización de productos por caja (productos/caja)
		Indicador visual del estado del proceso (Encendido/Apagado)	Cantidad de cajas empacadas	
Control Tanque	Layout gráfico de tanque, visualización en tiempo real de llenado/vaciado del tanque	Ingreso numérico SETPOINT	Valor real del tanque en números	Visualización de las constantes PID para el control del tanque
		Slide para control de SETPOINT	Valor del setpoint en números	

Tabla 12: Detalle acceso y restricción Panel ingeniero

Elaborado por: El Investigador

Proceso	Layout/Gráfico	Panel de control	Datos del proceso	Información de producción/Datos Adicionales
Ensamblaje	Gráfica estadística de producción (productos/tiempo)	Botón de Marcha (START)	No. De Lids ingresados	Productos por hora (productos/hora)
		Botón de Paro (STOP)	No. De Bases ingresadas	Ingreso manual y visualización de productos por caja (productos/caja)
Control Tanque	Layout gráfico de tanque, visualización en tiempo real de llenado/vaciado del tanque	Botón de Reseteo (RESET)	Cantidad de productos terminados	Tiempos de funcionamiento y tiempos por producto de los actuadores, como: bandas, posicionadores y brazo.
		Botón de paro de emergencia (PARO EMERGENCIA)	Cantidad de cajas empacadas	
		Indicador visual del estado del proceso (Encendido/Apagado)	Valor real del tanque en números	Ingreso manual y visualización de las constantes PID para el control del tanque
		Ingreso numérico SETPOINT	Porcentaje de apertura de válvula de entrada al tanque	

6.8.13 Análisis de Tráfico de Datos

Finalmente, de manera paralela a las pruebas de funcionamiento de la plataforma, su robustez y respuesta ante posibles problemas, es necesario comprobar la funcionalidad real de la propuesta implementada, a través del análisis de tráfico de red del protocolo MQTT mediante la herramienta “Wireshark”, ya que su funcionalidad nos permite capturar el tráfico de datos existente mediante Wi-Fi y estudiar la comunicación implementada. La estructura del análisis de datos efectuado se muestra en la Figura 36.

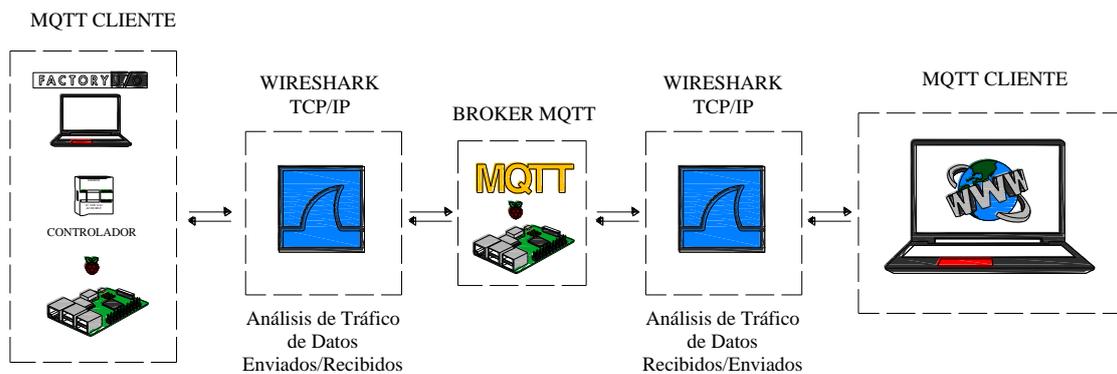


Figura 36: Estructura del análisis de tráfico de datos
Elaborado por: El Investigador

Mediante la ejecución de Wireshark y una vez levantado el bróker de comunicación Mosquitto junto con los respectivos clientes, se obtiene el tráfico de red de la comunicación implementada, conociendo que MQTT trabaja sobre WebSockets y ambos sobre TCP, se realiza el análisis de este último protocolo. La Figura 37 muestra la captura de tráfico una vez iniciado el bróker y enlazada la comunicación con los clientes.

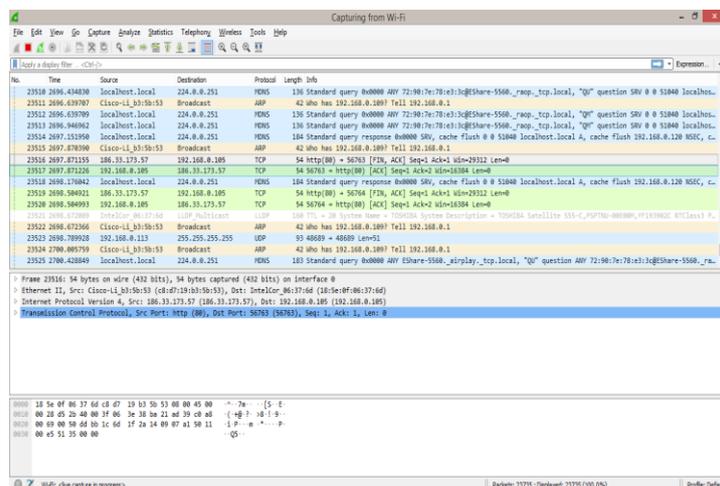


Figura 37: Visualización tráfico de datos al iniciar bróker
Elaborado por: El Investigador

Realizada la autenticación de credenciales en la Página Web del operario, e iniciado y puesto en marcha el proceso industrial, se observan las primeras comunicaciones por TCP y HTTP entre las direcciones IP específicas de los dispositivos utilizados en la plataforma desarrollada. Al igual se detalla la comunicación por el protocolo S7COMN utilizando funciones para escritura y lectura entre el PLC S71200 utilizado y el ordenador donde se ejecuta la Página Web final. En las Figuras 38 y 39, se observa el detalle de las interacciones entre la dirección IP específica del controlador y ordenador de monitorización, así también actualizaciones del tráfico de red en Wireshark e información adicional del protocolo y dispositivos de red.

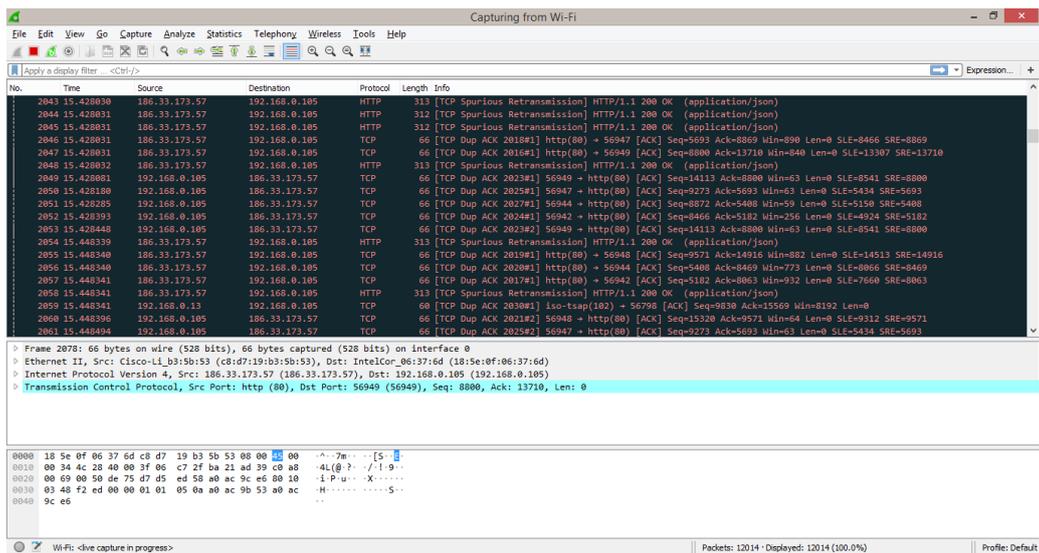


Figura 38: Visualización tráfico de datos Wireshark con el proceso en marcha
Elaborado por: El Investigador

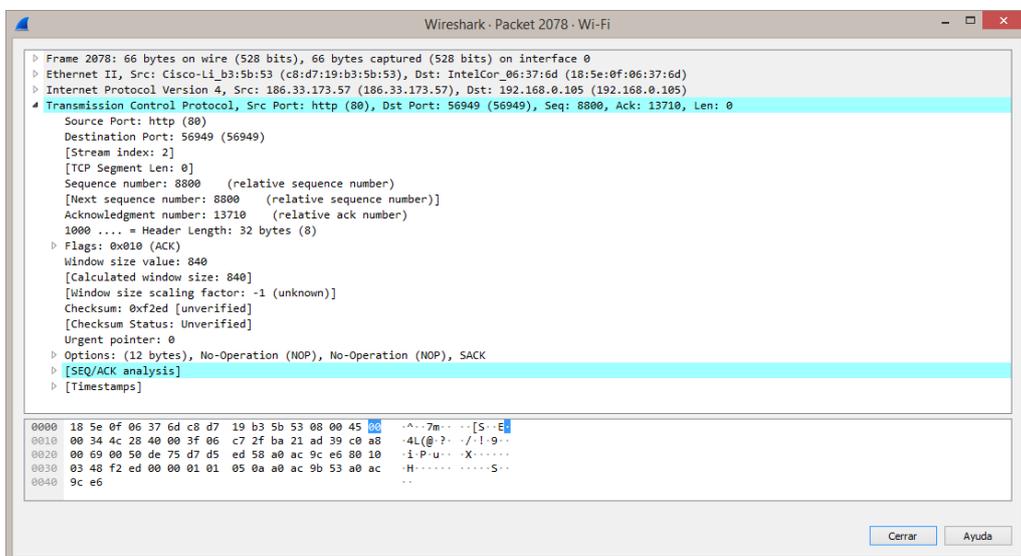


Figura 39: Interacciones de red en Wireshark, detalle de paquete 2078 Wi-Fi
Elaborado por: El Investigador

La Gráfica de Uso de Ancho de Banda, explícita en la Figura 40, mediante la herramienta Graph IO en Wireshark, puntualiza el análisis de comunicaciones y tráfico de datos de la plataforma desarrollada.

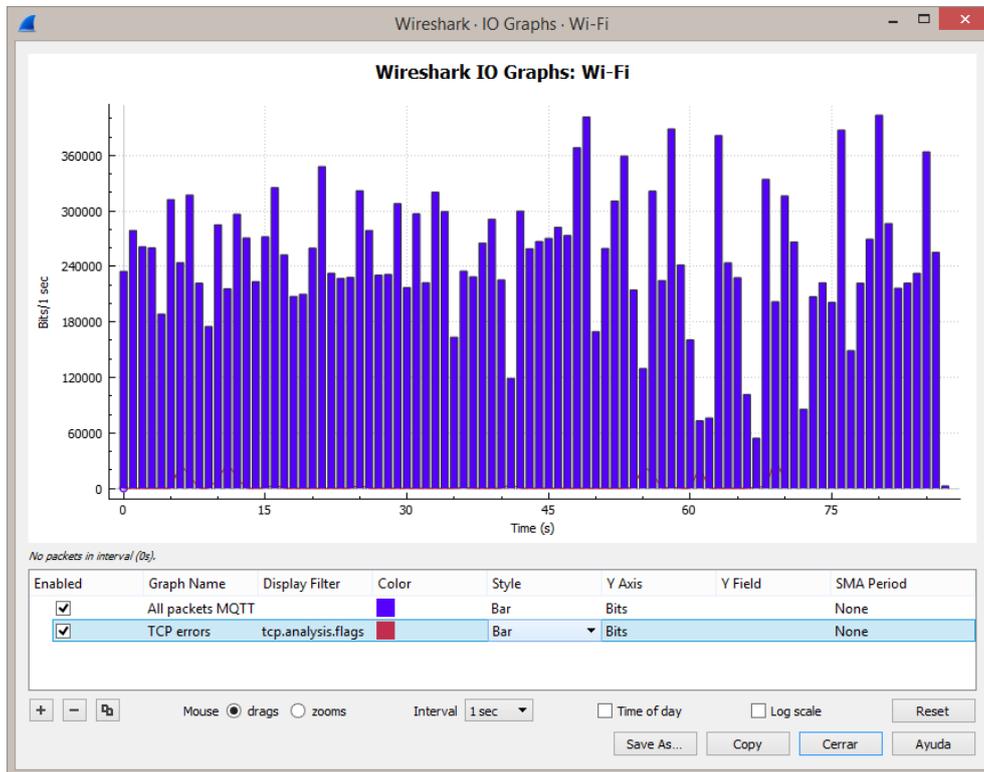


Figura 40: Ancho de Banda (bits/seg) del protocolo de comunicación
Elaborado por: El Investigador

La Figura 40, define el comportamiento de tráfico general de la comunicación efectuada por el Protocolo MQTT, especificando el consumo de ancho de banda utilizado. Se determina entonces que, en el intervalo de tiempo de ochenta y cinco segundos, a partir del funcionamiento de la plataforma, su ancho de banda alcanza un valor medio de 225000 bits por segundo (bits/seg). Por tanto, la interacción de comunicación publicación/suscripción entre cliente/brokers/cliente consume 0,03 MB/s (Megabytes por segundo).

En cuanto al análisis de paquetes del sistema y posibles errores en su entrega-recepción, la Figura 41 detalla la cantidad de paquetes por segundo tramitados en una transacción publicación/suscripción, y presenta un promedio de 221 paquetes por segundo (paquetes/seg). Es necesario insistir que, aunque la plataforma se ejecuta y opera en un ambiente relativamente controlado, se obtiene datos de paquetes perdidos, no útiles o con errores en su entrega/recepción, situación que puede imputarse a la operación del

sistema sobre una pila TCP/IP y el mayoritario nivel “0” utilizado para definir la Calidad de Servicio (QoS) del protocolo MQTT.

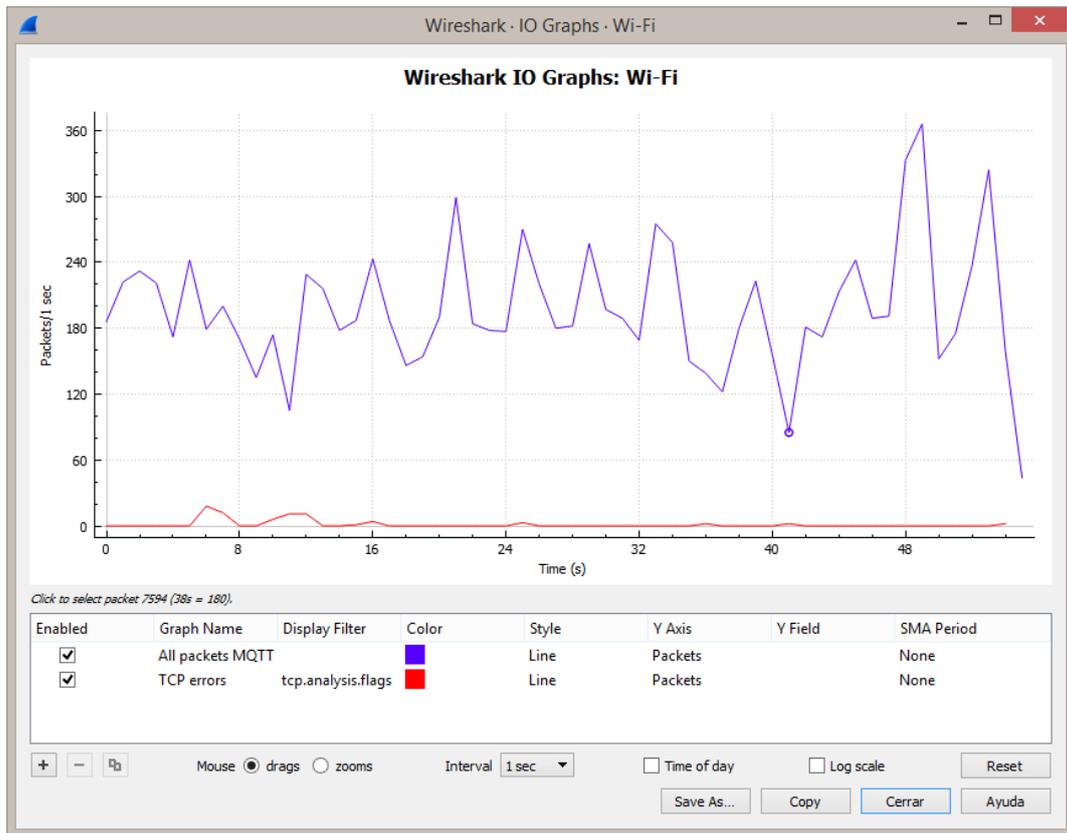


Figura 41: Detalle de paquetes en transacción publicación/suscripción
Elaborado por: El Investigador

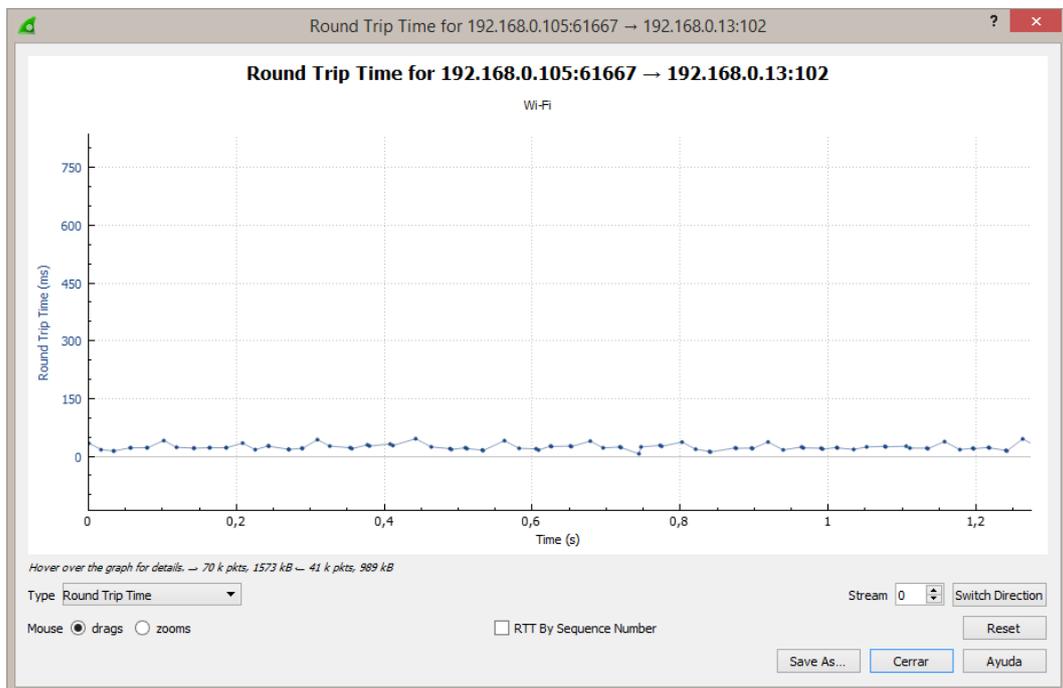


Figura 42: Tiempo de procesamiento paquetes
Elaborado por: El Investigador

La gráfica descrita en la Figura 42, expone el tiempo de captura de un paquete TCP, su reconocimiento y posterior procesamiento, el mismo opera entre los cien y seiscientos milisegundos (100ms – 600ms), valor intrínseco al proceso, su operación y funcionamiento, que produce un posible retardo en la comunicación y pérdida exigua de paquetes. Esta congestión en la red, ratifica el análisis y resultado detallado en la Figura 36, respecto al procesamiento de paquetes de red en la comunicación.

