



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

DIRECCIÓN ADJUNTA DE INNOVACIÓN Y CONOCIMIENTO
GERENCIA DE CAPITAL HUMANO
POSGRADOS

**“SISTEMA DE MONITOREO, TELEMETRIA
Y TELECONTROL MTT-SS17”**

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

Presenta:

Iraam Antonio López Salas.

Asesor:

M. en C. Elio Atenógenes Villaseñor.

Ciudad de México, febrero de 2018.



AUTORIZACIÓN DE IMPRESIÓN

Ciudad de México, a 30 de enero de 2018

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

"Sistema de monitoreo, telemetría y telecontrol MTT-SS17"

Desarrollada por el alumno

Nombre: **Iraam Antonio**

Apellido paterno: **López**

Apellido materno: **Salas**

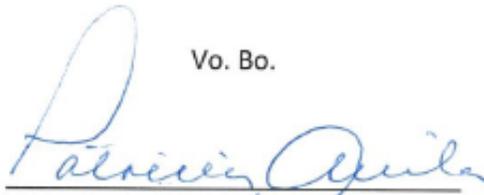
Desarrollado bajo la asesoría de la:

Mtro. Elio Atenógenes Villaseñor García

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

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

Vo. Bo.



Mtra. Patricia Ávila Muñoz
Gerencia de Capital Humano

Agradecimientos

Quiero dedicar la culminación de este trabajo, en primer lugar a mis padres. A **mi madre** que siempre, desde que tengo memoria, me impulsaba a prepararme mas y mejor, que nunca dejó de aconsejarme para ser mejor persona, para mi y para quienes me rodean, ejemplo de amor y respeto por su familia y por todos quienes tuvimos la gran fortuna de tenerla a nuestro lado; y a **mi padre**, que aunque nos dejo relativamente pronto, fue un excelente ejemplo, de un padre que rompió paradigmas muy arraigados para esos tiempos, dándonos una vida libre de prejuicios, donde lo importante no es lo que digan los demás, sino lo que uno quiere y desea, dentro de un marco de respeto hacia los demás. Se que ahora estarán juntos, y continúan cuidándonos desde donde están. Siempre los voy a extrañar.

Tambien, agradecer a mi esposa **Laura Guadalupe**, a mi hija **Jimena** y mi hijo **Ivar**, por su apoyo para lograr este paso importante para mi. Ustedes son el motor que me levanta e impulsa cada dia para ir sembrando y cosechando todos estos logros. Me siento orgulloso de ustedes. Gracias...!

A mis hermanas **Dinah, Aster, Selene, Citlalli**, mi hermano **Plinio**, se que siempre están al pendiente de mi, gracias por todo su cariño y apoyo.

A Robert, mi amigo y compañero de tantos años, que siempre es una tranquilidad saber que te quedas a cargo cuando quiero emprender nuevos proyectos.

Y finalmente, a todas las personas que tambien me apoyaron en este proceso, Gaby Ahumada, gracias por la confianza y amistad que me has brindado, Pedro y Enrique, por su compañerismo y apoyo en estos dos años, profesor Francisco...Gracias a todos...!

Tabla de contenido

Introducción	1
Capítulo 1. Resumen ejecutivo	4
1.1 Antecedentes.....	4
1.2. Problema	4
1.3. Hipótesis.....	5
1.4. Justificación	5
1.5. Objetivo principal.....	6
1.6. Objetivos específicos.....	6
Capítulo 2. Diseño conceptual del sistema de monitoreo, telemetría y telecontrol MTT-SS17.....	7
2.1. Metodología	8
2.2 Requerimientos	9
2.2.1. Requerimientos funcionales, módulo coordinador	9
2.2.2. Requerimientos no funcionales, módulo coordinador	9
2.2.3. Requerimientos funcionales, módulo router	10
2.2.4. Requerimientos no funcionales, módulo router	10
2.2.5. Requerimientos funcionales, módulo final	11
2.2.6. Requerimientos no funcionales, módulo final	11
2.3 Arquitectura general del sistema MTT-SS17	12
2.4. Arquitectura de Hardware, módulo coordinador.....	13
2.5 Arquitectura de hardware, módulo router y dispositivo final	14
2.6 Arquitectura de software y firmware	15
Capítulo 3. Diseño del sistema de monitoreo, telemetría y telecontrol MTT-SS17.....	18
3.1 Plataformas de desarrollo	19
3.1.1. Características de la Raspberry PI Zero.....	20
3.2. Microcontroladores.....	22
3.2.1. Microcontrolador ATMEGA328 (ATMEL, 2015).....	24
3.3. Sensores	25
3.3.1. Sensor de temperatura y Humedad Relativa (HR) DHT22 (AM2302)	25
3.3.2. Sensor de presión barométrica y temperatura BMP280.....	26
3.4 Web Service UBIDOTS [1]	28
3.4.1. Protocolo MQTT en UBIDOTS	28
3.4.2. Protocolo HTTP en UBIDOTS	30

3.5 Red tipo malla (Mesh Network)	31
3.5.1. Módulo coordinador	32
3.5.2. Módulo Router	33
3.5.3. Módulo final	33
3.6. Diseño físico del MTT-SS17	34
3.6.1. Módulo coordinador	34
3.6.2 Módulos router y final	34
3.7. Listas de Materiales (BOM)	39
3.7.1. Lista de materiales del módulo coordinador	39
3.7.2. Lista de materiales del módulo router y/o final	40
3.8. Esquema de Mantenimiento	41
3.8.1. Mantenimiento preventivo.....	41
3.8.2. Mantenimiento correctivo.....	43
<i>Capítulo 4. Implementación del MTT-SS17</i>	44
4.1 Módulo coordinador.....	44
4.2. Módulos router y finales.....	45
<i>Capítulo 5. Resultados</i>	47
5.1. Recepción de datos en el módulo coordinador	47
5.2. Recepción de datos en el web service de UBIDOTS.....	48
<i>Conclusiones</i>	50
<i>Bibliografía</i>	51

Índice de figuras

Figura 1. Esquema de metodología.....	8
Figura 2. Arquitectura de red del MTT-SS17.....	12
Figura 3. Arquitectura del módulo MTT-SS17 coordinador.....	13
Figura 4. Arquitectura del módulo MTT-SS17 router o dispositivo final.....	14
Figura 5. Arquitectura de software, módulo coordinador.....	15
Figura 6. Arquitectura de software módulos router y finales.....	17
Figura 7. Raspberry pi zero.....	21
Figura 8. Distribución de pines del microcontrolador ATMEGA328.....	24
Figura 9. Esquema de conexiones del sensor BMP280.....	27
Figura 10. Arquitectura de la red tipo malla.....	31
Figura 11. Diagrama esquemático del módulo coordinador.....	35
Figura 12. PCB del módulo coordinador.....	36
Figura 13. Diagrama esquemático de los módulos router y/o final.....	37
Figura 14. PCB del módulo router y/o final.....	38
Figura 15. Raspberry PI Zero y accesorios.....	44
Figura 16. Raspberry PI Zero y xbee.....	45
Figura 17 Módulos router y final.....	46
Figura 18. Datos recibidos por la RBPI Zero enviados desde los módulos Router y Final.....	47
Figura 19. Datos recibidos en el web service de UBIDOTS, enviados desde el módulo coordinador.....	48
Figura 20. Controles implementados en el web service de UBIDOTS.....	49

Índice de tablas.

