



ANALIZADOR INTELIGENTE DE CONSUMO ENERGÉTICO

Esther Gascó Marco

Grado en Tecnologías de las Telecomunicaciones
Sistemas embebidos

Jordi Bécares Ferrés

Pere Tuset Peiró

08/06/2019



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-

SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Analizador inteligente de consumo energético</i>
Nombre del autor:	<i>Esther Gascó Marco</i>
Nombre del consultor/a:	<i>Jordi Bécares Ferrés</i>
Nombre del PRA:	<i>Pere Tuset Peiró</i>
Fecha de entrega (mm/aaaa):	06/2019
Titulación:	<i>Grado de Tecnologías de las Telecomunicaciones</i>
Área del Trabajo Final:	<i>Sistemas emcastados</i>
Idioma del trabajo:	<i>Español</i>
Palabras clave	<i>Consumo, Eficiencia energética, IoT, MQTT</i>
Resumen del Trabajo:	
<p>Hacer de este un mundo sostenible es uno de los grandes retos de hoy en día. Para alcanzar este fin, no es suficiente con realizar investigaciones sobre nuevas fuentes de energía sino hay que actuar mediante un consumo más responsable y eficiente. Sin duda, la mejor forma de reducir este consumo es a través de la monitorización del uso de la energía mediante los analizadores de consumo energético.</p> <p>Los sistemas empotrados nos permiten diseñar dispositivos capaces de contribuir a este objetivo.</p> <p>Este trabajo trata de dar solución a este problema a través del uso del microcontrolador MSP432 donde se diseña un dispositivo capaz de monitorizar cualquier parámetro relacionado con la energía: voltaje, corriente, potencia y consumo instantáneo, en tiempo real. Además, se dota a este dispositivo con conectividad Wifi para posibilitar su acceso al Internet de las Cosas.</p> <p>Se establece el protocolo MQTT como canal de comunicación con la plataforma Web, donde se desarrolla una interfaz gráfica para la visualización de las mediciones e interacción del usuario.</p>	

Finalmente se obtiene un dispositivo listo para empezar a funcionar y controlar el consumo del aparato que se le conecte contribuyendo de esta forma a mejorar el consumo energético.

Abstract:

Making this a sustainable world is one of the great challenges of today. To achieve this goal, it is not enough to carry out researches about new sources of energy, but rather to act through more responsible and efficient consumption. Surely, the best way to reduce this consumption is through monitoring the use of energy through energy consumption analyzers.

Embedded systems allow us to design devices able of contributing to this goal.

This work tries to solve this problem through the use of the MSP432 microcontroller where a device is designed capable of monitoring any parameter related to energy: voltage, current, power and instantaneous consumption, in real time. In addition, this device is equipped with Wifi connectivity to enable your access to the Internet of Things.

The MQTT protocol is established as a communication channel with the Web platform, where a graphic interface for the visualization of the measurements and user interaction is developed.

Finally you get a device ready to start working and control the consumption of the device connected to it, contributing in this way to improve the energy consumption.

Índice

Capítulo 1 Introducción	1
1.1 Contexto y justificación del proyecto	1
1.2 Descripción del proyecto.....	3
1.2 Objetivos	4
1.3.1 Objetivos principales	4
1.3.2 Objetivos extraordinarios.....	4
1.3.3 Objetivos transversales	4
1.4 Enfoque y método seguido	5
1.5 Planificación del trabajo	6
1.5.1 Listado de tareas.....	7
1.5.2 Diagrama de Gantt	9
1.6 Recursos empleados	12
1.6.1 Recursos de software.....	12
1.6.2 Principales recursos del diseño de hardware	13
1.7 Producto obtenido.....	13
1.8 Descripción de los capítulos de la memoria.....	13
Capítulo 2 Antecedentes.....	15
2.1 Estado del arte	15
2.1.1 Evolución histórica de la eficiencia energética	15
2.1.2 Tecnologías de eficiencia energética en el mercado	17
2.1.3 Tecnología actual en sistemas empotrados	18
2.1.4 Tecnologías de comunicación para el IoT	18
2.2 Estudio de mercado.....	20
2.2.1 Sistemas de monitorización de consumo energético	20
Capítulo 3 Descripción funcional	22
3.1 Diseño del sistema	22
3.1.2 Protocolos de comunicación	23
3.2 Diseño del software del dispositivo.....	24

3.2.1 Aplicación de uso	25
3.3 Diseño de la aplicación Web	26
3.3.1 Aplicación de uso	27
3.4 Propuesta de comunicaciones	27
3.5 Propuesta de hardware	29
Capítulo 4 Descripción detallada.....	30
4.1 Diseño de Hardware	30
4.1.1 Circuito del sensor de corriente	31
4.1.2 Circuito del sensor de voltaje.....	33
4.1.3 Circuito de los relés de protección	36
4.1.4 Diseño general del dispositivo	37
4.2 Diseño de Software	40
4.2.1 Estructura del protocolo MQTT	40
4.2.2 Desarrollo del software del dispositivo	42
4.2.3 Desarrollo de la plataforma Web.....	43
Capítulo 5 Viabilidad técnica	49
Capítulo 6 Valoración económica.....	50
6.1 Presupuesto del prototipo	50
6.2 Presupuesto de industrialización.....	52
6.2.1 Mercado CE de dispositivos electrónicos.....	52
6.2.2 Presupuesto	53
Capítulo 7 Conclusiones.....	56
7.1 Objetivos	56
7.1.1 Objetivos cumplidos	56
7.1.2 Objetivos no cumplidos:.....	57
7.2 Conclusiones del trabajo.....	58
7.3 Autoevaluación	58
7.3.1 Tiempo dedicado a cada PEC	59
7.4 Líneas de trabajo futuro	59
Glosario	60
Bibliografía.....	61

Listado de figuras

Figura 1 Consumo y uso de la energía en los hogares españoles	2
Figura 2. Planificación inicial	10
Figura 3. Planificación final.....	11
Figura 4. Evolución histórica de la generación eléctrica en España	16
Figura 5. Efergy Engage Hub Pack	21
Figura 6. Pack Energy de Wattio	21
Figura 7. Diagrama de funcionamiento general del sistema.....	22
Figura 8. Protocolos de comunicación.....	23
Figura 9. Diagrama de bloques del Software	24
Figura 10. Caso de uso de ejecución de software.....	25
Figura 11. Configuración Web.....	26
Figura 12. Caso de uso de la aplicación Web.....	27
Figura 13. Protocolo MQTT y protocolo REST.....	28
Figura 14. MSP432P401R y CC3100	29
Figura 15. Diagrama de bloques del diseño de hardware	30
Figura 16. Sensor de corriente inductivo CST-1020.....	31
Figura 17. Inducción electromagnética	31
Figura 18. Curva de respuesta del sensor de corriente CST-1020	32
Figura 19. Offset DC de 2.5V	33
Figura 20. Transformador 230VAC/18VAC.....	33
Figura 21. Transformador monofásico ideal.....	34
Figura 22. Señal de tensión adaptada	36
Figura 23. Módulo de dos relés de 5VDC	36
Figura 24. Placa de sensores y relés.....	37
Figura 25. Cableado de los sensores y relés al MSP432P401R.....	37
Figura 26. Visualización interior del dispositivo	38
Figura 27. Producto final	38
Figura 28. Esquema electrónico	39
Figura 29. Conexión cliente - broker.....	41
Figura 30. Suscripción a mensajes.....	42
Figura 31. Flujo de recepción de mediciones.....	44
Figura 32. Flujo de envío de alarmas	45
Figura 33. Botón de apagado remoto	45
Figura 34. Flujo de temporización.....	46
Figura 35. Flujo de aviso de alarma por e-mail	46
Figura 36. Flujo de almacenamiento de las lecturas	47
Figura 37. Interfaz gráfica	47
Figura 38. Interfaz gráfica visualizando mediciones.....	48

Listado de tablas

Tabla 1. Listado de fases y tareas.....	8
Tabla 2. Comparativa de redes de comunicación IoT [8].....	19
Tabla 3. Presupuesto de componentes del prototipo	51
Tabla 4. Presupuesto de la mano de obra del prototipo	51
Tabla 5. Presupuesto final del prototipo.....	52
Tabla 6. Presupuesto de certificación CE	53
Tabla 7. Presupuesto de componentes para fabricación industrial	54
Tabla 8. Presupuesto de la mano de obra para fabricación industrial.....	54
Tabla 9. Presupuesto final de fabricación industrial	55
Tabla 10. Tiempo dedicado a cada PEC	59

Capítulo 1

Introducción

1.1 Contexto y justificación del proyecto

En los últimos años se ha escuchado hablar mucho sobre el calentamiento global, los problemas ambientales y la contaminación, donde cada vez es más urgente la actuación frente a estas preocupaciones que sacuden al planeta. Actualmente existen instituciones dedicadas a tal fin, las cuales buscan, a través de estudios y análisis ambientales, estimular a la población para la adopción de medidas correctoras respecto a la energía, reciclaje y combustible con el propósito de mejorar la calidad de vida presente y futura.

El consumo de recursos cada vez más desmedido y el aumento de la demanda y producción de energía, principalmente a través de combustibles fósiles, han provocado que se produzca este cambio climático en el planeta. Por ello, un primer paso para actuar frente a estos problemas ambientales, sociales y económicos, y de esta forma evitar sus consecuencias derivadas relacionadas con el aumento de temperatura, disminución de precipitaciones y aumento del nivel del mar, es el ahorro energético.

Hay muchas formas de reducir el consumo de energía: reemplazar equipos eléctricos viejos, instalar sistemas de iluminación remotos, reconfigurar sistemas de agua caliente, etc. Pero el problema de estas medidas es que se trata de soluciones con un gran valor económico asociado, por otro lado, hay cosas más pequeñas que se pueden hacer para conseguir este fin.

Sin duda, la mejor forma de reducir el consumo es a través de la monitorización del uso de la energía mediante los analizadores de consumo energético. Solo si se conoce dónde se usa realmente la energía se pueden idear soluciones para reducir el uso y los costes asociados, así

como el impacto ambiental. El monitoreo del consumo no solo proporciona una visión de la cantidad de energía utilizada en un período de tiempo determinado, sino que también señala irregularidades en el consumo energético.

A continuación, se muestra una imagen donde se pueden ver de forma gráfica los consumos según el tipo de electrodoméstico y servicios:

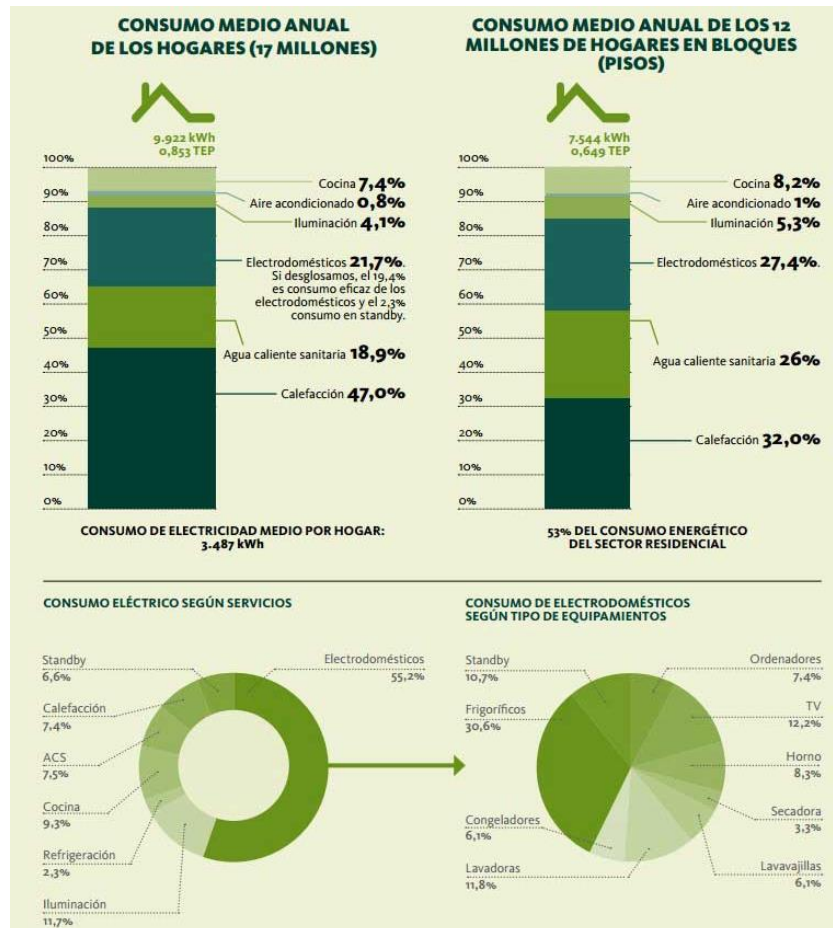


Figura 1 Consumo y uso de la energía en los hogares españoles

Monitorizar el consumo energético es una solución fundamental para adoptar las medidas correctoras respecto al uso de la energía, además, si se basa en el concepto de Internet de las cosas (IoT), el cual hace referencia a la interconexión digital de objetos cotidianos con internet, es posible utilizarlo remotamente como mecanismo de protección frente a las irregularidades energéticas. Teniendo en cuenta la importancia de los conceptos IoT y eficiencia energética en el desarrollo tecnológico, se realizará un estudio en el que ambos se complementen con el objetivo de buscar una reducción económica y energética, por ello, este proyecto, se centrará en hacer frente a estos problemas a través del desarrollo de un hardware económico y el uso de software libre.

1.2 Descripción del proyecto

El presente proyecto consiste en el diseño y configuración de un analizador de consumo energético conectado al IoT. Este dispositivo será capaz de monitorizar en tiempo real, así como de controlar y comunicar al servidor Web las mediciones del voltaje (V), corriente (A), potencia (kW) y consumo instantáneo (kWh) de cualquier aparato que tenga conectado.

No se limitará a la monitorización, sino que se tratará de un dispositivo con el que se puede interactuar. Desencadenará eventos cuando se cumplan ciertas condiciones: enviar e-mail en caso de recibir alguna alarma además de activar los relés para cortar la tensión. Estas alarmas serán configurables desde la interfaz Web donde también será posible configurar los periodos de medición, así como el apagado de forma remota.

Todo esto es posible gracias al empleo de una gran variedad de tecnologías, al uso de plataformas gratuitas y software de código abierto. El núcleo central es el microcontrolador MSP432 de Texas Instruments el cual se encarga de recibir y gestionar las lecturas de los diferentes sensores. Se han integrado dos tipos de sensores, por un lado, se obtienen las lecturas de la tensión a través de un sensor AC/AC, por otro lado se obtienen las lecturas de la corriente mediante un sensor inductivo. A través de estas lecturas es posible calcular el resto de los parámetros necesarios para este proyecto: potencia y consumo instantáneo.

Respecto a la conectividad con IoT, se empleará el módulo CC3100 de Texas Instruments, el cual se trata de un procesador de red Wifi que puede interactuar con el microcontrolador MSP432 a través de la interfaz SPI [1] que se trata de un estándar de comunicaciones usado para la transferencia de información entre circuitos integrados en equipos electrónicos. Se recibirán todos estos datos en el servidor Web Node-Red, incluido dentro de IBM Cloud, a través del protocolo de comunicación MQTT [2], basado en la publicación-subscripción.

1.2 Objetivos

1.3.1 Objetivos principales

1. Recopilación de los datos de los distintos sensores
2. Medir la potencia consumida por una línea de corriente
3. Proveer al dispositivo de mecanismos de seguridad ante la recepción de diversas alarmas
4. Dotación de conectividad Wifi al dispositivo para su implementación en IoT
5. Monitorizar el consumo energético de un determinado equipo en tiempo real a través de una interfaz Web
6. Posibilitar la interacción con el dispositivo a través de la configuración de alarmas de potencia, voltaje, corriente y consumo
7. Posibilitar la interacción con el dispositivo para el establecimiento de periodos de medición
8. Posibilidad de encender y apagar el dispositivo remotamente

1.3.2 Objetivos extraordinarios

1. Almacenamiento persistente de las alarmas en la memoria del dispositivo
2. Almacenamiento de los históricos de las mediciones y visualización a través de un calendario en la interfaz Web
3. Funcionalidades adiciones: visualización en un LCD incorporado en el dispositivo, de forma local, las mediciones

1.3.3 Objetivos transversales

Los objetivos transversales de este proyecto, fuera del alcance técnico, son los siguientes:

1. Mejorar la eficiencia energética

2. Concienciar sobre el uso de determinados electrodomésticos y el consumo que conllevan
3. Ofrecer un sistema robusto y de bajo coste que pueda ser utilizado fácilmente por cualquier persona sin conocimientos previos en informática

1.4 Enfoque y método seguido

Este proyecto tiene un enfoque práctico, por ello se ha empleado la siguiente metodología:

1. Análisis y estudio de las necesidades globales del proyecto
2. Diseño del prototipo
3. Elección de los componentes pasivos y activos del circuito
4. Adquisición de los materiales
5. Montaje del hardware y comprobación de su funcionamiento
6. Elección de la plataforma para el desarrollo de IoT
7. Programación del microcontrolador y realización de pruebas

En primer lugar, se ha realizado un análisis sobre las necesidades globales del sistema. Este análisis se centra en la elección de un dispositivo que cumpla con los objetivos marcados en el punto anterior, principalmente tiene que ser capaz de tomar mediciones de energía y disponer de conectividad Wifi. Se han planteado dos opciones: emplear un dispositivo comercial y mejorarlo para adaptarlo a las necesidades del proyecto o desarrollar un dispositivo nuevo.

Se ha descartado la primera opción ya que, aunque existen gran cantidad de dispositivos con funcionalidades similares a las buscadas, todos están basados en software propietario que dependen de un fabricante, por ello resulta muy difícil realizar una adaptación. Por otro lado, este proyecto pretende hacer uso de un hardware económico y el empleo de software de código abierto, por consiguiente, se ha decidido desarrollar un producto nuevo.