Mediante la aplicación y análisis de filtros de captura necesarios en la herramienta Wireshark e interpretación de gráficos respecto al ancho banda, procesamiento de paquetes y sus tiempos. Se instaura la comparación de paquetes entregados y recibidos por el protocolo de comunicación MQTT sobre TCP y se presenta en la Tabla 13, el resumen de datos e información depurada y tabulada para el respectivo análisis cuantitativo de tráfico de Red.

Tabla 13: Detalle cuantitativo análisis de tráfico de Red
Elaborado por: El Investigador

No.	Origen	Destino	PE(se)	PR(ce)	PP(nu)
1	192.168.0.13	192.168.0.105	249	246	3
2	192.168.0.105	192.168.0.13	159	158	2
3	192.168.0.111	192.168.0.13	178	171	7
4	192.168.0.13	192.168.0.111	159	151	8
5	192.168.0.111	192.168.0.105	124	121	3
6	192.168.0.105	192.168.0.111	286	284	2
7	192.168.0.105	186.33.173.57	213	209	4

PE(se): Paquetes entregados, sin errores

PR(ce): Paquetes recibidos, o con errores

PP(nu): Paquetes Perdidos, no útiles

6.8.14 Tiempos de Respuesta y Carga del Servidor (SBC)

El análisis del intervalo de tiempo de transacción solicitud y respuesta entre cliente/servidor/cliente, permite poner a prueba el protocolo de comunicación MQTT como un sistema de control y monitorización en tiempo real. Primero determinamos las características que influyen directamente en el análisis de tiempos y respuestas, como el

tiempo de ciclo en el código de programación, el ambiente y escena de simulación relativamente controlados para ejecutar las pruebas y el ancho de banda conferido por el proveedor de internet local. Por tanto, mediante el software de análisis de tráfico de datos Wireshark y la correspondiente captura de paquetes enviados/recibidos, se determina el tiempo de respuesta por solicitud realizada, como se muestra en la Figura 43 en el campo [Time since request] parámetro que indica el tiempo desde la solicitud por TCP requerida.

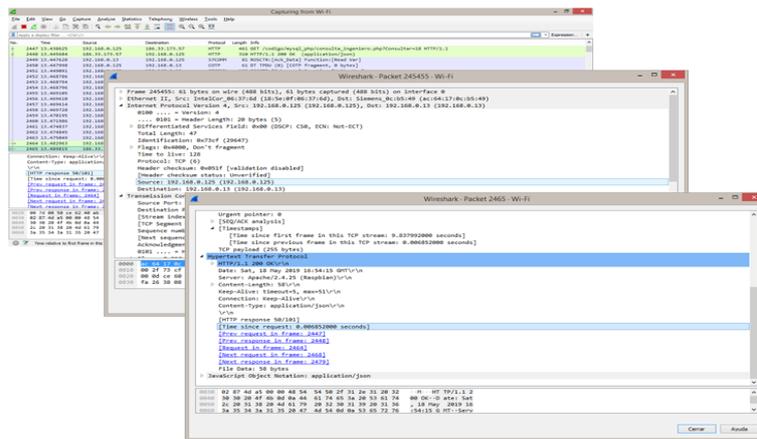


Figura 43: Tiempo desde solicitud, Wireshark
Elaborado por: El investigador

Se establece entonces, filtros de tiempo necesarios para las solicitudes de envío/recepción de información, el detalle de resultados obtenidos se muestra en la Tabla 14.

Tabla 14: Tiempos de respuesta de la comunicación
Elaborado por: El investigador

Tipo de Variable	Descripción	Dirección PLC	Nivel QoS	Tiempo de respuesta	
				Wireshark	Cronometrado
Bool	Inicio	%M0.0	0	155 ms	< 1 seg
Bool	Paro Emergencia	%M3.2	1	267 ms	< 1 seg
Int	SetPoint	%MW28	0	195 ms	< 1 seg
Int	Produccion	%MW32	1	478 ms	< 1 seg
Word	Kp	%MD82	0	712 ms	≥ 1 seg
Word	Td	%MD90	1	932 ms	≥ 1 seg

ms: milisegundos

seg: segundos

Kp: Constante PID ganancia; Td: Constante PID tiempo derivativo

De la información detallada en la Tabla 12 se determina que, un evento solicitud/respuesta presenta un tiempo promedio de 155 milisegundos con el nivel más bajo de QoS, nivel utilizado mayormente en transacciones publicación/suscripción de la plataforma desarrollada. Evidenciándose la característica de control, comunicación y monitorización en tiempo real del protocolo MQTT.

Debido a medidas de seguridad, los botones de Paro (Stop) y Emergencia (Emergency Stop) poseen un nivel de calidad de servicio en su publicación y suscripción de 1. Adicionalmente las pruebas determinaron un tiempo de respuesta de entre uno y un segundo y medio, en los eventos de solicitud/respuesta con un QoS de nivel 2, propiedad que obliga un mayor intercambio de mensajes, por tanto, mayor consumo de ancho de banda, aumento de carga al sistema y por consiguiente eleva los tiempos de respuesta.

El procesamiento y capacidad de operación del cliente y servidor alojados en las Computadoras de Placa Única (SBC, Raspberry Pi), representan ahora el limitante para alcanzar las características de escalabilidad propias del protocolo MQTT. Las Tablas 15 y 16 presentan el detalle de información de las SBC Cliente y Broker respectivamente.

Tabla 15: Información prueba de funcionamiento Raspberry Cliente
Elaborado por: El investigador

Computador de Placa Única (SBC)			
SBC Raspberry Pi 3 modelo B			
	Lenguaje:	Python	Protocolo: MQTT
	Servicio:	Cliente/Eclipse Paho Comunicación	
		PLC Siemens S7 1200	Variables
		Snap 7 Python, WiFi	45 (23Bool/14Int/8Word)
Prueba de Funcionamiento			
Fecha/hora inicio:	18/05/2019 – 10H00	Fecha/hora fin:	19/05/2019 – 16H00
Horas de funcionamiento	30 horas	Control	Frecuente
Memoria RAM	Total: 927 Mb	Usada: 117 Mb	Uso de RAM
Memoria Swap	Total: 99 Mb	Usada: 0 Mb	13%
Temperatura	Inicio: 32,8 °C	Estabilización: 33,5 °C	

Tabla 16: Información prueba de funcionamiento Raspberry Servidor
Elaborado por: El investigador

Computador de Placa Única (SBC)			
SBC Raspberry Pi 3 modelo B			
	Lenguaje:	C	Protocolo: MQTT
	Servicio:	Broker/Eclipse Mosquitto Comunicación	
		Cliente Raspberry	Variables
		Cliente WebSockets	60 (28Bool/22Int/10Word)
Prueba de Funcionamiento			
Fecha/hora inicio:	18/05/2019 – 10H00	Fecha/hora fin:	19/05/2019 – 16H00
Horas de funcionamiento	30 horas	Control	Frecuente
Memoria RAM	Total: 927 Mb	Usada: 146 Mb	Uso de RAM 17%
Memoria Swap	Total: 99 Mb	Usada: 0 Mb	
Temperatura	Inicio: 34,9 °C	Estabilización: 35,2 °C	

La prueba de funcionamiento efectuada permite determinar un consumo de memoria RAM no superior al 20% de su capacidad, la memoria de intercambio no es utilizada, no se comprueba un aumento importante en la temperatura de los equipos una vez que ésta se estabiliza y no existe alteración alguna en el normal funcionamiento del proceso, su control o monitorización. Por consiguiente y conforme la carga aplicada al sistema, sus características permiten nominalmente escalar y operar la plataforma de comunicación mediante el protocolo MQTT, con cinco clientes y alrededor de 300 variables de características similares a los establecidos en las pruebas realizadas, teniendo en cuenta en estas últimas el ambiente controlado y estable utilizado.

6.8.15 Comprobación de la Hipótesis de la propuesta

Al conocer los resultados del análisis de tráfico de red de la plataforma desarrollada, se efectúa la correspondiente comprobación estadística de la Hipótesis planteada para la presente investigación: La Industria 4.0 incide en la monitorización adecuada de procesos industriales

Particularmente este caso se puede comprobar de forma práctica y efectiva con el detalle de paquetes de información transferida por el protocolo específico utilizado, todo esto mediante la prueba de chi-cuadrado y la aplicación de sus fórmulas detalladas a continuación:

Cálculo de Chi-cuadrado, Ecuación 2.

Ecuación 2: Chi-cuadrado

$$\Lambda^2 = \sum \frac{(fo-fe)^2}{fe} \quad \text{Ec 2.}$$

Cálculo de frecuencias esperadas, Ecuación 3.

Ecuación 3: Frecuencias esperadas

$$fe = \frac{\sum \text{fila} * \sum \text{columna}}{\sum \text{Total}} \quad \text{Ec 3.}$$

Cálculo de grados de libertad, Ecuación 4.

Ecuación 4: Grados de libertad

$$v = (\text{cant. filas} - 1) * (\text{cant. columnas} - 1) \quad \text{Ec 4.}$$

Cálculo del parámetro p , Ecuación 5.

Ecuación 5: Parámetro p

$$p = 1 - (\text{nivel de significancia}) \quad \text{Ec 5.}$$

La Tabla 17, se muestran el detalle de frecuencias observadas mediante el análisis de tráfico de red.

Tabla 17: Detalle frecuencias observadas del sistema
Elaborado por: El Investigador

No. Transferencias	Paquetes entregados, sin errores	Paquetes Perdidos, no útiles	Suma filas
1	249	3	252
2	159	2	161
3	178	7	185
4	159	8	167
5	124	3	127
6	286	2	288
7	213	4	217
Suma columnas	1368	29	1397

Aplicando la Ecuación 3, la Tabla 18 especifica el cálculo de frecuencias esperadas.

Con los cálculos realizados y comprobados de frecuencias, y siguiendo fielmente el adecuado proceso del método utilizado, se calcula con la Ecuación 2 el valor de chi-cuadrado. Paralelamente también se determina el cálculo de grados de libertad aplicando la Ecuación 4.

$$\text{Chi-cuadrado} \quad \lambda^2 = 13,04$$

$$\text{Grados de libertad} \quad v = 6$$

Tabla 18: Detalle frecuencias esperadas del sistema
Elaborado por: El Investigador

No. Transferencias	fe PE(se)	fe PP(nu)	Suma filas
1	246,8	5,2	252
2	157,7	3,3	161
3	181,2	3,8	185
4	163,5	3,5	167
5	124,4	2,6	127
6	282	6	288
7	212,5	4,5	217
Suma columnas	1368	29	1397

El nivel de significancia se determina considerando el error al rechazar la hipótesis nula siendo verdadera; para el actual análisis el valor es de 95% es decir igual a 0.95, refiriendo que hay una probabilidad del 0,5% de que la hipótesis nula sea verdadera. El cálculo complementario del parámetro p se efectúa mediante la Ecuación 5.

$$\text{Parámetro } p \quad p = 0,95$$

Verificando los valores necesarios de los parámetros p y v en la Tabla de Valores Críticos de chi-cuadrado, el resultado es de 12,592.

$$\lambda^2 \text{ calculado (13,04)} \leq \lambda^2 \text{ crítico (12,592)}$$

No cumple

Para el actual caso, el valor de chi-cuadrado es mayor que su valor crítico, incumpléndose la comparación expuesta. Por tanto, se acepta la hipótesis de trabajo, es decir, la industria 4,0 incide en la monitorización de procesos industriales, y se rechaza la hipótesis nula.

6.9 Conclusiones y Recomendaciones

6.9.1 Conclusiones

- La plataforma implementada basada en el concepto de Industria 4.0, permite la integración completa de las unidades del proceso desarrollado, un acercamiento evidente, aplicable e importante a la transformación industrial tan necesaria en nuestro país. Evolución alcanzada en base al conocimiento técnico poseído y utilizando dispositivos de código abierto, adaptables, conocidos y sobre todo accesibles al usuario.
- MQTT es un protocolo de mensajería relativamente fácil de aplicar, liviano y abierto, que permite la comunicación en entornos restringidos. Es escalable, eficiente respecto al ancho de banda, posibilita niveles de calidad de servicio y permite el cifrado de mensajes.
- La plataforma opera bajo la autenticación y autorización de un usuario y contraseña para cada cliente. El puerto MQTT utilizado es el 8883, configuración establecida para permitir un nivel de seguridad nativo del protocolo para el envío y recepción de mensajes. La aplicación no utiliza cifrado de mensajes ni credenciales bajo SSL y TLS.
- Las características del protocolo MQTT, especialmente su velocidad y aplicabilidad en entornos restringidos, se deben a que realiza un desacoplamiento entre publicadores y suscriptores, es asíncrono, el envío/recepción de mensajes se realiza en tiempo real y utiliza un filtrado de mensajes basado en el sujeto (topic).

- La plataforma desarrollada está en línea, consta de dos clientes, el proceso es controlado mediante un PLC, simulado en el software Factory I/O y su comunicación con el cliente navegador es a través de WebSockets por el puerto 9001. Después de efectuadas las pruebas de funcionamiento, respuesta y análisis de tráfico de red, la propuesta presenta características apropiadas de funcionalidad, robustez, escalabilidad y seguridad.
- El nivel de Calidad del Servicio (QoS) acorde a la confiabilidad de la red establecida y la lógica de la aplicación desarrollada es “0”. Sin embargo, se realizó pruebas con QoS de niveles “1” y “2”, sin que exista ningún problema de comunicación u requerimiento adicional en código o implementación de desarrollo. El bróker realiza la administración y retransmisión de mensajes junto con la garantía de entrega.

6.9.2 Recomendaciones

- Efectuar la aplicación y comparación de MQTT, actual protocolo desarrollado en la propuesta, con sus principales competidores en comunicación, Industria 4.0 e Internet de las Cosas (IoT), estableciendo ventajas, desventajas, características y necesidades de cada protocolo, además de los conocimientos específicos necesarios para su adecuado desarrollo y eficiente funcionamiento.
- Indagar e implementar una investigación mayor y complementaria que permita analizar el tráfico de red, datos y uso de recursos con la aplicación del protocolo MQTT sobre SSL/TLS, además de analizar y estudiar el detalle de las características de seguridad correspondientes.
- Procurar no usar caracteres poco comunes como Topics, tal como: barras diagonales, espacios en blanco, entre otros. Aunque el protocolo permita asignar tales caracteres, los mismos no proporcionan ningún beneficio y pueden provocar confusión, sobre todo al momento de leer, depurar y corregir Topics y en general al código del programa.
- La elección y determinación del nivel de Calidad del Servicio (QoS) deberá realizarse según requerimiento específico de la aplicación, estado y

características de la comunicación, se recomienda: el uso de QoS “0”, con una comunicación estable y si no es relevante la pérdida de mensajes, ya sea por su importancia o intervalo de tiempo de envío. QoS “1” cuando es necesario recibir todos los mensajes, y puede manejar o procesar múltiples entregas. Utilice un nivel de QoS “2” cuando es crítico para el proceso u aplicación recibir todos los mensajes una sola vez, la interacción será más lenta y el consumo de recursos el máximo.

- Complementar el desarrollo de la plataforma implementando un segundo cliente que realice el control del nivel de tanque en el proceso industrial, mediante el regulador difuso desarrollado. Realizar pruebas de funcionamiento y respuesta conforme la integración de información efectuada.

BIBLIOGRAFÍA

- Alcure, G. (2013). Control de procesos y caracterización de situaciones industriales por métodos globalizantes. *ALCTRA Recherche & Développement*, 43-58.
- Aldakin. (2017). *Industria 4.0*. Obtenido de <http://www.aldakin.com/industria-4-0-que-es-ventajas-e-inconvenientes/>
- Apachecon . (2015). *Apache*. Obtenido de Http Server Project : http://httpd.apache.org/ABOUT_APACHE.html
- Arbildo, A. (2011). El control de procesos industriales y su influencia en el mantenimiento. *Ingeniería Industrial*, 35-49.
- Atiga. (2017). Estado del Arte IoT y CPS. *Oportunidades Industria 4.0 en Galicia*, 1-99.
- Babovic, Z., & Milutinovic, V. (2016). Web Performance Evaluation for Internet of Things Applications. *IEEEAccess*, 6974-6992. Obtenido de <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7586113>
- Bonilla, I., Tavison, A., Morales, M., Guajardo, L., & Laines, C. (2016). IOT, El Internet de las Cosas y la Innovación de sus Aplicaciones . *Vincula Téctica*, 2313-2340.
- Cámara Valencia. (Octubre de 2016). *Sistemas Ciberfísicos: la respuesta a las necesidades de la sociedad y la industria*. Obtenido de <https://ticnegocios.camaravalencia.com/servicios/tendencias/sistemas-ciberfisicos-la-respuesta-a-las-necesidades-de-la-sociedad-y-la-industria/>
- Campero Núñez, J. (2016). ¿La cuarta revolución industrial en Bolivia? *Análisis*, 1-24.
- Cárdenas, A. (28 de Noviembre de 2016). *Soluciones IoT*. Obtenido de Secmoti: <https://secmoti.com/plataforma-iot/>
- Dote, Y. (1995). Introduction to fuzzy logic. *Proceedings of IECON '95 - 21st Annual Conference on IEEE Industrial Electronics*, 1-4.
- Dow, C.-R., Cheng, S., & Hwang, S.-F. (2016). A MQTT-based Guide and Notification Service System. *IEEE 7th Annual Information Technology, Electronics and*

- Mobile Communication Conference (IEMCON)*, 1.4. Obtenido de <https://scihub.tw/https://ieeexplore.ieee.org/document/7746240>
- Eclipse. (10 de 02 de 2016). *Eclipse Mosquitto*. Obtenido de <https://mosquitto.org/>
- Escudero, A. (2018). Redefinición del “aprendizaje en red” en la cuarta revolución industrial. *apertura*, 149-163.
- Esmerio, J. (2018). Industria 4.0 y la calidad de la información. *Soldadura y Inspección*, 1-10.
- Félix, C., & Blesa, A. (2015). Industria 4.0 en el Grado de Ingeniería Electrónica y Automática. *Universitaria de la Informática*, 327-332.
- Fernandes, J., Lopez, I., Rodrigues, J., & Ullah, S. (2013). Performance Evaluation of RESTful Web Services and AMQP Protocol. *Fifth International Conference on Ubiquitous and Future Networks (ICUFN)*, 810-815. Obtenido de <https://scihub.tw/https://ieeexplore.ieee.org/document/6614932>
- García, M., Irisarri, E., & Pérez, F. (2017). *Integración Vertical en plantas industriales utilizando OPC UA e IEC-61499. Enfoque UTE*, 287-299. Obtenido de <http://scielo.senescyt.gob.ec/pdf/enfoqueute/v8s1/1390-6542-enfoqueute-8-s1-00287.pdf>
- García, M., Irisarri, E., Pérez, F., Estévez, E., & Marcos, M. (2018). Arquitectura de Automatización basada en Sistemas Ciberfísicos para la Fabricación Flexible en la Industria de Petróleo y Gas. *Revista Iberoamericana de Automática e Informática Industrial*, 156-166.
- González Páramo, J. (2017). *Cuarta Revolución Industrial, empleo y Estado de Bienestar*. Madrid, España: Real Academia de las Ciencias Morales y Políticas .
- Grgic, K., Speh, I., & Hedi, I. (2016). A Web-Based IoT Solution for Monitoring Data Using MQTT Protocol. *International Conference on Smart Systems and Technologies (SST)*, 249-253. Obtenido de <https://scihub.tw/https://ieeexplore.ieee.org/abstract/document/7765668/similar#similar>
- Hunkeler, U., Truong, H., & Stanford-Clark, A. (2008). MQTT-S – A Publish/Subscribe Protocol For Wireless Sensor Networks. *3rd International Conference on Communication Systems Software and Middleware and*

