



**INFOTEC CENTRO DE INVESTIGACIÓN E INNOVACIÓN
EN TECNOLOGÍAS DE LA INFORMACIÓN Y
COMUNICACIÓN**

“Modulo Sensor de Energía basado en Internet de las Cosas (MSE-IoT)”

IMPLEMENTACIÓN DE UN PROYECTO LABORAL
Que para obtener el grado de MAESTRO EN SISTEMAS EMBEBIDOS

Presenta:

Enrique de la Rosa González

Asesor:

Dr. Jesús Antonio Sosa Herrera

Ciudad de México, Noviembre de 2018.



AUTORIZACIÓN DE IMPRESIÓN

Ciudad de México, a 28 de noviembre de 2018

La Gerencia de Capital Humano/ Gerencia de Investigación hacen constar que el proyecto terminal titulado:

"Módulo Sensor de Energía basado en Internet de las Cosas (MSE-IoT)"

Desarrollada por el alumno

Nombre: Enrique

Apellido paterno: De la Rosa

Apellido materno: Gonzalez

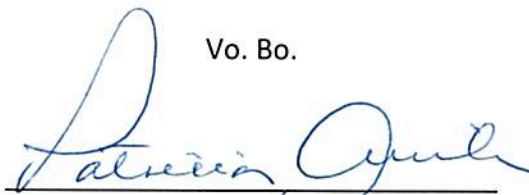
Desarrollado bajo la asesoría de la:

Dr. Jesús Antonio Sosa Herrera

Ha sido revisado y aprobado por miembro del Núcleo Académico Básico (NAB).

Por lo cual, se expide la presente autorización para impresión del proyecto terminal al que se ha hecho mención.

Vo. Bo.

A handwritten signature in blue ink, appearing to read "Patricia Ávila Muñoz", is written over a horizontal line.

Mtra. Patricia Ávila Muñoz
Gerencia de Capital Humano

Agradecimientos

Tabla de contenido

Resumen	1
Introducción	2
Antecedentes	3
Planteamiento y delimitación del problema	3
Objetivo general	3
Objetivos Particulares.....	4
Hipótesis General	4
Capítulo 1: Fundamentos Teóricos	5
1.1 Corriente Alterna.....	5
1.2 Transformadores de Corriente	7
1.3 Internet de las Cosas.....	9
1.3.1. Componentes de una IoT.....	10
1.3.2. Conexión Local Maquina a Maquina (M2M)	11
1.4. Computación en la nube (Cloud Computing).....	12
1.4.1. Software como servicio (SaaS).....	12
1.4.2. Infraestructura como servicio (IaaS).....	12
1.4.3. Plataforma como servicio (PaaS)	13
1.5. Protocolos de Comunicación IoT	14
1.5.1. HTTP/HTTPS.....	14
1.5.2. WebSocket.....	15
1.5.3. CoAP	15
1.5.4. AMQP.....	16
1.5.5. MQTT	17
1.6. IBM Bluemix.....	18
1.7. Watson IoT de Bluemix	20
1.7.1. Comunicación entre dispositivos IoT, Plataforma IBM IoT y Aplicación.....	20
1.7.2. Visualización de Datos en IBM Watson IoT de Bluemix.....	21
1.8. Node-RED.....	22
1.9. Cloudant NO-SQL Database.....	24
Capítulo 2: Modulo Sensor de Energía basado en Internet de las Cosas (MSE-IoT).....	26
2.1 NodeMCU	27
2.1.1. ESP-12	28
2.2. Sensor de Corriente SCT-013-030	29
2.2.1. Esquema de Montaje Sensor de Corriente	30
2.3. Sensor de Voltaje	31
2.4. Convertidor Analógico Digital ADS1115	31

2.5.	Diseño de Software	32
Capítulo 3: Pruebas		40
3.1.	Medición de Corriente	41
3.2.	Medición de Voltaje	41
3.3.	Selección del proveedor de la nube	42
3.3.1.	IBM Bluemix (IBM Cloud).....	42
3.4.	Crear la cuenta en Bluemix	43
3.5.	Internet de las cosas de IBM Bluemix	43
3.6.	Creando una aplicación.....	44
3.7.	Registro del dispositivo.....	46
Capítulo 4: Resultados		48
Conclusiones		53
Bibliografía		54
Anexos		56
Anexo I	Diagramas Eléctricos.....	56
Anexo II	PCB's (circuitos impresos)	58
Anexo III	Diagramas NodeMCU	59

Índice de figuras

Figura 1 Triangulo de la Potencia	6
Figura 2 Transformador de Corriente [3]	7
Figura 3 Conexión interna Transformador de Corriente	8
Figura 4 Protocolo HTTP/HTTPS	14
Figura 5 Protocolo WebbSocket	15
Figura 6 Protocolo CoAP	16
Figura 7 Protocolo AMQP	16
Figura 8 Protocolo MQTT	17
Figura 9 Esquema conexión IBM Bluemix [10]	18
Figura 10 Comunicación Dispositivo IoT a IBM Bluemix	20
Figura 11 Node-RED [13]	22
Figura 12 Node-RED, JSON [14]	23
Figura 13 Nodos básicos de Node-RED	23
Figura 14 Esquema MSE-IoT	26
Figura 15 Node-MCU [25]	27
Figura 16 Diagrama a bloques del ESP8266 [17]	28
Figura 17 Sensor SCT-013-030 [18]	29
Figura 18 Sensor FZ0430 [19]	31
Figura 19 Circuito Eléctrico Sensor FZ0430	31
Figura 20 Diagrama a Bloques MSE-IoT	33
Figura 21 Diagrama Funcional	35
Figura 22 Función para obtener Irms	36
Figura 23 Código función Irms	37
Figura 24 Parámetros Conexión a IBM Bluemix	37
Figura 25 Formación Payload	38
Figura 26 Diagrama Funcional IBM Bluemix [21]	38
Figura 27 Conexión del sensor de corriente a la toma principal	40
Figura 28 Prototipo MSE-IoT	40
Figura 29 Conexión para medición de corriente	41
Figura 30 Conexión para medición de voltaje	42
Figura 31 Cuenta Cloudant NoSQL	44
Figura 32 Aplicación MSE-IoT	45
Figura 33 Aplicación en Node-RED	45
Figura 34 Conexión del Dispositivo	46
Figura 35 Datos en Pantalla	48
Figura 36 Visualización de Datos en IBM Bluemix	49
Figura 37 Registro de en Base de Datos	49
Figura 38 Formato de Información en JSON	50
Figura 39 Información en plataforma Emoncms	51
Figura 40 Consumo Energético	51

Figura 41	Envío de alertas a correo electrónico.....	52
Figura 42	Envío de alertas a Twitter	52
Figura 43	Diagrama Eléctrico modulo Procesador y Display	56
Figura 44	Diagrama eléctrico módulo de potencia	56
Figura 45	Modulo de Procesador y Display	57
Figura 46	PCB modulo Procesador y display	58
Figura 47	PCB Modulo de Potencia	58
Figura 48	Diagrama a bloques NodeMCU [22].....	59
Figura 49	Asignación de pines Node-MCU [23].....	59

Resumen

En este trabajo se presenta el diseño e implementación de un medidor de energía eléctrica haciendo uso del microcontrolador ESP8266, Internet de las cosas, así como de servicios en la nube.

Internet de las cosas (IoT) es un campo emergente y los dispositivos basados en IoT han creado una revolución en la electrónica y la informática. La intención principal del medidor de energía propuesto es monitorear el consumo de energía eléctrica enviar la información a una plataforma en la nube, el sistema de monitoreo de energía calcula la potencia consumida por los dispositivos eléctricos y lo muestra a través de la plataforma IBM Watson IoT.

La ventaja de este dispositivo es que un usuario puede entender la energía consumida por los aparatos eléctricos y puede tomar medidas adicionales para controlarlos y, por lo tanto, ayudar en la conservación de energía. Adicionalmente, los usuarios pueden controlar el consumo de energía, así como la factura sobre una base diaria.

Introducción

La conservación de la energía es una de las necesidades más importantes de hoy en día. El concepto de dispositivos eficientes en energía ha surgido en diversos ámbitos tales como iluminación, aire acondicionado y así sucesivamente. El monitoreo de energía es una herramienta importante para determinar la eficiencia energética de varios dispositivos. El presente trabajo implementa un sistema de monitoreo de energía eléctrica que muestra la potencia consumida por dispositivos individuales o múltiples. Esto puede ayudar al usuario a detectar cualquier error en la factura de electricidad.

Un sistema inteligente de monitoreo de energía puede ayudar al usuario a analizar los datos de consumo de energía a nivel de dispositivo y administrarlo en lugar de suponer que se trata de un gasto mensual fijo. Además, ayuda al usuario a reemplazar los aparatos regulares por otros eficientes. Es importante destacar que el sistema de monitoreo puede alertar al usuario sobre el exceso de consumo inesperado causado por mal funcionamiento del equipo, falta de mantenimiento adecuado y similares. Además, el manejo adecuado de la energía puede hacer posible un mejor presupuesto.

Internet de las cosas (IoT) ha abierto una plétora de aplicaciones en numerosos campos como los sistemas médicos y de salud, la domótica inteligente y el monitoreo ambiental. Se espera que el IoT genere una gran cantidad de cambios en el campo de la computación ubicua. Un sistema de gestión de energía basado en IoT puede contribuir mucho en la conservación de la energía.

La facturación de consumo eléctrico es generada mensual o bimestralmente, con ello el usuario tiene la opción de analizar los detalles de consumo generados en este periodo. El medidor de energía instalado en los edificios residenciales muestra la energía consumida por el hogar. Muy a menudo, los dispositivos que funcionan en modo de espera consumen una cantidad significativa de energía sobre la cual el cliente final no es consciente. Por lo tanto, existe una fuerte necesidad de un nuevo sistema de monitoreo de energía que puede mostrar el consumo de energía de diferentes dispositivos en modo normal así como en modo de espera y

también alertar al usuario sobre el aumento inesperado en el consumo de energía sobre una base diaria.

Antecedentes

El monitoreo y la conservación de la energía tiene una importancia primordial en el mundo actual debido al desequilibrio entre la generación de energía y la demanda. En la actualidad la mayoría de la energía generada que es principalmente aportada por los combustibles fósiles, puede llegar a agotarse en los próximos 20 años.

Actualmente, hay sistemas de monitoreo de energía electrónicos muy precisos disponibles en el mercado. La mayoría de estos monitores de la energía consumida en un hogar doméstico, en caso de aplicaciones residenciales. Muchas veces, los consumidores están insatisfechos con la factura de energía, ya que no muestra la potencia consumida a nivel de dispositivo.

Planteamiento y delimitación del problema

Actualmente no se cuenta con dispositivos de bajo costo enfocados al uso doméstico que permitan hacer medición, almacenamiento y reporte de la información de consumo de energía eléctrica.

Medición de corriente y voltaje eléctrico para la determinación de consumo de energía eléctrica,

Objetivo general

El presente trabajo tiene como fin el diseño y construcción de un módulo que hace uso de la tecnología de internet de las cosas y servicios en la nube para el monitoreo de consumo de energía eléctrica de una residencia.

Objetivos Particulares

- Detección y registro de uso potencia eléctrica mediante el registro del voltaje y corriente eléctrica.
- Visualización datos de Corriente, Voltaje y Potencia Eléctrica en Display
- Conexión a plataforma de Internet de la Cosas
- Registro en Base de datos de mediciones para futuros análisis
- Visualización grafica de corriente, voltaje y potencia eléctrica así como en aplicación en la Nube.
- Envío de alertas de bajo o alto consumo a correo electrónico, Twitter.

Hipótesis General

Inferencias por parte del usuario a partir de los datos visuales mostrados y así como de alertas.

Detectar el funcionamiento de un aparato eléctrico con comportamiento cíclico de carga.

Detectar patrones de comportamiento del consumo energético, cuando se encuentran personas en casa, así como el comportamiento los fines de semana.

Hacer uso de plataforma cloud e IoT que no presenten un costo para el usuario.



Capítulo 1

Fundamentos Teoricos

Capítulo 1: Fundamentos Teóricos

Un medidor de potencia que mide la potencia real y otros parámetros necesita adquirir datos de voltaje y corriente. Por lo tanto, se requieren sensores de voltaje y corriente.

El diseño implica definiciones de los parámetros que debe calcular el software, a saber, voltaje RMS (VRMS), corriente RMS (IRMS), potencia real (P), potencia aparente (S), factor de potencia (PF).

1.1 Corriente Alterna

Voltaje y Corriente RMS (Root-Mean-Square): El voltaje RMS es la representación del valor efectivo de un voltaje de CA (corriente alterna). Tal valor se define como la magnitud equivalente a la tensión de CC que se aplica a la misma carga y que produce la misma potencia. Se puede definir como (1):

$$V_{RMS} = \sqrt{\frac{\sum_{j=1}^N V_{pVTj}^2}{N}} \quad (1)$$

$$I_{RMS} = \sqrt{\frac{\sum_{j=1}^N I_{pCTj}^2}{N}} \quad (2)$$

Donde V_{pTj} es la j -ésima tensión instantánea y N es el número de muestras.

La corriente media cuadrática (IRMS) definida en (2) usa el mismo concepto que en VRMS.

Potencia Real: hay tres tipos de energía en la electricidad de CA: potencia real / activa, potencia reactiva y potencia aparente. La potencia reactiva no genera trabajo y no se mide en este diseño de medidor de potencia. Por otro lado, la potencia real genera trabajo y es esencial para calcular el consumo de energía.

Se define como (3) siempre que las funciones de tensión instantánea ($V_{pVT}(t)$) y de corriente ($I_{pCT}(t)$) con respecto al tiempo (t) sean señales continuas. Sin embargo, la tensión y la corriente instantáneas en el medidor de potencia digital se muestrean, por lo tanto, valores discretos. Por lo tanto, la potencia real aproximada se puede definir como (4), donde V_{pVTj} es la j -ésima muestra de voltaje, I_{pCTj} es la j -ésima muestra de corriente, y N es el número total de muestras [1].

$$P = \frac{1}{T} \int_{\tau}^{\tau+T} (V_{pVT}(t) \times I_{pCT}(t)) dt \quad (3)$$

$$P = \frac{1}{N} \sum_{j=1}^N (V_{pVTj} \times I_{pCTj}) \quad (4)$$

Potencia aparente: la potencia aparente es la suma vectorial, cuando se representa en el triángulo de potencia como en la figura 1, de las potencias real y reactiva. Se puede definir por (5), donde S denota la potencia aparente.

$$S = V_{RMS} \cdot I_{RMS} \quad (5)$$

Factor de potencia: el factor de potencia se define como el coseno del ángulo entre las potencias real y aparente, cuando las potencias real P , Q reactiva y S aparente se ilustran como vectores como en la figura 1. 'es el ángulo entre la potencia real y aparente.

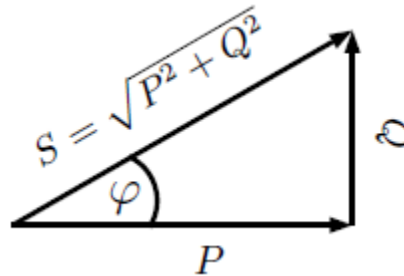


Figura 1 Triangulo de la Potencia

Fuente: T. T. P. I. Nattachart Tamkittikhun, «AC Power Meter Design for Home Electrical Appliances

$$PF = \frac{P}{S}$$

1.2 Transformadores de Corriente

Los transformadores de corriente (TC) son sensores que miden la corriente alterna (CA). Son particularmente útiles para medir el consumo o generación de electricidad.

El tipo de núcleo dividido, el CT puede acoplarse al cable vivo o neutro de la toma principal sin la necesidad de realizar ningún trabajo eléctrico de alto voltaje [2].

Como cualquier otro transformador, un transformador de corriente tiene un devanado primario, un núcleo magnético y un devanado secundario, la figura 2 muestra construcción básica y diagrama de un transformador/sensor de corriente del tipo no invasivo.

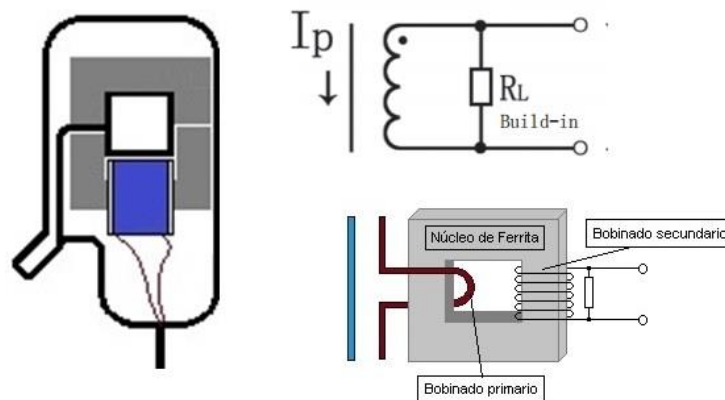


Figura 2 Transformador de Corriente [3]

Fuente: <http://www.naylampmechatronics.com/img/cms/Blog/Tutorial%20SCT-013/sensores%20CT.jpg>

La corriente en el devanado secundario es proporcional a la corriente que fluye en el devanado primario:

$$I_{\text{secundario}} = \text{TCRangoVueltas} \times I_{\text{primario}}$$

$$\text{TCRangoVueltas} = \text{Vueltasprimario} / \text{Vueltassecundario}$$

Se puede emplear esto para construir un sensor de corriente no invasivo. En un sensor de corriente el núcleo ferromagnético puede estar dividido de forma que pueda abrirse y arrollar un conductor.

