



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

DISEÑO DE UNA ARQUITECTURA USANDO EL PROTOCOLO MESSAGE QUEUE TELEMETRY TRANSPORT (MQTT) SOBRE PLATAFORMAS DE BAJO COSTE, PARA MONITORIZACIÓN DE PROCESOS INDUSTRIALES

EDWIN RODRIGO BORJA VEGA

Trabajo de Titulación modalidad: Proyectos de Investigación y Desarrollo, presentado ante el Instituto de Posgrado y Educación Continua de la ESPOCH, como requisito parcial para la obtención del grado de:

**MAGÍSTER EN SISTEMAS DE CONTROL Y AUTOMATIZACIÓN
INDUSTRIAL**

Riobamba - Ecuador

Agosto – 2020

©2020, Edwin Rodrigo Borja Vega

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

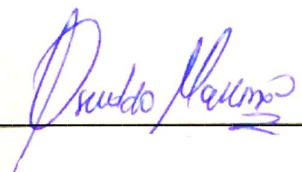
CERTIFICACIÓN:

EL TRIBUNAL DE TRABAJO DE TITULACIÓN CARTIFICA QUE:

El Trabajo de titulación modalidad Proyectos de Investigación y Desarrollo, titulado: “DISEÑO DE UNA ARQUITECTURA USANDO EL PROTOCOLO MESSAGE QUEUE TELEMETRY TRANSPORT (MQTT) SOBRE PLATAFORMAS DE BAJO COSTE, PARA MONITORIZACIÓN DE PROCESOS INDUSTRIALES”, de responsabilidad del señor Edwin Rodrigo Borja Vega, ha sido prolijamente revisado y se autoriza su presentación.

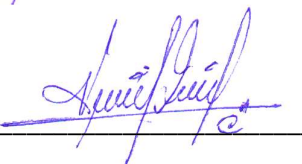
Ing. Oswaldo Geovanny Martínez Guashima; Msc.

PRESIDENTE



Ing. Marcelo Vladimir García Sánchez; PhD.

TUTOR



Ing. Geovanni Danilo Brito Moncayo; Mg.

MIEMBRO DEL TRIBUNAL



Ing. Santiago Mauricio Altamirano Meléndez; Mg.

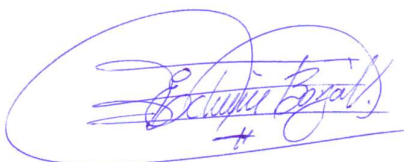
MIEMBRO DEL TRIBUNAL



Riobamba, agosto 2020

DERECHOS INTELECTUALES

Yo, Edwin Rodrigo Borja Vega, declaro que soy responsable de las ideas, doctrinas y resultados expuestos en el Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo, y que el patrimonio intelectual generado por el mismo pertenece exclusivamente a la Escuela Superior Politécnica de Chimborazo.

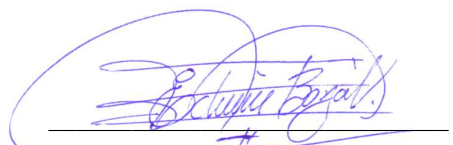


EDWIN RODRIGO BORJA VEGA

No. Cédula 020207662-6

Yo, Edwin Rodrigo Borja Vega, declaro que el Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo, es de mi autoría y que los resultados del mismo son auténticos y originales. Los textos constantes en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor, asumo la responsabilidad legal y académica de los contenidos este proyecto de investigación de maestría.



EDWIN RODRIGO BORJA VEGA

No. Cédula 020207662-6

DEDICATORIA

Este trabajo va dedicado a mi madre y padre (+) en gratitud a su enseñanza, consejo y dedicación en la ardua labor de formar un hombre de bien con valores y principios fundamentados en el amor a Dios.

Rodrigo

AGRADECIMIENTO

A Marcelo García por la apertura y apoyo incondicional en el desarrollo del presente trabajo, del mismo modo a Santiago Altamirano y Geovanny Brito quienes sin dudar decidieron brindarme el apoyo para la codirigir este proyecto, a Carlos García un gran profesional y amigo quien compartió sus conocimientos conmigo para lograr este objetivo y un agradecimiento especial a mi novia Lissette por cada palabra de aliento y su apoyo que sirvió para culminar este trabajo de titulación, gracias de todo corazón.

Rodrigo

CONTENIDO

RESUMEN.....	xiii
ABSTRACT.....	xiv

CAPÍTULO I

1.	INTRODUCCIÓN	1
1.1	Planteamiento del problema	1
1.1.1	<i>Situación problemática</i>	1
1.2	Formulación del problema.....	2
1.3	Preguntas específicas	2
1.4	Justificación de la investigación	2
1.5	Objetivos de la investigación.....	4
1.5.1	<i>Objetivo general</i>	4
1.5.2	<i>Objetivos específicos</i>	4
1.6	Hipótesis	4

CAPÍTULO II

2.	MARCO REFERENCIAL	5
2.1	Antecedentes del problema	5
2.2	Bases teóricas	7
2.3	Marco conceptual	9
2.3.1	<i>Industria 4.0</i>	9
2.3.2	<i>Internet de las cosas</i>	11
2.3.3	<i>Internet Industrial de las cosas</i>	13
2.3.4	<i>Sistemas ciberfísicos (CPS)</i>	13
2.3.5	<i>Sistemas ciberfísicos de producción (CPPS)</i>	14
2.3.6	<i>Protocolo Message Queue Telemetry Transport (MQTT)</i>	15
2.3.7	<i>Plataformas de hardware de bajo costo</i>	18
2.3.8	<i>Snap 7</i>	20
2.3.9	<i>MySQL</i>	22
2.3.10	<i>APACHE</i>	22
2.3.11	<i>GOOGLE CHARTS</i>	23
2.3.12	<i>WIRESHARK</i>	23

CAPÍTULO III

3.	METODOLOGÍA DE LA INVESTIGACIÓN	24
3.1	Tipo y Diseño.....	24
3.2	Método de Investigación.....	24
3.3	Enfoque de la Investigación	24
3.4	Alcance de la investigación	24
3.5	Población de Estudio	25
3.6	Unidad de Análisis.....	25
3.7	Técnica de recolección de datos primarios y secundarios	25
3.8	Instrumentos de recolección de datos primarios y secundarios.....	25
3.9	Operacionalización de variables.....	24
3.10	Matriz de consistencia	27

CAPÍTULO IV

4.	PRUEBAS Y RESULTADOS	28
4.1	Resultados de la monitorización con una arquitectura utilizando MQTT.....	28
4.1.1	<i>Análisis de velocidad de transmisión</i>	28
4.1.2	<i>Prueba de hipótesis.....</i>	30
4.2	Trabajos futuros	33

CAPÍTULO V

5.	PROPUESTA	34
5.1	Diseño de una arquitectura para moniorización	34
5.1.1	<i>Descripción de requerimientos.....</i>	34
5.1.2	<i>Componentes de la arquitectura.....</i>	34
5.2	Programas y selección de equipos	35
5.3	Caso de aplicación	37
5.3.1	<i>PLC y BeagleBone Black 1-Bróker.....</i>	38
5.3.2	<i>Aplicación del protocolo MQTT y BeagleBone Black 2-Client</i>	38
5.3.3	<i>Topología MQTT</i>	39
5.3.4	<i>Topic</i>	40
5.3.5	<i>Transmisión de mensajes, publicación - suscripción</i>	40
	CONCLUSIONES.....	42
	RECOMENDACIONES.....	43

BIBLIOGRAFÍA

ANEXOS

ÍNDICE DE TABLAS

Tabla 1-2:	Comparación de protocolos para IoT.....	15
Tabla 2-2:	Características de plataformas de hardware de bajo coste.....	18
Tabla 3-2:	Funciones administrativas para control de objeto cliente.....	21
Tabla 4-2:	Funciones E/S para intercambio de datos con el PLC.....	21
Tabla 1-4:	Muestras capturadas por protocolo.....	30
Tabla 2-4:	Cálculo de rangos para la prueba de hipótesis U de Mann – Whitney.....	31
Tabla 3-4:	Cálculo de la prueba de hipótesis U de Mann – Whitney.....	32
Tabla 4-4:	Prueba de la hipótesis, valoración de variables	32
Tabla 1-5:	Características de controladores para el proceso.....	35
Tabla 2-5:	Tabla comparativa de SBC.....	36
Tabla 3-5:	Equipos y software utilizado en la propuesta.....	36
Tabla 4-5:	Direcciones IP utilizadas en la propuesta	37

ÍNDICE DE FIGURAS

Figura 1-2:	Evolución de la industria.	10
Figura 2-2:	Tecnologías de la industria 4.0.	11
Figura 3-2:	Funcionamiento del protocolo MQTT.	16
Figura 4-2:	BeagleBone Black.	19
Figura 1-4:	Sistema de comunicación y captura con wireshark.	28
Figura 2-4:	Aplicación de filtro de protocolo MQTT	29
Figura 3-4:	Mensajes de protocolo MQTT filtrados en Wireshark.	29
Figura 4-4:	Herramienta protocol hierarchy	29
Figura 5-4:	Identificación de velocidad de protocolo MQTT.	29
Figura 6-4:	Identificación de velocidad de protocolo TCP.	30
Figura 1-5:	Caso de aplicación - Estación de clasificación Festo.	37
Figura 2-5:	Estructura de la propuesta y plataformas.	39
Figura 3-5:	Topología MQTT.	40
Figura 4-5:	Publicación de mensaje con MQTT - Bróker.	42
Figura 5-5:	Recepción del mensaje por suscripción - Cliente.	43

ÍNDICE DE GRÁFICOS

Gráfico 1-2:	Análisis del problema.....	6
Gráfico 2-2:	Expectativa de objetos conectados al 2020	12
Gráfico 3-2:	CPS, interacción humano máquina.....	14
Gráfico 4-2:	Estructura de un paquete de control MQTT.....	16
Gráfico 5-2:	Formato de encabezado fijo	16
Gráfico 1-5:	Arquitectura propuesta y sus componentes.....	35
Gráfico 2-5:	Diagrama de flujo, BeagleBone Black 1-Bróker.....	41
Gráfico 3-5:	Diagrama de flujo, BeagleBone Black 2-Cliente.....	42
Gráfico 4-5:	Presentación del mensaje en un Dashboard en el web service.....	43

RESUMEN

Este trabajo tuvo como objetivo diseñar una arquitectura para la monitorización de procesos industriales utilizando el protocolo MQTT sobre plataformas de bajo costo, como lo es la BeagleBone Black, alineado con la Industria 4.0 y aplicación del Internet de Cosas (IoT). MQTT es un protocolo de mensajería publicación suscripción cuya comunicación está basada en tópicos, con un esquema de operación cliente servidor. La arquitectura planteada se ejecuta sobre una aplicación de laboratorio para la clasificación de piezas por color, teniendo como objetivo la monitorización de dicha clasificación, el controlador del proceso es un PLC y la arquitectura de comunicación consta de dos clientes y un bróker con una topología tipo estrella, la funcionalidad de la propuesta es demostrada mediante pruebas de funcionamiento y comunicación, concluyéndose que el uso del protocolo MQTT sobre plataformas de bajo costo permite la monitorización de procesos industriales utilizando una página web y se obtiene velocidades de transmisión óptimas, esto se demuestra al identificar la velocidad de transmisión mediante la captura de paquetes usando Wireshark, tanto para el protocolo MQTT y TCP, los datos de las muestras obtenidas se someten a la prueba de hipótesis U de Mann Whitney con un nivel confianza del 95%, donde como resultado se obtiene, un nivel de significancia calculado de 0.017 menor al 0.05, por tanto, se rechaza la hipótesis nula y se acepta la hipótesis propuesta, debido a que se obtienen velocidades de transmisión superiores de MQTT sobre TCP, por ser un protocolo ligero y simple, que además tiene por característica ser escalable, este tipo de arquitectura es ideal para entornos restringidos.

Palabras clave: <MESSAGE QUEUE TELEMETRY TRANSPORT (MQTT)>, <SNAP 7>, <MONITORIZACION INDUSTRIAL>, <BEAGLEBONE BLACK>, <PLATAFORMA DE BAJO COSTO>, <MOSQUITTO (SOFTWARE)>, <PUBLICACIÓN-SUSCRIPCIÓN >, <INTERNET OF THINGS>.



ABSTRACT

The objective of this work was to design an architecture for the monitoring of industrial processes using the MQTT protocol on low-cost platforms, such as the BeagleBone Black, aligned with Industry 4.0 and the application of the Internet of Things (IoT). MQTT is a subscription-based publication messaging protocol whose communication is based on topics, with a client-server operation scheme. The proposed architecture is executed on a laboratory application for the classification, the process controller is a PLC and the communication architecture consists of two clients and a broker with a star type topology, the functionality of the proposal is demonstrated through operational and communication tests, concluding that the use of MQTT protocol on low cost platforms allows the monitoring of industrial processes using a web page and optimal transmission speeds are obtained, this is demonstrated by identifying the transmission speed by means of packet capture using Wireshark, both for MQTT and TCP protocol, the data of the samples obtained are submitted to Mann Whitney's U hypothesis test with a 95% confidence level, where as a result, a significance level calculated of 0.017 is obtained less than 0.05, therefore, the null hypothesis is rejected and the proposal hypothesis is accepted, because higher transmission speeds of MQTT over TCP are obtained, since it is a light and simple protocol, which also has the characteristic of being scalable, this type of architecture is ideal for restricted environments.

Key words: < MESSAGE QUEUE TELEMETRY TRANSPORT (MQTT)>, <SNAP 7>, <INDUSTRIAL MONITORING>, <BEAGLEBONE BLACK>, <LOW COST PLATFORM>, <MOSQUITTO>, <PUBLICATION-SUCRIPTION>, <INTERNET OF THINGS>



CAPÍTULO I

1. INTRODUCCIÓN

El presente capítulo detalla de forma clara el problema, planteamiento y formulación de la investigación, además se incluye la justificación, los objetivos generales, específicos y la hipótesis de la investigación.

1.1 Planteamiento del problema

1.1.1 *Situación problemática*

En la última década han surgido conceptos visionarios que buscan proyectar el desarrollo de un nuevo hito del ingenio de la humanidad, hace alrededor de 200 años se ideó una máquina que mecanizaba tareas, era la máquina de vapor, con ella llegó la primera revolución industrial, luego con el avance de la ciencia se obtuvo la electricidad y se consiguió optimizar la fabricación para producir en masa, más tarde, el desarrollo de la electrónica y la informática permitieron que las tareas se ejecuten de forma automática, hoy la mente humana se embarca en la cuarta revolución industrial, donde las tecnologías de fabricación se conjugan y viven cambios para alcanzar el paradigma de la Industria 4.0, siendo el tema más debatido por los profesionales de la industria y académicos (Kagermann et al., 2013).

La cuarta revolución industrial encapsula tendencias para el desarrollo de la industria con el objetivo de lograr procesos de fabricación más inteligentes e integrados, cuyas tecnologías clave son los sistemas ciberfísicos (CPS), cloud computing, big data, internet móvil e Internet de las Cosas (IoT), estos dos últimos, permitirán que exista interacción entre humanos y máquinas, estas innovaciones darán lugar a muchas oportunidades y desafíos, entre ellos será obtener una ventaja competitiva en calidad y eficiencia con bajo costo (Zhou, 2015). La fábrica inteligente busca mejorar significativamente la utilización de recursos, cuyo objetivo es integrar el mundo físico con el mundo virtual, cualquier parte de la fábrica se puede integrar con el uso de sistemas embebidos y otras tecnologías (Wan et al., 2015). En el desarrollo de la industria se sugiere la implementación de entornos de trabajo digitales asistidos, dando lugar a los sistemas ciberfísicos de producción (CPPS) (Kerpen et al., 2016).