- Workshops (COMSWARE '08)*, 1-8. Obtenido de <https://scihub.tw/https://ieeexplore.ieee.org/document/4554519>
- Jutadhamakorn, P., Pillavas, T., & Visoottiviseth, V. (2017). A Scalable and Low-Cost MQTT Broker Clustering System. *2nd International Conference on Information Technology (INCIT)*, 1-5. Obtenido de <https://scihub.tw/https://ieeexplore.ieee.org/document/8257870>
- Khattak, H., Ruta, M., & Di Sciascio, E. (2014). CoAP-based Healthcare Sensor Networks: a survey. *11th International Bhurban Conference on Applied Sciences & Technology (IBCAST)*, 499-503. Obtenido de <https://scihub.tw/https://ieeexplore.ieee.org/document/6778196>
- Kodali, R., & Mahesh, K. (2016). A low cost implementation of MQTT using ESP8266. *2nd International Conference on Contemporary Computing and Informatics (IC3I)*, 404-408. Obtenido de <https://scihub.tw/https://ieeexplore.ieee.org/document/7917998>
- Munner, B., Mohammed, Q., & Shadi, A. (2017). Internet of Things: Survey and open issues of MQTT protocol. *International Conference on Engineering & MIS (ICEMIS)*, 50-62.
- MySQL. (Enero de 2019). *MySQL.COM*. Obtenido de <https://www.mysql.com/>
- Oliván, R. (2016). La Cuarta Revolución Industrial, un relato desde el materialismo cultural. *Revista de Estudios Urbanos y Ciencias Sociales*, 101-111.
- Ortiz, J. (Septiembre de 2016). Entrevista, La cuarta revolución industrial. (S. Nebot, Entrevistador) doi: ISSN 1578-4967, N°. 637
- Páez, H., Zamora, R., & Bohórquez, J. (2015). Programación de Controladores Lógicos (PLC) mediante Ladder y Lenguaje de Control Estructurado (SCL). *Facultad de Ingeniería*, 109-119.
- Perrone, G., Vecchio, M., Pecori, R., & Giaffreda, R. (2017). The Day After Mirai: A Survey on MQTT Security Solutions After the Largest Cyber-attack Carried Out through an Army of IoT Devices. *In Proceedings of the 2nd International Conference on Internet of Things, Big Data and Security (IoTBDs 2017)*, 246-253. Obtenido de

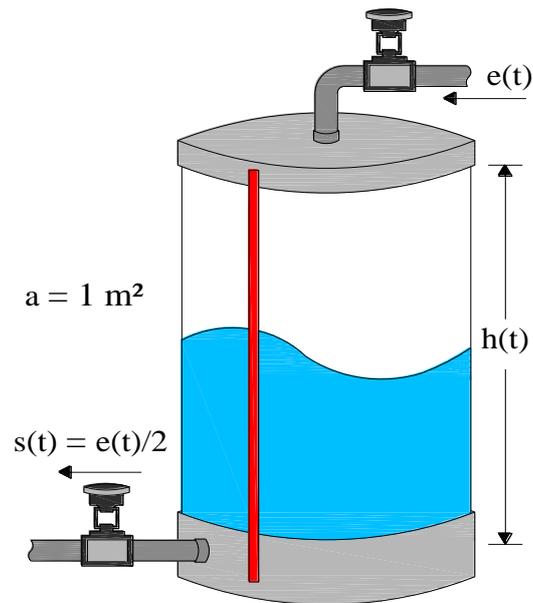
- <https://pdfs.semanticscholar.org/c118/faf091c1a2a538c66c5959efe6ecbaf222e.pdf>
- phpMyAdmin. (Enero de 2019). *Bringing MySQL to the web*. Obtenido de <https://www.phpmyadmin.net/>
- Pipatsakulroj, W., Visoottiviseth, V., & Takano, R. (2017). muMQ: A Lightweight and Scalable MQTT Broker. *IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*, 1-6. Obtenido de <https://scihub.tw/https://ieeexplore.ieee.org/document/7972165>
- Postscapes. (2017). *IoT Standards and Protocols*. Obtenido de <https://www.postscapes.com/internet-of-things-protocols/>
- Pradilla, J., González, R., Esteve, M., & Palau, C. (2016). Sensor Observation Service (SOS) / Constrained Application Protocol (CoAP) Proxy Design. *Mediterranean Electrotechnical Conference (MELECON)*. Obtenido de <https://scihub.tw/https://ieeexplore.ieee.org/document/7495411>
- python. (Enero de 2019). *Python.com*. Obtenido de <https://www.python.org/blogs/>
- RealGames. (12 de Enero de 2019). *Factory I / O*. Obtenido de 2006-2019 REAL GAMES: <https://factoryio.com/docs/> <https://factoryio.com/es/get-factory-io/>
- Rivoy, C. (2006). Contribución sobre los paradigmas de investigación. *Revista do Centro de Educação*, 31(1), 11-22. Obtenido de <http://www.redalyc.org/pdf/1171/117117257002.pdf>
- Sabalza, X. (2016). Reflexiones sobre la Industria 4.0 desde el caso vasco. *Ekonomiaz*, 142-173.
- Sanmartín, P., Ávila, K., Virola, C., & Jabba, D. (2016). Internet de las cosas y la salud centrada en el hogar. *Salud Uninorte*, 337-351.
- Sarafov, V. (2017). Comparison of IoT Data Protocol Overhead. *Seminar Future Internet* SS, 7-14. Obtenido de https://www.net.in.tum.de/fileadmin/TUM/NET/NET-2018-03-1/NET-2018-03-1_02.pdf

- Secretaría Nacional de Planificación y Desarrollo - Senplades 2017. (22 de Septiembre de 2017). Plan Nacional de Desarrollo 2017-2021. Toda una Vida. Quito, Ecuador, Pichincha, Ecuador: Consejo Nacional de Planificación (CNP).
- Shin, S., Kobara, K., Chuang, C.-C., & Huang, W. (2016). A Security Framework for MQTT. *IEEE Conference on Communications and Network Security (CNS)*, 1-5. Obtenido de <https://sci-hub.tw/https://ieeexplore.ieee.org/document/7860532>
- Siemens. (2014). Simatic S7 - 1200. *Siemens PLC*, 1-10. Obtenido de https://w5.siemens.com/spain/web/es/industry/automatizacion/sce_educacion/Documentacion-Didactica/Documents/SIMATIC%20S71200R.pdf
- Singh, M., Rajan, M., Shivraj, V., & Balamuralidhar, P. (2015). Secure MQTT for Internet of Things (IoT). *Fifth International Conference on Communication Systems and Network Technologies*, 746-751. Obtenido de <https://sci-hub.tw/https://ieeexplore.ieee.org/document/7280018>
- Sneha, M., & T.S., A. (2017). Performance optimization of PID controllers using fuzzy logic. *IEEE International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM)*, 438-442.
- Sonawala, N., Tank, B., & Patel, H. (2017). Implementation of MQTT protocol in Context with Industry 4.0. *Impact Factor (SJIF): 4.542*, 260-267.
- Tecnología para los negocios. (Diciembre de 2018). *Sistemas Ciberfísicos: la respuesta a las necesidades de la sociedad y la industria*. Obtenido de <https://ticnegocios.camaravalencia.com/servicios/tendencias/sistemas-ciberfisicos-la-respuesta-a-las-necesidades-de-la-sociedad-y-la-industria/>
- Terán , O., Espinosa, E., Hernández, P., & Flores, J. (2017). Internet de las Cosas IoT como herramienta para la optimización de la cadena de suministro del sector secundario. *Revista Global de Negocios*, 107-118.
- Ullas, B., Anush, S., Roopa, J., & Govinda, R. (2014). Machine to Machine Communication for Smart Systems using MQTT. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 8242-8248. Obtenido de https://www.ijareeie.com/upload/2014/march/67_Machine.pdf

- Universidad de Barcelona. (Enero de 2016). *En 2020, más de 30 mil millones de dispositivos estarán conectados a Internet*. Obtenido de <https://www.obs-edu.com/int/noticias/estudio-obs/en-2020-mas-de-30-mil-millones-de-dispositivos-estaran-conectados-internet>
- Upadhyay, Y., Borole, A., & Dileepan, D. (2016). MQTT Based Secured Home Automation System. *Symposium on Colossal Data Analysis and Networking (CDAN)*, 1-4. Obtenido de <https://sci-hub.tw/https://ieeexplore.ieee.org/abstract/document/7570945>
- Val Román, J., & Val Román, J. (2017). Industria 4.0: la transformación digital de la industria. *coddiinforme* (págs. 1-10). Bilbao, España: Facultad de Ingeniería de la Universidad de Deusto.
- Valverde, J. (2005). Software libre, la alternativa tecnológica para la educación. *Actualidades Investigativas En Educación*, 1-9.
- Villanueva, M. (2016). La Industria 4.0, una prioridad. *Revista de la ASOCIACIÓN ESPAÑOLA PARA LA CALIDAD*, 1-51.
- Wagner, G. (2016). Introduction to simulation using JavaScript. *IEEE*, 103-1011.
- Wang, S., Xu, D., & Yan, S. (2010). Analysis and application of Wireshark in TCP/IP protocol teaching. *International Conference on E-Health Networking Digital Ecosystems and Technologies (EDT)*, 269-272.
- Weyrich, M., Schmidt, J., & Ebert, C. (2014). Machine-to-Machine Communication. *IEEE*, 19-23.
- Wonderware. (Junio de 2016). *Interfaz Hombre-Máquina (HMI)*. Obtenido de <http://www.wonderware.es/hmi-scada/que-es-hmi/>
- Ynzunza, C., Izar, J., Bocarando, J., Aguilar, F., & Larios, M. (2017). El Entorno de la Industria 4.0: Implicaciones y Perspectivas Futuras. *Conciencia Tecnológica*, 1-19.
- Zhou, K., Liu, T., & Zhou, L. (2015). Industry 4.0: Towards Future Industrial Opportunities and Challenges. *12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, 2147-2152. Obtenido de <https://sci-hub.tw/https://ieeexplore.ieee.org/document/7382284>

ANEXOS

ANEXO 1: Modelado matemático del sistema



Área del tanque cilíndrico:

$$A = \pi * r^2$$

Y su volumen:

$$V = A * h$$

Entonces,

$$\Delta V = A * \Delta h$$

Relación de la variación del nivel del tanque,

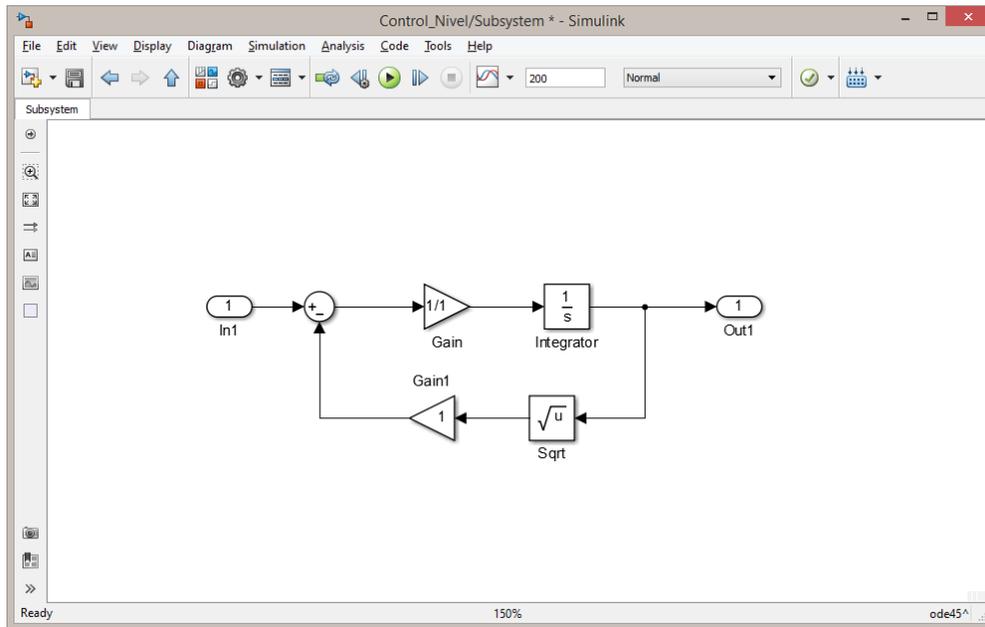
$$\Delta h = \frac{\Delta V}{a} = \frac{(V_{entrada} - V_{salida})}{(\pi * r^2)}$$

Es decir:

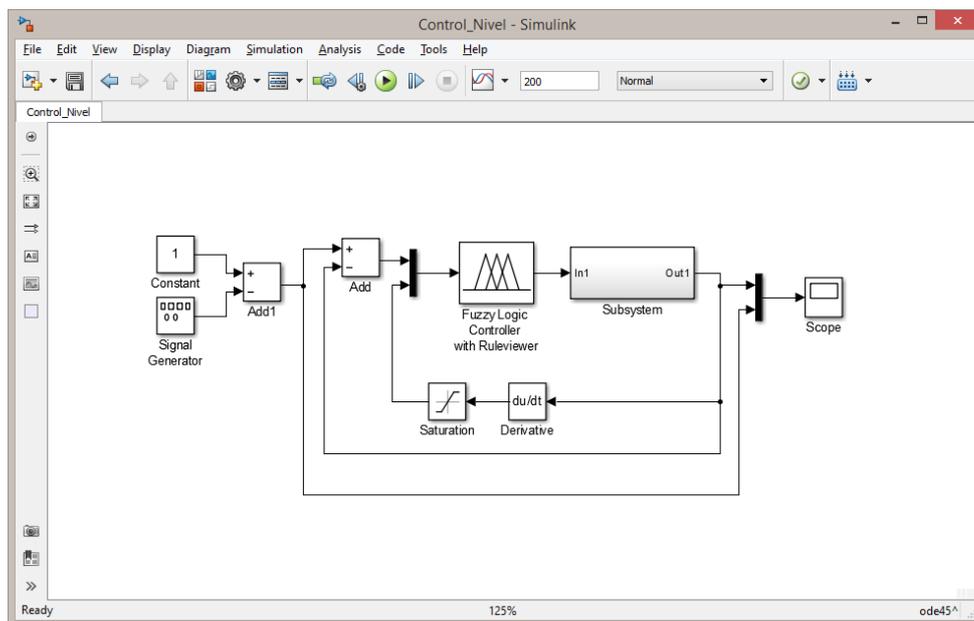
$$\dot{h} = \frac{(f_u - k^2 \sqrt{h})}{(\pi * r^2)}$$

$$\dot{h} = \frac{1}{A} * (f_u - k^2 \sqrt{h})$$

La siguiente figura muestra el modelo numérico del tanque en Matlab - Simulink, incluyendo las operaciones necesarias para la simulación de su llenado.



A continuación, se detalla la implementación de módulos en Matlab-Simulink para realizar la simulación numérica del proceso. La Figura muestra la programación de la plataforma de pruebas (Subsystem), y el controlador difuso, con sus respectivos módulos utilizados para observar del funcionamiento de cada controlador.



ANEXO 2: Tablas y gráficas de reglas de inferencia y funciones de membresía

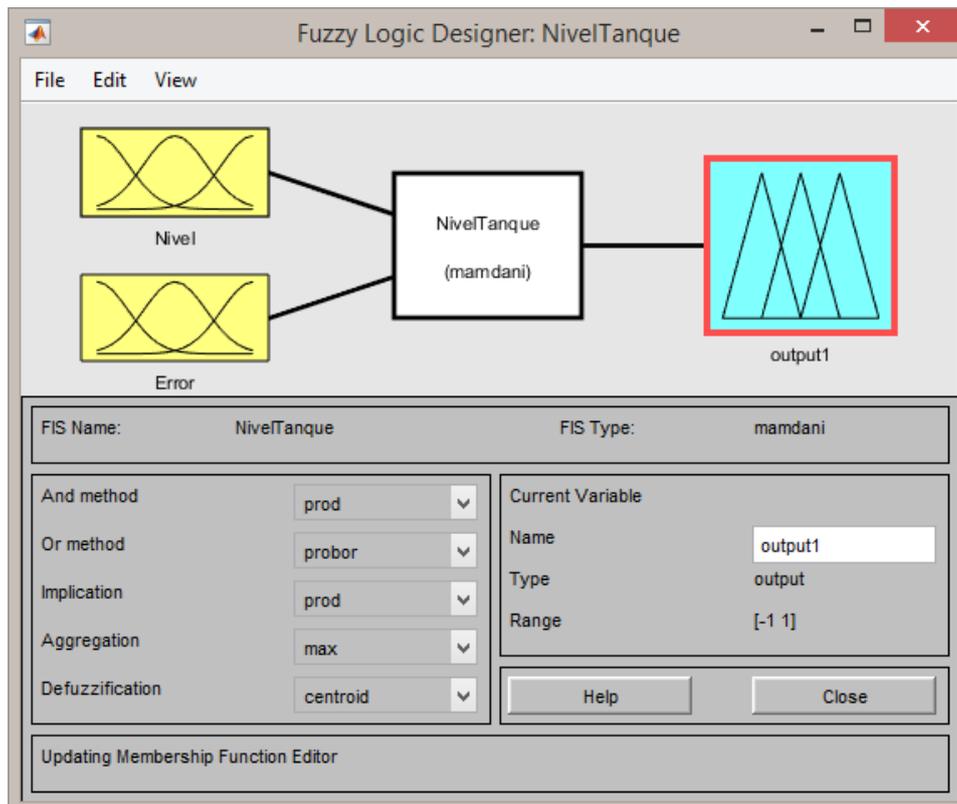
Tabla de estados de las variables lingüísticas

Estados de Nivel	Estados del Error	Estados de Válvula
Bajo	Negativo	Cerrar Rápido
Normal		Cerrar Lento
Alto	Positivo	No Mover
		Abrir Lento
		Abrir Rápido

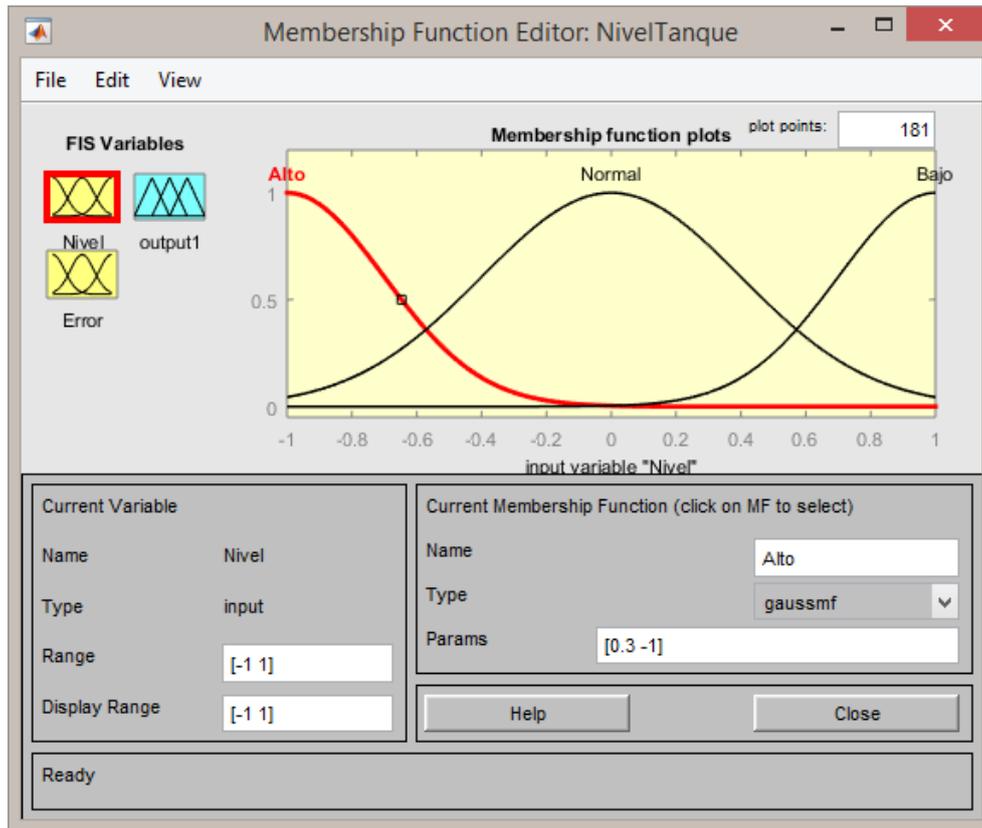
Tabla de reglas de inferencia del controlador difuso