<i>Tabla 1. Comparativo de plataformas de desarrollo.....</i>	<i>19</i>
<i>Tabla 2. Comparativo de microcontroladores de rango bajo.....</i>	<i>23</i>
<i>Tabla 3. Lista de materiales BOM, módulo coordinador.....</i>	<i>39</i>
<i>Tabla 4. Lista de materiales BOM, módulo router y/o final.....</i>	<i>40</i>

Introducción

El concepto de Smart City (GRUPO ENEL, 2016) requiere una base tecnológica, tanto de hardware como de software, además de la infraestructura urbana para lograr sus objetivos. El sistema de monitoreo, telemetría y telecontrol MTT-SS17, es una plataforma de hardware, que permite desarrollar soluciones dentro del área de Sensores Inteligentes (Smart Sensors) e infraestructura de comunicaciones, al proveer una red con conectividad WiFi y capacidad de E/S digitales, que puede ser utilizado para envío de datos de forma bidireccional, tanto de medición de variables tales como temperatura, humedad, y presión barométrica, entre otras, así como señales de control para diferentes dispositivos. Si se considera el desarrollo de un software capaz de interactuar con estos módulos, para analizar los datos recibidos y tomar acciones en respuesta a dicho análisis, se podría crear un sistema de gran utilidad, que permita, por ejemplo, medir el impacto ambiental en zonas o áreas específicas. Si estos módulos se instalaran de manera general en una ciudad, se podrían recolectar diferentes datos de manera masiva, lo que permitiría un análisis de distintas variables como mecanismos de contaminación, impacto ambiental, entre otras. Se podrían aplicar conceptos de Big Data para un mejor análisis de los datos recabados, pudiendo hacer más eficiente el transporte, la movilidad, el consumo energético, etc.

Con el diseño de este módulo, se pretende aportar una herramienta flexible y sencilla de utilizar para su uso en proyectos de medición de impacto ambiental o monitoreo de zonas específicas, entre otros.

“Una Smart City, o ciudad inteligente, se puede describir como aquella ciudad que aplica las tecnologías de la información y de la comunicación (TIC) con el objetivo de proveerla de una infraestructura que garantice:

- Un desarrollo sostenible.
- Un incremento de la calidad de vida de los ciudadanos.
- Una mayor eficacia de los recursos disponibles.
- Una participación ciudadana activa.

Por lo tanto, son ciudades que son sostenibles económica, social y medioambientalmente. La Smart City nace de la necesidad de mantener una armonía entre estos aspectos.

En décadas recientes la mayoría de las ciudades del planeta han experimentado un rápido crecimiento, lo que ocasiona que estén enfrentando problemas cada vez mayores, consecuencia de este crecimiento acelerado.

Entre estos problemas se encuentran los siguientes:

- El abastecimiento energético.
- Las emisiones de CO₂.
- La planificación del tráfico automovilístico.
- La provisión de bienes y materias primas.
- La prestación de servicios sanitarios y de seguridad a todos quienes residan en estos enormes y masificados centros de población.

Una ciudad inteligente se puede basar en diferentes subsistemas, entre los cuales se encuentran los siguientes:

Generación eléctrica distribuida. Reside en que la ciudad inteligente posea generación eléctrica repartida por el territorio: el abastecimiento es individualizado (micro-generación), no centralizado.

Smart Grids: Se conoce como Smart Grids a las redes inteligentes interconectadas, las cuales poseen una circulación bidireccional de datos entre el centro de control y el usuario.

Smart Metering: Se trata de la medición inteligente de los datos de gasto energético de cada usuario, a través de telecontadores donde se realizan las lecturas a distancia y a tiempo real.

Smart Buildings: Como modelo de eficiencia, los edificios deben ser inteligentes. Edificios domóticos que respetan el medio ambiente y que poseen sistemas de producción de energía integrados.

Smart Sensors: Los sensores inteligentes tendrán la función de recopilar todos los datos necesarios para hacer de la ciudad una Smart City. Son parte fundamental para mantener la ciudad conectada e informada, y hacer que cada subsistema cumpla su función.

eMobility: Implantación del vehículo eléctrico, y los respectivos puestos de recarga públicos y privados.

Tecnologías de la información y la comunicación (TIC): Son las tecnologías de la información que ayudarán a la hora de controlar los diferentes subsistemas que componen la Smart City, mediante las cuales los ciudadanos y las entidades administrativas pueden participar activamente en el control de la ciudad.

Smart Citizen: Los ciudadanos son sin duda la parte fundamental de una Smart City, ya que sin su participación activa no es posible poder llevar a cabo estas iniciativas.



Capítulo 1

Resumen ejecutivo



Capítulo 1. Resumen ejecutivo

Este capítulo inicia con una serie de antecedentes respecto al proyecto que se presenta, así como conceptos básicos de los temas abordados durante su desarrollo, además se presentan el problema, la hipótesis, la justificación y los objetivos, tanto general como específicos del proyecto, denominado “Sistema de Monitoreo, Telemetría y Telecontrol MTT-SS17”. El planteamiento que se hace en este proyecto es diseñar el hardware y software adecuado para la implementación del sistema mencionado. Además del hardware y software que se implementaran, el proyecto requiere de un servidor web para completar la funcionalidad del sistema. En este proyecto se utilizara un web service, denominado UBIDOTS (UBIDOTS, 2016), que es un servicio en la nube que nos permite almacenar e interpretar información de sensores en tiempo real, acelerando el desarrollo de aplicaciones de IoT.

1.1 Antecedentes

El sistema de monitoreo, telemetría y telecontrol MTT-SS17**, consiste en una serie de sensores con conectividad vía WiFi, que tiene la finalidad de medir y monitorear variables ambientales como temperatura, humedad y presión barométrica, entre otras y enviar comandos de control, que pueden utilizarse para sistemas como iluminación y ahorro de energía, etc. Estos módulos cuentan con entradas analógicas, E/S digitales y PWM completamente configurables desde un portal WEB, y son capaces de interconectarse con otros módulos similares para formar redes de tráfico de datos.

1.2. Problema

El desarrollo de grandes zonas urbanas es uno de los principales factores que están provocando un mayor calentamiento global. Debido a esto, se requieren sistemas que nos permitan medir el impacto ambiental de nuevos desarrollos dentro de las zonas urbanas, a fin de crear conciencia en la sociedad y legislar para tratar de reducir el impacto negativo que tiene las zonas urbanas en el ambiente. Sin un

sistema capaz de generar información precisa del impacto ambiental que genera, por ejemplo, la construcción de un nuevo centro comercial, un fraccionamiento o edificio, resulta muy difícil saber qué medidas ambientales deben ser tomadas en cuenta para reducir dicho impacto ambiental, sin menoscabo del progreso y crecimiento económico de la sociedad actual.

1.3. Hipótesis

¿Cómo desarrollar un sistema de medición de temperatura, Humedad Relativa (HR) y presión barométrica, con capacidad de conectarse en red y vía WiFi, para envío de información de forma bidireccional, a la plataforma UBIDOTS?

H1. Es posible desarrollar de un sistema de medición de temperatura, HR y presión barométrica, con capacidad de conectarse vía WiFi, para envío de información de forma bidireccional, a la plataforma UBIDOTS.

1.4. Justificación

El módulo MTT-SS17 es una herramienta especializada con alta flexibilidad de operación, para ser utilizada como plataforma de conectividad para los objetos, telemetría y control, dando soporte tanto al concepto de Internet de las cosas como el de Smart City y Big Data, al hacer más factible la recolección de datos en diferentes ambientes.

