



# ! POSGRADOS !

## MAESTRÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN

RPC-SO-30-NO.507-2019

OPCIÓN DE TITULACIÓN:  
PROYECTOS DE DESARROLLO

TEMA:

MONITOREO REMOTO DE VARIABLES  
PARA UNA RED INDUSTRIAL DE  
VARIADORES DE FRECUENCIA.  
APLICACIÓN BASADA EN IOT Y EL  
SISTEMA OPERATIVO MINDSPHERE

AUTOR:

PAÚL ESTEBAN ARPI COELLAR

DIRECTOR:

JULIO CÉSAR ZAMBRANO ABAD

CUENCA – ECUADOR  
2023



**Autor:**



**Paúl Esteban Arpi Coellar.**

Ingeniero Electrónico.

Candidato a Magíster en Electrónica y Automatización,  
Mención en Informática Industrial por la Universidad  
Politécnica Salesiana - Sede Cuenca.

parpi@est.ups.edu.ec

**Dirigido por:**



**Julio César Zambrano Abad**

Ingeniero Electrónico.

Magíster en Automatización y Control Industrial.

Doctor en Automática, Robótica e Informática  
Industrial.

jzambranoa@ups.edu.ec

Todos los derechos reservados.

Queda prohibida, salvo excepción prevista en la Ley, cualquier forma de reproducción, distribución, comunicación pública y transformación de esta obra para fines comerciales, sin contar con autorización de los titulares de propiedad intelectual. La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual. Se permite la libre difusión de este texto con fines académicos investigativos por cualquier medio, con la debida notificación a los autores.

**DERECHOS RESERVADOS**

©2023 Universidad Politécnica Salesiana.

CUENCA – ECUADOR – SUDAMÉRICA

PAÚL ESTEBAN ARPI COELLAR

Monitoreo remoto de variables para una red industrial de variadores de frecuencia.

Aplicación basada en IoT y el sistema operativo MindSphere

# Índice general

<b>Índice de Figuras</b>	<b>IV</b>
<b>Resumen</b>	<b>VI</b>
<b>Abstract</b>	<b>VI</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Descripción general del problema.....	3
1.2. Objetivos.....	4
1.2.1. Objetivo general.....	4
1.2.2. Objetivos específicos.....	4
1.3. Contribuciones.....	4
1.4. Organización del manuscrito.....	4
<b>2. Internet Industrial de las Cosas</b>	<b>6</b>
2.1. IoT: El Internet de las Cosas.....	7
2.2. Arquitecturas IoT.....	9
2.2.1. Arquitectura de 3 capas.....	9
2.2.2. Arquitectura Referencial de CISCO.....	10
2.2.3. Arquitectura IoT basada en la nube.....	13
2.3. IIoT: El Internet Industrial de las Cosas.....	14
2.3.1. Generalidades.....	14
2.3.2. Protocolos para el IIoT.....	17
2.3.3. Pasarelas Inteligentes.....	18
2.3.4. Plataformas para el IIoT.....	19
2.4. MindSphere.....	20
2.4.1. Uso de la información.....	22
2.4.2. Aplicaciones dentro de la plataforma.....	22

<b>3. Configuración de equipos e implementación de la red industrial</b>	<b>23</b>
3.1. Arquitectura de la red industrial .....	24
3.2. Implementación de la red industrial .....	25
3.2.1. Configuración de hardware .....	25
3.2.2. Programación en TIA Portal .....	27
3.2.3. Configuración de los Accionamientos.....	33
3.3. Configuración de la Pasarela IoT2040.....	37
3.3.1. Configuración de la pasarela IoT2040 .....	37
3.3.2. Flujo de Programa en Node-RED .....	38
<b>4. Implementación de la aplicación en Mindsphere y análisis de resultados</b>	<b>43</b>
4.1. Activos, aspectos y variables en MindSphere.....	44
4.2. Envío de datos desde Node-RED hacia MindSphere .....	50
4.3. Presentación de información en MindSphere .....	53
4.3.1. Visualización de series de tiempo .....	54
4.3.2. Visualización de valores actuales.....	56
4.3.3. Visualización del promedio y la desviación estándar .....	58
4.3.4. Visualización del tiempo de funcionamiento.....	59
4.4. Resultados y Discusión.....	60
<b>5. Conclusiones y trabajos futuros</b>	<b>67</b>
5.1. Conclusiones.....	67
5.2. Trabajos futuros .....	69

# Índice de Figuras

2.1. Arquitectura de 3 Capas [Mishra and Kertesz, 2020].....	10
2.2. Modelo de Referencia de IoT de CISCO [Cha et al., 2018] .....	11
2.3. SIMATIC IOT2040 [Siemens, 2016a] .....	18
2.4. Elementos de MindSphere [Siemens, 2016b].....	21
3.1. Diagrama de Implementación del Proyecto.....	24
3.2. Topología de la Red Industrial .....	26
3.3. Topología de la Red Industrial en TIA Portal.....	27
3.4. Acceso a la asignación de nombre para los variadores. ....	28
3.5. Asignación de nombre para los variadores.....	28
3.6. Bloque de datos con las variables a exportar.....	29
3.7. Diagrama de conexión entre el PLC y uno de los accionamientos. ....	30
3.8. Variables de programa. ....	31
3.9. Segmento de habilitación de los motores. ....	32
3.10. Segmento de control de sentido de giro.....	33
3.11. Bloque de control de velocidad el motor 1.....	34
3.12. Bloque de conteo del tiempo de funcionamiento del motor 2. ....	35
3.13. Carga de la configuración de hardware y del programa de control al PLC. ....	36
3.14. Pantalla de configuración IOP-2.....	37
3.15. Configuración del programa PuTTY para la conexión desde la PC a la pasarela. ....	38
3.16. Comandos para acceder a la configuración de la pasarela. ....	39
3.17. Configuración de conectividad de la pasarela. ....	39
3.18. Configuración de inicio de la pasarela.....	40
3.19. Comando para la instalación del nodo MindConnect.....	40
3.20. Comando para la instalación del nodo S7.....	40
3.21. Flujo de programa en Node-RED.....	40
3.22. Configuración inicial del nodo S7in.....	41

3.23. Configuración de conexión del nodo S7in. ....	41
3.24. Configuración de variables del nodo S7in (parte 1).....	42
4.1. Pantalla de inicio de MindSphere. ....	44
4.2. Inclusión de las variables del proceso industrial. ....	45
4.3. Aspectos creados dentro de Mindsphere. ....	46
4.4. Inclusión de las variables del proceso industrial. ....	47
4.5. Creación del activo de datos. ....	47
4.6. Creación del activo de conexión. ....	47
4.7. Información de seguridad de MindConnect.....	48
4.8. Inclusión de la fuente de datos. ....	48
4.9. Inclusión del punto de datos. ....	49
4.10. Ventana de mapeo de puntos de datos y variables. ....	49
4.11. Inicio del enlace de puntos de datos y variables. ....	50
4.12. Enlace de puntos de datos y variables exitoso.....	50
4.13. Resumen de enlace de puntos de datos y variables. ....	51
4.14. Lista de enlaces en MindSphere.....	51
4.15. Nodo Function. ....	52
4.16. Configuración del Agente hacia MindSphere. ....	53
4.17. Confirmación de configuración de conexión en MindSphere.....	54
4.18. Visual Flow Creator en MindSphere. ....	54
4.19. Configuración del panel de visualización. ....	55
4.20. Configuración del nodo INJECT. ....	55
4.21. Programa de series de tiempo.....	56
4.22. Adquisición de series de tiempo. ....	56
4.23. Selección de variables a visualizar. ....	57
4.24. Configuración de Visualización de series de tiempo. ....	57
4.25. Programa de visualización de valores actuales para el motor 1. ....	58
4.26. Lectura del último valor importado. ....	58
4.27. Filtrado del valor de interés.....	59
4.28. Visualización de velocidad actual. ....	59
4.29. Visualización de la velocidad promedio. ....	60
4.30. Visualización de la desviación estándar. ....	60
4.31. Programa de tiempo de funcionamiento.....	61
4.32. Configuración para la visualización del tiempo de funcionamiento. ....	61
4.33. Series de tiempo de las variables del motor 1.....	62
4.34. Series de tiempo de las variables del motor 2.....	63
4.35. Series de tiempo de las variables del motor 3.....	64

*ÍNDICE DE FIGURAS*

VI

4.36. Visualización de valores actuales, promedios y desviaciones estándar. ....	65
4.37. Panel de tiempos de funcionamiento. ....	66
5.1. Extracto de la visualización en <i>Visual Flow Creator</i> (Izquierda) y <i>Asset Manager</i> (Derecha). ....	70

# Resumen

El presente documento muestra los detalles de la implementación de un aplicación de monitoreo remoto de variables de una red industrial de variadores de frecuencia. Esta aplicación está basada en el Internet Industrial de las Cosas (IoT) y el sistema operativo de Mindsphere. La red industrial se implementa utilizando tres variadores SINAMICS G120 con las respectivas unidades de control, un Controlador Lógico Programable (PLC) S7-1500 y una pasarela inteligente IoT2040, la cual se encarga de transportar los datos a la nube. Toda esta tecnología pertenece al fabricante SIEMENS y se le ha integrado en una red utilizando el protocolo de comunicaciones Profinet. La aplicación de monitoreo remoto se construye en función de la información proporcionada por el Telegrama 20, que es un telegrama de control para accionamientos que operan bajo el sistema de comunicación Profinet. En la aplicación se puede visualizar de manera remota, las velocidades, torques, corrientes y potencias de los tres motores integrados en la red. Por otra parte, para propósitos de mantenimiento, la aplicación también incluye la visualización de los valores actuales, valores promedio y las desviaciones estándar de las variables del motor antes indicadas. Por otra parte, el panel de visualización también permite observar el tiempo de operación de los motores. Gracias al desarrollo de esta aplicación se ha podido explorar las facilidades que brinda Mindsphere para visualizar y administrar variables de procesos industriales en el contexto de la industria 4.0.



# Abstract

This document shows the details of the implementation of an application for remote monitoring of variables of an industrial network of variable frequency drives. This application is based on the Industrial Internet of Things (IoT) and the Mindsphere operating system. The industrial network is implemented using three SINAMICS G120 drives with the respective control units, a Programmable Logic Controller (PLC) S7-1500, and an IoT2040 intelligent gateway, which is responsible for transporting the data to the cloud. All this technology belongs to the manufacturer SIEMENS and has been integrated into a network using the Profinet communications protocol. The remote monitoring application is built based on the information provided by Telegram 20, which is a control telegram for drives operating under the Profinet communication system. In the application, it is possible to visualize remotely, the speeds, torques, currents, and powers of the three motors integrated in the network. Moreover, for maintenance purposes, the application also includes the visualization of the actual values, average values, and standard deviations of the above-mentioned motor variables. On the other hand, the display panel also allows for observing the operation time of the motors. Thanks to the development of this application, it has been possible to explore the facilities provided by Mindsphere to visualize and manage industrial process variables in the context of Industry 4.0.

# Capítulo 1

## Introducción

El Internet de las Cosas, o IoT (Internet of Things), ha trascendido su uso al área industrial para generar más beneficios en los procesos y aclarar los indicadores de producción en las empresas. El uso de IoT en la industria se le conoce como IIoT (Internet Industrial de las Cosas) y permite a las industrias tener más control y mejor información de sus procesos. Todo esto se logra mediante la adquisición de información de proceso por medio de sensores IoT que transmiten la información sin la interacción humana [Kumar and Smys, 2018] a plataformas de IIoT que reciben, almacenan, procesan y presentan esta información, pero sobre todo ayudan en la toma de decisiones en complemento con técnicas de inteligencia artificial dando mayor autonomía a los procesos. Esto no implica una desconexión completa del operario, ya que este también puede interactuar con las máquinas utilizando información precisa y oportuna [Chuquimarca et al., 2022], lo que conlleva a una mejor toma de decisiones para alimentar procesos de mantenimiento preventivo, incrementar la eficiencia operacional y reparar fallas de manera más precisa [Gavlas et al., 2018].

Las aplicaciones de IIoT van desde el uso de sensores básicos como muestra Katsikeas et al. [2017] hasta máquinas eléctricas de gran potencia, donde se dificulta mucho el acceso, como explica Kulkarni et al. [2019]. Un punto a destacar es que la información como voltaje, corriente, temperatura, frecuencia, factor de potencia y desempeño en tiempo real, permite planificar y realizar mantenimientos más específicos y con base a las necesidades reales de las máquinas, como explica Chacko and Deshmukh [2019] en su investigación. Normalmente, los mantenimientos se realizan con criterios muy aproximados a la realidad pero esta falta de precisión puede llevar a mantenimientos poco efectivos por su premura o a un mantenimiento

correctivo debido a la demora. El campo de implementación de IIoT es muy amplio y ha crecido de sobremanera debido a todas las funcionalidades que provee.

La empresa Amazon ha demostrado ser un gigante en la industria de desarrollo tecnológico e IIoT no es la excepción. Esta empresa ofrece la solución AWS IIoT para incursionar en la administración industrial del internet de las cosas, presentando importantes características como escalabilidad y procesamiento basado en Hadoop, con ventajas como la operación en tiempo real y herramientas analíticas muy poderosas que permiten, por ejemplo, la implementación de sistemas de gestión de micro redes como muestra [Marino et al. \[2022\]](#) en su trabajo. Este añade también ventajas sobre costos de uso de la plataforma que la vuelve bastante competitiva. Entre otras opciones también destaca la plataforma Watson de IBM, que presenta una alta integración de datos, buenos métodos de seguridad, además de las características básicas de una plataforma IIoT. [Marques and Pitarma \[2019\]](#) presenta las ventajas del uso de la plataforma de IBM en el desarrollo de un sistema de supervisión de calidad ambiental, que demuestra sus capacidades con respecto a recolección y acceso de datos, funcionalidades de blockchain, analítica y visualización.

Otras plataformas se extienden más allá en el uso de las potencialidades de la computación de borde como es el caso de Microsoft Azure IoT, el cual posee incluso aplicaciones de aprendizaje de máquina para clasificadores inteligentes mediante el uso de Microsoft Custom Vision, que, como evidencia el trabajo de [\[Ali and Ishak, 2020\]](#), serán de gran ayuda en la implementación de sistemas de vigilancia de futuras ciudades inteligentes. Una opción que está tomando fuerza en el ambiente del IIoT, es Google Cloud Platform IIoT, con características como comunicación M2M (Machine to Machine) por MQTT (MQ Telemetry Transport) y HTTP (Hypertext Transfer Protocol, o Protocolo de Transferencia de Hipertexto en español) y con la ventaja de que la información puede ser un gran conjunto de datos definidos arbitrariamente por el usuario. Esta última tiene un comportamiento similar a AWS con respecto a tiempos de servicio del Broker que publica los mensajes por segundo, lo cual lo vuelve bastante competitivo con respecto a la solución de IBM y Microsoft [\[Pierleoni et al., 2020\]](#).

## **1.1. Descripción general del problema**

La innovación es un aspecto esencial que debe incorporar cualquier industria para mejorar la eficiencia de sus procesos, desarrollar nuevos productos y servicios, y mejorar su capacidad para competir en el mercado. En el caso específico de Ecuador, la innovación resulta especialmente importante debido a varios factores. Actualmente, la producción en el país está centrada en productos y materias primas específicas como el petróleo, por lo que existe muy poco espacio para el desarrollo de otras industrias, lo cual ha limitado la diversificación económica del país.

En este contexto, la innovación puede ayudar a las empresas a mejorar su eficiencia y a reducir costos, lo que puede ser clave para mantenerse competitivos en el mercado global. Actualmente, todas las vías hacia la innovación industrial apuntan a la industria 4.0. Dentro de este contexto, el IIoT se presenta como una tecnología fundamental que da soporte a esta nueva era industrial.

Gracias a esta tecnología, hoy en día se puede mejorar la eficiencia y autonomía de los procesos industriales. El IIoT permite desarrollar sistemas de operación y monitoreo remoto, incorporando las tecnologías de la información a los procesos, permitiendo la generación de grandes bases de datos para el soporte en la toma de decisiones.

Actualmente, en nuestro país estas tecnologías no están siendo utilizadas o explotadas en toda su capacidad, ya sea por desconocimiento o por el temor natural al cambio. Bajo esta situación, existe la imperiosa necesidad de generar propuestas de innovación desde la academia hacia la industria.

La propuesta de este trabajo de titulación se sustenta en la necesidad actual a nivel industrial de monitorear de manera remota las variables de un proceso, para una toma de decisiones efectiva y oportuna. La aplicación de esta propuesta se centra específicamente en el monitoreo de las variables involucradas en la operación de motores industriales comandados por variadores de frecuencia que operan en red. A través de esta implementación práctica, se podrá monitorizar los valores de corriente, voltajes, potencias, velocidades, estados de operación, sentidos de giro, etc., de varios motores industriales en la plataforma de IIoT MindSphere.

## **1.2. Objetivos**

### **1.2.1. Objetivo general**

Desarrollar una aplicación de IoT en MindSphere para el monitoreo remoto de variables de una red industrial de variadores de frecuencia.

### **1.2.2. Objetivos específicos**

- Analizar el estado del arte sobre la recolección de variables y aplicaciones de IoT en el contexto industrial, para sentar las bases conceptuales y metodológicas para el desarrollo del proyecto.
- Definir los requerimientos de diseño mediante el análisis de las características de las aplicaciones desarrolladas en MindSphere para monitoreo de variables.
- Implementar a nivel de laboratorio una red industrial de variadores de frecuencia, para la integración con la plataforma basada en IoT.
- Implementar una aplicación de IoT en MindSphere mediante herramientas propias de dicho sistema operativo, para monitorizar las variables de la red industrial de variadores.
- Validar mediante pruebas de laboratorio la funcionalidad y desempeño de la aplicación desarrollada.

## **1.3. Contribuciones**

Este trabajo busca demostrar la utilidad de una aplicación de IIoT dentro de un proceso industrial con el fin de aplicar la analítica de datos a la información recibida y presentar datos que mejorarán la percepción de dicho proceso y podrán mostrar características de funcionamiento que de otra manera serían únicamente estimadas. El documento generado con esta investigación podrá servir de referencia para que las industrias del medio conozcan o potencien el uso de herramientas IoT para la operación y monitoreo de sus procesos.

## **1.4. Organización del manuscrito**

De aquí en adelante, se organiza de la siguiente manera. En el Capítulo 2, se abordan todas las definiciones previas y conceptos necesarios para el

completo entendimiento del tema de IIoT. En este capítulo se parte desde la definición formal y características del Internet de las Cosas o IoT, pasando por las generalidades del IIoT, sus características y taxonomía. Una vez definido esto, se explica las principales características de la plataforma de IIoT MindSphere y como se la puede usar. En el Capítulo 3, se presenta el diseño experimental a implementar y el desarrollo para devenir en los resultados y discusión de lo obtenido. Finalmente, en el Capítulo 4 se reportan algunas conclusiones y trabajos futuros.

## Capítulo 2

# Internet Industrial de las Cosas

En este capítulo se analizan los conceptos necesarios para entender a cabalidad lo que conlleva la aplicación de una solución de IIoT utilizando el sistema operativo MindSphere de Siemens. Se tendrá en cuenta un avance gradual con respecto al desarrollo tecnológico empezando por la aparición del Internet de las cosas o *IoT*, su adaptación para la industria y su concepción como industria 4.0 o *IIoT*. Una vez tratado esto, se procederá con una descripción de MindSphere como plataforma de servicio IIoT, junto con sus características de uso y las posibilidades de aplicación.

## 2.1. IoT: El Internet de las Cosas

Dentro de las tendencias de actuales de modernización, se han integrado diferentes tecnologías a tareas industriales y domésticas para volverlas más eficientes, más robustas o para tener un mayor control de estas. El Internet de las Cosas o IoT (Internet of Things), por sus siglas en inglés, ha mostrado un gran crecimiento en aplicaciones de seguridad, procesos industriales y monitores de actividad, entre muchas otras áreas de aplicación. IoT permite que dispositivos del mundo real se conecten al mundo digital para la recolección de información o para el control de dichos dispositivos en tiempo real. En ambos casos, se comparte que estas interacciones entre el mundo real y el mundo digital no requieran de la participación permanente del ser humano [Kumar and Smys, 2018]. Este acercamiento resulta muy simple tomando en cuenta las capacidades de desarrollo que presenta IoT frente a tecnologías anteriores. La evolución de IoT busca modificar y reestructurar el sector de comunicaciones M2M para tener procesos digitales y reales más rápidos donde los dispositivos se comportan de manera dinámica según lo que se le ha instruido, de manera autónoma. Si bien es importante la interacción con el ser humano, las características centrales de IoT permiten desarrollar nuevas aplicaciones donde se priorice la comunicación M2M, ya sea para un funcionamiento regular, como para la toma de decisiones.

IoT permite enlazar objetos reales con características digitales y de red, donde, cada objeto es identificado digitalmente. Por medio de la conexión a internet, se puede verificar de manera remota su estado, ubicación y demás características que van a depender del dispositivo de IoT bajo análisis. El despliegue de esta tecnología ha tenido un crecimiento muy rápido debido a sus características de simplicidad, a pesar de la crisis de escasez de chips que se vive desde el inicio de la pandemia en 2022 [Wu et al., 2021]. *International Data Corporation o IDC* estimó que el número de sensores IoT conectados se incrementará a 30 billones y el número de dispositivos conectados aumentará de 50 mil millones a 1 billón el final del año 2022 [Yu-Jie et al., 2022]. Este crecimiento acelerado denota que tanto la industria como las aplicaciones domiciliarias han adoptado dispositivos o sensores IoT en la gran mayoría de actividades. Un caso en particular que ha llamado la atención de los consumidores es la fácil disponibilidad de tecnología vestible (wearable) en formatos pequeños tales como relojes inteligentes y sensores de actividad deportiva a bajos precios, que entregan información detallada sobre frecuencia cardíaca, pasos, distancia recorrida, tiempo de actividad, etc. Los electrodomésticos no han sido la excepción al momento de integrar tecnología de IoT en modelos modernos que permiten funcionalidades nunca



antes vistas en refrigeradores, lavadoras, sistemas de aire acondicionado y calefacción, entre otros.