La relación de transformación de corriente depende de la relación entre el número de espiras (7).

$$\frac{I_s}{I_p} = \frac{V_p}{V_s} = \frac{N_p}{N_s} \quad (7)$$

A diferencia de los transformadores de tensión, en un transformador de corriente el circuito secundario nunca debería estar abierto, porque las corrientes inducidas podrían llegar a dañar el componente. Por ese motivo, sensores de corriente disponen de protecciones (resistencia burden en los sensores de salida por tensión, o diodos de protección en los sensores de salida por corriente, diagrama de conexión interna en la figura 3).

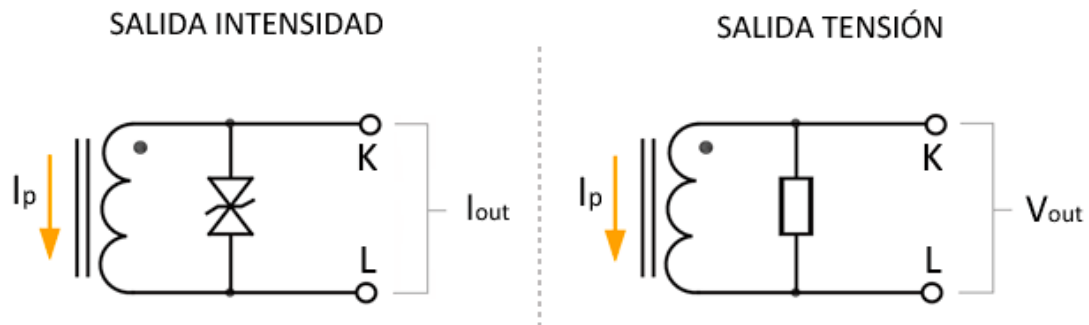


Figura 3 Conexión interna Transformador de Corriente

Fuente: <https://www.luisllamas.es/arduino-sensor-corriente-sct-013/>

Tanto los transformadores de tensión como los de corriente inducen el voltaje o la corriente en el primario al secundario. El voltaje o la corriente en el primario (V_p / I_p) suelen ser más altos que en el secundario (V_s / I_s). Sin embargo, lo opuesto también existe. El voltaje y la corriente tienen una relación con los giros primarios (N_p) y secundarios (N_s). Algunos transformadores de corriente también vienen con una resistencia de carga en el secundario como una pequeña carga ficticia para

producir voltaje en ambos extremos del TC secundario. Las señales en el secundario de estos transformadores son tan pequeñas que son apropiadas para la entrada de un módulo de procesamiento.

1.3 Internet de las Cosas

El Internet de las Cosas es un concepto que hace referencia a la interconexión de objetos cotidianos en una red, “los objetos adquieren una identidad propia, detectando el entorno en el que se encuentran e intercambiando información”. Dispositivos y personas están conectados y pueden comunicarse entre sí en lo que se ha denominado un entorno ubicuo [4].

Puede definirse como una infraestructura de red global dinámica con capacidades auto configurables basadas en estándares y protocolos de comunicación donde los “objetos” físicos y virtuales llevan asociados una identidad así como atributos físicos y personalidades virtuales que se integran perfectamente en la red de información.

IoT nace como una evolución de la tecnología actual de Internet y la rápida masificación de dispositivos inteligentes. A diferencia de otros procesos tecnológicos, este no es consecuencia de una teoría científica sino más bien de una sinergia entre conceptos de carácter comercial, experimental y técnico, siendo su principal impulsor el desarrollo de las etiquetas RFID. IoT es la forma de describir dos esquemas estrechamente relacionados. Uno es la “Internet de la información de productos” el cual se centra en el fácil acceso a través de la web de información útil sobre un producto (fecha de fabricación, caducidad, detalles técnicos de fabricación, distribución al consumidor, etc.) en este caso los objetos se comportan como una fuente de información cuyo contenido no depende de ellos de forma directa pues los datos son manipulados y permanecen en la web. El otro esquema es el “Internet de los sensores y actuadores” en el cual se busca tener no sólo la posibilidad de conocer información de un entorno, sino la capacidad de interactuar con él ya sea de forma local o remota mediante el uso de redes; en este caso la

información proviene de las cosas propiamente dichas y se hace uso de diversos recursos tecnológicos a fin de lograr la comunicación e interacción [5].

De forma general se podría hablar entonces de Internet de las cosas como la “Internet de información de objetos” sean estos artículos de consumo o redes de sensores-actuadores como los que se encuentran en los ambientes domóticos actuales. La gran mayoría de las conexiones a Internet se establecen entre los dispositivos utilizados directamente por los seres humanos, tales como computadoras y teléfonos móviles, la interacción principal es entre personas. La tendencia actual es que también los objetos se puedan conectar a la red. Las cosas, comprendidas estas como elementos cotidianos a los cuales se les ha añadido algún grado de inteligencia, tienen cada vez mayor capacidad para intercambiar información por sí mismas y entre ellas. Se estima que en algún momento la cantidad de “cosas” conectadas a Internet será mucho más grande que el número de “personas” y los seres humanos nos convertiremos en la minoría de los generadores y receptores de tráfico.

1.3.1. Componentes de una IoT

El esquema general del internet de las cosas consiste en un conjunto que posee los siguientes componentes:

Cosas: Son los distintos objetos que poseen alguna inteligencia, estos se deben considerar como los emisores-receptores de la información. Según sea su nivel de inteligencia pueden comportarse como “productos” o como sistemas de “sensores-actuadores”. Idealmente todas las cosas deben poseer una etiqueta que les identifique de forma única.

Identificadores: Son las entradas a la información contenida en una cosa, estas pueden ser soportes pasivos, como códigos QR o códigos de barras, o poseer capacidad de conexión como las etiquetas RFID, también podrían considerarse como etiquetas otras formas de identificación de dispositivos como solicitudes o ID de conexión a través de una red de dispositivos, en estos casos la identificación se da por la ubicación (proximidad) entre los objetos.

Zona de interfaz física: Corresponde a la ubicación física real de los dispositivos (cosas) que poseen algún grado de inteligencia e interacción con el entorno y son capaces de comunicarse a través de una red, dicha zona de interfaz física cumple con las condiciones generales de un sistema ubicuo.

Red de cosas: Los distintos objetos “inteligentes” pueden comunicarse al exterior o entre ellos a través de una red. Dada la gran cantidad de tecnologías de comunicación existentes, que en muchos casos son incompatibles entre sí, una red de cosas debe garantizar ante todo la posibilidad de conectar una cosa con otra a través de una pasarela adecuada.

Pasarelas: Son los dispositivos encargados de conectar la red de cosas con Internet. Como en el caso de las pasarelas Web, el hardware también podría ejecutar tareas de seguimiento y control e incluso comportarse como servidor local a fin de efectuar una gestión de la información.

Internet: Se convierte en el medio a través del cual se establece interacción remota entre cosas así como el ente que almacena y gestiona la información. IoT puede hacer uso, o poseer recursos paralelos a los de la Internet de las personas: Servidores Web, DNS, protocolos específicos para la localización y la comunicación, etc. En este aspecto el desarrollo apenas se encuentra en una fase experimental y la estandarización de conceptos y métodos se encuentra incompleta o aun sin establecer.

1.3.2. Conexión Local Maquina a Maquina (M2M)

Una de las críticas válidas que se hace a los esquemas del internet de las cosas es que requieren del uso de una gran cantidad de componentes a fin de lograr una interacción tanto local como remota; sin embargo en muchos casos la interactividad entre cosas se da únicamente en entornos locales en los cuales los objetos se encuentran muy cerca entre sí y la información que se comparte entre ellos sólo tiene importancia y validez en breves intercambios que se suceden entre estos. Es por esta razón que el establecimiento de las interacciones del tipo M2M locales revisten una gran importancia al momento de efectuar el diseño de una red de

objetos, así mismo representan un desafío ya que la gran mayoría de dispositivos que se encuentran actualmente en el mercado (PDA, Tablet, PC portátil, entre otros) aun cuando incluyen diversos sistemas de comunicación (infrarrojos, Bluetooth, Wi-fi, entre otros) que no permiten de una forma fácil el intercambio entre métodos de comunicación para una misma interacción.

1.4. Computación en la nube (Cloud Computing)

El término nube, junto con la descripción de una tecnología, también se utiliza para describir una forma de modelar las ofertas de negocios. La computación en nube es una entrega de la informática como un servicio, donde se comparte un conjunto de recursos físicos y / o virtuales, en lugar de implementar un hardware y software local o personal (Marechaux, 2015) [7]. Hay tres niveles de servicio de cloud computing (Amazon, n.d.-a):

1.4.1. Software como servicio (SaaS)

Es una oferta de computación en la nube que ofrece a los usuarios acceso al software basado en la nube de un proveedor. Los usuarios no instalan aplicaciones en sus dispositivos locales. En su lugar, las aplicaciones residen en una red de nube remota a la que se accede a través de la Web o una API. A través de la aplicación, los usuarios pueden almacenar y analizar datos y colaborar en proyectos.

1.4.2. Infraestructura como servicio (IaaS)

El modelo IaaS (Infraestructura como servicio) proporciona componentes de infraestructura a los clientes.

Los componentes pueden incluir máquinas virtuales, almacenamiento, redes, cortafuegos, equilibradores de carga, etc. Con IaaS, los clientes tienen acceso directo al software de nivel más bajo de la pila, es decir, al sistema operativo en

máquinas virtuales o al panel de administración de un firewall o balanceador de carga.

La infraestructura como servicio ayuda a las empresas a trasladar su infraestructura física a la nube con un nivel de control similar al que tendrían en un centro de datos tradicional en las instalaciones. IaaS proporciona el parecido más cercano al centro de datos interno en comparación con otros tipos de servicios.

Los componentes centrales de la infraestructura del centro de datos son almacenamiento, servidores (unidades de computación), la red misma y herramientas de administración para el mantenimiento y la supervisión de la infraestructura. Cada uno de estos componentes ha creado un nicho de mercado separado.

1.4.3. Plataforma como servicio (PaaS)

PaaS (Plataforma como servicio) es un modelo de servicio en la nube que ofrece a los clientes una plataforma de aplicaciones configurable que incluye una pila de software preinstalada. PaaS se puede ver como otra capa de abstracción sobre el hardware, el sistema operativo y la pila de virtualización.

El modelo PaaS aporta un valor significativo a las empresas porque reduce la complejidad de la infraestructura y el mantenimiento de las aplicaciones y permite concentrarse en el software central.

Características del servicio PaaS:

- Escalabilidad y autoaprovisionamiento de la infraestructura subyacente
- Seguridad y redundancia
- Herramientas de compilación e implementación para una rápida administración e implementación de aplicaciones
- Integración con otros componentes de infraestructura, como servicios web, bases de datos y LDAP
- Multi-tenancy, servicio de plataforma que puede ser utilizado por muchos usuarios concurrentes
- Registro, informes e instrumentación de código

1.5. Protocolos de Comunicación IoT

Para la comunicación en Internet y otras redes informáticas, se utiliza el stack de protocolos de Internet (IP) que consta de cuatro capas: capa de aplicación, transporte, IP y enlace. TCP (Transmission Control Protocol), un protocolo de capa de transporte, proporciona comunicación de extremo a extremo con entrega fiable, sin errores y en secuencia de paquetes IP. TCP utiliza una negociación de tres vías para garantizar una conexión fiable entre los extremos. UDP (User Datagram Protocol) es menos confiable, y la entrega y la duplicación no están garantizadas. REST (Representational State Transfer), un estilo de arquitectura, se utiliza para diseñar aplicaciones en red (Elkstein, 2008) [8].

1.5.1. HTTP/HTTPS

HTTP, IETF estándar (HTTP / 1.1 en RFC 2616), es un protocolo de capa de aplicación que opera a través del protocolo TCP (o similar) y se utiliza para transmitir datos en la Web. Transport Layer Security (TLS) y su predecesor Secure Sockets Layer (SSL) se utilizan para el cifrado y descifrado de paquetes de datos cuando se mueve de un punto final a otro. Esta información oculta se hace para evitar caer como la comunicación web no segura es vulnerable a los ataques cibernéticos. HTTPS es un protocolo HTTP seguro que utiliza el framework SSL / TLS. HTTP / HTTPS utiliza el método de solicitud / respuesta, como se muestra en la figura 4.

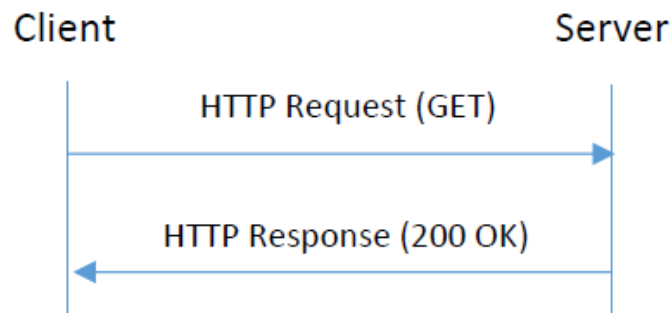


Figura 4 Protocolo HTTP/HTTPS

Fuente: D. B. O. E. Olivier Hersent, The Internet of Things: Key Applications and Protocols.

1.5.2. WebSocket

WebSocket (Kaazing Corporation, nd) El WebSocket, diseñado como un protocolo de capa de transporte, es un complemento de HTTP. La comunicación se realiza a través de un único socket entre cliente y servidor, mostrado en la figura 5. Otros protocolos como MQTT (Message Queuing Telemetry Transport), AMQP (Advanced Message Queue Server Protocol), pub / sub se pueden ejecutar en la parte superior de WebSockets para proporcionar más flexibilidad y potencia. HTML5 WebSocket es un estándar que ayuda a construir escalable en tiempo real Aplicaciones Web. WebSocket admite enviar tanto texto como frames binarios al mismo tiempo en qué dirección.

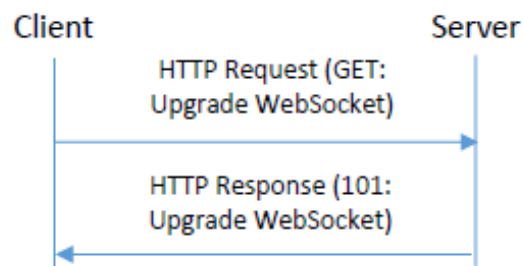


Figura 5 Protocolo WebbSocket

Fuente: D. B. O. E. Olivier Hersent, The Internet of Things: Key Applications and Protocols.

La conexión entre el cliente y el servidor permanece abierta cuando la conexión se actualiza a WebSocket desde el protocolo HTTP en el handshake inicial y, por lo tanto, no hay necesidad de enviar otra solicitud al servidor como era el caso con HTTP.

1.5.3. CoAP

El protocolo de aplicación restringido (CoAP), un estándar de IETF (IETF, 2014), es un protocolo de transferencia web para nodos con restricciones de recursos y redes restringidas (por ejemplo, de baja potencia, con pérdidas) como se muestra en la

figura 6. Los nodos constreñidos a menudo tienen una pequeña cantidad de ROM y RAM. Este protocolo se basa en el modelo de solicitud / respuesta y utiliza datagramas (Sheng, et al., 2013) CoAP optimiza un subconjunto de funciones HTTP para el entorno restringido (es decir, UDP).

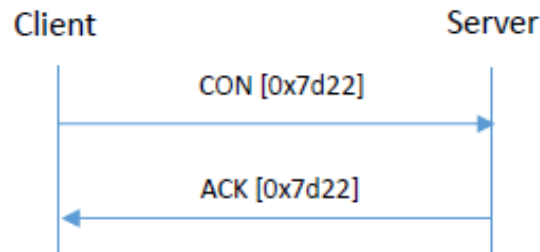


Figura 6 Protocolo CoAP

Fuente: D. B. O. E. Olivier Hersent, The Internet of Things: Key Applications and Protocols.

La confiabilidad es proporcionada por CON (mensaje confirmable) y su ACK (acuse de recibo) con el mismo ID de mensaje También tiene descubrimiento de recursos incorporado.

1.5.4. AMQP

AMQP (Advanced Message Queuing Protocol), es un protocolo abierto de mensajería (OASIS, 2012) que es posible construir aplicaciones multiplataforma usando diferentes intermediarios, bibliotecas y frameworks. Las características principales son una comunicación fiable (exactamente una vez) e interoperabilidad, fue diseñada como un reemplazo abierto para middleware de mensajería propietario existente.