El Internet de las cosas es una infraestructura tanto de hardware como de software que interconecta objetos físicos valiéndose del Internet. Los objetos utilizan hardware especializado que le permite no solo la conectividad a Internet, sino que además programa eventos específicos en función de las tareas que le sean dictadas remotamente desde una página web, accesible desde cualquier punto de acceso a internet en el mundo.

1.5. Objetivo principal

Diseñar un sistema de monitoreo, telemetría y telecontrol que pueda formar redes de datos y sea capaz de conectarse al web service de UBIDOTS, con el fin de transferir información de forma bidireccional desde los puntos de medición hasta el portal web mencionado, a fin de tener la capacidad de concentrar los datos generados por la red de sensores para su análisis, además de tener la capacidad de controlar diferentes tipos de cargas.

1.6. Objetivos específicos

1. Diseñar los módulos coordinador, router y final para formar una red tipo malla con acceso a internet.
2. El módulo coordinador, debe tener capacidad de conectarse a internet vía WiFi y es el que controla todo el flujo de datos desde y hacia la plataforma UBIDOTS.
3. El módulo coordinador, debe diferenciar de manera individual cada módulo router y final que se conecta a la red del sistema MTT-SS17.
4. Los módulos router y final no tienen conectividad WiFi.
5. Los módulos router y final leen, almacenan y transmiten la información de los sensores de temperatura, humedad relativa y presión barométrica.
6. Los módulos router y final tienen 3 salidas PWM (Modulación por ancho de pulso), para control de cargas desde la plataforma UBIDOTS.



Capítulo 2

Diseño conceptual del sistema de monitoreo, telemetría y telecontrol MTT-SS17



Capítulo 2. Diseño conceptual del sistema de monitoreo, telemetría y telecontrol MTT-SS17

En este capítulo se documenta el diseño conceptual del proyecto, se abordan los temas de la metodología utilizada para su desarrollo, se analizan los requerimientos, tanto funcionales, como no funcionales, y se define la arquitectura del sistema, respecto a la red de telemetría y telecontrol que se debe formar, así como de las secciones que lo componen, tanto de hardware como de software.

Se conoce como telemetría al sistema que permite la monitorización, mediación y/o rastreo de magnitudes físicas o químicas a través de datos que son transferidos a una central de control. El sistema de telemetría se realiza normalmente mediante comunicación inalámbrica pero también se puede realizar a través de otros medios como: teléfono, redes de ordenadores, enlace de fibra óptica, entre otros. La telemetría es usada en áreas muy diversas que va desde el automovilismo, aviación, astrología, pasando por la agricultura, industria de petróleo, medicina y hasta biología.

El telecontrol o telemando consiste en el envío de indicaciones a distancia mediante un enlace de transmisión (por ejemplo, a través de cables, radio, dirección IP, etc.), utilizando órdenes enviadas para controlar un sistema o sistemas remotos que no están directamente conectados al lugar desde donde se envía el telecontrol. La palabra viene de dos raíces tele = distancia (griego), y control = controlar. Los sistemas que necesitan medición remota y reporte de información de interés para el diseñador del sistema o el operador deben usar la contrapartida del telecontrol, la telemetría. El telecontrol se puede llevar a cabo en tiempo real, o no dependiendo de las circunstancias.

2.1. Metodología

En la figura siguiente se muestra un diagrama que representa la metodología seguida en este proyecto, en primer lugar se definieron los requerimientos tanto funcionales como no funcionales. En base a los requerimientos se define la arquitectura del sistema completo. Una vez definida la arquitectura se diseña un prototipo funcional, se realizan pruebas y se hacen ajustes en base a los resultados obtenidos, de manera cíclica. En cada ciclo se documentan las pruebas y correcciones realizadas. Una vez que se obtienen los resultados deseados, se implementa el prototipo final y se elabora la documentación definitiva.

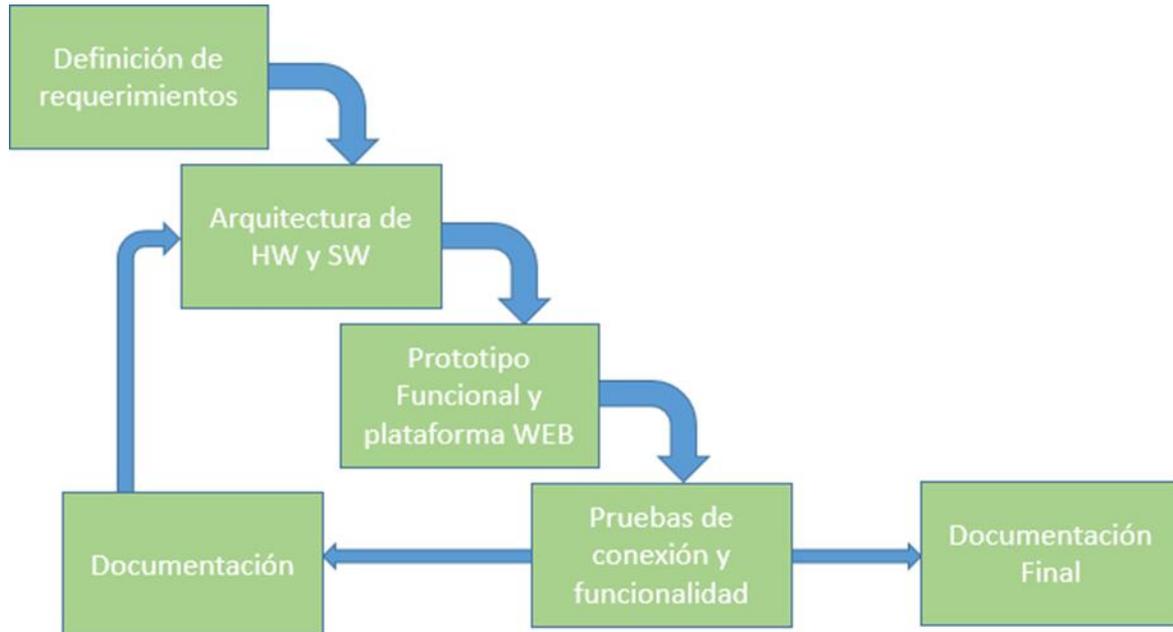


Figura 1. Esquema de metodología.

Fuente: Elaboración propia de acuerdo al ciclo de desarrollo de un sistema embebido.

2.2 Requerimientos

2.2.1. Requerimientos funcionales, módulo coordinador

- Es el módulo principal de la red mesh y se encarga de controlar el flujo de datos entre la plataforma UBIDOTS y todos los módulos que conforman la red mesh del sistema MTT-SS17.
- Debe tener conectividad WiFi y Xbee.
- Debe conectarse a internet vía WiFi, para acceder al servicio web de UBIDOTS.
- Debe conectarse a la red mesh vía protocolo Xbee.
- El coordinador debe ser capaz de diferenciar los datos de cada módulo router o final que conforman la red mesh del sistema MTT-SS17, para trabajarlos de manera independiente en UBIDOTS.
- El coordinador se alimenta de una fuente de voltaje de 120VCA de entrada y 5VCD de salida.

2.2.2. Requerimientos no funcionales, módulo coordinador

- El módulo debe ser capaz de funcionar hasta con 256 módulos router o finales.
- Operación en ambiente externo o al aire libre.
- Protección contra sobrecargas.
- Protección contra vandalismo.