Un detalle a considerar es que los dispositivos o sensores IoT se encuentran permanentemente recolectando y transmitiendo información siguiendo reglas específicas sin la necesidad de una interacción humano a humano o humano a máquina. La información recolectada puede entregarse a manera de informes en tiempo real o de manera acumulada, dando la oportunidad de realizar análisis de históricos de datos y predicciones. Esto ha demostrado ser uno de los requerimientos principales al momento de implementar sistemas en las áreas de negocio y la industria, donde la toma de decisiones informada es de vital importancia.

El IoT se caracteriza por varios aspectos clave que lo distinguen de otras tecnologías. Estos aspectos han hecho que IoT sea una fuerza disruptiva en nuestra sociedad y en el ámbito industrial. A continuación, se presentan algunas de las características más destacadas del IoT:

- **Conectividad:** La característica primordial del IoT es la capacidad de conectar una amplia gama de dispositivos físicos a través de la red, permitiendo la comunicación y el intercambio de datos entre ellos.
- **Recolección de datos:** Los dispositivos IoT están equipados con sensores que capturan información del entorno. Estos sensores pueden medir variables, proporcionando información valiosa para su análisis y utilización en la toma de decisiones.
- **Comunicación y interoperabilidad:** El IoT utiliza diferentes protocolos y estándares, lo que facilita la interoperabilidad y la integración de sistemas. Por consiguiente, los dispositivos de diferentes fabricantes y tecnologías pueden interactuar y trabajar juntos de manera fluida.
- **Análisis de datos y toma de decisiones:** Los datos recopilados pueden ser analizados mediante técnicas de inteligencia artificial. Esto permite extraer patrones significativos, facilitando la toma de decisiones dentro de un proceso.
- **Automatización y control remoto:** A través de aplicaciones web, el IoT brinda la posibilidad a los usuarios de controlar y gestionar dispositivos de forma remota.

Las aplicaciones de IoT son muy variadas pero han encontrado un desarrollo importante en el campo del monitoreo en tiempo real. En casos

reales de producción, resulta difícil prevenir fallos y tener información confiable sobre parámetros de funcionamiento como por ejemplo, tiempos de operación, número de paradas programadas y no programadas, entre otras. La realización de mantenimiento predictivo mediante el uso de IoT como herramienta de recolección de información y análisis de datos integra el monitoreo de condiciones, diagnóstico de fallas, predicción de fallas, soporte para las decisiones de mantenimiento y actividades de mantenimiento [Tian et al., 2021].

## 2.2. Arquitecturas IoT

Al estar conectado al internet, IoT tiene una estrecha relación con la infraestructura disponible en la nube de internet como almacenamiento (*cloud storage*), procesamiento (*cloud computing*), aplicaciones y servicios [Mishra and Kertesz, 2020]. La naturaleza de funcionamiento de los dispositivos y sensores IoT causa que se generen grandes cantidades de información que pueden ser manejadas de manera sencilla por servicios en la nube. De otra manera se necesitaría un mayor esfuerzo tecnológico que incrementaría el costo de utilización. Se busca que el IoT cumpla con los criterios de las arquitecturas estándar que presentan características como modularidad, flexibilidad, escalabilidad e interoperabilidad. Las arquitecturas propuestas actualmente dan un uso correcto a IoT pero no logran converger en un modelo arquitectónico ampliamente aceptable debido al gran número de dispositivos heterogéneos que se conectan. A continuación, se presentan las arquitecturas más conocidas.

### 2.2.1. Arquitectura de 3 capas

La estructura general de IoT puede denotar 3 partes: capa de percepción o física, capa de red y capa de aplicación (véase la Figura 2.1). A continuación se explica la misión que cumplen cada una de estas capas:

**Capa física o de percepción:** define como se transmite la información recolectada del mundo real por medio de sensores, actuadores y dispositivos a otros elementos mediante las funciones de comunicación propias de cada elemento.

**Capa de red:** puede ser cableada o inalámbrica y maneja la transmisión de la información entre los dispositivos IoT y la capa de aplicación. Esta capa maneja el direccionamiento, enrutamiento y encriptado de la información. Utiliza protocolos de baja potencia y baja tasa de transmisión debido a las limitaciones energéticas y de procesamiento propias de los sistemas IoT.



Figura 2.1: Arquitectura de 3 Capas [Mishra and Kertesz, 2020]

**Capa de aplicación:** puede ser una plataforma o una aplicación que procesa la información recolectada para generar nueva información y controlar los diferentes dispositivos interconectados [Simnani et al., 2018, Mishra and Kertesz, 2020]. Esta capa es la que interactúa con el usuario. Permite la representación de la información y colabora con la toma de decisiones. Esta capa se adapta a las necesidades del usuario en cuestión de representaciones visuales, manejo de datos y entrega de información.

Esta descripción de la estructura resulta útil para entender el funcionamiento general de IoT pero no describe de manera técnica y precisa los componentes en los que se basa el funcionamiento de esta tecnología.

### 2.2.2. Arquitectura Referencial de CISCO

La situación mencionada en la sección anterior fue solucionada por CISCO, que establece un modelo referencial para IoT que se presenta a continuación:

CISCO define el estándar del modelo de referencia para IoT, describe las funciones que debe cumplir cada nivel y como interactúan los niveles entre sí. La información se puede desplazar en diferentes direcciones por las capas de manera bidireccional dependiendo si la información va a ser usada o almacenada [Bakhshi et al., 2018]. La arquitectura de este modelo se puede apreciar en la Figura 2.2. A continuación se explica cada una de las capas que componen el modelo:

**Nivel 1: Dispositivos Físicos y Controladores.** Esta capa contiene a los dispositivos físicos y dispositivos controladores que son normalmente llamados “cosas” dentro de los conceptos y el tema de IoT. Estos elementos pueden diferir en tamaño, forma, ubicación, origen, fabricación, uso y demás características propias de los dispositivos. Estas “cosas” son dispositivos de extremo con la capacidad de producir información, enviar y recibir información, y realizar conversiones analógico-digitales [Abed et al., 2020].

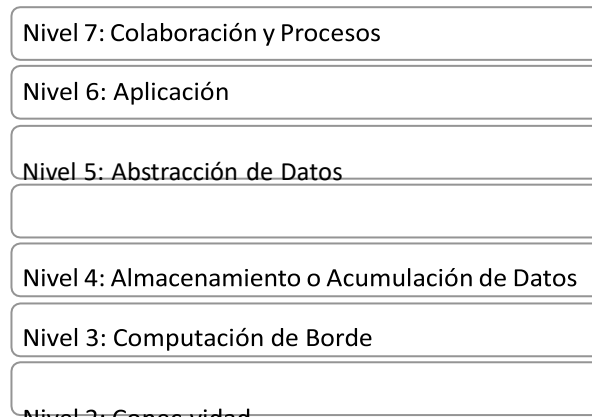


Figura 2.2: Modelo de Referencia de IoT de CISCO [Cha et al., 2018]

**Nivel 2: Conectividad.** Debido a los criterios de diseño que se tienen para IoT respecto a comunicación M2M, puede trabajar de manera perfecta en redes existentes. Este nivel toma en cuenta la importancia de una buena y confiable conectividad. Esto incluye el intercambio de información con y entre dispositivos de nivel 1, la implementación de varios protocolos, conmutación y enrutamiento, traducción entre protocolos, seguridad a nivel de red y analíticas de red [Abed et al., 2020].

**Nivel 3: Computación de Borde.** Este nivel se encarga de la conversión de flujos de datos de red en información adecuada para el almacenamiento y el procesamiento posterior en el nivel 4. Se centra en el análisis y la transformación de datos de gran volumen. El principio fundamental del modelo de referencia de IoT es que los sistemas inteligentes inician el procesamiento de datos lo antes posible y lo más cerca posible del borde de la red, en lugar de depender de la nube para hacer todo el trabajo. A menudo se denomina Edge Computing (computación de borde) o Fog Computing (computación de niebla). Como los datos generados por las cosas generalmente se envían a los dispositivos de red de nivel 2, el nivel 3 implica un procesamiento limitado, sin sesión, sin transacciones, paquete por paquete. Las funciones de elementos de datos del nivel 3 incluyen el filtrado de datos, limpieza, agregación, inspección de contenido de paquetes, una combinación de análisis de red y nivel de datos. Las funciones de generación y procesamiento de eventos incluyen la evaluación de datos para que se procese un nivel superior, reformato de datos para un procesamiento de nivel superior consistente, expansión o decodificación, lo que contiene la agregación

de información adicional como el origen de los datos, destilación/reducción, referido a minimizar el impacto de los datos y el tráfico en la red y los sistemas de procesamiento de nivel superior, y la evaluación del umbral o la alerta, cuyo proceso puede redirigir los datos a destinos adicionales [Bakhshi et al., 2018].

**Nivel 4: Almacenamiento o Acumulación de Datos.** Este nivel usa mecanismos para que las aplicaciones puedan utilizar los datos de la red. Convierte a los datos en movimiento a datos en reposo, formatea paquetes de red en tablas de bases de datos relacionales, permite la transición de la informática basada en eventos a la informática basada en consultas y reduce los datos a través del filtrado y el almacenamiento selectivo. El nivel 4 captura y almacena datos para ser utilizados por aplicaciones que no operan en tiempo real cuando sea necesario. Cierra la brecha entre las redes en tiempo real y las aplicaciones que no operan en tiempo real al funcionar como un límite entre la generación de datos basada en eventos y el uso de datos basado en consultas de nivel superior [Abed et al., 2020].

**Nivel 5: Abstracción de Datos.** Los sistemas IoT requieren adaptarse a la necesidad en cualquier escenario. Para habilitar el escalado en la gran cantidad de datos recibidos por sistemas IoT y no IoT, se requieren varios sistemas de almacenamiento, lo que origina la necesidad de integración de información de varias fuentes de datos. El nivel de abstracción de datos concilia el conflicto entre múltiples formatos de datos de diferentes fuentes y asegura una semántica consistente de los datos en todas las fuentes. Asegura que los datos estén completos para ser utilizados por cualquier aplicación de nivel superior, integra los datos en un solo lugar o brinda acceso a múltiples repositorios mediante la virtualización de datos. Facilita la protección de datos mediante la autenticación, autorización, normalización y desnormalización y utiliza la indexación para proporcionar un acceso rápido a la aplicación [Bakhshi et al., 2018].

**Nivel 6: Aplicación.** Las aplicaciones de esta capa interactúan con el nivel 5 y con la información almacenada o en reposo. Este nivel monitorea la información proveniente de los dispositivos, controla los dispositivos, combina la información que no pertenece a los dispositivos con estos, interpreta y reporta la información [Abed et al., 2020].

**Nivel 7: Colaboración y Procesos.** Este nivel funciona más allá del modelo técnico, incluyendo a los usuarios y procesos de negocio. Para que la información creada por varios sistemas IoT sea de utilidad, debe permitir la producción de análisis detallados. Esto permite tomar una decisión informada y emprender la acción adecuada, lo que a menudo involucra personas y procesos. Las personas necesitan comunicarse y colaborar para

aprovechar al máximo los datos de IoT para tomar las decisiones comerciales correctas en el momento adecuado. Por lo tanto, el Nivel 7 representa un nivel más alto que involucra a los usuarios reales antes que una sola aplicación [Abed et al., 2020].

### 2.2.3. Arquitectura IoT basada en la nube

La arquitectura basada en la nube se implementa a nivel de sistema y busca usar eficientemente los recursos de programa y equipos para aplicaciones específicas del cliente IoT. La capa intermedia está formada por los servidores que contienen el software, plataformas y almacenamiento. La red de acceso al soporte permite la comunicación de aplicaciones y objetos (sensores y actuadores) con la nube. Esta arquitectura utiliza técnicas de virtualización que permite la coexistencia de varios servidores en la misma máquina física, utilizando de manera más eficiente sus recursos. A continuación se describen los elementos principales:

**A. Grupo de recursos virtuales.** Son las técnicas de virtualización que se utilizan para lograr la escalabilidad de los servidores y la utilización eficiente de los componentes físicos y de programa. La máquina virtual ayuda a coordinar e implementar dinámicamente varios servicios en la nube en la misma máquina, lo que reduce los requisitos de máquina y ofrece un alto rendimiento a bajo costo [Simnani et al., 2018].

**B. Servidores de aplicaciones.** El paradigma de IoT generalmente se ejecuta en un escenario restringido por lo que se utilizan protocolos de mensajería asíncrona como MQTT, a diferencia de los entornos y protocolos de aplicaciones tradicionales como HTTP que funciona mediante conexiones TCP en formato de solicitud/respuesta. MQTT es un protocolo ligero que se ejecuta en un modelo de publicación/suscripción, donde los destinatarios solo reciben mensajes cuando están suscritos a ese servicio [Simnani et al., 2018].

**C. Base de datos.** Un servidor de administración de bases de datos en la nube proporciona el almacenamiento de contenido en forma de datos relacionales y no relacionales. El servidor de la base de datos es responsable del mantenimiento y la disponibilidad de estos datos. El lenguaje de consulta estructurado (SQL) se utiliza para el mantenimiento de datos relacionales almacenados en tablas, mientras que los datos no relacionales o distribuidos presentes en forma documentada se mantienen mediante el uso de NoSQL. Se prefiere NoSQL para esta gestión de bases de datos de IoT, ya que los datos asociados son enormes y complejos [Simnani et al., 2018].

**D. Proxy inverso y equilibrio de carga.** Por el gran número de datos

y solicitudes asociados con los nodos de IoT se usa el equilibrio de carga. El equilibrador de carga y proxy inverso distribuyen las solicitudes entre los servidores inactivos para minimizar la carga de trabajo y lograr una alta eficiencia del sistema [Simnani et al., 2018].

## 2.3. IIoT: El Internet Industrial de las Cosas

### 2.3.1. Generalidades

Dentro del marco internacional de desarrollo, la innovación tecnológica juega un papel muy importante en la mejora de la producción y la eficiencia en las diferentes áreas industriales. La mejora de estas dos características influye directamente en la competitividad y en la influencia en el mercado que presenta cualquier empresa, sin embargo, este concepto no es nuevo ya que se ha manejado de manera clara desde el inicio de la industria 2.0. La tendencia mundial va hacia la integración de tecnologías digitales a los procesos industriales para potenciar la automatización de procesos, evitar o disminuir los errores en los procesos, mejorar el control y monitorización de la producción y, principalmente, hacer mucho más, con menos [Soldatos et al., 2019].

En este contexto, toma fuerza el desarrollo de plataformas de Internet de las Cosas para el ámbito industrial, donde su diseño, implementación y funcionamiento cumpla con las exigencias de los ambientes productivos e industriales. Este tipo de plataformas proveen de la infraestructura digital necesaria para integrar a dispositivos industriales con las redes digitales para la recolección y procesamiento de datos, y dependiendo de la aplicación, permite situaciones de análisis históricos de datos, planificación de mantenimientos preventivos e incrementar la eficiencia y control de procesos [Pauli et al., 2021].

Una definición más clara de IIoT es dada por Pauli et al. [2021], cuando propone que dichas plataformas funcionan como un intermediario tecnológico para la comunicación de equipos industriales con los servicios digitales prestados. Esta apreciación incluiría el desarrollo posterior de aplicaciones que trabajen con esta información adquirida de un ambiente real para cumplir con las ventajas mencionadas anteriormente. El concepto general de IIoT combina la producción tradicional con la automatización y la ciencia e inteligencia de datos para generar un sistema más complejo que controle todos los aspectos de la industria. Existen varias opciones presentes en el mercado del IIoT que presentan características diferentes al momento de integrar este desarrollo en la infraestructura tecnológica existente, lo cual

dificulta la fácil elección de una solución.

Las plataformas existentes de IIoT son muy diversas y cada una presenta características específicas. Varios estudios proponen clasificaciones que dependen del criterio del autor pero varias de ellas coinciden en definir una taxonomía que ayude a definir arquetipos generales. [Arnold et al. \[2022\]](#) propone una taxonomía para la mayoría de las plataformas que considera el análisis de capas de cada caso. Inicia con la capa de infraestructura que esta representada por las cosas inteligentes (smart things) del mundo real que se conectan a la plataforma de IIoT, caracterizada por el soporte de equipos, alojamiento de la plataforma y procesamiento de datos como también lo analiza [Sisinni et al. \[2018\]](#). La capa de red propuesta está definida por los modelos OSI y TCP/IP, considerando el transporte de datos físicos (WiFi, Bluetooth, LTE, LoRa o SigFox) y lógicos (HTTP, XMPP, MQTT, CoAP, AMQP, CAN, BACnet o Modbus). La capa intermediaria es la responsable de la recepción, almacenamiento y procesamiento de la información recolectada por las cosas. Corresponde a todas las funcionalidades necesarias para que un sistema digital y físico pueda operar. Sus características principales son la estructura de datos, tipos de análisis, tecnología de análisis, integración externa y código fuente de la plataforma. La última capa corresponde a la capa de aplicación, que se aloja dentro de la plataforma IIoT y trabaja con la información y las funcionalidades de las capas anteriores, integrando aplicaciones de desarrollo de la propia plataforma o de terceros. Esta capa considera las interfaces de programación de aplicaciones o API, el despliegue de aplicaciones y la tienda de aplicaciones.

Tomando en cuenta la taxonomía de las configuraciones arquitectónicas de las plataformas de IIoT presentada anteriormente, se presenta una clasificación de los arquetipos de las plataformas de IIoT. Esta clasificación resulta muy útil al momento de definir que plataforma de IIoT utilizar ya que presenta cinco arquetipos con sus principales características. El Arquetipo 1 consiste en plataformas multifunción que incluyen varias opciones de protocolos de transportación y transmisión de datos, alta capacidad de análisis, posibilidad de integración de sistemas externos. Al proveer de varios servicios, estas plataformas normalmente tienen códigos fuente de componentes libres que promueven el desarrollo de aplicaciones de terceros sin dejar de lado un mercado interno. Los ejemplos más prominentes de este tipo de plataforma IIoT son Predix de GE Digital y MindSphere de Siemens. El Arquetipo 2 se refiere a controladores de dispositivos que se enfocan en almacenamiento público en la nube y procesamiento de datos en la plataforma. Por esto, ofrecen limitadas opciones de protocolos de transportación, transmisión de datos y tecnología de analítica básica que



genera un análisis de datos menos detallado. Dos ejemplos importantes de este arquetipo son KITE y Airtel IoT. El Arquetipo 3 se confiere los centros de datos que ofertan características de procesamiento y análisis de datos, tanto en modalidad de computación de borde como Fog Computing. Los protocolos de transmisión que utilizan son específicos para la industria, mientras que las opciones de transportación pueden ser varias. Estas plataformas IIoT poseen opciones de análisis de datos muy eficientes y poderosas que se encuentran respaldadas por la integración avanzada de otros sistemas. Se puede tener como ejemplo a Foghub y Fog-horn. El Arquetipo 4 trata de plataformas habilitadoras de servicio que combinan las características de los arquetipos 2 y 3. Centran gran cantidad de sus esfuerzos en las opciones de alojamiento de la plataforma, en transporte y transmisión de información lógica y en procesamiento de datos ya sea en borde o dentro de la plataforma. Poseen gran capacidad de análisis y se integran a instancias web externas aprovechando los API estandarizados. Su desarrollo utiliza código cerrado y desarrollan aplicaciones para despliegue interno y externo. Se tiene ejemplos como 5G IoT de Starhub y Relayr. Finalmente, el arquetipo 5 o plataformas conectoras que presentan un alto desarrollo en las características de las capas de intermediario y red. El procesamiento de datos se realiza principalmente por Fog Computing y presentan muchas opciones de integración externa con varias opciones de transportación y transmisión de datos. El análisis de datos que realizan resulta básico y con tipos limitados de analíticas disponibles. La especialidad de estas plataformas es la adquisición y almacenamiento de información, con un énfasis en usar APIs estandarizadas y equipos que sean autorizados por las plataformas. Se tienen ejemplos como IoT Control Center de Cisco y device-WISE de Telit [[Arnold et al., 2022](#)].

Las aplicaciones de IIoT han trascendido a muchas áreas productivas, la versatilidad que presenta ayuda a que el análisis de su aplicación se ramifique hacia grupos generales de uso. Una importante área es la de la logística de procesos industriales con aplicaciones como redes inteligentes y administración de recursos energéticos, sin olvidar que la reducción de los tiempos ocupados en un proceso mejora la eficiencia de las cadenas de suministro. [[Golpîra et al., 2021](#)].