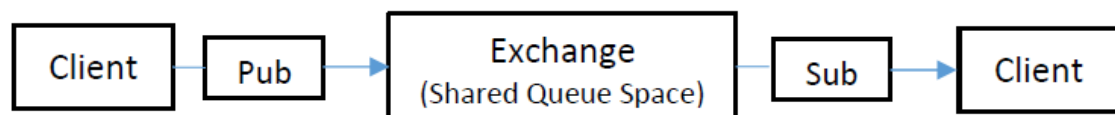


Figura 7 Protocolo AMQP

Fuente: D. B. O. E. Olivier Hersent, The Internet of Things: Key Applications and Protocols.

La Figura 7 muestra los principales componentes del modelo de comunicación AMQP. El cliente publica el mensaje en Exchange (una parte del intermediario) donde se procesa y enruta a la cola. Desde la cola, el mensaje se envía al Cliente que se ha suscrito para ello. El módulo de encapsulamiento encamina el mensaje entre Exchange y Queue (por defecto vincula los mapas de la clave de enrutamiento al nombre de la cola). AMQP proporciona una secuencia de tramas full-duplex, confiablemente ordenada.

1.5.5. MQTT

MQTT es un protocolo de capa de aplicación que funciona sobre TCP / IP en la capa de transporte para la transferencia de mensajes. Es un protocolo adecuado para dispositivos con recursos limitados y fue inventado por el Dr. Andy Stanford-Clark y Arlen Nipper en 1999. Es un protocolo liviano, abierto, simple y fácil de usar, lo que lo hace ideal para la comunicación en entornos con recursos limitados, como IoT. El objetivo de diseño de MQTT es garantizar la entrega confiable de mensajes en entornos restringidos, como un bajo ancho de banda de red y redes no confiables para dispositivos con recursos limitados. Y la entrega del mensaje se realiza utilizando el protocolo de mensajería de publicación-suscripción de cliente-servidor, mostrado en la figura 8.

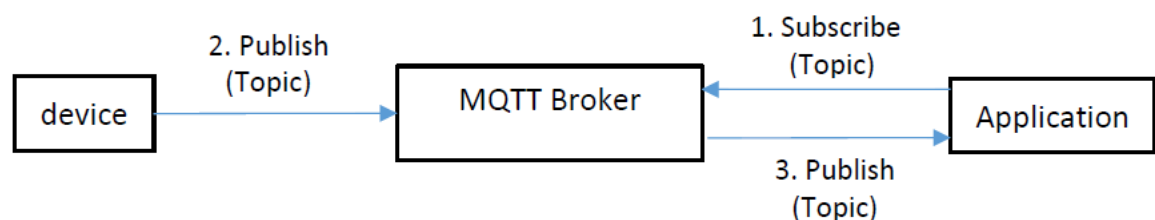


Figura 8 Protocolo MQTT

Fuente: D. B. O. E. Olivier Hersent, The Internet of Things: Key Applications and Protocols.

El protocolo MQTT define dos tipos de entidades en la red: un intermediario de mensajes y un número de clientes. El intermediario es un servidor que recibe todos los mensajes de los clientes y los envía a clientes de destino relevantes. Un cliente es cualquier cosa que pueda interactuar con el corredor para enviar y recibir

mensajes. Un cliente puede ser un sensor IoT en el campo o una aplicación en un centro de datos que procesa datos IoT.

1.6. IBM Bluemix

IBM Bluemix es una Plataforma como servicio (PaaS) basada en la nube, que se ejecuta en Softlayer, que es la infraestructura mundial de IBM como Servicio (IBM, 2016-b). IBM Bluemix ofrece tres métodos de implementación: Public, Dedicated Blueprints El servicio DevOps, un método de desarrollo de software, hace hincapié en la comunicación, la colaboración, la integración y la automatización. Con DevOps, se puede planificar, desarrollar y probar una aplicación en línea .Es también una posibilidad para probar si el proyecto se hace público o se almacenan con acceso privado Bluemix proporciona incorporado en la gestión de software, y editor en línea para desarrollar una aplicación [9]. La estructura de IBM Bluemix se muestra en la Figura 9.

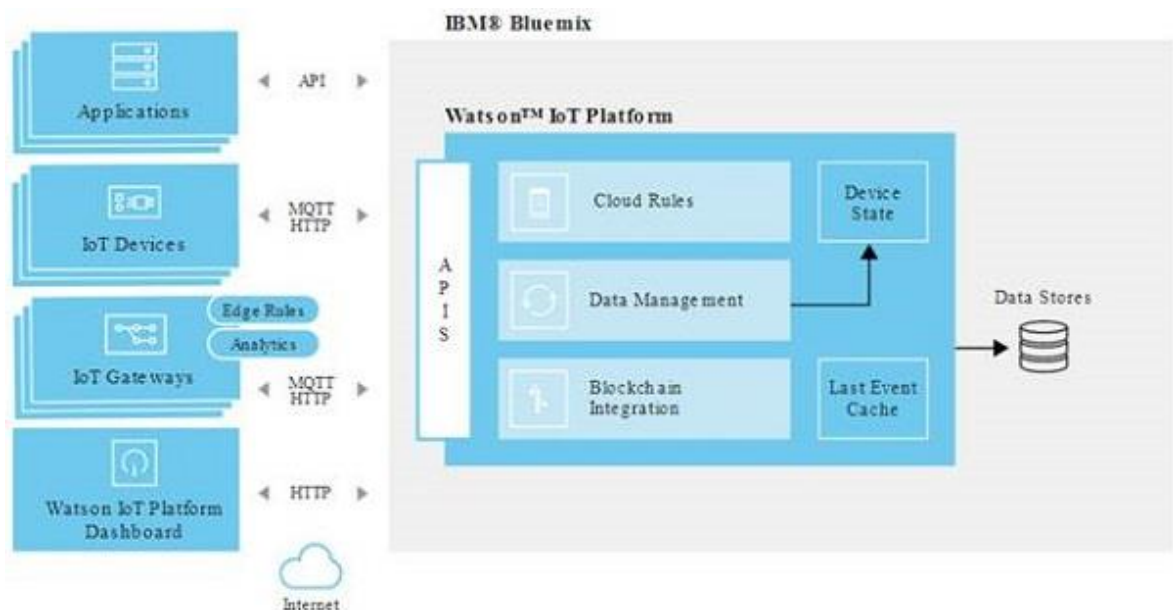


Figura 9 Esquema conexión IBM Bluemix [10]

https://www.codeguru.com/imagesvr_ce/4489/Watson2.jpg

Bluemix, está basado Cloud Foudry, tiene el soporte de múltiples lenguajes de programación incluyendo Java, Node.js, Go, PHP, Python, Ruby Sinatra, Ruby on Rails, Scala (extendido a través del paquete de compilación) Bluemix también

ofrece diferentes modelos de computación para ejecutar el código IBM Bluemix también proporciona servicios de base de datos y base de datos de números: No hay servicio de base de datos SQL (por ejemplo, Cloudant, MongoLab), servicio de base de datos SQL (por ejemplo, DB2, ClearDB, Elephant SQL), servicio de base de datos de pares de claves (por ejemplo, Redis Cloud, IBM Data Cache), base de datos de series temporales.

En el enfoque de PaaS de Bluemix, el cliente de Bluemix gestiona las aplicaciones y los datos, mientras que el proveedor (Bluemix) gestiona el tiempo de ejecución, el middleware, el o / s, la virtualización, el servidor, el almacenamiento y las redes. Como se muestra en la Figura 9, el cliente de Bluemix (es decir, cliente de aplicaciones móviles, clientes de aplicaciones web), utiliza la interfaz basada en navegador de Bluemix para interactuar con los servicios de Bluemix. , desarrolladores de aplicaciones) pueden utilizar las API REST o HTTP para encaminar las solicitudes a la instancia de la aplicación Bluemix o al servicio compuesto [11].

IBM Bluemix usa las siguientes terminologías:

Space: Un espacio en Bluemix es un agrupamiento lógico de todos los recursos que usted consume en Bluemix.

Boillierplate: Boilerplate proporciona los enlaces rápidos de las aplicaciones y el entorno de ejecución asociado, así como los servicios predefinidos para un dominio en particular.

Service: Bluemix ofrece una gran cantidad de servicios que se pueden elegir mientras se desarrolla una aplicación web o móvil. El servicio de la plataforma Internet of Things se puede utilizar para administrar y controlar dispositivos IoT. Otros servicios como Watson (para construir aplicaciones cognitivas), servicio de base de datos, etc. se pueden combinar con los servicios de plataforma IoT para construir aplicaciones avanzadas.

Application: IBM Bluemix proporciona un entorno de tiempo de ejecución para crear la aplicación basada en web o móvil. Bluemix admite múltiples idiomas, como antes, que pueden utilizarse para crear aplicaciones interactivas.

1.7. Watson IoT de Bluemix

IBM Watson para IoT está disponible a través de la plataforma IBM Bluemix. IBM Bluemix permite crear aplicaciones basadas en web en diferentes lenguajes, incluyendo PHP, java, node.js. El servicio IoT de Bluemix colecta los datos de los dispositivos inteligentes y puede enviarlo a las aplicaciones que se unen a él. Así mismo puede incluir las ofertas de IBM Watson IoT para Bluemix incluyen analítica, administración de riesgos, conexión y administración de información.

Se pueden conectar los dispositivos habilitados para IoT usando el protocolo MQTT a la plataforma Watson IoT, que también soporta la administración de dispositivos y la visualización de datos en tiempo real.

La función de gestión de riesgo de la nube de IBM Watson IoT garantiza proporcionar información correcta y actualizada.

1.7.1. Comunicación entre dispositivos IoT, Plataforma IBM IoT y Aplicación.

Un dispositivo IoT puede conectarse a la plataforma IoT basada en la nube de IBM a través del protocolo MQTT.

IBM Watson IoT para Bluemix permite a sus usuarios crear aplicaciones basadas en web o móviles.

La comunicación entre la plataforma IoT y la aplicación puede ser posible a través del protocolo HTTP a través de socket web. La figura 10 muestra el modo de comunicación de un dispositivo IoT hacia la plataforma en la nube. .

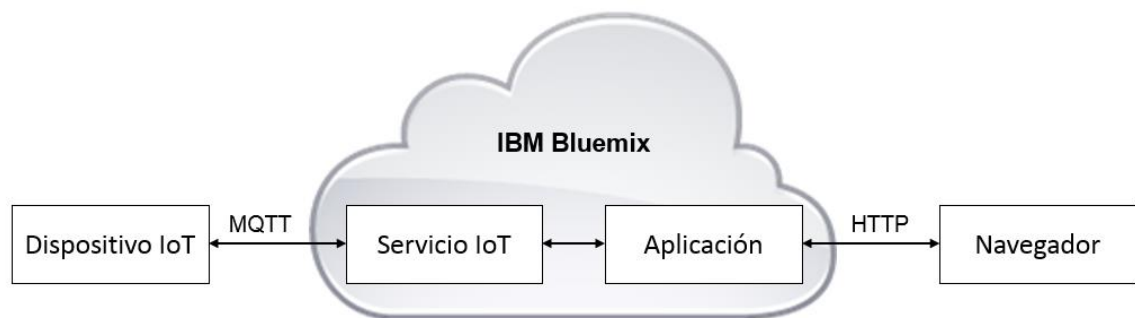


Figura 10 Comunicación Dispositivo IoT a IBM Bluemix

Fuente: Elaboración propia de acuerdo a datos mostrados en el esquema de IoT.

Lo siguiente es necesario para habilitar la comunicación entre el dispositivo IoT y la plataforma Bluemix IoT, y entre la plataforma Bluemix IoT y la aplicación IoT.

1. Iniciar sesión en la interfaz gráfica de usuario de la plataforma IBM Bluemix (por ejemplo, la consola de la plataforma).

2. Las opciones de autenticación del dispositivo (es decir, clave de acceso, token definido por el usuario, etc.) se pueden seleccionar en este paso.

3. Conectar el (los) dispositivo (s) con la plataforma IoT utilizando el protocolo MQTT Utilizar las opciones de acceso decididas en el paso 2.

4. Después de una conexión correcta, el dispositivo puede enviar los datos a la plataforma IoT conectada.

5. La plataforma IoT ahora puede recibir los datos y tomar medidas si los datos están en formato JSON.

6. Los datos generados desde el dispositivo pueden ser visualizados y analizados, o pueden ser enviados a la aplicación asociada para futuras acciones.

7. El dispositivo IoT enviará un mensaje al dispositivo en cuestión para realizar la acción en el comando.

8. Se pueden tener reglas predefinidas para tomar acción sobre los datos del dispositivo, por ejemplo, si un dispositivo está enviando los datos de temperatura, la regla predefinida (es decir, si sobrepasa a un límite umbral) puede enviar de nuevo la orden de control al dispositivo.

1.7.2. Visualización de Datos en IBM Watson IoT de Bluemix

Los datos recogidos de los dispositivos IoT se pueden visualizar utilizando la aplicación de visualización incorporada en Bluemix.

El servicio Internet de Cosas de Bluemix permite crear gráficos en tiempo real para visualizar los datos de los sensores.

El gráfico en tiempo real también se puede configurar para visualizar los datos del sensor disponibles en un gráfico de barras.

Junto con el registro de diagnóstico, los códigos de error y los registros de conexión. El servicio IoT en Bluemix también es información relacionada (un dispositivo de conexión, eventos recientes, la información del sensor y el tiempo que se recibe).

1.8. Node-RED

Node-RED es una herramienta de programación para conectar dispositivos de hardware, API y servicios en línea de maneras nuevas e interesantes.

Node-RED proporciona un editor de flujo basado en navegador (figura 11) que facilita el cableado de flujos utilizando la amplia gama de nodos en la paleta. Los flujos se pueden implementar en el tiempo de ejecución con un solo clic [12].

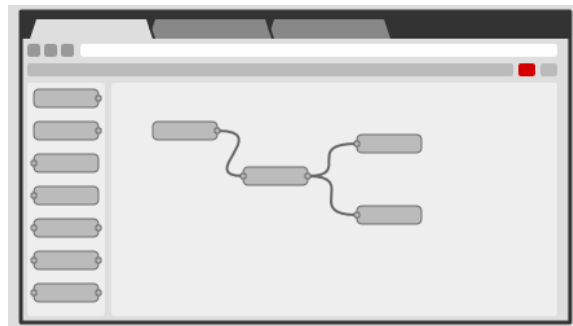


Figura 11 Node-RED [13]

<https://nodered.org/images/nr-image-1.png>

Las funciones de JavaScript se pueden crear dentro del editor usando un editor de texto enriquecido.

El tiempo de ejecución ligero se basa en Node.js, aprovechando al máximo su modelo basado en eventos y sin bloqueo. Esto lo hace ideal para ejecutar en el borde de la red en hardware de bajo costo como Raspberry Pi y en la nube.

Los flujos creados en Node-RED se almacenan usando JSON (figura 12), que se puede importar y exportar fácilmente para compartir con otros.

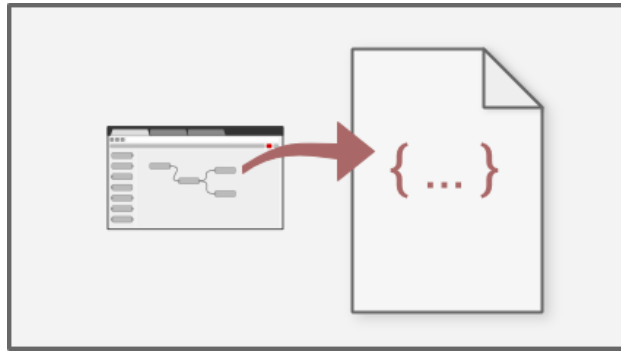


Figura 12 Node-RED, JSON [14]
<https://nodered.org/images/nr-image-3.png>

La figura 13 muestra algunos de los flujos disponibles en línea que permite compartir con la comunidad, así mismo es posible construir propios flujos para aplicaciones específicas.