2.2.3. Requerimientos funcionales, módulo router

- Lee información de los sensores de temperatura, humedad relativa y presión barométrica.
- Debe tener 3 salidas PWM (Modulación por ancho de pulso), controladas desde la plataforma UBIDOTS.
- Debe tener conectividad Xbee.
- Enviar los datos recolectados de los sensores hacia el coordinador de la red, agregando un identificador que permita al coordinador diferenciar de que módulo está recibiendo la información.
- Se encarga de re-direccionar el flujo de datos entre los módulos finales, otros router y el coordinador.
- Debe conectarse a internet vía módulo coordinador, para acceder al servicio web de UBIDOTS.
- Debe conectarse a la red mesh vía protocolo Xbee.
- El router se alimenta de una batería recargable de voltaje de 5VCD de salida.
- Rango de medición de temperatura: -20 a 60 grados centígrados.
- Rango de medición de humedad relativa: 0 a 100%.
- Rango de medición de presión barométrica: 500 a 825 mmHg.
- Sensores con protocolo digital I2C o 1 wire

2.2.4. Requerimientos no funcionales, módulo router

- Operación en ambiente externo o al aire libre.
- La instalación de los sensores debe considerar que éstos no se vean afectados por la operación del módulo en sí, como por ejemplo, evitar el auto calentamiento del sensor de temperatura.
- Protección contra sobrecargas.
- Protección contra vandalismo.

2.2.5. Requerimientos funcionales, módulo final

- Lee información de los sensores de temperatura, humedad relativa y presión barométrica.
- Debe tener 3 salidas PWM (Modulación por ancho de pulso), controladas desde la plataforma UBIDOTS.
- Debe tener conectividad Xbee.
- Enviar los datos recolectados de los sensores hacia el coordinador de la red, agregando un identificador que permita al coordinador diferenciar de que módulo está recibiendo la información.
- Debe conectarse a internet vía módulo coordinador y/o router, para acceder al servicio web de UBIDOTS.
- Debe conectarse a la red mesh vía protocolo Xbee.
- El final se alimenta de una batería recargable de voltaje de 5VCD de salida.
- Rango de medición de temperatura: -20 a 60 grados centígrados.
- Rango de medición de humedad relativa: 0 a 100%.
- Rango de medición de presión barométrica: 500 a 825 mmHg.
- Sensores con protocolo digital I2C o 1 wire.

2.2.6. Requerimientos no funcionales, módulo final

- Operación en ambiente externo o al aire libre.
- La instalación de los sensores debe considerar que éstos no se vean afectados por la operación del módulo en sí, como por ejemplo, evitar el auto calentamiento del sensor de temperatura.
- Protección contra sobrecargas.
- Protección contra vandalismo.

2.3 Arquitectura general del sistema MTT-SS17

En la siguiente figura se muestra la arquitectura de red del MTT-SS17, que consta de la conectividad a un web service (UBIDOTS), un coordinador y módulos router y dispositivos finales de red.

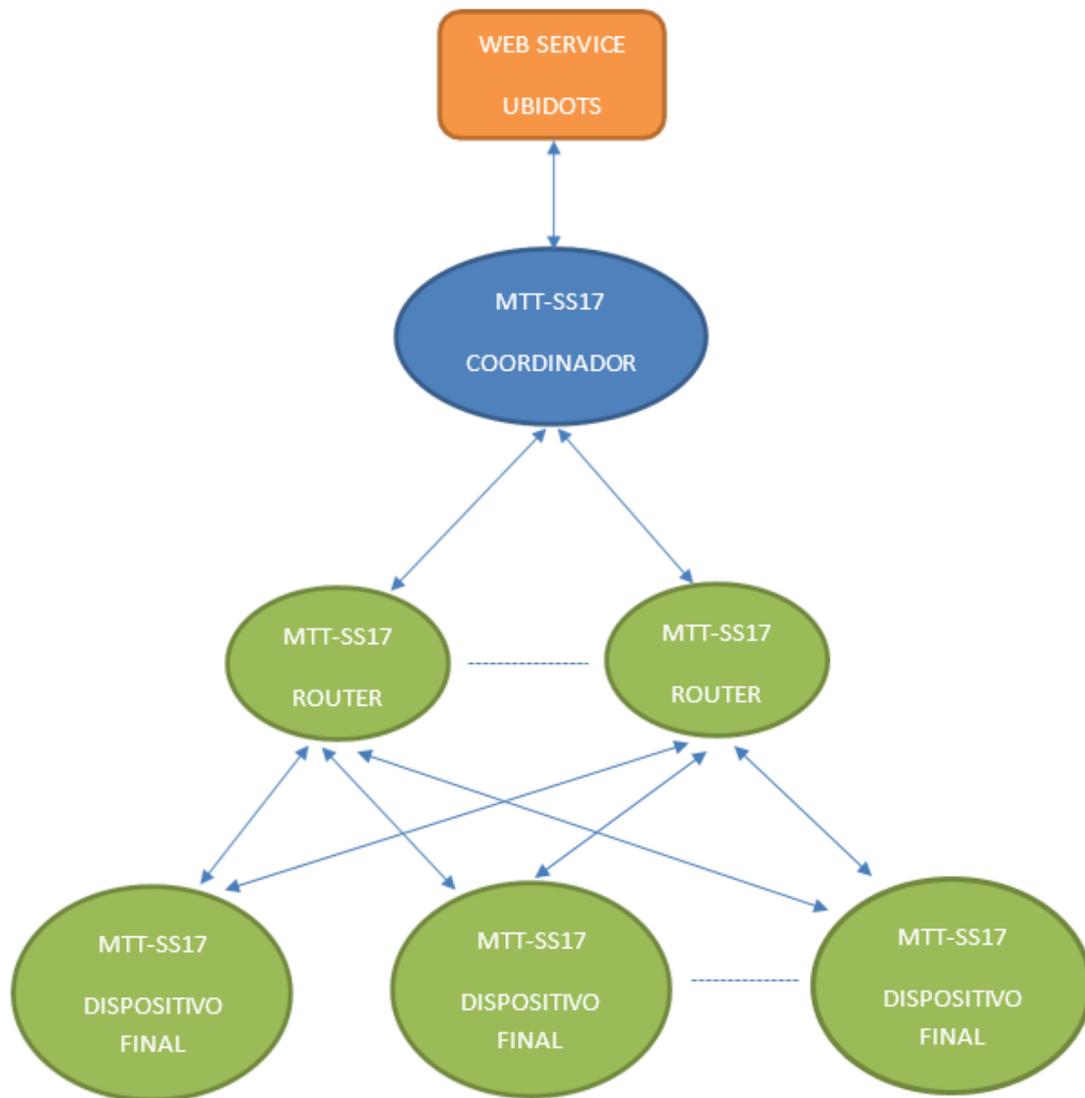


Figura 2. Arquitectura de red del MTT-SS17.

Fuente: Elaboración propia de acuerdo a la documentación de redes mesh.

2.4. Arquitectura de Hardware, módulo coordinador

El módulo coordinador tiene la función de ser el enlace entre el servicio WEB de UBIDOTS y los módulos router o final. Este módulo tiene conectividad tanto WiFi como Zigbee, y tiene la capacidad de contener una base de datos. Baza su operación en una tarjeta Raspberry PI zero, que ya incluye conectividad WiFi.