El internet de las cosas es otro de los conceptos que apuntalan el rumbo del desarrollo, que a futuro tiene como expectativa una alta cantidad de dispositivos conectados a internet, por lo que

la escalabilidad, el ancho de banda y las limitantes de uso de memoria supone un reto importante, como posible solución se propone el uso de protocolos como Constrained Application Protocol (COAP) y Message Queue Telemetry Transport (MQTT) (Frigieri et al., 2015).

La automatización busca nuevos enfoques flexibles y eficientes para las operaciones de producción, los ordenadores de placa reducida (SBC) denotan capacidades de conectividad y rendimiento computacional para sistemas de control industrial, dando lugar a una arquitectura CPPS de bajo costo (García, 2016).

El trabajo propuesto tiene tres puntos principales: 1) Analizar las características del protocolo MQTT para transmisión de datos; 2) Esquematizar una arquitectura que permita monitorizar un proceso industrial a través del protocolo MQTT; 3) Desarrollar una aplicación sobre plataformas de hardware de bajo costo para monitorización de procesos de producción. Las pruebas experimentales se realizan sobre una aplicación de laboratorio con el objetivo de ilustrar un sistema de automatización industrial.

1.2 Formulación del problema

Los procesos industriales requieren de monitorización remota, donde el ancho de banda y las velocidades de transmisión de datos son un reto que se debe considerar al implementarse en medios físicos de transmisión, por lo que una alternativa para la monitorización es el uso del protocolo MQTT de fácil implementación y útil para la industria 4.0.

1.3 Preguntas específicas

- ¿Cómo operan los sistemas ciberfísicos de producción y las últimas tendencias en la industria?
- ¿Cómo funciona el protocolo MQTT?
- ¿Cómo monitorizar de manera remota un proceso industrial a través del protocolo MQTT y una Beaglebone Black?
- ¿Qué velocidad tiene MQTT en relación a TCP IP, usualmente utilizado en la industria?

1.4 Justificación de la investigación

Las tendencias tecnológicas y la forma de vivir han dado un salto cualitativo hacia el futuro, donde todo lo que nos rodea está conectado: personas, datos y máquinas; ahora la industria propende a desarrollarse sobre dicha tendencia, donde el concepto de consumo tiene nuevas formas de

fabricar, mediante una transformación de sistemas de software, sensores, potentes procesadores y novedosas tecnologías de comunicaciones que se han denominado Industria 4.0

El concepto Industria 4.0 se relaciona con varios otros, denotando avances significativos en lo relacionado a la flexibilidad e individualización de la producción y sus procesos, dentro de este paradigma los procesos interactúan entre sí y con procesos externos, generándose una gran cantidad de datos que deben ser manipulados para convertirse en una información útil, creando escenarios para el desarrollo de nuevos profesionales ligados a la electrónica y la automática (Catalán et al., 2015).

En el entorno industrial la disponibilidad es la prioridad absoluta, luego tenemos la integridad y la confidencialidad de la información, al ser un punto sensible de la industria 4.0, se requiere el uso de protocolos de comunicación de bajo coste y bajos consumos de ancho de banda, que sean flexibles y permitan conseguir niveles de seguridad adecuados para la transmisión de los datos de una planta.

El protocolo MQTT es un protocolo abierto de conectividad Machine to Machine (M2M), diseñado como transporte de mensajería ultraligero, proyectándose como el ideal para su aplicación en el Internet de las cosas, dando facilidades para su incorporación en sistemas embebidos, este protocolo en la actualidad, después de un proceso de normalización ha sido declarado como un estándar OASIS (mqtt-v3.1.1, 2014).

Un Sistema Ciberfísico (CPS) tiene software embebido y una alta flexibilidad con capacidades de computación y almacenamiento con el fin de controlar y/o monitorizar objetos del mundo exterior, permitiendo desarrollar una nueva concepción de soluciones (Acatech, 2011). Según (Kagermann et al., 2013) se puede decir que la Industria 4.0, es en esencia la integración técnica de los sistemas ciberfísicos, en la fabricación y en el uso de internet en los procesos industriales.

El desarrollo de la industria busca capturar todos los datos de forma digitalizada y correlacionada, de manera que se permita un análisis en tiempo real, para poder evaluar, predecir y tomar decisiones, siendo necesario que la información generada se distribuya en forma eficiente por toda la planta e incluso de manera remota fuera de ella.

Por lo descrito y con el afán de crear facilidades flexibles para la industria en general, éste trabajo tiene como objetivo monitorizar procesos industriales en aplicación del protocolo Message Queue Telemetry Transport (MQTT) sobre plataformas de hardware de bajo costo, *i.e.*, usando ordenadores de placa reducida establecer una arquitectura, que facilite el diseño para futuras

aplicaciones, sobre las cuales se incorpora el protocolo de comunicación MQTT, mismo que se implementará sobre TCP-IP para interactuar con los procesos de producción existentes en la red mediante aplicaciones móviles o web.

1.5 Objetivos de la investigación

1.5.1 *Objetivo general*

Diseñar una arquitectura que permita monitorizar procesos industriales a través del protocolo MQTT, usando plataformas de hardware de bajo costo.

1.5.2 *Objetivos específicos*

- Analizar las características del protocolo MQTT y su aplicación en el IoT, para adaptarlo a los requerimientos de la industria.
- Desarrollar una aplicación sobre plataformas de hardware de bajo costo, mediante el uso del protocolo MQTT, para monitorización de procesos de producción, utilizando como referencia un equipo de laboratorio con el objeto de ilustrar un sistema de automatización industrial.
- Evaluar las velocidades de transmisión del protocolo MQTT y del protocolo TCP IP usualmente utilizado en aplicaciones industriales.

1.6 Hipótesis

Implementar una arquitectura basada en el protocolo MQTT sobre plataformas de hardware de bajo costo permitirá monitorizar de forma remota los procesos industriales y se obtendrá velocidades de transmisión superiores a las del protocolo TCP IP.

CAPÍTULO II

2. MARCO REFERENCIAL

2.1 Antecedentes del problema

En el mundo las empresas se ven obligadas a reconfigurar sus procesos, teniendo que adaptarse con la competitividad global, innovación y desarrollo tecnológico, buscando darles un mayor valor a las necesidades de sus clientes. La constante evolución industrial y el aporte de varios investigadores, que hasta hace años atrás se miraban lejanas en el horizonte, han permitido que la comunidad científica tome interés en los conceptos del paradigma Industria 4.0, IoT y CPPS, apalancados en las plataformas de hardware comercial de bajo coste, con el fin de realizar planteamientos novedosos para la industria y sus formas de producción, mismos que buscan obtener ciertas ventajas, como la flexibilidad y su proyección a la optimización de recursos.

Estados Unidos, Dinamarca, Reino Unido, Corea del Sur trabajan constantemente en el desarrollo de estrategias de fabricación inteligente, sin embargo; la más destacada es la iniciativa alemana denominada “Industria 4.0”, tal es así que la Unión Europea tiene como objetivo incrementar el peso de la industria, de manera que represente un 20% en el PIB para el 2020, teniendo como meta dar atención a los mercados basados en la personalización, creación de productos nuevos y servicios innovadores.

La realidad latinoamericana es otra en la que ningún país está dentro de las grandes potencias mencionadas, sin embargo; el factor económico no es lo dominante para una transformación industrial, en la cual tenemos herramientas como la investigación, nuevas tecnologías e iniciativas de fabricación que se pueden utilizar para satisfacer las necesidades de los clientes y adaptarse a los nuevos modelos de negocio de la industria (Val Román, 2017).

Latinoamérica y Ecuador tienen una oportunidad para aprovechar las posibilidades de evolucionar y desarrollarse, mediante una colaboración articulada de la investigación y la industria en uso de herramientas como el internet, la digitalización y las nuevas tecnologías que permiten una transformación hacia un modelo productivo basado en internet, que, sin duda se convierte un desafío estrictamente necesario para buscar competitividad en el mercado global.

Conforme lo expuesto, con el objetivo de brindar una herramienta para la monitorización de procesos industriales, el problema de la investigación se centra en la falta de monitorización de procesos industriales que mantenga armonía con los nuevos conceptos de la industria.

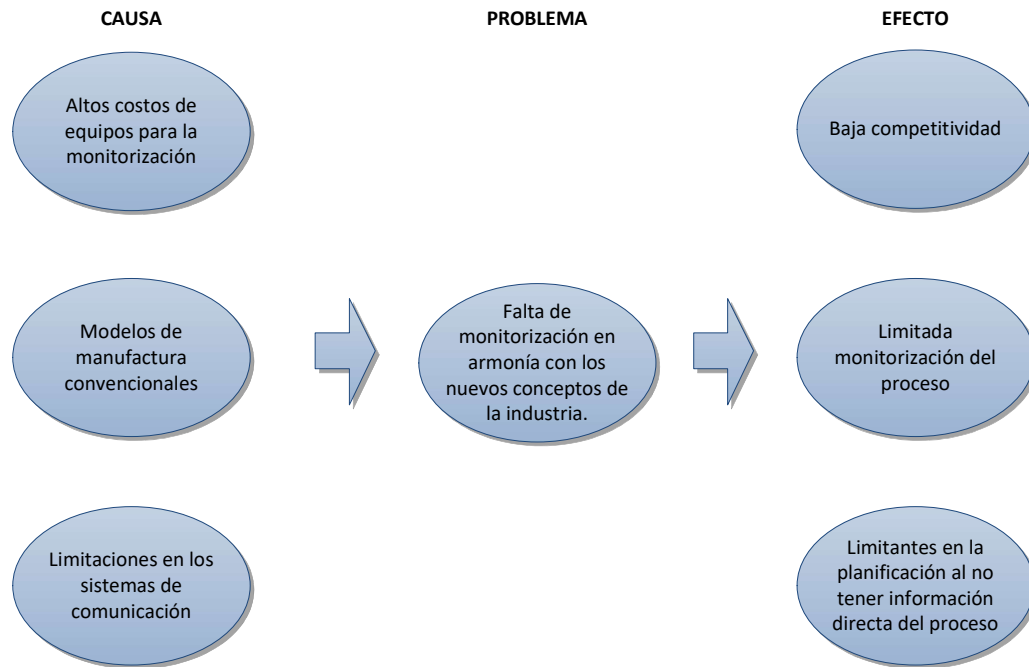


Gráfico 1-2: Análisis del problema.

Realizado: Borja Edwin, 2020.

Como se describe en el gráfico 1-2 el problema suscita en una época en la que la información se convierte en uno de los bienes más valiosos para la toma de decisiones, la industria ecuatoriana tiene limitaciones en la disponibilidad de los datos, debido a una falta de monitorización integrada y de acceso para diferentes niveles de gestión de un proceso productivo.

Por una parte, los altos costos de los equipos de comunicación y monitorización impiden mantener una característica adecuada para los procesos industriales, restándole competitividad con relación a la industria internacional. La demanda en crecimiento de los diferentes productos hace que la industria busque a toda costa cumplir con la oferta que requiere el mercado, esto da cabida a la adquisición de maquinaria e incremento de la mano de obra, dando lugar a procesos de manufactura no convencionales y por ende su monitorización tiene limitaciones, que no se adecuan a las exigencias del mercado actual.

Así también la limitación en los sistemas de comunicación entre procesos, hace que no exista una planificación adecuada, puesto que no se cuenta con una disponibilidad de los datos de los procesos productivos que permitan establecer metas y objetivos para el desarrollo de la empresa y sobre todo que estos estén alineados con una mejora continua y optimización de recursos.

2.2 Bases teóricas

Realizada la búsqueda en artículos y proyectos afines al tema, se presenta lo relevante de esta información:

El concepto de industria 4.0 marca un hito para la transformación de la manufactura, es así que hoy en día se habla de la cuarta revolución industrial, sin duda esto genera expectativa conforme las necesidades de los clientes se vuelven más exigentes con el pasar de los días, por lo cual la industria busca adaptarse con productos personalizados, mayor valor agregado y sobre todo permitir la interacción entre humanos y máquinas; para las grandes potencias el satisfacer esta necesidad incluso se ha convertido en parte de la política económica con el afán de trazar estrategias de alta tecnología y digitalización de procesos para lograr un intercambio dinámico de la información(Deloitte, 2015; Val Román, 2017; Zhou, 2015).

La industria 4.0 tiene como base tecnologías como el big data, robots autónomos, Internet de las cosas, realidad aumentada, la nube, la ciber seguridad, los sistemas de integración horizontal y vertical entre otras, cuyo objetivo es lograr un trabajo conjunto en el que la industria se transforma adaptándose al comportamiento de sus consumidores y que por su puesto esto se refleja en una nueva forma de hacer negocios, estos cambios tecnológicos implican también un cambio social en el que las estructuras empresariales se ven modificadas ante estos nuevos entornos de innovación (Blanco et al., 2017; Casalet R., 2017; Val Román, 2017).

Actualmente las grandes potencias ya se encuentra en aplicación y desarrollo continuo de sistemas ciberfísicos de producción para el desarrollo de un sistema de ejecución de fabricación (MES) sobre la red de una empresa, lo cual ha dado como resultado una mejora efectiva del nivel de gestión empresarial permitiendo optimizar recursos haciendo contacto con el plan de producción real, de manera que los objetivos de producción se alinean con la planificación y las políticas de una empresa (Ruan et al., 2016).

La búsqueda constante de la competitividad ha provocado que los procesos industriales sufran cambios de manera acelerada en el que no exista una planificación e integración de estos procesos, como es el caso de Ecuador, con esta transformación emergen necesidades como la supervisión y monitorización de los procesos, hoy el internet de la cosas es una de las técnicas más favorables, que en combinación de sistemas embebidos y sistemas de comunicación permite que los sistemas industriales estén conectados a internet, lo cual da la posibilidad de monitorizarlos a través de ordenadores y equipos móviles, dando ventajas como modularidad, facilidad de mantenimiento, reacción sobre fallas y monitorización remota (Shinde et al., 2017).

Para lograr una monitorización a través de sistemas embebidos y que esta se adapte a las necesidades de las diferentes empresas es factible el uso de Single Board Computer's (SBCs) y librerías de software libre, con las cuales es posible desarrollar un sistema que se comunique con una máquina para su monitorización, pudiéndose recoger datos para análisis de productividad, mantenimiento y control, de manera que permitan alcanzar objetivos empresariales con información directa del proceso (Ríos et al, 2011).

Para lograr una monitorización apropiada existen trabajos de investigación en los que se establece que el uso de normas para formar arquitecturas, permitiendo una integración vertical adecuada para la supervisión de los procesos industriales, mediante el desarrollo de sistemas ciberfísicos de producción que colaboran a la gestión de todo el proceso de la cadena de valor (Irisarri et al., 2016).

Un limitante para la monitorización suele ser el costo del ancho de banda y la gran cantidad de datos que se obtienen de los procesos, sin embargo, para ello se presenta como opción nuevos protocolos más livianos y sencillos que facilitan el desarrollo e implementación de sistemas que buscan este objetivo, considerando que en la era de la información la humanidad se sumerge en grandes cantidades de datos, donde el servicio de información efectiva y activa, mejora el acceso a la misma y su adecuación sobre aplicaciones móviles, para ello el protocolo MQTT es una buena opción que hace posible la comunicación de mensajes en tiempo real a sus clientes (Tang et al., 2013).