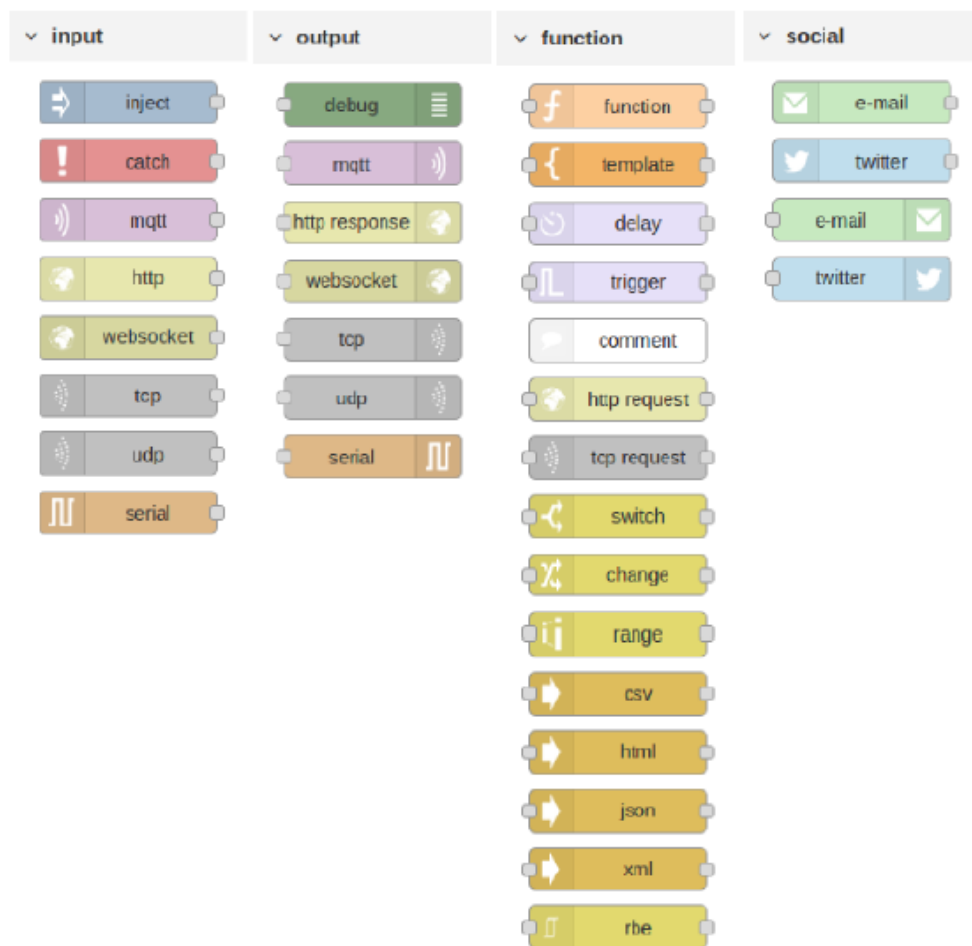


Figura 13 Nodos básicos de Node-RED
Fuente: <https://nodered.org>

1.9. Cloudbant NO-SQL Database

IBM Cloudbant NoSQL DB for Bluemix es una base de datos NoSQL como servicio (DBaaS). Está creada desde la base para escalar globalmente, ejecutarse sin parar y manejar una gran variedad de tipos de datos como JSON, texto completo y geoespacial. Cloudbant NoSQL DB es un almacén de datos operativos optimizado para manejar lecturas y escrituras simultáneas y proporcionar alta disponibilidad y durabilidad de datos.



Capítulo 2

Modulo Sensor de Energía basado en Internet de las Cosas (MSE-IoT)

Capítulo 2: Modulo Sensor de Energía basado en Internet de las Cosas (MSE-IoT)

En este trabajo se propone un sistema inteligente de energía que detecta los valores actuales en tiempo real, calcula la potencia instantánea y carga los valores en la nube utilizando el módulo Wi-Fi. El diagrama de bloques de este Sistema es el que se como se muestra en la figura 14.

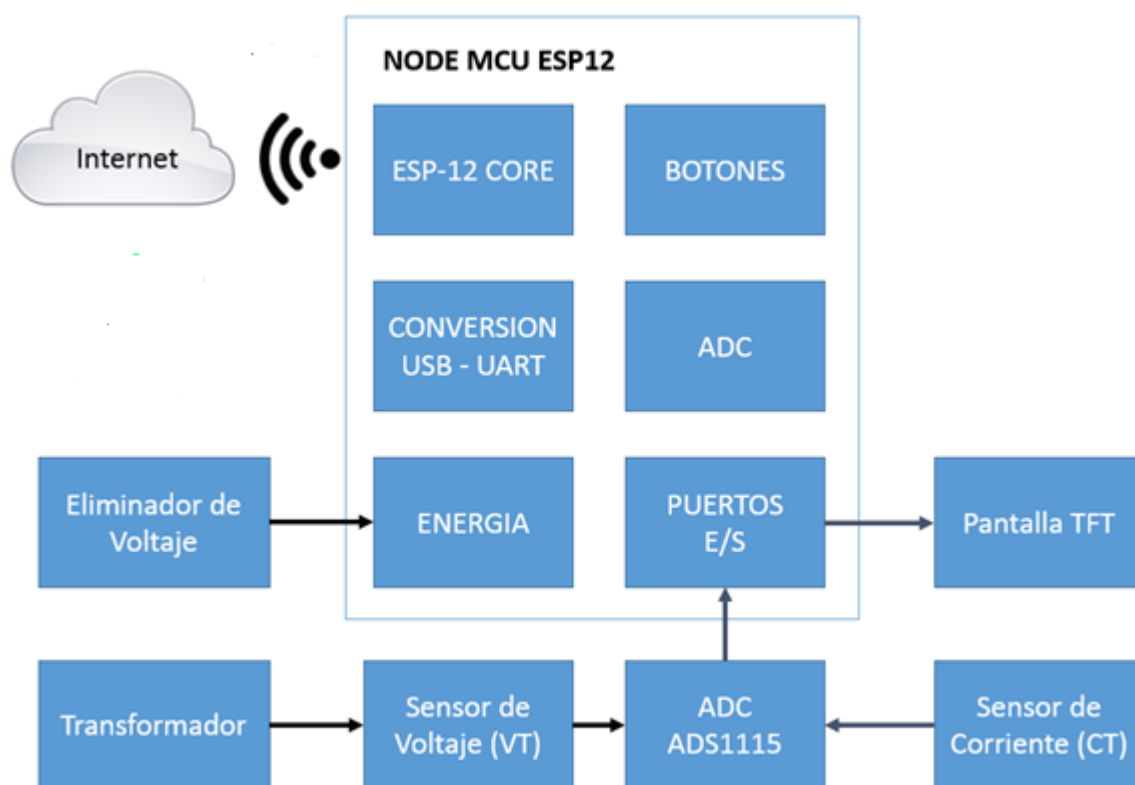


Figura 14 Esquema MSE-IoT

Fuente: Elaboración propia de acuerdo a datos mostrados en el diagrama a bloques

Los valores son detectados en tiempo real y se transmiten de forma inalámbrica a la plataforma IBM Watson IoT a través de Internet.

2.1 NodeMCU

MSE-IoT utiliza como procesador principal el NodeMCU que es un kit de desarrollo y firmware de código abierto, entre sus características principales se tiene:

- Open-source (código abierto)
- Interactivo
- Programable
- Bajo costo
- Simple
- Inteligente
- WI-FI habilitado
- Hardware IO similar a Arduino



Figura 15 Node-MCU [25]

Fuente: http://www.nodemcu.com/index_en.html

La figura 15 muestra una imagen de la placa de desarrollo Node-MCU

Especificaciones:

- Procesador ESP8266, de 32 bits a 80MHz o incluso 160MHz.
- 35K de IRAM.
- 80K de DRAM
- Tensión de trabajo 3.3V.
- Corriente de consumo 50mA aproximadamente aunque tiene picos de más de 300mA.
- Protocolo 802.11 b/g/n .
- Wifi Direct (P2P) Soft-AP.
- Lenguaje de programación “Lua”.
- Stack TCP/IP.
- Interface WIFI a 2.4 Ghz.
- Pulsador para Flash.
- Pulsador para Reset.

- Conector micro usb.
- Pins GPIO (input-Output) y PWM de 0-12. [15]

2.1.1. ESP-12

El módulo WiFi ESP-12E es un desarrollo de Ai-thinker Team incluye un procesador central ESP8266, en el módulo encapsula microprocesador Tensilica L106 de 32 bits y ultra bajo consumo de 32, así como modo corto de 16 bits. Soporte de velocidad de reloj 80 MHz, 160 MHz, admite RTOS, Wi-Fi integrado MAC / BB / RF / PA / LNA, antena de a bordo.

El módulo admite el estándar IEEE802.11 b / g / n, completa la pila de protocolos TCP / IP [16].

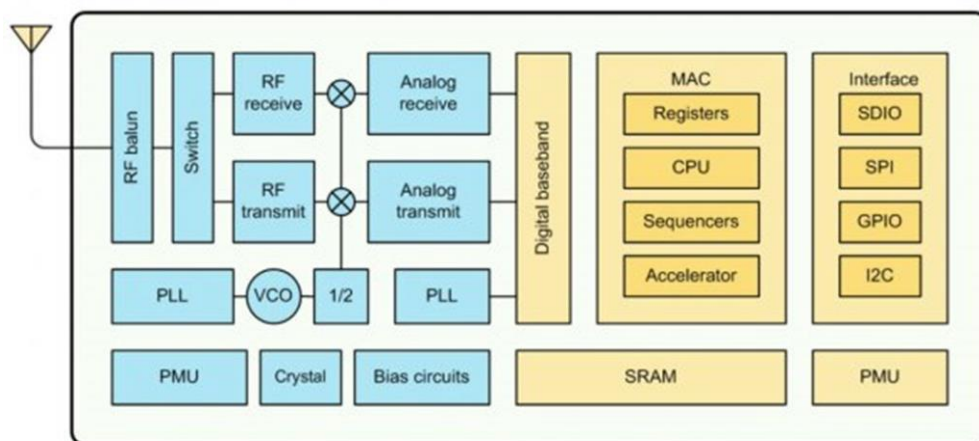


Figura 16 Diagrama a bloques del ESP8266 [17]

Fuente: <https://electrosome.com/wp-content/uploads/2016/11/ESP8266EX-Block-Diagram.png>

ESP8266 ofrece una solución de red WiFi completa e independiente; se puede usar para alojar la aplicación o para descargar funciones de red WiFi de otro procesador de aplicaciones. Cuando el ESP8266EX aloja una aplicación, se inicia directamente desde un flash externo. Contiene un caché integrado para mejorar el rendimiento del sistema en dichas aplicaciones, en la figura 16 se muestra el diagrama a bloques del ESP8266.

Alternativamente, puede servir como un adaptador de WiFi. El acceso inalámbrico a Internet se puede agregar a cualquier diseño basado en micro controlador con conectividad simple (interfaz SPI / SDIO o I2C / UART).

2.2. Sensor de Corriente SCT-013-030

El SCT-013-030 es un sensor de corriente no invasivo que se sujeta alrededor de la línea de suministro, puede medir hasta 30 amperes, permitiendo calcular la cantidad de corriente que pasa a través de él. Puede ser útil para construir un monitor de energía o para construir un dispositivo de protección de sobrecorriente para una carga de CA. La figura 17 muestra una imagen de SCT-013-030.



Figura 17 Sensor SCT-013-030 [18]

Fuente: https://www.hobbyist.co.nz/sites/default/files/ux_a11051400ux0258_ux_c.jpg

Resistencia de carga

Un TC con "salida de corriente" debe usarse con una resistencia de carga. La resistencia de carga completa o cierra el circuito secundario de TC. El valor de carga se elige para proporcionar un voltaje proporcional a la corriente secundaria. El valor de carga debe ser lo suficientemente bajo para evitar la saturación del núcleo de TC.

Aislamiento

El circuito secundario está aislado galvánicamente del circuito primario. (Es decir, no tiene contacto metálico)

En general, un TC nunca debe estar en circuito abierto una vez que está conectado a un conductor portador de corriente. Un TC es potencialmente peligroso si está en circuito abierto.

Si se abre el circuito con la corriente que fluye en el primario, el secundario del transformador intentará continuar impulsando la corriente hacia lo que efectivamente es una impedancia infinita. Esto producirá un voltaje alto y potencialmente peligroso a través del secundario

Algunos TC tienen protección incorporada. Algunos tienen diodos Zener protectores, como es el caso con el SCT-013-000 recomendado para su uso en este proyecto. Si el TC es del tipo 'salida de voltaje', tiene una resistencia de carga incorporada. Por lo tanto, no puede ser de circuito abierto.

2.2.1. Esquema de Montaje Sensor de Corriente

Para la interconexión de sensor de corriente SCT-013-030, hay que considerar que el sensor produce un voltaje de corriente alterna, por lo que es necesario la adaptación de este voltaje. Los procesadores solo pueden realizar mediciones de voltajes entre 0 y el voltaje de alimentación (V_{cc}).

Los voltajes positivos y negativos que entrega el sensor pueden ser medidos con convertidor analógico digital que hace uso de lecturas de modo diferencial tal es el caso del ADS1115.

De igual manera debe considerarse el voltaje pico que es mayor al voltaje promedio (V_{rms}). Esto con el fin de no superar el voltaje de alimentación del convertidos analógico digital y producir un daño en el.

$$V_{pico} = \sqrt{2} \cdot V_{RMS} = R \cdot I_{pico} = \sqrt{2} \cdot R \cdot I_{RMS}$$

$$V_{picoapico} = 2 \cdot V_{pico} = 2 \cdot \sqrt{2} \cdot V_{RMS} = R \cdot I_{picoapico} = 2 \cdot \sqrt{2} \cdot R \cdot I_{RMS}$$

En el caso de los sensores con salida de $\pm 1V$ RMS, la tensión de pico será $\pm 1.414V$, y la tensión pico a pico $2.828V$.

2.3. Sensor de Voltaje

Para la medición de Voltaje fue utilizado en módulo FZ0430 (mostrado en la figura 18) que permite medir tensiones de hasta 25V.



Figura 18 Sensor FZ0430 [19]

Fuente: <https://1.bp.blogspot.com/-YwLTtcWucQc/Wjgm5tpjKjI/AAAA>

En realidad el FZ0430 es un simple divisor de tensión con resistencias de 30kOhm y 7.5kOhm (figura 19), lo que supone que la tensión percibida tras el módulo sea de divida por un factor de 5 ($7.5/(30+7.5)$)

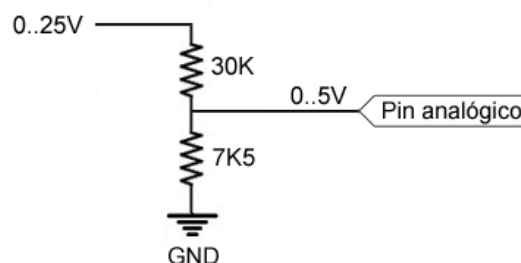


Figura 19 Circuito Eléctrico Sensor FZ0430

Fuente: <https://1.bp.blogspot.com>

Por tanto, la tensión máxima que se puede medir es 25V para un procesador de tensión de alimentación Vcc 5V, y 16.5V para un procesador de Vcc 3.3V. En caso de emplear una carga de 25V, la corriente que atraviesa el divisor es de 0.7mA, y las pérdidas del divisor 16.67mW.

2.4. Convertidor Analógico Digital ADS1115

El ADS1115 es un conversor analógico digital (ADC) externo que se conecta al NodeMCU que solo dispone de un ADC de 10 bit de resolución, y dado que también

es necesario la medición del voltaje es requerido de al menos de dos ADC para poder calcular la potencia real.

El ADS1115 proporciona 4 ADC de 16 bits, 15 para la medición y un último para el signo. El ADS1115 se conecta por I2C, Dispone de 4 direcciones, que se elige mediante la conexión del pin ADDRESS.

El interés de emplear un ADC como el ADS1115 es obtener una mayor precisión, liberando carga al procesador. Además, en ciertas configuraciones, es posible medir voltajes negativos.

El ADS1115 tiene dos modos de medición, single ended y diferencial. En el modo single ended se dispone de cuatro canales de 15 bits. En el modo diferencial se usa dos ADC para cada medición, por lo que el número de canales se reduce a 2, pero se tiene la ventaja de poder medir voltajes negativos y mayor inmunidad al ruido.

También dispone de un modo comparador en el que el ADS1115 genera una alerta por el pin ALERT cuando cualquiera de los canales supera un valor de umbral que fijamos por código.

El ADS1115 incorpora un PGA que permite ajustar la ganancia desde 6.144V a 0.256V. Esto permite obtener precisiones superiores cuando se mide voltajes inferiores a 5V. Independientemente del PGA elegido la máxima tensión que podemos medir será siempre la tensión de alimentación [20].

2.5. Diseño de Software

Para la implementación del software se hizo uso del IDE de Arduino como lenguaje principal C++, así mismo de las siguientes librerías:

- **ESP8266WiFi.h**, para el manejo del NodeMCU y propiedades WiFi
- **PubSubClient.h**, necesaria para manejo de protocolo MQTT y publicación de datos en la nube.
- **SPI.h**, para la conexión del ADC por medio del protocolo SPI
- **Adafruit_GFX.h**, para el manejo de gráficos en el Display
- **Adafruit_ILI9341.h**, para la operación del Display TFT

- **Wire.h**, conexión I2C
- **Adafruit_ADS1015.h**, operación del convertidor Analógico Digital

La figura 20 muestra el diagrama a bloques general de la operación del módulo MSE-IoT.