Se presentan también aplicaciones fuera del ámbito industrial, por ejemplo existen importantes aplicaciones en la agricultura a gran escala donde recolecta información como humedad, velocidad del viento, contenido del suelo, temperatura y existencia de plagas, que luego es utilizada en la mejora de las técnicas de agricultura. [[Müller et al., 2018](#)] reporta casos de aplicaciones de investigación en áreas militares donde se establece una

red de comunicación presentada como IoBT o Internet of Battlefield Things que busca incrementar el potencial de los soldados en el campo de batalla por medio del uso de dispositivos IoT como sensores, robots, información biométrica y vehículos especializados. En el área de investigación marina se ha visto su importancia con el proyecto Ocean of Things que busca recopilar información ambiental y de transporte marítimo en largas áreas oceánicas que, previamente, no hubiese sido factible por la limitación de las redes de comunicación tradicionales. [Sharma, 2020]

### 2.3.2. Protocolos para el IIoT

- **MQTT (Message Queuing Telemetry Transport):** es un protocolo de transmisión de datos desarrollado por IBM y Eurotech para la comunicación de redes de sensores inalámbricos, comunicación de máquina a máquina (M2M) y IIoT. Este protocolo presenta gran estabilidad y simplicidad en su implementación, debido a que usa pocos recursos para su funcionamiento. Este protocolo presenta el esquema de publicador y suscriptor para el envío de mensajes, con la particularidad de que si el consumidor no se ha suscrito todavía, el mensaje no se almacenará en el intermediario (*broker*) para esperar a que lo reciba un suscriptor [Uy and Nam, 2019].
- **AMQP (Advanced Message Queuing Protocol):** fue desarrollado inicialmente por JP Morgan Chase y presenta numerosas distribuciones experimentales desde 2014. Usa tanto el esquema publicador/suscriptor y el esquema solicitud/respuesta para el intercambio de mensajes. Su versatilidad ha causado que sea tan utilizado como MQTT a pesar de no ser claro en la estructura de su *broker* y de proveer solo una especificación de protocolo [Iqbal et al., 2023].
- **XMPP (Extensible Messaging and Presence Protocol):** es un protocolo para transmisión en tiempo real que se usa para mensajería instantánea y aplicaciones de colaboración, pero también es gran utilidad en comunicación asíncrona por su eficiencia. Los mensajes usan un mecanismo de publicador/suscriptor eficiente para el envío síncrono de información que, debido a su diseño abierto, facilita cambios y permite escalabilidad [Gerodimos et al., 2023].
- **CoAP (Constrained Application Protocol):** es un protocolo de simple aplicación y de poco consumo de recursos basado en el intercambio asíncrono de mensajes en esquema de solicitud/respuesta.

Es usado por dispositivos actuadores o sensores y, normalmente, se conectan mediante DNS o por recursos de descubrimiento propios de CoAP. Los mensajes usados tienden a ser simples y livianos para evitar la sobrecarga típica de las cabeceras de este tipo de protocolos [Seoane et al., 2021].

### 2.3.3. Pasarelas Inteligentes

Las pasarelas inteligentes son dispositivos diseñados para ambientes industriales que buscan digitalizar o convertir la información entregada por procesos o sensores industriales para ser utilizada por las plataformas IIoT. Esto se debe a dos causas. Primero, algunas implementaciones de IIoT buscan integrar la información de un proceso que no fue diseñado para enviar datos a internet. Segundo, cada fabricante presenta sus propios desarrollos con interfaces de red y protocolos específicos a su marca. Esto no significa que el desarrollador se vea obligado al uso de una marca en particular, ya que las pasarelas inteligentes buscan la adaptación de protocolos, de la capa de transporte, enlace de datos, red y transporte [Kulik and Kirichek, 2018].

La pasarela de Siemens SIMATIC IOT2040 es un equipo basado en el procesador Intel Quark X1020 que utiliza la distribución Yocto de Linux. Tiene conectividad física con puertos RS232, RS422, RS485, Ethernet, USB y posee un puerto de conexión para placas Arduino y tarjetas Mini PCIe, que se pueden almacenar dentro del equipo. Permite el uso de Node-RED de manera nativa para la recepción, transformación y transmisión de datos, junto con la opción de instalar nodos específicos para diferentes fabricantes [Ganga and Ramachandran, 2018].



Figura 2.3: SIMATIC IOT2040 [Siemens, 2016a]

### 2.3.4. Plataformas para el IIoT

La variedad de plataformas IIoT crece con cada desarrollo que se da en esta tecnología. La gran cantidad de información que manejan estos sistemas requieren ser almacenados, procesados y presentados de manera eficiente y sencilla, por lo que, las plataformas IIoT deben tener la suficiente capacidad para alojar la creciente demanda de sus servicios en las diferentes áreas de la industria. Se puede definir a las plataformas IIoT como una variedad de importantes bloques que proveen de las capacidades técnicas necesarias para equipar a los servicios de IIoT [Moura et al., 2018]. A continuación se presentan las principales características de las plataformas IIoT más utilizadas hasta el momento de este trabajo.

- **Watson IoT de IBM:** es una plataforma de IoT altamente escalable que permite combinar la información obtenida por dispositivos IoT con herramientas de análisis y analíticas basadas en procesos cognitivos. Debido a estas características, permite el entrenamiento de modelos de aprendizaje de inteligencia artificial específicos para la extracción de datos, por medio de IBM Watson Knowledge Studio [Conde Camillo da Silva et al., 2022].
- **Azure de Microsoft:** es una plataforma que permite el desarrollo de soluciones de IIoT muy personalizadas para cada caso, desarrolladas por los socios de Microsoft. Debido a esto, se pueden encontrar ejemplos de implementaciones IoT que utilizan componentes IaaS, PaaS y SaaS. Azure ofrece servicios de computo, almacenamiento, inteligencia artificial, analítica, integración de componentes, redes de entrega de contenido y entornos de desarrollo web y móvil [Stackowiak, 2019].
- **ThingWorx de PTC:** es una plataforma dedicada a las implementaciones industriales para el desarrollo, despliegue y extensión de aplicaciones de IoT y realidad aumentada. Se basa en la provisión de aplicaciones desarrolladas que se podrán adaptar o aplicar directamente dependiendo del caso de solución. También incluye la capacidad de análisis en tiempo real de la información recibida de procesos de manufactura, sistemas de ingeniería y equipos de servicio [Mijuskovic et al., 2021].
- **AWS IoT de Amazon:** esta plataforma de IIoT permite la fácil y segura conexión de dispositivos para su posterior almacenamiento y procesamiento [Waterman et al., 2020]. Las principales características son la escalabilidad y altos niveles de

seguridad para los datos importados. Presenta servicios de integración de dispositivos, autenticación, comunicación segura entre dispositivos, almacenamiento, lectura y procesamiento de datos en tiempo real [Supekar and Ahmadina, 2020].

Si bien existe gran variedad al momento de decidir que plataforma IIoT se va a utilizar, la plataforma MindSphere es la más utilizada actualmente debido a sus características y capacidades que se expondrán en la siguiente sección.

## 2.4. MindSphere

MindSphere es una plataforma industrial de internet de las cosas de propiedad de la marca Siemens. Se trata de una aplicación de arquitectura abierta alojada en servidor para el monitoreo, almacenamiento de información, procesamiento estadístico y control de procesos industriales. Entre sus principales funciones está la adquisición de datos que se obtienen de sensores que requieren la recolección y transferencia de información por medio de equipos y herramientas especializadas, junto con los programas inteligentes que permitan su gestión y utilización dentro de la plataforma. MindSphere provee funciones de seguridad que promueven la operación segura de plantas, sistemas, máquinas y redes. [Siemens, 2016b].

De manera general se puede concebir a MindSphere como una plataforma que permite la digitalización de procesos de una organización por medio de la interconexión de dispositivos, máquinas, plantas e infraestructura. Provee servicios de almacenamiento de datos y capacidad de cómputo para procesar dicha información y obtener información analítica de comportamiento de los procesos, información necesaria para toma de decisiones y planificación de mantenimiento preventivo y predicción de fallas [Krok, 2018].

En la Figura 2.4, se puede apreciar los elementos que intervienen en el funcionamiento de una solución utilizando MindSphere. Este sistema consta de 3 elementos separados gráficamente por recuadros: Sección de control, sección de despliegue y sección de plataforma.

**Sección de Control:** El recuadro 1 muestra cómo se realiza la gestión de los recursos de la plataforma. El acceso es por medio de cualquier navegador que posea HTML5 y puede ser un computador, una Tablet o un teléfono inteligente. Los requerimientos mínimos para el acceso a la sección de control son:

- Conexión a Internet y un navegador que soporte HTML5.

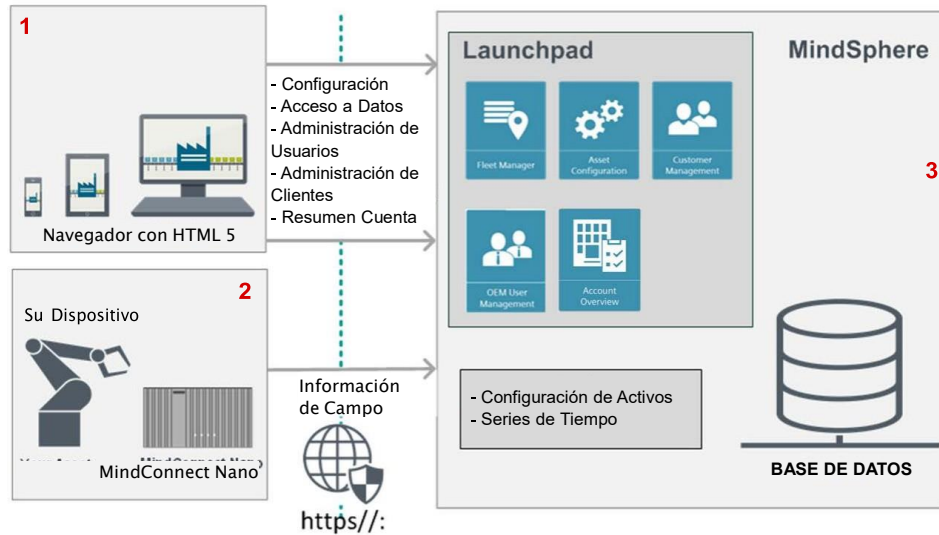


Figura 2.4: Elementos de MindSphere [Siemens, 2016b]

- Conexión a Internet para el dispositivo que recolecta la información para MindSphere. Capacidades estándar HTTPS con conexiones de salidas libres en el puerto 443.
- Enlace hacia Launchpad con usuario y contraseña correctamente registrado por Siemens.
- Dispositivo con una resolución de pantalla mínima de 1024x768.

**Sección de Despliegue:** El recuadro 2 muestra la sección que interactúa con el proceso industrial directamente, siempre se necesitará de sensores para recabar la información de las maquinas, equipos o dispositivos industriales. En este caso, el dispositivo encargado de realizar la transmisión de la información de campo es un equipo MindConnect Nano, pero esta acción puede realizarse con otros equipos como la pasarela inteligente SIMATIC IOT 2040 o por un PLC de la familia S7 1500 que poseen la capacidad de comunicarse con la plataforma MindSphere de manera nativa.

**Sección de Plataforma:** El recuadro 3 muestra el entorno de la plataforma MindSphere, en esta sección se realizan las configuraciones de los recursos a manejar y la base de datos que almacenará toda la información.

La sección de launchpad integra las APIs y aplicaciones disponibles en la plataforma que pueden ser gratuitas o de paga [[Siemens, 2016b](#)].

#### 2.4.1. Uso de la información

La información recolectada puede tener varios usos dentro de MindSphere, pero se nota gran relevancia en las maneras en las cuales se usala información, las cuales son reconocidas por Sadowski, (2019) en su categorización del uso de la información dentro del capitalismo basado en la información. Estas actividades de uso de datos son:

- Optimización de sistemas.
- Manejo y control de las cosas.
- Modelamiento probabilístico.
- Provisión de información para procesos de construcción y creación.
- Crecimiento del valor de los activos que forman parte de la infraestructura.

Este uso dependerá principalmente de la aplicación que realice el programador y el objetivo del proceso industrial. Si bien la gran mayoría de procesos y plantas pueden usar sus datos para todas estas actividades, no significa que será una regla en todos los casos de uso.

#### 2.4.2. Aplicaciones dentro de la plataforma

Esta plataforma permite gestionar la información que recibe de dos maneras, con aplicaciones provistas por la misma plataforma o por aplicaciones desarrolladas por el usuario. MindSphere provee de todas las herramientas necesarias para desarrollar aplicaciones pero tomando en cuenta que esta deberá ser aprobada y deberá seguir lineamientos y reglas de diseño establecidas por el MindSphere Design System, como por ejemplo, una forma visual adecuada, antes de estar disponibles para el uso [[Kulawiak, 2021](#)].

## Capítulo 3

# Configuración de equipos e implementación de la red industrial

En esta capítulo se brindan detalles para la implementación de la red industrial que integra tres variadores de frecuencia, un PLC S7-1500 y la pasarela inteligente IoT2040 para la conexión del proceso con la nube. En primera instancia se presenta la arquitectura de la red detallando la información particular a tener en cuenta antes de cualquier implementación similar. Luego, se presenta la programación del PLC que va a controlar la red industrial mediante el programa TIA Portal. A continuación, se brindan detalles de la configuración de los accionamientos. Una vez implementada la red industrial, se describe los pasos a seguir para conectar la pasarela con la aplicación IIoT en MindSphere.



### 3.1. Arquitectura de la red industrial

En primera instancia, se determinarán las prestaciones de la plataforma MindSphere en modalidad gratuita para definir el alcance de la aplicación que mostrará la información recolectada. Luego, se implementará una red de variadores de frecuencia utilizando el sistema de comunicación Profinet. Los equipos a utilizar en esta etapa serán los variadores G120 de Siemens con las unidades de control CU 250S-2. Esta red estará manejada por un PLC SIMATIC S7-1500 que se conectará a dicha red por medio de un concentrador SCALANCE, ambos de marca Siemens. Todos los elementos mencionados anteriormente serán configurados y puestos en marcha por medio del paquete informático TIA Portal, de Siemens. En una tercera etapa, se diseñará e implementará el sistema de monitoreo de variables en MindSphere y se integrará a la red industrial implementada anteriormente. La conexión entre la plataforma y la red industrial se realizará por medio de la pasarela SIMATIC IoT2040. En la Figura 3.1 se presenta el diagrama de implementación para esta propuesta.

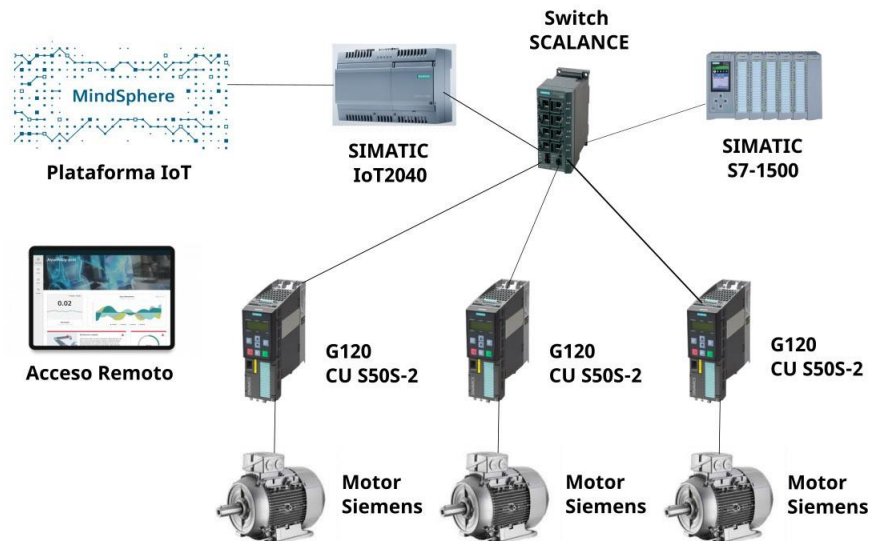


Figura 3.1: Diagrama de Implementación del Proyecto

Para esta implementación se utilizarán los siguientes equipos.

- Programadora con TIA Portal V15.1 y herramienta Sinamic STARTER (PC).

- Controlador Lógico Programable Siemens SIMATIC S7 1500 con CPU 1516-3 PN/DP.
- Switch industrial SCALANCE X208.
- Pasarela inteligente Siemens SIMATIC IoT2040.
- 3 variadores de frecuencia Siemens SINAMICS G120.
- 3 unidades de control de motores Siemens CU S50S-2 (comunicación profinet).
- 3 motores SIMOTICS GP 1AV3082B.
- Plataforma Minsphere en versión demo.

El desarrollo del presente proyecto se ha dividido en dos grandes partes. En primera instancia se realizará la red industrial que permite la integración de los accionamientos SINTAMICS G120, el PLC y la pasarela inteligente. Para este caso la integración se realizará a través de un sistema de comunicación Profinet. Por otra parte, la segunda parte del proyecto incluye lo relacionado con la capa de aplicación. En esta parte se explicará la configuración y programación de Mindsphere para el monitoreo del sistema.

## 3.2. Implementación de la red industrial

En la Figura 3.2 se muestra la topología de la red industrial a implementar.

La red industrial consta de 3 accionamiento SINAMICS G120 con CU250S-2 de la marca SIEMENS. Estos dispositivos serán controlados por un PLC Siemens S7-1500 que se conecta a la red por medio de un switch industrial SCALANCE X208. Toda la red utiliza el protocolo PROFINET. La configuración y programación de los equipos se realizó en la herramienta TIA Portal v15.1.

### 3.2.1. Configuración de hardware

Dentro de TIA Portal se debe crear el proyecto de automatización y agregar tanto el PLC como los accionamientos. En este caso los accionamientos se agregaron directamente del catálogo de hardware, no obstante, se debe tener en cuenta que si los dispositivos no son visibles en este catálogo se deben descargar los archivos GSD ((General Station Description)) correspondientes para luego instalarlos en TIA Portal. Este

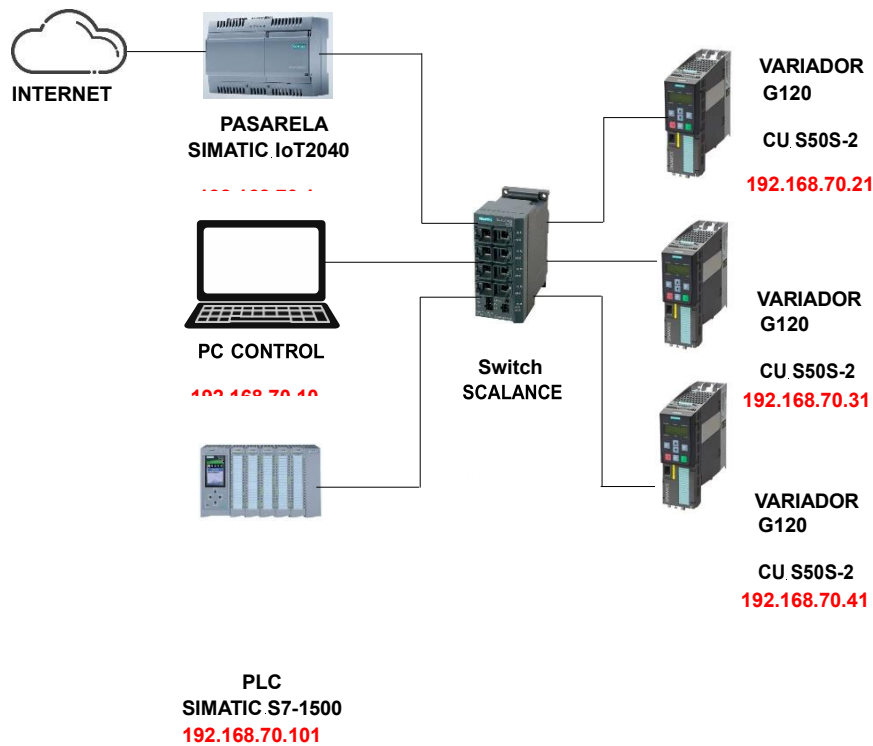


Figura 3.2: Topología de la Red Industrial

procedimiento hará que los equipos se encuentren disponibles en el catálogo. La Figura 3.3 muestra la topología de red implementada dentro de TIA Portal.

Junto con la configuración de la dirección IP en el PLC se habilita la comunicación remota en las propiedades del PLC: *General* → *Protección y Seguridad* → *Mecanismos de conexión* → *Permitir acceso por comunicación PUT/GET para usuario remoto*.

Antes de la inclusión de los accionamientos es necesario incluir la librería *LSINExt\_v15.1*, disponible en la página oficial del fabricante.

Los variadores de frecuencia SINAMICS G120 se pueden encontrar en: *Vista de redes* → *Catálogo de hardware* → *Otros dispositivos de campo* → *PROFINET IO* → *Drivers* → *SIEMENS AG SINAMICS* → *SINAMICS*

*G120 CU250S2 VECTOR V4.7*. Luego, se añade el Telegrama 20 que entregará los datos de los motores en la siguiente ruta: *Catálogo de hardware* → *Submódulos Telegrama Estándar 20, PZD-2/6*. El siguiente paso es habilitar la conexión con los variadores por medio de la asignación del nombre del dispositivo haciendo click derecho en la red PROFINET como se muestra

en la Figura 3.4

### *3.2. IMPLEMENTACIÓN DE LA RED INDUSTRIAL*

27

Al realizar el proceso descrito anteriormente, se abre la ventana de diálogo

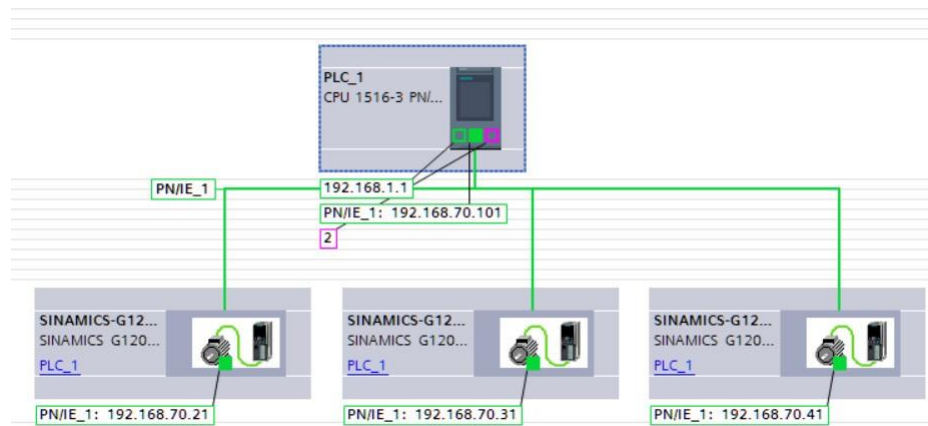


Figura 3.3: Topología de la Red Industrial en TIA Portal.

donde se selecciona el dispositivo PROFINET *sinamics-g120sv-pn*, para el primer accionamiento, y se selecciona actualizar lista (véase la Figura 3.5). Una vez encontrado el accionamiento en la sección inferior, se selecciona asignar nombre para iniciar la conexión. Este proceso se deberá repetir para los otros 2 accionamientos, teniendo en cuenta los nombres de dispositivo *sinamics-g120sv-pn\_1* y *sinamics-g120sv-pn\_2*.