El mundo de hoy es el del internet, tal es así, que ha tenido un fuerte impacto en la ciencia, educación, comunicaciones y negocios, por lo que cada persona está conectada con sus dispositivos móviles al internet, para el año 2020 se estima tener alrededor de 50 MM de dispositivos conectados, el IoT se establece como un sistema integrado de comunicación que permite conectar dispositivos a la red y a su vez estos se pueden utilizar para aplicaciones industriales a pequeña escala, en el que se monitoree de forma inalámbrica, sin duda esto permitirá tener beneficios de modularidad, eliminación de cableado y fácil mantenimiento (Evans et al., 2011; Shinde et al., 2017)

Como punto de partida para la investigación se analiza la implementación de la monitorización de un sensor Light Dependent Resistor (LDR) y un sensor de temperatura y humedad (DHT11) usando un chip integrado con WiFi (ESP8266) con capacidad de integrarse a la red; los sensores y un computador se conectan al chip integrado en calidad de cliente, mientras que la aplicación Adafruit.io es el servidor MQTT que hace las veces de intermediario para la supervisión de los sensores descritos (Kodali et al., 2016).

En todo proceso la monitorización tiene un papel importante, en especial a nivel de la industria, con su evolución da lugar a nuevas necesidades, es por ello que las investigaciones descritas tienden a buscar soluciones alineadas a estos conceptos vanguardistas, relacionados con la Industria 4.0, IoT, IIoT, sistemas ciberfísicos y los sistemas ciberfísicos de producción; mediante el uso de protocolos como MQTT, considerado como ideal para transmisión de mensajes en tiempo real, y el uso de ordenadores de placa reducida, lo que otorga facilidades flexibles para la industria, éste trabajo tiene por objetivo proponer una arquitectura que permita monitorizar procesos industriales en aplicación del protocolo en MQTT sobre plataformas de hardware de bajo coste.

En virtud de lo expuesto y conforme los antecedentes investigativos, el tema sujeto de análisis y la propuesta de solución, guardan relación con la necesidad de otorgar herramientas que respondan como alternativas para la innovación y desarrollo de la industria actual.

2.3 Marco conceptual

2.3.1 *Industria 4.0*

El concepto de industria 4.0 surge en la segunda década del siglo XXI de la mano del gobierno Alemán, que tiene por antecedente no solo un cambio tecnológico, sino también los modelos económicos de desarrollo, donde las economías emergentes crecieron en el periodo comprendido entre 1991 y 2011 (Serna, 2017). Las tres primeras revoluciones industriales tuvieron como protagonistas, en la primera a la máquina de vapor y los motores de combustión, en la segunda la energía eléctrica y la división del trabajo y en la tercera a inicios de los setentas estuvo marcada por el desarrollo de la electrónica y las TIC's, como lo muestra la figura 1-2 (Bauernhansl et al., 2014).

El término Industria 4.0 se refiere a un hito más en el desarrollo de la organización y gestión de la producción, también denominado “Cuarta revolución industrial”, donde una de las características es establecer una red vertical de los sistemas de producción, utilizando sistemas CPPSs con la finalidad de reaccionar a cambios de demanda y a fallos (Deloitte, 2015).

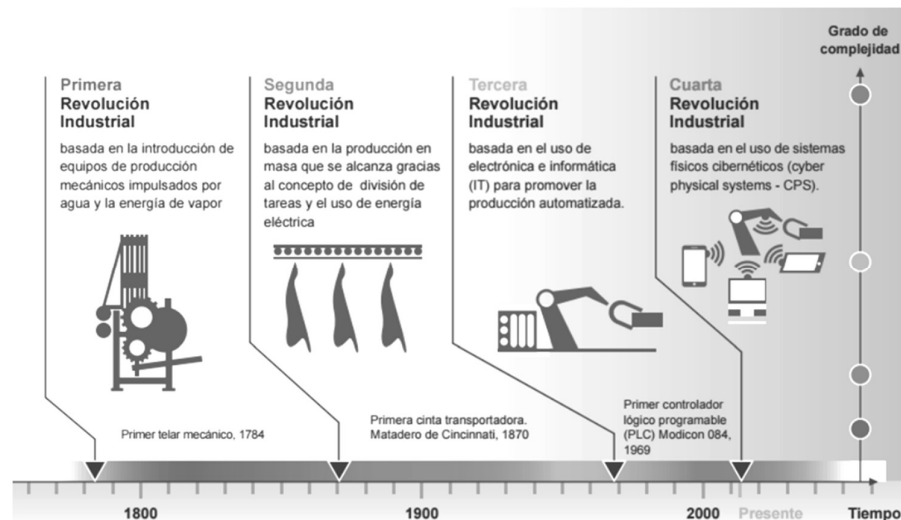


Figura 1-2: Evolución de la industria.
Fuente:(Blanco et al., 2017).

Los avances tecnológicos constituyen una gran inversión que hace posible un cambio sobre los procesos de producción convencional, propendiendo a la producción descentralizada, la Industria 4.0 no solo simplemente prevé una máquina que procesa un producto, sino un producto que se comunica con la máquina, transformando la cadena valor, los modelos comerciales y de producción (Sendler, 2013).

El Ministerio Federal de Educación e Investigación de Alemania ha venido impulsando el proyecto Industria 4.0, en referencia a los sistemas de monitorización inteligente, planteado con el avance de la automatización, es prioritario el desarrollo de procesos de monitorización inteligentes y sistemas con autonomía en la toma de decisiones, optimizando procesos en la cadena de valor con el objetivo de mejorar la fiabilidad y eficiencia en la producción y suministro de productos (“Industrie 4.0 - BMBF,” 2018).

La industria 4.0 se fundamenta en nueve tecnologías, como lo muestra la figura 2-2, que en la actualidad se están utilizando en la manufactura de manera aislada, sin embargo el World Economic Forum (2016) añade a la cuarta revolución industrial adelantos en genética, nanotecnología y biotecnología, afirmando que la construcción de los sistemas inteligentes, aportara a la solución de una gran variedad de problemas (Blanco et al., 2017).

La respuesta del paradigma va orientada al ahorro de energía, gestión de recursos naturales y humanos, apalancado sobre una red de comunicaciones con un intercambio de información permanente e instantánea, permitiendo una gestión eficaz que reflejara ventajas en productividad y economía de recursos; sin embargo el concepto de la industria 4.0 es muy complejo y poco

conocido por las Pymes, en tal sentido se hace necesario formular políticas y estrategias que posibiliten el desarrollo industrial de la nueva era (Serna, 2017).

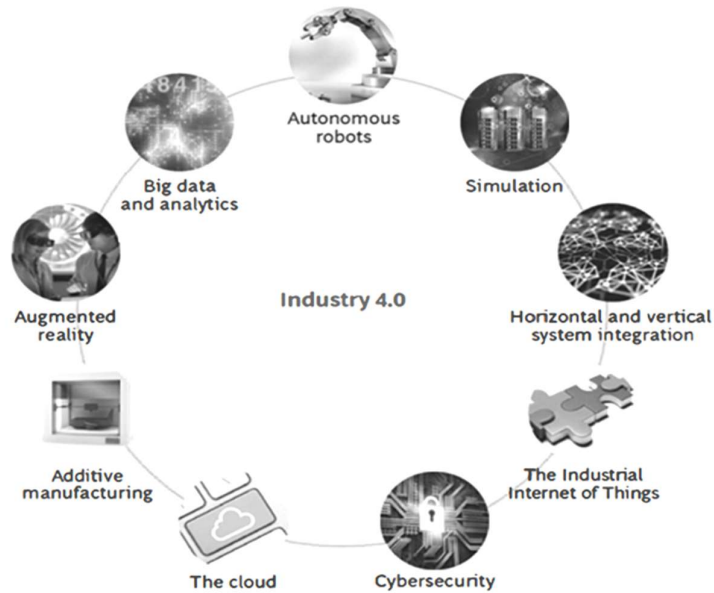


Figura 2-2: Tecnologías de la industria 4.0.
Fuente:(Blanco et al., 2017).

Las consecuencias de la industria 4.0 no representan solo cambios tecnológicos sino también cambios sociales al modificarse la estructura organizacional de las empresas con el afán de construir un entorno de innovación; la sinergia de las tecnologías proporciona adaptabilidad y flexibilidad para la producción personalizada, siendo determinante la combinación de los procesos físicos con sistemas de monitorización, comunicación y procesamiento para decisiones productivas y de negocio acertadas (Casalet R., 2017).

2.3.2 *Internet de las cosas*

El internet que utilizamos en el diario vivir actualmente tiene por objetivo integrar cosas de tecnología heterogénea, digitalizando el mundo físico con una sincronización que establezca procesos más eficientes, potenciando así objetos que hasta hace tiempo atrás se conectaban mediante un circuito cerrado, ahora lo hagan mediante la red logrando una acción-reacción entre objetos, éstos se verán desarrollados mediante sistemas embebidos, los cuales recibirán ordenes de manera remota o cumplirán acciones previamente programadas, sin embargo la seguridad supone un reto importante, puesto que la información personal puede estar sujeta a posibles ataques.

El internet de la cosas se perfila como el cambio del todo e incluso de nosotros, considerando el impacto que ha tenido en la educación, ciencia, comunicaciones y negocios; actualmente se registra 3,47 dispositivos conectados por persona en el mundo y se estima al 2020 tener alrededor de 50 MM de dispositivos conectados, como se muestra en el gráfico 2-2, con los que se prevé evolucionar a través de la conversión de los datos en información, conocimiento y sabiduría necesaria para progresar en las próximas décadas y siglos (Evans et al., 2011).

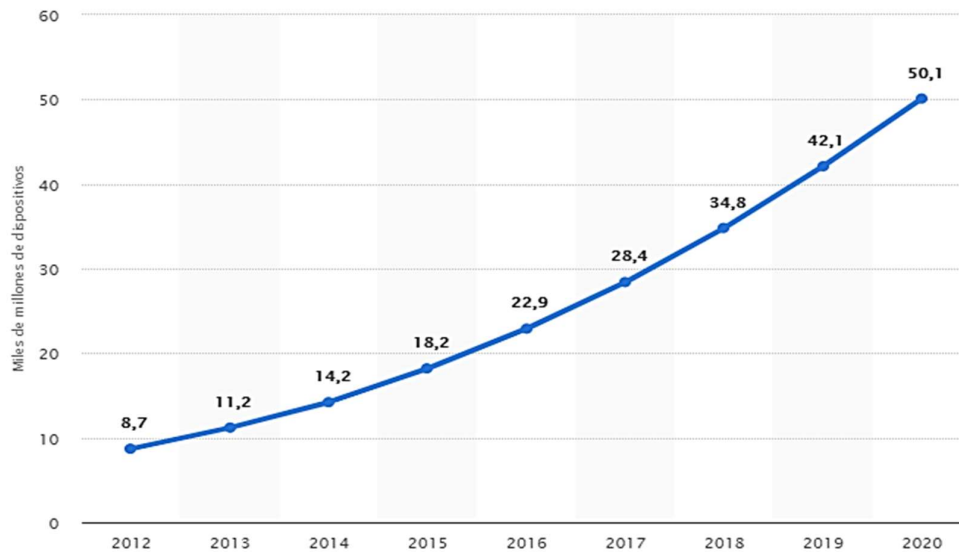


Gráfico 2-2: Expectativa de objetos conectados al 2020.

Fuente: (“Evolución Internet de las cosas: conexiones 2012-2020 | Statista,” 2019).

Hoy en día el Internet de las Cosas (IoT) se ha convertido en un término popular para describir escenarios en los que la capacidad de cómputo y la conectividad a internet se amplían a una variedad de objetos cotidianos y heterogéneos, *i.e.*, todas las cosas tienen una representación y presencia en internet, donde IoT tiene por objetivo ofrecer nuevos servicios y aplicaciones que permita enlazar los mundos físicos y virtuales, en los que la comunicación Machine to Machine (M2M) es la base que permite la interacción entre las cosas y las aplicaciones en la nube (Internet Society, 2015).

Una de las aplicaciones esenciales del Internet de las Cosas busca desarrollar ciudades inteligentes, permitiendo una mejor calidad de vida con costos razonables, proyectándose a mejorar una serie de servicios públicos y privados con el fin de aprovechar con eficiencia los recursos, donde las tecnologías que permitirán dichos objetivos se encuentran en etapas de estandarización y madurez, mientras que la industria realiza lo propio para su aprovechamiento en aplicaciones de interés (Zanella et al., 2014).

2.3.3 *Internet Industrial de las cosas*

La evolución natural del Internet de las Cosas (IoT) proyecta su aplicación en el área industrial dando origen al concepto del Internet Industrial de las Cosas (IIoT), en la actualidad la tecnología celular está siendo utilizada en el acceso inalámbrico IIoT y se va generando nuevas necesidades de aplicaciones que impulsen la tendencia del paradigma Industria 4.0, donde la ausencia de la intervención humana hace que los datos sean predecibles y más fáciles de observar las validaciones (Mumtaz et al., 2017).

El nombre internet de las cosas se otorga al futuro de los dispositivos conectados, dando lugar a dos subconjuntos, el IO consumidor y IO industrial, mientras este último crece más lento se visualiza con más impacto económico, siendo el futuro estratégico para las empresas industriales y proveedores de infraestructura, el IIoT controlará sistemas de misión crítica por lo que la fiabilidad es un desafío, sin embargo el futuro IIoT es más prometedor que la optimización de los sistemas actuales, donde se anuncia que los sistemas revolucionen el mundo mediante la combinación de inteligencia artificial, aprendizaje inteligente y conectividad de alta calidad (Schneider, 2017).

2.3.4 *Sistemas ciberfísicos (CPS)*

Los sistemas ciberfísicos (CPS) hacen referencia a una nueva generación de sistemas integrados con capacidades computacionales y físicas, pudiendo bajo varias modalidades interactuar con los humanos y el mundo exterior, mediante la integración de conocimientos de la ingeniería, buscando por mucho exceder los niveles actuales de la autonomía, funcionalidad, fiabilidad, seguridad cibernética y facilidad de uso, por lo que es importante ir desarrollando y estandarizando arquitecturas que permitan el diseño y desarrollo de sistemas ciberfísicos (Baheti et al., 2011).

Los sistemas ciberfísicos contribuyen a encontrar respuestas para dar atención a nuestra sociedad en desafíos relevantes de la industria y otros campos de aplicación, brindando soporte a las empresas en la optimización de procesos, costes y tiempos, mientras que para el usuario común los CPS se enfocan en un elevado nivel de bienestar, así por ejemplo, en movilidad asistida, seguridad, atención médica, etc. (Acatech, 2011).

2.3.5 *Sistemas ciberfísicos de producción (CPPS)*

Los sistemas ciberfísicos de producción es un nuevo concepto que se trata como la intersección de la física y la cibernética, combinando modelos de la ingeniería eléctrica, electrónica, mecánica, aeronáutica e industrial con los modelos y métodos de la informática, mismos que no se combinan fácilmente por lo que constituye una nueva disciplina de la ingeniería. Los CPS se encuentran continuamente relacionados con la industria 4.0, Machine To Machine (M2M) y el internet de la cosas, dado que se planean a obtener tecnologías que permitan conectar el mundo físico con el de la información (Saumeth et al., 2015).

Teniendo en cuenta las técnicas basadas en modelos y los estándares de campo, los sistemas ciberfísicos CPPS, permiten realizar una arquitectura que busca integrar verticalmente los procesos productivos y las aéreas de negocio, por lo que es prioritario tener acceso a la información y extraer los datos de los procesos de producción (Pérez et al., 2015).

Los CPPS promueven una comunicación interactiva entre los seres humanos, las máquinas y el producto mediante interfaces, mediante el uso de software, electrónica de hardware, sistemas embebidos entre otras como lo muestra el gráfico 3-2, donde sus componentes son capaces de autocontrolar ciertas tareas y adquirir datos del proceso de producción.