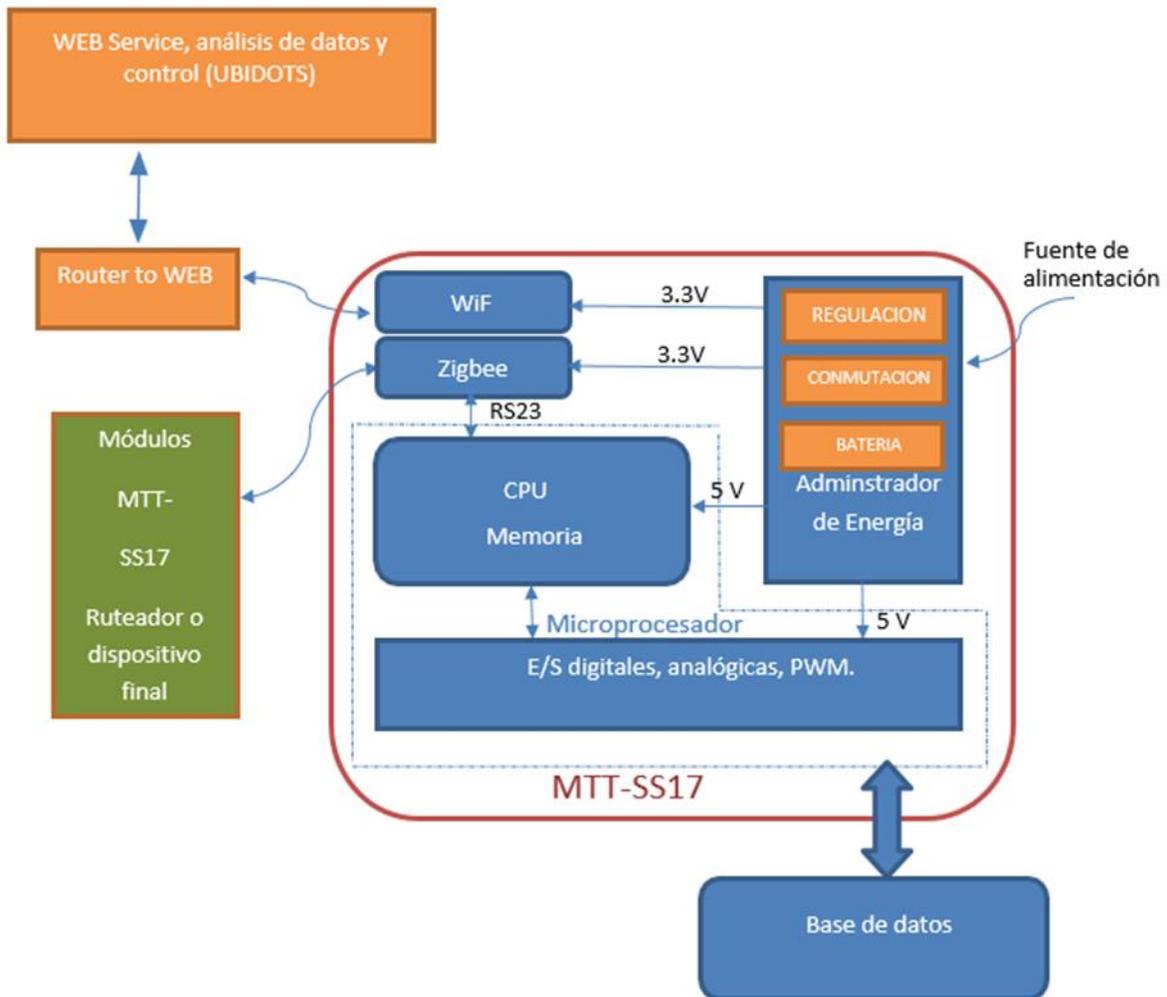


Figura 3. Arquitectura del módulo MTT-SS17 Coordinador

Fuente: Elaboración propia de acuerdo a los esquemas de arquitectura de hardware de sistemas embebidos.

2.5 Arquitectura de hardware, módulo router y dispositivo final

Los módulos router y dispositivos finales incluyen el mismo hardware, y basan su operación en el microcontrolador ATMEGA328. La diferencia está en el firmware que controla los dispositivos de comunicación Xbee. Estos módulos toman las lecturas de los sensores y envían la información hacia el coordinador, quien a su vez, reenvía la información hacia el Web Service.

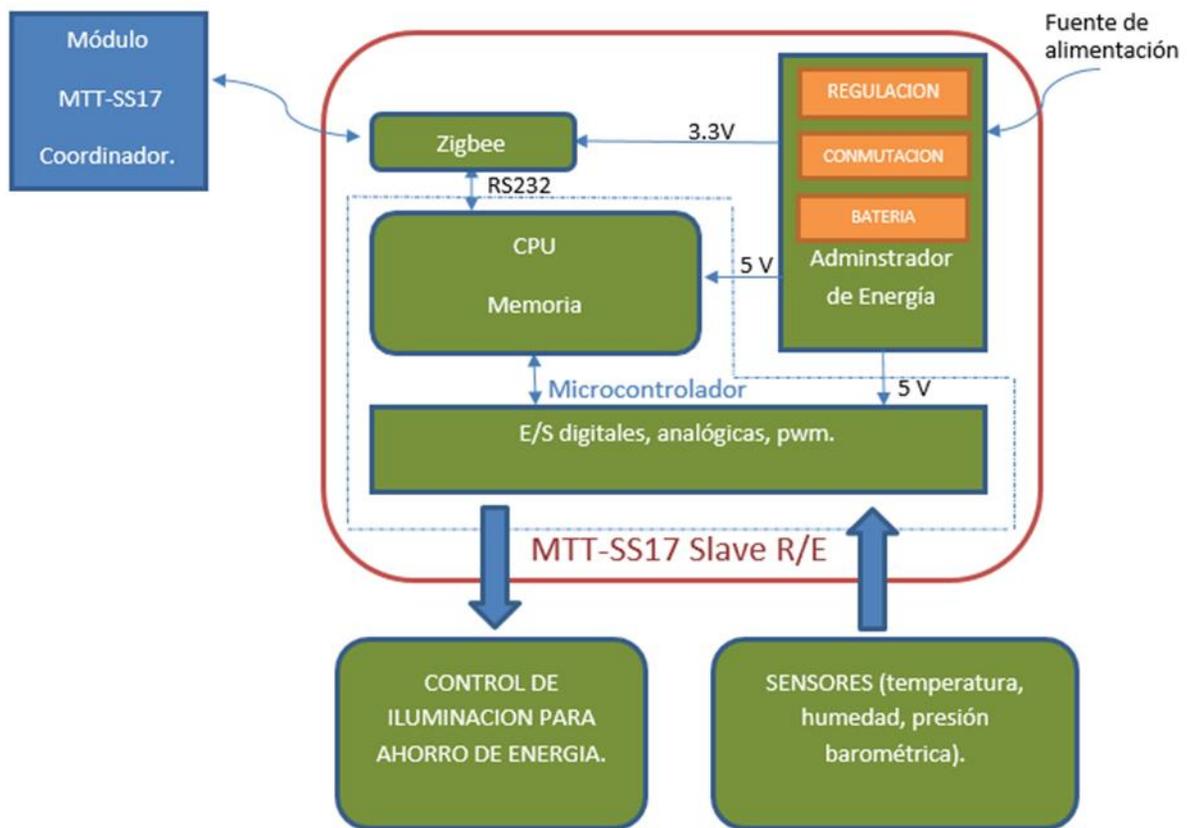


Figura 4. Arquitectura del módulo MTT-SS17 Router o dispositivo final.