Luego de finalizar la configuración de los variadores y el PLC, se crea un bloque de datos donde se alojarán los datos a transmitir hacia la plataforma IIoT con cada variable debidamente configurada.

Se debe configurar el bloque de datos quitando el acceso optimizado al bloque en la siguiente ruta dentro de las propiedades del bloque de datos: *General Atributos De-seleccionar Acceso optimizado al bloque*. Las variables dentro del bloque de datos quedan como se ve en la Figura 3.6.

Las variables dentro del bloque de datos incluyen principalmente las velocidades, corrientes, torques y potencias de cada uno de los motores. Todas estas variables son de tipo Real. En complemento se han generado otras variables de tipo entero para registrar el tiempo de funcionamiento de cada motor. Todas estas variables pueden ser constatadas de forma intuitiva en la Figura 3.6.

### 3.2.2. Programación en TIA Portal

A continuación se detalla el programa implementado en TIA Portal v15.1. Se han utilizado 8 salidas digitales del PLC. El panel de control general

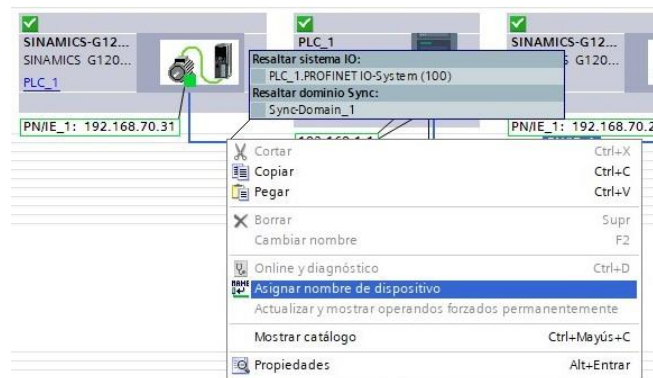


Figura 3.4: Acceso a la asignación de nombre para los variadores.

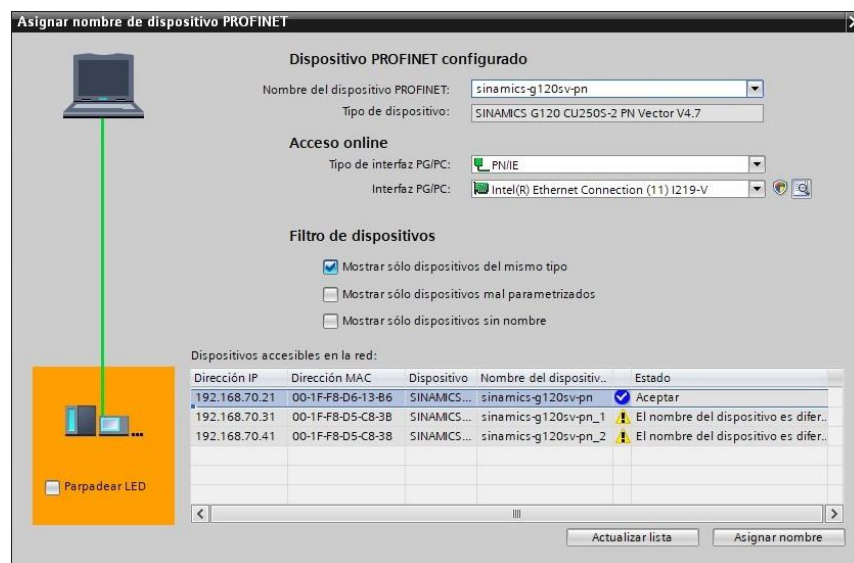


Figura 3.5: Asignación de nombre para los variadores.

tendrá un pulsante de paro general, 3 pulsantes normalmente abiertos de inicio de marcha de los motores, 3 interruptores para la elección del sentido de giro de los motores y un interruptor para el reconocimiento de los errores que puedan aparecer en los motores. De acuerdo con las especificaciones del fabricante, los motores se conectan en configuración delta. El diagrama de conexión entre el PLC y uno de los variadores se puede ver en la Figura

DBTxMS (snapshot created: 4/17/2023 8:55:48 PM)								
	Name	Data type	Offset	Start value	Retain	Accessible f...	Writa...	Visible in ...
1	Static							
2	Velocidad_Actual_M1	Real	0.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	Corriente_Actual_M1	Real	4.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	Torque_Actual_M1	Real	8.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	Potencia_Actual_M1	Real	12.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6	Velocidad_Actual_M2	Real	16.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	Corriente_Actual_M2	Real	20.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8	Torque_Actual_M2	Real	24.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9	Potencia_Actual_M2	Real	28.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10	Velocidad_Actual_M3	Real	32.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	Corriente_Actual_M3	Real	36.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12	Torque_Actual_M3	Real	40.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13	Potencia_Actual_M3	Real	44.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14	Segundos_M1	Int	48.0	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	Minutos_M1	Int	50.0	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
16	Horas_M1	Int	52.0	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
17	Segundos_M2	Int	54.0	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
18	Minutos_M2	Int	56.0	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
19	Horas_M2	Int	58.0	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
20	Segundos_M3	Int	60.0	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
21	Minutos_M3	Int	62.0	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
22	Horas_M3	Int	64.0	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Figura 3.6: Bloque de datos con las variables a exportar.

3.7. De acuerdo con el esquema de interconexión planteado, se asignan las variables de programa como se muestra en la Figura 3.8.

En total se han creado 41 variables para la programación del autómata. Las variables auxiliares listadas desde la fila 1 hasta la fila 6, son variables de propósito general que facilitan la programación del usuario. Las variables etiquetadas como Clock, las cuales se listan desde la fila 7 hasta la fila 15, son señales de reloj con la frecuencia especificada en el nombre de cada variable. Las variables etiquetadas como Config\_Axis, listadas desde la fila 16 hasta la fila 18 se utilizan para definir el sentido de giro del motor. Las variables etiquetadas como Eje\_habilitado, listadas desde la fila 19 hasta la fila 21, se utilizan para habilitar los accionamientos correspondientes. Las variables etiquetadas como Error\_VIKNAMUR, listadas desde la fila 22 hasta la fila 24 se utilizan para registrar una palabra de control o, en su defecto, una palabra de error. Las variables etiquetadas como Habilitar, listadas desde la fila 25 hasta la fila 27 se utilizan para la lógica de encendido y apagado de los motores. Las variables etiquetadas como Ingresar\_Velocidad, listadas desde la fila 28 hasta la fila 30 se utilizan para definir la velocidad en rpm de los motores. Las variables etiquetadas como Inicio, listadas desde la fila

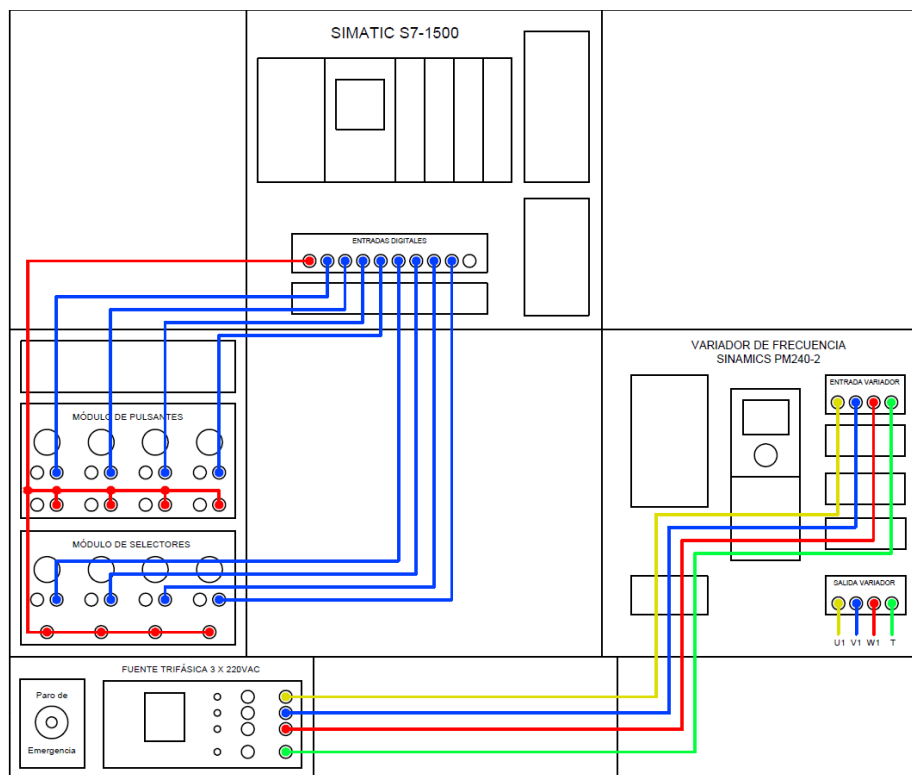


Figura 3.7: Diagrama de conexión entre el PLC y uno de los accionamientos.

31 hasta la fila 33 se utilizan para representar a las entradas digitales de inicio del motor, relacionadas con los pulsantes del tablero de control. Las variables etiquetadas como Mantenimientos, listadas desde la fila 34 hasta la fila 36 se utilizan para definir los tiempos de funcionamiento máximos en los cuales se deberá realizar el mantenimiento de los motores. Las variables Paro\_General y Reset\_motores, como su nombre lo indican son variables asociadas a elementos de entrada que permiten parar el funcionamiento de los motores y resetear la configuración. Finalmente, las variables etiquetadas como Sentido\_Giro, sirven para seleccionar el giro de los tres motores.

El primer segmento del programa presenta la habilitación de la marcha de los motores. Por otra parte, se ha programado un segmento para el control de cada motor. La primera sección del control de sentido de giro de uno de los motores se puede ver en la Figura 3.10.

El control de velocidad de los motores se realiza con el bloque controlador



PLC tags					
	Name ▲	Tag table	Data type	Address	Retain
1	Aux1	Tabla de variables e..	Bool	%M2.0	<input type="checkbox"/>
2	Aux11	Tabla de variables e..	Bool	%M2.1	<input type="checkbox"/>
3	Aux2	Tabla de variables e..	Bool	%M2.2	<input type="checkbox"/>
4	Aux22	Tabla de variables e..	Bool	%M2.3	<input type="checkbox"/>
5	Aux3	Tabla de variables e..	Bool	%M2.4	<input type="checkbox"/>
6	Aux33	Tabla de variables e..	Bool	%M2.5	<input type="checkbox"/>
7	Clock_0.5Hz	Tabla de variables e..	Bool	%M0.7	<input type="checkbox"/>
8	Clock_0.625Hz	Tabla de variables e..	Bool	%M0.6	<input type="checkbox"/>
9	Clock_1.25Hz	Tabla de variables e..	Bool	%M0.4	<input type="checkbox"/>
10	Clock_10Hz	Tabla de variables e..	Bool	%M0.0	<input type="checkbox"/>
11	Clock_1Hz	Tabla de variables e..	Bool	%M0.5	<input type="checkbox"/>
12	Clock_2.5Hz	Tabla de variables e..	Bool	%M0.2	<input type="checkbox"/>
13	Clock_2Hz	Tabla de variables e..	Bool	%M0.3	<input type="checkbox"/>
14	Clock_5Hz	Tabla de variables e..	Bool	%M0.1	<input type="checkbox"/>
15	Clock_Byte	Tabla de variables e..	Byte	%M80	<input type="checkbox"/>
16	Config_Axis_M1	Tabla de variables e..	Word	%MW100	<input type="checkbox"/>
17	Config_Axis_M2	Tabla de variables e..	Word	%MW108	<input type="checkbox"/>
18	Config_Axis_M3	Tabla de variables e..	Word	%MW110	<input type="checkbox"/>
19	Eje_Habilitado_M1	Tabla de variables e..	Bool	%M1.4	<input type="checkbox"/>
20	Eje_Habilitado_M2	Tabla de variables e..	Bool	%M1.6	<input type="checkbox"/>
21	Eje_Habilitado_M3	Tabla de variables e..	Bool	%M1.7	<input type="checkbox"/>
22	Error_VIKNAMUR_M1	Tabla de variables e..	Word	%MW102	<input type="checkbox"/>
23	Error_VIKNAMUR_M2	Tabla de variables e..	Word	%MW112	<input type="checkbox"/>
24	Error_VIKNAMUR_M3	Tabla de variables e..	Word	%MW114	<input type="checkbox"/>
25	Habilitar_M1	Tabla de variables e..	Bool	%M1.2	<input type="checkbox"/>
26	Habilitar_M2	Tabla de variables e..	Bool	%M1.3	<input type="checkbox"/>
27	Habilitar_M3	Tabla de variables e..	Bool	%M1.5	<input type="checkbox"/>
28	Ingresar_Velocidad_M1	Tabla de variables e..	Real	%MD104	<input type="checkbox"/>
29	Ingresar_Velocidad_M2	Tabla de variables e..	Real	%MD116	<input type="checkbox"/>
30	Ingresar_Velocidad_M3	Tabla de variables e..	Real	%MD120	<input type="checkbox"/>
31	Inicio_M1	Tabla de variables e..	Bool	%I16.1	<input type="checkbox"/>
32	Inicio_M2	Tabla de variables e..	Bool	%I16.2	<input type="checkbox"/>
33	Inicio_M3	Tabla de variables e..	Bool	%I16.3	<input type="checkbox"/>
34	Mantenimiento_M1	Tabla de variables e..	Int	%IW0	<input type="checkbox"/>
35	Mantenimiento_M2	Tabla de variables e..	Int	%IW2	<input type="checkbox"/>
36	Mantenimiento_M3	Tabla de variables e..	Int	%IW4	<input type="checkbox"/>
37	Paro_General	Tabla de variabl... <input type="text" value=""/>	Bool <input type="text" value=""/>	%I16.0	<input type="checkbox"/>
38	Reset_Motores	Tabla de variables e..	Bool	%I16.7	<input type="checkbox"/>
39	Sentido_Giro_M1	Tabla de variables e..	Bool	%I16.4	<input type="checkbox"/>
40	Sentido_Giro_M2	Tabla de variables e..	Bool	%I16.5	<input type="checkbox"/>
41	Sentido_Giro_M3	Tabla de variables e..	Bool	%I16.6	<input type="checkbox"/>

Figura 3.8: Variables de programa.

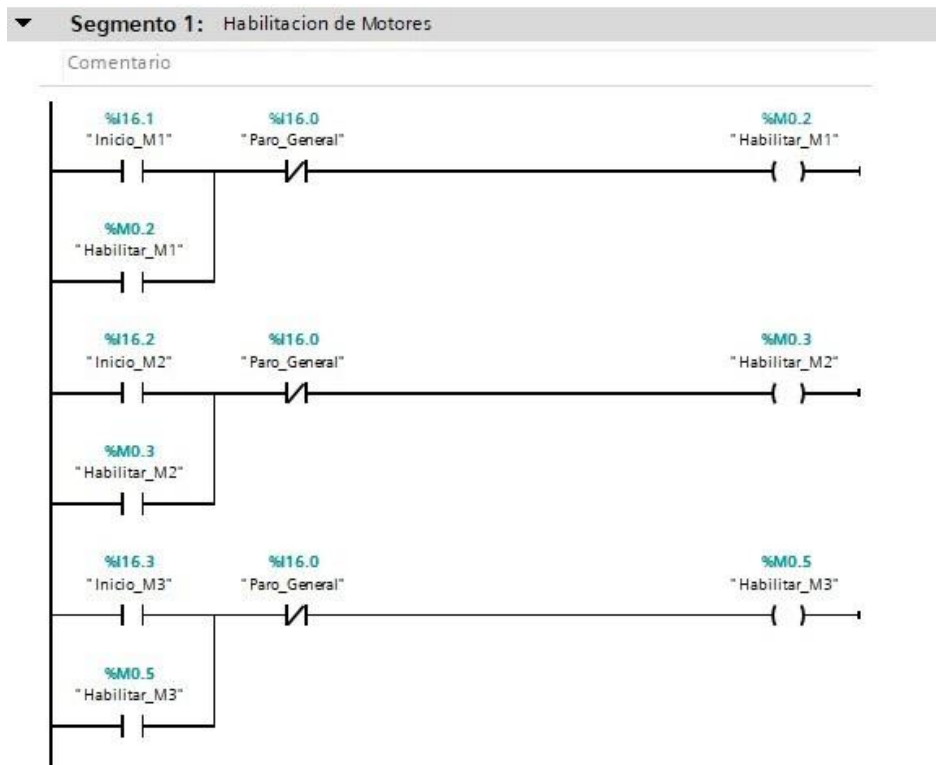


Figura 3.9: Segmento de habilitación de los motores.

disponible en: *Librerías* → *Librerías globales* → *LSINAEExt\_V15.1* → *Plantillas maestras* → *SINA\_SPEED\_TLG20*.

Cada motor utiliza la correspondiente configuración según se ilustra en la Figura 3.11. Par este bloque de control se utilizan las variables declaradas anteriormente teniendo en cuenta las terminaciones M1 para el motor 1, M2 para el motor 2 y M3 para el motor 3. Se debe tener especial atención al asignar las variables HWIDSTW y HWIDZSW, ya que estas deben corresponder con los nombres asignados a cada variador en el orden correcto.

Luego, se habilita los relojes internos en la siguiente ruta de la configuración del PLC: *Propiedades* → *Memoria de sistema y de reloj* → *Bits de memoria de reloj* → *Habilitar el uso del byte de memoria de reloj*. Esto habilita varios relojes en la marca M0 y permite implementar la medición del tiempo de funcionamiento en minutos y horas como se muestra el caso del motor 2 en la Figura 3.12.

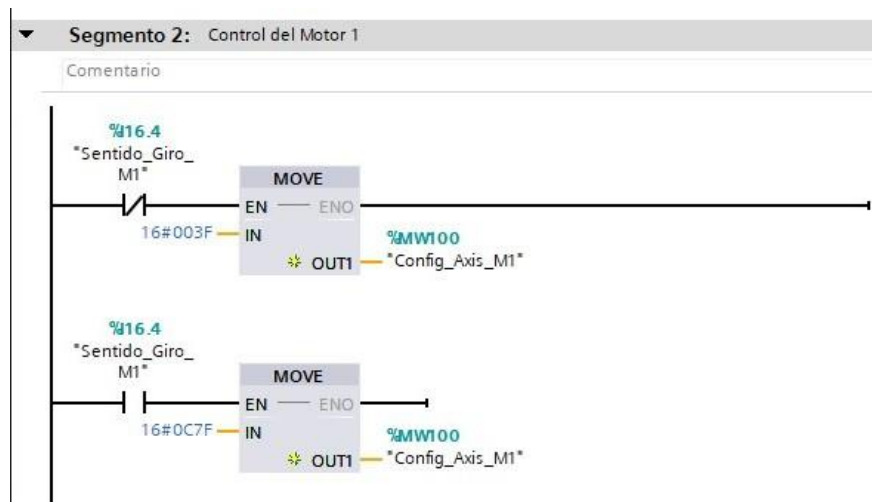


Figura 3.10: Segmento de control de sentido de giro.

Una vez terminado el programa de control, se realiza el procedimiento para la carga en el PLC (véase la Figura 3.13). Mientras se ejecuta este proceso aparecerá una alerta preguntando si se desea agregar una dirección IP, la cual se aceptará. En las acciones de carga al PLC se debe asegurar que se detengan todos los módulos.

### 3.2.3. Configuración de los Accionamientos

La configuración de los accionamientos a utilizar puede hacerse de varias maneras. En este caso, por la facilidad de acceso se escogió realizar la configuración manual mediante el Panel Inteligente de Operación o IOP-2 en cada variador. La Figura 3.14 muestra una vista frontal del panel de configuración IOP-2.