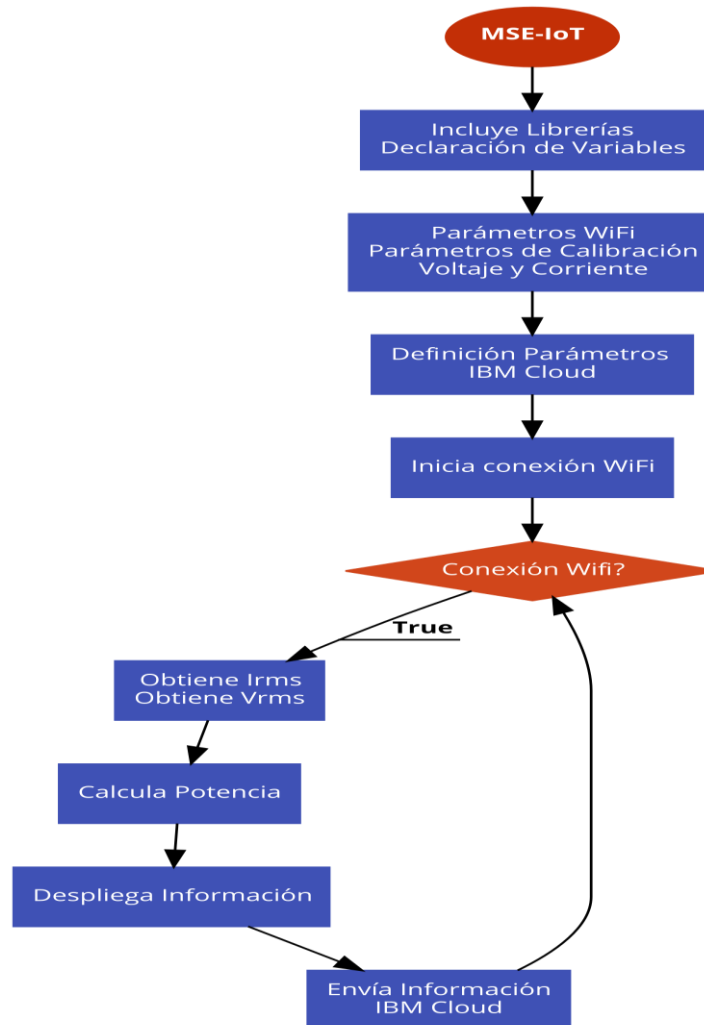


Figura 20 Diagrama a Bloques MSE-IoT

Fuente: Elaboración propia de acuerdo al diagrama general de operación

Después de la inclusión de las librerías se definen los parámetros de conexión a la red inalámbrica, así como los parámetros que identifican al módulo que será conectado a la nube de IBM Bluemix. Si existe conexión a Internet se obtiene el valor de Irms y Vrms, mediante el cálculo de las muestra tomadas en los

sensores de corriente y voltaje, con estos datos se calcula la potencia, estos tres datos son desplegados en el display del módulo. Adicionalmente son enviados a la nube de IBM Bluemix.

La figura 21 muestra el diagrama funcional, destacan funciones principales, función para obtener lectura y calcular I_{rms} , así como para el cálculo de V_{rms} , y determinar la Potencia Real. Obteniendo los datos correspondientes, se verifica si el cliente tiene conexión hacia la plataforma, en caso afirmativo envía la información y esta queda registrada.

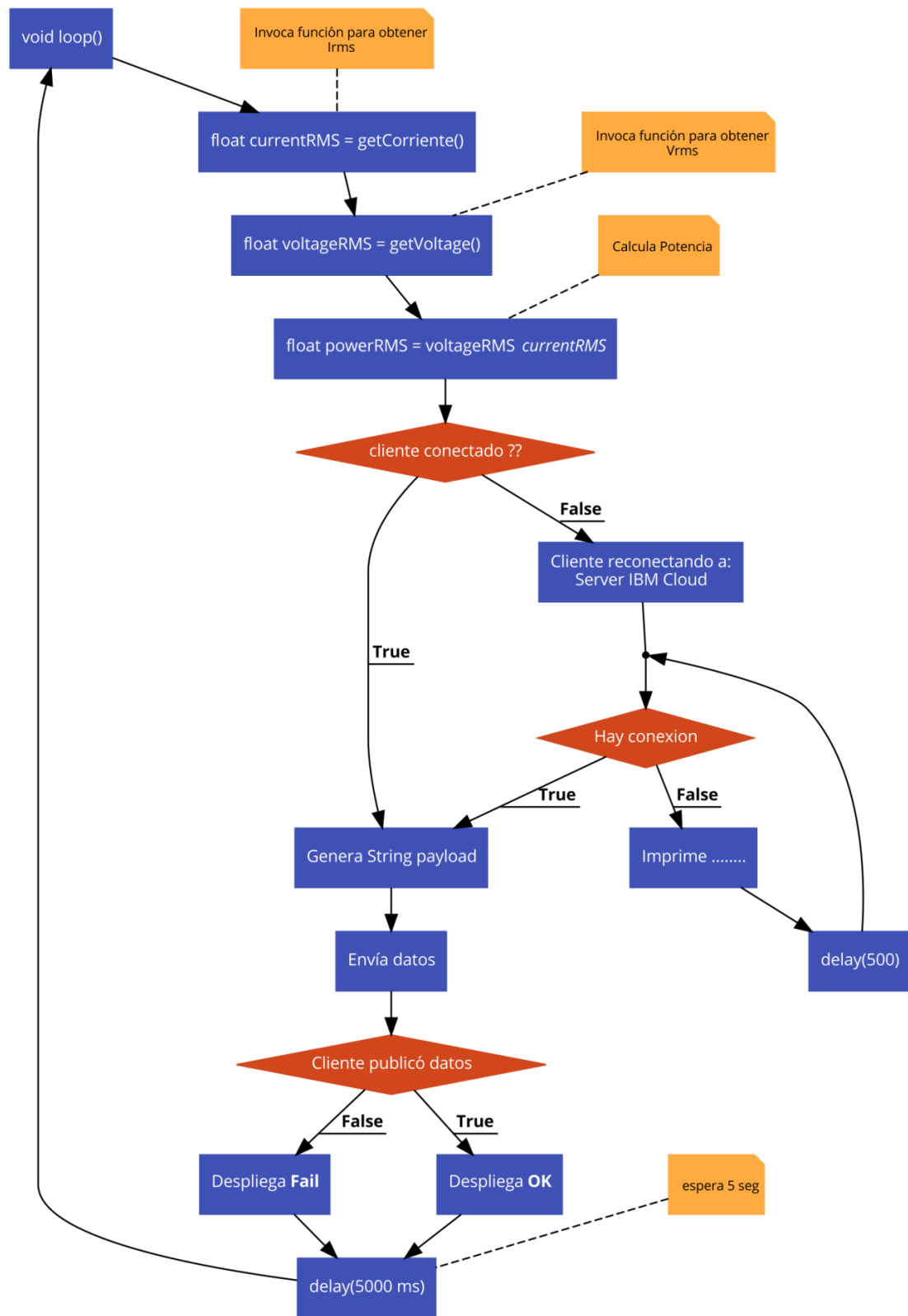


Figura 21 Diagrama Funcional

Fuente: Elaboración propia de acuerdo con la funcionalidad del sistema

Como ejemplo de obtencion de datos, en la figura 22 se muestra la funcion `getCorriente()`, inicialmente se declaran variables, se determina un tiempo de muestra de 1 seg., se toma lectura diferencial en el el ADC. De las diferentes muestras obtedidas se hace el calculo de valor eficaz de la corriente Irms.

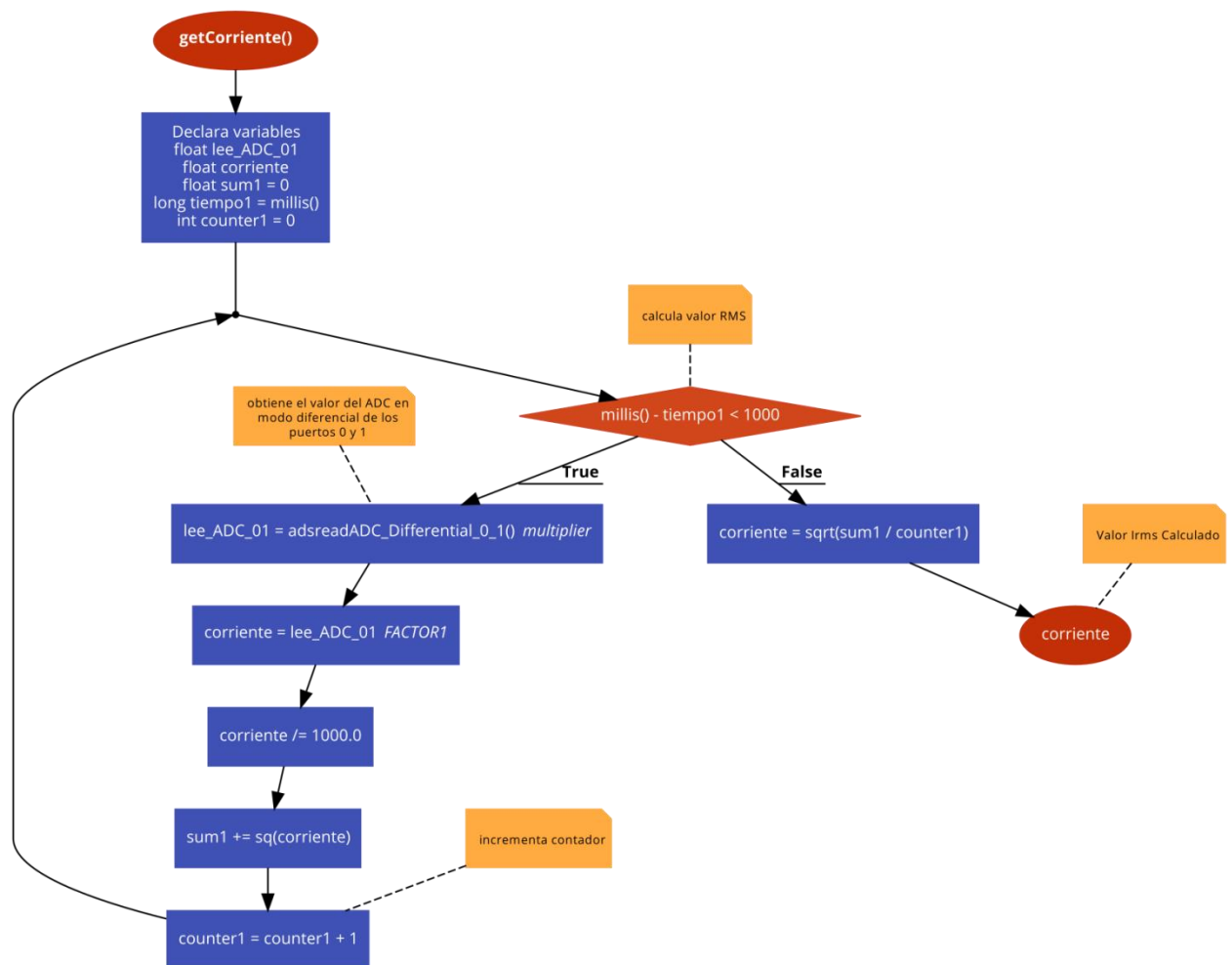


Figura 22 Función para obtener Irms

Fuente: Elabracion propia, de acuerdo a la función para la obtención de la corriente RMS

En la figura 23 se muestra el codigo de la funcion `getCorriente()`. Figura 24 parametos de conexión hacia la plataforma en la nube. Figura 25 payload que ser enviado a la plataforma, conformado por el nombre del dispositivo, Irms, Vrms y potencia

```

float getCorriente() {

    float lee_ADC_01;
    float corriente;
    float sum1 = 0;
    long tiempo1 = millis();
    int counter1 = 0;

    // calcula valor RMS

    while (millis() - tiempo1 < 1000) {

        lee_ADC_01 = ads.readADC_Differential_0_1() * multiplier;
        corriente = lee_ADC_01 * FACTOR1;
        corriente /= 1000.0;
        sum1 += sq(corriente);
        counter1 = counter1 + 1;
    }

    corriente = sqrt(sum1 / counter1);
    return(corriente);
}

```

Figura 23 Código función Irms

Fuente: Captura de pantalla de código de la función Irms

```

// Parametors de conexioci a IBM Cloud
#define ORG "1q15i6"
#define DEVICE_TYPE "NodeMCU"
#define DEVICE_ID "5CCF7FD6567C"
#define TOKEN "evmn0cIFlC(bcPQT1"

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char topic[] = "iot-2/evt/status/fmt/json";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;

```

Figura 24 Parámetros Conexión a IBM Bluemix

Fuente: Captura de pantalla de código conexión a plataforma

```
// Deficion del payload

String payload = "{\"d\":{\"Name\":\"5CCF7FD6567C\"";
payload += "\",\"Irms\":";
payload += currentRMS;
payload += "\",\"Vrms\":";
payload += voltageRMS;
payload += "\",\"Potencia\":";
payload += powerRMS;
payload += "}}";
```

Figura 25 Formación Payload

Fuente: Captura de pantalla de código

Haciendo uso de Node-RED, se configuraron los bloques para lograr las diferentes funcionalidades como, almacenamiento a base de datos, configuración de alertas, envío de mensajes a Twitter y Correo Electrónico, así como de envío de información a plataforma Emoncms, diagrama mostrado en la figura 26.

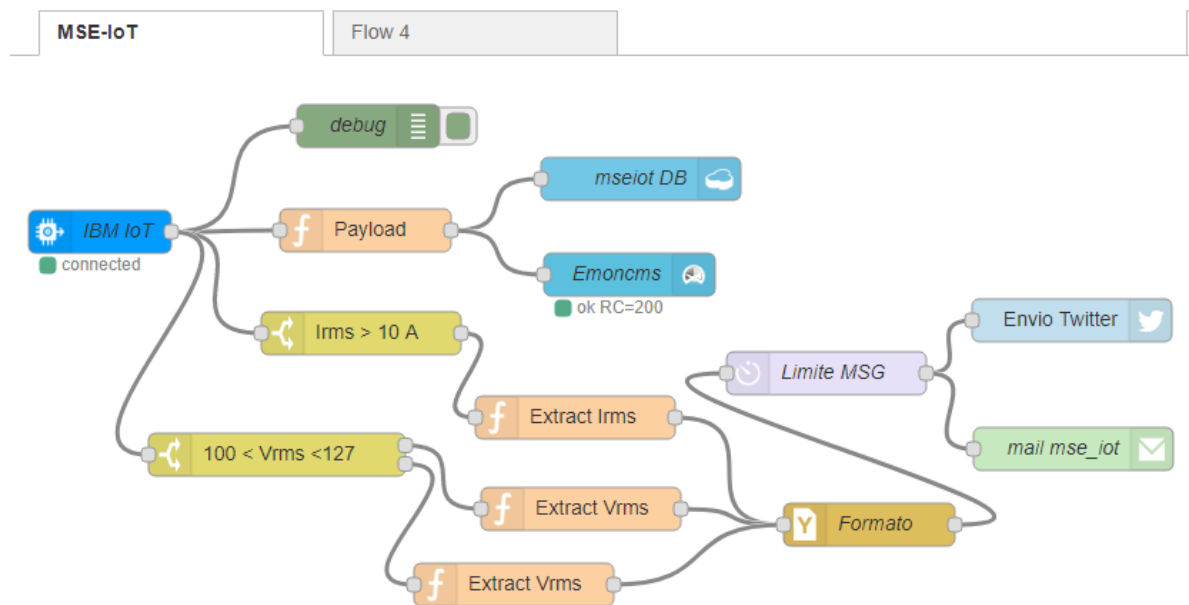


Figura 26 Diagrama Funcional IBM Bluemix [21]

Fuente: Elaboracion propia de acuerdo a captura de pantalla de aplicacion en la nube



Capítulo 3

Pruebas

Capítulo 3: Pruebas

Para las pruebas y registro de mediciones se colocó un sensor de corriente a la toma principal de la energía eléctrica de una casa habitación como se muestra en la figura 27.



Figura 27 Conexión del sensor de corriente a la toma principal

Fuente: Elaboracion propia de acuerdo a fotografía tomada del sensor de corriente en la toma principal

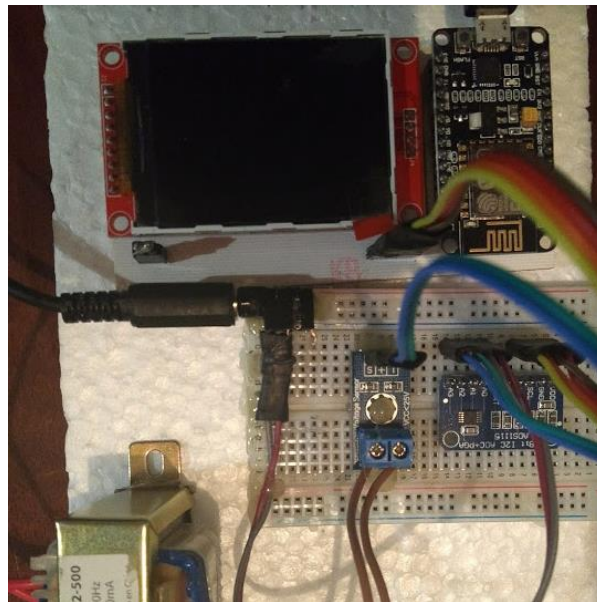


Figura 28 Prototipo MSE-IoT

Fuente: Elaboracion propia de acuerdo a fotografía del prototipo

Así como un sensor de voltaje para con la corriente y el voltaje medidos poder determinar el valor de la potencia real.