Otro punto importante dentro de esta metodología es la elección del entorno de desarrollo para la conexión del hardware con IoT. Después de analizar las diferentes opciones, tales como

Crosser, Flow, ioBroker, etc. se ha decido emplear Node-Red [3] ya que se trata de una herramienta de fácil manejo, con un gran apoyo de la comunidad y gratuita. Node-Red está escrito en JavaScript y se ejecuta en la plataforma NodeJS. Una de sus principales ventajas es la extensa biblioteca de nodos que contiene, casi todo lo que se quiera hacer ya está disponible como un módulo y es fácilmente accesible a través del repositorio integrado donde muchos de estos módulos son proporcionados por la comunidad ya que se basa en el código abierto.

Node-Red se puede alojar en diversas plataformas:

- De forma local en un PC con Windows o Linux
- En dispositivos de hardware de bajo coste como Raspberry Pi
- En la nube

Tras valorar estas opciones, se ha considerado que tanto el alojamiento de forma local como en dispositivos de hardware de bajo coste suponen un coste innecesario, por ello, se decide alojar Node-Red en la nube, concretamente en el Cloud de IBM. Además, este Cloud nos proporcionará todas las herramientas necesarias, dashboard y base de datos, para desarrollar el proyecto.

Finalmente, después de realizar este análisis de necesidades, tanto de hardware como de software, se ha fragmentado el proyecto en bloques funcionales y se ha definido un listado de tareas donde se establecen varios hitos a alcanzar en cada una de las fases.

1.5 Planificación del trabajo

Una vez fijados los objetivos y la metodología del proyecto, es necesaria la realización de una planificación donde se irá controlando el desarrollo del proyecto con el fin de alcanzar todos los objetivos con éxito.

1.5.1 Listado de tareas

En la siguiente tabla se recogen las diversas tareas en las que se descompone el proyecto, así como los objetivos conseguidos:

Número de tarea	Tarea	Descripción
FASE 0: Definición del proyecto		
0	Primeros pasos	Práctica para conocer la tecnología que se va a emplear durante el proyecto, así como el funcionamiento del aula y la Wiki. Se cargarán diversos ejemplos en el MSP432 para familiarizarse con su programación.
1	Propuesta de proyecto	Se elegirá el proyecto a realizar y se redactará la propuesta.
2	Plan de trabajo	Se distribuirá el tiempo que requerirá cada una de las partes que forman el proyecto: diseño, montaje y programación así como la definición de objetivos.
FASE 1: Diseño de hardware y programación local		
3	Diseño de hardware	Realización del esquema y montaje de todas las partes del analizador de energía.
4	Toma de medidas	Antes de realizar la conexión a las entradas analógicas A0 y A1 del kit de desarrollo, se comprobará con un osciloscopio que la señal está dentro de los rangos en los que trabaja el microcontrolador (0-5V).
5	Programación de las tareas. Se subdividen en las siguientes:	Se llevará a cabo la programación del microcontrolador con las tareas referentes a los pines analógicos y digitales de este.
	Sensor de Corriente	Tarea que recoge los datos del sensor de corriente conectado al pin A0.
	Sensor de Voltaje	Tarea que recoge los datos del sensor de voltaje conectado al pin A1.
	Activación de los relés	Tarea de activación los relés frente a alarmas y encendido de led en caso de que no esté activado.
	Alarmas	Tarea que recoge las alarmas.
6	Pruebas en local	A través del puerto serie se comprobará que las mediciones y cálculos son correctos.
7	Entrega 1	Se realizará el informe de la PEC 1
FASE 2: Conexión a IoT		
8	Integración de Wifi	Se añadirá al hardware el módulo CC3100 para dotar al dispositivo de conectividad Wifi.
9	Programación de las tareas. Se subdividen en las siguientes:	Se programarán las tareas relacionadas con la conectividad IoT del dispositivo.
	Configuración de eventos	Se implementará MQTT en el código del proyecto para establecer comunicación con IoT.
	Lectura de sensores	Se recogerá la lectura de los diferentes sensores además de posibilitar la introducción de alarmas para estos.

	Alarmas	En caso de que se active alguna alarma, se enviará un correo electrónico con el aviso a una dirección e-mail preconfigurada.
	Interruptor inteligente	Se incorporará la función de interruptor inteligente
	Temporización	Dentro de la tarea MQTT se implementará la temporización para que sea el usuario quien decida el tiempo de medida.
10	Desarrollo de la interfaz gráfica.	Diseño de la interfaz Web para la interacción con el usuario.
11	Pruebas en remoto	Se realizarán pruebas en remoto para comprobar la correcta recepción de los eventos a través de la Web.
12	Entrega 2	Se redactará el informe referente a la PEC 2.
FASE 3: Código final		
13	Persistencia de datos. (Opcional) Se subdivide en las siguientes subtareas:	Se almacenarán las mediciones y valores de las alarmas.
	Log	Se almacenan las lecturas en una base de datos.
	Configuración de históricos	Se implementará en la interfaz Web la visualización de históricos de las diferentes mediciones almacenadas en la base de datos.
	Almacenamiento de alarmas en flash	Se almacenarán las últimas alarmas configuradas en la memoria flash del dispositivo para que estén disponible en el próximo arranque.
14	Incorporación de pantalla. (Opcional)	En el caso de que el proyecto se finalice con tiempo, se incorporará una pantalla para visualizar el consumo y las alarmas de forma local
15	Entrega 3	Se redactará el informe correspondiente a la PEC 3
Documentación		
16	Memoria	Se redactará la memoria del proyecto
17	Documento presentación	Se realizará la presentación del proyecto en ppt y vídeo
18	Defensa	Presentación del TFG online frente al tribunal

Tabla 1. Listado de fases y tareas

Como se observa en la tabla, se han dividido las distintas tareas en 5 fases:

- En primer lugar, se encuentra la fase de búsqueda y proposición del Trabajo Fin de Grado. Después de familiarizarse con los sistemas encastados, se realiza la propuesta de proyecto donde se define el tipo de trabajo que se va a realizar. Por otro lado, a través del plan de trabajo, se lleva a cabo el proceso de investigación donde se definen todos los objetivos para posteriormente elegir los componentes de hardware adecuados para el desarrollo del dispositivo. Además, se distribuirá el tiempo de cada una de las fases.
- La fase 1 es la referente al diseño de hardware y programación en local. En esta fase se lleva a cabo el proceso de diseño donde se estudiará el funcionamiento del módulo de desarrollo MSP432P401R y se elegirán los sensores que cumplan con los requisitos del proyecto. Tras el montaje del prototipo, se procederá a realizar la programación para visualizar las diferentes mediciones de forma local.

- La conectividad con IoT se alcanza en la segunda fase, donde se dota al dispositivo de conexión Wifi y se programan las tareas para desencadenar eventos. Por otra parte, se desarrollará la interfaz Web para la visualización de las mediciones en tiempo real.
- La cuarta fase recoge todos los avances hasta el momento y proporciona mejoras respecto a la interacción del usuario por remoto con el dispositivo. También se proponen funcionalidades adicionales a implementar en la interfaz Web como la persistencia de datos y la incorporación de una pantalla para visualizar los eventos en local.
- El último punto será el referente a la elaboración de la documentación. Esta última fase se dedicará a la redacción de la memoria del proyecto y la elaboración de la presentación para la posterior defensa frente al tribunal.

1.5.2 Diagrama de Gantt

A través del diagrama de Gantt se expone de forma más visual el tiempo de dedicación previsto para la realización de las diferentes tareas y fases a lo largo del semestre. Se consideran días de 5 horas, respetando fines de semana y festivos. Además, se han incorporado horas reservadas para afrontar cualquier tipo de retraso o modificación en las tareas y poder alcanzar los diferentes hitos.

Capítulo 1. Introducción

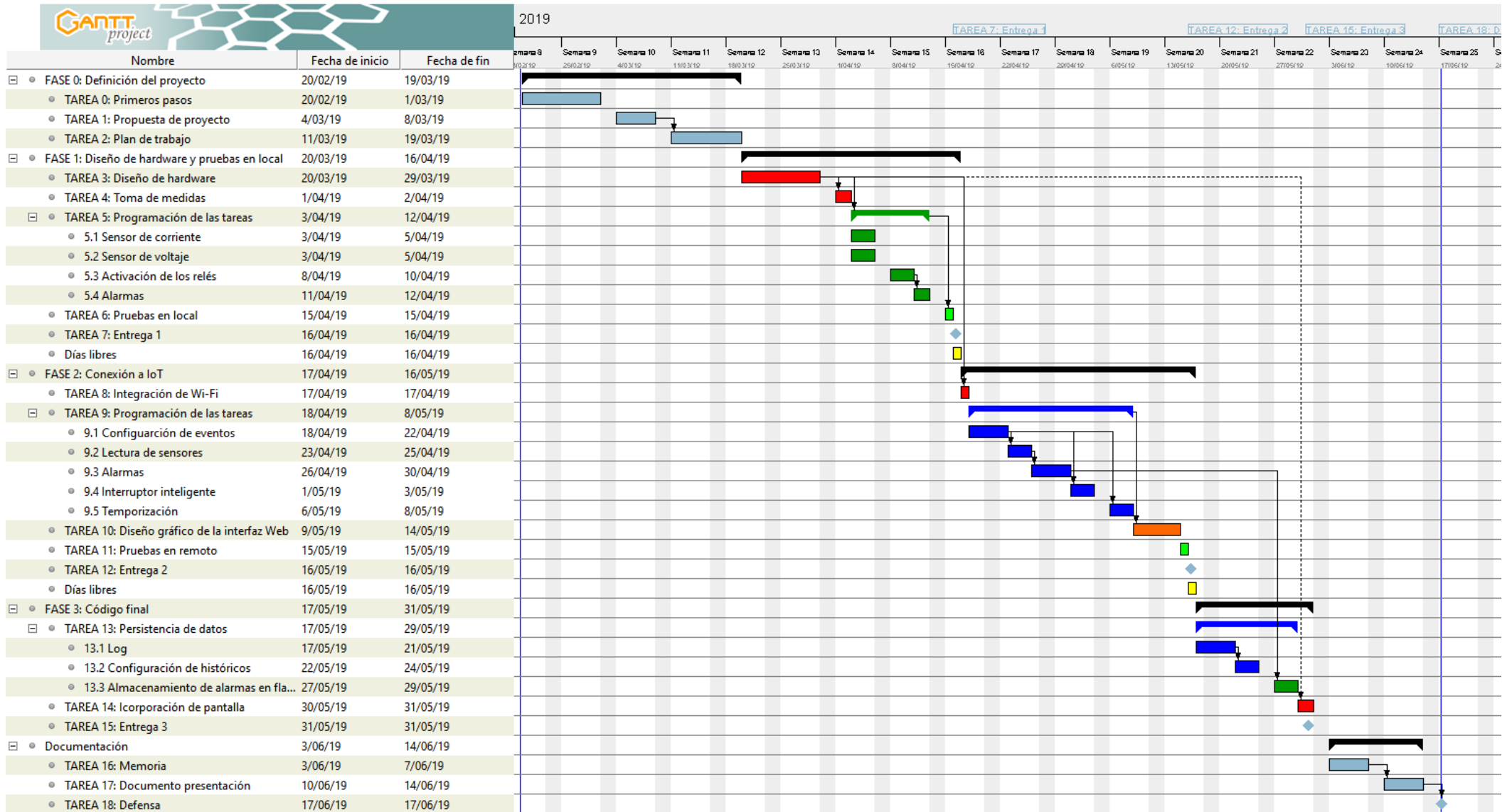


Figura 2. Planificación inicial

Capítulo 1. Introducción

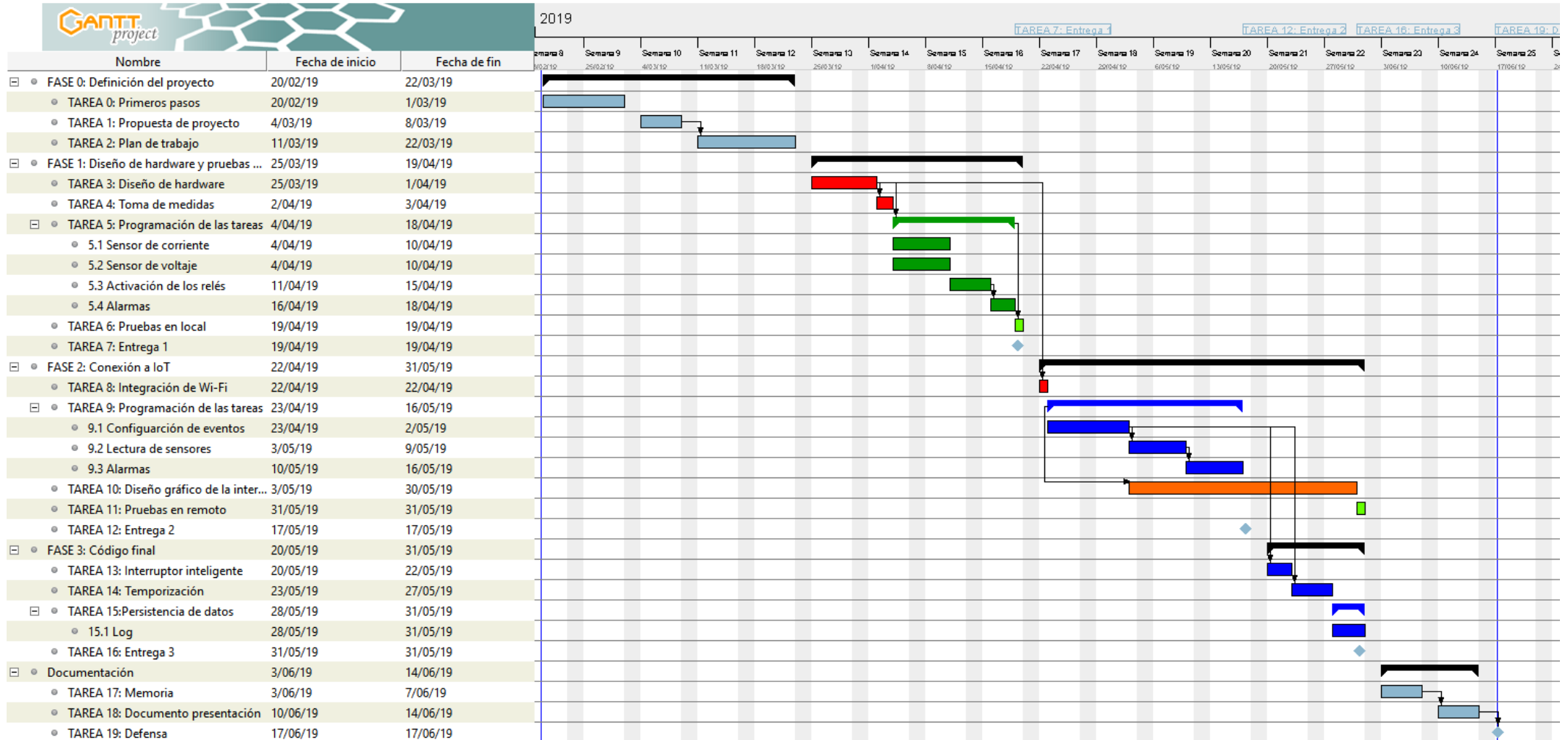


Figura 3. Planificación final

Según se observa, la planificación ha sufrido varios retrasos y no se han alcanzado las tareas marcadas como opcionales. Se pudo seguir correctamente hasta alcanzar la tarea 7, punto donde se situó el primer hito en la entrega de la PEC 1, ya que los conocimientos previos para afrontarla eran consistentes. Posteriormente se ha requerido de un estudio preliminar de la plataforma de desarrollo Web, por consiguiente, se ha necesitado mucho más tiempo del planificado. Después de superar estos conocimientos necesarios, se han afrontado todas las tareas principales, aunque la función de interruptor inteligente ha quedado limitada a apagado remoto, por ello vemos un solapamiento entre la fase 2 y 3 en el diagrama de Gantt final, así como el diseño de la interfaz gráfica al mismo tiempo que se han ido programando las diferentes tareas.

Se han descartado las tareas marcadas como opcionales que coinciden con los objetivos opcionales, pero se ha implementado el almacenamiento de las mediciones en una base de datos para poder emplearlos en una futura mejora del dispositivo.

1.6 Recursos empleados

1.6.1 Recursos de software

- CCS Cloud: Software de programación del microcontrolador en la nube de Texas Instruments. Se empleará el lenguaje de programación de Arduino.

- IBM Cloud: Plataforma de desarrollo que contiene todas las herramientas necesarias para la evolución de este proyecto, se emplearán las siguientes:
 - Node-Red: Herramienta de desarrollo basada en flujos para la programación visual. Se programará también con JavaScript.
 - Node-Red Dashboard: Interfaz gráfica para la visualización de los eventos programados en Node-Red.
 - Cloudant: Servicio de base de datos distribuida no relacional.

1.6.2 Principales recursos del diseño de hardware

- Placa de desarrollo de Texas Instruments MSP-EXP432P401R
- Módulo Wifi de Texas Instruments CC3100BOOST
- Sensores:
 - CST1020 de Triad Magnetics: Sensor de corriente inductivo de 20 Amperios con salida en tensión.
 - Convertidor AC/AC de 220V-18V de Block Trafo.
- Relés de 5VDC
- Fuente de alimentación AC/DC de 220V-5V de Mean Well con PN: RPS-30-5 para la alimentación de la placa de desarrollo.

1.7 Producto obtenido

A la finalización de este proyecto, se ha obtenido un dispositivo analizador inteligente de consumo energético completamente funcional, conectado a IoT. Por otro lado, se ha elaborado la memoria con los detalles del desarrollo.

1.8 Descripción de los capítulos de la memoria

Este proyecto se organiza según los siguientes capítulos:

Capítulo 1. Introducción: En esta primera parte se da a conocer la motivación que impulsó a la realización de este proyecto, así como los objetivos y planificación para su realización.