En el variador, se inicia en la pantalla principal de IOP-2 y se restablece los ajustes de fábrica con los siguientes pasos: *Instalación (icono de engranaje)* → *Puesta en marcha rápida* → *OK* → *Si, reestablecer ajustes de fábrica* → *OK*. Esto asegura que el variador use una configuración adecuada para la red y elimina cualquier configuración ajena a esta aplicación. Luego de este proceso, se configura la información del motor con los siguientes valores: *Norma del motor: Nema (60 Hz, SI)* → *Sel. tipo de motor: Asíncrono* → *Tensión de alimentación: 220 V* → *Corriente del motor: 2.20 Aef* → *Potencia del motor: 0.55kW* → *Velocidad del motor: 1440.00 rpm* → *Tensión del*

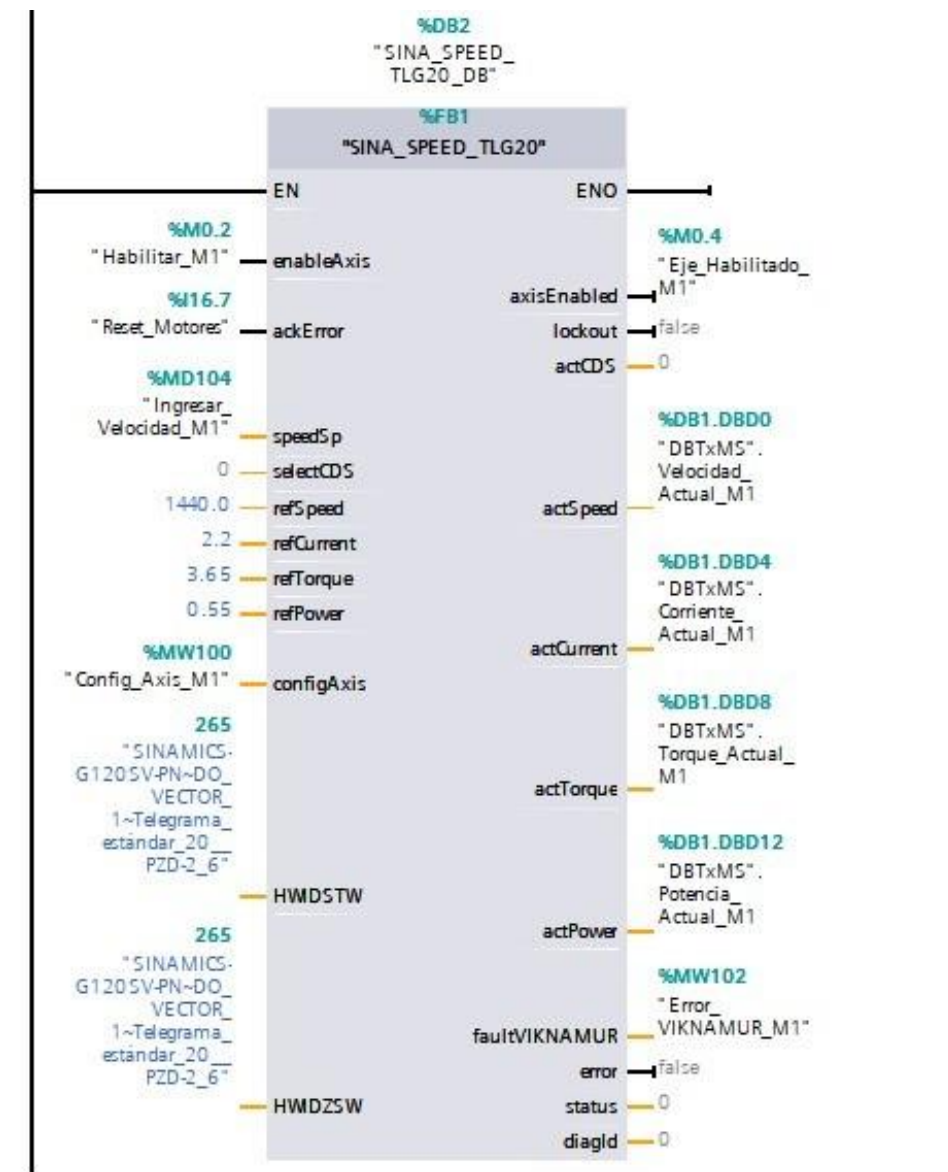


Figura 3.11: Bloque de control de velocidad el motor 1.

motor: 220 V → Frecuencia del Motor: 60 Hz → Velocidad mínima: 0.00 rpm → Velocidad máxima: 1440.00 rpm → Tiempo de aceleración: 10 s →

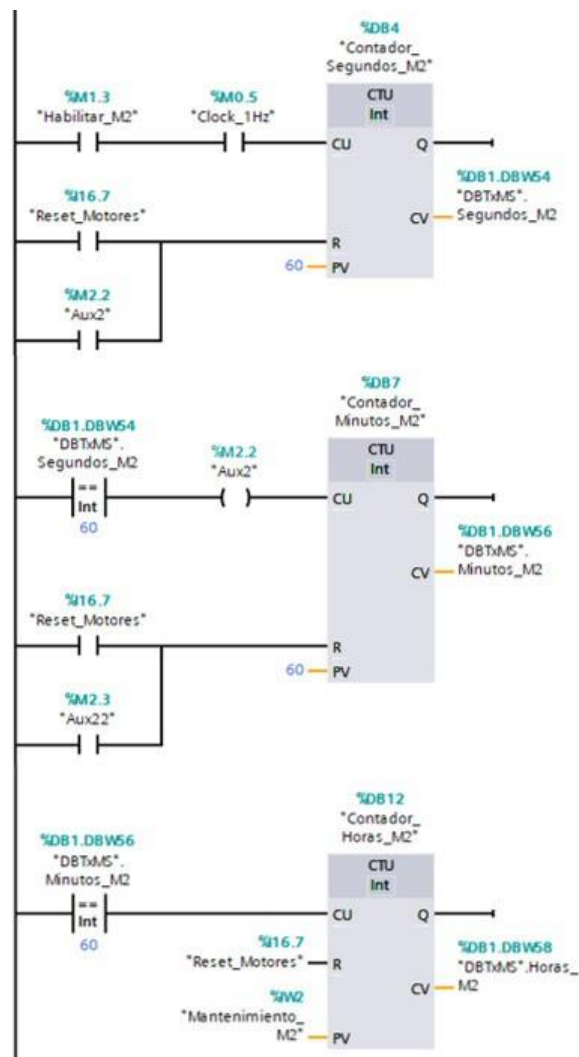


Figura 3.12: Bloque de conteo del tiempo de funcionamiento del motor 2.

Tiempo de desaceleración: 10 s → Configuración de E/S: (7) Regulación PN/PD → OK (presionado por 2 segundos).

A continuación, se configura el telegrama 20: Parámetros → Búsqueda por número → 922 → OK → PZD Selec\_telegr → 20. Teleg estándar 20 → OK.

Una vez configurado, se debe iniciar el reconocimiento del motor, siguiendo

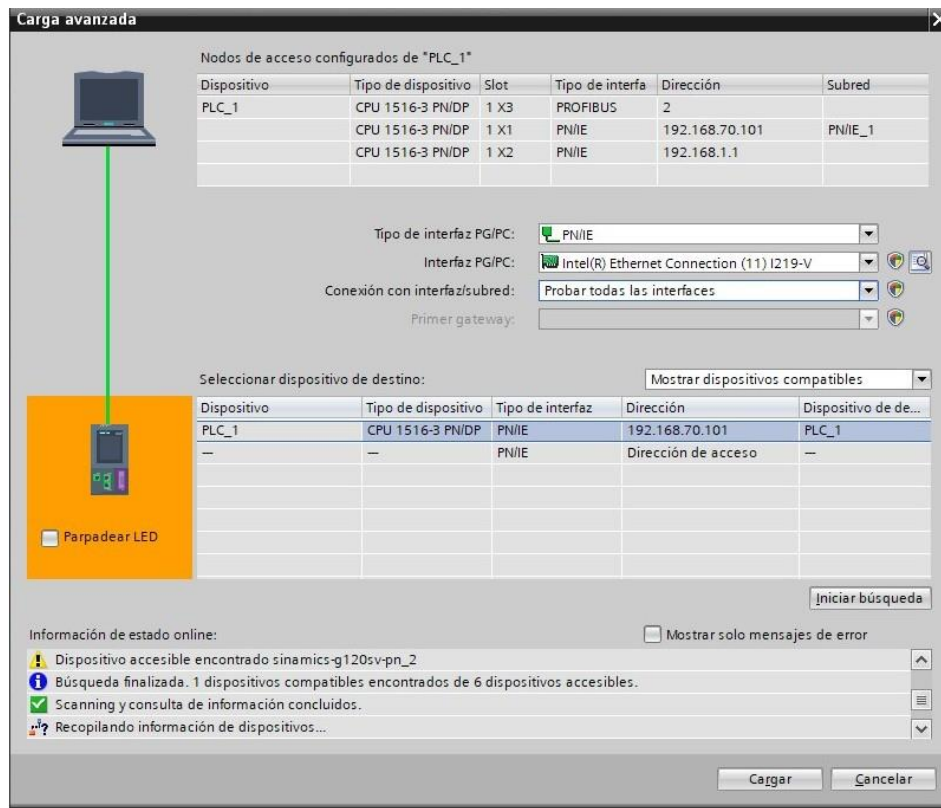


Figura 3.13: Carga de la configuración de hardware y del programa de control al PLC.

estos pasos: *Presionar botón HAND/AUTO Presionar el botón inicio verde al lado derecho. Para finalizar esta operación: Presionar el botón parorjo al lado izquierdo Presionar botón HAND/AUTO.*

Este proceso debe repetirse para los tres accionamientos a utilizar. Para este caso los tres motores son idénticos, por lo que las configuraciones también son idénticas. Luego de realizar todo este proceso los tres accionamientos con sus respectivos motores estarán listos para su utilización dentro de la red industrial.



Figura 3.14: Pantalla de configuración IOP-2.

### 3.3. Configuración de la Pasarela IoT2040

#### 3.3.1. Configuración de la pasarela IoT2040

La pasarela inteligente SIMATIC IoT2040 de Siemens requiere una tarjeta de memoria microSD de entre 8GB y 32GB para alojar el firmware necesario para su funcionamiento. El fabricante provee una imagen de disco de la distribución Yocto del sistema operativo Linux y está disponible en la página oficial de soporte del equipo. Esta se puede montar utilizando herramientas como Win32 Disk Imager.

La configuración de la pasarela se realiza desde cualquier programa de gestión de clientes SSH. En este caso se utilizó la herramienta PuTTY. Primero se conecta al dispositivo con la dirección por defecto: 192.168.200.1, como se indica en la Figura 3.15.

Una vez conectado por SSH, se debe ingresar como usuario root y se accede a la configuración del dispositivo por el comando `iot2000setup`. Esto se puede ver en la Figura 3.16.

Una vez que se tenga acceso al dispositivo, lo primero que se configura es la zona horaria en: *OS Settings* → *Change Timezone* → *America/Bogota* → *OK*. Segundo, se configura la conectividad del dispositivo en la opción *Networking*. Aquí se debe habilitar las dos interfaces Ethernet disponibles (véase la Figura 3.17).



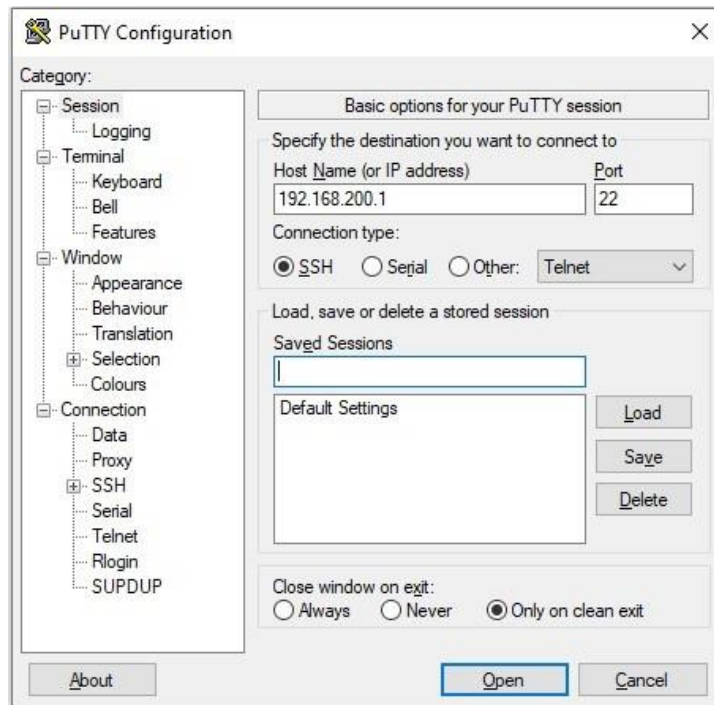


Figura 3.15: Configuración del programa PuTTY para la conexión desde la PC a la pasarela.

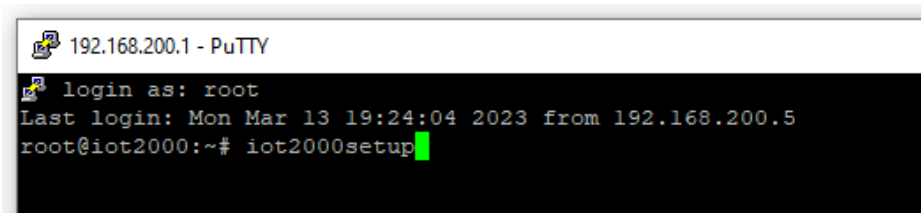
A continuación, se configuran los programas que se ejecutarán al encender el dispositivo. Esto se realiza a través de la siguiente ruta: *Software* → *Manage Autostart Options*. Este procedimiento se ilustra en la Figura 3.18.

En la consola del cliente SSH se instalan los nodos de comunicación utilizando la ubicación */usr/lib/node\_modules*. En este caso, se debe instalar MindConnect para la comunicación con MindSphere. Luego, se debe instalar el nodo para la comunicación con los PLC SIMATIC S7. Las especificaciones para estas instalaciones se ilustran en las Figuras 3.19 y 3.20, respectivamente.

### 3.3.2. Flujo de Programa en Node-RED

Una vez configurada la pasarela IoT, se implementa el flujo de programa ilustrado en la Figura 3.21 para enviar la información desde el PLC hacia MindSphere.





```
192.168.200.1 - PuTTY
login as: root
Last login: Mon Mar 13 19:24:04 2023 from 192.168.200.5
root@iot2000:~# iot2000setup
```

Figura 3.16: Comandos para acceder a la configuración de la pasarela.

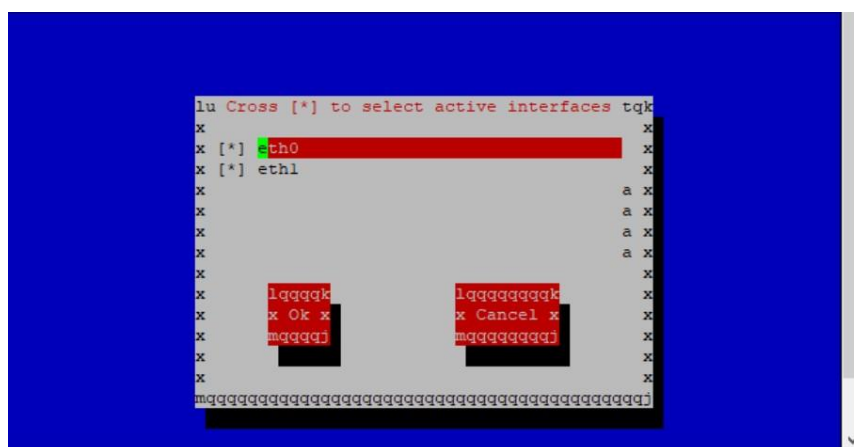


Figura 3.17: Configuración de conectividad de la pasarela.

Se inicia las configuraciones con el nodo de conexión S7in de la paleta de nodos instalados recientemente. Al dar doble click, se tiene acceso a la primera configuración que deberá corresponder a la información mostrada en la Figura 3.22.

Para acceder a la siguiente configuración, se deberá hacer click en el icono con forma de lápiz que se encuentra junto a la dirección de red del PLC. La configuración de conexión se ilustra en la Figura 3.23

Luego, se debe configurar las variables a ser recibidas desde el PLC, estas deben coincidir con la configuración de variables en la base de datos del programa de TIA Portal. En la Figura 3.24 se muestra una parte de la ventana de diálogo que contiene las variables configuradas. Una vez finalizada la configuración de variables se da click al botón *Update*.

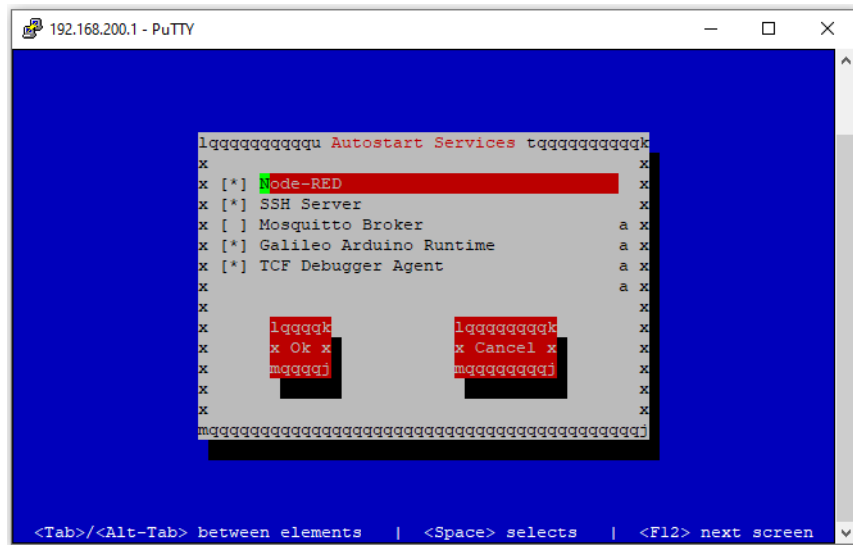


Figura 3.18: Configuración de inicio de la pasarela.

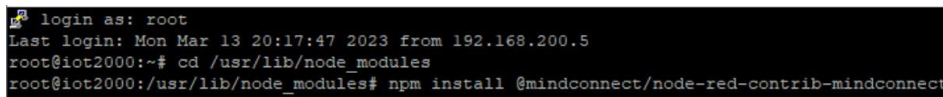


Figura 3.19: Comando para la instalación del nodo MindConnect.

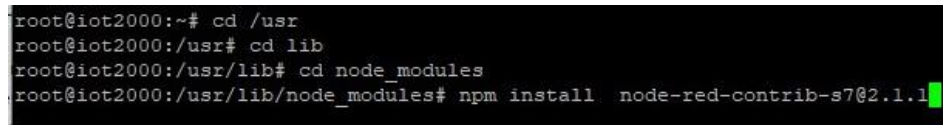


Figura 3.20: Comando para la instalación del nodo S7.

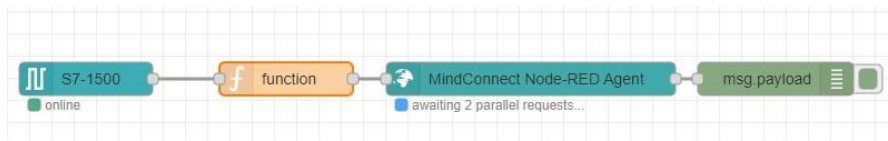
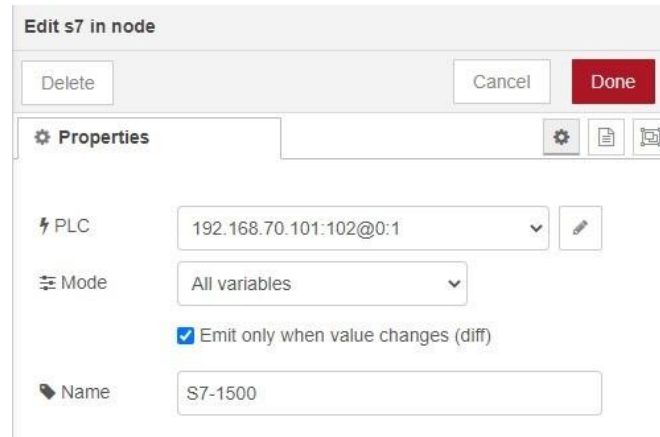


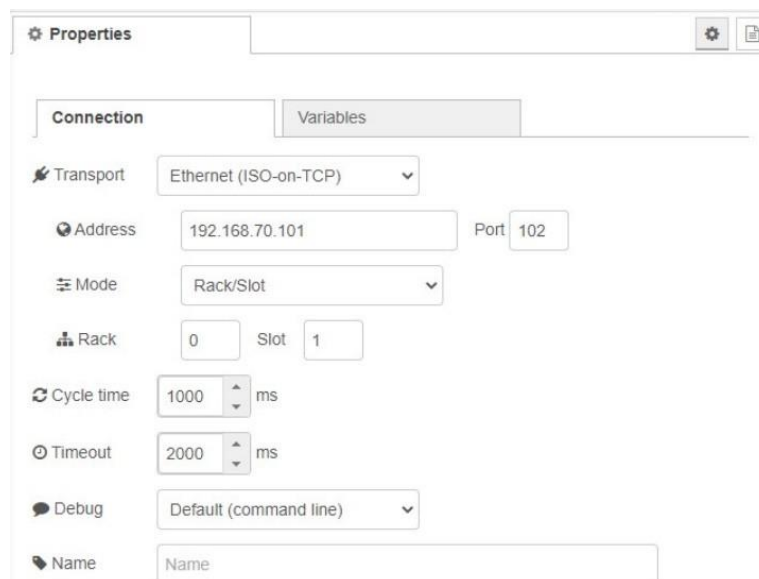
Figura 3.21: Flujo de programa en Node-RED.



The screenshot shows the 'Edit s7 in node' configuration window. At the top, there are buttons for 'Delete', 'Cancel', and 'Done'. Below this is a 'Properties' section with a gear icon and a refresh icon. The configuration includes:

- PLC:** A dropdown menu showing '192.168.70.101:102@0:1' with an edit icon.
- Mode:** A dropdown menu showing 'All variables'.
- Emit only when value changes (diff):** A checked checkbox.
- Name:** A text input field containing 'S7-1500'.

Figura 3.22: Configuración inicial del nodo S7in.



The screenshot shows the 'Properties' window with the 'Connection' tab selected. The configuration includes:

- Transport:** A dropdown menu showing 'Ethernet (ISO-on-TCP)'.
- Address:** A text input field containing '192.168.70.101' and a 'Port' field containing '102'.
- Mode:** A dropdown menu showing 'Rack/Slot'.
- Rack:** A text input field containing '0' and a 'Slot' field containing '1'.
- Cycle time:** A spinner field set to '1000' ms.
- Timeout:** A spinner field set to '2000' ms.
- Debug:** A dropdown menu showing 'Default (command line)'.
- Name:** A text input field containing 'Name'.

Figura 3.23: Configuración de conexión del nodo S7in.