No. Regla	Detalle
Regla 1	Si el valor de Nivel es Normal, entonces la Válvula deberá NoMover
Regla 2	Si el valor de Nivel es Bajo, entonces la Válvula deberá AbrirRapido
Regla 3	Si el valor de Nivel es Alto, entonces la Válvula deberá CerrarRapido
Regla 4	Si el valor de Nivel es Normal y el Error es Negativo, entonces la Válvula deberá AbrirLento
Regla 5	Si el valor de Nivel es Normal y el Error es Positivo, entonces la Válvula deberá AbrirLento

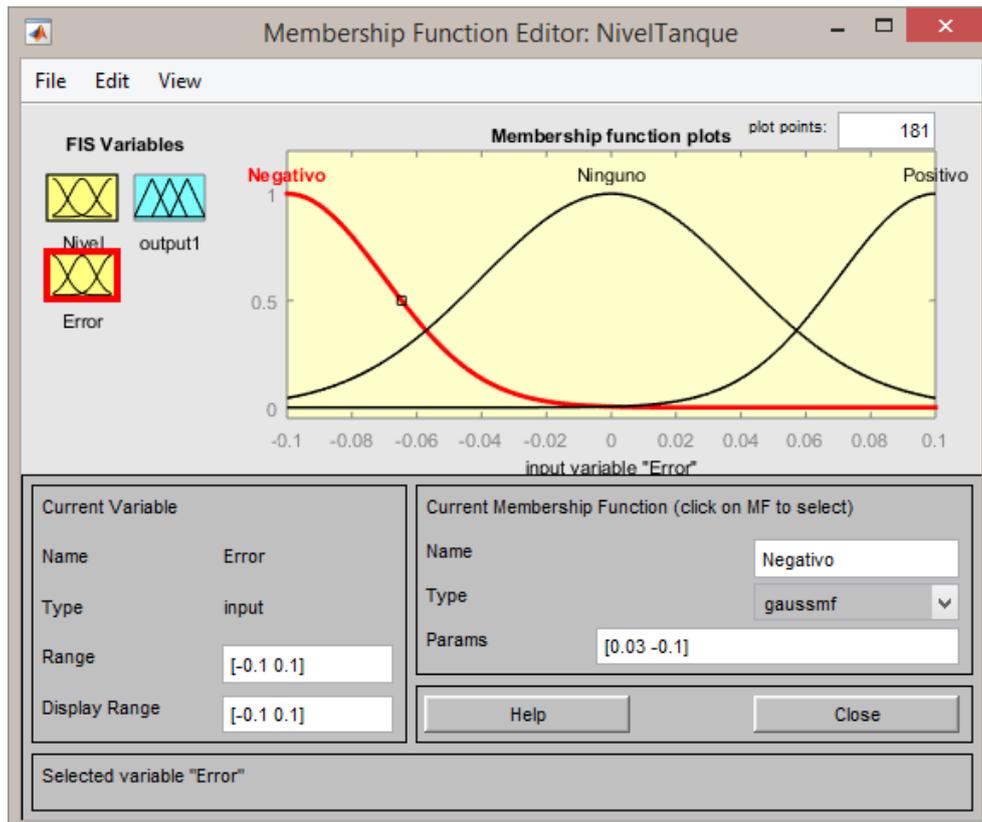
Gráfica de visualización del modelo difuso Matlab



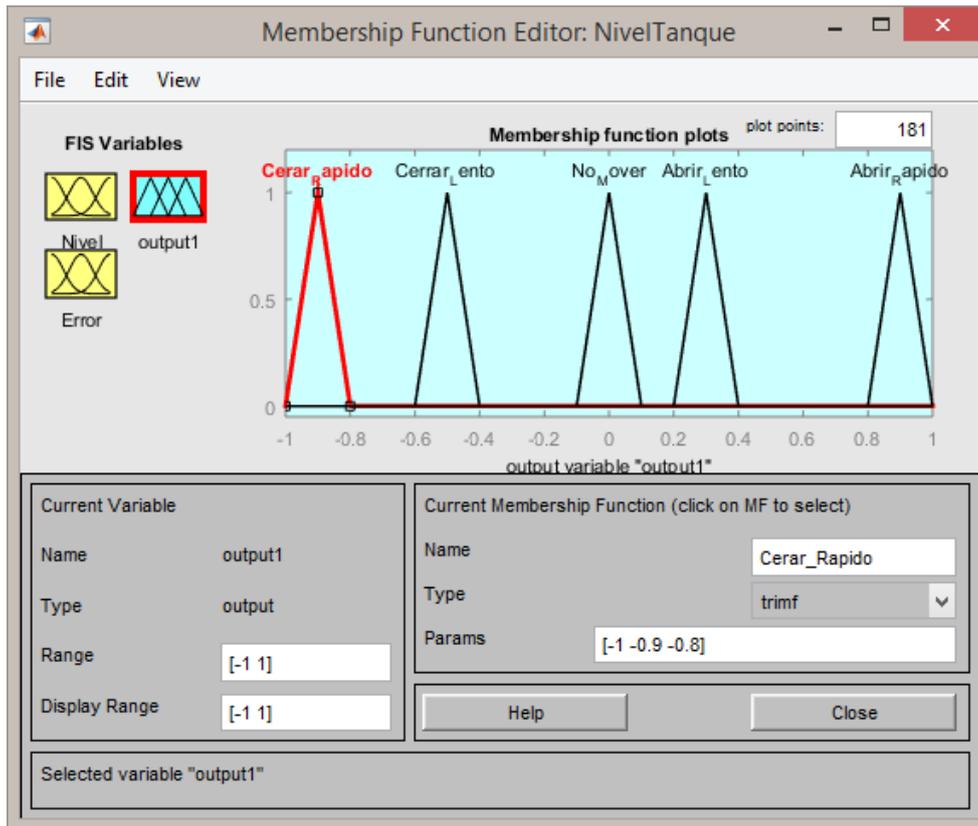
Gráfica de la Función de Membresía Nivel



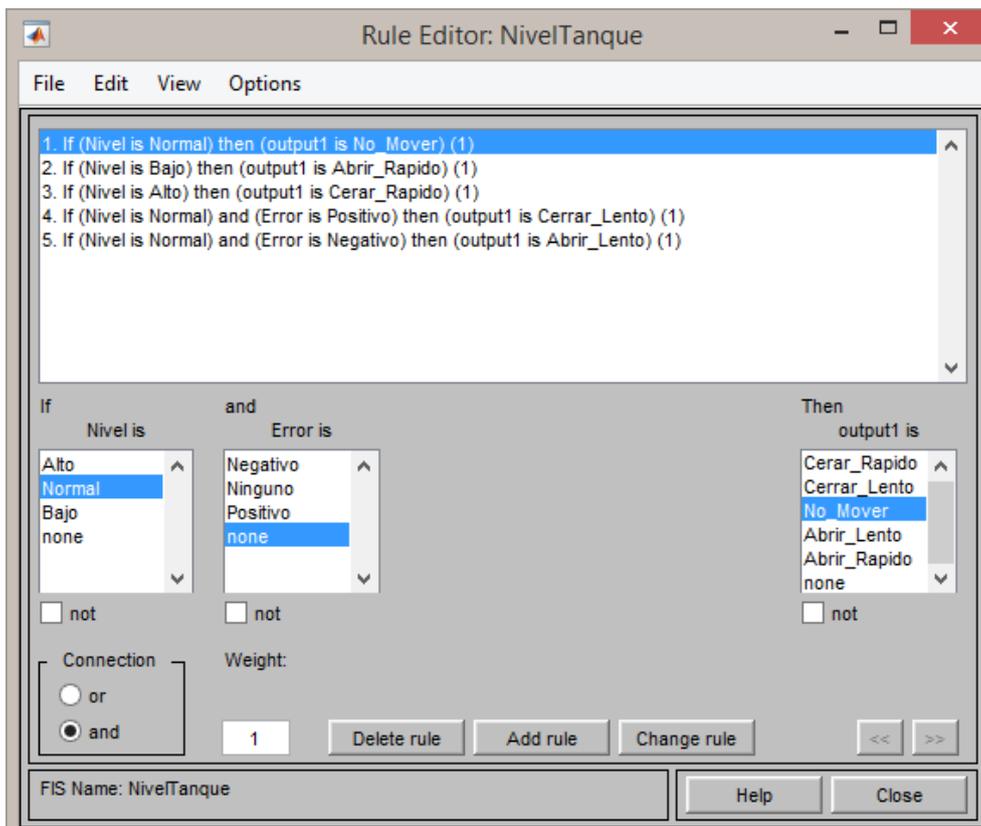
Gráfica de la Función de Membresía Error



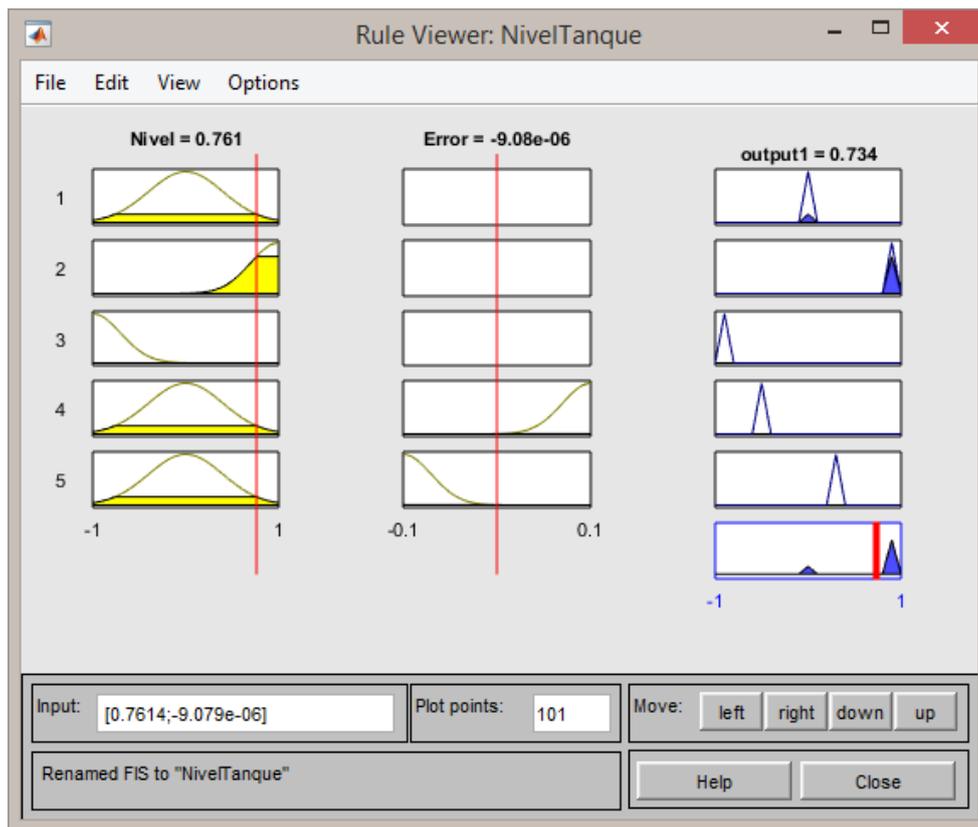
Gráfica de la Función de Membresía salida Válvula



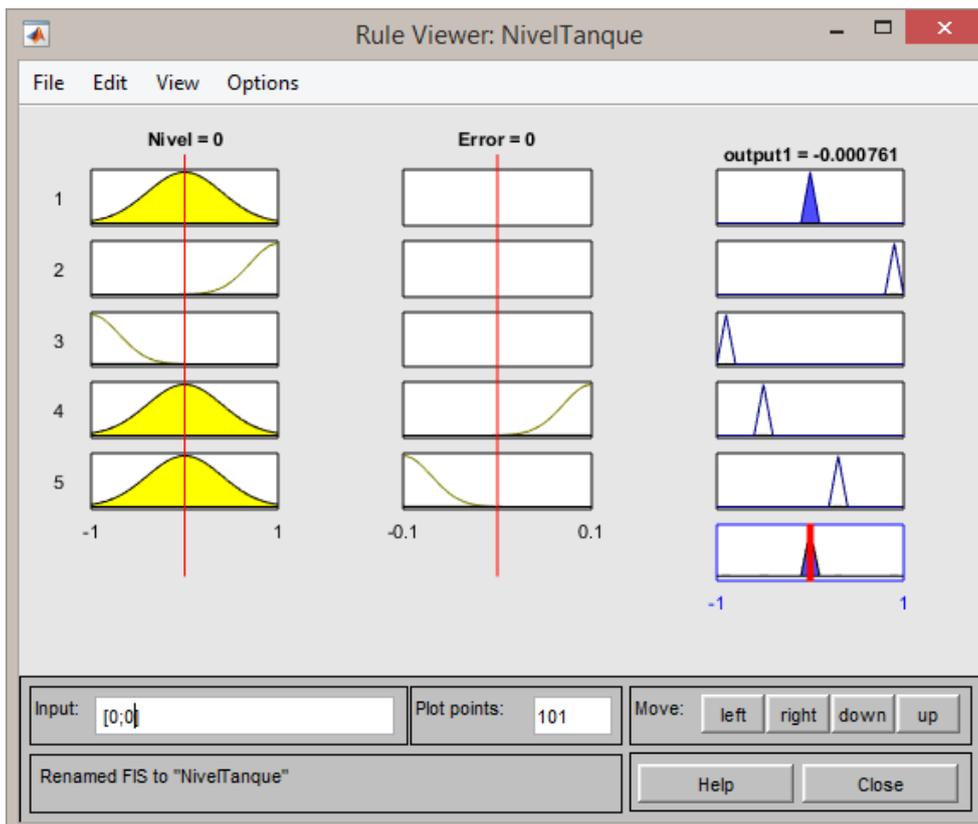
Gráfica Reglas del Sistema de Inferencia difuso



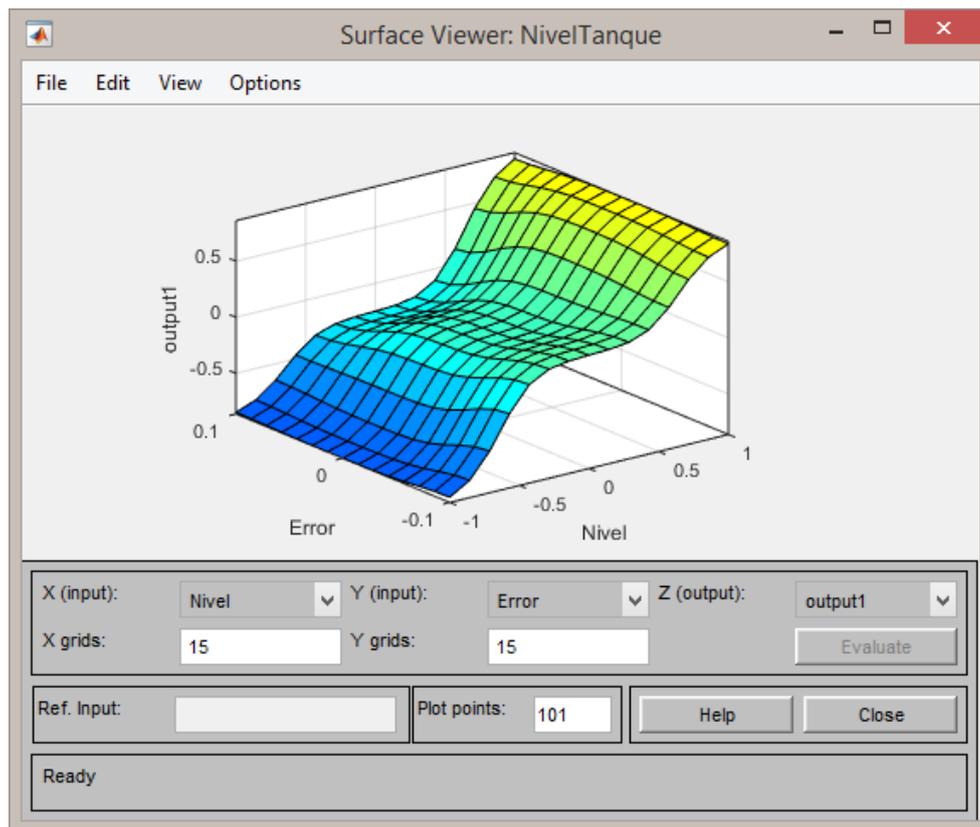
Gráfica 1 de Visualización de evaluación de reglas



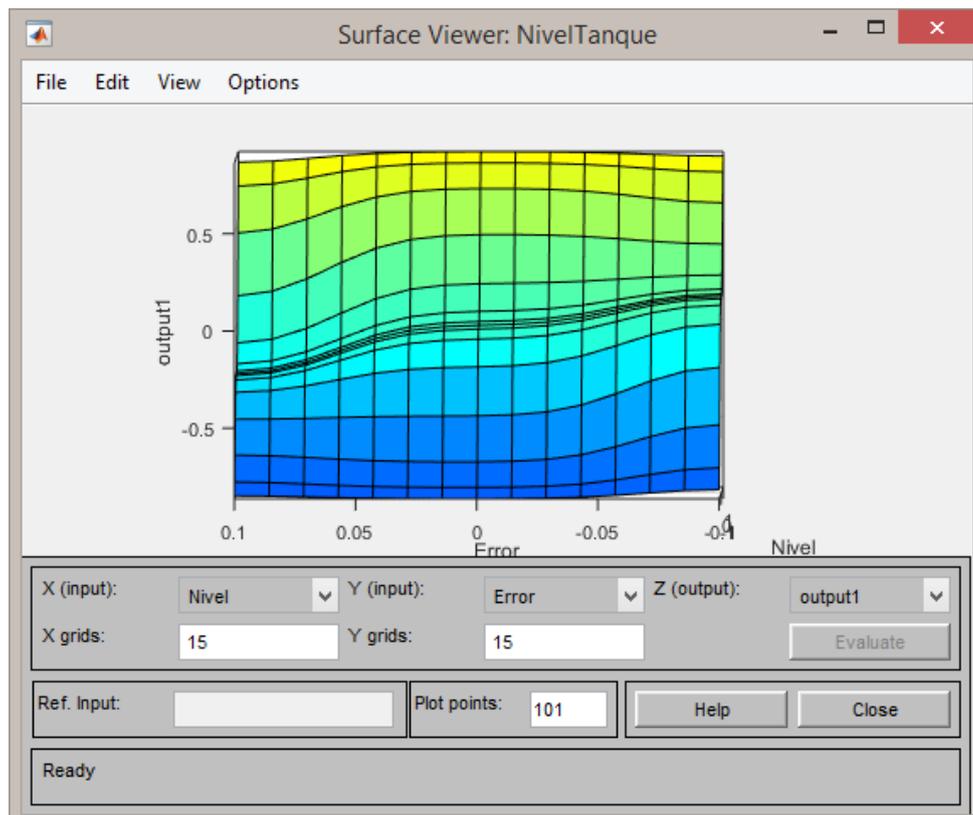
Gráfica 2 de Visualización de evaluación de reglas