3.1. Medición de Corriente

Haciendo uso del sensor SCT-013-30 con salida de tensión $\pm 1V$ RMS y resistencia burden interna, junto con un ADC como el ADS1115 en modo diferencial, La figura 29 muestra el modo de conexión del sensor de corriente y el ADC, fueron utilizados los puertos A0 y A1 los cuales se conecta al secundario del transformador de corriente.

Ajustando la ganancia del ADS1115 a 2.048V estará dentro del rango de $\pm 1.414V$. En el caso de un sensor de 30A tendremos una precisión de 1.87mA.

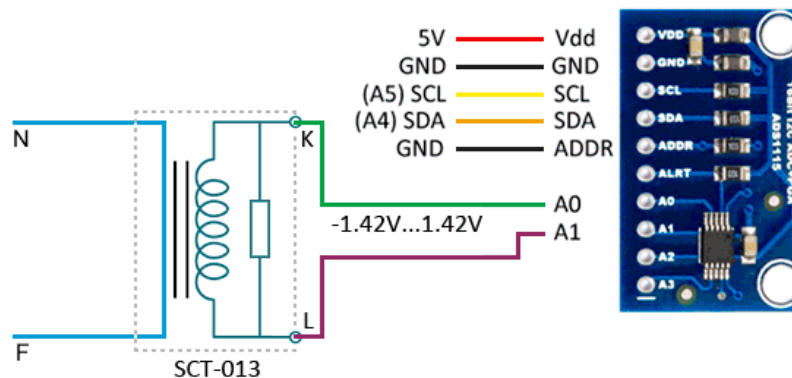


Figura 29 Conexión para medición de corriente

Fuente: Elaboracion propia de acuerdo al diagrama de conexión en el ADC para la medición de corriente

3.2. Medición de Voltaje

Para la medición de voltaje se utilizó el modulo sensor FZ0430 y fue necesario la inclusión de un transformador para reducir la toma de 127 VAC a 12 VAC. La figura 30 muestra el esquema de conexión del sensor de voltaje y el ADC, también en este caso se utilizan dos puertos de ADC para poder operar en modo diferencial.

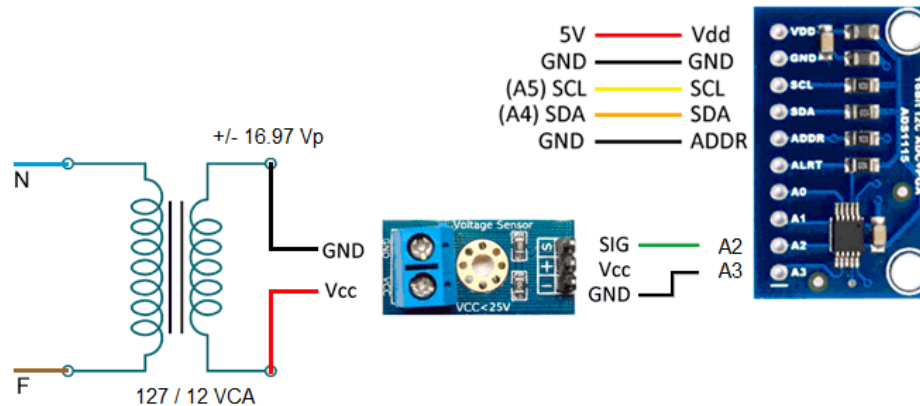


Figura 30 Conexión para medición de voltaje

Fuente: Elaboración propia de acuerdo al diagrama de conexión en el ADC para la medición de voltaje

3.3. Selección del proveedor de la nube

Existen diferentes tipos de proveedores de servicios en la nube que van desde el código abierto hasta los privados. Los proveedores de la nube que se consideraron durante el proyecto fueron Microsoft Azure, AWS e IBM Bluemix, siendo este último el utilizado para el proyecto.

Se eligió IBM Bluemix porque admite proyectos de código abierto para Internet de las cosas. Por ejemplo Node-RED, que se utiliza para Internet de las cosas.

3.3.1. IBM Bluemix (IBM Cloud)

IBM Bluemix es la solución de IBM para la nube. Bluemix es una plataforma como una solución de servicio, así como infraestructura como servicio. En PaaS (Platform as a Service), Bluemix es posible desarrollar, construir, probar, implementar, ejecutar y administrar la aplicación en la nube. En IaaS (Infrastructure as a Service), el hardware, el almacenamiento y la red se pueden administrar en la nube.

Bluemix admite la construcción de la aplicación utilizando una infraestructura privada. Esto es posible solo cuando la capa de virtualización utilizada es Bluemix.

Existen varios tipos de capas de virtualización que se mantienen en la parte superior de la infraestructura de Bluemix. Bluemix los admite y hace que las aplicaciones se puedan implementar. Estos incluyen máquinas virtuales OpenStack y contenedores Docker que son máquinas y Cloud Foundry.

Cloud Foundry se utilizó en este proyecto. Cloud Foundry es una plataforma como servicio. Cloud foundry es un software de código abierto que permite a los desarrolladores codificar en múltiples idiomas como Java, Node.js Go, PHP y Ruby

3.4. Crear la cuenta en Bluemix

La cuenta de Bluemix puede ser almacenada en el sitio web de Bluemix www.ibm.com/Bluemix. Después de crear la cuenta en Bluemix, el siguiente paso fue crear la aplicación a través de la plantilla de Bluemix. En general, una plantilla repetitiva es una plantilla que incluye una aplicación, su entorno de tiempo de ejecución asociado y servicios predefinidos para un dominio.

3.5. Internet de las cosas de IBM Bluemix

En el catálogo, se eligió el inicio de Internet of Things porque la aplicación se basa en Internet of Things. Uno de los servicios en el estándar fue llamado base de datos Cloudant NoSQL que es adecuado para la aplicación en el almacenamiento de los datos en tiempo real de los sensores, en la figura 31 se aprecia la instancia de base de datos creada.



Figura 31 Cuenta Cloudant NoSQL

Fuente: Captura de pantalla de la cuenta Cloudant NoSQL

La plataforma de Internet de las cosas, que es el centro de todas las cosas en IBM IoT, fue uno de los servicios que se incluyeron en la repetición. Es donde se puede hacer toda la configuración y administración de todos los dispositivos conectados. En Internet of Things, la aplicación puede acceder a los datos en vivo e históricos.

Otro servicio que estaba disponible en la plantilla era SDK para Node.js que se utiliza para compilar la aplicación de JavaScript del lado del servidor.

3.6. Creando una aplicación

El nombre de la aplicación creada fue MSE-IoT. El nombre de host fue elegido automáticamente por Bluemix después de completar el nombre de la aplicación (que luego puede cambiarse). La Id. De organización también se configura automáticamente cuando se crea la aplicación. La identificación de la organización para este proyecto fue **1q15i6**. El tipo de organización era Bluemix-free y fue elegido porque ofrece varios servicios en comparación con otros. La ubicación geográfica elegida fue US-SOUTH, debido a la cantidad de servicios disponibles en la región

en comparación con otras regiones. Tales parámetros son mostrados en la figura 32.

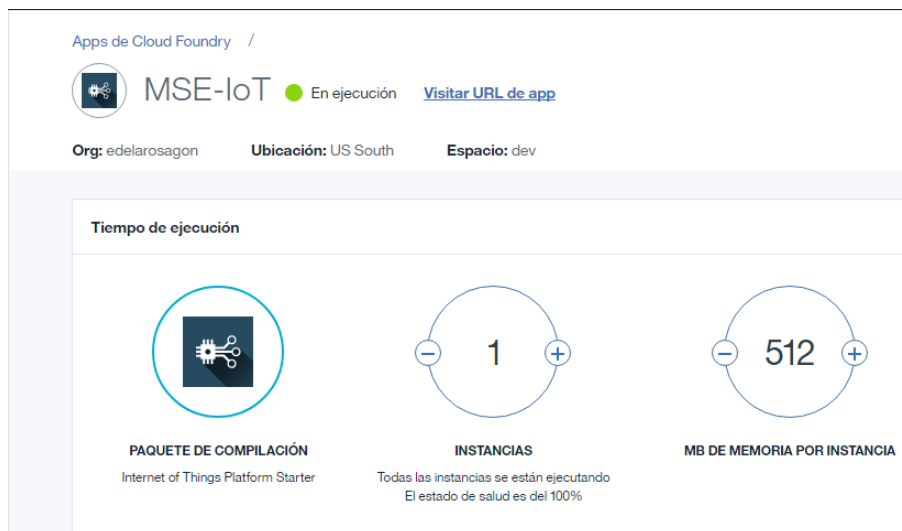


Figura 32 Aplicación MSE-IoT

Fuente: Captura de pantalla de la aplicación MSE-IoT en la plataforma de IBM Bluemix

La aplicación está construida usando Node-RED. Node-RED es un potente editor visual de IBM utilizado para crear aplicaciones de IoT. Node-RED está construido sobre Node.JS. A continuación muestra la aplicación lista para la edición Node-RED, figura 33.

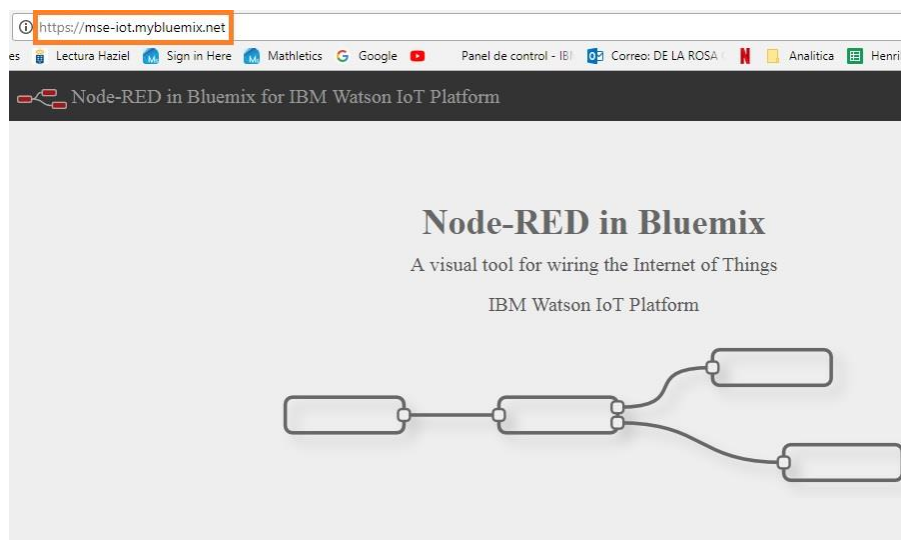


Figura 33 Aplicación en Node-RED

Fuente: Captura de pantalla de la aplicación en Node-RED

3.7. Registro del dispositivo

El dispositivo se agrega a IBM Watson IoT después de que se haya creado la aplicación. Al crear el tipo de dispositivo, hay dos opciones: tipo de dispositivo y tipo de puerta de enlace. Esta aplicación usa tipo de dispositivo. El siguiente paso es proporcionar el nombre del dispositivo (para esta aplicación fue NodeMCU) y proporcionar una descripción del dispositivo.

Cuando se haya creado el dispositivo, el siguiente paso es agregar el dispositivo proporcionando la ID del dispositivo que es la dirección Mac de la NodeMCU.

El siguiente paso es proporcionar una contraseña que sea un token entre el dispositivo y Bluemix por razones de seguridad y para una conexión segura. Después de proporcionar la información del dispositivo, el siguiente paso es copiar la información del dispositivo al NodeMCU en la carpeta donde se encuentra la aplicación. La información del dispositivo se copia en un archivo llamado configure.properties. La Figura 34 a continuación muestra la información del dispositivo después de haber sido creado y agregado a IBM Watson IoT.

Información de conexión	
ID de dispositivo	18FE34D42891
Tipo de dispositivo	NodeMCU
Fecha adición	jueves, 27 de julio de 2017
Añadido por	edelarosagon@gmail.com
Estado de conexión	Registrada Renovar

Figura 34 Conexión del Dispositivo

Fuente: Captura de pantalla durante el registro del dispositivo



Capítulo 4

Resultados

Capítulo 4: Resultados

En este apartado se presentan los resultados de las pruebas realizadas, en primer lugar se muestra el funcionamiento del módulo MSE-IoT con los resultados desplegados en pantalla.

Se puede apreciar en la figura 35, los tres principales parámetros medidos, Irms, Vrms y Potencia, así como la validación de que la información de estos parámetros fue enviada a la plataforma de IBM Bluemix, en el caso se falla por no conexión se despliega el mensaje de envió de datos fallido.



Figura 35 Datos en Pantalla

Fuente: Elaboracion propia, fotografía del sistema en operación

Con la conexión del dispositivo a la plataforma de IBM Bluemix, se crearon paneles para la visualización en tiempo real de los valores de Irms, Vrms y Potencia, se obtuvo grafica de consumo de Irms observándose las variaciones del nivel cuando se conectaban diferentes equipos eléctricos (lavadora, microondas, etc.) a la toma eléctrica, en la figura 36 se muestran los resultados de estas pruebas.

Con el registro en tiempo real y durante varios días, en la figura 37 se muestra el monitoreo a la base de datos donde fueron almacenados los documentos en formato JSON que contiene los valores de medición así como la identificación del dispositivo y la fecha y hora de registro (figura 38).

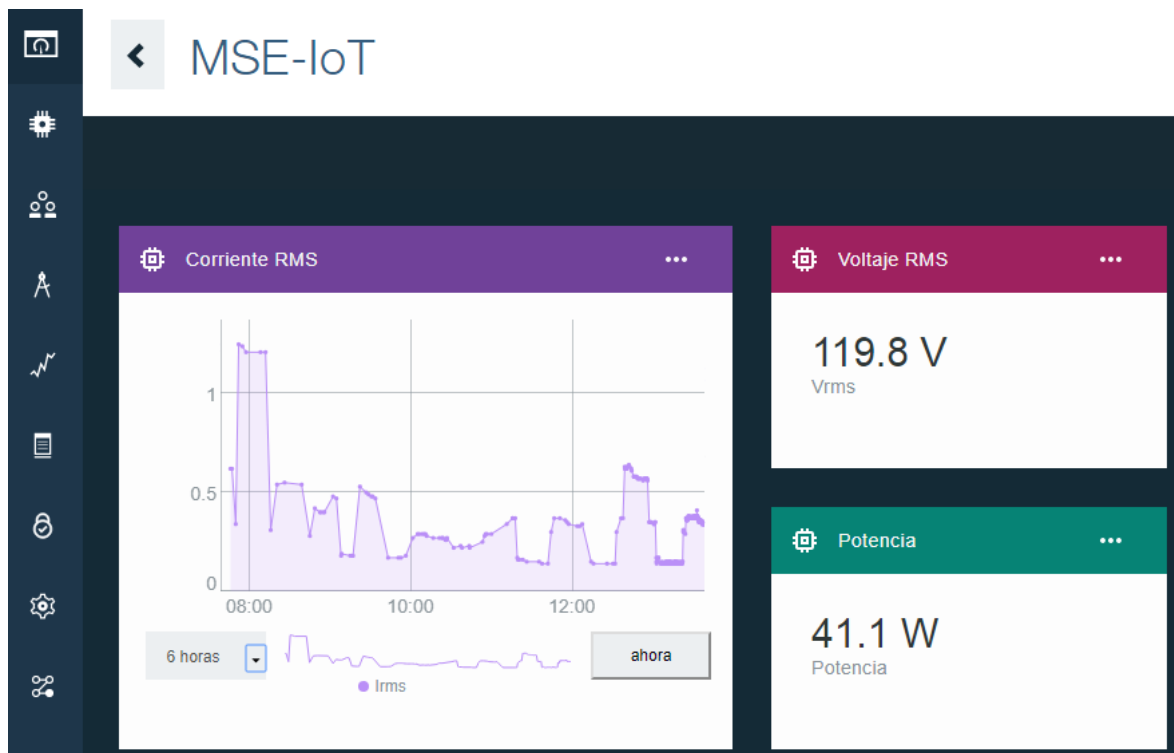


Figura 36 Visualización de Datos en IBM Bluemix

Fuente: Captura de pantalla para la visualización de datos en la plataforma IBM Bluemix