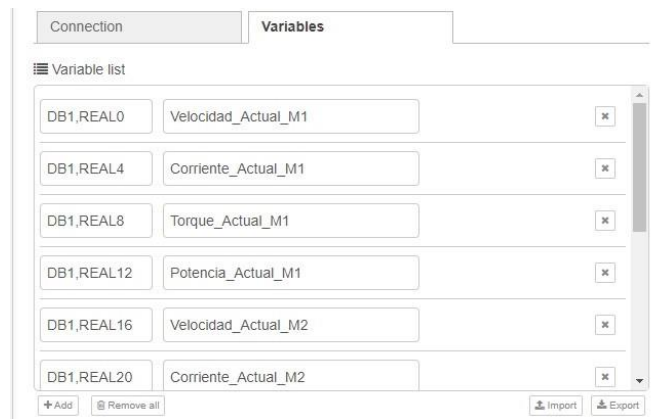


Figura 3.24: Configuración de variables del nodo S7in (parte 1).

## **Capítulo 4**

# **Implementación de la aplicación en Mindsphere y análisis de resultados**

En esta capítulo se presentan los detalles de la implementación de la aplicación en Mindsphere para el monitoreo web de los parámetros de funcionamiento de los tres motores trifásicos que son comandados por los accionamientos que operan dentro de una red industrial Profinet.

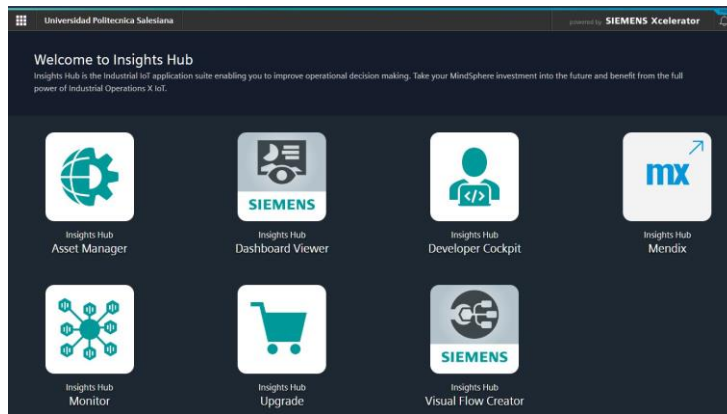


Figura 4.1: Pantalla de inicio de MindSphere.

## 4.1. Activos, aspectos y variables en MindSphere

Para el uso de MindSphere es necesaria una cuenta que se puede crear en la página oficial de SIEMENS. Para este caso se utilizó una versión gratuita. Una vez que se registra y verifica el correo ingresado, se tendrá la pantalla inicial como se muestra en la Figura 4.1.

En primera instancia, en la sección *Asset Manager*, se debe crear los tipos de aspectos o *Aspect Types* que contendrán las variables a importar del proceso industrial. Para esto, se debe escoger la opción *Library Aspect Types Create aspect*. En esta instancia se escoge un nombre para el tipo de aspecto a crearse y este deberá estar categorizado como dinámico. Al final de la página, se añaden las variables a importar como ilustra en la Figura 4.2.

Las variables deberán coincidir en tipo y unidades durante toda la implementación. El número máximo de variables que puede incluir un aspecto es 6, por lo que se crearon 3 aspectos con las siguientes variables que tendrán los campos de nombre, unidad y tipo de dato:

- INFO\_IOT\_2040
  - VELOCIDAD\_M1, rpm, DOUBLE.
  - CORRIENTE\_M1, A, DOUBLE.
  - TORQUE\_M1, Nm, DOUBLE.
  - POTENCIA\_M1, kW, DOUBLE.

— Variables

i

- Variable names must be unique inside an aspect.
- The data type BIG\_STRING is only available for a dynamic aspect.

+ Add variable  
 ↶ Import variables  
 ↓ Download template  
 ↓ Export variables  
 🗑 Delete all new

Name	Unit	Data type
<input type="text" value="CORRIENTE_M1"/>	<input type="text" value="A"/>	DOUBLE
<input type="text" value="CORRIENTE_M2"/>	<input type="text" value="A"/>	DOUBLE
<input type="text" value="POTENCIA_M1"/>	<input type="text" value="kW"/>	DOUBLE
<input type="text" value="TORQUE_M1"/>	<input type="text" value="Nm"/>	DOUBLE
<input type="text" value="VELOCIDAD_M1"/>	<input type="text" value="rpm"/>	DOUBLE
<input type="text" value="VELOCIDAD_M2"/>	<input type="text" value="rpm"/>	DOUBLE

Save
Cancel

Figura 4.2: Inclusión de las variables del proceso industrial.

- VELOCIDAD\_M2, rpm, DOUBLE.
- CORRIENTE\_M2, A, DOUBLE.
- INFO\_IOT\_2040\_2
  - TORQUE\_M2, Nm, DOUBLE.
  - POTENCIA\_M2,kW, DOUBLE.
  - VELOCIDAD\_M3, rpm, DOUBLE.
  - CORRIENTE\_M3, A, DOUBLE.
  - TORQUE\_M3, Nm, DOUBLE.
  - POTENCIA\_M3, kW, DOUBLE.
- INFO\_IOT\_2040\_3
  - MINUTOS\_M1, min, INT.
  - HORAS\_M1, h, INT.
  - MINUTOS\_M2, min, INT.

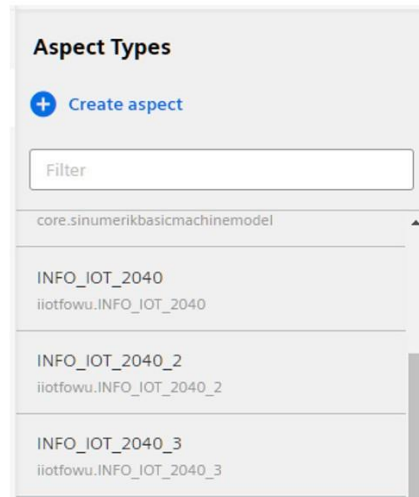


Figura 4.3: Aspectos creados dentro de Mindsphere.

- HORAS\_M2, h, INT.
- MINUTOS\_M3, min, INT.
- HORAS\_M3, h, INT.

Si se añadieron correctamente, se podrán encontrar los aspectos creados en la lista como se ve en la Figura 4.3.

A continuación, en la opción Asset Types, se crea el tipo de activo o *asset* que incluirá los aspectos y variables creadas anteriormente. Se deberá escoger un nombre para el activo a crearse.

Al final de la página se añaden los aspectos creados recientemente al hacer click en *Browse aspects* y se seleccionan los elementos correspondientes. El resultado de este proceso se muestra en la Figura 4.4.

Al finalizar este proceso, el tipo de activo se encontrará en la lista de disponibilidad y mostrará como característica que está enlazado a 3 aspectos.

Se necesita crear dos activos, un activo que contenga la información importada desde el proceso industrial y otro que conecte a la pasarela con la plataforma. Esto se realiza en la pestaña superior izquierda donde se escogió anteriormente la opción *Library*. La creación de estos activos se muestra en las Figuras 4.5 y 4.6, respectivamente. El segundo activo que conecta la pasarela IOT2040 con la plataforma debe ser del tipo *MindConnectLib*

Dentro de la lista de activos, se selecciona *CONEXION\_PLC\_IOT* y se ingresa en la configuración de conectividad. Dentro del plugin de





— Aspects

[+ Add aspect](#) [🔍 Browse aspects](#)

Name	Aspect	Category
> INFO_IOT_2040	iiofowu.INFO_IOT_2040	Dynamic
> INFO_IOT_2040_2	iiofowu.INFO_IOT_2040_2	Dynamic
> INFO_IOT_2040_3	iiofowu.INFO_IOT_2040_3	Dynamic

[Save](#) [Cancel](#)

Figura 4.4: Inclusión de las variables del proceso industrial.

## Add asset



— General

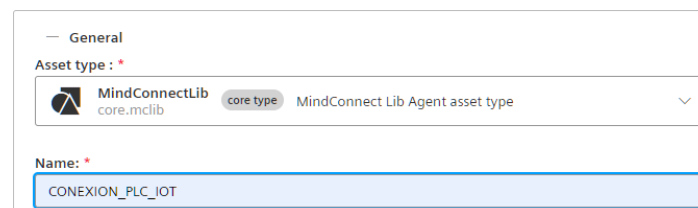
Asset type : \*

 IOT\_2040\_MOTOR\_TIPO  
iiofowu.IOT\_2040\_MOTOR\_TIPO

Name: \*


DATOS\_PLC\_IOT

Figura 4.5: Creación del activo de datos.



— General

Asset type : \*

 MindConnectLib core.mclib core type MindConnect Lib Agent asset type

Name: \*

CONEXION\_PLC\_IOT

Figura 4.6: Creación del activo de conexión.

configuración de MindConnect, se escoge el perfil de seguridad que se utilizará para la conexión. En este caso, se selecciona la opción *SHARED SECRET* y se continúa con la configuración. Al finalizar este proceso

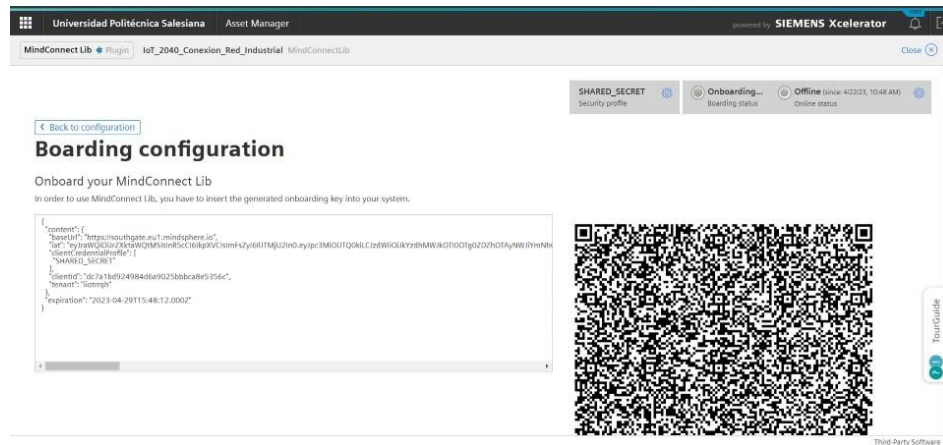


Figura 4.7: Información de seguridad de MindConnect.

Figura 4.8: Inclusión de la fuente de datos.

aparecerá una pantalla con la llave que se ingresará en el nodo MindConnect de la pasarela IOT2040 (véase la Figura 4.7).

Si el proceso fue exitoso, la plataforma indicará: *Your MindConnect Lib is onboarded!*, el estado de *Onboarding* se pondrá en verde. A continuación, se selecciona *Pestaña Configuration Enter edit mode Add data source* para crear las fuentes de información (véase la Figura 4.8). Se utilizó el identificador DS para diferenciarlo de los puntos de datos (DP).

Una vez creadas las fuentes de datos, se añaden los puntos de datos o *Data Points* haciendo click en *Add data point* al lado derecho del nombre de la fuente creada. Se tendrá el menú que se ilustra en la Figura 4.9

En este, se ingresa la información de los datos del proceso con las unidades

The screenshot shows a dark-themed 'Add data point' dialog box. At the top, it says '1 of 6 data points are linked'. The 'Name' field contains 'DP-VELOCIDAD\_M1'. The 'Description' field is empty. The 'Type' dropdown is set to 'DOUBLE'. The 'Unit' dropdown is set to 'rpm'. Below the unit field, there is a note: '\* required input field'. At the bottom, there are 'Accept' and 'Cancel' buttons. A 'TourGuide' icon is visible on the right side of the dialog.

Figura 4.9: Inclusión del punto de datos.

Configuration		Data mappings
Data mappings (0 deprecated, 0 invalid)		
DS-INFO_IOT_2040		4 of 6 Data points linked
Data point	Data mapping	
VELOCIDAD_M2 rpm, DOUBLE		<a href="#">Link variable</a>
VELOCIDAD_M1 rpm, DOUBLE		<a href="#">Link variable</a>

Figura 4.10: Ventana de mapeo de puntos de datos y variables.

y tipo que se establecieron al momento de la creación de las variables, como se puede ver en la Figura 4.9. Luego, se debe enlazar las variables con los puntos de datos creados en la pestaña *Data mappings* (véase la Figura 4.10).

Cada DP se va a enlazar a una variable por medio del enlace *Link variable* al lado derecho de cada punto de datos. A continuación se selecciona *Change* y se tendrá una ventana como se observa en la Figura 4.11. Luego, se selecciona *Up* y finalmente se selecciona *CONEXION\_PLC\_IOT* y la variable a conectar. Si el enlace fue exitoso, se podrá observar una línea que indica como se transportarán los datos dentro de la plataforma. El resultado de este proceso se muestra en la Figura 4.12.

En la vista del grupo de datos DS se tendrá un resumen indicando cuantas

## 4.2. ENVÍO DE DATOS DESDE NODE-RED HACIA MINDSPHERE 50



Figura 4.11: Inicio del enlace de puntos de datos y variables.

Configuration		Data mappings
18 Data mappings (0 deprecated, 0 invalid)		
▼ DS-INFO_IOT_2040		
Data point	Data mapping	
VELOCIDAD_M2 rpm , DOUBLE	DATOS_PLC_IOT > INFO_IOT_2040 > VELOCIDAD_M2	
VELOCIDAD_M1 rpm , DOUBLE	DATOS_PLC_IOT > INFO_IOT_2040 > VELOCIDAD_M1	

Figura 4.12: Enlace de puntos de datos y variables exitoso.

variables han sido enlazadas, como se observa en la Figura 4.13.

## 4.2. Envío de datos desde Node-RED hacia MindSphere

Este procedimiento se inicia ingresando la llave obtenida en la configuración de MindSphere que se explicó en la sección anterior. Luego, se verifica la información configurada en MindSphere haciendo click en el botón *Agent Information* de la misma ventana de configuración. En esta se podrá observar primero la configuración de las fuentes de datos. En la Figura 4.14 se puede observar la lista de enlaces disponibles.

Luego de verificar que la información es correcta, se da click al botón

## 4.2. ENVÍO DE DATOS DESDE NODE-RED HACIA MINDSPHERE 51

Configuration ID: CF-iiotfowu.IOT_2040_MOTOR_TIPO		3 Data sources		18/18 Data points are linked	
DS-INFO_IOT_2040		6 Data points			
Data point	Data point ID	Linked	Data type	Unit	
VELOCIDAD_M2	DP-VELOCIDAD_M2	✓	DOUBLE	rpm	
VELOCIDAD_M1	DP-VELOCIDAD_M1	✓	DOUBLE	rpm	
TORQUE_M1	DP-TORQUE_M1	✓	DOUBLE	Nm	
POTENCIA_M1	DP-POTENCIA_M1	✓	DOUBLE	kW	
CORRIENTE_M2	DP-CORRIENTE_M2	✓	DOUBLE	A	
CORRIENTE_M1	DP-CORRIENTE_M1	✓	DOUBLE	A	
DS-INFO_IOT_2040_2		6 Data points			
Data point	Data point ID	Linked	Data type	Unit	
VELOCIDAD_M3	DP-VELOCIDAD_M3	✓	DOUBLE	rpm	
TORQUE_M3	DP-TORQUE_M3	✓	DOUBLE	Nm	

Figura 4.13: Resumen de enlace de puntos de datos y variables.

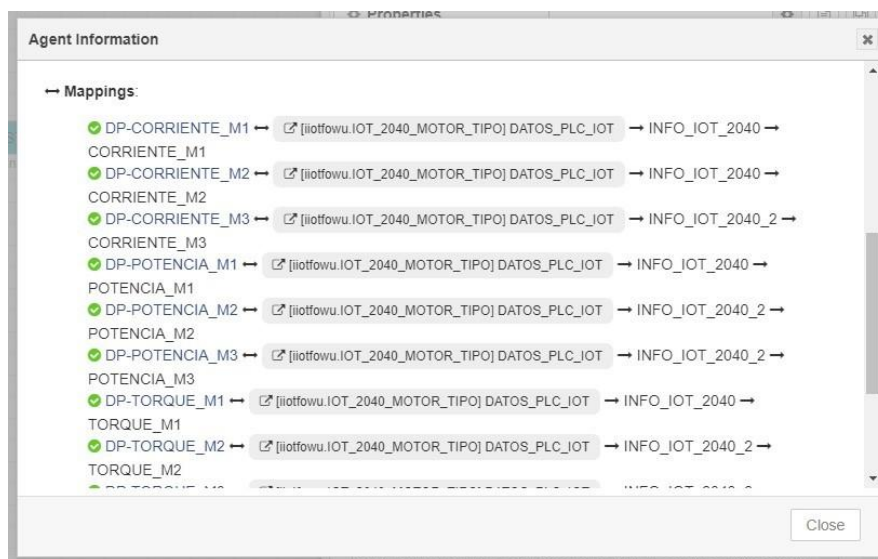


Figura 4.14: Lista de enlaces en MindSphere.

*TimeSeries Template* en la parte superior izquierda de esta ventana y se pega esta información en el nodo *Function*. Por cada variable, se tendrá 3 valores: *dataPointId*, *qualityCode* y *value*. *dataPointId* es el identificador de dato que se asigna en la configuración de variables de MindSphere y *qualityCode* deberá permanecer en valor 0. *value* viene configurado de inicio

#### 4.2. ENVÍO DE DATOS DESDE NODE-RED HACIA MINDSPHERE 52

como *DOUBLE*, pero deberá editarse en cada caso para importar el valor del PLC que coincida con el identificador de dato. Esto convertirá la información adquirida al formato que requiere MindSphere y se puede observar un extracto de esta configuración en la Figura 4.15.

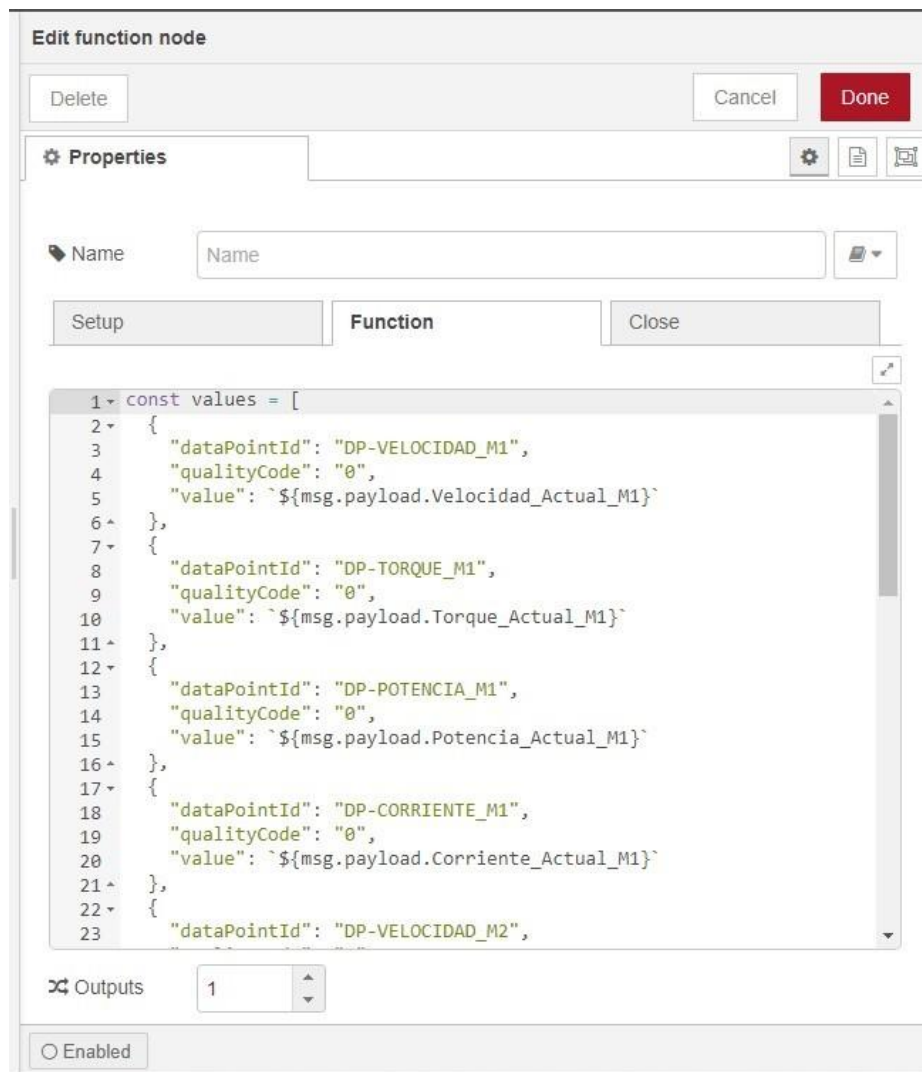


Figura 4.15: Nodo Function.