Capítulo 2. Antecedentes: En este capítulo se realizará tanto el estado del arte como el estudio de mercado donde se revisará la evolución histórica de la eficiencia energética para posteriormente centrarse en la actualidad de las tecnologías empleadas para la eficiencia

energética, sistemas empotrados y conectividad IoT. Por otro lado, en el estudio de mercado se analizarán distintos productos relacionados con el presente proyecto.

Capítulo 3. Descripción funcional: En esta parte se describe el desarrollo funcional del trabajo sin entrar en detalle. Se incluirán los diagramas de las distintas partes del proyecto.

Capítulo 4. Descripción detallada: En este capítulo se describen todas las características técnicas de todos los componentes que forman el sistema, tanto de hardware como de software, descritos en el capítulo anterior.

Capítulo 5. Viabilidad técnica: Se estudiará la viabilidad del proyecto donde se intenta predecir el eventual éxito o fracaso del este.

Capítulo 6. Valoración económica: Se realizará la valoración económica del proyecto y se demostrará que utilizando software libre nos encontramos ante un proyecto económico.

Capítulo 7. Conclusiones: Por último, se realizarán las conclusiones acerca del trabajo realizado y las posibles mejoras.

Capítulo 2

Antecedentes

2.1 Estado del arte

En la actualidad, se considera que, a nivel global, un 98% de la energía se desperdicia. Esto es debido al bajo nivel de eficiencia energética en diversos sectores, tales como la industria, el transporte y la vivienda. Solamente con alcanzar la tasa del 3% en productividad energética, el gasto de combustibles fósiles disminuiría considerablemente cada año.

Son las principales economías mundiales quienes más demandan energía y deben hacer frente a este problema. Estas naciones tienen un gran potencial para impulsar el desarrollo económico, medioambiental y social a través de la optimización de la eficiencia energética y la mejora de producción de energía en los puntos de consumo. Solamente con el uso de la iluminación LED, se podría mejorar la eficiencia energética en una vivienda un 500%, igualmente, con el empleo de dispositivos de ahorro energético, como los analizadores inteligentes de energía, o mejorar el aislamiento en las viviendas, se contribuiría positivamente a la reducción de la factura energética, y, por consiguiente, a la mejora de la eficiencia energética.

2.1.1 Evolución histórica de la eficiencia energética

Desde que el fuego fue descubierto por el hombre, hace alrededor de 790.000 años, y empezó a usarlo para diversas tareas como cocinar o calentarse, se dio cuenta de la importancia de la energía y la necesidad de hacer que estas fuentes de energía durasen. También se consiguió energía transformando energía mecánica en térmica simplemente frotando dos piezas de madera o piedra. Posteriormente, se empezó a utilizar otro tipo de fuentes de energía: agua, y viento, hasta el descubrimiento de los combustibles fósiles.

A partir de los inicios de la Revolución Industrial hasta la actualidad, la sociedad ha realizado un gran cambio respecto a su forma de consumir energía. Anteriormente a la era industrial, la mayor fuente de energía provenía de la madera, utilizada para encender fuegos. Durante esta revolución, se comienza a emplear el carbón, principalmente como combustible de la máquina de vapor, el cual lideró el crecimiento industrial hasta que fue revelado por el petróleo como primera fuente [4].

Entre mediados y finales del siglo XIX surge la Segunda Revolución Industrial donde el consumo de energía se incrementa exponencialmente. Junto con el petróleo surge una nueva fuente de energía: el gas. La aparición de la electricidad es decisiva, con ella se establece una forma de almacenar y transportar la energía refinada y estandarizada, facilitando su consumo.

La crisis del petróleo, en 1973, sirve como impulso para el despegue de otros tipos de energías. Es aquí cuando la población pasa a ser consciente del futuro agotamiento de los recursos naturales, lo que conlleva al auge de las centrales nucleares. Por otro lado, surgen las energías renovables como solución ante la gran demanda de energía que está produciendo el denominado Calentamiento Global. Es así como surge un nuevo concepto: la sostenibilidad.

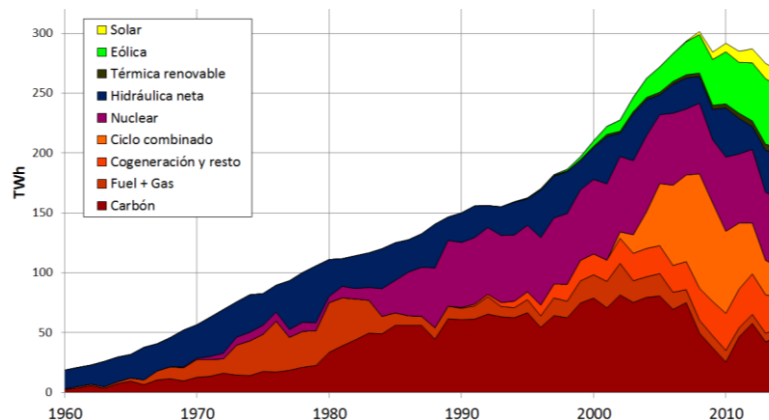


Figura 4. Evolución histórica de la generación eléctrica en España

Como se puede ver en la Figura 4, actualmente se tiene una mayor consciencia de las necesidades del ahorro energético donde empieza a ser común el empleo de fuentes de energías renovables como la eólica, solar, hidráulica o biomasa. La eficiencia energética tanto en las viviendas como en los edificios es clave para la sostenibilidad del planeta, por ello, a través de las instalaciones más eficientes y la reducción de transmisiones, se consigue reducir esta demanda de energía creciente y, por lo tanto, se contribuye a mejorar la eficiencia energética.

2.1.2 Tecnologías de eficiencia energética en el mercado

Las tecnologías de eficiencia energética se pueden encontrar en todas las partes de la cadena de conversión de energía: desde la exploración y producción de recursos de energía primaria hasta la generación de energía y desde las refinerías de petróleo hasta las redes eléctricas, el uso final en la industria, los edificios y el transporte, pero es a nivel comercial el que nos concierne.

La tecnología está continuamente cambiando, a su vez, hace que todo lo que esté relacionado con ella cambie. El sector energético evoluciona juntamente con ella ya que incorpora cambios impulsados por tendencias como puede ser la digitalización del sector la cual se extiende a toda su infraestructura, es decir, tanto de la red como del consumidor. También surgen cambios en la energía distribuida, la electrificación de la economía y gradual penetración de fuentes de energía renovables junto con el vehículo eléctrico [5].

Por parte del consumidor surge un nuevo modelo de relación a través de la conectividad debido a que demanda servicios y productos cada vez más personalizados, sobre la gestión activa del consumo energético, a los que hay que hacer frente. En España se ha creado una red de distribución inteligente la cual permite conocer en todo momento tanto curvas de consumo como la potencia máxima establecida, o, incluso, la forma en que es distribuido el consumo energético. Gracias a esto es posible hacer un uso más eficiente de la energía ya que a través de los resultados obtenido se puede conocer el mejor tipo de tarifa eléctrica para cada consumidor.

El hogar inteligente es clave para la optimización del consumo energético. Surgen dispositivos capaces de controlar la calefacción, climatización, el termostato, todo ello a distancia donde a través de una aplicación se puede monitorizar el consumo de los aparatos conectados además de poder establecer programas que cambian su funcionamiento en tiempo real. Además, aparecen soluciones Smart para el uso doméstico sin necesidad previa de ningún tipo de instalación. Se trata de los sistemas inteligentes de monitorización de energía donde el avance de la tecnología hace que cada vez se desarrollen nuevos tipos de dispositivos.

2.1.3 Tecnología actual en sistemas empotrados

Con la aparición de las placas de desarrollo se abren las puertas a la realización de todo tipo de proyectos. Arduino y Raspberry surgieron como principal elección de mucha gente que empezaba con la programación de sistemas empotrados, pero anteriormente a esto, la tarea de programación resultaba mucho más complicada.

Fue el microcontrolador PIC de Microchip Technology, en 1975, el pionero en este tipo de dispositivos. Se caracterizaba por su bajo coste, fácil disponibilidad y la proliferación de herramientas de programación libres [6]. El PIC se trata de un microcontrolador, esto significa que el procesador está integrado, así como memoria y E/S (entradas/salidas) programables. Hoy en día aún está disponible en el mercado y su uso se da sobre todo en la educación ya que se requiere de conocimientos de C de bajo nivel para su programación.

Arduino surge ante la necesidad que tienen los artistas en sus proyectos de diseño interactivo, de esta forma acceden fácilmente a los microcontroladores empotrados ya que se permite el desarrollo de prototipos electrónicos rápidos y de bajo coste. Se ha convertido en una plataforma de desarrollo sólida, donde no destaca en el hardware, sino en el software, haciendo que su lenguaje de programación sea un estándar el cual se trata de un lenguaje propio basado en el lenguaje de alto nivel Processing, similar a C++ [7].

En la actualidad hay muchas alternativas al uso de Arduino. Se está viendo constantemente una explosión de nuevas tarjetas que salen al mercado y que desaparecen igual de rápido. Entre todas ellas, cabe destacar la gama LaunchPad de Texas Instruments. Se trata de tarjetas de desarrollo basadas en el microcontrolador ARM Cortex M4, son superiores a nivel de hardware que los Arduino, además adoptan el lenguaje de programación de este gracias a la plataforma de prototipado de código abierto Energia. Esto significa que se pueden emplear prácticamente igual como si se tratase de un Arduino. Por otro lado, para proyectos que requieran otras características, se puede programar en lenguaje C.

2.1.4 Tecnologías de comunicación para el IoT

Las redes de comunicaciones son las responsables de conectar los dispositivos, sensores, máquinas o “cosas” que generan datos desde cualquier punto del planeta. Este tipo de redes ha sufrido una gran evolución hacia el sector del IoT en los últimos años donde

numerosas empresas han invertido en ellas. En la tabla siguiente, se pueden ver las redes de comunicación más utilizadas en este sector:

Tecnología	Consumo	Alcance	Madurez	Disponibilidad	Seguridad	Usabilidad	Tasa de datos
GSP/FPRS	Muy alto	Alto	Muy alto	Muy alta	Alta	Alta	alta
SigFox	Bajo	Medio	Alto	Medio	Media	Alta	Muy baja
LoRa	Bajo	Medio	Bajo	Muy bajo (ad hoc)	N.A	Baja	Muy baja
WiFi	Alto	Bajo	Muy alto	Alto	Baja	Alta	Muy alta
BLE	Muy bajo	Muy bajo	Alto	Bajo	Baja	Media	Baja
ZigBee	Medio	Bajo	Medio	Muy bajo	Alta	Baja	Baja

Tabla 2. Comparativa de redes de comunicación IoT [8]

- **GMS/GPRS (M2M):** se trata de redes máquina-a-máquina, vinculadas al uso de una tarjeta SIM para la transferencia y recepción de datos. Su inconveniente es que este tipo de red M2M dificulta la escalabilidad además de que su cobertura depende de un operador con unos costes asociados.
- **SigFox:** es la red de comunicaciones LPWAN (Low-power Wide-area Network) opera en la banda de los 868MHz en Europa y se trata de una de las redes más extendidas a nivel mundial. Su bajo coste, así como la adaptación de los fabricantes de dispositivos a esta tecnología hacen de SigFox una buena opción para las comunicaciones IoT.
- **LoRa:** es una red LPWAN, similar a Sigfox pero mejor preparada para establecer una comunicación bidireccional en tiempo real ya que hace uso de un espectro de comunicaciones más amplio. Su inconveniente es que su despliegue no es tan extenso como el de SigFox
- **WiFi [9]:** es un protocolo de comunicación sin hilos propiedad de Alianza Wifi. Cumple con los estándares 802.11 relacionados con redes inalámbricas de área local. El gran problema es de este tipo de comunicación es la saturación del espectro radioeléctrico causado por la masificación de usuarios.
- **BLE (bluetooth de baja energía):** es una tecnología de red de área personal (PAN). Su uso se establece para la conexión entre dispositivos bluetooth de baja energía, con un consumo y coste reducidos pero el mismo alcance que el bluetooth original. Se emplea para determinadas aplicaciones IoT principalmente a través del uso de los dispositivos *beacons* que se trata de balizas de localización.

- **ZigBee (Radiodifusión digital de bajo consumo):** consiste en una tecnología inalámbrica no en una red de comunicaciones en sí. Su principal uso se da en los proyectos domóticos gracias a sus características:
 - Bajo consumo,
 - Ofrece una seguridad superior al resto de tecnologías
 - Robustez
 - Alta escalabilidad
 - Su topología de red en malla ofrece capacidad para soportar un gran número de nodos

2.2 Estudio de mercado

En el mercado actual se pueden encontrar varios dispositivos, cada uno con un enfoque o características diferentes pero que cumplen el mismo fin: monitorizar el consumo energético. Algunos de estos están conectados a IoT, aunque la gran mayoría son para el uso local. A través de la incorporación de una pantalla se pueden visualizar las mediciones obtenidas, también, en algunos es posible exportar los datos para obtener gráficas e históricos, y otros, los que realmente nos interesan en este estudio, incorporan la conectividad con el Internet de las Cosas.

Para dar solución al problema de la eficiencia energética, muchos fabricantes han decidido desarrollar productos Smart que mejoran el consumo energético ya que a través de ellos se consigue una mejora en hábitos de consumo de los usuarios.

2.2.1 Sistemas de monitorización de consumo energético

Dentro del diseño de productos Smart, Efergy es un fabricante de referencia, el cual dispone de un medidor de energía online que muestra el consumo energético en tiempo real además de proporcionar una interfaz completa y visual. Se trata de su producto Engage [10] el cual se forma por una pinza amperimétrica, que requiere que se pinche el cable de alimentación del cuadro eléctrico, la cual está conectada al transmisor de datos en tiempo real.



Figura 5. Efergy Engage Hub Pack

La empresa Wattio [11] también desarrolla productos para solucionar los problemas de eficiencia energética. Uno de los sistemas más completos del mercado es su pack “Wattio Energy”. Se trata de un sistema compuesto por varios dispositivos que recogen datos de diferentes fuentes y son gestionados a través de una centralita. Incluye:

- Gate: centralita domótica de pantalla táctil
- Thermic: termostato inteligente
- Pod: enchufe inteligente
- Bat: monitor de electricidad

Lo interesante es que se trata de un sistema ampliamente flexible, con la posibilidad de incorporar nuevos dispositivos con otras funcionalidades, tales como domótica o seguridad.



Figura 6. Pack Energy de Wattio

Es también interesante comentar los proyectos DIY (hazlo tú mismo) que existen en la actualidad respecto a este sector. Uno de los más importante es OpenEnergyMonitor [12] el cual ofrece una plataforma para el fácil aprendizaje donde cada uno puede crear su propio medidor de consumo inteligente. Es un producto totalmente de código libre enfocado para su configuración con Arduino. Se facilitan las instrucciones detalladas de cada uno de los componentes de hardware o software del sistema, por ello, es una opción que valorar si se tienen ciertas habilidades en este sector.

Capítulo 3

Descripción funcional

El principal objetivo de este proyecto consiste en ofrecer un sistema robusto capaz de obtener mediciones de consumo energético en tiempo real para que cualquier persona pueda hacer uso de este dispositivo. Se desarrollará a través de un sistema sencillo con una interfaz Web amigable para facilitar su funcionamiento. Esto será posible gracias a la conexión con IoT del dispositivo que enviará los eventos que se visualizarán a través de esta interfaz. A continuación, se describen las diferentes partes del sistema de forma funcional.

3.1 Diseño del sistema

Este sistema está compuesto tanto por elementos de hardware como de software. A continuación, se muestra la configuración general del sistema:

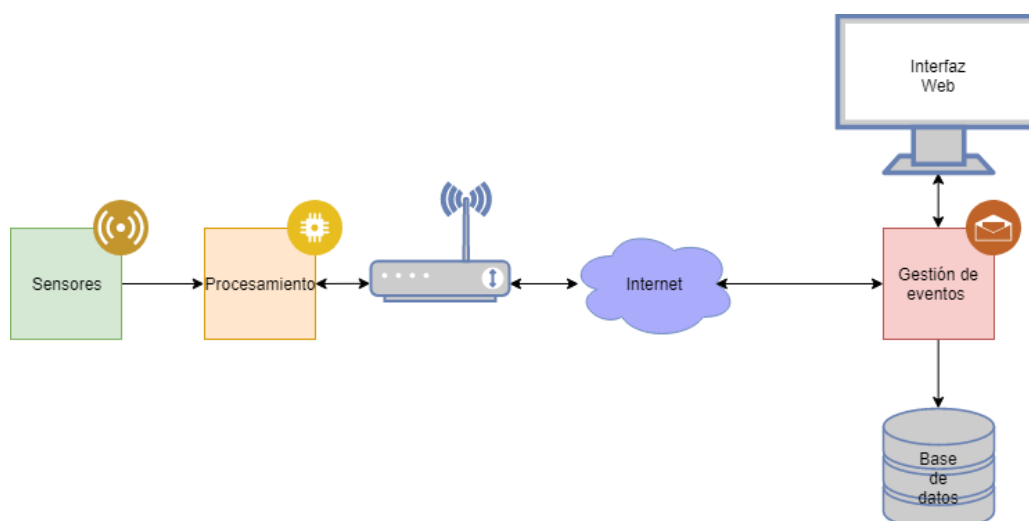


Figura 7. Diagrama de funcionamiento general del sistema

De forma general, como se ve en la Figura 7, se ha establecido un flujo entre los diferentes bloques que lo componen. El bloque de sensores es el encargado de obtener las distintas mediciones procedentes del sensor de voltaje y de corriente, este enviará la información al bloque de procesamiento de datos que corresponde con el microcontrolador MSP432. El bloque de procesamiento de datos es el encargado de solicitar esta información a los sensores para posteriormente procesarla y realizar las tareas correspondientes. Como no dispone de wifi integrado, se incorpora el módulo wifi CC3100 quien le proporciona conectividad con Internet, de esta forma, se consiguen transmitir los datos de manera inalámbrica.