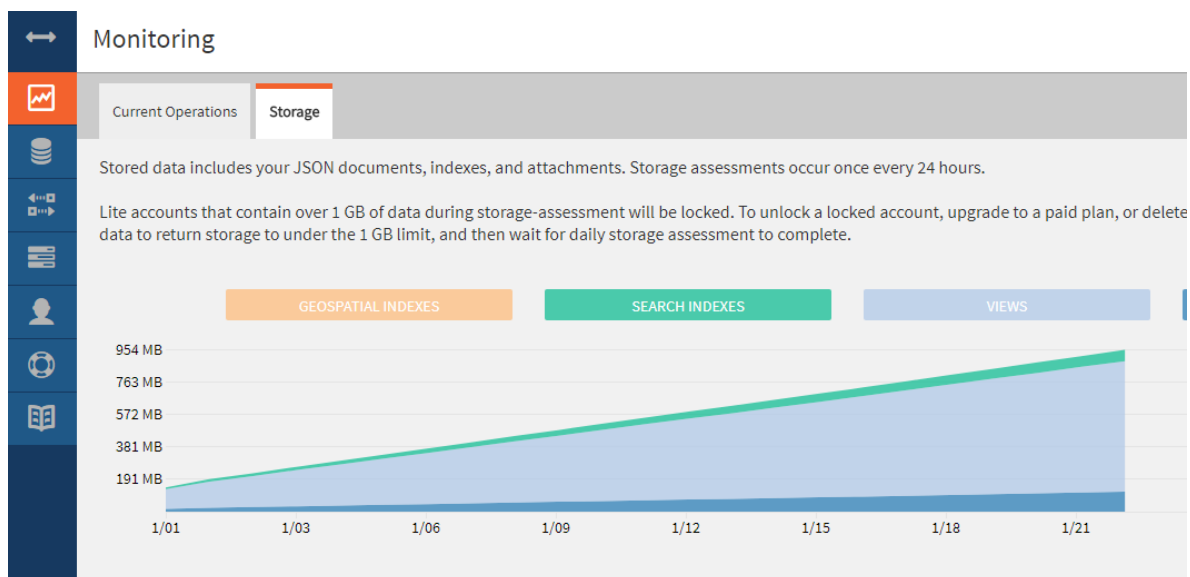


Figura 37 Registro en Base de Datos

Fuente: Captura de pantalla del registro de eventos en base de datos

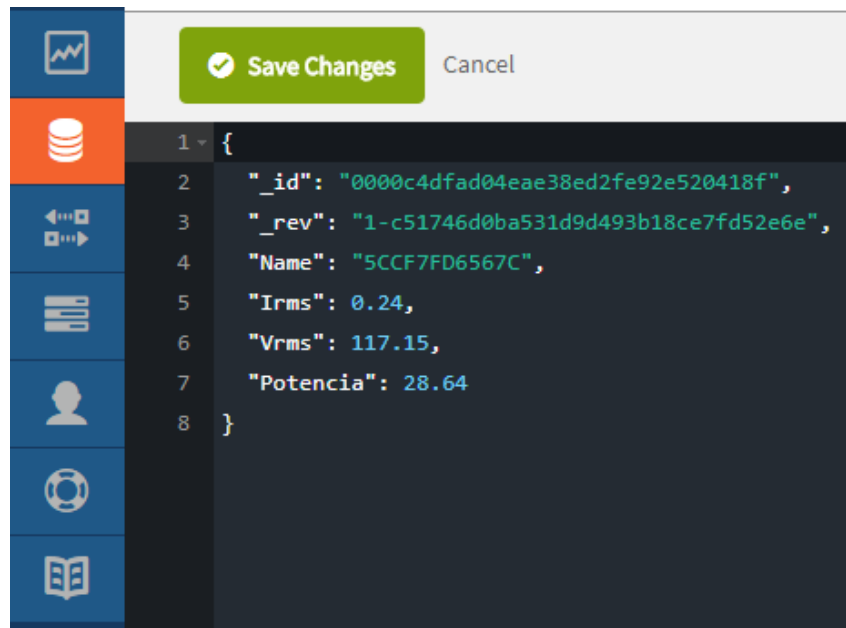


Figura 38 Formato de Información en JSON

Fuente: Captura de pantalla del tipo de información en formato JSON

También fue posible el reenvío de información desde IBM-Bluemix hacia la plataforma Emoncms que es una aplicación web de código abierto para procesar, registrar y visualizar energía, temperatura y otros datos ambientales, es parte del proyecto OpenEnergyMonitor, la figura 39 y 40 muestra los datos obtenidos, donde se puede tener inferencias por parte del usuario a partir de los datos visuales mostrados se detecta el funcionamiento de un aparato eléctrico con comportamiento cíclico de carga, así patrones de comportamiento del consumo energético, cuando se encuentran personas en casa, y el comportamiento los fines de semana

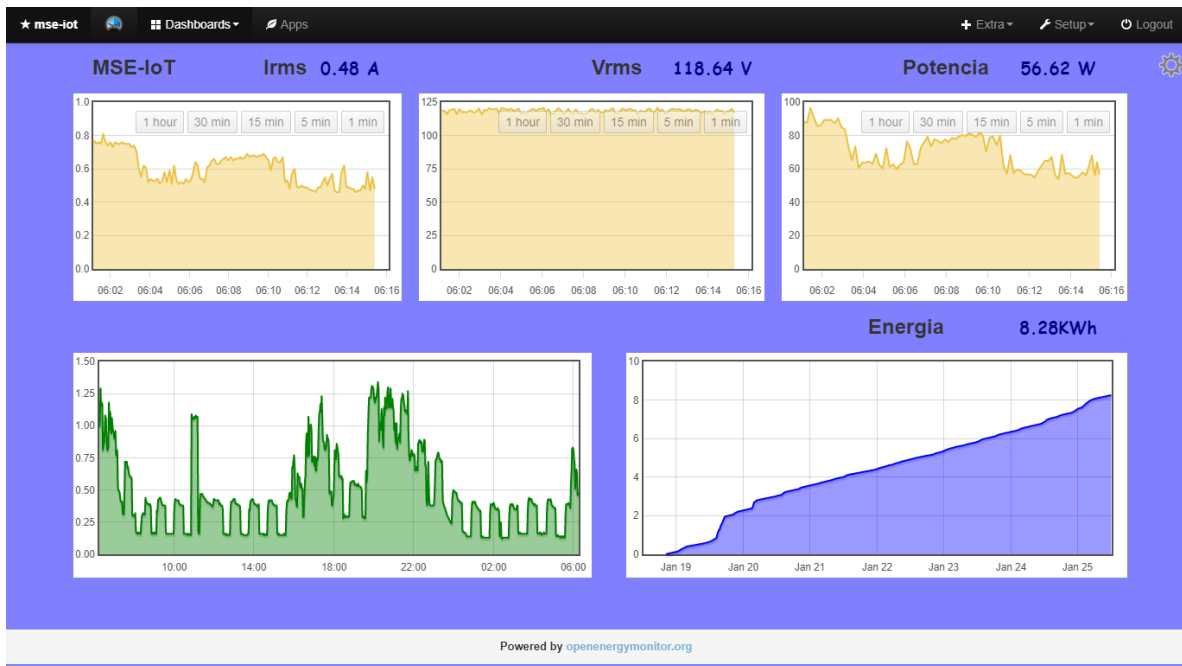


Figura 39 Información en plataforma Emoncms

Fuente: Captura de Pantalla de la visualización de la información en la plataforma Emoncms

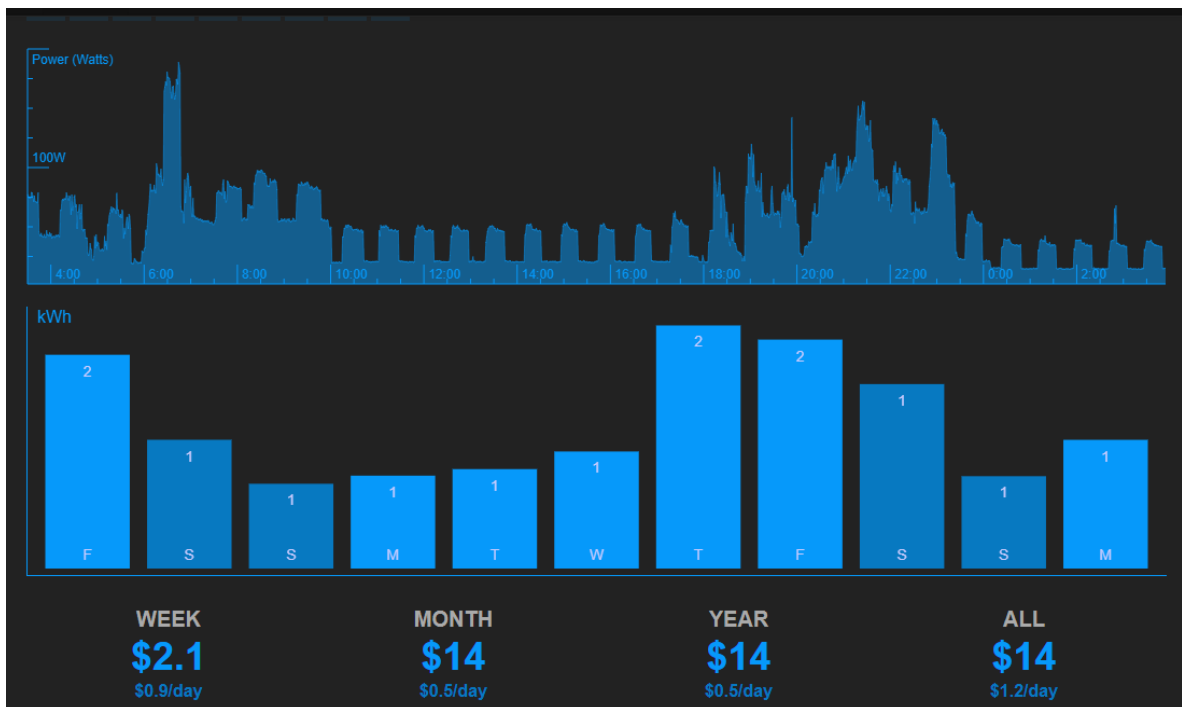


Figura 40 Consumo Energético

Fuente: Captura de pantalla del cosumo energético en plataforma Emoncm

Se configuro el envio de alertas de hacia correo electronico y Twitter estableciendo niveles de bajo voltaje y alta corriente, en la figura 41 se aprecia el correo recibido cuando el voltaje de alimentacion cae a cero, cabe hacer mencion que esta caida en delieverada con el fin validar la recepcion de alertas.

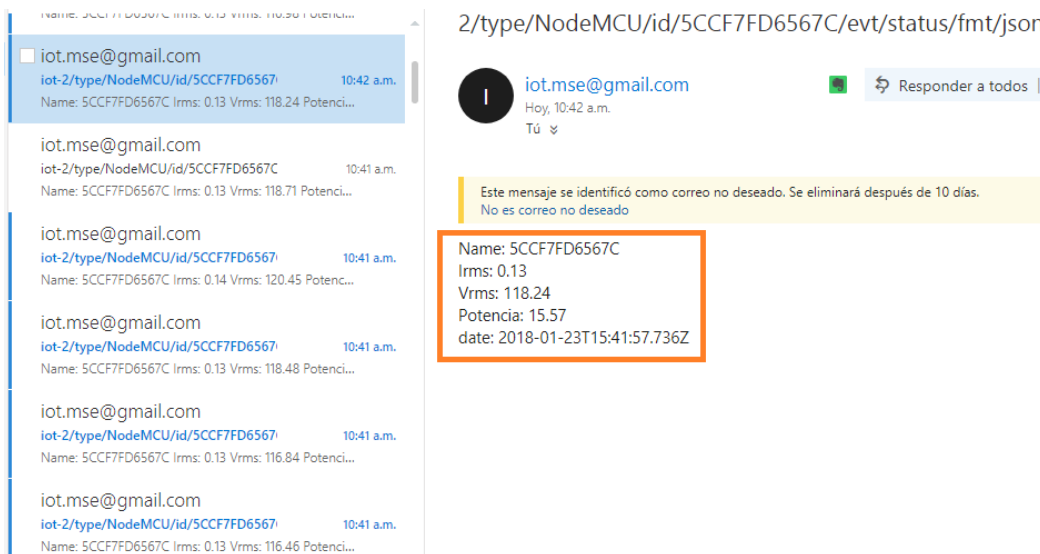


Figura 41 Envío de alertas a correo electrónico

Fuente: Captura de pantalla de aplicación de correo electrónico

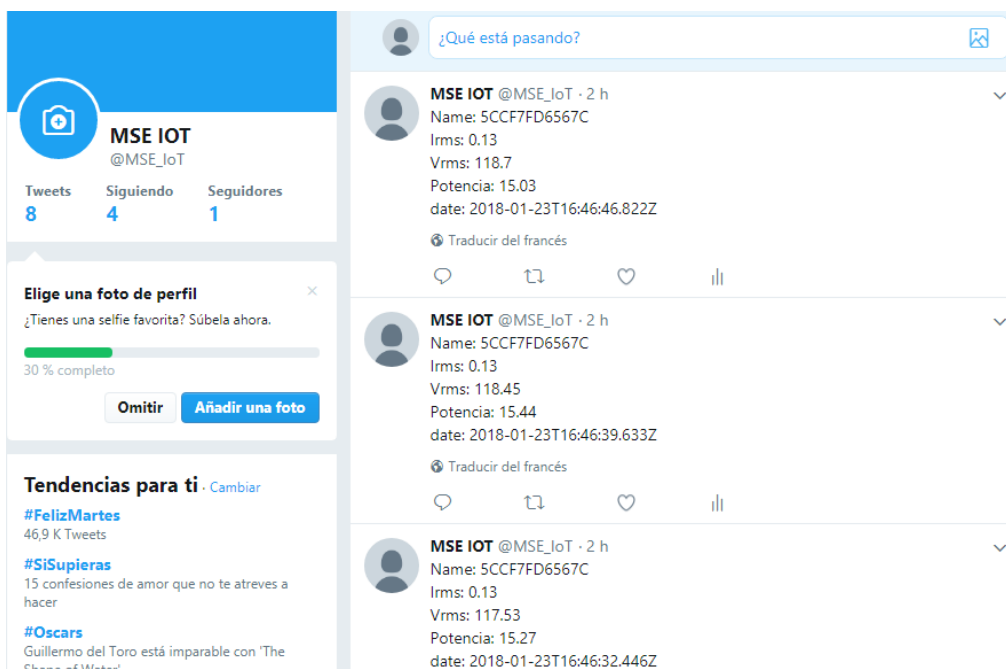


Figura 42 Envío de alertas a Twitter

Fuente: Captura de pantalla de aplicación Twitter recibiendo alertas



Conclusiones

Conclusiones

Se pudo hacer uso de una plataforma de IoT en la nube, para el monitoreo y análisis de energía, sin que representara un costo económico, pudiéndose tener hasta un 1 GB de almacenamiento en base de datos, con un registro de hasta 20 documentos por seg.

Se incorpora plataformas Cloud, las soluciones ofrecidas en el mercado ofrecen sistemas web propietarios, así mismo fue posible incorporar a otras plataformas de monitoreo energético (Emoncms) y servicios tales como correo electrónico y redes sociales (Twitter).

Las plataformas en la nube presenta ventajas competitivas, incorpora servicios para Análisis de Datos, Aprendizaje Automático, así como incorporación a otros servicios (correo electrónico, redes sociales, big data...)

Se tienen inferencias por parte del usuario a partir de los datos visuales mostrados y así como de alertas.

Se detecta el funcionamiento de aparatos eléctricos con comportamiento cíclico de carga.

Se detectan patrones de comportamiento del consumo energético, cuando se encuentran personas en casa, así como el comportamiento los fines de semana.

Bibliografía

- [1] T. T. P. I. Nattachart Tamkittikhun, «AC Power Meter Design for Home Electrical Appliances».
- [2] L. LLAMAS, «SENSOR DE CORRIENTE ELÉCTRICA NO INVASIVO CON ARDUINO Y SCT-013,» 24 Enero 2017. [En línea]. Available: <https://www.luisllamas.es/arduino-sensor-corriente-sct-013/>. [Último acceso: Junio 2017].
- [3] Naylamp Mechatronics SAC, «Naylamp Mechatronics,» Abril 2018. [En línea]. Available: <http://www.naylampmechatronics.com/img/cms/Blog/Tutorial%20SCT-013/sensores%20CT.jpg>.
- [4] P. Waher, Learning Internet of Things, Packt Publishing, 2015.
- [5] varios, The Internet of Things: Connecting Objects to the Web, Wiley-ISTE, 2010.
- [6] C. B. a. A. Kervefors, «Internet of Things Exploring and Securing a Future,» 2015.
- [7] A. V. R. E. Toby Velte, Cloud Computing, A Practical Approach, McGraw-Hill Osborne Media, 2009.
- [8] D. B. O. E. Olivier Hersent, The Internet of Things: Key Applications and Protocols, Wiley, 2012.
- [9] S. Sankaranarayanan, Learning IBM Bluemix, Packt Publishing, 2016.
- [10] R. Gravelle, «IoT Development Platforms: IBM Watson IoT Overview,» [En línea]. Available: https://www.codeguru.com/imagesvr_ce/4489/Watson2.jpg. [Último acceso: 2018].
- [11] IBM, «IBM Cloud Docs,» [En línea]. Available: <https://console.bluemix.net/docs/>. [Último acceso: Noviembre 2016].
- [12] Node-Red, «Node-Red,» [En línea]. Available: <https://nodered.org/>. [Último acceso: Julio 2017].
- [13] Node-RED, «Node-RED,» [En línea]. Available: <https://nodered.org/images/nr-image-1.png>. [Último acceso: 2008].
- [14] Node-RED, «Node-RED,» [En línea]. Available: <https://nodered.org/images/nr-image-3.png>. [Último acceso: 2018].
- [15] N. Team, «NodeMcu Connect Things EASY,» [En línea]. Available: http://www.nodemcu.com/index_en.html. [Último acceso: Mayo 2017].