Gráfica 1 de la Superficie de decisión NivelTanque

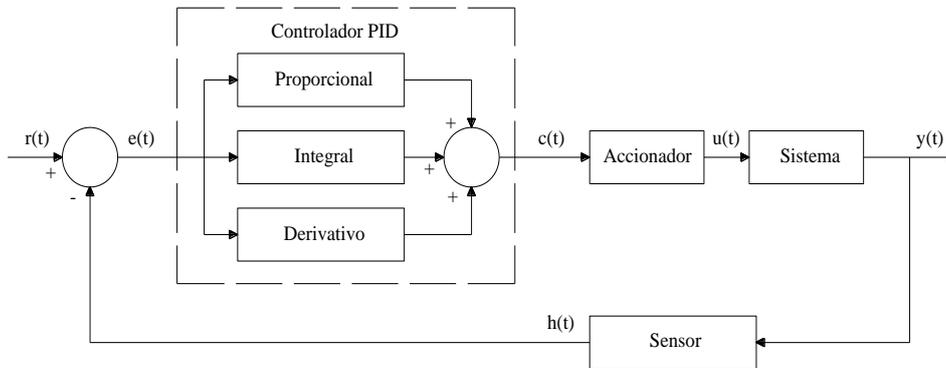


Gráfica 2 de la Superficie de decisión NivelTanque

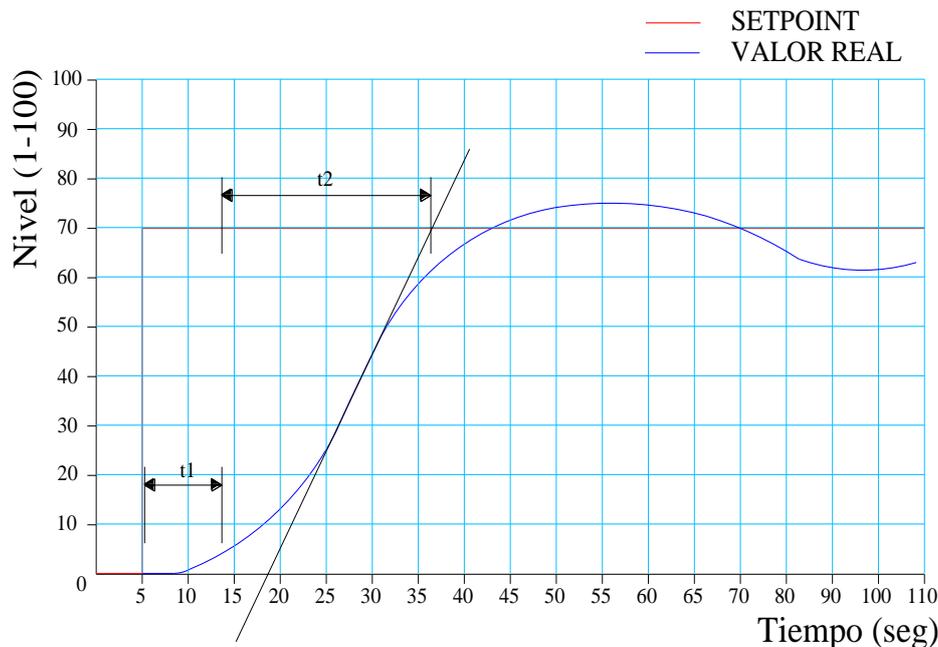


ANEXO 3: Cálculos de sintonización del controlador PID

Mediante el método de Ziegler-Nichols logramos ajustar y/o sintonizar nuestro controlador PID, sin necesidad de conocer la ecuación de la planta o sistema controlado. Los valores propuestos por este método permiten obtener en el sistema realimentado una respuesta al escalón con un sobre pulso robusto con buenas características de rapidez y estabilidad. Definimos entonces las ganancias proporcional, integral y derivativa a partir de la respuesta del sistema en lazo cerrado.

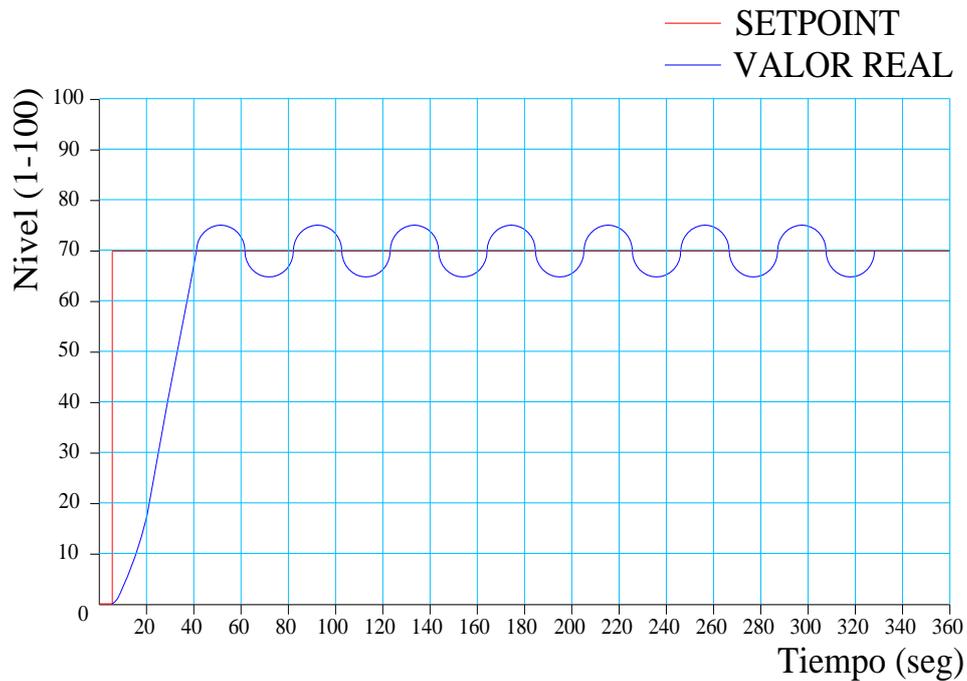


A continuación, se presenta las gráficas de respuesta del sistema ante una entrada escalón $c(t)$ de valor 70, teniendo como resultado un t_1 (tiempo muerto) y t_2 (tiempo de subida) de valores.



Al evidenciar una respuesta oscilatoria mantenida, se reduce al mínimo la acción derivativa y la acción integral del regulador PID. La ganancia proporcional debe lograr

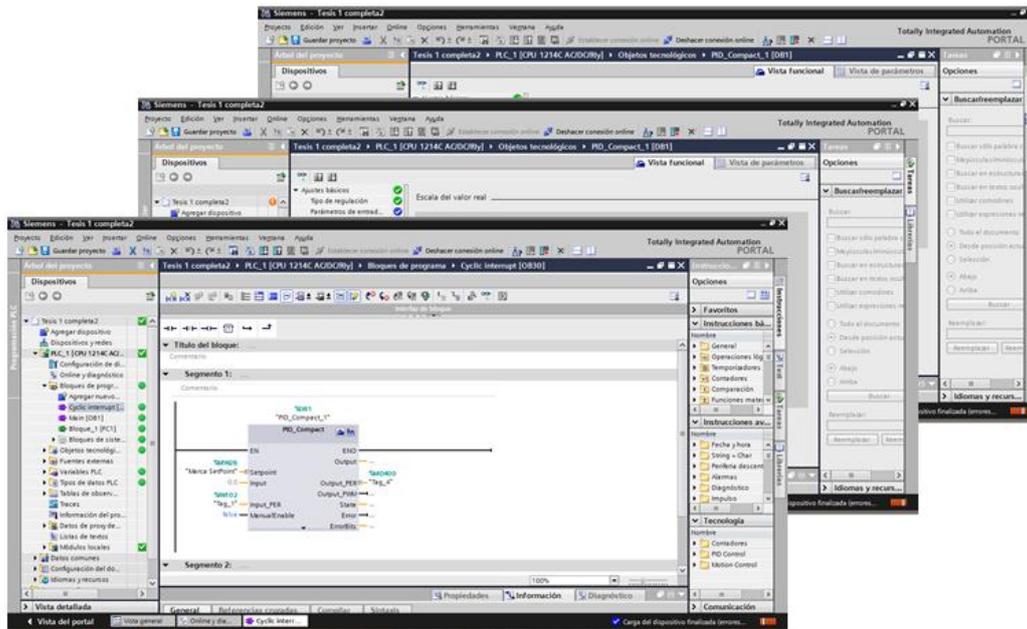
que el sistema oscile de forma mantenida ante cualquier perturbación. Esta oscilación debe ser lineal, sin saturaciones. En este momento se determina la ganancia proporcional, llamada ganancia crítica o K_c , y el periodo de oscilación T_c en segundos.



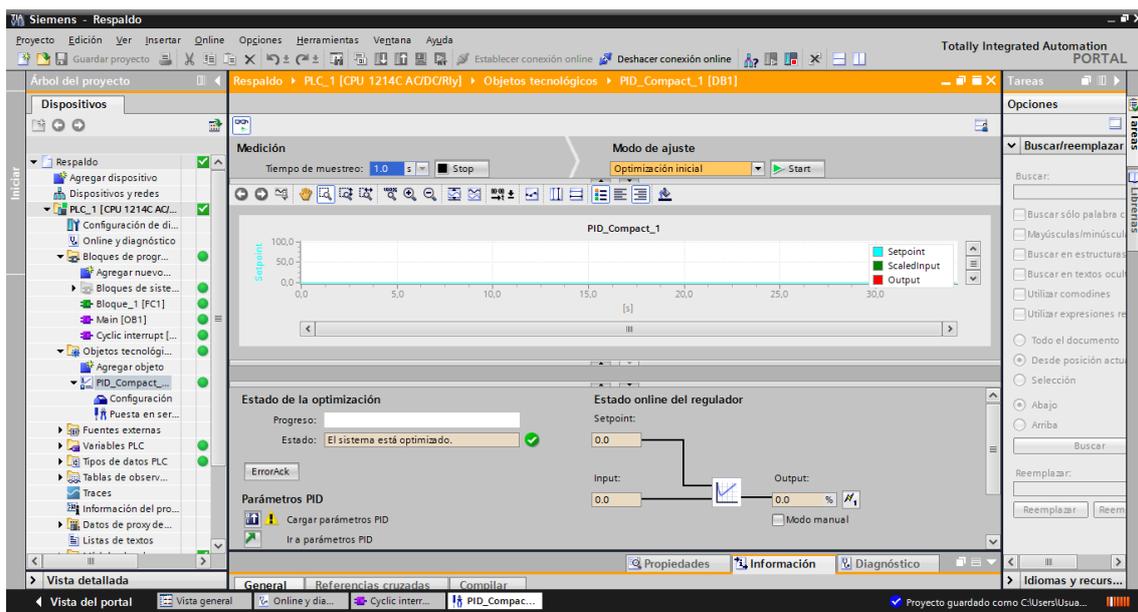
La ganancia proporcional de valor $K_p = 16.13$ se obtiene una respuesta con oscilaciones mantenidas. Por lo tanto, la ganancia crítica y el periodo son: $K_c = 9,51$ y $T_c = 0,000334$.

Controlador/Constantes	K_p	T_i	T_d
Formula PID	$0,59 * K_c$	$0,50 * T_c$	$0,125 * T_c$
Parámetros PID	9,5167	0,000167	0,00004175

Adicionalmente se utilizó el sintonizador automático del Controlador Lógico Programable PLC Siemens S7 1200, a través de su software de control TIA Portal V13, y la puesta en servicio del Bloque PID Compact



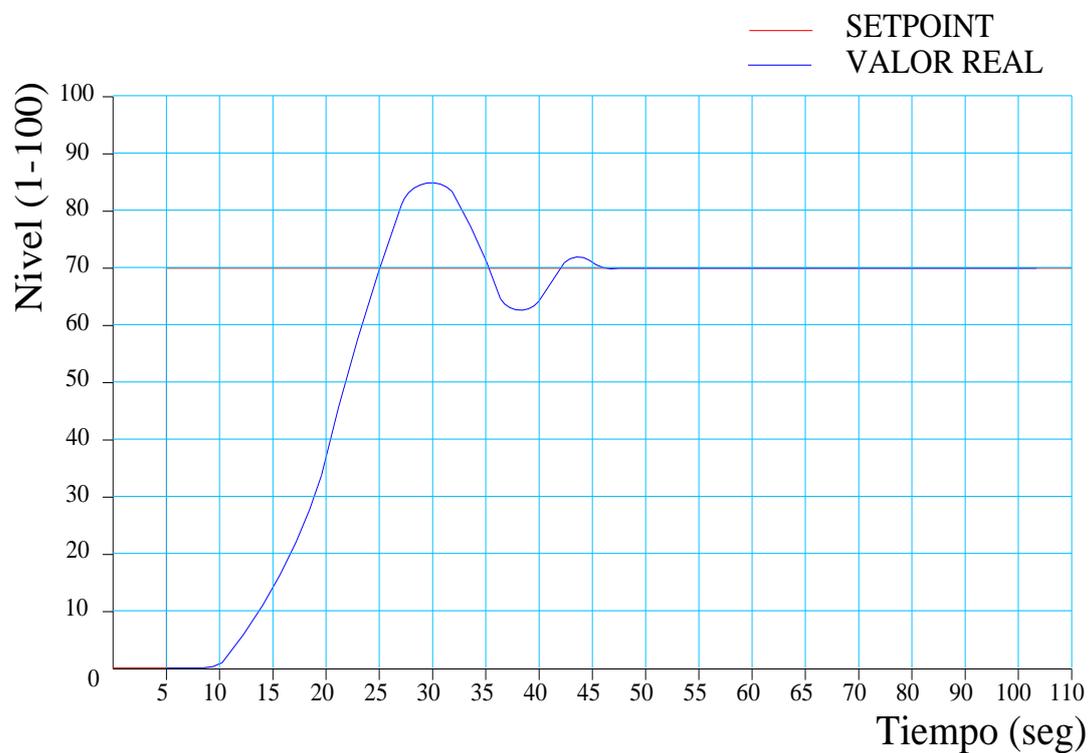
La opción Autotunnig permite realizar pruebas del comportamiento y ajuste automático del regulador PID determinando los valores y constantes para un control lo más optimo posible. Sin errores existentes y después de comprobar la medición y optimización de la herramienta, se carga en el sistema los parámetros obtenidos.



Los valores obtenidos por la herramienta Autotuning son semejantes a los valores obtenidos anteriormente con el método de Ziegler-Nichols.

Método/Constantes	Kp	Ti	Td
Ziegler-Nichols	9,5167	0,000167	0,00004175
Autotuning S7 1200	9,987	0,0001	0,01
Afinación manual	9,67	0,000101	0,1

Particularmente y en base a los valores obtenidos de los métodos de sintonización se terminó de afinar el regulador PID a mano para conseguir una respuesta un poco más rápida y estable. Introduciendo estos valores y realizando las pruebas respectivas se obtiene la siguiente respuesta del sistema de nivel con controlador PID.



ANEXO 4: Tabla de variables y Topics del sistema

VARIABLE	TIPO	DIRECCIÓN	TOPIC	SUB TOPIC 1	SUB TOPIC 2	SUB TOPIC 3	SUB TOPIC 4
Válvula Ent.	Int	%QW104	PLC	PLC1	TANQUE	TANQUE	VALVULA_ENTRADA
Banda 1	Bool	%Q0.0	PLC	PLC1	ENSAMBLAJE	BANDAS	BANDA1
Banda 2	Bool	%Q0.1	PLC	PLC1	ENSAMBLAJE	BANDAS	BANDA2
Posicionador 1	Bool	%Q0.2	PLC	PLC1	ENSAMBLAJE	SENSOR	POSICIONADOR_1
Marca SetPoint	Int	%MW28	PLC	PLC1	CONTROL	TANQUE	SET_POINT
					TANQUE	TANQUE	SET_POINT
M. Valor Real Tanque	Int	%MW202	PLC	PLC1	TANQUE	TANQUE	VALOR_REAL
Marca Star	Bool	%M0.0	PLC	PLC1	ENSAMBLAJE	CONTROL	START
Brazo X	Bool	%Q0.5	PLC	PLC1	ENSAMBLAJE	BRAZO	BRAZO_X
Elevador	Bool	%Q0.3	PLC	PLC1	ENSAMBLAJE	BRAZO	BRAZO_Z
Sensor 5	Bool	%I0.5	PLC	PLC1	ENSAMBLAJE	SENSOR	SALIDA
Sensor 1	Bool	%I1.0	PLC	PLC1	ENSAMBLAJE	SENSOR	ENTRADA_1
Sensor 2	Bool	%I1.1	PLC	PLC1	ENSAMBLAJE	SENSOR	ENTRADA_2
Tiempo producción	Int	%MW32	PLC	PLC1	ENSAMBLAJE	TIEMPO	PRODUCCION
Productos/hora	Int	%MW34	PLC	PLC1	ENSAMBLAJE	PRODUCCION	PRODUCTOS_HORA
Med. banda 1, seg	Word	%MW36	PLC	PLC1	ENSAMBLAJE	TIEMPO	BANDA_1
Tiempo banda 1 t/p	Int	%MW38	PLC	PLC1	ENSAMBLAJE	PRODUCCION	BANDA_1
Med. banda 2, seg	Int	%MW40	PLC	PLC1	ENSAMBLAJE	TIEMPO	BANDA_2
Tiempo banda 2 t/p	Int	%MW42	PLC	PLC1	ENSAMBLAJE	PRODUCCION	BANDA_2
Med. T posic. 1, seg	Int	%MW46	PLC	PLC1	ENSAMBLAJE	TIEMPO	POSICIONADOR_1
T. Posic. 1 t/p	Int	%MW50	PLC	PLC1	ENSAMBLAJE	PRODUCCION	POSICIONADOR_1
Med. Posic.2, seg	Int	%MW52	PLC	PLC1	ENSAMBLAJE	TIEMPO	POSICIONADOR_2