Una vez enviados estos datos al bloque de Internet, es el Broker quien actúa como *gateway* y los redistribuye al bloque de gestión de eventos donde se encuentra Node-Red, encargado de dar formato a los mensajes recibidos para almacenarlos en la base de datos Cloudant Db y mostrarlos de forma visual en la interfaz Web Node-Red Dashboard.

A través de las flechas se establece el sentido de la comunicación del sistema. Como se observa, se ha diseñado un sistema para permitir una comunicación bidireccional, es decir, de dos hilos, lo que hace posible la configuración por parte del usuario para el establecimiento de ciertos parámetros. Será el bloque de interfaz Web el encargado de obtener estos parámetros y enviarlos al bloque de gestión de eventos donde se gestionarán para reenviarlos a través del Broker de nuevo al microcontrolador, quien actuará ante estos cambios. No es necesario establecer una comunicación bidireccional con la base de datos ni con los sensores ya que estos solamente tienen una única función: enviar o recibir información.

3.1.2 Protocolos de comunicación

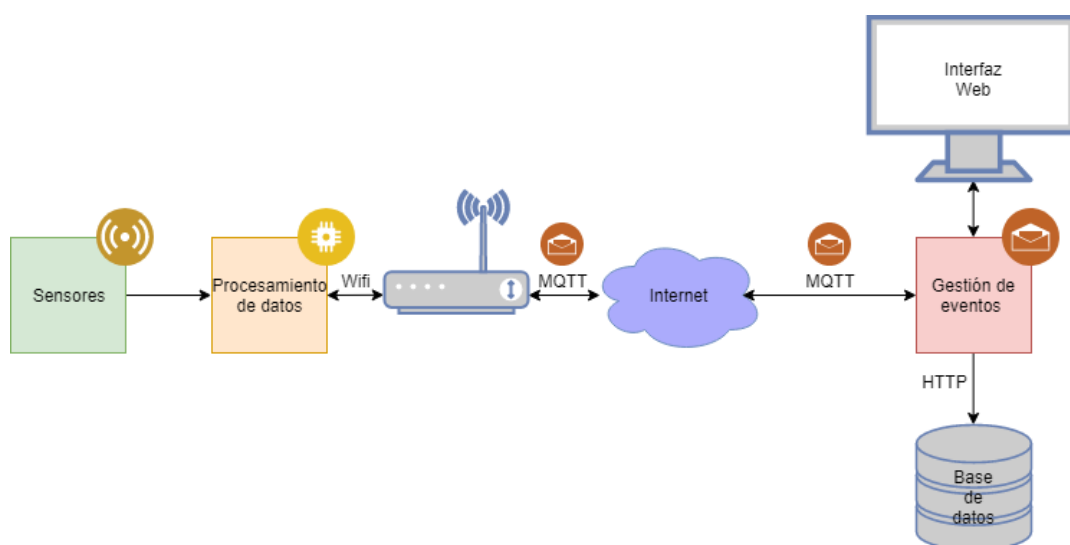


Figura 8. Protocolos de comunicación

Se han implementado tres protocolos de comunicación en el sistema:

- **MQTT (Message Queue Telemetry Transport):** es el protocolo principal de comunicación del sistema, está ampliamente integrado en el Internet de las Cosas (IoT) ya que su comunicación se basa en la estructura máquina-a-máquina (M2M), orientado a la comunicación de sensores, debido a que consume muy poco ancho de banda y puede ser utilizado en la mayoría de los dispositivos empotrados con pocos recursos.
- **HTTP (Hypertext Transfer Protocol):** es un sencillo protocolo cliente-servidor que articula los intercambios de información entre los clientes Web y los servidores HTTP. Será el que emplee Node-Red para comunicarse con la base de datos y de esta forma, almacenar la información.
- **WiFi:** se trata de la tecnología que permite la interconexión inalámbrica del dispositivo con el router a través del estándar 802.11.

3.2 Diseño del software del dispositivo

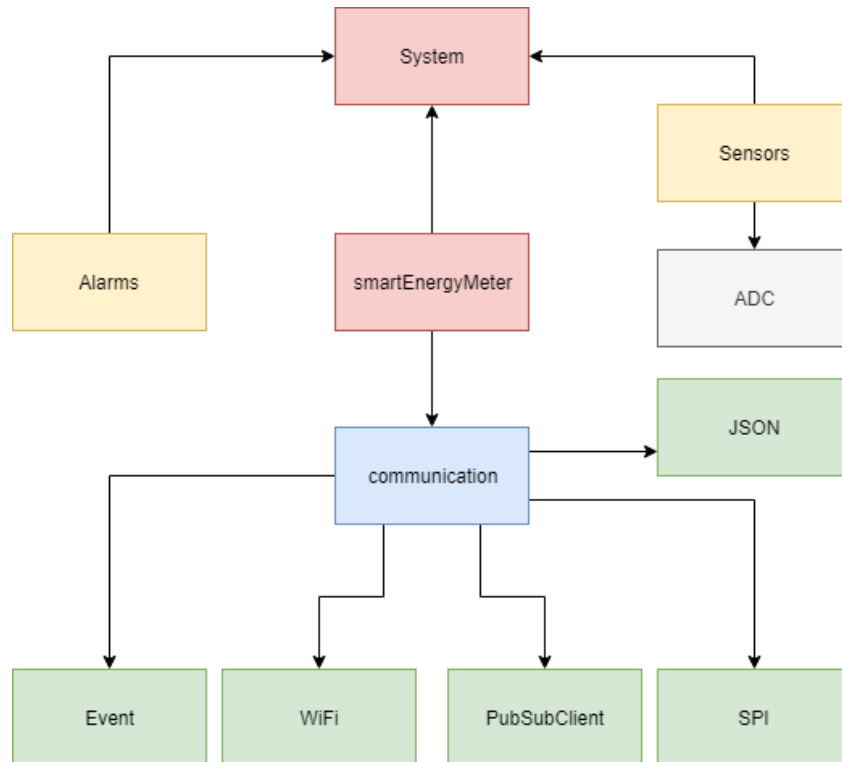


Figura 9. Diagrama de bloques del Software

Como se puede ver en la Figura 9, el diagrama representa la relación entre los distintos bloques del programa, además, se incluye el enlace del bloque de sensores con el *driver*, conversor analógico-digital (ADC), ya que es fundamental para que el sistema pueda procesar los valores obtenidos por los sensores. A continuación, se comenta brevemente cada uno de estos bloques ya que en el próximo capítulo se tratarán con mayor detalle.

- **smartEnergyMeter:** en esta sección del programa se cargan todas las librerías que necesitará el Firmware del dispositivo para que después cada tarea use la que le corresponda.
- **Communication:** es el bloque encargado de establecer tanto la conectividad Wifi como la comunicación MQTT para el envío o recepción de eventos. Hace uso de las librerías Event, Wifi, PubSubClient, SPI y JSON para este fin.
- **Alarms:** este bloque del programa es el encargado de procesar todos los parámetros referentes a las alarmas para actuar frente a cualquier situación de emergencia.
- **Sensors:** recoge las diferentes medidas de los sensores de corriente y voltaje.
- **System:** es el bloque que alberga los métodos getter y setter de cada parámetro para que estén disponibles en cualquier tarea.

3.2.1 Aplicación de uso

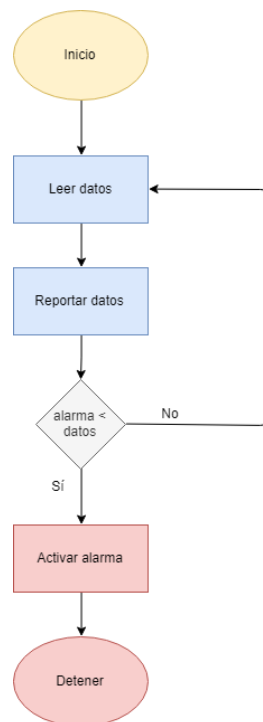


Figura 10. Caso de uso de ejecución de software

El diagrama de flujo anterior representa la ejecución del programa. Este estará continuamente leyendo hasta que se cumpla la condición $alarma < datos$, esto significa que, si hay alguna alarma establecida, ante una lectura de datos mayor se activará la alarma y, por lo tanto, se detendrá el programa.

3.3 Diseño de la aplicación Web

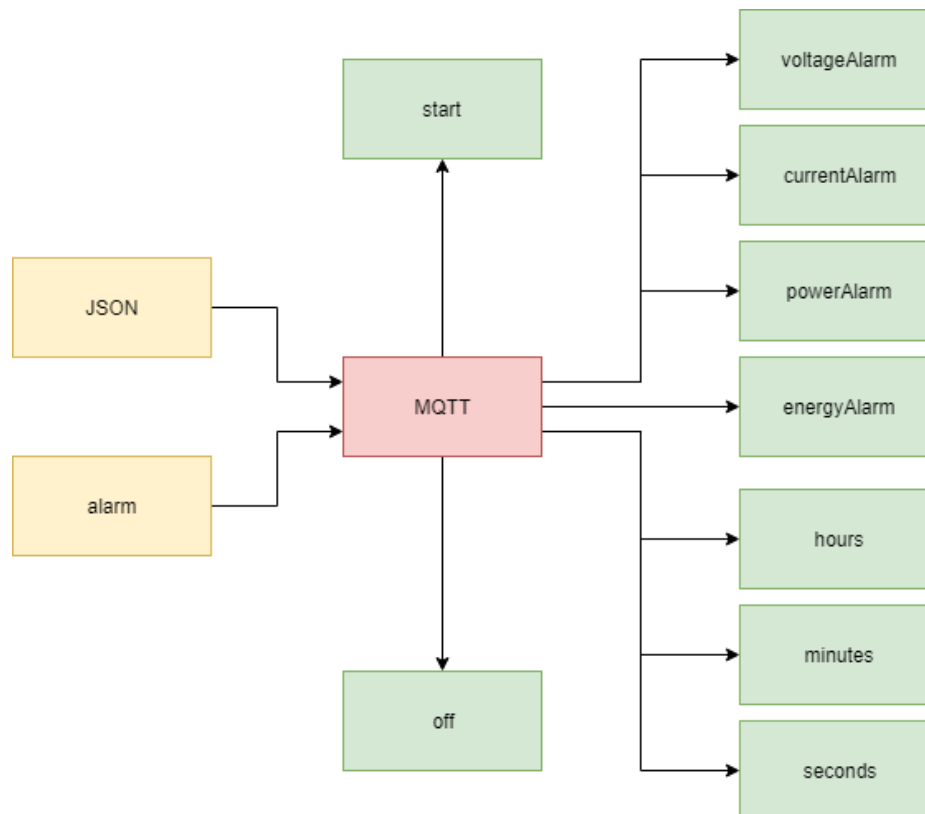


Figura 11. Configuración Web

La aplicación Web se estructura en eventos, estos eventos pueden ser de publicación o suscripción. En la **¡Error! No se encuentra el origen de la referencia.** se pueden identificar según el sentido de las flechas hacia el bloque MQTT:

- **Suscripción:** son los bloques de color amarillo, con el sentido de la flecha hacia el bloque MQTT. Estos son los eventos que publicará el dispositivo de hardware y a los que se suscribirá la plataforma de desarrollo Web Node-Red.
- **Publicación:** son los eventos que se generan en la plataforma Web. Se identifican con una flecha que sale del bloque central MQTT. Estos eventos serán a los que el dispositivo de hardware se suscribirá para actuar frente a las acciones correspondientes.

3.3.1 Aplicación de uso

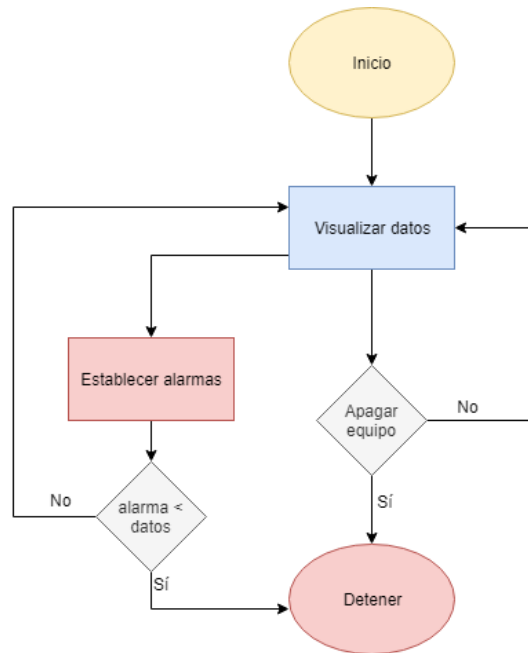


Figura 12. Caso de uso de la aplicación Web

La Figura 12 representa un caso de uso de la aplicación Web. Como se muestra, se estarán visualizando datos hasta que se dé la orden de apagar el equipo o se establezcan las alarmas y estas sean menores que los datos.

3.4 Propuesta de comunicaciones

Como se puede observar en la Figura 8, MQTT es el protocolo principal de comunicación del sistema. Es quien hace posible la comunicación bidireccional entre el dispositivo y la interfaz Web.

Este proyecto se basa fundamentalmente en el Internet de las cosas (IoT), esto requiere de la capacidad de envío y recepción de información en función de unas determinadas funciones. A medida que esta tecnología evoluciona, también deben hacerlo los protocolos mediante los cuales opera. Se pueden emplear varios protocolos para este fin, tales como AMQP, CoAP, MQTT y REST sobre HTTP, pero los principales para trabajar con IoT y que se han valorado en la realización de este proyecto son estos dos últimos: MQTT y REST sobre HTTP.

La mayoría de las implementaciones de IoT utilizan REST a través de la conectividad basada en HTTP de cliente-servidor, pero este tiene su propia limitación que aparece mientras su solución se amplía a mayor cantidad de dispositivos y más traducciones por segundo. MQTT, es un protocolo ligero diseñado exclusivamente para IoT y tiene su ventaja sobre REST.

Los principales beneficios de MQTT son la ligereza, transfiere datos como una matriz de bytes, y el modelo de publicación-suscripción, el cual lo hace perfecto para dispositivos con recursos limitados además de ayudar en el ahorro de batería. Este modelo de comunicación proporciona a los clientes una existencia independiente entre sí y mejora la confiabilidad de todo el sistema. Cuando un cliente está fuera de servicio, todo el sistema puede seguir funcionando correctamente.

Si los comparamos, MQTT está siempre conectado gracias a la conexión permanente en contra de las llamadas REST intermitentes que requiere la necesidad de conectarse y desconectarse para cada transferencia de datos. El ping *keep-alive* de MQTT hace que tenga una sobrecarga mucho menor en comparación con las llamadas de conexión y reconexión que realiza REST. Esto también provoca que REST consuma más energía ya que esta se pierda en los recursos utilizados. Además, MQTT puede transferir datos a una velocidad de 20 a 25 veces más rápida que las llamadas REST.

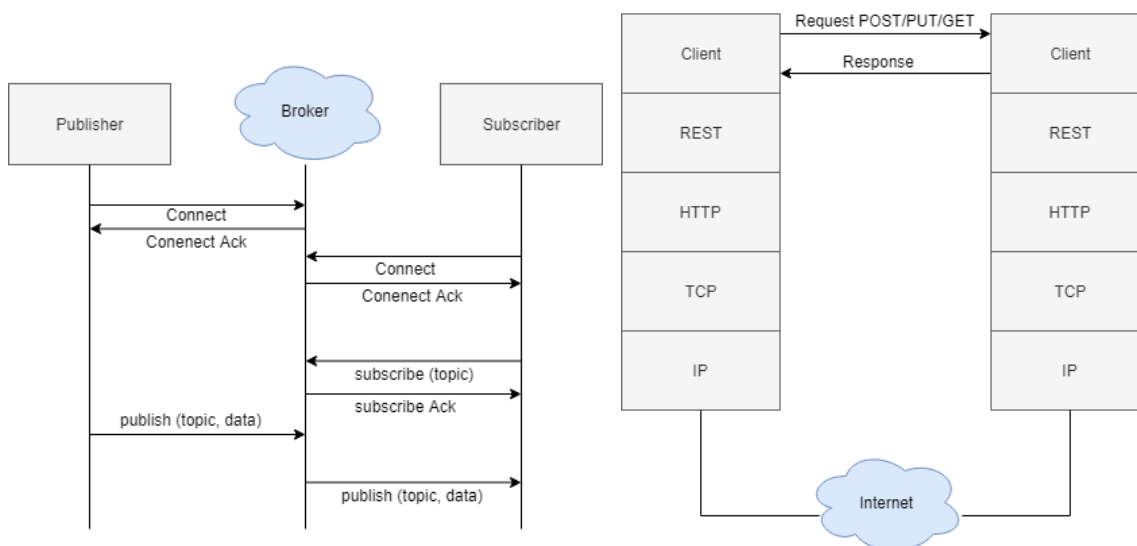


Figura 13. Protocolo MQTT y protocolo REST

Por estos motivos se ha elegido MQTT. Es un protocolo sencillo de usar, esencial cuando el tiempo de respuesta, el rendimiento, el menor uso de la batería y el ancho de banda están en primer lugar para futuras soluciones y también es perfecto en caso de conectividad intermitente.

3.5 Propuesta de hardware

Existe una amplia variedad de microcontroladores disponibles en el mercado, cuál elegir depende de la aplicación. La selección de componentes de ingeniería se trata de evaluar qué opción se adapta mejor a los requisitos y restricciones de la aplicación en particular. Para el propósito de este proyecto se requiere de un acoplamiento rápido con prototipos y un rendimiento eficiente, por ello se ha elegido el MSP432P401R, no solamente por las posibilidades que ofrece, sino por el gran soporte en los foros de TI y la posibilidad de usar Composer Studio.

Aunque la otra alternativa era Arduino, el cual puede ser bastante sencillo de programar gracias a la gran cantidad de APIS disponibles, el MSP432 tiene ventaja cuando se requiere de un mejor control de hardware. Con la ayuda de CCS, no solamente se puede acceder a su espacio de direcciones, sino que también se pueden cambiar los valores de diferentes registros que afectarán de manera adecuada a las diferentes configuraciones.