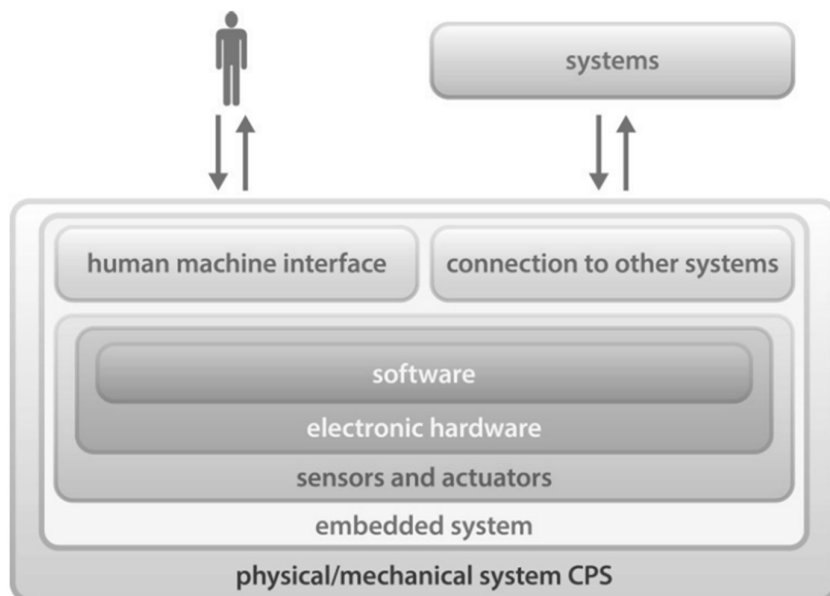


Gráfico 3-2: CPS, interacción humano máquina

Fuente: (Monostori, 2014).

2.3.6 Protocolo Message Queue Telemetry Transport (MQTT)

El protocolo MQTT es un protocolo de mensajería publicación/suscripción diseñado para ser abierto, ligero y simple, ideal para entornos restringidos donde la red es costosa o los recursos de memoria de los dispositivos son limitados, se implementa sobre de TCP-IP y tiene tres cualidades de calidad del servicio, donde el mensaje se entregará como mucho una vez, al menos una vez o exactamente una vez, lo cual permite garantizar que el mensaje llegue a su destino y cuyo transporte es independiente de la carga (IBM & Eurotech, 2010).

La aplicación ligera y sencilla del protocolo MQTT, de acuerdo al trabajo M2M Protocols for Constrained Environments in the Context of IoT: A Comparison of Approaches, presenta una mejor conformidad a los patrones de comunicación (calidad, seguridad y escalabilidad) de acuerdo al escenario de aplicación (Frigieri et al., 2015).

Tabla 1-2: Comparación de protocolos para IoT.

	MQTT	MQTT-SN	Coap
PROTOCOLO DE RED	TCP/IP	No especificado	UDP
TIPO DE CARGA ÚTIL	Binario	Binario	Binario
ADECUADO PARA MICROCONTROLADORES	Sí	Sí	Sí
SEGURIDAD	SSL/TLS	No especificado	DTLS
ESCALABILIDAD	Simple	Simple	Complejo
ARQUITECTURA DE RED	basado en el bróker (publicar/suscribir)	Cliente/servidor basado en bróker, cliente/cliente	Cliente/servidor
PATRÓN DE COMUNICACIÓN	Basado en Topic's	Basado en Topic's	Arquitectura REST
OPCIONES DE QOS	Sí	Sí	Sí
DISPONIBILIDAD DE CÓDIGO ABIERTO	Sí	Aplicación específica	Sí

Fuente: (Frigieri et al., 2015)

Realizado por: Borja Edwin, 2020.

MQTT se ejecuta sobre protocolos de red que permiten mantener una conexión bidireccional ordenada y sin pérdidas, entre sus características está la distribución de mensajes de uno a muchos, un transporte de mensajes independiente de su contenido, calidad de servicio en la entrega de mensajes, minimiza el tráfico de red y notifica a los interesados una desconexión anormal (mqtt-v3.1.1, 2014).

El protocolo MQTT sigue una topología de estrella con un nodo central denominado bróker capaz de contar con hasta 10000 clientes, éste se encarga de transmitir los mensajes, gestionar la red y

mantener activo el canal, periódicamente los clientes envían un paquete (PINGREQ) esperando recibir la respuesta del bróker (PINGRESP), como se muestra en la figura 3-2. La comunicación se basa en “TOPICS” (temas), representados por una cadena jerárquica, misma que es creada por el cliente que publica el mensaje y a la cual los nodos que se suscriben reciben el mensaje, por lo que es posible una comunicación de uno a uno y de uno a muchos.

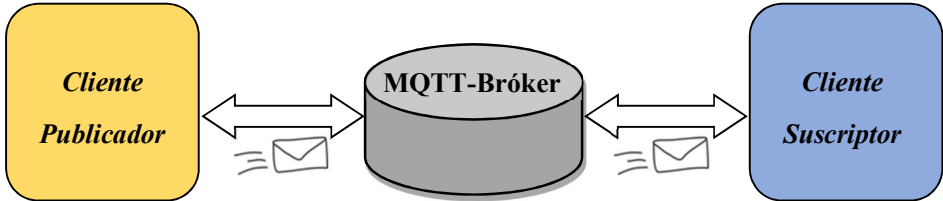


Figura 3-2: Funcionamiento del protocolo MQTT
Realizado por: Borja Edwin, 2020

Los mensajes en MQTT son publicados en Topics, no se requiere de configuración alguna, con publicarlo es suficiente, los topics se tratan como una jerarquía separada con una barra inclinada (/), así, por ejemplo, varios clientes pueden acceder a los mensajes conforme la jerarquía de acceso a la información de interés.

2.3.6.1 Formato de un paquete de control MQTT

El protocolo MQTT intercambia una serie de paquetes de control de manera definida, el cual consta de hasta tres partes, en el siguiente orden:

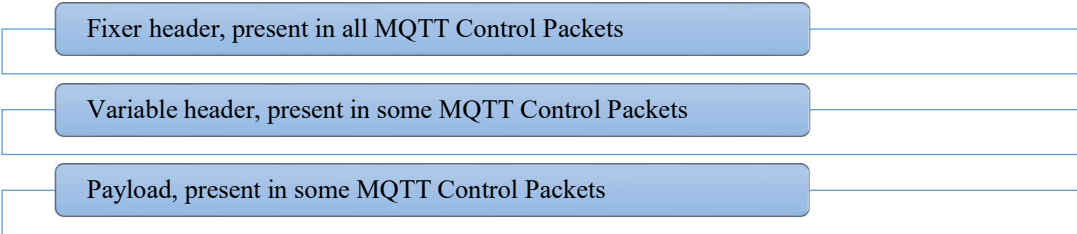


Gráfico 4-2: Estructura de un paquete de control MQTT
Fuente: (mqtt-v3.1.1, 2014).

a. Encabezado Fijo

Cada paquete de control contiene un encabezado fijo con el siguiente formato:

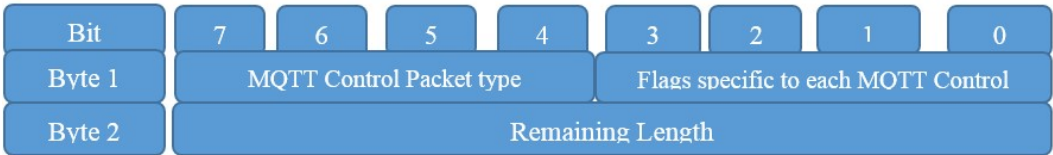


Gráfico 5-2: Formato de encabezado fijo
Fuente: (mqtt-v3.1.1, 2014).

El tipo de paquete de control se define en el primer byte, por un valor sin signo de 4 bits (7-4), cuyos valores se describen a continuación:

Los bits (3-0) del primer byte permiten determinar indicadores para cada tipo de paquete de control, los marcados como reservados está establecido para su uso a futuro, si se reciben indicadores inválidos el receptor deberá cerrar la conexión de red.

La longitud restante, la cual comienza en el segundo byte, se codifica con un esquema de longitud variable utilizando un solo byte para valores de hasta 127, para valores más grandes, los 7 bits más significativos de cada byte codifican el dato y el bit más significativo se utiliza para indicar la existencia de más bytes en la representación, cuyo número máximo de bytes es de cuatro.

2.3.6.2 Niveles de calidad y de servicio

MQTT utiliza ciertos niveles de calidad para la entrega de mensajes, el protocolo de entrega es simétrico y se refiere únicamente a la entrega de un mensaje de aplicación de un único remitente a un único receptor, en el caso de que el servidor esté entregando un mensaje a más de un cliente, cada cliente se trata de manera independiente, por lo que el nivel de calidad utilizado para entregar un mensaje de aplicación saliente puede diferir de un mensaje de aplicación entrante.

a. QoS 0: como máximo una vez entregada

El mensaje se entrega conforme las capacidades de la red, donde el receptor no envía ninguna repuesta y el remitente no realiza ningún intento extra, por lo que el mensaje llega a su destino una vez o ninguna.

b. QoS 1: al menos una entrega

El mensaje deberá entregarse como mínimo una vez, si el remitente no recibe confirmación del receptor, el mensaje se envía hasta que acuse de recibido, una vez confirmado el mensaje se elimina del emisor.

c. QoS 2: entrega exactamente una vez

Este es el servicio de mayor calidad, pues no se aceptan duplicados ni pérdida de mensajes, por lo que deben existir entre el remitente y el receptor al menos dos pares de transmisiones antes de que el remitente elimine el mensaje.

2.3.6.3 Mosquitto

MQTT utiliza una arquitectura típica tipo estrella, donde todos los clientes se conectan a un servidor, el cual se llama bróker, éste es quien gestiona la entrega de los mensajes dado que todos ellos deberán pasar por él, uno de los más utilizados es Mosquitto, el cual es un bróker de código

abierto, mismo que proporciona una biblioteca en C para la implementación de clientes MQTT, entre los más usuales se encuentran el `mosquitto_pub` y `mosquitto_sub` (A Light, 2017).

2.3.7 Plataformas de hardware de bajo costo

El uso de plataformas con sistemas embebidos, de fácil acceso comercial, en el desarrollo del IoT se han convertido en una práctica común, dado que los dispositivos son tratados como una caja negra, el rendimiento de los mismos no ha sufrido evaluación alguna, por lo que en el trabajo *Benchmarking Internet of Things Devices*, se realiza una micro y macro evaluación comparativa de algunos SBCs, de cuyos resultados se desprende, que la BeagleBone Black utiliza una menor potencia de procesamiento y cantidad de memoria, demostrando no solo velocidad sino también un baja latencia frente a otras plataformas, como se indica en la tabla 1-2 (Kruger et al., 2014).

Un sistema embebido es un sistema electrónico diseñado para realizar funciones específicas, muchas de las ocasiones dentro de un sistema más grande y complejo, hoy en día estos dispositivos se encuentran en la mayoría de equipos de uso habitual; a nivel de hardware estos sistemas embebidos tienen en su núcleo una o varias CPU, misma que gobernará un proceso.

Tabla 2-2: Características de plataformas de hardware de bajo coste.

Característica	Raspberry Pi B	BeagleBone	Arduino Uno
Procesador	Arm11	AM335x	ATMega 328
Velocidad	700 MHz	1 GHz	16 MHz
RAM	512 MB	512 MB	2 KB
Sistema Operativo	Linux	Android, Linux, Cloud9, Windows, CE, etc.	n/a
Entorno	Linux, IDLE, Open-Embedded, QEMU, Scratchbox, Eclipse	Python, Scratch, Linux, Eclipse, Android ADK	Arduino IDE
USB	2	1	n/a
Audio	HDMI, Analógico	HDMI	n/a
Video	HDMI, Analógico	Mini-HDMI	n/a
Ethernet	10/100	10/100	n/a
I/O	8 GPIO	69 GPIO, LCD, GPMC, MMC1, MMC2, 7 AIN, 4 temporizadores, 4 puertos seriales, CAN0	14 GPIO, 06-10 bit analog
Tamaño	3.37" x 2.125"	3.4" x 2.1"	2.95" x 2.1"
Min Power	700 mA (3.5 W)	170 mA (.85 W)	42 mA (.3 W)
Voltaje	5 V	5 V	7 - 12 V

Fuente: Raspberry Pi, Arduino, Beaglebone Black - Comparación y Aplicaciones.
Realizado por: Borja Edwin, 2020.

2.3.7.1 BeagleBone Black

La Beaglebone Black es una plataforma compacta de bajo costo, código abierto y capaz de ser usada en aplicaciones complejas e incorpora múltiples mejoras con respecto a su antecesora, ampliando así su memoria RAM, velocidad de procesamiento y memoria flash, además incluye un puerto microHDMI, su procesador es capaz de realizar 2 mil millones de instrucciones por segundo y su costo oscila entre los 45 y 65 usd, su hardware abierto es compatible con diferentes herramientas de software libre por lo que se tiene acceso a amplias bibliotecas de código basado en Linux (Molloy, 2015).

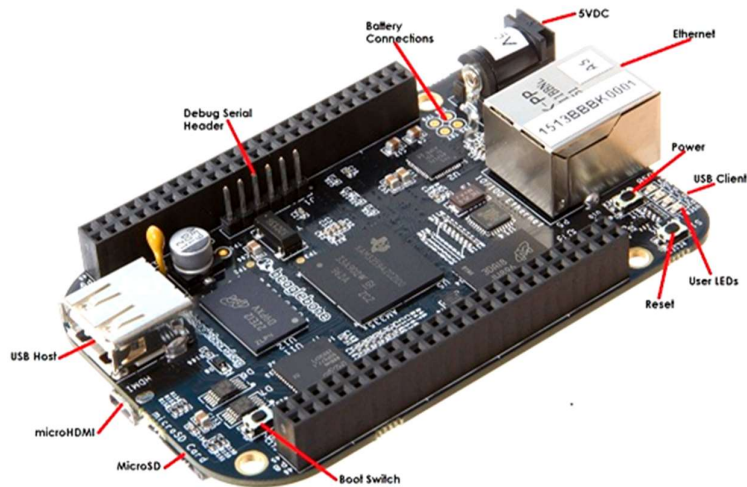


Figura 4-2: BeagleBone Black
Fuente: ("Beagleboard:BeagleBoneBlack - eLinux.org," 2018).

Entre las características principales de la BeagleBone black se encuentran un procesador AM355x 1GHz Arm Cortex A8, cuenta con dos unidades de procesamiento en tiempo real de 32 bits, un acelerador gráfico 3D y una memoria RAM de 512 MB, la BeagleBone Black cuenta con puertos de comunicación microHDMI, USB, Ethernet y conectores de 46 pines x 2, como se muestra en la figura 4-2, para el desarrollo de aplicaciones electrónicas.

En referencia al software la Beaglebone Black utiliza las principales distribuciones de código abierto basado en Linux entre las cuales se incluyen Debian, Ångström, Ubuntu y Arch Linux. Debian es una distribución de Linux cuyo énfasis es el desarrollo de código abierto y de uso común sobre la Beaglebone Black, en una de las formas más habituales para acceder a la BBB es mediante Secure Shell (SSH), el cual es un protocolo de comunicación seguro y encriptado entre los dispositivos de red, se puede utilizar un cliente terminal SSH para conectarse al servidor SSH mediante el puerto 22, Putty es un emulador terminal open-source que utiliza SSH, además la BBB cuenta con un conjunto de tecnologías como Cloud9, el cual es un entorno de desarrollo de

software en el que se puede escribir código Node.js y BoneScript para interactuar con el hardware de la BBB (Molloy, 2015).