T. Posic. 2 t/p	Int	%MW54	PLC	PLC1	ENSAMBLAJE	PRODUCCION	POSICIONADOR_2
Med. brazo T, seg	Int	%MW58	PLC	PLC1	ENSAMBLAJE	TIEMPO	BRAZO
Tiempo brazo t/p	Int	%MW66	PLC	PLC1	ENSAMBLAJE	PRODUCCION	BRAZO
Med. banda caja, seg	Int	%MW68	PLC	PLC1	ENSAMBLAJE	TIEMPO	BANDA_CAJA
T. banda caja t/p	Int	%MW70	PLC	PLC1	ENSAMBLAJE	PRODUCCION	BANDA_CAJA
Marca Start	Bool	%M3.1	PLC	PLC1	CONTROL	GENERAL	START_WEB
M. Paro Emergencia	Bool	%M3.0	PLC	PLC1	CONTROL	GENERAL	PARO_WEB
Marca Stop	Bool	%M3.2	PLC	PLC1	CONTROL	GENERAL	STOP_WEB
Cant. productos	Int	%MW74	PLC	PLC1	ENSAMBLAJE	PRODUCCION	PRODUCTOS
Cant. cajas	Int	%MW76	PLC	PLC1	ENSAMBLAJE	PRODUCCION	CAJAS
Marca reset	Bool	%M3.3	PLC	PLC1	CONTROL	GENERAL	RESET_WEB
Cant. Prod. base	Int	%MW78	PLC	PLC1	ENSAMBLAJE	PRODUCCION	PRODUCTOS_BASE
Cant. Prod. Lid	Int	%MW80	PLC	PLC1	ENSAMBLAJE	PRODUCCION	PRODUCTOS_LIQ
Marca Kp	Word	%MD82	PLC	PLC1	CONTROL	TANQUE	KP
Marca Ki	Word	%MD86	PLC	PLC1	CONTROL	TANQUE	KI
Marca Td	Word	%MD90	PLC	PLC1	CONTROL	TANQUE	KD
Productos/caja	Word	%MW72	PLC	PLC1	ENSAMBLAJE	PRODUCCION	PRODUCTOS_POR_CAJA
Marca Kp	Word	%MW84	WEB	PLCWeb	TANQUE	PID	KP
Marca Ki	Word	%MW88	WEB	PLCWeb	TANQUE	PID	KI
Marca Td	Word	%MW92	WEB	PLCWeb	TANQUE	PID	KD
No. Productos	Int	%MW94	PLC	PLC1	CONTROL	ENSAMBLAJE	PRODUCTOS_POR_CAJA

ANEXO 6: Código de Programa

SNAP7

```
from time import sleep
import snap7
from snap7.util import *
import struct

class output(object):
    bool=1
    int=2
    real=3
    word=4
    dword=5

class S71200():

    def __init__(self,debug=False):
        self.debug = debug
        self.plc = snap7.client.Client()
    #método para conectar y reconectar
    def connect(self,plc,ip):
        self.plc = snap7.client.Client()
        self.ip=ip
        while True:
            #Verificar conexion
            if self.plc.get_connected():
                break
            try:
                #si no se ha verificado, intentar conectar
                self.plc.connect(ip,0,0)
            #si no se pudo, pasar y esperar 5 segundos para repetir
            except:
                pass
                sleep(5)

    #Método para la lectura de las direcciones de memoria
    def getMem(self,mem,returnByte=False):
        try:
            #definicion de variables por defecto
            area=0x83
            length=1
            type=0
            out=None
            bit=0
            start=0
            if(mem[0].lower()=='m'):
                area=0x83
            if(mem[0].lower()=='q'):
```

```

        area=0x82
    if(mem[0].lower()=='i'):
        area=0x81
    if(mem[1].lower()=='x'): #bit
        length=1
        out=output().bool
        start = int(mem.split('.')[0][2:])
    if(mem[1].lower()=='b'): #byte
        length=1
        out=output().int
        start = int(mem[2:])
    if(mem[1].lower()=='w'): #word
        length=2
        out=output().int
        start = int(mem[2:])
    if(mem[1].lower()=='d'):
        out=output().dword
        length=4
        start = int(mem.split('.')[0][2:])
    if(mem[1].lower()=='r'): #double word (real numbers)
        length=4
        start = int(mem.split('.')[0][2:])
        out=output().real
    #print start,hex(area)
    if(output().bool==out):
        bit = int(mem.split('.')[1])
    if(self.debug):
        print mem[0].lower(),bit
    self.plc.read_area(area,0,start,length)

    mbyte=self.plc.read_area(area,0,start,length)

    if(returnByte):
        return mbyte
    elif(output().bool==out):
        return get_bool(mbyte,0,bit)
    elif(output().int==out):
        return get_int(mbyte,0)
    elif(output().real==out):
        return get_real(mbyte,0)
    elif(output().dword==out):
        return get_dword(mbyte,0)
    elif(output().word==out):
        return get_int(mbyte,start)

except:
    self.connect(self.plc,self.ip)
def writeMW(self,mem,value):
    try:
        area=0x83
        length=1
        type=0

```

```

out=None
bit=0
start=0
if(mem[0].lower()=='m'):
    area=0x83
if(mem[0].lower()=='q'):
    area=0x82
if(mem[0].lower()=='i'):
    area=0x81
if(mem[1].lower()=='x'): #bit
    length=1
    out=output().bool
    start = int(mem.split('.')[0][2:])
if(mem[1].lower()=='b'): #byte
    length=1
    out=output().int
    start = int(mem[2:])
if(mem[1].lower()=='w'): #word
    length=2
    out=output().int
    start = int(mem[2:])
if(mem[1].lower()=='d'): #dword
    out=output().dword
    length=4
    start = int(mem.split('.')[0][2:])
if(mem[1].lower()=='r'): #double word (real numbers)
    length=4
    start=int(mem.split('.')[0][2:])
    out=output().real
if(output().bool==out):
    bit = int(mem.split('.')[1])
if(self.debug):
    print mem[0].lower(),bit

bit = int(mem.split('.')[1])

self.plc.read_area(area,0,start,length)
mbyte=self.plc.read_area(area,0,start,length)
if(output().bool==out):
    set_bool(mbyte,0,bit,int(value))
elif(output().int==out):
    set_int(mbyte,0,int(value))
elif(output().real==out):
    set_real(mbyte,0,value)
elif(output().dword==out):
    set_dword(mbyte,0,int(value))

self.plc.write_area(area,0,start,mbyte)

except:
    self.connect(self.plc,self.ip)

```

RASPBERRY QUE OBTIENE DATOS DEL PLC

```
import paho.mqtt.client as mqtt
import os, urlparse
import time
from time import sleep
import snap7
from snap7.util import *
import struct
import S71200

plc=S71200.S71200()#defino la variable que llama a la clase S71200

plc.connect(plc,"192.168.0.13") #IP del PLC

# Definicion de eventos de la comunicacion MQTT
def on_connect(client, userdata, flags, rc):
    print("rc: " + str(rc))
#Cuando llega un mensaje de un topic al que se suscribio
def on_message(client, obj, msg):
    print(msg.topic + " " + str(msg.qos) + " " + str(msg.payload))
    if msg.topic == "PLC/PLC1/CONTROL/GENERAL/START_WEB":
        plc.writeMW('MX3.1',True)
        plc.writeMW('MX3.2',False) #STOP
        plc.writeMW('MX3.3',False) #RESET
        plc.writeMW('MX3.0',False) #paro emergencia
    if msg.topic == "PLC/PLC1/CONTROL/GENERAL/STOP_WEB":
        plc.writeMW('MX3.1',False)
        plc.writeMW('MX3.2',True) #STOP
        plc.writeMW('MX3.3',False) #RESET
        plc.writeMW('MX3.0',False) #Paro emergencia
    if msg.topic == "PLC/PLC1/CONTROL/GENERAL/PARO_WEB":
        plc.writeMW('MX3.1',False)
        plc.writeMW('MX3.2',False) #STOP
        plc.writeMW('MX3.3',False) #RESET
        plc.writeMW('MX3.0',True) #Paro emergencia
    if msg.topic == "PLC/PLC1/CONTROL/GENERAL/RESET_WEB":
        plc.writeMW('MX3.1',False)
        plc.writeMW('MX3.2',False) #STOP
        plc.writeMW('MX3.3',True) #RESET
        plc.writeMW('MX3.0',False) #Paro emergencia
    if msg.topic == "PLC/PLC1/CONTROL/TANQUE/SET_POINT":
        print(plc.getMem('MW28'))
        plc.writeMW('MW28',msg.payload) #SET POINT desde WEB
        print(plc.getMem('MW28'))
    if msg.topic == "PLC/PLC1/CONTROL/TANQUE/KP":
        print(plc.getMem('MR82')) #R indica el tipo de dato real
        plc.writeMW('MR82',msg.payload) #KP desde WEB
        print(plc.getMem('MR82'))
```

```

if msg.topic == "PLC/PLC1/CONTROL/TANQUE/KI":
    print(plc.getMem('MR86'))
    plc.writeMW('MR86',msg.payload) #KI desde WEB
    print(plc.getMem('MR86'))
if msg.topic == "PLC/PLC1/CONTROL/TANQUE/KD":
    print(plc.getMem('MR90'))
    plc.writeMW('MR90',msg.payload) #KD desde WEB
    print(plc.getMem('MR90'))
if msg.topic == "PLC/PLC1/CONTROL/ENSAMBLAJE/PRODUCTOS_POR_CAJA":
    print(plc.getMem('MW94'))
    plc.writeMW('MW94',msg.payload) #Productos por caja
    print(plc.getMem('MW94'))

def on_publish(client, obj, mid):
    print("mid: " + str(mid))

def on_subscribe(client, obj, mid, granted_qos):
    print("Subscribed: " + str(mid) + " " + str(granted_qos))

def on_log(client, obj, level, string):
    print(string)

#Se define como cliente, llama a la libreria
mqttc = mqtt.Client()

# Asignacion de eventos, llama a los metodos
mqttc.on_message = on_message
mqttc.on_connect = on_connect
mqttc.on_publish = on_publish
mqttc.on_subscribe = on_subscribe

# Descomentar para abilitar el debug de mensajes
mqttc.on_log = on_log

# Conexion al broker mosquitto
mqttc.username_pw_set("USER_PLC1", "CONTR_PLC1") # Usuario, contraseña
mqttc.connect("186.33.173.57", "8883") #Direccion IP y puerto

# Iniciar suscripcion al topic.. con QoS nivel 0
mqttc.subscribe("PLC/PLC1/CONTROL/#", 0)

#Escritura de valores iniciales en el PLC al momento de encendido
#Productos por caja por defecto = 1
print plc.getMem('MW94')
if plc.getMem('MW94')==0:
    plc.writeMW('MW94',1)
    print plc.getMem('MW94')
# Bucle para la ejecucion del programa
rc = 0

```

```

dato=0
ciclo=0
banda1=0
banda2=0
brazo=0
salida=0
while rc == 0:
    rc = mqttc.loop()
    ciclo=ciclo+1
    #Productos por caja no debe estar en 0
    if plc.getMem('MW94')==0:
        plc.writeMW('MW94',1)
        print plc.getMem('MW94')
    #Sistema encendido o apagado
    # BANDA TRANSPORTADORA
    if plc.getMem('QX0.0')==False and banda1!=0:
        banda1=0
    mqttc.publish("PLC/PLC1TOWEB/ENSAMBLAJE/BANDAS/BANDA_1","False")
    if plc.getMem('QX0.0')==True and banda1==0:
        mqttc.publish("PLC/PLC1TOWEB/ENSAMBLAJE/BANDAS/BANDA_1","True")
        banda1=1
    if plc.getMem('QX0.1')==False and banda2!=0:
        mqttc.publish("PLC/PLC1TOWEB/ENSAMBLAJE/BANDAS/BANDA_2","False")
        banda2=0
    if plc.getMem('QX0.1')==True and banda2==0:
        mqttc.publish("PLC/PLC1TOWEB/ENSAMBLAJE/BANDAS/BANDA_2","True")
        banda2=1
    if plc.getMem('QX0.7')==False and brazo!=0:
        mqttc.publish("PLC/PLC1TOWEB/ENSAMBLAJE/BRAZO/BRAZO_ENSAMBLE","False")
        brazo=0
    if plc.getMem('QX0.7')==True and brazo==0:
        mqttc.publish("PLC/PLC1TOWEB/ENSAMBLAJE/BRAZO/BRAZO_ENSAMBLE","True")
        brazo=1
    if salida==0:
        mw74=plc.getMem('MW74')
        salida=1
    if salida==1 and mw74 != plc.getMem('MW74'):
        mqttc.publish("PLC/PLC1TOWEB/ENSAMBLAJE/SENSOR/SALIDA","True")
        mw74=plc.getMem('MW74')
        salida=1

    if ciclo==10: #Control de intervalo de tiempo para la accion

mqttc.publish("PLC/PLC1/TANQUE/TANQUE/SET_POINT",plc.getMem('MW28')) # EL USUARIO
DETERMINA ESTE VALOR
mqttc.publish("PLC/PLC1/TANQUE/TANQUE/VALOR_REAL",plc.getMem('MW202'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/CONTROL/START",plc.getMem('MX0.0'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/TIEMPO/PRODUCCION",plc.getMem('MW32'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/PRODUCTOS_HORA",plc.getMem('MW34'))

```

```

mqttc.publish("PLC/PLC1/ENSAMBLAJE/TIEMPO/BANDA_1",plc.getMem('MW36'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/BANDA_1",plc.getMem('MW38'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/TIEMPO/BANDA_2",plc.getMem('MW40'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/BANDA_2",plc.getMem('MW42'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/TIEMPO/POSICIONADOR_1",plc.getMem('MW46'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/POSICIONADOR_1",plc.getMem('MW50'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/TIEMPO/POSICIONADOR_2",plc.getMem('MW52'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/POSICIONADOR_2",plc.getMem('MW54'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/TIEMPO/BRAZO",plc.getMem('MW58'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/BRAZO",plc.getMem('MW66'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/TIEMPO/BANDA_CAJA",plc.getMem('MW68'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/BANDA_CAJA",plc.getMem('MW70'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/PRODUCTOS_POR_CAJA",plc.getMem('MW72')
)
mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/PRODUCTOS",plc.getMem('MW74'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/CAJAS",plc.getMem('MW76'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/PRODUCTOS_BASE",plc.getMem('MW78'))
mqttc.publish("PLC/PLC1/ENSAMBLAJE/PRODUCCION/PRODUCTOS_LIQ",plc.getMem('MW80'))
mqttc.publish("PLC/PLC1/TANQUE/PID/KP",plc.getMem('MR82'))
mqttc.publish("PLC/PLC1/TANQUE/PID/KI",plc.getMem('MR86'))
mqttc.publish("PLC/PLC1/TANQUE/PID/KD",plc.getMem('MR90'))

```

```
ciclo = 0
```

```
#print("rc: " + str(rc))
```

```
plc.plc.disconnect() #Desconexión de la comunicación con el PLC
```

RASPBERRY QUE RECIBE DATOS PARA LA PÁGINA WEB

```

#include <mysql/mysql.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "MQTTClient.h"
#define ADDRESS      "tcp://192.168.0.13:8883"
#define CLIENTID     "SERVER"
#define TOPIC        "PLC/PLC1/#"
#define PAYLOAD      "Hello C....."
#define QOS          1
#define TIMEOUT      10000L

MYSQL *conn;
MYSQL_RES *res;
MYSQL_ROW row;
char *server = "127.0.0.1"; //El broker se encuentra en el mismo PC
char *user = "root";
char *password = "proyecto"; /* set me first */
char *database = "PROYECTO";
char buf[1024]={};

```

```

char *query_string;
int insertar=0;
char mensaje[200] = {0x00}; // Mensaje de mqtt

volatile MQTTClient_deliveryToken deliveredtoken;
void delivered(void *context, MQTTClient_deliveryToken dt)
{
    printf("Message with token value %d delivery confirmed\n", dt);
    deliveredtoken = dt;
}

void consulta(){
res=mysql_use_result(conn);
    while ((row = mysql_fetch_row(res)) != NULL){ // Recorrer la variable res con
    todos los registros obtenidos para su uso
        if (strcmp(row[0],mensaje) != 0){
            insertar=1;
        }
    }
    if(insertar==1){
        sprintf(buf, query_string,mensaje);
        if (mysql_query(conn,buf)){
            fprintf(stderr, "Error al insertar\n");
        }
    }
}

int msgarrvd(void *context, char *topicName, int topicLen, MQTTClient_message
*message)
{
    int i;
    char* payloadptr;
    insertar=0;
    printf("Message arrived\n");
    printf("topic: %s\n", topicName);
    printf("  message: ");
    payloadptr = message->payload;

    for(i=0; i<message->payloadlen; i++)
    {
        mensaje[i] = putchar(*payloadptr++);
    }
    putchar('\n');

    if(strcmp("PLC/PLC1/TANQUE/TANQUE/SET_POINT",topicName)==0){
        mysql_query(conn, "SELECT `VALOR` FROM `TANQUE_SETPOINT` ORDER BY `ID` DESC
LIMIT 1");
        query_string= "INSERT INTO `TANQUE_SETPOINT` (`ID`, `VALOR`,`USUARIO`,
`FECHA`) VALUES (NULL, %s,'PLC1', CURRENT_TIMESTAMP)" ;
    }
}

```

```

        consulta();
    }

if(strcmp("PLC/PLC1/TANQUE/TANQUE/VALOR_REAL",topicName)==0){
    mysql_query(conn, "SELECT `VALOR` FROM `TANQUE_REAL` ORDER BY `ID` DESC
LIMIT 1");
    query_string= "INSERT INTO `TANQUE_REAL` (`ID`, `VALOR`, `FECHA`) VALUES
(NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/TIEMPO/PRODUCCION",topicName)==0){
    mysql_query(conn, "SELECT `VALOR` FROM `TIEMPO_PRODUCCION` ORDER BY `ID`
DESC LIMIT 1");
    query_string= "INSERT INTO `TIEMPO_PRODUCCION` (`ID`, `VALOR`, `FECHA`)
VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/PRODUCCION/PRODUCTOS_HORA",topicName)==0){

    mysql_query(conn, "SELECT `UNIDADES` FROM `PRODUCTOS_HORA` ORDER BY `ID`
DESC LIMIT 1");
    query_string= "INSERT INTO `PRODUCTOS_HORA` (`ID`, `UNIDADES`, `FECHA`)
VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/TIEMPO/BANDA_1",topicName)==0){
    mysql_query(conn, "SELECT `VALOR` FROM `TIEMPO_BANDA1` ORDER BY `ID` DESC
LIMIT 1");
    query_string= "INSERT INTO `TIEMPO_BANDA1` (`ID`, `VALOR`, `FECHA`) VALUES
(NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/PRODUCCION/BANDA_1",topicName)==0){
    mysql_query(conn, "SELECT `UNIDADES` FROM `PRODUCCION_BANDA1` ORDER BY `ID`
DESC LIMIT 1");
    query_string= "INSERT INTO `PRODUCCION_BANDA1` (`ID`, `UNIDADES`, `FECHA`)
VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/TIEMPO/BANDA_2",topicName)==0){
    mysql_query(conn, "SELECT `VALOR` FROM `TIEMPO_BANDA2` ORDER BY `ID` DESC
LIMIT 1");
    query_string= "INSERT INTO `TIEMPO_BANDA2` (`ID`, `VALOR`, `FECHA`) VALUES
(NULL, %s, CURRENT_TIMESTAMP)" ;
}