Respecto a los otros componentes de hardware, al haber elegido el microcontrolador MSP432, se ha usado el módulo Wifi CC3100 y no se han valorado otros, ya que ese integra de forma sencilla. Por otro lado, los sensores se han elegido según las características necesarias del proyecto.

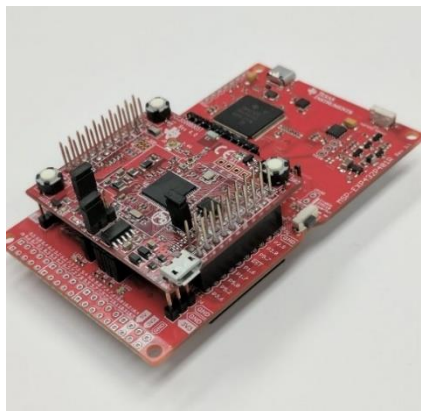


Figura 14. MSP432P401R y CC3100

Capítulo 4

Descripción detallada

4.1 Diseño de Hardware

Según se ha visto en la tabla 1, en la primera fase del proyecto se ha llevado a cabo el desarrollo del hardware. El diseño detallado de todas las partes del hardware es el siguiente:

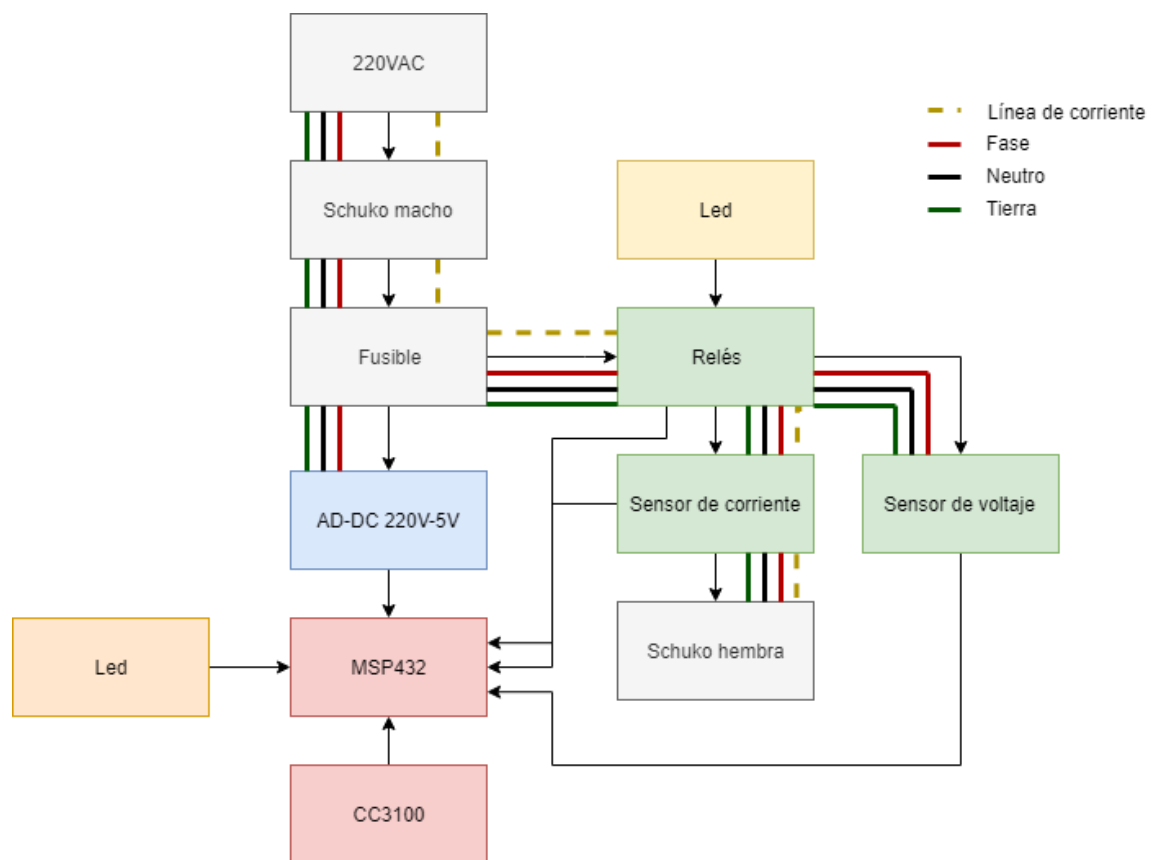


Figura 15. Diagrama de bloques del diseño de hardware

4.1.1 Circuito del sensor de corriente

Para realizar el diseño del circuito del sensor de corriente, en primer lugar, se ha elegido un sensor que cumple con los requisitos deseados en el sistema. El CST-1020 se trata de un transformador de corriente inductivo que puede obtener mediciones de hasta 20A, su salida es en tensión y trabaja perfectamente en los 50-60Hz.



Figura 16. Sensor de corriente inductivo CST-1020

Los sensores de corriente son realmente transformadores de corriente, similares a los transformadores de tensión y están basados en los mismos principios, aunque su finalidad es distinta. A través del sensor de corriente circulará una corriente por el devanado primario, el cual corresponde al cable del aparato a medir, y gracias al efecto de la inducción magnética, en el devanado secundario se producirá una intensidad de corriente proporcional a la que ha pasado por el primario.

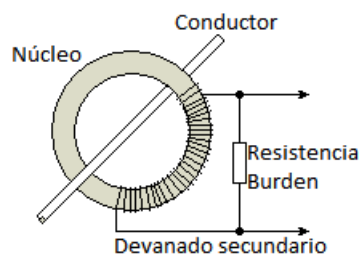


Figura 17. Inducción electromagnética

Este tipo de transformadores, a diferencia de los de tensión, requieren que el circuito secundario permanezca cerrado, en caso contrario, las corrientes inducidas podrían dañar el dispositivo que lo atraviesa, por ello es necesaria una resistencia de protección, denominada resistencia Burden o de carga, que podemos ver en la Figura 17.

Como se está trabajando en alterna, la señal del voltaje que obtenemos del sensor de corriente es alterna, es decir, va alternando su valor de salida entre valores positivos y negativos, obteniendo una señal sinusoidal. Para que la entrada analógica del microcontrolador MSP432 sea capaz de trabajar con esta señal, se ha de adaptar el rango de tensiones en la salida del sensor. Esto se consigue con el cálculo de la resistencia de carga además de desplazar la señal con el fin de obtener voltajes positivos en un rango de 0-5V.

Para el cálculo de esta resistencia, se ha consultado la hoja de características del fabricante donde se muestra la curva de respuesta del sensor [13]:

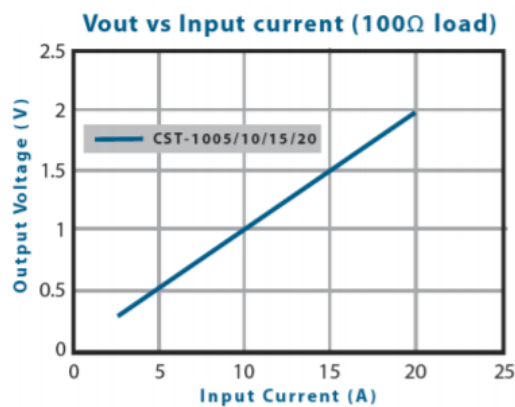


Figura 18. Curva de respuesta del sensor de corriente CST-1020

Como se observa, para una resistencia de carga de 100Ω tenemos una salida en tensión de 2V para la corriente máxima de 20A. Se ha de tener en cuenta que al hablar de tensión alterna normalmente se emplean valores RMS, por ello, necesitamos calcular el Valor de pico:

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

$$V_{picoapico} = 2 \cdot \sqrt{2} \cdot V_{RMS}$$

Si seguimos la hoja de características, con una resistencia de 100Ω tenemos una tensión de pico de $\pm 2.82V$ y una tensión de pico a pico de $5.65V$, esto complicaría la adaptación de la tensión a la salida del sensor ya que superamos los 5V, por ello, se decide emplear una resistencia menor, en este caso de 50Ω con la que conseguimos una salida en tensión de 1V para una corriente de 20A:

$$V_{pico} = \sqrt{2} \cdot 1 = \pm 1.41V$$

$$V_{picoapico} = 2 \cdot 1.41 = 2.82 \text{ V}$$

Tras calcular la resistencia de carga, se ha de adaptar la señal de salida ya que, como se ha comentado anteriormente, solamente se puede trabajar con valores positivos. Esto se consigue a través de un circuito de Offset DC de 2.5V, configurado con un divisor de tensión formado por dos resistencias de 470kΩ y un condensador de 10μF, este proporcionará un punto medio entre GND y Vcc (+5V).

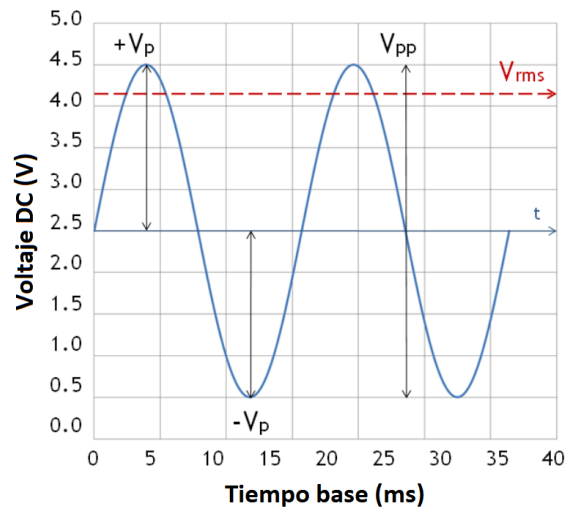


Figura 19. Offset DC de 2.5V

4.1.2 Circuito del sensor de voltaje

Para el diseño de este circuito, se procederá de forma similar al apartado anterior, ya que, como se ha visto, los transformadores de corriente son similares a los transformadores de tensión y están basados en los mismos principios. Se empleará un transformador de 230VAC/18VAC para realizar las mediciones de la red eléctrica.



Figura 20. Transformador 230VAC/18VAC

Un transformador de tensión alterna hace uso de campos magnéticos para la adaptación de niveles de tensión entre dos circuitos, de esta forma, se puede adaptar la tensión del circuito al rango en el que puede trabajar el microcontrolador MSP432. Una ventaja de su uso es el aislamiento galvánico que posee, aunque sigue siendo susceptible a los picos de corriente, por ello, se incorporará un condensador a la salida del secundario.

El principio de funcionamiento de los transformadores se basa en el fenómeno de la inducción electromagnética [14]. La corriente que lo atraviesa produce una fuerza electromotriz en el devanado primario, lo que produce la inducción de flujo magnético en su interior. En el devanado secundario se genera una tensión inducida (o fuerza electromotriz). De este modo, ambos circuitos quedan acoplados mediante un campo magnético.

El flujo que se ha generado por el devanado primario es proporcional a su número de espiras y la tensión inducida en el devanado secundario también es proporcional a su número de espiras, por consiguiente, la relación entre ambas tensiones es proporcional a la de las espiras de ambos devanados. Dicha relación se denomina relación de transformación.

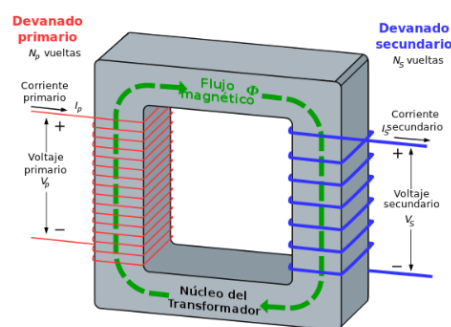


Figura 21. Transformador monofásico ideal

Se ha de tener en cuenta que los transformadores reales tienen limitaciones y pérdidas, aunque su rendimiento alcanza fácilmente un 95%. Además, generan distorsiones en la señal transmitida debido a la falta de linealidad del material del que está compuesto el núcleo, sobre todo en zonas de saturación, esto provoca que se introduzcan componentes armónicos en la salida.

La tensión de la red eléctrica monofásica en España es de 230V 50Hz, esta tensión tiene una tolerancia del 10% por lo que esta podrá oscilar entre 210-247V. Como son valores RMS, si

se calcula tanto la tensión de pico como la de pico a pico, de igual forma que en el apartado anterior, tenemos que:

$$V_{pico} = \sqrt{2} \cdot 247 = 349.31 V$$

$$V_{picoapico} = 2 \cdot 349.31 = 698.62 V$$

A través del transformador de 220VAC/18VAC adaptamos este rango de tensión. Ahora se tendrá a la salida del transformador una tensión de pico de 25.45V y una tensión de pico a pico de 50.91V. Estos valores son muy elevados por ello, hay utiliza un divisor de tensión y conseguir reducir esta tensión de pico a pico. Un divisor de tensión es una configuración de un circuito eléctrico que reparte la tensión de una fuente entre una o más impedancias conectadas en serie [15], en el diseño de este proyecto, se ha utilizado un divisor de tensión resistivo, donde:

$$V_{out} = \frac{R_2}{R_1 + R_2} \cdot V_{in}$$

Se han elegido dos resistencias en serie de valores: $R_1 = 100k\Omega$, $R_2 = 10k\Omega$, si se realiza el cálculo anterior:

$$\frac{10 \cdot 10^3}{100 \cdot 10^3 + 10 \cdot 10^3} \cdot 50.91 = 4.42V$$

Por lo que obtendremos una tensión de pico a pico de 4.42V. Como se está trabajando con valores en alterna hay que desplazar esta tensión para que esté dentro del rango de valores con los que puede trabajar el microcontrolador. Se utilizará el mismo diseño que para el sensor de corriente del offset DV de 2.5V, la señal resultante es:

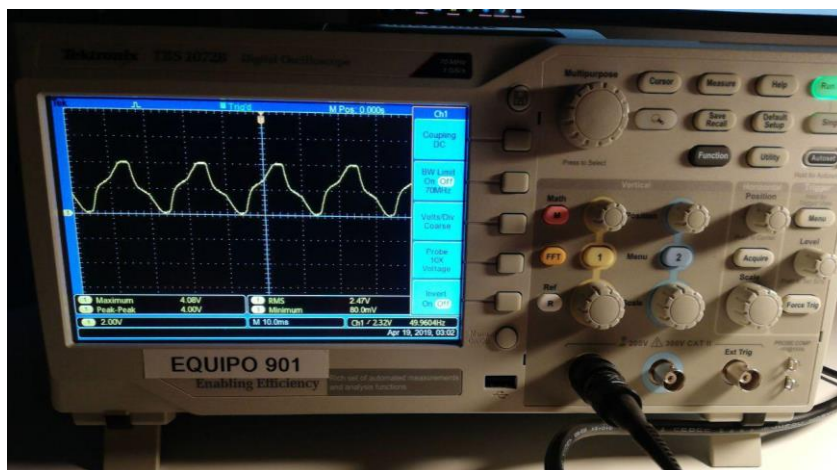


Figura 22. Señal de tensión adaptada

4.1.3 Circuito de los relés de protección

Un relé funciona como un interruptor controlado por un circuito eléctrico en el que, por medio de una bobina y un electroimán, se acciona un juego de uno o varios contactos que permiten abrir o cerrar otros circuitos eléctricos independientes [16].

En este diseño se han implementado dos relés para cortar tanto fase como neutro como medida de protección. Se ha empleado un módulo comercial con su circuito integrado donde no ha sido necesario realizar ninguna modificación.



Figura 23. Módulo de dos relés de 5VDC

El diseño final de la placa de sensores y relés se puede ver en la siguiente imagen:

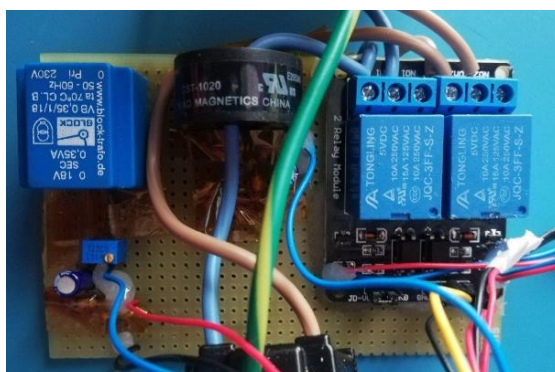


Figura 24. Placa de sensores y relés

4.1.4 Diseño general del dispositivo

Como se ha comentado en la propuesta de hardware del capítulo anterior, todo el proyecto está basado en el microcontrolador MSP432P401R y el módulo Wifi CC3100. Estas placas serán el núcleo central del diseño del proyecto ya que a través de ellas se recogerán los valores obtenidos por los diferentes sensores, además proporcionarán la conectividad con Iot.