Fuente: Elaboración propia de acuerdo a los esquemas de arquitectura de hardware de sistemas embebidos.

El esquema de red del sistema MTT-SS17 consta de 3 tipos de módulos diferentes: Módulo Coordinador, módulo router y dispositivo final.

El Módulo coordinador es el único módulo que tiene acceso a internet mediante su conectividad WiFi, y hace las veces de coordinador de la red tipo malla (mesh network) formada por los dispositivos Zigbee.

Los Módulos Router y final no tienen acceso directo a WiFi, solo a través del módulo coordinador. La función router o de dispositivo final está definida por la función del Zigbee y el esquema de red tipo malla.

2.6 Arquitectura de software y firmware

El módulo coordinador debe ser capaz de 2 cosas principalmente, conectarse a internet vía WiFi, y controlar el tráfico de datos desde los sensores (módulos router y final) hacia el internet web service y desde el internet hacia cada módulo, la conexión entre el módulo coordinador y los módulos router y final deberá realizarse vía UART hacia el módulo transmisor de RF.

Para realizar sus funciones, se requiere un software en el módulo coordinador, que debe manejar los protocolos de WiFi, HTML y UART, desde el nivel de capa de aplicación hasta la capa física, además de tener la capacidad de controlar los periféricos asociados del hardware, por lo que se requiere un Sistema Operativo base.

En la siguiente figura se muestra un diagrama de la arquitectura de software del módulo coordinador.

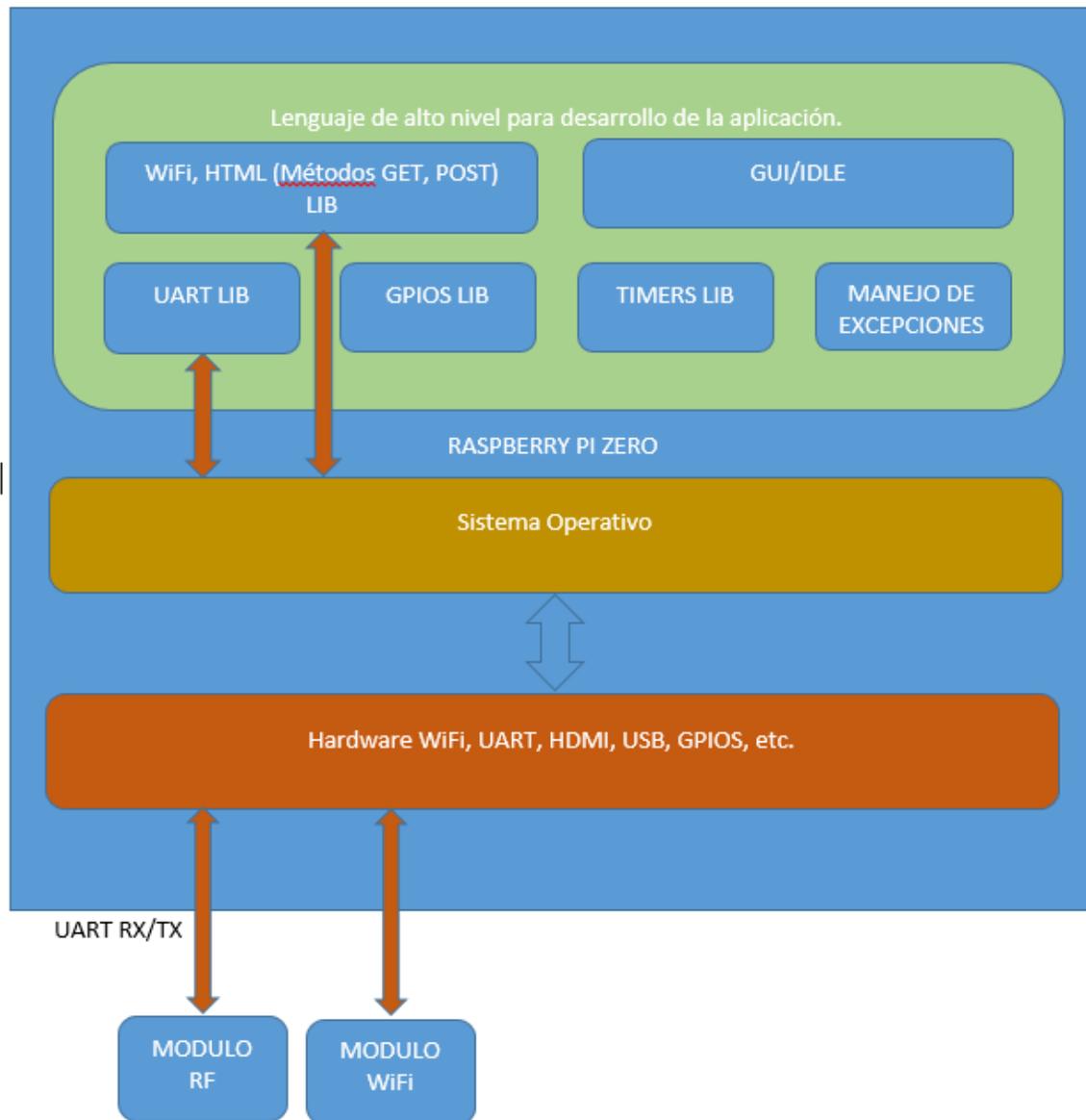


Figura 5. Arquitectura de software, módulo coordinador.

Fuente: Elaboración propia de acuerdo a los esquemas de arquitectura de software de sistemas embebidos.

Los módulos router y final utilizan como procesador principal un microcontrolador, por lo que se requiere de software embebido o firmware para que dicho procesador lleve a cabo las funciones requeridas por los requerimientos del sistema. En estos módulos, se requiere que el procesador sea capaz de leer los sensores de temperatura, humedad relativa y presión barométrica, guardar dichos valores en memoria y enviarlos hacia la red vía UART, en primer lugar hacia el módulo inalámbrico de RF y después del módulo RF hacia el coordinador, cuando los datos sean solicitados.

Si el módulo es tipo router, los mensajes recibidos via RF pueden ser solo mensajes “de paso”, que solo debe leer y reenviar si no son dirigidos hacia ese módulo en particular.