Una vez finalizada esta configuración, se procede a configurar la conexión entre la pasarela y la plataforma IoT. Dentro de la configuración general del

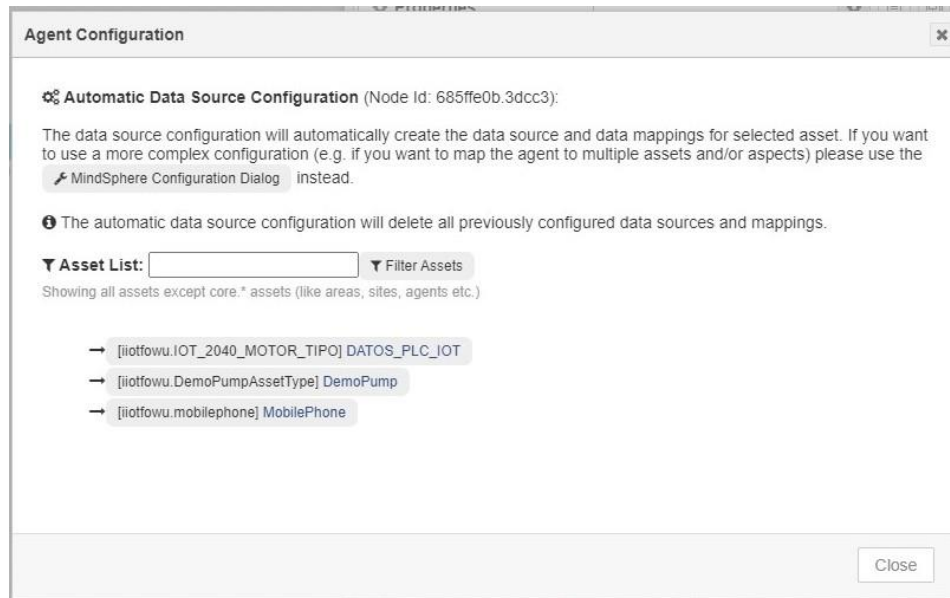


Figura 4.16: Configuración del Agente hacia MindSphere.

nodo MindConnect, se da click al botón *Agent Configuration*, lo que abrirá la ventana que se ilustra en la Figura 4.16.

Se selecciona el activo correspondiente a la implementación [iiotfowu.IOT\_2040\_MOTOR\_TIPO]DATOS\_PLC\_IOT. Esta configuración puede tomar varios minutos. Una vez lista, se tendrá confirmación correspondiente.

Al terminar esta configuración, el flujo de Node-RED empezará a transmitir datos a la plataforma y se tendrá señalamientos de color verde debajo de los nodos como confirmación en el entorno de programación. Mientras tanto en MindSphere, se muestra la confirmación de enlace en el activo configurado para la recolección de datos (véase la Figura 4.17).

### 4.3. Presentación de información en MindSphere

La configuración de la presentación de información se realiza en la aplicación *Visual Flow Creator* de la plataforma (véase la Figura 4.18). Esta tiene un entorno de programación por nodos similar a Node-RED. Los datos se presentan en los paneles Series de tiempo, Valores actuales y Tiempo de



Figura 4.17: Confirmación de configuración de conexión en MindSphere

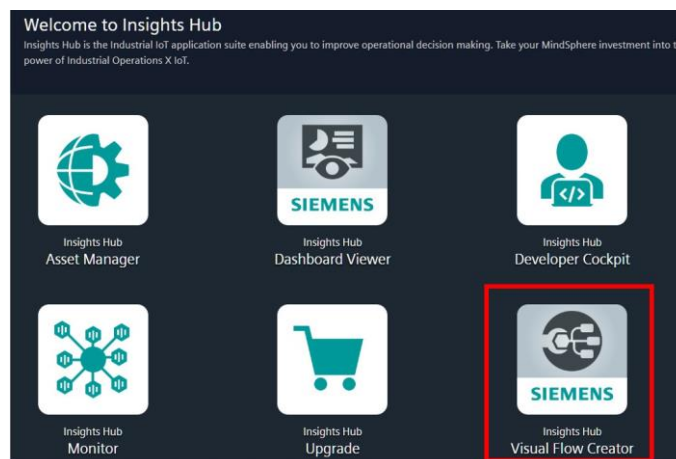


Figura 4.18: Visual Flow Creator en MindSphere.

funcionamiento (véase la Figura 4.19). La adquisición de datos se realiza cada 30 segundos con un nodo *INJECT* (véase la Figura 4.20).

#### 4.3.1. Visualización de series de tiempo

La pestaña de series de tiempo se programa conforme se indica en la Figura 4.21. Cada variable se obtiene por un nodo *read timeseries* donde se configura el nombre del nodo y la fuente de los datos en el botón *Topic* (véase la Figura 4.22)

Dentro de la configuración del nodo de importación de series de tiempo, se escoge la variable a visualizar como se muestra en la Figura 4.23.

Luego, se conecta un nodo *chart* para visualizar la serie de tiempo,



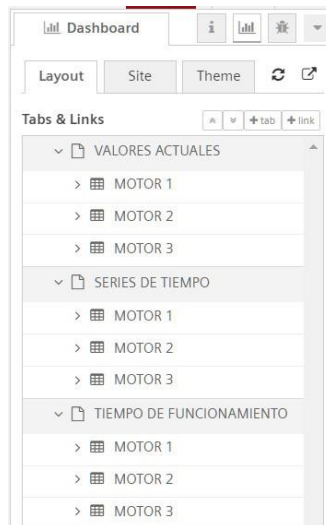


Figura 4.19: Configuración del panel de visualización.

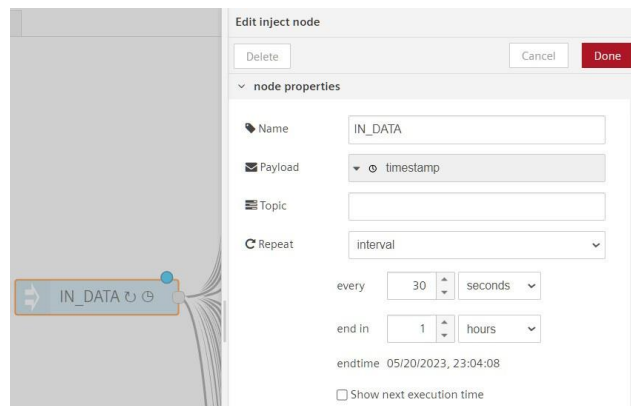


Figura 4.20: Configuración del nodo INJECT.

configurando la unidad, valores máximos y mínimos para las variables. En este caso las series de tiempo corresponderían a las variables de velocidad, corriente, torque y potencia. La configuración de la variable de velocidad se puede ver en la Figura 4.24. Este proceso se repite para cada variable de velocidad, corriente, torque y potencia de cada motor.

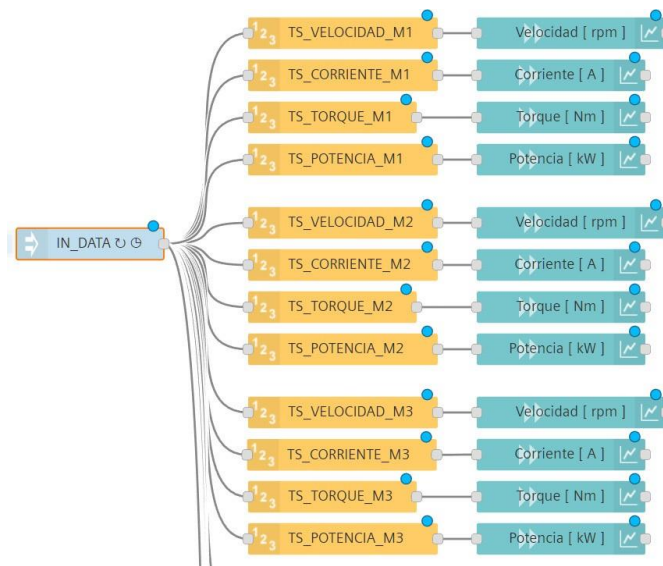


Figura 4.21: Programa de series de tiempo.



Figura 4.22: Adquisición de series de tiempo.

### 4.3.2. Visualización de valores actuales

Este proceso parte del nodo *INJECT* hacia la lectura de las series de tiempo. Por ejemplo, para el motor 1, de la serie de tiempo de velocidad se desprende la lectura de la velocidad actual (véase la Figura 4.25). El nódulo función *Reciente* filtra el último valor con el código que se ilustra en la Figura 4.26.

Este nodo entrega un valor que incluye la estampa de tiempo completa con la que MindSphere importa los datos del proceso, por tanto, se extrae el valor de interés con el nodo *filtro* que contiene el código mostrado en la

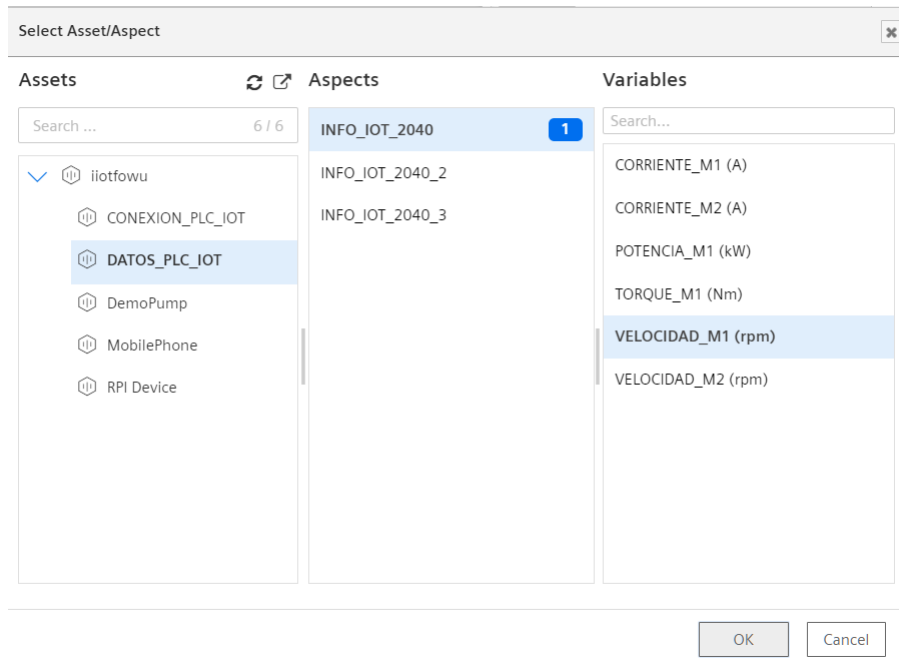


Figura 4.23: Selección de variables a visualizar.

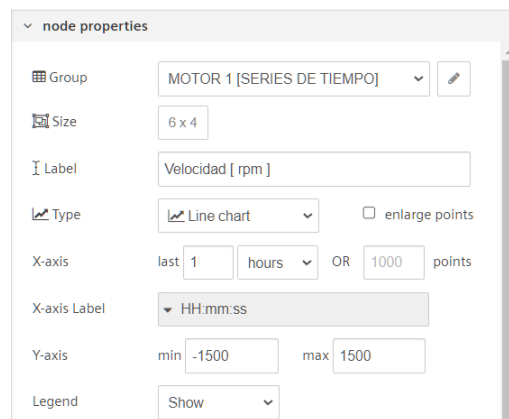


Figura 4.24: Configuración de Visualización de series de tiempo.

Figura 4.27.

A continuación, se conecta un nodo *gauge* para presentar visualmente este



Figura 4.25: Programa de visualización de valores actuales para el motor 1.

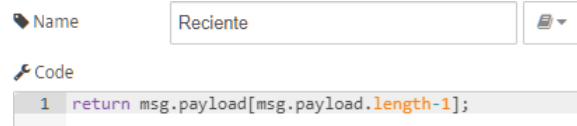


Figura 4.26: Lectura del último valor importado.

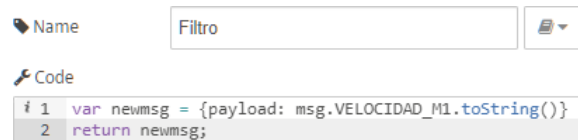
valor con la configuración mostrada en la Figura 4.28.

Este procedimiento se repite para la corriente, torque y potencia, con la diferencia que en estos dos últimos se cambia el tipo de *gauge* a *donut*. También, se debe especificar la variable a extraer en el nodo *Filtro* y cambiar las unidades, valores máximo y mínimo en cada caso.

### 4.3.3. Visualización del promedio y la desviación estándar

Mindsphere ofrece la posibilidad de obtener el promedio y la desviación estándar de una serie de tiempo. Estos valores podrían resultar de utilidad en la supervisión de un proceso, pues podrían facilitar las tareas de mantenimiento de un equipo tanto de actuación como de medición. Para obtener el promedio se requiere de un nodo *moving average* entre el nodo de series de tiempo y el nodo *Reciente*. Este nodo no requiere configuración. Esta información se visualizará como etiqueta con un nodo *text* que tendrá la configuración que se ilustra en la Figura 4.29.

Para este caso, se presenta también la desviación estándar de la corriente, con el fin de tener una medida precisa y rápida de la variación de corriente en cada motor. Esta información puede resultar importante al monitorear un proceso industrial ya que puede indicar la presencia de comportamientos no



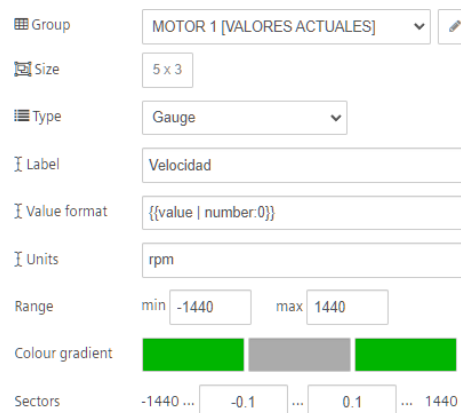
The screenshot shows a configuration interface with a 'Name' field containing 'Filtro' and a 'Code' editor. The code editor contains two lines of JavaScript code:

```

1 var newmsg = {payload: msg.VELOCIDAD_M1.toString()}
2 return newmsg;

```

Figura 4.27: Filtrado del valor de interés.



The screenshot shows a configuration interface for a gauge visualization. The settings are as follows:

- Group: MOTOR 1 [VALORES ACTUALES]
- Size: 5 x 3
- Type: Gauge
- Label: Velocidad
- Value format: {{value | number:0}}
- Units: rpm
- Range: min -1440, max 1440
- Colour gradient: A gradient bar with green, grey, and green segments.
- Sectors: -1440 ... -0.1 ... 0.1 ... 1440

Figura 4.28: Visualización de velocidad actual.

esperados. De manera ideal, este valor debería ser 0, pero por la naturaleza real de los motores es posible que se observen pequeñas desviaciones. La obtención de la desviación estándar se hace por medio de un nodo *standard deviation* cuyo modo esté configurado como escalar. La visualización de esta información se hará en un nodo *text* con la configuración mostrada en la Figura 4.30.

El proceso descrito en esta sección se repetirá para los motores 2 y 3 de la red, teniendo especial cuidado en la configuración de las variables correctas.

#### 4.3.4. Visualización del tiempo de funcionamiento

El panel de tiempo de funcionamiento se obtiene mediante el programa ilustrado en la Figura 4.31. En este programa se utilizan configuraciones similares a las de los paneles anteriores con respecto a la adquisición de datos en series de tiempo, la obtención del último dato importado y el filtrado de información. La presentación de esta información se realiza mediante un



Figura 4.29: Visualización de la velocidad promedio.

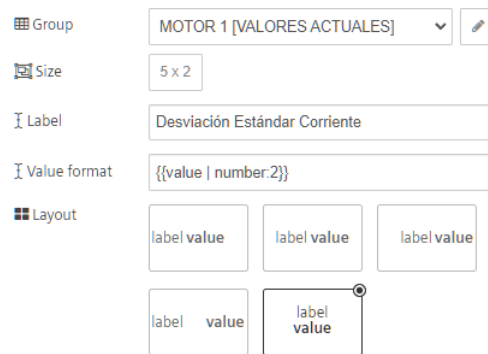


Figura 4.30: Visualización de la desviación estándar.

nodo *text* con la configuración mostrada en la Figura 4.32. Cabe notar que la configuración del valor mostrado varía ya que, en el formato del valor, se especifica que no se visualizarán valores decimales por la naturaleza del dato importado. El proceso de configuración se debe repetir para todos los motores de la red. Se debe resaltar la importancia de este parámetro, pues el tiempo de operación de un equipo es un dato importante dentro de la administración y mantenimiento de un proceso.

#### 4.4. Resultados y Discusión

Al ejecutar la aplicación, el usuario puede observar las variables y los valores previamente configurados en Mindsphere. En este caso se podrá

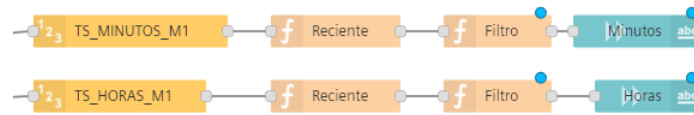


Figura 4.31: Programa de tiempo de funcionamiento.

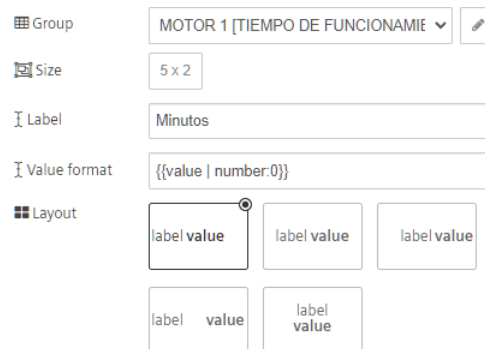


Figura 4.32: Configuración para la visualización del tiempo de funcionamiento.

observar, la velocidad, la corriente, el torque y la potencia de cada motor. En complemento y como insumos para un mantenimiento efectivo, también se pueden visualizar los valores promedio y las desviaciones estándar de las variables antes indicadas. Finalmente, con el mismo propósito se visualiza también el tiempo de operación de cada motor.

En la Figura 4.33 se muestra la información del motor 1, la Figura 4.34 corresponde al motor 2 y la Figura 4.35 ilustra las variables del motor 3. Estas figuras muestran los resultados de una prueba de campo donde se varió la velocidad de los motores un mínimo de 3 veces y se cambió el sentido de giro al menos una vez para evidenciar los cambios en las variables planteadas.

Como se puede evidenciar en las series de tiempo, Mindsphere ofrece una posibilidad de visualización que es muy utilizada en la industria para analizar y comprender el comportamiento de variables en un entorno a lo largo del tiempo. Gracias a este tipo de visualización es posible identificar tendencias, patrones, estacionalidad y anomalías en los datos recopilados. Todos estos datos pueden ser almacenados en históricos y pueden resultar de utilidad para estimar otras variables del motor utilizando técnicas de inteligencia

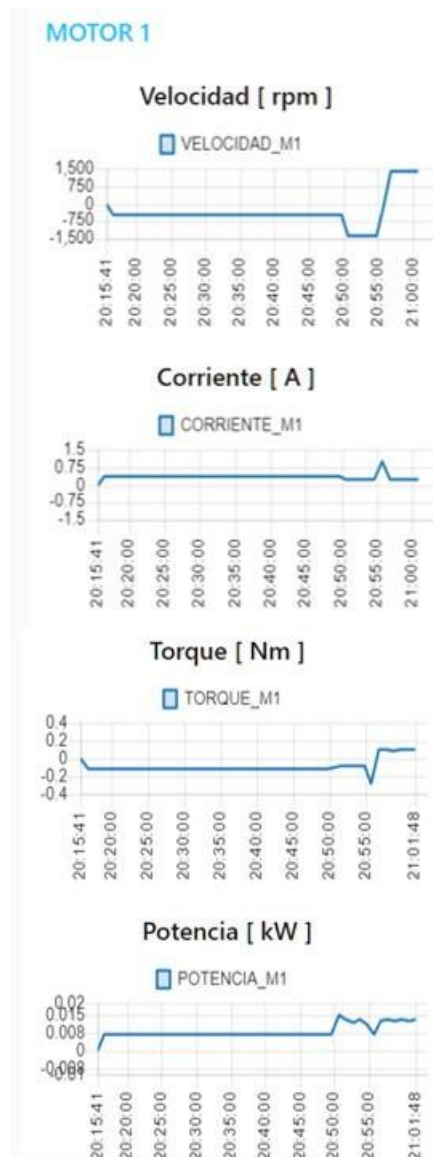


Figura 4.33: Series de tiempo de las variables del motor 1.

artificial.

El panel donde se puede visualizar los valores actuales, promedios y desviaciones estándar se puede observar en la Figura 4.36. Por otra parte,



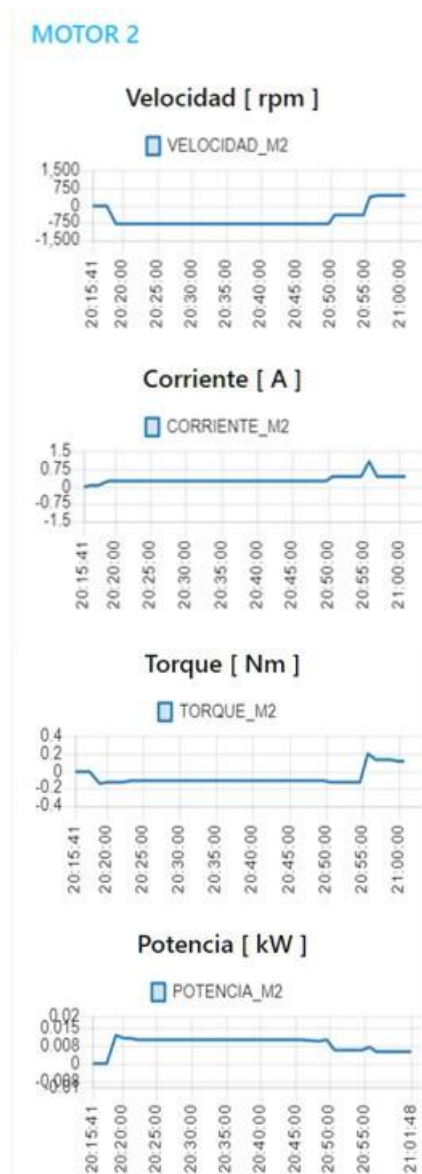


Figura 4.34: Series de tiempo de las variables del motor 2.

el panel para la visualización de los tiempos de operación de los motores se ilustra en la Figura 4.37.