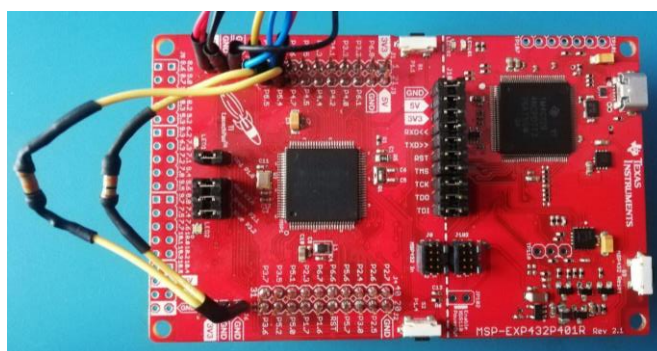


Figura 25. Cableado de los sensores y relés al MSP432P401R

El resultado final de la parte de hardware es el siguiente:

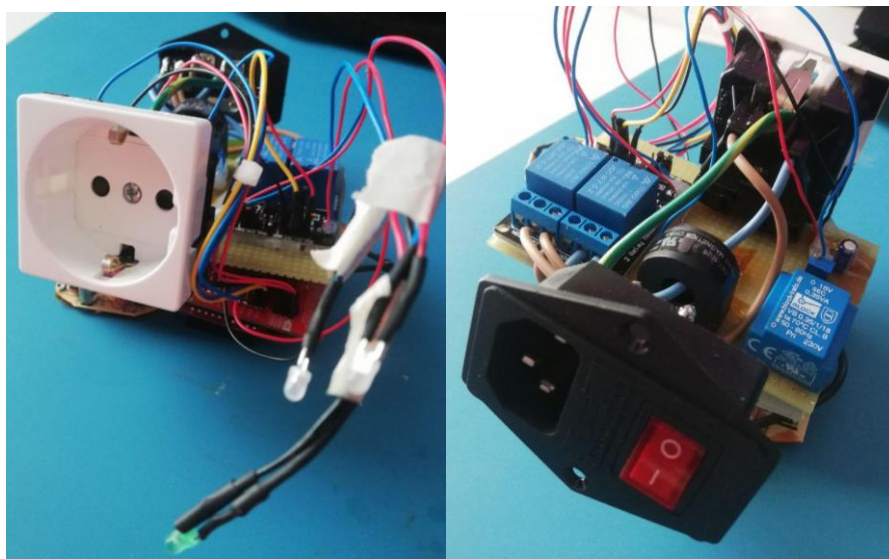
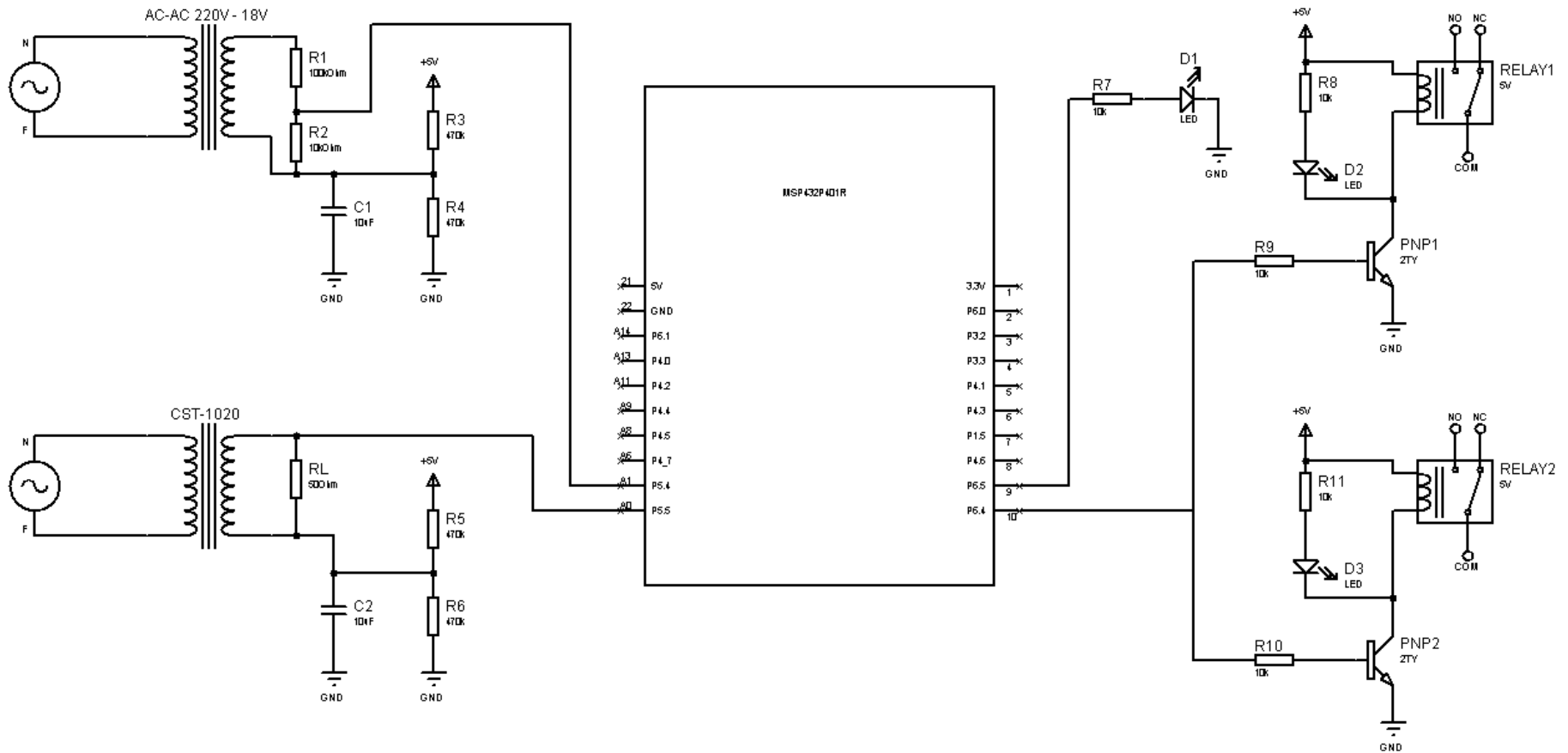


Figura 26. Visualización interior del dispositivo



Figura 27. Producto final

A continuación, se adjunta el esquemático electrónico de todas las partes diseñadas anteriormente, así como la conexión detallada al MSP432P401R:



FILE NAME:	Esquema_electrónico.pdsprj	DATE:	06/06/2019
DESIGN TITLE:	Esquema electrónico	PAGE:	1 of 1
PATH:	D:\Esquema_electrónico.pdsprj	TIME:	11:49:34
BY:	Esther Gasco Marco	REV:	2

Figura 28. Esquema electrónico

4.2 Diseño de Software

En la Figura 9 y Figura 11 se ha visto el diagrama de bloques funcional del diseño de software, tanto de la parte del dispositivo físico como de la Web, ambos están enlazados a través de eventos mediante la figura intermediaria del *bróker*. Antes de pasar a explicar detalladamente cada una de estas partes de Software, se dará una visión global del protocolo MQTT y su implementación en el sistema.

4.2.1 Estructura del protocolo MQTT

Se establece la comunicación MQTT a través del puerto TCP/IP 1883 configurado en el código del programa. Debido a que este protocolo desacopla al publicador del suscriptor, las conexiones del cliente siempre son gestionadas por un bróker [17]:

- Cliente: tanto los editores como los suscriptores son clientes de MQTT donde las etiquetas del editor y del suscriptor hacen referencia a si el cliente está publicando mensajes o suscribiéndose a ellos, ambas funcionalidades se pueden implementar en el mismo cliente. Un cliente MQTT es cualquier dispositivo que ejecuta una biblioteca MQTT y se conecta a un bróker MQTT a través de una red. En el caso de este proyecto, el cliente es la parte física del analizador inteligente de consumo energético.
- Broker: es el eje central de cualquier protocolo de publicación/subscripción. Dependiendo de su configuración este podría manejar miles de clientes MQTT conectados al mismo tiempo. Es el encargado de recibir todos los mensajes, filtrarlos y determinar quién está suscrito a cada uno de ellos para posteriormente enviarlos a estos. Otra de sus funciones es facilitar la autenticación, autorización e integración personalizadas en los sistemas back-end. Es importante que el bróker sea altamente escalable, se pueda integrar en sistemas de back-end, que sea fácil de monitorear y tolerante a fallos. Eclipse Paho cumple con estos requisitos, será el bróker que se empleará en el proyecto.

La conexión MQTT siempre es entre un cliente y el intermediario (bróker) donde los clientes nunca se conectan directamente entre sí. Cuando se quiere iniciar una conexión el cliente envía un mensaje CONNECT al bróker donde este responde con un CONNACK y un código de estado. Una vez que se establece la conexión, el bróker la mantiene abierta hasta que el cliente envía un comando de desconexión o se interrumpe la conexión.

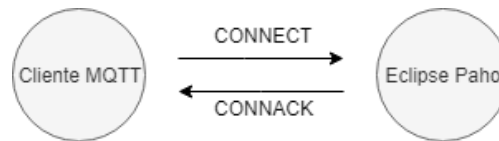


Figura 29. Conexión cliente - bróker

4.2.1.1 Publicación y subscripción a mensajes

En el momento que se conecta el analizar inteligente de consumo energético al bróker, puede empezar a publicar mensajes. MQTT hace uso del filtrado basado en temas (topics), por ello, cada mensaje debe contener un tema que el bróker utilizará para reenviar a los suscriptores. Estos mensajes contienen una carga útil con los datos a transmitir en formato byte donde es el cliente que publica quien decide el tipo de datos a enviar por ello en este proyecto se ha empleado el formato JSON.

Para resolver este desarrollo, se crea un código que convierte todos los datos que se quieren enviar a través de MQTT en un solo mensaje JSON, este formato de texto consiste colecciones no ordenadas de pares de la forma <nombre>:<valor> [18] modeladas como un objeto, es decir, en un formato de árbol. Este objeto JSON tendrá un *topic* asociado que será el mensaje que se publica para, posteriormente, a través de la aplicación Web, filtrarlo por su contenido (*payload*) y visualizar cada parámetro gráficamente.

Por otro lado, el mismo cliente puede subscribirse a topics generados por otros clientes, en este caso, el otro extremo de la conexión es Node-Red, donde es el usuario final quien publica los valores de las alarmas relacionadas con los distintos parámetros, temporización o la orden de apagado. Cada una de estas publicaciones tiene un topic asociado, donde el dispositivo se subscribe.

Después de que un cliente envíe el mensaje SUBSCRIBE y recibe el mensaje SUBACK, este recibirá todos los mensajes publicados que coinciden con los topics de las subscripciones que contenía el mensaje de SUBSCRIBE [19], esto se ve representado en la Figura 30.

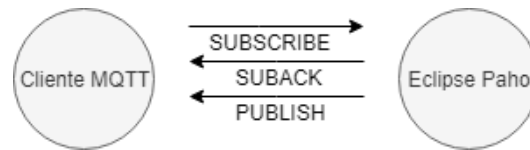


Figura 30. Suscripción a mensajes

4.2.2 Desarrollo del software del dispositivo

Se ha realizado la programación del dispositivo físico con el lenguaje de programación de Arduino, basado en C++ y Java, ya que, gracias a Energia IDE es compatible con el microcontrolador MSP432.

En la Figura 9 se ha visto una descripción funcional del sistema. Internamente, este sistema está formado por distintas tareas relacionadas con una librería principal que alberga las variables globales necesarias para el funcionamiento del programa.

Las tareas en las que se descompone el Software son las siguientes:

- **Tarea de lectura de los sensores:** es la encargada de recibir los valores de las entradas analógicas del microcontrolador y procesarlos. Se ha programado un bucle (*loop*) por cada una de las lecturas: voltaje y corriente. Esta función es la que se ejecutará un número infinito de veces mientras el microcontrolador esté encendido e irá calculando instantáneamente el valor correspondiente, además, dentro de esta tarea se realizarán los cálculos para la obtención de la potencia y el consumo energético. La ecuación para obtener la potencia es:

$$P = V \cdot I \text{ (W)}$$

El consumo energético dependerá de la potencia del equipo que se esté midiendo y del tiempo que se encuentre conectado.

- **Tarea de alarmas:** es la tarea encargada de proteger al dispositivo, en ella se encuentra la activación de los relés frente a las alarmas establecidas por el usuario o, las predefinidas. Esto se consigue comparando el valor instantáneo obtenido en cada

muestra con el que se establezca. Si el valor que se está midiendo es superior al establecido, se activará el relé, en caso contrario, el sistema seguirá funcionando. Cuando salta una alarma, se ha programado de tal forma para que sea necesario apagar físicamente el analizador para poder seguir utilizándolo.

- **Tarea de comunicaciones:** responsable de gestionar tanto publicaciones como suscripciones. En ella se han programado todos los topics a los que se suscribe el dispositivo para realizar las acciones correspondientes: activar alarmas, establecer tiempo de medición y apagado remoto. También enviará las publicaciones que recogen los diferentes parámetros de medición. Por otro lado, hace posible la conexión Wifi con la red preconfigurada en el código.

Además, este código recoge todas las librerías necesarias, que se han comentado en el capítulo anterior, para el correcto funcionamiento del sistema. Estas son:

- SPI: permite la comunicación entre las dos placas que componen el sistema, MSP432 y CC3100. De esta forma el módulo Wifi queda completamente integrado en el sistema facilitando su configuración a través de la placa de desarrollo principal.
- PubSubClient: permite el envío y recepción de mensajes a través del protocolo MQTT. La comunicación se basa en topics (temas), que el cliente que publica o se suscribe a ellos.
- Event: es la encargada de gestionar los eventos dentro del programa.
- JSON: da formato a los parámetros que se van a enviar a través de MQTT. Se trata de un formato basado en texto estándar para la representación de datos estructurados en la sintaxis de objetos de JavaScript.
- WiFi: permite la comunicación inalámbrica del dispositivo.
- System: es la librería principal del sistema, donde se almacenan todas las variables globales para poder ser utilizadas en el resto del programa. Proporciona los diferentes métodos getter y setter para cada una de ellas.

4.2.3 Desarrollo de la plataforma Web

Como se ha comentado anteriormente, para la implementación Web se ha empleado Node-Red el cual utiliza un enfoque de programación visual que permite conectar bloques de

código predefinidos, conocidos como nodos, para realizar una tarea. Los nodos conectados, generalmente una combinación de nodos de entrada, nodos de procesamiento y nodos de salida, forman un flujo

Resulta interesante para entender con detalle este apartado, describir cada una de las partes que se han configurado en Node-Red.

Se ha creado un flujo para cada una de las funciones necesarias:

- **Flujo de lectura de datos de los sensores:** es el encargado de recibir el mensaje JSON y filtrarlo para obtener las distintas lecturas de los sensores. La programación se basa en nodos, donde cada uno de ellos responde a una función determinada. También, es posible codificar parte de estos nodos si queremos que se realice alguna tarea concreta. En este caso, el nodo json se trata de un nodo función, en él se ha programado mediante JavaScript el mensaje recibido en formato JSON para poder procesarlo por Node-Red.

Como se observa en la imagen, el nodo JSON es la subscripción a la publicación que se ha realizado desde el dispositivo del mensaje JSON. Como salida tenemos los nodos que forman la interfaz gráfica. Se representará gráficamente cada una de las medidas obtenidas.

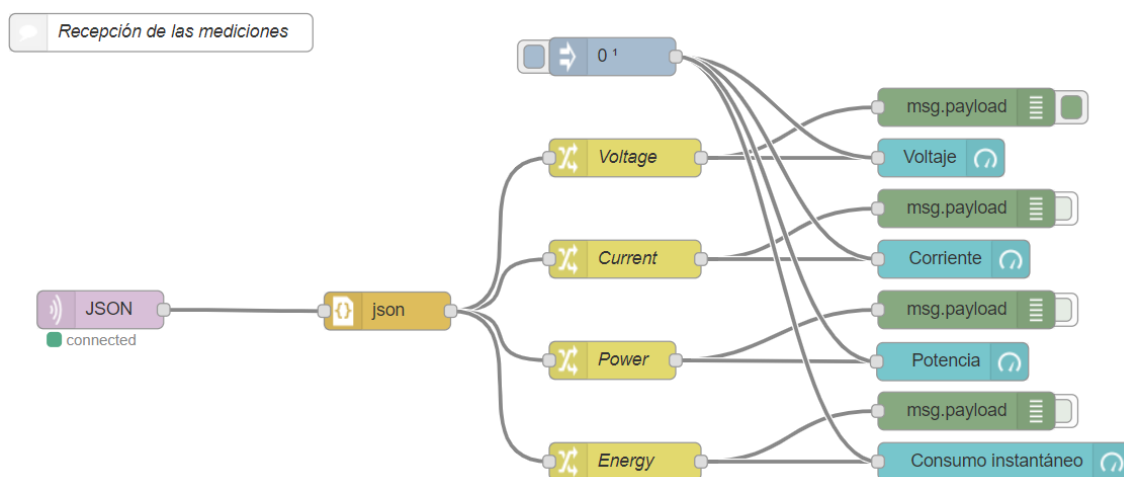


Figura 31. Flujo de recepción de mediciones

- **Flujo de establecimiento de alarmas:** a través de la interfaz gráfica, será el usuario quien elija los parámetros de alarma. Estos se publican desde Node-Red, cada uno de ellos con un topic asociado. Como se puede ver en la imagen, los nodos de publicación están en estado conectado, esto quiere decir que están conectados al bróker Eclipse Paho. El dispositivo se suscribirá a estos topics para actuar.

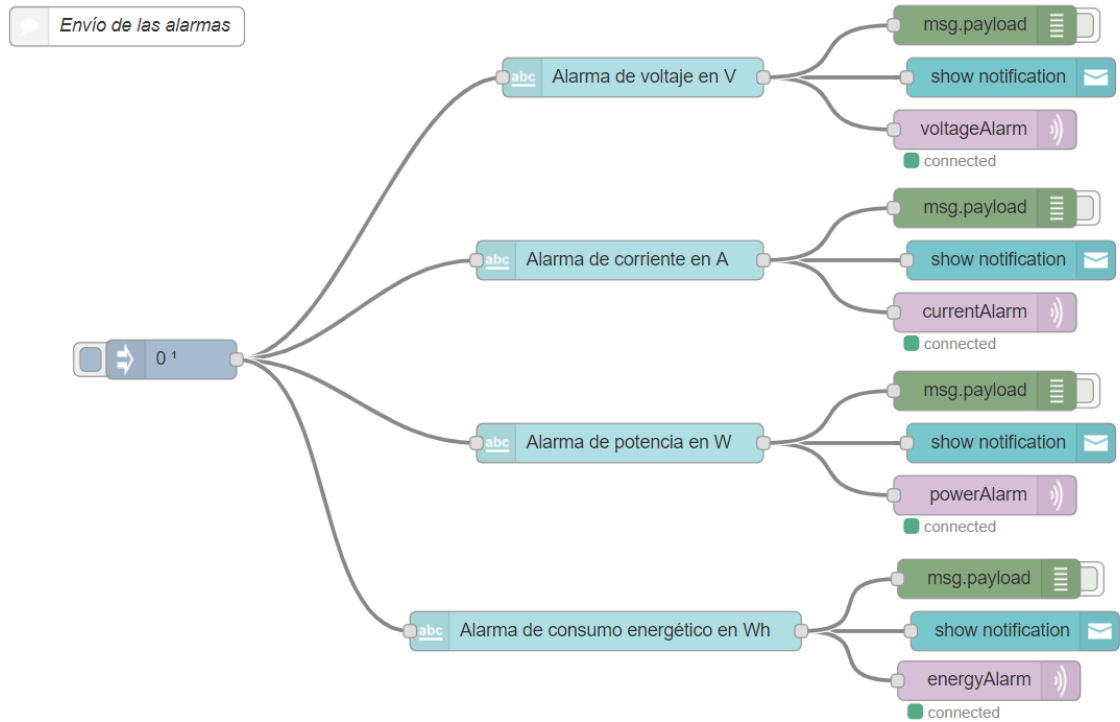


Figura 32. Flujo de envío de alarmas

- **Flujo de apagado remoto:** se ha configurado un botón en la interfaz gráfica que facilita el apagado remoto del dispositivo. Este botón se utilizará como mecanismo de protección y actuará de la misma forma como si se estableciera una alarma y el dispositivo tuviera que actuar frente a ella. Es decir, al enviar el topic off del botón, se activarán los relés que cortarán la tensión del dispositivo.