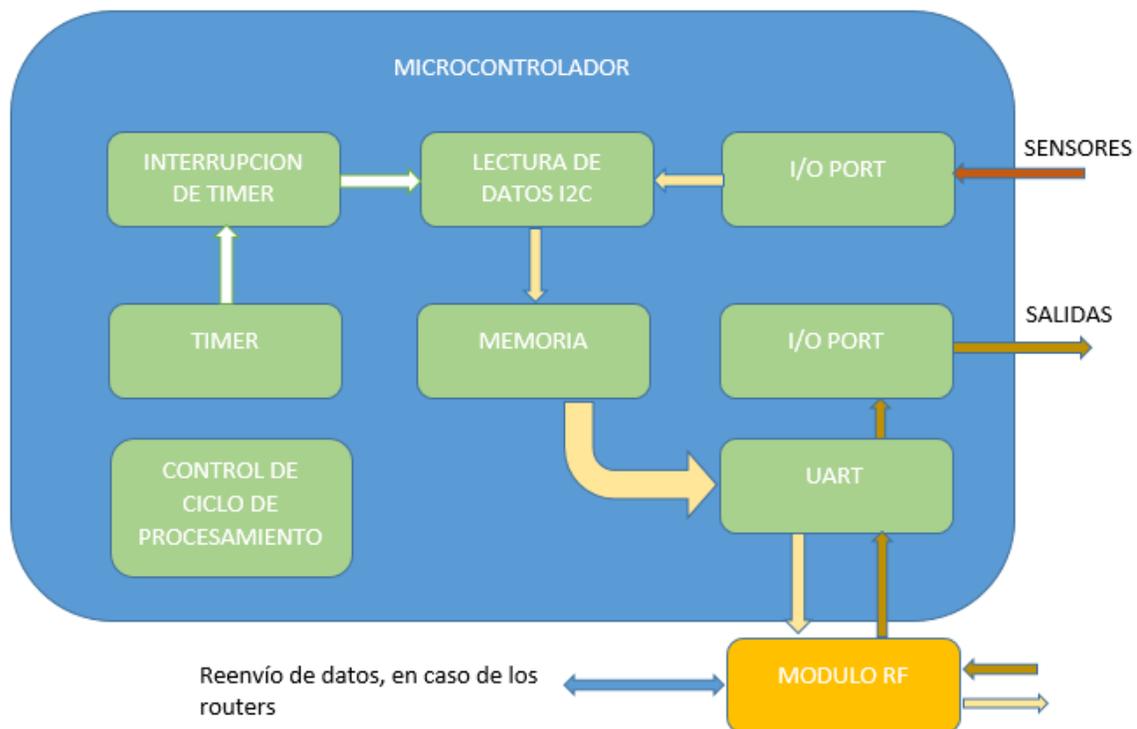


Figura 6. Arquitectura de software módulos router y final (Flujo de datos).

Fuente: Elaboración propia de acuerdo a los esquemas de arquitectura de software de sistemas embebidos.



Capítulo 3

Diseño del sistema de monitoreo, telemetría y telecontrol MTT-SS17



Capítulo 3. Diseño del sistema de monitoreo, telemetría y telecontrol MTT-SS17

En este capítulo se describe el diseño físico del MTT-SS17, inicia con una revisión técnica del hardware que se requiere para desarrollar el proyecto. El módulo consta de 3 módulos, de los cuales 2 requieren el mismo tipo de hardware y el otro (el módulo master) requiere un hardware más complejo, ya que requiere una mayor capacidad de procesamiento que los módulos router y final. Una vez definido el hardware requerido para diseñar el sistema, se procede a diseñar cada módulo, se realizan los diagramas esquemáticos, posteriormente el diseño de los pcsb y las listas completas de componentes para poder fabricar los prototipos.

El sistema MTT-SS17, consta de 3 tipos de módulos diferentes, un módulo coordinador que debe ser capaz de controlar el tráfico de datos (mediciones de temperatura, H.R. y presión barométrica) entre los módulos router o finales y el servicio WEB (UBIDOTS), por lo que debe tener capacidad de comunicación WiFi para el envío y recepción de datos hacia y desde el Web Service y capacidad de comunicación Xbee para el envío y recepción de datos desde los módulos router y finales, por lo que requiere un puerto UART. Debido a esto, su procesador debe tener una capacidad de procesamiento mayor al de los módulos router o finales, además de tener una capacidad de memoria para soportar posibles desconexiones temporales del Web Service.

Los módulos router y finales están contruidos de la misma manera (El hardware es el mismo), su diferencia radica en la funcionalidad dentro del esquema de red, por lo que la única distinción está en el firmware de configuración de los módulos de comunicación Xbee. En este caso, el procesador no necesita ser de alto grado de desempeño, solo requiere que contenga entradas y salidas digitales, manejo de protocolos seriales UART, ONE WIRE e I2C.

3.1 Plataformas de desarrollo

Existen en el mercado diferentes opciones de plataformas de desarrollo que tienen la capacidad técnica de hardware requeridas por el módulo coordinador, que entre sus principales características están la conectividad WiFi, GPIOs, UART, capacidad de memoria para almacenamiento de datos y procesador de alto rendimiento.

En la siguiente tabla se hace un comparativo de diferentes plataformas de desarrollo que cumplen con las características de hardware requeridas para la implementación del módulo coordinador.

Plataforma	Tipo de procesador y características principales.	Núcleos	Vel	Data Bus	Costo
Raspberry PI3	ARM V8, WiFi, 1MB ram, SD card, mini HDMI. SO Linux	QuadCore	1.2 GHz	64 bit	55.00 usd
Raspberry PI Zero	Broadcom BCM2835, 512 MB ram, 1 USB, WiFi, Bluetooth, SD Card. SO Linux. 40 GPIO	Single core	1.0 GHz	32 bit	7.00 usd
BBB	ARM Cortex-A8, 1 USB, SD Card, micro HDMI. SO Linux	Single Core	1 GHz	32 bit	90.00 usd
Radxa Rock	ARM Cortex-A9, 2048 MRAM, WiFi, 80 GPIO, 2 USB. SO Linux	Quad Core	1.6 GHz	32 bit	59.00 usd
Zybo	Xilinx Z-7010. Integra un ARM Cortex- A9. SO Linux	Dual Core	650MH z	32 bit	245.00 usd

Tabla 1. Comparativo de plataformas de desarrollo.

Fuente: Elaboración propia de acuerdo a investigación comparativa de tarjetas de desarrollo.

En base a esta tabla, la tarjeta de desarrollo más adecuada para la implementación del módulo coordinador, es la Raspberry PI Zero (Raspberry PI Foundation, 2017), ya que cuenta con los elementos de hardware requeridos, como lo son, sistema operativo, conectividad WiFi, UART, capacidad de procesamiento y memoria requeridos para la correcta operación del sistema MTT-SS17, módulo coordinador, y a un costo muy bajo.

3.1.1. Características de la Raspberry PI Zero

La Raspberry PI Zero es un computador completo, que basa su operación en una distribución Linux como sistema operativo, aunque han aparecido versiones de sistemas operativos como Windows o Android de terceros que parecen funcionar de manera adecuada. Las principales características de la Raspberry Pi Zero se enumeran a continuación:

- Procesador Broadcom BCM2835 @ 1Ghz ARM 11.
- 512 MB de memoria RAM LPDDR2.
- Ranura para tarjeta Micro-SD.
- Salida de vídeo mini-HDMI a 1080p.
- Dos conectores micro-USB para corriente e intercambio de datos.
- WiFi.
- Bluetooth.
- 40 pin GPIO Header
- Sistema Operativo: Raspbian Stretch Linux Distribution.
- Tamaño. L=65mm X W=30mm, A=5mm.
- Voltaje de alimentación: 5V.

Gracias a la distribución Linux en la que se basa su operación, se tienen varias herramientas de desarrollo dentro del sistema, tales como C++, OPenCV, Java, Scratch, Python, entre otras. Para este proyecto se utiliza Python como lenguaje de programación para el desarrollo de la aplicación.