```

```

        consulta();
    }

if(strcmp("PLC/PLC1/ENSAMBLAJE/PRODUCCION/BANDA_2",topicName)==0){
    mysql_query(conn, "SELECT `UNIDADES` FROM `PRODUCCION_BANDA2` ORDER BY `ID`
DESC LIMIT 1");
    query_string= "INSERT INTO `PRODUCCION_BANDA2` (`ID`, `UNIDADES`, `FECHA`)
VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/TIEMPO/POSICIONADOR_1",topicName)==0){
    mysql_query(conn, "SELECT `VALOR` FROM `TIEMPO_POSICIONADOR1` ORDER BY `ID`
DESC LIMIT 1");
    query_string= "INSERT INTO `TIEMPO_POSICIONADOR1` (`ID`, `VALOR`, `FECHA`)
VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/PRODUCCION/POSICIONADOR_1",topicName)==0){
    mysql_query(conn, "SELECT `UNIDADES` FROM `PRODUCCION_POSICIONADOR1` ORDER
BY `ID` DESC LIMIT 1");
    query_string= "INSERT INTO `PRODUCCION_POSICIONADOR1` (`ID`, `UNIDADES`,
`FECHA`) VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/TIEMPO/POSICIONADOR_2",topicName)==0){
    mysql_query(conn, "SELECT `VALOR` FROM `TIEMPO_POSICIONADOR2` ORDER BY `ID`
DESC LIMIT 1");
    query_string= "INSERT INTO `TIEMPO_POSICIONADOR2` (`ID`, `VALOR`, `FECHA`)
VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/PRODUCCION/POSICIONADOR_2",topicName)==0){
    mysql_query(conn, "SELECT `UNIDADES` FROM `PRODUCCION_POSICIONADOR2` ORDER
BY `ID` DESC LIMIT 1");
    query_string= "INSERT INTO `PRODUCCION_POSICIONADOR2` (`ID`, `UNIDADES`,
`FECHA`) VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/TIEMPO/BRAZO",topicName)==0){
    mysql_query(conn, "SELECT `VALOR` FROM `TIEMPO_BRAZO` ORDER BY `ID` DESC
LIMIT 1");
    query_string= "INSERT INTO `TIEMPO_BRAZO` (`ID`, `VALOR`, `FECHA`) VALUES
(NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

```

```

    }

if(strcmp("PLC/PLC1/ENSAMBLAJE/PRODUCCION/BRAZO",topicName)==0){
    mysql_query(conn, "SELECT `UNIDADES` FROM `PRODUCCION_BRAZO` ORDER BY `ID`
DESC LIMIT 1");
    query_string= "INSERT INTO `PRODUCCION_BRAZO` (`ID`, `UNIDADES`, `FECHA`)
VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

    if(strcmp("PLC/PLC1/ENSAMBLAJE/TIEMPO/BANDA_CAJA",topicName)==0){
        mysql_query(conn, "SELECT `VALOR` FROM `TIEMPO_BANDACAJA` ORDER BY `ID` DESC
LIMIT 1");
        query_string= "INSERT INTO `TIEMPO_BANDACAJA` (`ID`, `VALOR`, `FECHA`)
VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
        consulta();
    }

if(strcmp("PLC/PLC1/ENSAMBLAJE/PRODUCCION/BANDA_CAJA",topicName)==0){ //MW70
    mysql_query(conn, "SELECT `UNIDADES` FROM `PRODUCCION_BANDACAJA` ORDER BY
`ID` DESC LIMIT 1");
    query_string= "INSERT INTO `PRODUCCION_BANDACAJA` (`ID`, `UNIDADES`,
`FECHA`) VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/PRODUCCION/PRODUCTOS_POR_CAJA",topicName)==0){
//MW72
    mysql_query(conn, "SELECT `UNIDADES` FROM `PRODUCTOS_CAJA` ORDER BY `ID`
DESC LIMIT 1");
    query_string= "INSERT INTO `PRODUCTOS_CAJA` (`ID`, `UNIDADES`, `FECHA`)
VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/PRODUCCION/PRODUCTOS",topicName)==0){ //MW74
    mysql_query(conn, "SELECT `UNIDADES` FROM `PRODUCTOS_TERMINADOS` ORDER BY
`ID` DESC LIMIT 1");
    query_string= "INSERT INTO `PRODUCTOS_TERMINADOS` (`ID`, `UNIDADES`,
`FECHA`) VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

if(strcmp("PLC/PLC1/ENSAMBLAJE/PRODUCCION/CAJAS",topicName)==0){ //MW76
    mysql_query(conn, "SELECT `UNIDADES` FROM `CAJAS_EMPACADAS` ORDER BY `ID`
DESC LIMIT 1");
    query_string= "INSERT INTO `CAJAS_EMPACADAS` (`ID`, `UNIDADES`, `FECHA`)
VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
    consulta();
}

```

```

    }

    if(strcmp("PLC/PLC1/ENSAMBLAJE/PRODUCCION/PRODUCTOS_BASE",topicName)==0){ //MW78
        mysql_query(conn, "SELECT `UNIDADES` FROM `PRODUCTOS_BASE` ORDER BY `ID`
DESC LIMIT 1");
        query_string= "INSERT INTO `PRODUCTOS_BASE` (`ID`, `UNIDADES`, `FECHA`)
VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
        consulta();
    }

    if(strcmp("PLC/PLC1/ENSAMBLAJE/PRODUCCION/PRODUCTOS_LIQ",topicName)==0){ //MW80
        mysql_query(conn, "SELECT `UNIDADES` FROM `PRODUCTOS_LIQ` ORDER BY `ID` DESC
LIMIT 1");
        query_string= "INSERT INTO `PRODUCTOS_LIQ` (`ID`, `UNIDADES`, `FECHA`)
VALUES (NULL, %s, CURRENT_TIMESTAMP)" ;
        consulta();
    }

    if(strcmp("PLC/PLC1/TANQUE/PID/KP",topicName)==0){ //MW84
        mysql_query(conn, "SELECT `KP` FROM `KP` ORDER BY `ID` DESC LIMIT 1");
        query_string= "INSERT INTO `KP` (`ID`, `KP`, `FECHA`) VALUES (NULL, %s,
CURRENT_TIMESTAMP)" ;
        consulta();
    }

    if(strcmp("PLC/PLC1/TANQUE/PID/KI",topicName)==0){ //MW88
        mysql_query(conn, "SELECT `KI` FROM `KI` ORDER BY `ID` DESC LIMIT 1");
        query_string= "INSERT INTO `KI` (`ID`, `KI`, `FECHA`) VALUES (NULL, %s,
CURRENT_TIMESTAMP)" ;
        consulta();
    }

    if(strcmp("PLC/PLC1/TANQUE/PID/KD",topicName)==0){ //MW92
        mysql_query(conn, "SELECT `KD` FROM `KD` ORDER BY `ID` DESC LIMIT 1");
        query_string= "INSERT INTO `KD` (`ID`, `KD`, `FECHA`) VALUES (NULL, %s,
CURRENT_TIMESTAMP)" ;
        consulta();
    }

    memset(mensaje,'\0',200); //Reiniciar la variable mensaje
    MQTTClient_freeMessage(&message);
    MQTTClient_free(topicName);
    return 1;
}

void connlost(void *context, char *cause)
{
    printf("\nConnection lost\n");
}

```

```

    printf("    cause: %s\n", cause);
}

```

```

int main(int argc, char* argv[])

```

```

{

```

MySQL

```

    conn = mysql_init(NULL);

    /* Connect to database */
    if (!mysql_real_connect(conn, server,
        user, password, database, 0, NULL, 0)) {
        fprintf(stderr, "%s\n", mysql_error(conn));
        //      exit(1);
    }
    else{
        printf("Conectado a MySQL \n");
    }
}

```

MQTT

```

MQTTClient client;
MQTTClient_connectOptions conn_opts= MQTTClient_connectOptions_initializer;
int rc;
int ch;
MQTTClient_create(&client, ADDRESS, CLIENTID,
MQTTCLIENT_PERSISTENCE_NONE, NULL);
conn_opts.keepAliveInterval = 20;
conn_opts.cleansession = 1;
MQTTClient_setCallbacks(client, NULL, connlost, msgarrvd, delivered);

if ((rc = MQTTClient_connect(client, &conn_opts)) != MQTTCLIENT_SUCCESS)
{
    printf("Fallo al conectarse, código de retorno %d\n", rc);
    exit(EXIT_FAILURE);
}
printf("Suscribiendose al topic %s\npara el cliente %s usando QoS%d\n\n"
"Presione Q<Enter> para detener el programa\n\n", TOPIC, CLIENTID, QOS);
MQTTClient_subscribe(client, TOPIC, QOS);
do
{
    ch = getchar();
} while(ch!='Q' && ch != 'q');
MQTTClient_disconnect(client, 10000);
MQTTClient_destroy(&client);
return rc;
}

```

SERVIDOR PÁGINA WEB

```
<!doctype html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1.0"&gt;>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

<title>Panel del Operario</title>

<script src="https://cdnjs.cloudflare.com/ajax/libs/paho-mqtt/1.0.1/mqttws31.min.js"
type="text/javascript"></script>
<script type="text/javascript" language="javascript">

var mqtt;
var reconnectTimeout = 2000;
var host = "186.33.173.57"; //DIRECCION IP
var port = 9001;
var ruedas_banda1=1;
var ruedas_banda2=0;
var ensamblada=0; // Cuando no hay ninguna unidad ensamblada previamente.
var ensamblada_salida=0; // Para desaparecer la caja ya ensamblada de la salida
var pos_c1bx=0;//Posicion de la caja de la banda 1 para el ensamblaje
var valor_start=0; //0 si el sistema está apagado, 1 si esta encendido
var usuario="<?php echo($_POST['usuario'])?>";
if(usuario==""){
    alert("Por favor ingrese sus credenciales");
    location.href = "../index.html";
}

function onConnect(){
    alert("Conectado");
    mqtt.subscribe("PLC/PLC1TOWEB/ENSAMBLAJE/BRAZO/BRAZO_ENSAMBLE"); //CUANDO SE
ENSAMBLA
    mqtt.subscribe("PLC/PLC1TOWEB/ENSAMBLAJE/BANDAS/BANDA_1");
    mqtt.subscribe("PLC/PLC1TOWEB/ENSAMBLAJE/BANDAS/BANDA_2");
    mqtt.subscribe("PLC/PLC1TOWEB/ENSAMBLAJE/SENSOR/SALIDA");
    mqtt.subscribe("PLC/PLC1/ENSAMBLAJE/CONTROL/START");

    message = new Paho.MQTT.Message("Hola");
    message.destinationName = "PLC/PLC1";
    mqtt.send(message);
}

function onFailure(message){
    alert("Connection Attempt tp Host "+host+" Failed");
    setTimeout(MQTTconnect,reconnectTimeout);
```

```

}

function onMessageArrived(msg){
    out_msg="Message received "+msg.payloadString+"<br>";
    out_msg=out_msg+"Message received Topic "+msg.destinationName;
    //alert(out_msg);
    if(msg.destinationName=="PLC/PLC1/ENSAMBLAJE/CONTROL/START" ){ //##
ensamblada_salida==1
        if(msg.payloadString=="True"){
            valor_start=1;
        }
        if(msg.payloadString=="False"){
            valor_start=0;
        }
    }

    if(msg.destinationName=="PLC/PLC1TOWEB/ENSAMBLAJE/SENSOR/SALIDA" ){ //##
ensamblada_salida==1
        caja_salida.style.visibility='hidden';
        // salida();
    }
    if(msg.destinationName=="PLC/PLC1TOWEB/ENSAMBLAJE/BRAZO/BRAZO_ENSAMBLE" &&
msg.payloadString=="True"){
        ruedas_banda1=0;
        Ensamble();
    }

    if(msg.destinationName=="PLC/PLC1TOWEB/ENSAMBLAJE/BANDAS/BANDA_1" &&
msg.payloadString=="True" && ensamblada==0){
        ruedas_banda1=1;
        banda1();
    }
    if(msg.destinationName=="PLC/PLC1TOWEB/ENSAMBLAJE/BANDAS/BANDA_1" &&
msg.payloadString=="True" && ensamblada==1){
        ruedas_banda1=1;
        banda1_completa();
    }
    if(msg.destinationName=="PLC/PLC1TOWEB/ENSAMBLAJE/BANDAS/BANDA_1" &&
msg.payloadString=="False" ){
        ruedas_banda1=0;
    }

    if(msg.destinationName=="PLC/PLC1TOWEB/ENSAMBLAJE/BANDAS/BANDA_2" &&
msg.payloadString=="True"){
        banda2();
    }
}
function MQTTconnect(){

```

```

    alert("Conectando a broker MQTT Mosquitto");
    mqtt = new Paho.MQTT.Client(host,port,usuario);
    //document.write("connecting to "+ host);
    var options = {
        timeout: 3,
        onSuccess: onConnect,
        onFailure: onFailure,
    };
    mqtt.onMessageArrived = onMessageArrived

    mqtt.connect(options); //connect

}
MQTTconnect();

</script>

<!--usado para el tanque -->
<link rel="stylesheet" href="css/industrial.css">
<!--<link rel="stylesheet" href="css/introjs.min.css">-->
<link rel="stylesheet" href="css/foundation.css">

</head>

    <style>
body {
    background-color: #F1F1F1; /*#F1F1F1*/
    height: 100%;
}
#botoneras {
    width: 90%;
    height: 130px;
    position: relative;
    top: 0px;
}
#banda {
    width: 90%;
    height: auto;
    position: relative;
    top: 50px;
}
#TANQUE_CONT {
    width: 90%;
    height: 260px;
    position: relative;
    top: 0px;
}
#tanque {
    width:290px;

```

```

    height: 350px;
    position: relative;
    top: 0px;
}
#selector_tanque {
    width:30%;
    height: 160px;
    position: relative;
    top: 0px;
    float: left;
}
#envio_tanque {
    width:20%;
    height: 50px;
    position: relative;
    top: 0%;
    float: left;
}
#valores_tanque {
    width:50%;
    height: 50px;
    position: relative;
    top: 0%;
    float: left;
}
#datos_ensamblaje {
    width: 90%;
    height: auto;
    position: relative;
    top: 260px;
}
#datos_produccion {
    width: 90%;
    height: 140px;
    position: relative;
    top: 290px;
}
#PID {
    width: 90%;
    height: 130px;
    position: relative;
    top: 40px;
}
#contenedor_izq {
    width: 49%;
    height: auto;
    position: relative;
    top: 10px;
    float: left;
}

```

```

        /*border-color: cadetblue;
        border-width: medium;
        border-radius:15px;
        border-top-style: solid;
        border-right-style: solid;
        border-bottom-style: solid;
        border-left-style: solid;*/
    }
    #contenedor_dere {
        width: 49%;
        height: auto;
        position: relative;
        top: 10px;
        float: left;
    }
    #master {
        width:auto;
        height: auto;
        align-self: center;
        align-content: center;
    }
    #titulo {
        border-color: cadetblue;
        border-width: medium;
        border-radius:15px;
        border-top-style: solid;
        border-right-style: solid;
        border-bottom-style: solid;
        border-left-style: solid;
    }
    #opciones {
        border-color:aliceblue;
        border-width: medium;
        border-radius:15px;
        border-top-style: solid;
        border-right-style: solid;
        border-bottom-style: solid;
        border-left-style: solid;
    }
        /**CCS para determinar slide vertical*/
    #vertical-slider {
        height: 250px;
        margin-left: 30px;
    }

        /*creacion de boton personalizado*/
    .boton_personalizado{
        text-decoration: none;
        padding: 10px;

```

```

font-weight: 600;
font-size: 15px;
color: #ffffff;
background-color:#888484;
border-radius: 6px;
border: 2px solid #3D3939;
}
.boton_personalizado:hover{
color: #1883ba;
background-color: #ffffff;
}

.boton_actual{
text-decoration: none;
padding: 10px;
font-weight: 600;
font-size: 15px;
color: #888484;
background-color:#ffffff;
border-radius: 6px;
border: 2px solid #3D3939 ;
}
</style>
<body>

<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>

<div id="master">

    <div id="opciones">

        <table width="100%">
            <tr>
                <th align="left" width="20%">
                    <div align="center" >
                        <h4>Estado del Proceso</h4>
                        <div class="industrial led size two">
                            <div class="meter green"></div>
                            <script src="http://code.jquery.com/jquery-
1.9.1.min.js"></script>
                            <script src="js/industrial.js"></script>
                            <script>
                                $(document).ready(function() {
                                    setTimeout(function() {
                                        $(".industrial").industrial({});
                                    },250);
                                }

```

```

                setInterval(function(){
                    $(".industrial.led").each(function() {
                        $(this).industrial(valor_start);
                    });
                },1000);
            });
        </script>
    </div>
</div>
</th>
<th><h1 align="center">UNIVERSIDAD TÉCNICA DE AMBATO</h1>
<h2 align="center">MAESTRÍA EN AUTOMATIZACIÓN Y SISTEMAS DE CONTROL</h2>
<h2 align="center">PANEL DE MONITOREO</h2></th>
<th width="20%" align="right">

<button class="boton_actual">USUARIO</button>
<a class="boton_personalizado" href="ingeniero.php">INGENIERO</a>
<p> </p>
<a class="boton_personalizado" href="../index.html">CERRAR
SESION</a>

</th>
</tr>
</table>
</div>

<div id="contenedor_izq" align="center" >
    <div id="titulo_1" align="center">
        <h2 align="center">Proceso Ensamblaje</h2>
    </div>

    <div id="botoneras">
        </img>
        </img>
        </img>
        </img>
    </div>

    <div id ="banda">

```

```

    </img>
    </img>

```

```

    </img>
    </img>

```

```

</div>

```

```

<div id="datos_ensamblaje">
    <table style="width:90%; border:1px solid #7B7979; ">
        <tr>
            <th style="background:#2ba6cb; color: aliceblue "> Número de
Liquidos Ingresadas</th>
            <th style="width:30%; height:100%;border:1px solid #7B7979;
background:#fff; color:#000"><div id="n_productos_liq" align="center" ></div></th>
        </tr>
        <tr>
            <th style="background:#2ba6cb; color: aliceblue ">Número de
Bases Ingresadas </th>
            <th style="width:30%; height:100%;border:1px solid #7B7979;
background:#fff; color:#000"><div id="n_productos_base" align="center" ></div></th>
        </tr>
        <tr>
            <th style="background:#2ba6cb; color: aliceblue "> Cantidad de
productos terminados</th>

```

```

                <th style="width:30%; height:100%;border:1px solid #7B7979;
background:#fff; color:#000"><div id="n_productos_terminados" align="center"
></div></th>
            </tr>
            <tr>
                <th style="background:#2ba6cb; color: aliceblue ">Cantidad de
cajas empacadas </th>
                <th style="width:30%; height:100%;border:1px solid #7B7979;
background:#fff; color:#000"><div id="n_cajas_empacadas" align="center" ></div></th>
            </tr>
        </table>
    </div>
    <div id="datos_produccion">
        <p align="left"><strong>PRODUCCIÓN</strong></p>
        <table style="width:90%; border:1px solid #7B7979; ">
            <tr>
                <th style="background:#2ba6cb; color: aliceblue "> Productos por
hora</th>
                <th style="width:30%; height:100%;border:1px solid #7B7979;
background:#fff; color:#000"><div id="n_productos_hora" align="center" ></div></th>
            </tr>
            <tr>
                <th style="background:#2ba6cb; color: aliceblue ">Productos por
caja </th>
                <th style="width:30%; height:100%;border:1px solid #7B7979;
background:#fff; color:#000"><div id="n_productos_caja" align="center" ></div></th>
            </tr>
        </table>
    </div>
</div>
<div id="contenedor_dere" align="center">

    <div id="titulo_1" align="center">
        <h2 align="center">Proceso de Control de Tanque</h2>
    </div>

    <div id="TANQUE_CONT" align="center">

        <div id="envio_tanque" >
            <h4>Envio Set Point</h4>
            <input type="text" id="set_point_ingreso" maxlength="3"
style="width: 80%">
            <button style=" padding-top: 0.35em; padding-right: 1.5em; padding-
bottom: 0.35em; padding-left: 1.5em;" onClick="set_point_enviar()"></img></button>

        </div>
        <div id= "selector_tanque">
            <!-- Para usarse con el slide-->