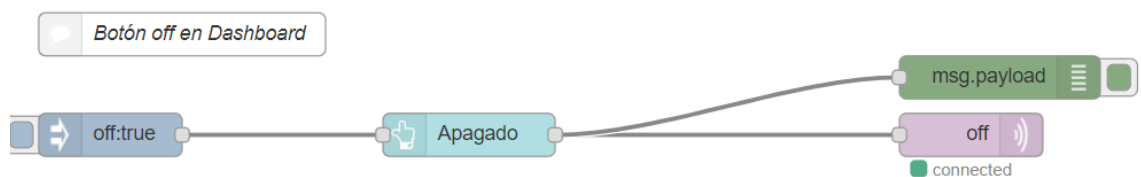


Figura 33. Botón de apagado remoto

- Flujo de configuración de tiempo de medición:** al igual que en el establecimiento de alarmas, los parámetros son recogidos desde la interfaz gráfica. Estos se publican desde Node-Red, cada uno de ellos con un topic asociado, además del botón de start que envía un 1 cuando se pulsa. En el código de comunicaciones del dispositivo se recibirán estas subscripciones y empezará la cuenta atrás de tiempo a partir de que se pulse el botón de start. Cuando el tiempo total sea igual a cero, finalizará la medición.

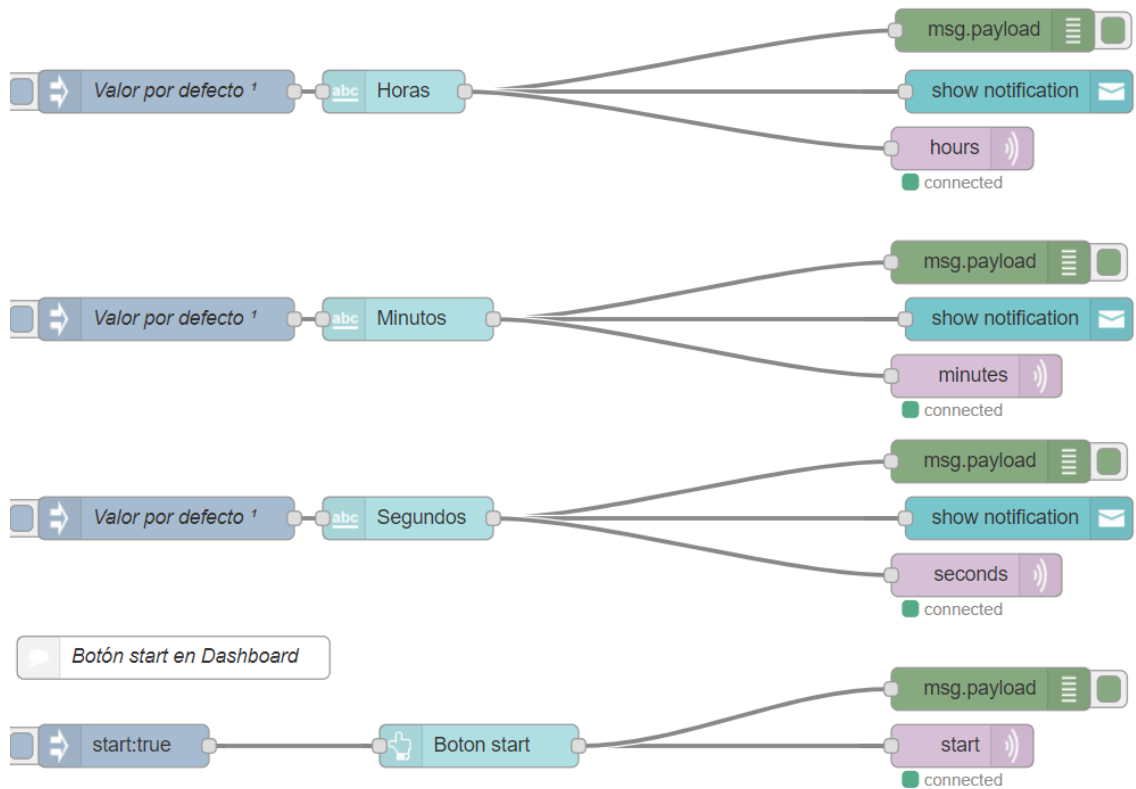


Figura 34. Flujo de temporización

- Flujo de envío de alertas:** cuando se activa una alarma en el dispositivo, este publica un topic de alarma al cual se suscribe Node-Red. Cuando se ha recibido este mensaje, automáticamente envía un correo de alerta a la dirección e-mail preconfigurada.

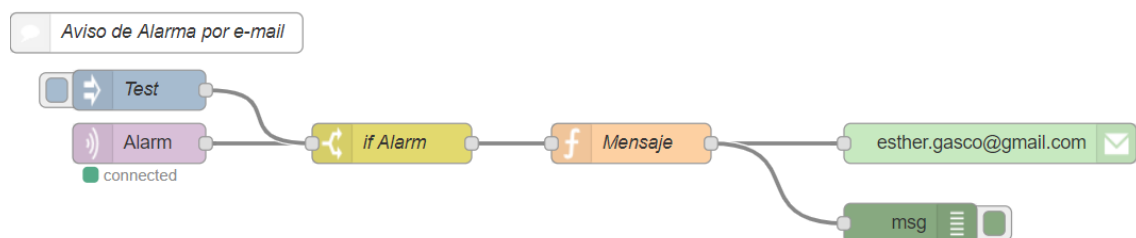


Figura 35. Flujo de aviso de alarma por e-mail

- **Flujo de persistencia de datos:** a través de este flujo será posible almacenar los datos procedentes de las mediciones en la base de datos Cloudant que incluye el Cloud de IBM. Node-Red, al igual que en el flujo de la lectura de datos de los sensores, se suscribe al evento JSON y filtra por tipo de dato para almacenar su contenido.

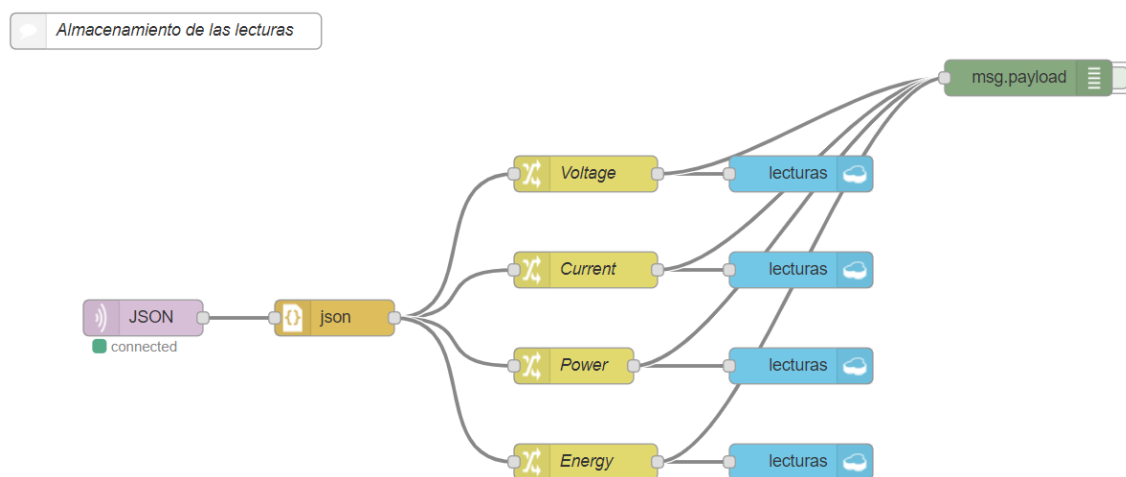


Figura 36. Flujo de almacenamiento de las lecturas

El resultado final de la interfaz gráfica, configurado a través de los flujos anteriores, es el siguiente:

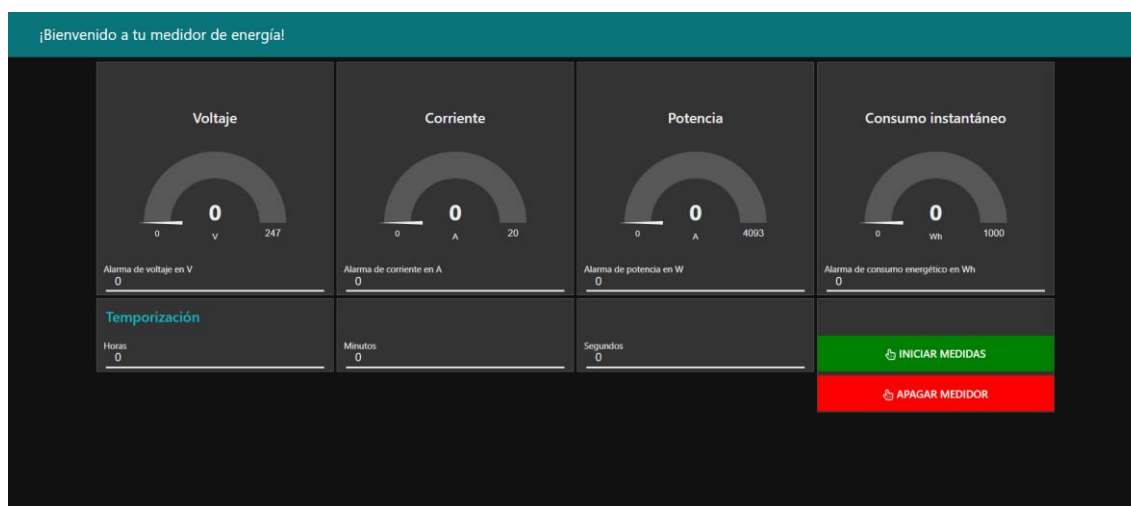


Figura 37. Interfaz gráfica

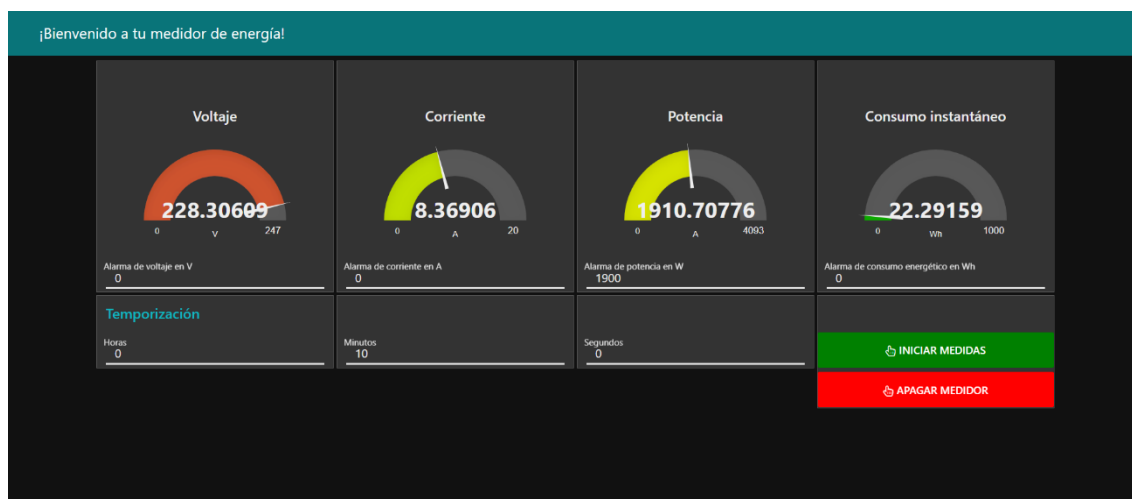


Figura 38. Interfaz gráfica visualizando mediciones

Como se observa en la Figura 38, se ha establecido una alarma en 1900W, cuando alcanza este valor, los relés del dispositivo actúan y cortan la tensión, pero en la interfaz Web, queda reflejada fija la última medida obtenida para que el usuario tenga constancia de lo ocurrido. También será posible consultar la base de datos donde se han almacenado todas las mediciones.

Capítulo 5

Viabilidad técnica

El presente proyecto reúne las características y condiciones técnicas que aseguran el cumplimiento de sus metas y objetivos. Se basa en la mejora y rediseño de un producto que se comercializa en la actualidad, el medidor de energía, por ello se trata de un dispositivo con una tecnología madura viable técnicamente. Se toma como ejemplo este tipo de dispositivos y se realiza un diseño de bajo coste conectado a IoT.

Por otro lado, las herramientas de software empleadas en el proyecto destacan por ser de código abierto o libre, esto implica que desarrolladores de todo el mundo contribuyen al código del software, donde se encuentran errores y defectos de forma proactiva, junto con el desarrollo de nuevas funciones y códigos para mejorar el rendimiento, por ello, presentan un alto nivel de confianza además de ofrecer un gran soporte documental.

Respecto al diseño de hardware, se ha realizado un diseño sencillo y de una dificultad baja de montaje, donde a través de los *datasheets* de los fabricantes podemos conocer las características de cada uno de sus componentes para que este sea óptimo y funcional. También, la plataforma sobre la que está montado, MSP432, es ampliamente configurable y facilita el trabajo de desarrollo.

La solución propuesta ofrece al usuario un dispositivo *plug-and-play* de fácil uso. El único punto débil es que la configuración tanto de la red Wifi como del aviso por e-mail ante alertas se ha de hacer sobre el código o nodos. Esto se podría solucionar ofreciendo soporte al producto donde se configuraría acorde con los requerimientos de cada cliente. A pesar de esto, el producto es competente y plenamente viable, se sitúa en el mismo rango o mejor que otros productos comerciales debido a sus características funcionales y económicas, además de ser una solución necesaria para la mejora de la eficiencia energética.

Capítulo 6

Valoración económica

6.1 Presupuesto del prototipo

En las siguientes tablas se desglosa en detalle el precio de los componentes y la mano de obra para el montaje completo de hardware y programación de software del analizador inteligente de consumo energético. Se ha tenido en cuenta cada uno de los componentes y materiales utilizados ya que se trata de un prototipo y esto hace que su coste sea elevado, por otro lado, tanto para el diseño de hardware como programación se ha considerado el precio por hora de un ingeniero experto en el campo. El diseño de la caja en 3D para el ensamblaje del producto y el montaje está realizado por técnicos con un coste menor asociado.

Concepto	Unidades	Precio unitario	Precio total
Materiales			
Microcontrolador MSP432P401R	1	18.58 €	18.58 €
Módulo Wifi CC3100BOOST	1	18.58 €	18.58 €
Sensor de corriente CST-1020	1	6.60 €	6.60 €
Transformador 220VAC/18VAC VB 0.35/1/18	1	4.56 €	4.56 €
Fuente de alimentación AC/DV 5V	1	10.86 €	10.86 €
Módulo de dos relés de 5VDC	1	2.15 €	2.15 €
Placa de Prototipado, Resina Epoxi con Fibra de Vidrio RE012-LF	1	2.18 €	2.18 €
Separadores Nylon	4	0.22 €	0.88 €
Tornillos Nylon	4	0.18 €	0.72 €
Tuercas Nylon	4	0.05 €	0.2 €

Resistencia axial 470kΩ	4	0.31 €	1.24 €
Resistencia axial 100kΩ	1	0.25 €	0.25 €
Resistencia axial 10kΩ	1	0.18 €	0.18 €
Resistencia axial 50Ω	1	0.37 €	0.37 €
Condensador electrolítico 10uF	2	0.98 €	1.96 €
Led rojo 3mm	1	0.59 €	0.59 €
Led verde 3mm	1	0.59 €	0.59 €
Schuko macho con interruptor	1	3.25 €	3.25 €
Schuko hembra	1	2.16 €	2.16€
Filamento PLA (250gr)	1	4.16 €	4.16 €
Bridas de plástico	10	0.26 €	2.6 €
Tubo termoretráctil (6m)	1	1.24 €	1.24 €
Bobina de cable (1m)	1	3.25 €	3.25 €
Bobina de estaño (100gr)	1	3.42 €	3.42 €
Flux de soldadura (10cc)	1	2.12 €	2.12 €
Total			92.69 €

Tabla 3. Presupuesto de componentes del prototipo

Concepto	Horas	Precio por hora	Precio total
Diseño			
Desarrollo	24	45 €	1080 €
Programación	112	45 €	5040 €
Diseño 3D	1	25 €	25 €
Ensamblaje			
Montaje	4	25 €	100 €
Total			6245 €

Tabla 4. Presupuesto de la mano de obra del prototipo

Concepto	Precio total
Componentes	92.69 €
Mano de obra	6220 €
Total	6312.69 €

Tabla 5. Presupuesto final del prototipo

6.2 Presupuesto de industrialización

El proyecto se ha desarrollado utilizando una placa de desarrollo de uso educacional, para poder hacer este proyecto viable industrialmente, se tendría que buscar una alternativa y realizar un nuevo diseño de hardware de la placa controladora con conectividad Wifi. Por otro lado, el ensamblaje propuesto tampoco sería el correcto, habría que diseñar una PCB que albergara todos los módulos de sensores y relés.