2.3.8 *Snap 7*

Snap 7 es una multiplataforma de código abierto para la comunicación ethernet para la interfaz nativa con los PLC Siemens S7, mismos que cuentan con amplia presencia en la industria mundial, Snap 7 ha sido diseñado para superar las limitaciones de los servidores OPC, se adapta bien al trabajar con pequeñas plataformas como raspberry Pi, Beaglebone Black, Cubieboard, pcDuino, Arduino Yun y UDDO, sus tres componentes especializados, Client, Server y Partner, permite integrar sistemas basados en PC sobre una cadena de automatización de PLC (Nardella, 2016).

Entre sus características principales destaca ser una plataforma independiente compatible con Windows, Linux, Oracle Solaris 11 y Mac OSX, así también, con procesadores Intel, AMD i386/x86_64, ARM, Sun Sparc, Mips, siendo escalable desde servidores Blade hasta SBC raspberry PI, compatibilidad de Snap 7 se probó con éxito en 60 sistemas operativos de 32 y 64 bits, como regla general se establece que cualquier sistema operativo de 16 bit no es compatible. Snap 7 por su diseño maneja comunicaciones ethernet, cabe recalcar que no se hace mención al bus de campo, se enfoca en comunicaciones PC-PLC, teniendo varias ventajas frente profibus/mpi, es más rápido, fácil de solucionar problemas y económico.

Los PLC Siemens mediante sus procesadores de comunicación (CP) pueden comunicarse por ethernet mediante TCP/IP y el protocolo S7, el primero es un protocolo estándar para comunicar el PLC con hardware que no pertenece al fabricante, a diferencia del protocolo S7 el cual es la columna vertebral de las comunicaciones de Siemens y está orientado a bloques (Nardella, 2016).

En las comunicaciones con Siemens hay tres componentes al igual que con Snap 7:

- Client: puede realizar solo consultas
- Server: solo responde
- Partners: pueden intercambiar datos por su propia iniciativa.

Los clientes se conectan al servidor interno del procesador de comunicaciones (CP) y realiza una solicitud S7 y el servidor responde con un telegrama S7, en el caso del servidor no necesita configuración dado que es manejado automáticamente por el firmware del CP.

En el caso de Snap 7 los tres componentes se pueden usar simultáneamente en la misma aplicación, el objetivo de Snap 7 es integrar completamente una estación de PC en una red PLC sin que nadie note la diferencia.

Snap 7 tiene su código fuente escrito en ISO C++ se debe aclarar el Snap 7 no es un OPC o recopilador de datos del PLC, lo que hace es aceptar clientes externos y responde a sus solicitudes al igual que un procesador de comunicaciones, en las tablas 2-2 y 3-2 se muestran sus funciones principales, con las cuales se puede extraer información del PLC e interactuar con el proceso sujeto de control de ser el caso.

Tabla 3-2: Funciones administrativas para control de objeto cliente.

Función	Descripción
Cli_Create	Crea un objeto cliente.
Cli_Destroy	Destruye un objeto cliente.
Cli_SetConnectionType	Establecer el tipo de conexión (PG / OP / S7 Basic)
Cli_SetConnectionParams	Establece IP, TSAP local y remoto
Cli_ConnectTo	Conecta un objeto cliente a un PLC.
Cli_Connect	Conecta un objeto cliente a un PLC con parámetros implícitos.
Cli_Disconnect	Desconecta a un cliente.
Cli_GetParam	Lee un parámetro interno del cliente.
Cli_SetParam	Escribe un parámetro interno del cliente.

Realizado por: Borja Edwin, 2020.

Fuente: (Nardella, 2016)

Tabla 4-2: Funciones E/S para intercambio de datos con el PLC.

Función	Descripción
Cli_ReadArea	Lee un área de datos de un PLC.
Cli_WriteArea	Escribe un área de datos en un PLC.
Cli_DBRead	Lee una parte de un DB desde un PLC.
Cli_DBWrite	Escribe una parte de un DB en un PLC.
Cli_ABRead	Lee una parte del área de la IPU desde un PLC.
Cli_ABWrite	Escribe una parte del área de la IPU en un PLC.
Cli_EBRead	Lee una parte del área IPI desde un PLC.
Cli_EBWrite	Escribe una parte del área IPI en un PLC.
Cli_MBRead	Lee una parte del área de Merkers desde un PLC.
Cli_MBWrite	Escribe una parte del área de Merkers en un PLC.
Cli_TMRead	Lee los temporizadores de un PLC.

Cli_TMWrite	Escribir temporizadores en un PLC.
Cli_CTRead	Lee contadores de un PLC.
Cli_CTWrite	Escribir contadores en un PLC.
Cli_ReadMultiVars	Lee diferentes tipos de variables desde un PLC simultáneamente.
Cli_WriteMultiVars	Escribe diferentes tipos de variables en un PLC simultáneamente.

Realizado por: Borja Edwin, 2020.

Fuente: (Nardella, 2016)

2.3.9 MySQL

Es el sistema para gestión de bases de datos de código abierto más utilizada del mundo y que a partir de 2010 forma parte de Oracle Corporation, teniendo una gran variedad de interfaces como C, C++, C#, PHP, Python entre otras, para el fácil acceso a las bases de datos MySQL. Del mismo modo MySQL es frecuentemente utilizado en aplicaciones web sobre plataformas Linux, Windows, Mac OS X, Solaris, etc., se caracteriza por ser rápido y multiusuario.

MySQL admite tipos de datos como Float, Char, Varchar, Double, date, time, set, year timestamp, enum y text, con registros de longitud fija y variable, permitiendo hasta 64 índices por tabla, los clientes pueden conectarse con el server de MySQL mediante TCP/IP en cualquier plataforma.

Su característica open source la hace una de las más convenientes para utilizar en la gestión de datos obtenidos de desarrollos IoT, facilitando así su integración con nuevas plataformas como en el presente proyecto la BeagleBone Black.

2.3.10 APACHE

Es un proyecto de software colaborativo que tiene como fin crear un código fuente robusto, funcional, comercial y de libre disponibilidad, este proyecto es administrado por un grupo de voluntarios de todo el mundo, además cientos de usuarios aportan con ideas, documentación y códigos para el desarrollo del proyecto de servidor Apache HTTP, el cual forma parte de la Apache Software Foundation, entre sus características son altamente configurables, permite manejar bases de datos, es multiplataforma y con facilidad para conseguir soporte (“About the Apache HTTP Server Project - The Apache HTTP Server Project,” 2018).

El servidor apache HTTP se puede implementar sobre plataformas Unix, Windows, Mac y otros, por lo que lo hace habitualmente universal para el envío de páginas web estáticas y dinámicas, es de uso común cuando se tiene por objetivo publicar un contenido con seguridad y confiabilidad, para el caso de Linux el archivo httpd.conf es el que define la configuración de Apache para los

host virtuales, cualquier cambio en la configuración requiere reiniciar el servidor, con Apache es posible hacer páginas web y alojarlas en una PC convirtiéndola en un servidor.

2.3.11 GOOGLE CHARTS

Es una herramienta de Google potente y gratuita, se constituye como una excelente opción para visualizar datos de forma gráfica en un sitio web, para ello Google Charts cuenta con algunas bibliotecas que permite enumerar los datos a representar y la personalización de gráficos, por lo que es posible ajustarse a la apariencia del sitio web, los gráficos son interactivos y exponen sucesos que permiten conectarlos para la creación de tableros complejos u otras acciones integradas a la página web.

Los gráficos se rellenan con los datos utilizando la clase DataTable, la cual proporciona métodos para clasificar, filtrar y modificar datos, pudiéndose también ingresar datos de la página web, una base de datos u otro proveedor de datos ya que Google charts incluye un lenguaje de consulta similar a SQL (“Using Google Charts | Charts | Google Developers,” 2018).

2.3.12 WIRESHARK

Wireshark es una herramienta de software de código abierto, que sirve para el análisis y desarrollo de soluciones de los problemas detectados en la red, así también su uso ha tenido un espacio importante para fines didácticos y educativos, en los que se hace necesario examinar y evaluar posibles vulnerabilidades de seguridad de la red o depurar ejecuciones de protocolos. La captura de paquetes permite analizar la estructura MAC, direccionamiento IP, segmento de paquetes, contenido y velocidades de transmisión, es compatible con aproximadamente 500 protocolos en los que se incluye MQTT, siendo una buena herramienta para la obtención de los datos de transmisión de la propuesta para la monitorización.

CAPÍTULO III

3. METODOLOGÍA DE LA INVESTIGACIÓN

3.1 Tipo y Diseño

La presente investigación es de tipo experimental, considerando que se propone una arquitectura para monitorizar procesos industriales e implementarla usando el protocolo MQTT sobre plataformas de hardware de bajo costo misma que se pone a prueba en una aplicación de laboratorio, la monitorización es una vía solo para adquirir datos del proceso.

3.2 Métodos de Investigación

El desarrollo de la monitorización con protocolo MQTT sobre plataformas de hardware de bajo costo utiliza el método deductivo, considerando que antes de la implementación y validación, se debe proponer una arquitectura en la que se toma en cuenta las características del protocolo y de la SBCs.

3.3 Enfoque de la Investigación

El enfoque de la presente investigación es cuantitativo, dado que es necesaria la adquisición de datos de velocidad de transmisión de paquetes, en aplicación del protocolo MQTT a fin de que conlleve a un análisis para validar la superioridad del protocolo propuesta en la hipótesis.

3.4 Alcance de la Investigación

Exploratorio: Examinar las tendencias de la industria y conceptos innovadores, que se encuentran relacionados al paradigma de la Industria 4.0 y estudios relacionados a la aplicación del protocolo Message Queue Telemetry Transport (MQTT), adquiriendo información sobre características técnicas y de configuración para ser utilizados en la propuesta de investigación, misma que se prevé, facilite futuros desarrollos para la industria a bajos costos.

Descriptivo: Información por minorizada del trabajo de investigación a fin de que facilite futuras propuestas en base a la presente arquitectura y determinar la velocidad del protocolo MQTT aplicado a procesos industriales y su superioridad frente TCP IP.

3.5 Población de Estudio

La propuesta y experimentación se aplica a un equipo de laboratorio que permite ilustrar un sistema de automatización industrial, las conclusiones son válidas para cualquier tipo de sistema y/o proceso industrial.

3.6 Unidad de Análisis

Ordenador de placa reducida de bajo costo, Beaglebone black, con aplicación del protocolo MQTT para monitoreo remoto de un equipo de laboratorio que permita ilustrar un sistema de automatización industrial.

La programación será realizada con software libre mediante el cual se ejecutará la aplicación del protocolo MQTT, para establecer las comunicaciones del bróker, mismo que gestionará la data del publicador/suscriptor, a su vez se programará un MQTT client para la recepción de datos de la planta.

3.7 Técnica de recolección de datos primarios y secundarios.

La principal técnica de recolección de datos será mediante experimentación y análisis documental de la velocidad de transmisión de paquetes de los protocolos de comunicación.

3.8 Instrumentos de recolección de datos primarios y secundarios.

Se realizará mediante el uso del software Wireshark para la medición de la velocidad de transmisión de paquetes de los protocolos de comunicación para su análisis.

3.9 Operacionalización de variables

OPERACIONALIZACIÓN DE VARIABLES TÍTULO: DISEÑO DE UNA ARQUITECTURA USANDO EL PROTOCOLO MESSAGE QUEUE TELEMETRY TRANSPORT (MQTT) SOBRE PLATAFORMAS DE BAJO COSTE, PARA MONITORIZACIÓN DE PROCESOS INDUSTRIALES AUTOR: EDWIN R. BORJA				
VARIABLE	TIPO DE VARIABLE	DEFINICIÓN CONCEPTUAL	INDICADOR	UNIDAD DE MEDIDA
Protocolo MQTT	Independiente	Normas y reglas de comunicación para transmisión de información	Ventajas y desventajas	U.A
Velocidad de transmisión en un proceso industrial	Dependiente	Cantidad promedio de datos que se transfieren por unidad de tiempo	Variación de bits en un intervalo de tiempo	bps

3.10 Matriz de consistencia

MATRIZ DE CONSISTENCIA							
FORMULACIÓN DEL PROBLEMA	OBJETIVO	HIPÓTESIS GENERAL	VARIABLE	INDICADOR	TÉCNICA	INSTRUMENTO	
¿Cómo operan los sistemas ciberfísicos de producción y las últimas tendencias en la industria?	Diseñar una arquitectura que permita monitorear procesos industriales a través del protocolo MQTT y SBCs de bajo costo, facilitando así la supervisión remota.	Implementar una arquitectura basada en el protocolo MQTT sobre plataformas de hardware de bajo costo, permite monitorizar procesos industriales, obteniéndose velocidades de transmisión superiores a las del protocolo TCP IP.	Variable independiente: Protocolo MQTT.	Ventajas y desventajas	Investigación Bibliográfica	Internet Libros Bases de datos científicas Software libre SBCs – Beaglebone Black, Snap 7, MySQL Estación de laboratorio para ilustración de sistema industrial	
¿Cómo funciona el protocolo MQTT?	Analizar las características del protocolo Message Queue Telemetry Transport (MQTT) y su aplicación en el IoT, para adaptarlo a los requerimientos de la industria.						
¿Qué velocidad tiene MQTT en relación a TCP IP usualmente utilizado en la industria?	Evaluar las velocidades de transmisión del protocolo MQTT y del protocolo TCP IP usualmente utilizado.		Variable dependiente: Velocidad de transmisión en un proceso industrial	Variación de bits en un intervalo de tiempo			Experimentación
¿Cómo monitorear de manera remota un sistema y/o proceso industrial a través del protocolo MQTT y una Beaglebone Black?	Desarrollar una aplicación sobre plataformas de hardware de bajo costo, mediante el uso del protocolo Message Queue Telemetry Transport (MQTT), para monitorización de procesos de producción, utilizando como referencia un equipo de laboratorio con el objeto de ilustrar un sistema de automatización industrial.						

CAPÍTULO IV

4. PRUEBAS Y RESULTADOS

4.1. Resultados de la monitorización con una arquitectura utilizando MQTT

4.1.1 *Análisis de velocidad de transmisión*

Para identificar la velocidad con la que se transmiten los paquetes se utiliza el software Wireshark, el cual de manera paralela al funcionamiento de la plataforma nos permitirá saber la velocidad de transmisión de paquetes por protocolo, para el caso de estudio nos interesa conocer la velocidad del protocolo MQTT y TCP en el sistema de comunicación implementado como se muestra en la figura 1-4.

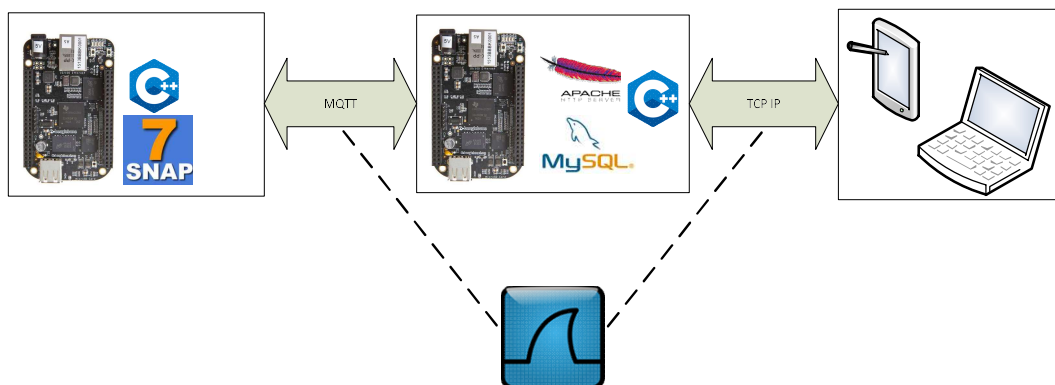


Figura 1-4: Sistema de comunicación y captura con wireshark

Realizado por: Borja Edwin, 2020.