- [16 M. Schwartz, ESP8266 Internet of Things Cookbook, Packt Publishing, 2017.
]
- [17 Espressif, [En línea]. Available: <https://electrosome.com/wp-content/uploads/2016/11/ESP8266EX-Block-Diagram.png>. [Último acceso: 2018].
- [18 [En línea]. Available:
] https://www.hobbyist.co.nz/sites/default/files/ux_a11051400ux0258_ux_c.jpg. [Último acceso: 2018].
- [19 [En línea]. Available: <https://1.bp.blogspot.com/-YwLTtcWucQc/Wjgm5tpjKjI/AAAA>.
]
- [20 L. LLAMAS, «ENTRADA ANALÓGICA DE 16 BITS CON ARDUINO Y ADC ADS1115,» 14
] Noviembre 2016. [En línea]. Available: <https://www.luisllamas.es/entrada-analogica-adc-de-16-bits-con-arduino-y-ads1115/>. [Último acceso: 2017].
- [21 E. d. I. Rosa, «mse-iot.mybluemix.net,» 2018. [En línea]. Available: <https://mse-iot.mybluemix.net/red/#flow/273d4f64.5b428>.
]
- [22 NodeMCU, 2018. [En línea]. Available:
] <https://cdn.instructables.com/FOG/POMF/IKLFPR1Q/FOGPOMFIKLFPR1Q.LARGE.jpg?auto=webp&width=651>.
- [23 [En línea]. Available: http://www.esploradores.com/wp-content/uploads/2016/08/PINOUT-NodeMCU-1_1.8.png.
]
- [24 E. d. I. Rosa, «EMONCMS,» 2018. [En línea]. Available: <https://mse-iot.mybluemix.net/red/#flow/273d4f64.5b428>.
]
- [25 Node-MCU, [En línea]. Available: <https://statics3.seeedstudio.com/seeed/img/2017-03/qluwTVU7FQlvaC8dZy6x2JaM.jpg>. [Último acceso: 2017].
]



Anexos



Anexos

Anexo I Diagramas Eléctricos

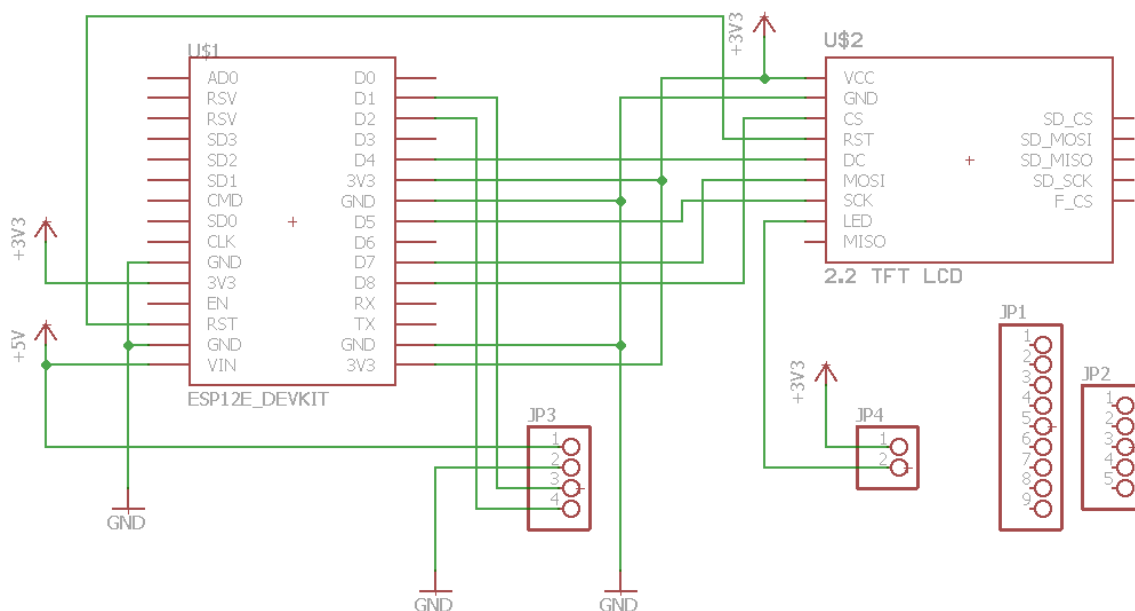


Figura 43 Diagrama Eléctrico modulo Procesador y Display

Fuente: Elaboracion propia de acuerdo al diagrama electrico

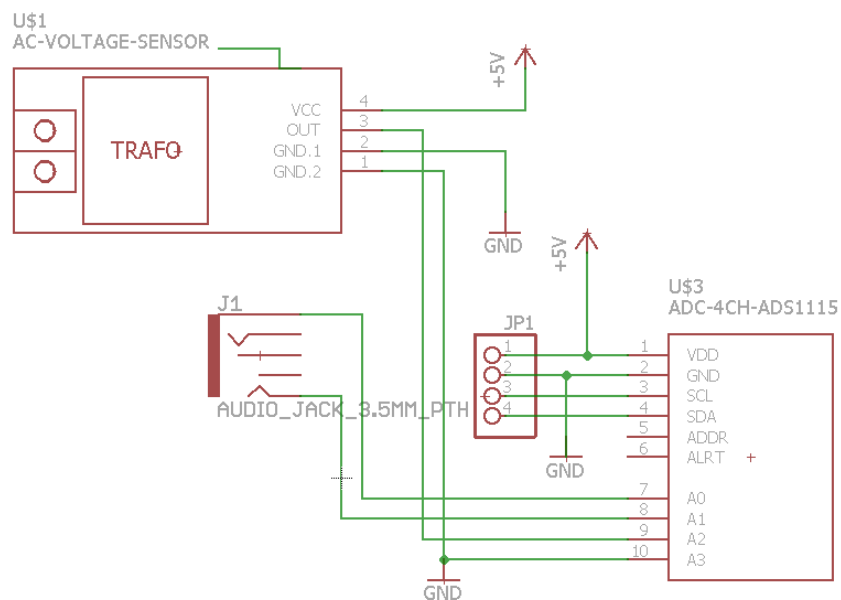


Figura 44 Diagrama eléctrico módulo de potencia

Fuente: Elaboracion propia de acuerdo al diagrama electrico

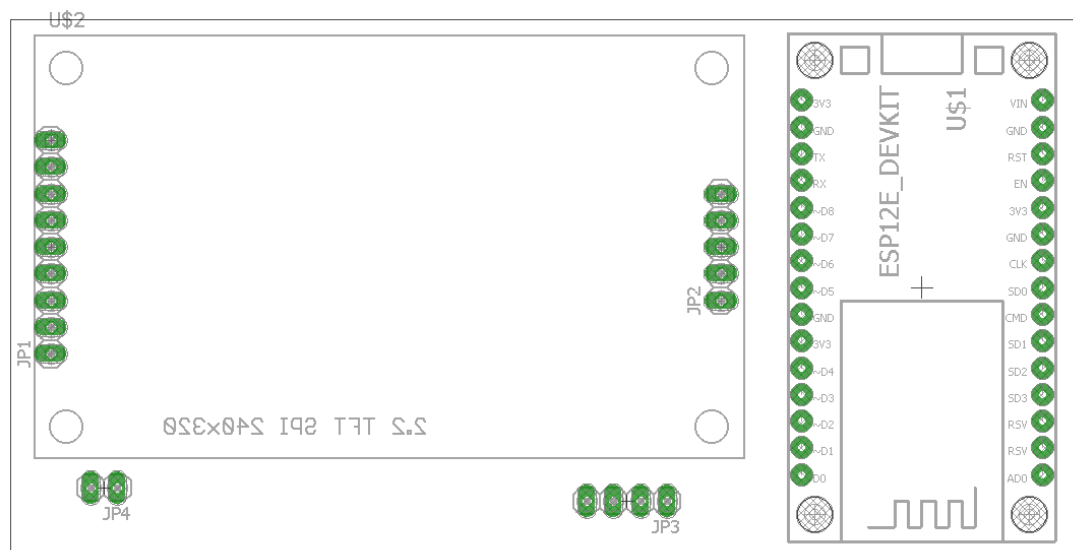


Figura 45 Modulo de Procesador y Display

Fuente: Elaboracion propia de acuerdo al PCB del Modulo principal

Anexo II PCB's (circuitos impresos)

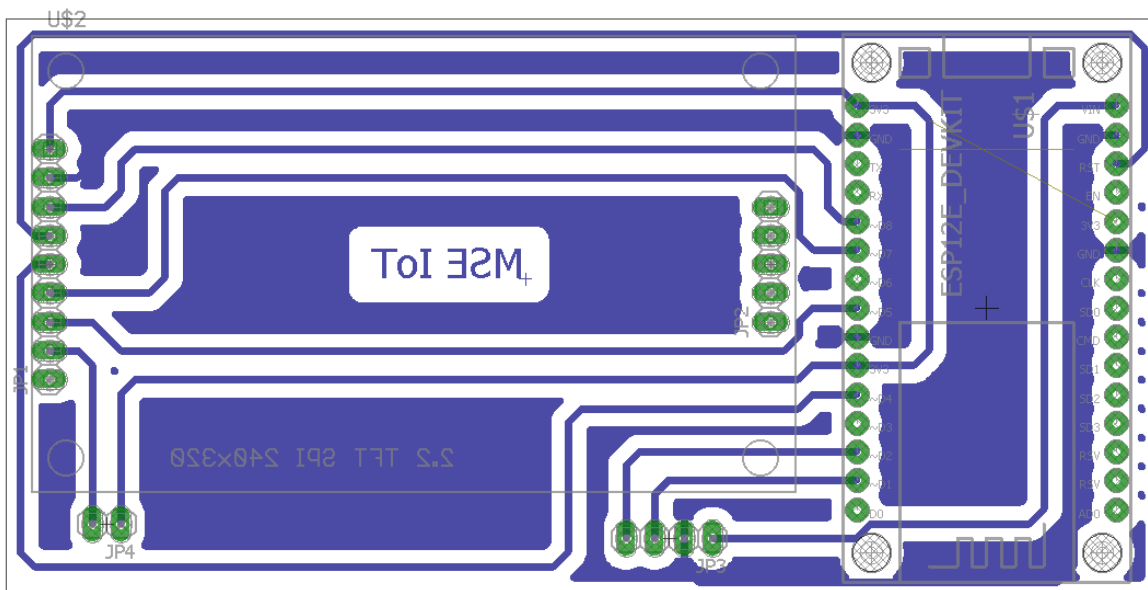


Figura 46 PCB modulo Procesador y display

Fuente: Elaboracion propia del PCB del modulo de Procesador y display

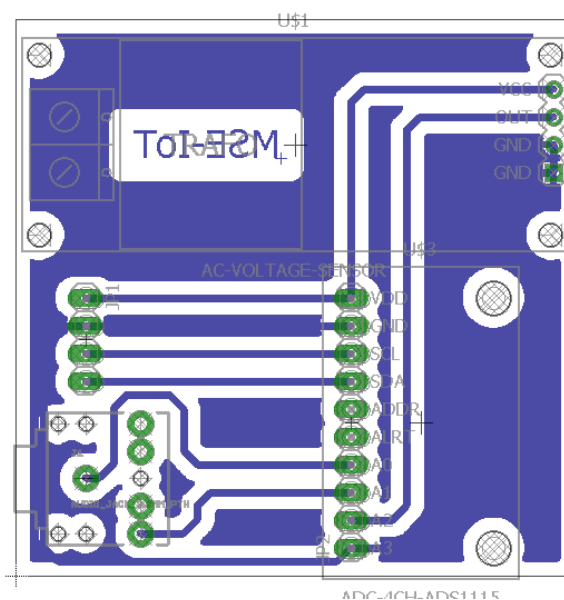


Figura 47 PCB Modulo de Potencia

Fuente: Elaboracion propia del PCB del modulo de potencia

Anexo III Diagramas NodeMCU

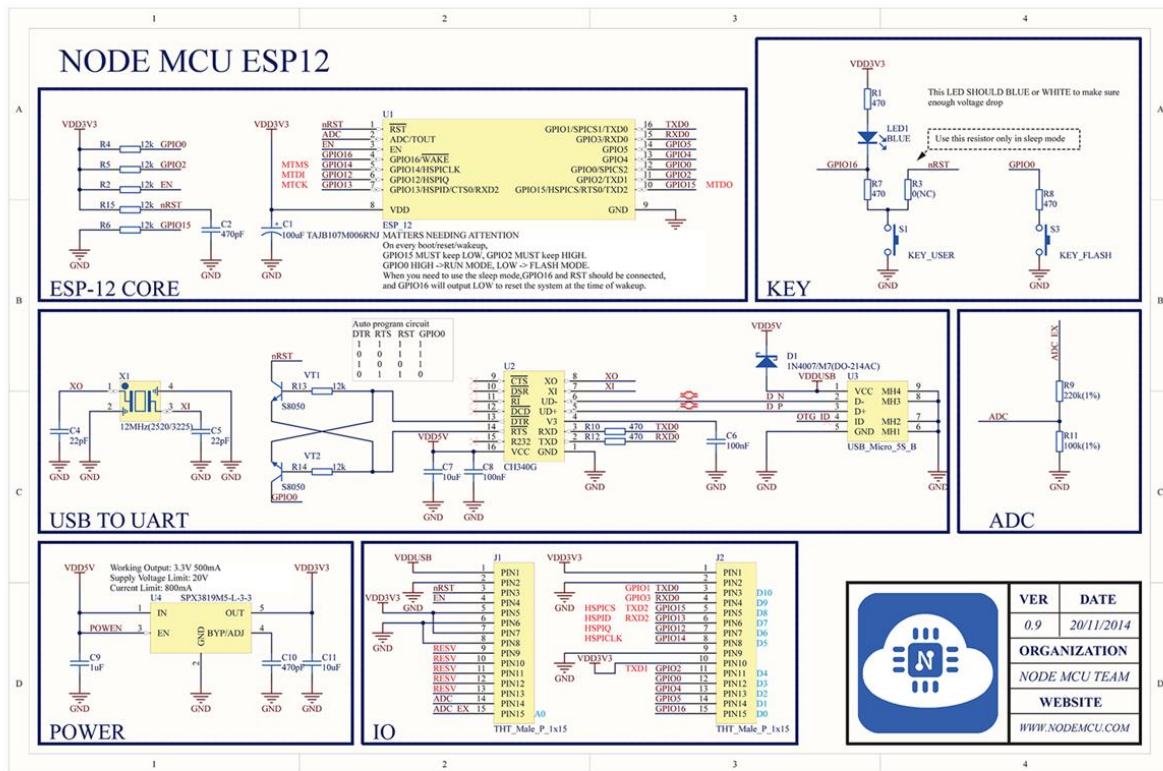


Figura 48 Diagrama a bloques NodeMCU [22]

Fuente: <https://cdn.instructables.com/FOG/POMF/IKLFPR1Q/FOGPOMFIKLFPR1Q.LARGE.jpg?auto=webp&width=651>

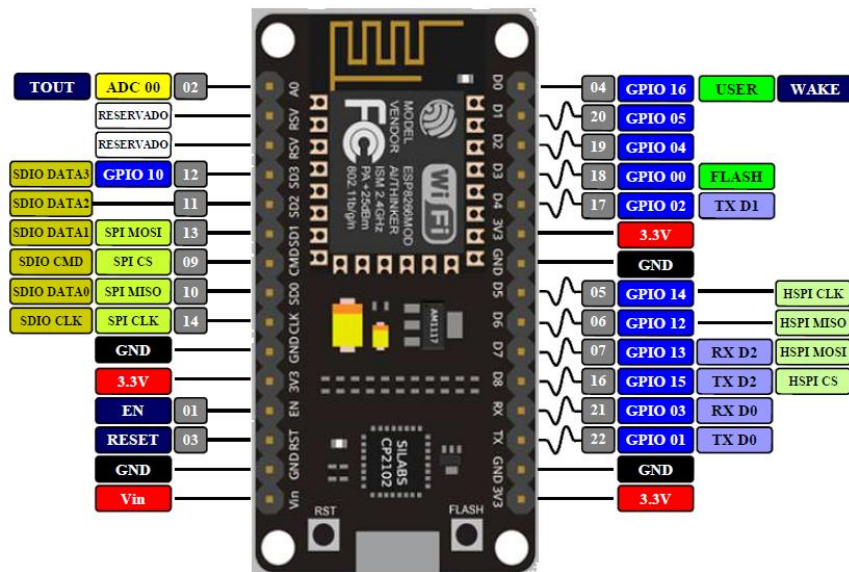


Figura 49 Asignación de pines Node-MCU [23]

Fuente: http://www.esploradores.com/wp-content/uploads/2016/08/PINOUT-NodeMCU-1_1.8.png