El diseño de la placa de sensores y relés es sencillo de obtener a partir del esquema electrónico, solamente habría que añadir más horas en concepto de desarrollo, pero como placa controladora, se va a suponer que el precio sigue siendo el mismo que en el punto anterior.

6.2.1 Mercado CE de dispositivos electrónicos

La Directiva 2014/35/UE es la aplicable al material eléctrico que se destina a usarse con una tensión nominal con un límite establecido entre 50 y 1000 V en corriente alterna y entre 75 y 1500 V en corriente continua [20], además, es aplicable a dispositivos electrónicos que supongan una novedad en el mercado europeo y productos novedosos importados de otros países.

El mercado CE y la Declaración de conformidad que lo acompaña es un contrato entre un fabricante y el estado que cumple con las regulaciones pertinentes. El cumplimiento de la normativa CE se puede establecer siguiendo una ruta de autocertificación, se puede determinar mediante el uso de un organismo notificado para llevar a cabo el análisis y prueba de un nuevo

producto, o una combinación de ambos [21]. En este caso se supone que la certificación la lleva a cabo una empresa externa.

Este marcado CE engloba 21 directivas aplicables según el tipo de dispositivo. Tras realizar un análisis de estas, el dispositivo propuesto en este proyecto tiene que cumplir las siguientes:

- Baja tensión (LVD)
- Compatibilidad Electromagnética (EMC)
- Equipos radioeléctricos y equipos terminales de telecomunicación y reconocimiento mutuo de su conformidad (RTTE)
- Restricción uso de sustancias peligrosas en aparatos eléctricos y electrodomésticos (ROHS)

Los costes derivados de la obtención de esta certificación para este tipo de dispositivos están alrededor de 10.000€. Esto incluye tanto el papeleo como las pruebas realizadas en laboratorio para el cumplimiento de la normativa.

Concepto	Precio total
Certificación	10.000 €

Tabla 6. Presupuesto de certificación CE

6.2.2 Presupuesto

Si realizamos el presupuesto para un solo dispositivo en fabricación, con su respectiva certificación, su coste sería excesivo ya que los costes asociados de diseño y programación también repercutirían en él. Por ello, cuando se industrializa un producto, los costes derivados de cualquier factor, tales como compra de componentes, gastos de certificación, diseño y programación, se ven reducidos drásticamente.

Los fabricantes ofrecen un precio muy competitivo ante grandes cantidades de compra de componentes, estos precios se ven reflejados en el siguiente presupuesto. Por otro lado, los

gastos de certificación incluyen las pruebas y mantenimiento del producto, ya que se ha contratado a una empresa externa para tal fin.

Concepto	Unidades	Precio unitario	Precio total
Materiales			
Microcontrolador MSP432P401R	1000000	3.45 €	3450000 €
Módulo Wifi CC3100BOOST	1000000	2.40 €	3450000 €
Sensor de corriente CST-1020	1000000	1.26 €	1260000 €
Transformador 220VAC/18VAC VB 0.35/1/18	1000000	1.06 €	1060000 €
Fuente de alimentación AC/DV 5V	1000000	2.87 €	2870000 €
Módulo de dos relés de 5VDC	1000000	0.15 €	150000€
Resistencia axial 470kΩ	4000000	0.005 €	2000 €
Resistencia axial 100kΩ	1000000	0.005 €	5000 €
Resistencia axial 10kΩ	1000000	0.005 €	5000 €
Resistencia axial 50Ω	1000000	0.005€	5000 €
Condensador electrolítico 10Uf	2000000	0.008 €	16000 €
Led rojo 3mm	1000000	0.009 €	9000 €
Led verde 3mm	1000000	0.009 €	9000 €
Schuko macho con interruptor	1000000	0.15 €	150000 €
Schuko hembra	1000000	0.10 €	100000 €
Caja	1000000	0.25	
PCB para el montaje de sensores y relés	1000000	5 €	5000000 €
Total			17.791.000 €

Tabla 7. Presupuesto de componentes para fabricación industrial

Concepto	Horas	Precio por hora	Precio total
Diseño			
Desarrollo	28	45 €	1260 €
Programación	112	45 €	5040 €
Total			6300 €

Tabla 8. Presupuesto de la mano de obra para fabricación industrial

Concepto	Precio total
Componentes	177.910.000 €
Mano de obra	6300 €
Certificación	10000 €
Total	17.807.300 €

Tabla 9. Presupuesto final de fabricación industrial

Si dividimos este coste final de producción entre 1.000.000 de unidades, obtenemos un coste final por unidad de **17.8073 €**.

Capítulo 7

Conclusiones

7.1 Objetivos

Para alcanzar los objetivos propuestos se planteó al inicio del proyecto una serie de prioridades. Estas prioridades se dieron según la categoría de estos objetivos: principales y extraordinarios. Los principales son los que el trabajo pretende alcanzar y los extraordinarios los que se podrían haber realizado en caso de dar el proyecto como finalizado. A continuación, se analizará cada uno de los objetivos para establecer las conclusiones de cumplimiento del proyecto.

7.1.1 Objetivos cumplidos

1. **Recopilación de los datos de los distintos sensores:** se diseñó un hardware adecuado para adaptar la señal de los sensores al rango de valores de las entradas analógicas del microcontrolador. Por otro lado, se programa el sistema empotrado para procesar estas mediciones.
2. **Medir la potencia consumida por una línea de corriente:** el dispositivo desarrollado permite obtener el consumo instantáneo del aparato que tenga conectado.
3. **Proveer al dispositivo de mecanismos de seguridad:** ante la recepción de diversas alarmas, dentro de la programación del dispositivo se crea una tarea donde se activan los relés si se cumplen ciertas condiciones. Estos relés son los encargados de cortar tanto fase como neutro de la alimentación para proteger al dispositivo.
4. **Dotación de conectividad Wifi al dispositivo para su implementación en IoT:** se integra el módulo Wifi CC3100 que dota de conectividad inalámbrica al dispositivo.

5. **Monitorizar el consumo energético de un determinado equipo en tiempo real a través de una interfaz Web:** se ha diseñado una interfaz Web basada en un *dashboard* para visualizar las diferentes mediciones en tiempo real.
6. **Posibilitar la interacción con el dispositivo a través de la configuración de alarmas de potencia, voltaje, corriente y consumo:** se programa en el dispositivo la tarea de comunicaciones que es la responsable del envío y recepción de eventos a través del protocolo MQTT. Por otro lado, en la interfaz Web se implementan los módulos para obtener los parámetros de alarma por parte del usuario.
7. **Posibilitar la interacción con el dispositivo para el establecimiento de periodos de medición:** se implementa en el código y en la interfaz Web el envío y recepción del periodo de medición establecido por el usuario.

7.1.2 Objetivos no cumplidos:

1. **Posibilidad de encender y apagar el dispositivo remotamente:** este objetivo, dentro de los principales, se ha cumplido parcialmente. Dentro de la interfaz Web se configura un botón de apagado remoto. Al pulsarlo se activarán los relés y el dispositivo dejará de funcionar. No se posibilita el encendido remoto ya que como se ha implementado esta función en la tarea de alarmas del dispositivo, cuando se activa un relé es necesario desconectar el dispositivo para que este vuelva a funcionar.
2. **Almacenamiento persistente de las alarmas en la memoria del dispositivo:** al principio del proyecto se planteó esta opción para que no fuera necesaria la configuración de las alarmas cada vez que se iniciara el dispositivo. En su lugar, se han implementado valores fijos ante los que el dispositivo actuará frente a una alarma, aunque esta no se configure. Estos valores coinciden con los máximos con los que se podría trabajar.
3. **Almacenamiento de los históricos de las mediciones y visualización a través de un calendario en la interfaz Web:** es interesante recoger este tipo de información para un posterior análisis por parte del usuario. No se ha podido implementar ya que el proyecto ha sufrido ciertos contratiempos y ha sido imposible seguir profundizando en Node-Red para aplicar esta idea. Sí ha sido posible almacenar todas las mediciones obtenidas en una base de datos para su posible futura aplicación en esta tarea.
4. **Funcionalidades adiciones: visualización en un LCD incorporado en el dispositivo, de forma local, las mediciones:** no se ha visto necesaria su implementación en el proyecto.

7.2 Conclusiones del trabajo

Como se ha visto en los puntos anteriores, los objetivos principales se han cumplido a falta de posibilitar el encendido remoto. Gracias a ellos se ha conseguido diseñar un dispositivo que cumple con el propósito principal del proyecto: medir energía en tiempo real a través de IoT. Los objetivos secundarios y el principal no completados son mejoras para el dispositivo y la interfaz Web, por ello no afectan a su correcto funcionamiento.

Tras la finalización de este proyecto se tiene un dispositivo de uso doméstico listo para empezar a funcionar y controlar el consumo del aparato que se le conecte consiguiendo de esta forma una mejora en la eficiencia energética.

7.3 Autoevaluación

Para concluir este documento, se hace énfasis en la consecución más que satisfactoria de casi todos los objetivos fijados. Además, no solo se han cubierto expectativas si no que se han conseguido características cercanas a equipos comerciales.

Esto ha sido posible gracias a muchas horas de trabajo, ya que se partía de una base no muy elevada respecto a programación, lo que ha dificultado y retrasado el cumplimiento de los objetivos según la planificación inicial.

Se han requeridos elevados conocimientos, sobre todo para la estructuración eficiente del código, que se ha ido consiguiendo progresivamente durante el transcurso del proyecto. Esta programación eficiente ha dado paso al cumplimiento de la mayoría de los objetivos, aunque ha quedado uno de ellos por completar debido a la falta de práctica en programación y tiempo.

Una vez que se ha profundizado en el lenguaje de programación de Arduino, además del funcionamiento del entorno de Energia dentro de Composer Studio, la programación ha resultado sencilla ya que se trata de un lenguaje estructurado de fácil entendimiento y con un amplio soporte en foros.

Respecto al seguimiento de la planificación, esta se ha visto alterada por varios motivos: dedicación parcial a la realización del proyecto y fallo en el funcionamiento de hardware. El haber cursado este proyecto con otras asignaturas ha supuesto un gran esfuerzo completarlo, además,

en el último momento, tras la solicitud de una actualización de software de la placa de desarrollo MSP432 provocó que esta dejara de funcionar. Al final, se consiguió otra placa y se volvió a montar el hardware desde cero, siguiendo adelante con el proyecto.

El éxito de este ha sido posible gracias a la dedicación y empeño. Este proyecto ha supuesto un gran reto personal y una gran satisfacción el haberlo conseguido.

7.3.1 Tiempo dedicado a cada PEC

PEC	Número de horas
Primeros pasos	30
1. Plan de trabajo	30
2. Seguimiento	100
3. Seguimiento + previa de la memoria	160
Código final	200

Tabla 10. Tiempo dedicado a cada PEC

7.4 Líneas de trabajo futuro

Una vez finalizado el proyecto, se plantean una serie de mejoras y ampliaciones como líneas a seguir en el futuro desde el punto de vista de los objetivos principales y extraordinarios, partiendo siempre del mismo prototipo.

Se propone como mejora la incorporación de los objetivos extraordinarios:

- Función de interruptor inteligente: agregar la opción de encendido remoto
- Persistencia de datos: dotar al dispositivo de memoria para el almacenamiento de las últimas alarmas configuradas.
- Interfaz Web con más funciones: permitir al usuario la visualización de históricos de consumo energético a través de la interfaz Web, además, posibilitar la descarga de estos para un posterior análisis.
- Incorporación de una pantalla táctil: esto facilitaría al usuario la interacción con el dispositivo en local y este pasaría a ser mucho más competitivo comercialmente. Como ha sido diseñado sobre la plataforma de desarrollo MSP432O401R, se puede incorporar la pantalla BOOSTXL-K350QVG-S1 de TI.

Glosario

AC/AC: dispositivo electrónico que convierte una señal de entrada de corriente alterna en una señal de salida alterna.

AC/DC: dispositivo electrónico que convierte una señal de entrada de corriente alterna en una señal de salida continua.

ADC: dispositivo electrónico capaz de convertir una señal analógica, ya sea de tensión o corriente, en una señal digital.

API: (Application programming interface). conjunto de subrutinas, funciones y procedimientos que ofrece el uso de bibliotecas para ser utilizado por otro software como una capa de abstracción.

Broker: eje central de cualquier protocolo de publicación/subscripción.

CCS: siglas de Composer Studio. Entorno de desarrollo integrado para desarrollar aplicaciones para los procesadores de Texas Instruments.

Dashboard: interfaz donde el usuario puede administrar el equipo y/o software.

HTTP: (Hypertext Transfer Protocol) es un sencillo protocolo cliente-servidor que articula los intercambios de información entre los clientes Web y los servidores HTTP.

IoT: siglas del Internet de las Cosas. Concepto que se refiere a una interconexión digital de objetos cotidianos con internet.

MSP432: placa de desarrollo de la familia de microcontroladores de señal mixta de Texas Instruments.

MQTT: (Message Queue Telemetry Transport). Protocolo usado para la comunicación machine-to-machine (M2M) en el Internet de las cosas (IoT)

M2M: (Machine-to-machine). Concepto genérico que se refiere al intercambio de información o comunicación en formato de datos entre dos máquinas remotas.

Node-Red: herramienta de programación para conectar dispositivos de hardware, API y servicios en línea.

SPI: (Serial Peripheral Interface). Estándar de comunicaciones usado principalmente para la transferencia de información entre circuitos integrados en equipos electrónicos.

Topic: temas de publicación o subscripción dentro del protocolo MQTT

Bibliografía

[1] Wikipedia, 2019. SPI. [En línea]

Obtenido de: https://es.wikipedia.org/wiki/Serial_Peripheral_Interface

[Último acceso: junio 2019]

[2] Wikipedia, 2019. MQTT. [En línea]

Obtenido de: <https://en.wikipedia.org/wiki/MQTT>

[Último acceso: junio 2019]

[3] Wikipedia, 2019. Node-Red. [En línea]

Obtenido de: <https://en.wikipedia.org/wiki/Node-Red>

[Último acceso: junio 2019]

[4] Blogthinkbig.com. Pablo G. Bejerano, 2013. Consumo de energía en el siglo XVII [En línea]

Obtenido de: <https://blogthinkbig.com/consumo-de-energia-siglo-xviii>

[Último acceso: junio 2019]

[5] elEconomista.es, 2019. La tecnología y el uso eficiente de la energía. [En línea]

Obtenido de: <https://www.economista.es/empresas-finanzas/noticias/9648960/01/19/La-tecnologia-y-el-uso-eficiente-de-la-energia.html>

[Último acceso: junio 2019]

[6] Hacedores.com, 2014. ¿Qué tarjeta de desarrollo elegir? (Parte 1). [En línea]

Obtenido de: <https://hacedores.com/que-tarjeta-de-desarrollo-elegir-parte-1/>

[Último acceso: junio 2019]

[7] Wikibooks.org, 2018. Lenguaje de programación Arduino. [En línea]

Obtenido de:

https://es.wikibooks.org/wiki/Lenguaje_de_programaci%C3%B3n_Arduino#Lenguaje_de_programaci%C3%B3n_Arduino

[Último acceso: junio 2019]

[8] efor.es, 2016. Tecnología de comunicación para IoT. [En línea]

Obtenido de: <https://www.efor.es/sites/default/files/tecnologias-de-comunicacion-para-iot.pdf>

[Último acceso: junio 2019]

[9] Wikipedia, 2019. Wifi. [En línea]

Obtenido de: <https://es.wikipedia.org/wiki/Wifi>

[Último acceso: junio 2019]

[10] Efergy Technologies SL, 2019. Medidores de energía online. [En línea]

Obtenido de: <https://efergy.com/medidores-de-energia-online/>

[Último acceso: junio 2019]

[11] wattio.com, 2019. Pack Energy 8. [En línea]

Obtenido de: <https://wattio.com/es/tienda/pack-energy-8>

[Último acceso: junio 2019]

[12] Open Energy Monitor Project, 2019. Introducing the OpenEnergyMonitor system. [En línea]

Obtenido de: <https://openenergymonitor.org/>

[Último acceso: junio 2019]

[13] Triad Magnetics, 2017. CST-1020. [En línea]

Obtenido de: <http://catalog.triadmagnetics.com/asset/cst-1020.pdf>

[Último acceso: junio 2019]

[14] Wikipedia, 2019. Transformador. [En línea]

Obtenido de: <https://es.wikipedia.org/wiki/Transformador>

[Último acceso: junio 2019]

[15] Wikipedia, 2019. Divisor de tensión. [En línea]

Obtenido de: https://es.wikipedia.org/wiki/Divisor_de_tensi%C3%B3n

[Último acceso: junio 2019]

[16] Wikipedia, 2019. Relé. [En línea]

Obtenido de: <https://es.wikipedia.org/wiki/Rel%C3%A9>

[Último acceso: junio 2019]

[17] The HiveMQ Team, 2015. MQTT Essentials Part 3: Client, Broker and Connection Establishment. [En línea]

Obtenido de: <https://www.hivemq.com/blog/mqtt-essentials-part-3-client-broker-connection-establishment/>

[Último acceso: junio 2019]

[18] Wikipedia, 2019. JSON. [En línea]

Obtenido de: <https://es.wikipedia.org/wiki/JSON>

[Último acceso: junio 2019]

[19] The HiveMQ Team, 2015. MQTT Essentials Part 4: MQTT Publish, Subscribe & Unsubscribe. [En línea]

Obtenido de: <https://www.hivemq.com/blog/mqtt-essentials-part-4-mqtt-publish-subscribe-unsubscribe/>

[Último acceso: junio 2019]

[20] Omologic, 2019. Mercado CE dispositivos electrónicos. [En línea]

Obtenido de: <https://omologic.es/marcado-ce-dispositivos-electronicos/>

[Último acceso: junio 2019]

[21] IBEX, Technology Development Specialists, 2016. CE Marking. [En línea]

Obtenido de: <https://electronic-products-design.com/resources/regulations/ce-marking>

[Último acceso: junio 2019]