Con la operación del bróker mosquitto y su cliente se obtiene la captura de los mensajes transmitidos bajo el esquema de publicación suscripción conforme el protocolo MQTT, para ello conforme van ingresando las piezas en la estación clasificadora se van generando los mensajes, cuyos paquetes se identifican con Wireshark, por lo cual hacemos uso de un potente filtro con el que cuenta la herramienta, éste ayuda al rastreo de paquetes, permitiendo visualizar solo los paquetes que son de interés en este caso los que cuentan con el protocolo MQTT y TCP, tal como se muestra en la figura 2-4.

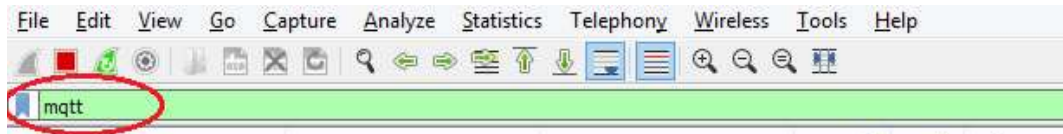


Figura 2-4: Aplicación de filtro de protocolo MQTT
 Realizado por: Borja Edwin, 2020.

Al igual que con el protocolo MQTT realizamos el filtro para TCP, ejecutado el filtro se obtiene una lista de los mensajes transmitidos, como se muestra en la Figura 8-4, de los cuales es de interés conocer la velocidad de transmisión .

No.	Time	Source	Destination	Protocol	Length	Info
4	0.000762387	192.168.0.248	192.168.0.240	MQTT	102	Connect Command
6	0.000826759	192.168.0.240	192.168.0.248	MQTT	70	Connect Ack
8	0.002342301	192.168.0.248	192.168.0.240	MQTT	85	Publish Message [TopicTest]
9	0.002439010	192.168.0.248	192.168.253...	MQTT	85	Publish Message [TopicTest]
12	0.038850523	192.168.0.248	192.168.0.240	MQTT	68	Disconnect Req

Figura 3-4: Mensajes de protocolo MQTT filtrados en Wireshark
 Realizado por: Borja Edwin, 2020.

Realizado el filtro de los protocolos por cada mensaje, sea este mediante MQTT o TCP, cuando una pieza ingresa a la estación de clasificación o se realiza una petición de actualización de la pagina web respectivamente, es necesario identificar la velocidad de transmisión para ello se utiliza la herramienta jerarquía de protocolo del paquete capturado, como se muestra en la figura 4-4.

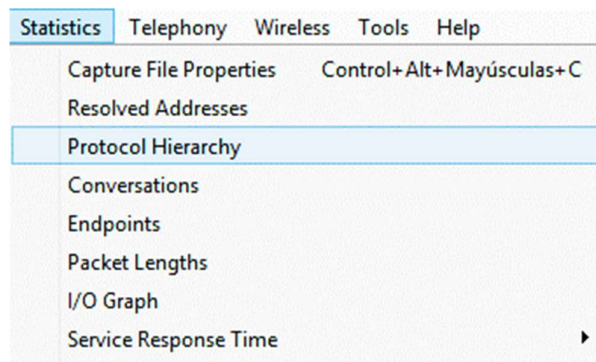


Figura 4-4: Herramienta protocol hierarchy
 Realizado por: Borja Edwin, 2020.

Obteniéndose el detalle de los paquetes capturados, en los que consta, el nombre del protocolo, numero relativo y total de bytes del protocolo, así como los Bits/s que son de interés para la demostración de hipótesis, como se muestra en la figura 5-4 y 6-4 tanto para MQTT y TCP.

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	5	100.0	410	86 k	0	0	0
Ethernet	100.0	5	17.1	70	14 k	0	0	0
Internet Protocol Version 4	100.0	5	24.4	100	21 k	0	0	0
Transmission Control Protocol	100.0	5	58.5	240	50 k	0	0	0
MQ Telemetry Transport Protocol	100.0	5	19.5	80	16 k	5	80	16 k

Figura 5-4: Identificación de velocidad de protocolo MQTT
 Realizado por: Borja Edwin, 2020.

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	5	100.0	286	4572	0	0	0
Ethernet	100.0	5	24.5	70	1119	0	0	0
Internet Protocol Version 4	100.0	5	35.0	100	1598	0	0	0
Transmission Control Protocol	100.0	5	40.6	116	1854	5	116	1854

Figura 6-4: Identificación de velocidad de protocolo TCP

Realizado por: Borja Edwin, 2020.

El procedimiento descrito se realiza por cada mensaje transmitido por protocolo, obteniéndose los datos capturados conforme la tabla 1-4, cuyos datos son base, para ser sometidos a la prueba de hipótesis.

Tabla 1-4: Muestras capturadas por protocolo.

Captura	kBits/s - TCP	kBits/s - MQTT
M1	2.811	15.000
M2	1.854	15.000
M3	0.073	16.000
M4	2.177	16.000
M5	3.010	16.000
M6	4.964	0.311
M7	1.966	0.163
M8	3.968	15.000
M9	2.043	6.424
M10	0.073	

Realizado por: Borja Edwin, 2020.

4.1.2 Prueba de hipótesis

Una vez obtenido los datos de velocidad de la plataforma implementada, se realiza la comprobación de la hipótesis, al ser un estudio transversal en el que se compara los protocolos de manera independiente, se puede sugerir la aplicación de una prueba paramétrica, sin embargo los datos obtenidos no provienen de una distribución normal, por lo cual se utiliza una prueba no paramétrica equivalente para muestras pequeñas, siendo una alternativa la prueba U de Mann – Whitney para pruebas independientes (Flores et al., 2017).

Para la prueba de hipótesis determinamos un nivel de significancia del 5% por lo tanto el valor 0.05, siendo la regla de decisión si el valor obtenido de la prueba U de Mann – Whitney es mayor o igual a 0.05 se acepta la hipótesis nula, caso contrario se acepta hipótesis alternativa o hipótesis estadística planteada en la propuesta.

La hipótesis Nula: “Implementar una arquitectura basada en el protocolo MQTT sobre plataformas de hardware de bajo costo permitirá monitorizar de forma remota los procesos industriales y se obtendrá velocidades de transmisión iguales a las del protocolo TCP IP”.

La hipótesis Alternativa: “Implementar una arquitectura basada en el protocolo MQTT sobre plataformas de hardware de bajo costo permitirá monitorizar de forma remota los procesos industriales y se obtendrá velocidades de transmisión superiores a las del protocolo TCP IP”

Cálculo del estadístico U de la prueba U de Mann – Whitney

$$U_1 = n_1 n_2 + \frac{n_1(n_1+1)}{2} - R_1 \quad \text{Ec 1.}$$

$$U_2 = n_1 n_2 + \frac{n_2(n_2+1)}{2} - R_2 \quad \text{Ec 2.}$$

Donde n_1 y n_2 corresponden a los tamaños de los datos obtenidos por protocolo, R_1 y R_2 son la sumatoria de los rangos de las observaciones realizadas, como se muestra en la tabla 2-4, el estadístico U se define como el menor entre U_1 y U_2 , este se compara con los valores críticos de U Mann – Whitney.

Tabla 2-4: Cálculo de rangos para la prueba de hipótesis U de Mann – Whitney.

Rangos			
Protocolo	N	Rango promedio	Suma de rangos
kBits_por_segundo TCP	10	7.10	71.00
MQTT	9	13.22	119.00
Total	19		

Realizado por: Borja Edwin, 2020.

Se obtiene una descripción de los datos comparados, la suma de rangos y el valor de la mediana con mayor valor, este dato es el resultado de la suma de rangos dividido para la cantidad de elementos del grupo, interpretándose como las observaciones de la transmisión por TCP obtuvo un rango promedio de 7.10, mientras que el mayor rango promedio le corresponde a la transmisión mediante MQTT con un valor de 13.22.

Tabla 3-4: Cálculo de la prueba de hipótesis U de Mann – Whitney.

Estadísticos de contraste ^a	
	kBits_por_segundo
U de Mann-Whitney	16.0000
W de Wilcoxon	71.0000
Z	-2.3772
Sig. asintót. (bilateral)	0.0174
Sig. exacta [2*(Sig. unilateral)]	,017 ^b

Realizado por: Borja Edwin, 2020.

El estadígrafo de U Mann – Whitney obtenido es de 16 y el p (sig. asintót. (bilateral)) es 0.0174, como se muestra en la tabla 3-4, se identifica que el valor de la prueba es menor a 0.05, por lo tanto, se rechaza la hipótesis nula y se acepta en forma plena la hipótesis alternativa, concluyéndose que afirmativamente que el implementar una arquitectura basada en el protocolo MQTT sobre plataformas de hardware de bajo costo permite monitorizar de forma remota los procesos industriales y se obtiene velocidades de transmisión superiores a las del protocolo TCP IP, con un nivel de significancia del 5%.

Tabla 4-4: Prueba de la hipótesis, valoración de variables.

VARIABLES	VALOR	CRITERIO DE EVALUACIÓN	HIPOTESIS ACEPTADA/ RECHAZADA
Protocolo	MQTT	Registra un bajo consumo de ancho de banda Liviano para su implementación sobre el modelo TCP IP Escalable con posibilidad de conexión de gran cantidad de clients al broker Bajo costo, fácil integración con varias plataformas de uso industrial Se requiere reforzar la seguridad de usuarios	Aceptada
Velocidad de Transmisión	Tabla 1-4: Muestras capturadas por protocolo.	Tabla 3-4: Cálculo de la prueba de hipótesis U de Mann – Whitney.	Aceptada

Realizado por: Borja Edwin, 2020.

4.2. Trabajos futuros

- La arquitectura propuesta deberá ser sometida a un proceso industrial más grande para un análisis de comportamiento con gran volumen de datos.
- La seguridad de la información es muy importante en sistemas industriales de alta criticidad por lo que es un tema prioritario a ser considerado para su fortalecimiento, a fin de evitar se produzcan acciones no deseadas dentro de los procesos industriales.
- El uso de MQTT en el IoT debe fusionarse con otras tecnologías como el Big Data, de modo que, se pueda desarrollar modelos predictivos para los diferentes niveles gestión de un proceso industrial.
- El uso de SNAP 7 y MQTT no solo permite realizar la monitorización de un proceso, dentro de sus capacidades esta la lectura y escritura de variables, por lo tanto, la propuesta realizada en el presente trabajo puede llegar a tener como alcance el control de procesos industriales con un bajo consumo de ancho de banda.

CAPÍTULO V

5. PROPUESTA

5.1 Diseño e implementación de una arquitectura para monitorización

5.1.1 Descripción de requerimientos

Para plantear una arquitectura, conforme el problema, se tiene como requerimientos de alto nivel obtener una arquitectura que permita:

- Integración con la red existente
- Integración con equipos existentes
- Comunicación con controladores de procesos industriales
- Capacidad de lectura remota sobre los controladores
- Visualización de datos sobre una aplicación de acceso remoto

5.1.2 Componentes de la arquitectura

La arquitectura propuesta se presenta en el gráfico 1-5, en la cual se muestra un proceso industrial controlado por un PLC del cual se extrae los valores u estados de la variable a motorizar, hasta la BeagleBone Black 1 (BBB1), la cual es el bróker central de comunicación y publicador de dichos valores, así también se cuenta con una BeagleBone Black 2 (BBB2), la cual será un cliente y gestor de una base de datos de la información recopilada del proceso, aquí se aloja un servidor web el que permitirá visualizar el estado de la variable a monitorizar desde otros dispositivos.

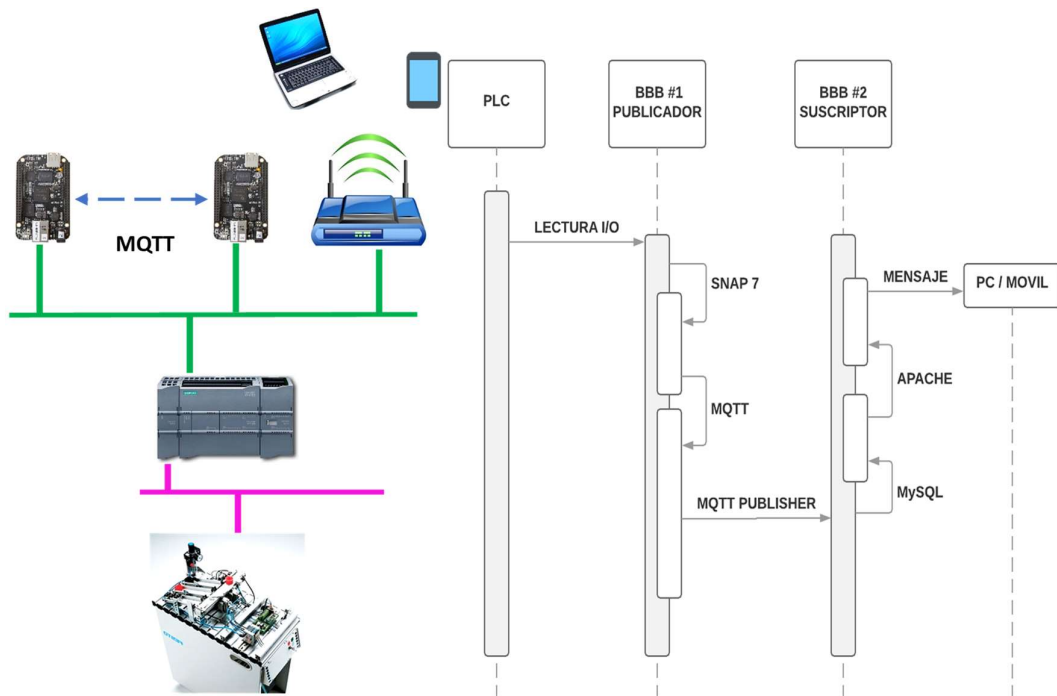


Gráfico 1-5: Arquitectura propuesta y sus componentes.
Realizado por: Borja Edwin, 2020.

5.2 Programas y selección de equipos

Se identifica las especificaciones, características y ventajas de los equipos presentados en la tabla 1-5 de los posibles controladores del proceso. El controlador seleccionado para el proceso es un PLC Siemens S7 1200, este tipo de controlador es de los más comunes en las aplicaciones industriales, tiene entre sus ventajas un costo moderado frente a las prestaciones con las que cuenta.

Tabla 1-5: Características de controladores para el proceso.

Característica		PLC	SBC	LOGO
Aplicabilidad	Industria	Alta	Bajo	Medio
	Plataforma	Alta	Medio	Medio
Robustez		Alta	Ninguna	Medio
Programación		Ladder	Código estructurado	Ladder
Escalabilidad		Alta	Alta	Medio
Precio		Relativamente costoso	Bajo costo	Mediano Costo

Realizado por: Borja Edwin, 2020.

Para efectuar la comunicación mediante MQTT se utiliza un SBC (BeagleBone Black), el cual presenta características y ventajas superiores frente a sus competidores como se muestra en la tabla 2-5, entre las que destacan su conectividad, velocidad de procesamiento y funcionalidad adecuada para la operación del bróker para la gestión de mensajería en uso del protocolo MQTT.

Tabla 2-5: Tabla comparativa de SBC

Característica	Raspberry Pi B	BeagleBone	Arduino Uno
Procesador	Arm11	AM335x	ATMega 328
Velocidad	700 MHz	1 GHz	16 MHz
RAM	512 MB	512 MB	2 KB
Programación	Código estructurado	Código estructurado	Código estructurado
Ethernet	10/100	10/100	n/a
I/O (Input/Output)	8 GPIO	69 GPIO, LCD, GPMC, MMC1, MMC2, 7 AIN, 4 temporizadores, 4 puertos seriales, CAN0	14 GPIO, 06-10 bit analog
Aplicación General	Funcionalidad (servidor)	Funcionalidad (servidor)	Controlador (I/O)
Precio	Accesible	Accesible	Muy Accesible

Realizado por: Borja Edwin, 2020.