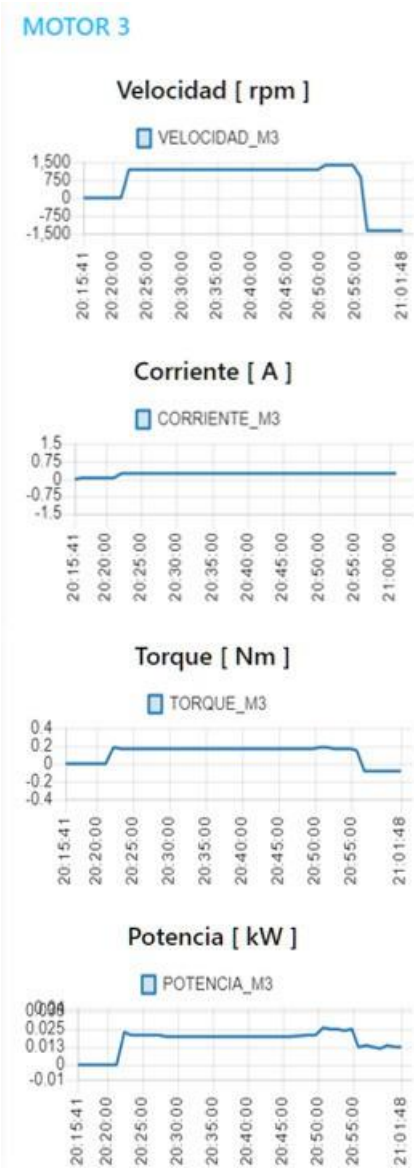


Figura 4.35: Series de tiempo de las variables del motor 3.



Figura 4.36: Visualización de valores actuales, promedios y desviaciones estándar.



Figura 4.37: Panel de tiempos de funcionamiento.

## Capítulo 5

# Conclusiones y trabajos futuros

### 5.1. Conclusiones

En la presente implementación se pudo evidenciar la utilidad de la herramienta MindSphere para la monitorización de las variables de un proceso industrial. El uso de una cuenta gratuita de MindSphere para una aplicación industrial permite tener una idea clara de las posibilidades de implementación pero restringe el número de variables a importar y el ancho de banda disponible para la comunicación. Por esto, se tuvo que limitar los valores, la frecuencia de envío de datos y el número de procesos que se realizan dentro de la plataforma para la presentación de información.

La conexión entre la pasarela SIMATIC IOT2040 y la plataforma IoT resulta relativamente simple, debido a que están diseñadas para trabajar en conjunto. Como es común, será más sencillo desarrollar una implementación que tenga elementos de la misma marca por cuestiones de compatibilidad que se vuelven levemente más complejas al usar equipos de otras marcas. La gran mayoría de plataformas IIoT permiten la interoperabilidad entre marcas, por lo que es importante revisar la documentación de los equipos y plataformas antes de realizar una implementación.

A pesar de que la aplicación *Visual Flow Creator* trabaja en un entorno similar a Node-RED, existen varios nodos que se tienen únicamente en esta plataforma (nodos de IIoT, análisis estadístico, almacenamiento, etc.) y que tienen una descripción de uso muy básica. Esto se debe a que esta información es propietaria de la marca y su detalle está disponible solo para usuarios con licencias pagadas bajo un acuerdo de confidencialidad.

La preparación de las variables para ser ingresadas en la plataforma resulta bastante puntual, con la consideración de que debe haber coherencia a lo

largo de toda la implementación con respecto a tipos de datos, unidades de medida y nombres. En caso de errores, la plataforma no permite continuar con los procesos de configuración, indicando el tipo de error, por lo que la resolución de conflictos es rápida.

La plataforma permite la importación de cualquier tipo de dato, lo que presenta gran versatilidad al momento de la implementación de paneles más completos. En este caso, se tuvo 3 paneles que utilizaron datos de varios tipos para la visualización de las variables, su análisis y para registrar el tiempo de funcionamiento, que permitiría la planificación más precisa de mantenimientos preventivos de las máquinas utilizadas en la red industrial.

La configuración de los equipos de la red industrial resultó simple debido a la gran cantidad de documentación y soporte que existe de parte del fabricante. Esto se debe a que estos han estado en el mercado un tiempo considerable y se han realizado muchas aplicaciones que han dejado recomendaciones útiles para este trabajo, por ejemplo, en la configuración de los tres variadores que trabajaron dentro de la misma red PROFINET.

Esto no se reflejó al momento de trabajar en la implementación de MindSphere, a causa de que IIoT es una tecnología relativamente reciente. Si bien existe documentación de aplicaciones de parte de desarrolladores externos a Siemens, cada autor presenta un proceso diferente al momento de implementar la adquisición de datos y configuraciones, especialmente, al momento de crear los activos del proceso. Esto puede resultar confuso, ya que el fabricante recomienda la configuración de los tipos, variables y activos en ese orden. Pero, algunas implementaciones de desarrolladores externos interrumpen este proceso con otras configuraciones que pueden realizarse luego, como es el caso de la configuración de Node-RED en la pasarela en medio de la configuración de las variables. Este trabajo siguió las recomendaciones del fabricante.

Una ventaja que presenta MindSphere frente a sus competidores es la cantidad de opciones de procesamiento de la información importada, junto con la capacidad de integrar aplicaciones de propias de la plataforma o de desarrolladores independientes con fines específicos. Estas aplicaciones están más orientadas hacia el análisis estadístico a detalle de grandes cantidades de información. En el caso de este trabajo, a pesar de que se utilizaron las herramientas estadísticas básicas disponibles, se pudo obtener información valiosa sobre el proceso como valores promedio y desviaciones estándar que permiten tener información más detallada.

La falta de información con respecto al uso de los nodos dentro de *Visual Flow Creator* podría abrir varios estudios donde se pueda describir a detalle las utilidades de dichos nodos para la presentación, procesamiento,

modificación y exportación de información desde la plataforma.

La presentación gráfica de la información en MindSphere, de manera general, depende de las herramientas disponibles y permite su configuración de manera parcial. A diferencia de otras plataformas, como Grafana por ejemplo, que permite un mayor control del tamaño, apariencia y ubicación de las gráficas. En este caso, MindSphere usa un esquema de columnas para ubicar los elementos gráficos que se integran en los paneles.

Por otro lado, la adquisición de las variables del proceso resultó robusta y, a pesar de importar datos cada minuto por las restricciones de tráfico de la cuenta gratuita, no se evidenció la pérdida de información durante las pruebas realizadas en el laboratorio.

Un punto a favor de la manera de presentar los datos en MindSphere es el gran detalle que da al análisis de series de tiempo con varias herramientas dedicadas que permiten una presentación detallada del comportamiento de las variables tanto en la aplicación *Visual Flow Creator*, que se utilizó para la creación de los paneles, como en la sección de monitoreo interno en *Asset Manager*. La Figura 5.1 muestra un extracto de la visualización en *Visual Flow Creator* (Izquierda) y *Asset Manager* (Derecha) donde se pueden notar las diferencias en la presentación de la variable de velocidad del motor 1. El uso de este último recurso mencionado no se detalla en este trabajo debido a que es una herramienta propia del sistema y no permite la modificación de valores o de sus ejes para una visualización correcta. Pero, es de gran ayuda durante la implementación para revisar si se están importando los datos y si la información que se procesará tiene el formato deseado.

## 5.2. Trabajos futuros

Es importante considerar para trabajos futuros la implementación del control de elementos IIoT de un proceso industrial desde la plataforma para tener una comunicación de dos vías. Teniendo en cuenta que si se requiere un tráfico de datos considerable, va a ser necesario utilizar una cuenta pagada de MindSphere para poder acceder a todas las funcionalidades. También, se puede pensar en implementaciones que utilicen la gran capacidad de procesamiento estadístico, el almacenamiento en la nube y la capacidad de interoperabilidad de la plataforma para integrar dispositivos de otras marcas, ya que estas son las áreas que destacan frente a sus competidores.

También se plantea la posibilidad de explorar nuevas herramientas de la plataforma, como aquellas relacionadas con la Inteligencia Artificial o aquellas que permiten desarrollos propios de usuario a través de Python.

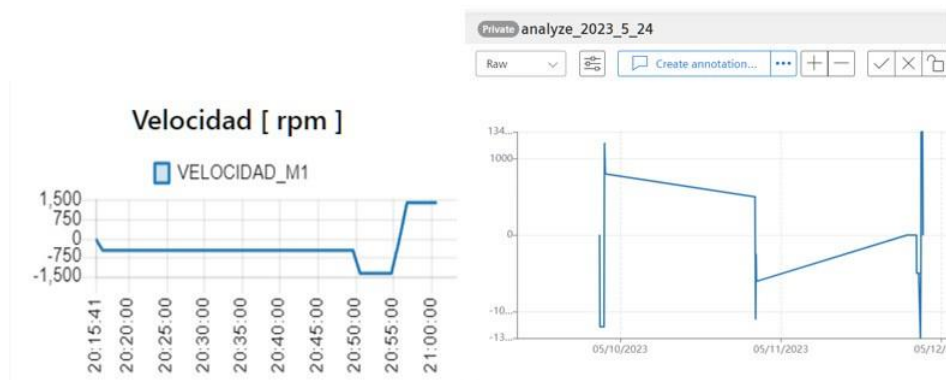


Figura 5.1: Extracto de la visualización en *Visual Flow Creator* (Izquierda) y *Asset Manager* (Derecha).

Esto abrirá un abanico de posibilidades para construir aplicaciones IIoT que pueden seguir aportando al desarrollo de la industria local.



# Bibliografía

- A. H. Abed, M. Nasr, and B. Sayed. The principle internet of things (iot) security techniques framework based on seven levels iot's reference model. In A. Z. Ghalwash, N. El Khameesy, D. A. Magdi, and A. Joshi, editors, *Internet of Things—Applications and Future*, pages 219–237, Singapore, 2020. Springer Singapore. ISBN 978-981-15-3075-3.
- O. Ali and M. K. Ishak. Bringing intelligence to iot edge: Machine learning based smart city image classification using microsoft azure iot and custom vision. *Journal of Physics: Conference Series*, 1529(4):042076, apr 2020. doi: 10.1088/1742-6596/1529/4/042076. URL <https://dx.doi.org/10.1088/1742-6596/1529/4/042076>.
- L. Arnold, J. Jöhnk, F. Vogt, and N. Urbach. Iiot platforms' architectural features – a taxonomy and five prevalent archetypes. *Electronic Markets*, 32:927–944, 6 2022. ISSN 14228890. doi: 10.1007/S12525-021-00520-0.
- Z. Bakhshi, A. Balador, and J. Mustafa. Industrial iot security threats and concerns by considering cisco and microsoft iot reference models. In *2018 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, pages 173–178, 2018. doi: 10.1109/WCNCW.2018.8368997.
- H. J. Cha, H. K. Yang, and Y. J. Song. A study on the design of fog computing architecture using sensor networks. *Sensors (Switzerland)*, 18, 2018. ISSN 14248220. doi: 10.3390/S18113633. URL [https://www.researchgate.net/publication/328550159\\_A\\_Study\\_on\\_the\\_Design\\_of\\_Fog\\_Computing\\_Architecture\\_Using\\_Sensor\\_Networks](https://www.researchgate.net/publication/328550159_A_Study_on_the_Design_of_Fog_Computing_Architecture_Using_Sensor_Networks).
- S. T. Chacko and V. Deshmukh. Iot based online power consumption monitoring of a distribution transformer feeding domestic/commercial consumer loads. *2019 4th International Conference on Information Systems and Computer Networks, ISCON 2019*, pages 441–445, 11 2019. doi: 10.1109/ISCON47742.2019.9036178.

- L. Chuquimarca, A. Asencio, W. Torres, S. Bustos, J. Sánchez, and C. Saldaña. Evaluation of data transfer from plc to cloud platforms-based real-time monitoring using the industrial internet of things. *EAI/Springer Innovations in Communication and Computing*, pages 331–344, 2022. ISSN 25228609. doi: 10.1007/978-3-030-75123-4\_14. URL [https://bibliotecas.ups.edu.ec:3401/chapter/10.1007/978-3-030-75123-4\\_14](https://bibliotecas.ups.edu.ec:3401/chapter/10.1007/978-3-030-75123-4_14).
- R. Conde Camillo da Silva, M. P. Oliveira Camargo, M. Sanches Quessada, A. Claiton Lopes, J. Diassala Monteiro Ernesto, and K. A. Pontara da Costa. An intrusion detection system for web-based attacks using ibm watson. *IEEE Latin America Transactions*, 20(2):191–197, 2022. doi: 10.1109/TLA.2022.9661457.
- D. Ganga and V. Ramachandran. Iot-based vibration analytics of electrical machines. *IEEE Internet of Things Journal*, 5(6):4538–4549, 2018. doi: 10.1109/JIOT.2018.2835724.
- A. Gavlas, J. Zwierzyna, and J. Koziorek. Possibilities of transfer process data from plc to cloud platforms based on iot. *IFAC-PapersOnLine*, 51: 156–161, 1 2018. ISSN 2405-8963. doi: 10.1016/J.IFACOL.2018.07.146.
- A. Gerodimos, L. Maglaras, M. A. Ferrag, N. Ayres, and I. Kantzavelou. Iot: Communication protocols and security threats. *Internet of Things and Cyber-Physical Systems*, 3:1–13, 2023. ISSN 2667-3452. doi: <https://doi.org/10.1016/j.iotcps.2022.12.003>. URL <https://www.sciencedirect.com/science/article/pii/S2667345222000293>.
- H. Golpîra, S. A. R. Khan, and S. Safaeipour. A review of logistics internet-of-things: Current trends and scope for future research. *Journal of Industrial Information Integration*, 22:100194, 2021. ISSN 2452-414X. doi: <https://doi.org/10.1016/j.jii.2020.100194>. URL <https://www.sciencedirect.com/science/article/pii/S2452414X20300698>.
- F. Iqbal, M. Gohar, H. Karamti, W. Karamti, S.-J. Koh, and J.-G. Choi. Use of quic for amqp in iot networks. *Computer Networks*, 225:109640, 2023. ISSN 1389-1286. doi: <https://doi.org/10.1016/j.comnet.2023.109640>. URL <https://www.sciencedirect.com/science/article/pii/S1389128623000853>.
- S. Katsikeas, K. Fysarakis, A. Miaoudakis, A. Van Bemten, I. Askoxylakis, I. Papaefstathiou, and A. Plemenos. Lightweight and secure industrial iot

- communications via the mq telemetry transport protocol. In *2017 IEEE Symposium on Computers and Communications (ISCC)*, pages 1193–1200, 2017. doi: 10.1109/ISCC.2017.8024687.
- M. Krok. Mindsphere v2 for cnc machine tools. *Mechanik*, 91: 658–660, 9 2018. ISSN 0025-6552. doi: 10.17814/MECHANIK.2018. 8-9.103. URL <http://mechanik-science.com/index.php/mechanik/article/view/309>.
- K. E. Kulawiak. Manufacturing the platform economy an exploratory case study of mindsphere, the industrial digital platform from siemens. Master’s thesis, University of Oslo, 2021. URL <http://www.duo.uio.no>.
- V. Kulik and R. Kirichek. The heterogeneous gateways in the industrial internet of things. In *2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pages 1–5, 2018. doi: 10.1109/ICUMT.2018.8631232.
- P. P. Kulkarni, M. Patil, S. Shibi, M. Patle, and R. Kale. Review on online monitoring of electrical machine using iot. *2019 International Conference on Nascent Technologies in Engineering, ICNTE 2019 - Proceedings*, 1 2019. doi: 10.1109/ICNTE44896.2019.8945927.
- R. P. Kumar and S. Smys. A novel report on architecture, protocols and applications in internet of things (iot). *Proceedings of the 2nd International Conference on Inventive Systems and Control, ICISC 2018*, pages 1156–1161, 6 2018. doi: 10.1109/ICISC.2018.8398986.
- C. A. Marino, F. Chinelato, and M. Marufuzzaman. Aws iot analyticsplatform for microgrid operation management. *Computers and Industrial Engineering*, 170:108331, 2022. ISSN 0360-8352. doi: <https://doi.org/10.1016/j.cie.2022.108331>. URL <https://www.sciencedirect.com/science/article/pii/S0360835222003837>.
- G. Marques and R. Pitarma. Environmental quality supervision for enhanced living environments and laboratory activity support using ibm watson internet of things platform. In J. M. F. Rodrigues, P. J. S. Cardoso, J. Monteiro, R. Lam, V. V. Krzhizhanovskaya, M. H. Lees, J. J. Dongarra, and P. M. Sloat, editors, *Computational Science – ICCS 2019*, pages 680–691, Cham, 2019. Springer International Publishing. ISBN 978-3-030-22744-9.

- A. Mijuskovic, I. Ullah, R. Bemthuis, N. Meratnia, and P. Havinga. Comparing apples and oranges in iot context: A deep dive into methods for comparing iot platforms. *IEEE Internet of Things Journal*, 8(3): 1797–1816, 2021. doi: 10.1109/JIOT.2020.3016921.
- B. Mishra and A. Kertesz. The use of mqtt in m2m and iot systems: A survey. *IEEE Access*, 8:201071–201086, 2020. ISSN 21693536. doi: 10.1109/ACCESS.2020.3035849.
- R. Moura, L. Ceotto, A. Gonzalez, and R. Toledo. Industrial internet of things (iiot) platforms - an evaluation model. In *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 1002–1009, 2018. doi: 10.1109/CSCI46756.2018.00194.
- J. M. Müller, D. Kiel, and K.-I. Voigt. What drives the implementation of industry 4.0? the role of opportunities and challenges in the context of sustainability. *Sustainability*, 10(1), 2018. ISSN 2071-1050. doi: 10.3390/su10010247. URL <https://www.mdpi.com/2071-1050/10/1/247>.
- T. Pauli, E. Fiel, and M. Matzner. Digital industrial platforms. *Business and Information Systems Engineering*, 63:181–190, 4 2021. ISSN 18670202. doi: 10.1007/S12599-020-00681-W/TABLES/1. URL <https://link.springer.com/article/10.1007/s12599-020-00681-w>.
- P. Pierleoni, R. Concetti, A. Belli, and L. Palma. Amazon, google and microsoft solutions for iot: Architectures and a performance comparison. *IEEE Access*, 8:5455–5470, 2020. doi: 10.1109/ACCESS.2019.2961511.
- V. Seoane, C. Garcia-Rubio, F. Almenares, and C. Campo. Performance evaluation of coap and mqtt with security support for iot environments. *Computer Networks*, 197:108338, 2021. ISSN 1389-1286. doi: <https://doi.org/10.1016/j.comnet.2021.108338>. URL <https://www.sciencedirect.com/science/article/pii/S1389128621003364>.
- N. Sharma. Evolution of iot to iiot: Applications and challenges. *SSRN Electronic Journal*, 05 2020.
- Siemens. Simatic iot2040 - la pasarela inteligente para las soluciones iot industriales, Oct 2016a. URL <https://support.industry.siemens.com/cs/ec/es/view/109741795>.
- Siemens. Mindsphere - getting started, 2016b. URL <https://support.industry.siemens.com/>.

- S. S. A. Simnani, S. Shah, and M. T. Bandy. Architectural standards for internet of things: From research to industrial adaption. *Proceedings of the 3rd International Conference on Contemporary Computing and Informatics, IC3I 2018*, pages 314–319, 10 2018. doi: 10.1109/IC3I44769.2018.9007292.
- E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund. Industrial internet of things: Challenges, opportunities, and directions. *IEEE Transactions on Industrial Informatics*, 14(11):4724–4734, 2018. doi: 10.1109/TII.2018.2852491.
- J. Soldatos, O. Lazaro, and F. Cavadini. *The digital shopfloor : industrial automation in the industry 4.0 era : performance analysis and applications*. River Publishers, 1 edition, 2019. ISBN 9781000795042.
- R. Stackowiak. *Azure IoT Solutions Overview*, pages 29–54. Apress, Berkeley, CA, 2019. ISBN 978-1-4842-5470-7. doi: 10.1007/978-1-4842-5470-7\_2. URL [https://doi.org/10.1007/978-1-4842-5470-7\\_2](https://doi.org/10.1007/978-1-4842-5470-7_2).
- V. S. Supekar and A. Ahmadina. Sensor data visualization on google maps using aws, and iot discovery board. In *2020 7th International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, pages 1–6, 2020. doi: 10.1109/IOTSMS52051.2020.9340202.
- Y. Tian, F. Gao, and Y. Chai. Application of iot and stresswave analysis in online monitoring and diagnosis of large generating. *2021 4th International Conference on Energy, Electrical and Power Engineering, CEEPE 2021*, pages 704–708, 4 2021. doi: 10.1109/CEEPE51765.2021.9475806.
- N. Q. Uy and V. H. Nam. A comparison of amqp and mqtt protocols for internet of things. In *2019 6th NAFOSTED Conference on Information and Computer Science (NICS)*, pages 292–297, 2019. doi: 10.1109/NICS48868.2019.9023812.
- J. Waterman, H. Yang, and F. Muheidat. Aws iot and the interconnected world – aging in place. In *2020 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 1126–1129, 2020. doi: 10.1109/CSCI51800.2020.00209.
- X. Wu, C. Zhang, and W. Du. An analysis on the crisis of “chips shortage” in automobile industry —based on the double influence of covid-19 and trade friction. *Journal of Physics: Conference Series*, 1971:012100, 7 2021. ISSN 1742-6596. doi: 10.1088/1742-6596/1971/1/012100. URL

<https://iopscience.iop.org/article/10.1088/1742-6596/1971/1/012100><https://iopscience.iop.org/article/10.1088/1742-6596/1971/1/012100/meta>.

- S. Yu-Jie, W. Hui, and Z. Cheng-Xiang. Balanced computing offloading for selfish iot devices in fog computing. *IEEE Access*, 10:30890–30898, 2022. ISSN 21693536. doi: 10.1109/ACCESS.2022.3160198.