```

```

        <script
src="http://ajax.googleapis.com/ajax/libs/jqueryui/1.10.4/jquery-
ui.min.js"></script>
        <link rel="stylesheet" type="text/css"
href="http://code.jquery.com/ui/1.10.4/themes/flick/jquery-ui.css"/>
        <link href="jQuery-ui-Slider-Pips-master/src/css/jquery-ui-slider-
pips.css" rel="stylesheet" type="text/css">
        <script src="jQuery-ui-Slider-Pips-master/src/js/jquery-ui-slider-
pips.js"></script>
        <!-- fin importacion slide -->

        <div class="slider">
        <script>
        $(".slider")

        .slider({
            min: 0,
            max: 100,
            orientation: "vertical"

        })

        .slider("pips", {
            rest: "label",
            step: "20",

        })
        .slider("float")
        .on("slidechange", function(e,ui) {
            //CUANDO SE INGRESA UN VALOR ATRAVEZ DEL SLIDE
            message = new Paho.MQTT.Message((ui.value).toString());
            message.destinationName="PLC/PLC1/CONTROL/TANQUE/SET_POINT";
            mqtt.send(message);

        });
        </script>

        </div>
</div>

<div id="valores_tanque">
    <table width="90%">
        <tr>
            <th width="50%"><strong>Valor Real</strong></th>
            <th width="50%"><strong>Set Point</strong></th>
        </tr>
        <tr>

```

```

        <th style="width:50%; height:100%;border:1px solid #7B7979;
background:#fff; color:#000"><div id="valor_real" align="center" ></div></th>
        <th style="width:50%; height:100%;border:1px solid #7B7979;
background:#fff; color:#000"><div id="set_point" align="center" ></div></th>
    </tr>
</table>
</div>
<div style="width: 99%; float: left"; align="center" >
    <div id= "tanque" align="center" style="background-
image:url(imagenes/tanque.png); background-repeat: no-repeat; background-size:cover"
>

        <div style="height: 10%"> </div>

        <div class="industrial tank size two" align="center" >
            <span class="ticks" data-amount="5" data-scale-
freq="0.5"></span>
            <div class="space"></div>
            <div class="meter"></div>
        </div>

<script src="http://code.jquery.com/jquery-1.9.1.min.js"></script>
<script src="js/industrial.js"></script>

<script>

$(document).ready(function() {
    setTimeout(function() {
        $(".industrial").industrial({});
    },250);

    setInterval(function() {

        src="https://ajax.googleapis.com/ajax/libs/jquery/1.8.2/jque
ry.min.js";

        $.get( "mysql_php/consulta_operario.php?Consultar=7",
function( UltimosDatosTR ) {
            var tanque_real=parseFloat(UltimosDatosTR[0].VALOR);

            document.getElementById('valor_real').innerHTML=tanque_r
eal;

            $(".industrial.tank, .industrial.thermometer,
.industrial.gauge").each(function() {

                $(this).industrial(Math.floor(tanque_real))

            });

```

```

        });
    }, 1000);
});

</script>
<script src="js/intro.min.js"></script>
</div>
</div>

</div>

<div id="PID" align="center">

    <table style="width:90%; ">
        <tr>
            <th><strong>CONSTANTES PID</strong></th>
        </tr>
        <tr>
            <th style="background:#2ba6cb; color: aliceblue ; border:1px
solid #7B7979;">KP</th>
            <th style="background:#2ba6cb; color: aliceblue; border:1px
solid #7B7979; "> KI</th>
            <th style="background:#2ba6cb; color: aliceblue ; border:1px
solid #7B7979;"> KD</th>
        </tr>
        <tr>
            <th style="width:30%; height:100%;border:1px solid #7B7979;
background:#fff; color:#000"><div id="KP" align="center" ></div></th>
            <th style="width:30%; height:100%;border:1px solid #7B7979;
background:#fff; color:#000"><div id="KI" align="center" ></div></th>
            <th style="width:30%; height:100%;border:1px solid #7B7979;
background:#fff; color:#000"><div id="KD" align="center" ></div></th>
        </tr>
    </table>
</div>
</div>

<script type="text/javascript">
src="https://ajax.googleapis.com/ajax/libs/jquery/1.8.2/jquery.min.js";

```

Funiones de los Botones

```

/*function boton () { // Gauge
var text_vtanque = document.getElementById("valor_tanque").value;
var integer_vtanque = parseFloat(text_vtanque, 10);
message = new Paho.MQTT.Message(text_vtanque);
message.destinationName = "PLC/PLC1/CONTROL/GENERAL/SET_POINT";

```

```

mqtt.send(message);
//chart1.series[0].setData([integer_vtanque]);
//$.get("mysql_php/datos_tanque_insert.php", { valor:integer_vtanque });
}
*/

function start(){
    var usuario="operariogg";
    //$.get("mysql_php/start.php", { valor:usuario }); Para insertar en BD
    message = new Paho.MQTT.Message("START");
    message.destinationName = "PLC/PLC1/CONTROL/GENERAL/START_WEB";
    mqtt.send(message);
}
function stop(){
    message = new Paho.MQTT.Message("Detener Proceso");
    message.destinationName = "PLC/PLC1/CONTROL/GENERAL/STOP_WEB";
    mqtt.send(message);
}
function emergencia(){
    message = new Paho.MQTT.Message("PARO EMERGENCIA Proceso");
    message.destinationName = "PLC/PLC1/CONTROL/GENERAL/PARO_WEB";
    mqtt.send(message);
}
function reset(){
    message = new Paho.MQTT.Message("RESET Proceso");
    message.destinationName = "PLC/PLC1/CONTROL/GENERAL/RESET_WEB";
    mqtt.send(message);
}

function set_point_enviar(){

    var text_setpoint = document.getElementById("set_point_ingreso").value;
    if(text_setpoint == ""){
        alert("El campo esta vacío, ingrese un número del 1 al 100");
        return false;
    }
    var int_setpoint = parseInt(text_setpoint);
    if(int_setpoint>=0 && int_setpoint<101){
        message = new Paho.MQTT.Message(text_setpoint);
        message.destinationName="PLC/PLC1/CONTROL/TANQUE/SET_POINT";
        mqtt.send(message);
    }else{
        alert("Valor fuera de rango 0-100");
    }
}

window.addEventListener("load", function() {

```

```

        document.getElementById("set_point_ingreso").addEventListener("keypress",
soloNumeros_setpoint, false);

});

function soloNumeros_setpoint(e){
    var key = window.event ? e.which : e.keyCode;

    if (key < 48 || key > 57) {
        e.preventDefault();
    }
}

var bandas = document.getElementById("bandas");
var posicion_banda;

// Bandas en movimiento
var rueda11 = document.getElementById("rueda11");
var rueda12 = document.getElementById("rueda12");
var rueda21 = document.getElementById("rueda21");
var rueda22 = document.getElementById("rueda22");
var rueda31 = document.getElementById("rueda31");
var rueda32 = document.getElementById("rueda32");

posicion_banda = bandas.getBoundingClientRect();
rueda11.style.left = ((posicion_banda.width*8)/100)+'px'; //10%del ancho del dibujo
banda
rueda11.style.top = ((posicion_banda.height*27)/100)+'px'; //19%del ancho del dibujo
banda
rueda12.style.left = ((posicion_banda.width*13)/100)+'px'; //10%del ancho del dibujo
banda
rueda12.style.top = ((posicion_banda.height*27)/100)+'px'; //19%del ancho del dibujo
banda
rueda21.style.left = ((posicion_banda.width*1)/100)+'px'; //10%del ancho del dibujo
banda
rueda21.style.top = ((posicion_banda.height*73)/100)+'px'; //19%del ancho del dibujo
banda
rueda22.style.left = ((posicion_banda.width*35)/100)+'px'; //10%del ancho del dibujo
banda
rueda22.style.top = ((posicion_banda.height*73)/100)+'px'; //19%del ancho del dibujo
banda
rueda31.style.left = ((posicion_banda.width*65)/100)+'px'; //10%del ancho del dibujo
banda
rueda31.style.top = ((posicion_banda.height*27)/100)+'px'; //19%del ancho del dibujo
banda
rueda32.style.left = ((posicion_banda.width*92)/100)+'px'; //10%del ancho del dibujo
banda
rueda32.style.top = ((posicion_banda.height*27)/100)+'px'; //19%del ancho del dibujo
banda

```

```

rueda11.style.visibility='visible';
rueda12.style.visibility='visible';
rueda21.style.visibility='visible';
rueda22.style.visibility='visible';
rueda31.style.visibility='visible';
rueda32.style.visibility='visible';

var pos_deg=40;

var id1 = setInterval(frame_ruedas, 10);

function frame_ruedas() {

    posicion_banda = bandas.getBoundingClientRect();
    rueda11.style.left = ((posicion_banda.width*8)/100)+'px'; //10%del ancho del
dibujo banda
    rueda11.style.top = ((posicion_banda.height*27)/100)+'px'; //19%del ancho
del dibujo banda
    rueda12.style.left = ((posicion_banda.width*13)/100)+'px'; //10%del ancho
del dibujo banda
    rueda12.style.top = ((posicion_banda.height*27)/100)+'px'; //19%del ancho
del dibujo banda
    rueda21.style.left = ((posicion_banda.width*1)/100)+'px'; //10%del ancho del
dibujo banda
    rueda21.style.top = ((posicion_banda.height*73)/100)+'px'; //19%del ancho
del dibujo banda
    rueda22.style.left = ((posicion_banda.width*35)/100)+'px'; //10%del ancho
del dibujo banda
    rueda22.style.top = ((posicion_banda.height*73)/100)+'px'; //19%del ancho
del dibujo banda
    rueda31.style.left = ((posicion_banda.width*65)/100)+'px'; //10%del ancho
del dibujo banda
    rueda31.style.top = ((posicion_banda.height*27)/100)+'px'; //19%del ancho
del dibujo banda
    rueda32.style.left = ((posicion_banda.width*92)/100)+'px'; //10%del ancho
del dibujo banda
    rueda32.style.top = ((posicion_banda.height*27)/100)+'px'; //19%del ancho
del dibujo banda
    pos_deg=pos_deg+2;
    //elem.style.top = pos + 'px';

    if(ruedas_banda2==1 || ruedas_banda1==1){
        rueda11.style.transform = "rotate(" + pos_deg + "deg)";
        rueda12.style.transform = "rotate(" + pos_deg + "deg)";
        rueda31.style.transform = "rotate(" + pos_deg + "deg)";
        rueda32.style.transform = "rotate(" + pos_deg + "deg)";
        rueda21.style.transform = "rotate(" + pos_deg + "deg)";
        rueda22.style.transform = "rotate(" + pos_deg + "deg)";
    }
}

```

```

    }
}
var caja1 = document.getElementById("caja1");
var caja2 = document.getElementById("caja2");

function banda1() {

    posicion_banda = bandas.getBoundingClientRect();
    caja1.style.left = ((posicion_banda.width*13)/100)+'px'; //10%del ancho del
dibujo banda
    caja1.style.top = ((posicion_banda.height*8)/100)+'px'; //19%del ancho del
dibujo banda
    //var pos_c1= ((posicion_banda.width*8)/100);
    var pos_c1=((posicion_banda.width*13)/100);
    var id = setInterval(frame, 10);
    function frame() {

        if (pos_c1>= (posicion_banda.width*0.54)) {
            clearInterval(id);
            caja1.style.visibility='visible';
            pos_c1bx=pos_c1;
        }
        else {
            pos_c1++;
            caja1.style.left = pos_c1 + 'px';
            caja1.style.visibility='visible';
        }
    }
}

function banda2() {
    posicion_banda = bandas.getBoundingClientRect();
    caja2.style.left = ((posicion_banda.width*1)/100)+'px'; //10%del ancho del
dibujo banda
    caja2.style.top = ((posicion_banda.height*53)/100)+'px'; //19%del ancho del
dibujo banda
    var pos_c2x= ((posicion_banda.width*8)/100);
    var id = setInterval(frame, 10);

    function frame() {

        if (pos_c2x >= (posicion_banda.width*0.47)) {
            clearInterval(id);
            caja2.style.visibility='visible';
        }
        else {
            pos_c2x++;
            //elem.style.top = pos + 'px';
            caja2.style.left = pos_c2x + 'px';
        }
    }
}

```

```

        caja2.style.visibility='visible';
    }
}
var caja1_ensamblaje=document.getElementById("caja1_ensamblaje");
var caja2_ensamblaje=document.getElementById("caja2_ensamblaje");
var caja_ensamblada= document.getElementById("caja_ensamblada");
var caja_salida= document.getElementById("caja_salida");

function Ensamble() {

    posicion_banda = bandas.getBoundingClientRect();

    //pos_c1bx=posicion_banda.width*0.54;
    if(pos_c1bx==0){
        pos_c1bx=posicion_banda.width*0.54;
    }

    caja1_ensamblaje.style.left = pos_c1bx+'px';
    caja1_ensamblaje.style.top = ((posicion_banda.height*8)/100)+'px';

    caja2_ensamblaje.style.left = (posicion_banda.width*0.47)+'px';
    caja2_ensamblaje.style.top = ((posicion_banda.height*53)/100)+'px';

    caja1_ensamblaje.style.visibility='visible';
    caja2_ensamblaje.style.visibility='visible';

    caja1.style.visibility='hidden';
    caja2.style.visibility='hidden';

    var posx = (posicion_banda.width*0.47);
    var posy= ((posicion_banda.height*53)/100);
    var id = setInterval(frame, 10);
    var id_down ;
    var id_left;

    function frame() { // sube la segunda caja

        if (posy<= ((posicion_banda.height*4)/100)) {
            clearInterval(id);
            id_left = setInterval(frameleft, 10);
            //elem.style.visibility='visible';
        }
        else {
            posy--;
            caja2_ensamblaje.style.top = posy + 'px';
            caja2_ensamblaje.style.visibility='visible';
        }
    }
}

```

```

function frameleft() { // mueve la segunda caja hacia la izquierda

    if (posx >= pos_c1bx) {

        id_down = setInterval(framedown, 10);
        clearInterval(id_left);
    }
    else {

        posx++;
        caja2_ensamblaje.style.left = posx + 'px';
        caja2_ensamblaje.style.visibility='visible';
    }
}

function framedown() { // mueva la caja 2 sobre la caja 1

    if (posy >= ((posicion_banda.height*8)/100)) {
        clearInterval(id_down);
        clearInterval(id);
        clearInterval(id_left);
        caja_ensamblada.style.top = ((posicion_banda.height*8)/100)+'px';
        caja_ensamblada.style.left = (posicion_banda.width*0.54)+'px';
        ensamblada = 1;
        caja_ensamblada.style.visibility='visible';
        caja2_ensamblaje.style.visibility='hidden';
        caja1_ensamblaje.style.visibility='hidden';
    }
    else {
        posy++;
        caja2_ensamblaje.style.top = posy + 'px';
        caja2_ensamblaje.style.visibility='visible';
    }
}

var pos_salidax;

function banda1_completa(){ // si esq ya se encuentra una caja ensamblada en la
banda 1

    posicion_banda = bandas.getBoundingClientRect();
    caja1.style.left = ((posicion_banda.width*13)/100)+'px'; //10%del ancho del
dibujo banda
    caja1.style.top = ((posicion_banda.height*8)/100)+'px'; //19%del ancho del
dibujo banda
    //var pos_c1= ((posicion_banda.width*8)/100);
    var pos_c1=((posicion_banda.width*13)/100);
    var pos_ce= (posicion_banda.width*0.54);

```

```

var id = setInterval(frame, 10);

function frame() { // movimiento de la banda

    if (pos_c1>= (posicion_banda.width*0.54)) {
        clearInterval(id);
        caja1.style.visibility='visible';
        pos_salidax=pos_ce;
        caja_salida.style.left= pos_salidax+'px';
        caja_salida.style.top = ((posicion_banda.height*8)/100)+'px';
        caja_salida.style.visibility='visible';
        caja_ensamblada.style.visibility='hidden';
        ensamblada_salida=1; // SI YA ESTA EN LA POSICION DE SALIDA
    }
    else {
        pos_c1++;
        pos_ce=pos_ce+(0.75);
        caja1.style.left = pos_c1 + 'px';
        caja_ensamblada.style.left = pos_ce + 'px';
        caja1.style.visibility='visible';
        caja_ensamblada.style.visibility='visible';
    }
}
}

function salida(){

    caja_salida.style.visibility='hidden';
}

```

Datos de Producción

```

ver_produccion();

function ver_produccion(){

    // variables
    var banda_a, banda_b, salida,cajas,kp,kd,ki, productos_liq, productos_base,
    productos_ter, cajas_empacadas,productos_hora,productos_caja, set_point;

    setInterval(function () { //permite actualizar la informacion cada cierto tiempo

////////usando el fichero consulta_operario.php////////////////////////////////////

        $.get( "mysql_php/consulta_operario.php?Consultar=1", function(
UltimosDatosPL ) {
            productos_liq=parseFloat(UltimosDatosPL[0].UNIDADES);
        });
    });
}

```

```

$.get( "mysql_php/consulta_operario.php?Consultar=2", function(
UltimosDatosPB ) {
    productos_base=parseFloat(UltimosDatosPB[0].UNIDADES);
});

$.get( "mysql_php/consulta_operario.php?Consultar=3", function(
UltimosDatosPT ) {
    productos_ter=parseFloat(UltimosDatosPT[0].UNIDADES);
});

$.get( "mysql_php/consulta_operario.php?Consultar=4", function(
UltimosDatosCE ) {
    cajas_empacadas=parseFloat(UltimosDatosCE[0].UNIDADES);
});

$.get( "mysql_php/consulta_operario.php?Consultar=5", function(
UltimosDatosPH ) {
    productos_hora=parseFloat(UltimosDatosPH[0].UNIDADES);
});

$.get( "mysql_php/consulta_operario.php?Consultar=6", function(
UltimosDatosPC ) {
    productos_caja=parseFloat(UltimosDatosPC[0].UNIDADES);
});

/// PID CONSULTA

$.get( "mysql_php/consulta_operario.php?Consultar=8", function(
UltimosDatosTS ) {
    set_point=parseFloat(UltimosDatosTS[0].VALOR);
});

$.get( "mysql_php/consulta_operario.php?Consultar=9", function(
UltimosDatosKP ) {
    kp=parseFloat(UltimosDatosKP[0].KP);
});
$.get( "mysql_php/consulta_operario.php?Consultar=10", function(
UltimosDatosKI ) {
    ki=parseFloat(UltimosDatosKI[0].KI);
});
$.get( "mysql_php/consulta_operario.php?Consultar=11", function(
UltimosDatosKD ) {
    kd=parseFloat(UltimosDatosKD[0].KD);
});

//// mostrar los valores en los elementod html
document.getElementById('n_productos_liq').innerHTML=productos_liq+"
unidades";

```

```
        document.getElementById('n_productos_base').innerHTML=productos_base+"
unidades";
        document.getElementById('n_productos_terminados').innerHTML=productos_ter+"
unidades";
        document.getElementById('n_cajas_empacadas').innerHTML=cajas_empacadas+"
unidades";
        document.getElementById('n_productos_hora').innerHTML=productos_hora+"
productos/hora";
        document.getElementById('n_productos_caja').innerHTML=productos_caja+"
productos/caja";
```

PID

```
        document.getElementById('set_point').innerHTML=set_point;
        document.getElementById('KP').innerHTML=kp;
        document.getElementById('KI').innerHTML=ki;
        document.getElementById('KD').innerHTML=kd;

    }, 1000); // cada segundo
}

</script>
</body>
</html>
```