Una vez revisada las características de los equipos y definido los requerimientos para el desarrollo de la propuesta, se determina el software por cada equipo a utilizar, como se describe en la tabla 3-5.

Tabla 3-5: Equipos y Software utilizado en la propuesta

Equipo	Detalle	Descripción	Software	Conexión con:
PLC	S7 1200 1214 C	Controlador del proceso	TIA Portal V13	BBB con SNAP7
BeagleBone	Black	Bróker MQTT	C++	PLC, Client MQTT
BeagleBone	Black	Client MQTT/Web Service	C++/php	Bróker, Dashboard Cliente MQTT
Ordenador/Móvil	Acceso a internet	Dashboard Cliente MQTT	Página Web	Client MQTT, Web Service

Realizado por: Borja Edwin, 2020.

Conforme las necesidades y el desarrollo del proyecto se asignan las direcciones IP que tendrá cada dispositivo a utilizar, siendo de utilidad para la programación y configuración de la propuesta, las direcciones utilizadas en desarrollo del proyecto se muestran en la tabla 4-5.

Tabla 4-5: Direcciones IP utilizadas en la propuesta

DISPOSITIVO	DIRECCION IP
PLC S7 1200 1214 C - Controlador del proceso	192.168.1.101
BeagleBone Black Bróker MQTT	192.168.0.248
BeagleBone Black Client MQTT/Web Service	192.168.0.240
Ordenador/Móvil con Acceso a internet	192.168.100.5

Realizado por: Borja Edwin, 2020.

5.3 Caso de aplicación

Para demostrar la aplicabilidad del protocolo MQTT en la monitorización de procesos industriales se usa una aplicación de laboratorio, de la cual se busca extraer el estado de una variable y que la misma sea transmitida mediante MQTT para su posterior visualización mediante un web service, la aplicación de laboratorio a utilizar es una maqueta de marca Festo como se encuentra en la Figura 1-5, misma que se describe como una estación de clasificación de piezas por color, ésta cuenta con un sistema de electroválvulas y un conjunto de actuadores neumáticos de carrera corta, controlados por un PLC, cuyo funcionamiento consiste en detectar una pieza mediante un sensor de reflexión directa para en lo posterior activar una banda transportadora, que ubica la pieza ante un sensor de color, el cual detecta la característica de la pieza, es decir, negra, roja o plateada, conforme la identificación de color la banda transporta la pieza hasta la posición de un actuador neumático, el cual dirige la pieza a la rampa correspondiente según su clasificación.

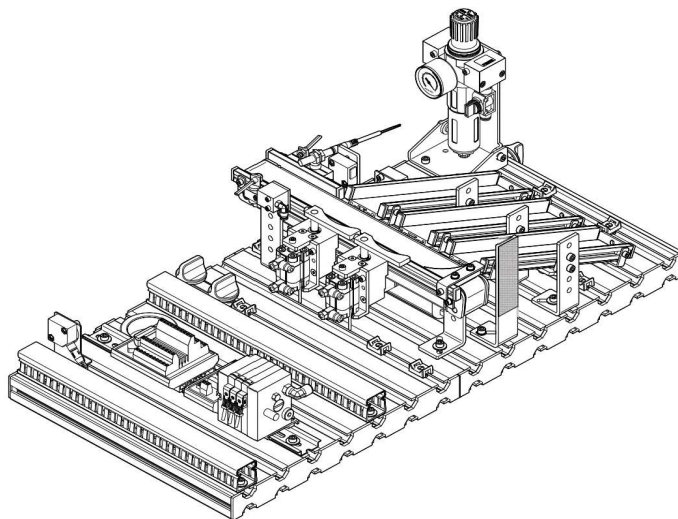


Figura 1-5: Caso de aplicación – Estación de clasificación Festo
Fuente: Festo Didactic, 2015.

5.3.1 *PLC y BeagleBone Black 1 – Bróker*

La BeagleBone Black se comporta como un cliente, misma que establece comunicación con el PLC, utilizando el paquete de comunicación de SNAP 7, permitiendo extraer los datos del PLC Siemens S7 a través de Profibus Ethernet, por lo que es necesario instalar y declarar las librerías definir variables, constantes a utilizar en la programación, así como también se requiere conocer la dirección ip y dirección de memoria del controlador de donde se realizará la lectura.

La obtención de datos del PLC se realiza mediante el uso de la librería Snap7.h, así también se establece funciones genéricas para utilizar el mismo proceso en diferentes partes del programa, para la adecuada visualización de los datos se utiliza la función HEXDUMP, la cual devuelve datos en hexadecimal, para el caso de estudio las áreas de memoria a leer son la 108, 106 y 104.

Sintaxis para conectar las BBB con el controlador PLC:

```
int res = Client->ConnectTo(Address, Rack, Slot);  
return res == 0;
```

Sintaxis para la lectura de datos para visualización:

```
int res1=Client->ReadArea(S7AreaMK, 0, 108, 1, S7WLByte, &DB1); //Lectura de piezas  
negras  
int res2=Client->ReadArea(S7AreaMK, 0, 106, 1, S7WLByte, &DB2); //Lectura de piezas  
plateadas  
int res3=Client->ReadArea(S7AreaMK, 0, 104, 1, S7WLByte, &DB3); //Lectura de piezas  
Rojas
```

5.3.2 *Aplicación del protocolo MQTT y BeagleBone Black 2 -Client*

El protocolo MQTT es un protocolo de mensajería que funciona mediante la publicación/suscripción haciendo uso de un bróker, en el caso de estudio se aplica el bróker Mosquitto, creado por la fundación Eclipse e IBM, éste se ha elegido para gestionar la entrega y distribución de mensajes entre los dispositivos suscritores, así también, el bróker actúa como puente con otros servidores de MQTT, facilitando la construcción de redes y escalabilidad.

La puesta en operación del bróker Mosquitto comienza con la instalación de las librerías de mosquitto en la BeagleBone Black, una vez configurado el bróker se obtiene el detalle de los puertos de comunicación, para el caso de MQTT tiene por defecto el uso del puerto 1883, la estructura de la propuesta con las plataformas se describe en la figura 2-5.

Sintaxis para la instalación de Mosquito bróker y client:

```
apt-get install mosquitto
```

```
apt-get install mosquitto-clients
```

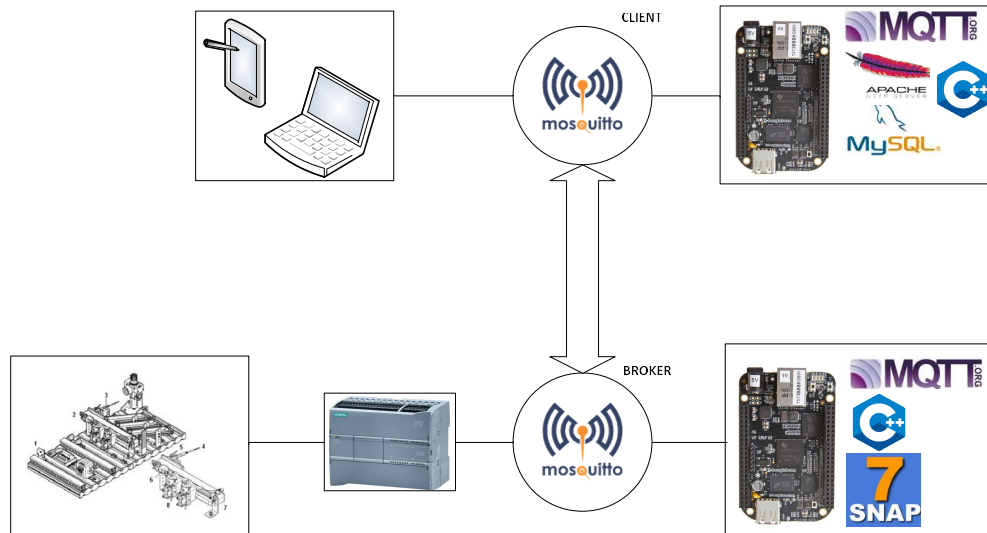


Figura 2-5: Estructura de la propuesta y plataformas

Realizado por: Borja Edwin, 2020.

En la BeagleBone Black cliente se instala mosquitto client en ella es necesario deshabilitar algunos recursos y herramientas como cloud9 y node, a fin de evitar conflictos con la administración de la base de datos y el web service, el cual permitirá el acceso a los datos obtenidos del proceso sujeto de monitorización desde cualquier computador o smartphone desde la red.

5.3.3 Topología MQTT

El protocolo MQTT tiene una configuración de topología en estrella, con un nodo central denominado bróker, el cual se comunica con dos clientes o más, por lo que es posible tener una comunicación de uno a uno y de uno a muchos, el bróker tiene como objetivo el gestionar la red y transmitir los mensajes, los cuales serán receptados por los suscriptores según el tópico al que se hayan suscrito, la topología se muestra en la figura 3-5.

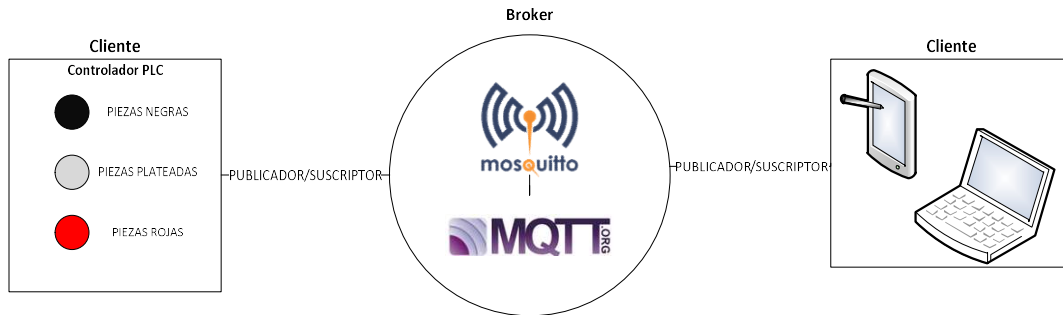


Figura 3-5: Topología MQTT.
Realizado por: Borja Edwin, 2020.

5.3.4 *Topic*

La comunicación por MQTT se basa en “Topics” (temas), representadas por una cadena jerárquica, cada nivel es separado por una barra diagonal, de manera que el agente MQTT decida qué cliente recibe qué mensaje, es decir, el bróker filtra los mensajes para cada cliente conectado, hay que considerar que un topic debe contener al menos un carácter y que la cadena permite espacios vacíos, sin embargo el uso de éste último no es recomendable, para el caso de estudio se define el tópic “TopicTest” en el cual se registrará cualquier cambio en la pila de color (Negro, Plateado o Rojo).

Sintaxis definición del tópic para caso de estudio:

```
%define MQTT_TOPIC "TopicTest"
```

5.3.5 *Transmisión de mensajes, publicación – suscripción*

La transmisión de mensajes con MQTT utiliza un esquema de publicación – suscripción, como consta en la Figura 4-5, tiene una arquitectura simple, los dispositivos que desean suscribirse, deben hacerlo al bróker especificando los tópicos que sean de interés, así también, se requiere conocer la dirección IP del bróker, el puerto por donde se escucha y la definición de una calidad de servicio (QoS).

Sintaxis para establecer conexión con el servidor MQTT y publicación del topic desde el bróker:

```
int ret=mosquitto_connect(mosq, mosq_ip, MQTT_PORT, 0);
int ret=mosquitto_publish(mosq, NULL, MQTT_TOPIC, strlen(cj1), 0, false);
```

Sintaxis para la suscripción del topic desde el cliente:

```
mosquitto_subscribe (mosq, NULL, MQTT_TOPIC, 0);
```

La calidad de servicio utilizada en la propuesta es 0, para garantizar la recepción del mensaje, dado que ésta certifica una entrega conforme las capacidades de la red, para el caso de estudio la red propone una conexión con estabilidad y confiabilidad.

El sistema funciona con dos SBC, de los cuales uno servirá como cliente y otro como bróker, en el bróker se alberga SNAP 7, para extraer las variables a monitorizar del PLC que comanda el proceso, las cuales serán publicadas usando Mosquitto – MQTT, mediante Topics, como se indica en el diagrama de flujo de la Figura 5-5 y en la Figura 6-5, en el cliente de Mosquitto – MQTT se receptorá las publicaciones conforme se suscriba a los Topics requeridos y los almacenará usando MySQL server, como se muestra en el diagrama de flujo del gráfico 2-5 y gráfico 3-5, teniendo en su parte final tanto del bróker como del cliente una función para liberar los recursos inicializado, mediante el cual se desconecta, se destruye la estructura reinicializándola y se procedo con una limpieza del recurso.

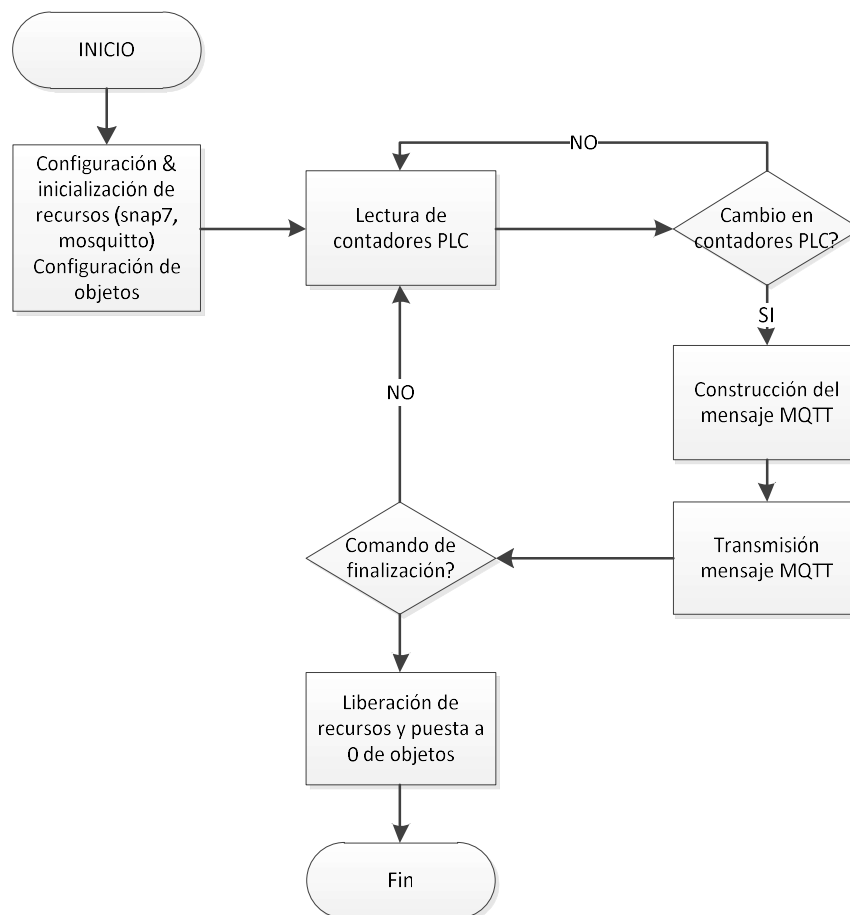


Gráfico 2-5: Diagrama de flujo, BeagleBone Black 1 - Bróker.
Realizado por: Borja Edwin, 2020.


```

192.168.0.248 - PuTTY
root@beaglebone:/home/debian/Desktop/Rodri# ./mqtt_snap
PLC FOUND!!
Hi Mosquitto....
New piece sorted!!
Black Pieces=0, Silver Pieces= 0, Red Pieces= 1

```

Figura 4-5: Publicación de mensaje con MQTT - Bróker
 Realizado por: Borja Edwin, 2020.

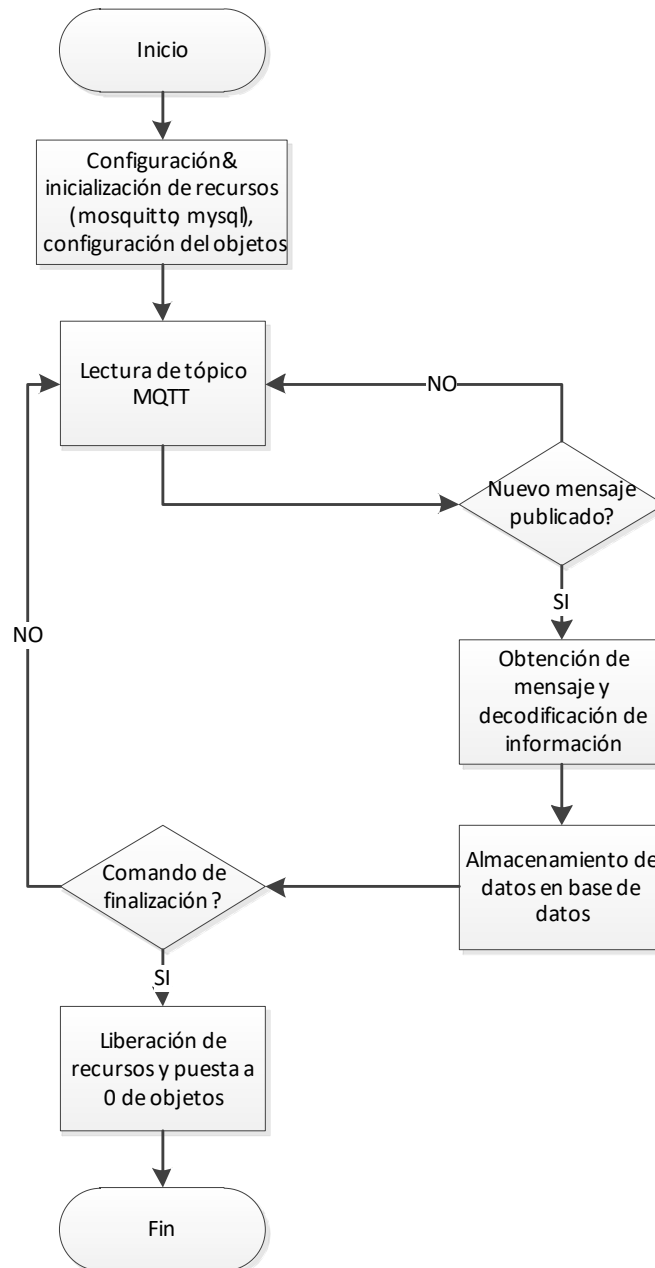


Gráfico 3-5: Diagrama de flujo, BeagleBone Black 2 - Cliente.
 Realizado por: Borja Edwin, 2020.

```
192.168.0.240 - PuTTY
root@beaglebone:/var/www/cgi_bin# ./mqtt
Hi Mosquitto.....
Got message: 0,0,1
B= 0 S= 0 R= 1
█
```

Figura 5-5: Recepción del mensaje por suscripción - Cliente.
Realizado por: Borja Edwin, 2020.

Una vez que tenemos desarrollado en C++ (anexo A) un programa que enlaza una lectura SNAP 7 y una publicación MQTT y otro programa en C++ (anexo B) que se comporta como intermediario entre una suscripción MQTT y una escritura base de datos en MySQL, hace falta un programa en php para el servidor web, el cual nos permita visualizar los mensajes, para eso utilizamos un script que facilite la presentación de los datos mediante los paquetes de Google charts, creamos una gráfica considerando el tiempo, el número de piezas negras, rojas y plateadas, el var data será la que se va a graficar, por lo que nos conectamos al cliente MySQL para imprimir los datos conforme se van ingresando en la data base, para finalmente plasmarlos en una gráfica como se muestra en el gráfico 4-5.



Gráfico 4-5: Presentación del mensaje en un Dashboard en el web service
Realizado por: Borja Edwin, 2020.

CONCLUSIONES

El proyecto desarrollado ha puesto su atención en la aplicación del protocolo MQTT como una alternativa para la monitorización de procesos industriales, buscando plantear el uso de herramientas de software y hardware, open source y de bajo costo respectivamente, que permitan a las industrias tener opciones para modernizar sistemas que mejoren su gestión en los niveles requeridos. Para lo cual, se ha logrado complementar SNAP 7, MQTT y MySQL en dos SBC de bajo costo, como lo es la BeagleBone Black, orientando a obtener un proceso en comunión con el internet de las cosas.

La implementación de la propuesta bajo el nuevo concepto de industria, permite una aproximación al cambio y desarrollo de la industria de nuestro país, fundamentado en el conocimiento, software libre y el uso de dispositivos adaptables de fácil acceso tanto técnico como económico.

El uso de MQTT da flexibilidad para monitorizar diferentes variables o tópicos de interés para los gestores del proceso industrial, así también como su acceso a los datos, siendo posible llevar la información por medio de la red a diferentes plataformas sean estos servidores, computadores o dispositivos móviles.

Por el hecho de ser un protocolo liviano que usa un menor ancho de banda, MQTT es ideal para comunicarse en entornos restringidos, además que permite tener un sistema escalable y con niveles de calidad de servicio aceptables, así por ejemplo para QoS nivel "0" se ha obtenido velocidades de transmisión alentadoras que han permitido aceptar de forma plena la hipótesis planteada mediante la prueba U de Mann - Whitney.

RECOMENDACIONES

Se recomienda implementar la arquitectura sobre un sistema de control que permita una mayor interacción con el proceso industrial, es decir la monitorización y control, donde se analice los niveles de seguridad de información requerida por la industria y posibles vulnerabilidades en aplicación de arquitecturas de comunicación con protocolo MQTT.

Desarrollar una investigación orientada a realizar un análisis profundo del tráfico de datos y utilización de recursos en aplicación de MQTT, así también, realizar una revisión de la actuación de la SBC en ambientes hostiles.

Se debería someter a la arquitectura a un proceso industrial más grande a fin de analizar el comportamiento con más variables y volumen de información, que permita establecer limitantes de procesamiento, almacenamiento y sus posibles opciones.

Resulta importante realizar un análisis en los diferentes niveles de la industria, para identificar los efectos económico administrativos, de obtener información directa del proceso industrial mediante MQTT, de forma que se plasme las ventajas, limitantes y oportunidades que permite la aplicación de este tipo de arquitectura.

BIBLIOGRAFÍA

- A Light, R. (2017). Mosquitto: server and client implementation of the MQTT protocol. *The Journal of Open Source Software*, 2(13), 1–2. <http://doi.org/10.21105/joss.00265>
- About the Apache HTTP Server Project - The Apache HTTP Server Project. (2018). Retrieved September 1, 2018, from http://httpd.apache.org/ABOUT_APACHE.html
- Acatech. (2011). Cyber-Physical Systems - Driving force for innovation in mobility, health, energy and production. *Acatech Position Paper*, (December), 48. <http://doi.org/10.1007/978-3-642-27567-8>
- Baheti, R., & Gill, H. (2011). Cyber-physical Systems. *The Impact of Control Technology*, (1), 161--166. <http://doi.org/10.1145/1795194.1795205>
- Bauernhansl, T., ten Hompel, M., & Vogel-Heuser, B. (2014). *Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung · Technologien · Migration*. <http://doi.org/10.1007/978-3-658-04682-8>
- Beagleboard:BeagleBoneBlack - eLinux.org. (2018). Retrieved April 18, 2018, from <https://elinux.org/Beagleboard:BeagleBoneBlack>
- Blanco, R., Fontrodona, J., & Poveda, C. (2017). La industria 4.0: el estado de la cuestión. *Economía Industrial*, (406), 151–164.
- Casalet R., M. (2017). La convergencia del conocimiento y las tecnologías: Una propuesta explicativa para las estrategias de fabricación., (January).
- Catalán, C., Serna, F., & Blesa, A. (2015). Industria 4.0 en el Grado de Ingeniería Electrónica y Automática. *Actas de Las XXI Jornadas de La Enseñanza Universitaria de Informática*, 327–332. Retrieved from https://upcommons.upc.edu/bitstream/handle/2117/78299/JENUI2015_337-342.pdf
- Deloitte. (2015). Industry 4.0. Challenges and solutions for the digital transformation and use of exponential technologies. *Deloitte*, 1–30.
- Evans, D., Figuerola, N., Fundación de la Innovación Bankinter, Isenstadt, S., Security, N. C.,

- Proves, A., ... Robla, I. (2011). Internet de las cosas: Cómo la próxima evolución de Internet lo cambia todo. *Journal of Food Engineering*, 49(Emim), 314–318. <http://doi.org/10.2991/emim-15.2015.61>
- Evolución Internet de las cosas: conexiones 2012-2020 | Statista. (2019). Retrieved September 29, 2019, from <https://es.statista.com/estadisticas/517654/prevision-de-la-evolucion-de-los-dispositivos-conectados-para-el-internet-de-las-cosas-en-el-mundo/>
- Flores-ruiz, E., Miranda-novalés, M. G., & Villasís-keever, M. Á. (2017). The research protocol VI: How to choose the appropriate statistical test . Inferential statistics El protocolo de investigación VI: cómo elegir la prueba estadística adecuada . *Estadística inferencial*, 64(3), 364–370.
- Frigieri, E. P., Mazzer, D., & Parreira, L. F. C. G. (2015). M2M Protocols for Constrained Environments in the Context of IoT: A Comparison of Approaches. *XXXIII Brazilian Telecommunications Symposium (SBrT)*, (September), 1–4. <http://doi.org/10.14209/sbrt.2015.2>
- García, M. V. (2016). Integración de Comunicaciones de Planta usando una Arquitectura CPPS de bajo costo, 157–161.
- IBM, & Eurotech. (2010). MQTT V3.1 Protocol Specification, 1–42. Retrieved from http://public.dhe.ibm.com/software/dw/webservices/ws-mqtt/MQTT_V3.1_Protocol_Specific.pdf
- Industrie 4.0 - BMBF. (2018). Retrieved January 6, 2018, from <https://www.bmbf.de/de/zukunftsprojekt-industrie-4-0-848.html>
- Internet Society, I. (2015). the Internet of Things :, (October). Retrieved from www.internetsociety.org
- Irisarri, E., García, M. V., Pérez, F., Estévez, E., & Marcos, M. (2016). A Model-Based Approach for Process Monitoring in Oil Production Industry. *IEEE 21th Conference on Emerging Technologies & Factory Automation (ETFA2016)*, 21, 1–2. <http://doi.org/10.1109/ETFA.2016.7733633>
- Kagermann, H., Wahlster, W., & Helbig, J. (2013). Recommendations for implementing the

strategic initiative INDUSTRIE 4.0. *Final Report of the Industrie 4.0 WG*, (April), 82. <http://doi.org/10.13140/RG.2.1.1205.8966>

Kerpen, D., Lohrer, M., Saggiomo, M., Kemper, M., Lemm, J., & Gloy, Y. S. (2016). Effects of cyber-physical production systems on human factors in a weaving mill: Implementation of digital working environments based on augmented reality. *Proceedings of the IEEE International Conference on Industrial Technology, 2016–May*, 2094–2098. <http://doi.org/10.1109/ICIT.2016.7475092>

Kodali, R. K., & Mahesh, K. S. (2016). A low cost implementation of MQTT using ESP8266. *Proceedings of the 2016 2nd International Conference on Contemporary Computing and Informatics, IC3I 2016*, 404–408. <http://doi.org/10.1109/IC3I.2016.7917998>

Kruger, C. P., & Hancke, G. P. (2014). Benchmarking Internet of things devices. *Proceedings - 2014 12th IEEE International Conference on Industrial Informatics, INDIN 2014*, 611–616. <http://doi.org/10.1109/INDIN.2014.6945583>

Molloy, D. (2015). *Exploring BeagleBone Tools and Techniques for Building with Embedded Linux. Climate Change 2013 - The Physical Science Basis*. <http://doi.org/10.1017/CBO9781107415324.004>

Monostori, L. (2014). Cyber-physical production systems: Roots, expectations and R&D challenges. *Procedia CIRP*, 17, 9–13. <http://doi.org/10.1016/j.procir.2014.03.115>

mqtt-v3.1.1. (2014). MQTT Version 3.1.1. Retrieved January 6, 2018, from <http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html>

Mumtaz, S., Alsohaily, A., Pang, Z., Rayes, A., Tsang, K. F., & Rodriguez, J. (2017). Massive Internet of Things for Industrial Applications. *Industrial Electronics Magazine*, (march), 28–33. <http://doi.org/10.13140/RG.2.1.3901.9609>

Nardella, D. (2016). Snap7 Reference manual, 1–289.

Pérez, F., Irisarri, E., Orive, D., Marcos, M., & Estevez, E. (2015). A CPPS Architecture approach for Industry 4.0. *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA, 2015–Octob*. <http://doi.org/10.1109/ETFA.2015.7301606>

- Ríos, J., Ríos, J., & Barrientos, M. (2011). *Monitoreo a través de Sistemas Embebidos a Procesos Industriales*. Universidad Eafit.
- Ruan, J., Yu, W., Yang, Y., & Hu, J. (2016). Design and Realize of Tire Production Process Monitoring System Based on Cyber-Physical Systems. *Proceedings - 2015 International Conference on Computer Science and Mechanical Automation, CSMA 2015*, 175–179. <http://doi.org/10.1109/CSMA.2015.42>
- Saumeth, K., Pinilla, F., Fernandez, A., Muñoz, D., & Cruz, L. (2015). Sistema Ciber-Físico producción de circuitos impresos, (4), 154–165.
- Schneider, S. (2017). The industrial internet of things (IIoT): Applications and taxonomy. *Internet of Things and Data Analytics Handbook*, 41–81. <http://doi.org/10.1002/9781119173601.ch3>
- Sendler, U. (2013). Industrie 4.0. *Beherrschung Der Industriellen Komplexität Mit SysLM*, Online-Ressource (XII, 144 S. 71 Abb, online resou. <http://doi.org/10.1007/978-3-642-36917-9>
- Serna, E. (2017). *Desarrollo e innovación en ingeniería*. Shinde, K. S., & Bhagat, P. H. (2017). Industrial Process Monitoring Using IoT, 38–42.
- Tang, K., Wang, Y., Liu, H., Sheng, Y., Wang, X., & Wei, Z. (2013). Design and Implementation of Push Notification System Based on the MQTT Protocol. *International Conference on Information Science and Computer Applications (ISCA 2013)* , (October 2013), 838–841. <http://doi.org/10.2991/isca-13.2013.20>
- Using Google Charts | Charts | Google Developers. (2018). Retrieved September 3, 2018, from <https://developers.google.com/chart/interactive/docs/?nav=true>
- Val Román, J. (2017). Industria 4.0: la transformación digital de la industria. *Coddiinforme*, 10.
- Wan, J., Cai, H., & Zhou, K. (2015). Industrie 4.0: Enabling technologies. *Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things*, 135–140. <http://doi.org/10.1109/ICAIOT.2015.7111555>
- Zanella, a, Bui, N., Castellani, a, Vangelista, L., & Zorzi, M. (2014). Internet of Things for Smart

Cities. *IEEE Internet of Things Journal*, 1(1), 22–32.
<http://doi.org/10.1109/JIOT.2014.2306328>

Zhou, K. (2015). Industry 4.0: Towards Future Industrial Opportunities and Challenges. *IEEE Xplore*, 2147–2152. <http://doi.org/10.1109/FSKD.2015.7382284>