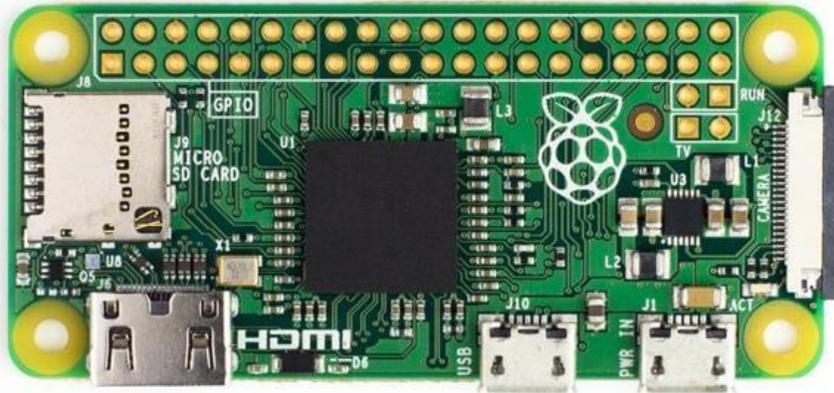


Figura 7. Raspberry Pi Zero.

Fuente: Elaboración propia, imagen de la tarjeta Raspberry pi zero.

3.2. Microcontroladores

Un microcontrolador es un circuito integrado programable, capaz de ejecutar instrucciones almacenadas en su memoria. Se compone de diferentes bloques funcionales, los cuales cumplen una tarea específica. Un microcontrolador incluye en su interior las tres principales unidades funcionales de una computadora: unidad central de procesamiento, memoria y periféricos de entrada/salida. Los microcontroladores están diseñados para reducir el costo económico y el consumo de energía de un sistema en particular. Por eso, el tamaño de la unidad central de procesamiento, la cantidad de memoria y los periféricos incluidos dependerán de la aplicación. Para el diseño e implementación de los módulos router y finales, las necesidades de hardware se reducen, en comparación al módulo coordinador, solo se requiere un procesador de 8 bits, con hardware para comunicación serial tanto síncrona como asíncrona, entradas y salidas digitales con capacidad para PWM. En el mercado existen diferentes opciones con estas características, entre ellas los microcontroladores Microchip, Atmel y TI.

A continuación se muestra una tabla comparativa de diferentes microcontroladores.

Características	Microchip PIC18F2550	ATMEL ATMEGA328	TI MSP 430G2XX3
CPU	8 bits	8 bits	16 bits
Memoria de programa	32Kb	32Kb	16 kb
EEPROM	256 bytes	1024 bytes	
SRAM	2 Kb	2 Kb	512Kb
SPI	1	1	1
I2C	1	1	1
USB	1	0	0
UART	1	1	1
PWM	2	6	2
ADC	10 BITS	10 BITS	10 BITS.
TIMERS	1X8 BITS, 3X16 BITS	2X8 BITS, 1X16 BITS	2 x 16 bits
COMPARADORES	2	1	1
VOLTAJE DE OP	2 a 5.5 V	1.8 a 5.5V	1.8 a 3.6V
E/S	24	23	24
Frecuencia de op.	48 MHz	20 MHz	25 MHz
Empaque	28 pin dip	28 pin dip	28 pin dip
Consumo de corriente			
5V @ 25 grados C. Run	2.2 mA (1 MHz)	5.2 mA (8 MHz)	2.8mA (8 MHz)
5V @ 25 grados C. Idle	1 mA (1 MHz)	1.2 mA (8MHz)	Menor a 1uA
5V @ 25 grados C. Sleep	2.0 uA	Menor a 1uA	Menor a 1uA
Precio.	4.50 usd	4.00 usd	3.50 usd
Plataforma de desarrollo.	MPLAB, PICC, MIKRO C	ATMEL STUDIO ARDUINO	Code Composer Studio

Tabla 2. Comparativo de microcontroladores de rango bajo.

Fuente: Elaboración propia de acuerdo a la investigación comparativa de microcontroladores realizada para este trabajo.

Para el desarrollo de los módulos router y final, se resolvió por utilizar el microcontrolador ATMEGA328 (ATMEL, 2015), ya que cumple con las características de hardware requeridas, principalmente el voltaje de operación, cantidad de salidas PWM y puertos serie SPI/I2C y UART. En cuanto a la plataforma de desarrollo (IDE), el ATMEGA trabaja tanto con ATMEL Studio como con Arduino. El uso del IDE de Arduino con el ATMEGA328 permite un tiempo de desarrollo más acelerado debido a la gran cantidad de librerías disponibles, lo que proporciona una gran ventaja sobre las demás plataformas u opciones disponibles.

3.2.1. Microcontrolador ATMEGA328 (ATMEL, 2015)

Este microcontrolador de 8 bits está basado en una arquitectura RISC, que combina memoria flash de 32Kbytes para programa, 1Kb de memoria EEPROM para datos y 2 KB de memoria RAM, 23 puertos de E/S de uso general y hardware periférico especializado como interrupciones internas y externas, puertos serie síncronos (SPI/I2C) y asíncrono (USART), ADC de 10 bits y 5 modos de ahorro de energía. Este dispositivo trabaja en un rango de voltaje de 1.8 a 5.5 volts. En la figura 8 se muestra la distribución de pines y sus funciones del microcontrolador ATMEGA328.

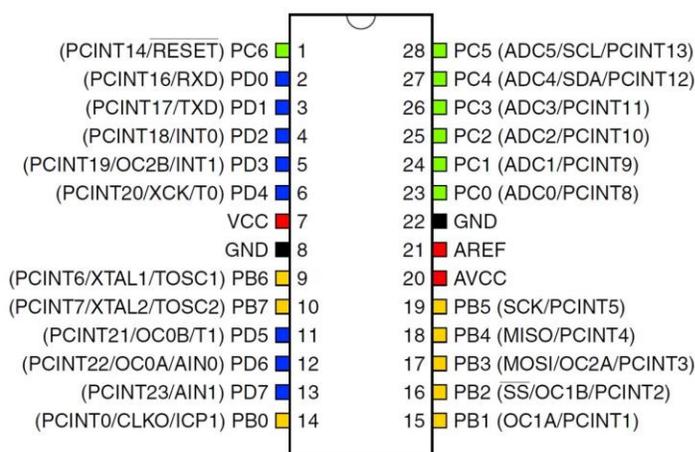


Figura 8. Distribución del microcontrolador ATMEGA328

Fuente. (ATMEL, 2015). Recuperado de

[http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42735-8-bit-AVR-](http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf)

[Microcontroller-ATmega328-328P_Datasheet.pdf](http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf)

3.3. Sensores

3.3.1. Sensor de temperatura y Humedad Relativa (HR) DHT22 (AM2302)

El DHT22 (AOSONG ELECTRONICS CO. LTD) es un sensor que permite mediciones de temperatura y humedad relativa (RH). El sensor posee una interfaz serial propietaria, que solo requiere de un pin para comunicarse con un microcontrolador. Los elementos de sensado vienen compensados y calibrados de fábrica. El coeficiente de calibración se guarda en una memoria OTP, dicho coeficiente es leído por el procesador interno cada vez que ejecuta una medición para mantener una buena precisión de lectura.

Sus principales características son:

Temperatura:

- Rango de operación: -40°C hasta 80°C de temperatura.
- Precisión: $\pm 0.5^\circ\text{C}$, $\pm 1^\circ\text{C}$ como máximo en condiciones adversas.
- Tiempo de respuesta: <10 segundos.
- Resolución decimal.
- Tiempo de muestreo: 2 segundos.

Humedad relativa:

- Rango de valores desde 0% hasta 99.9% de Humedad Relativa.
- Precisión: $\pm 2\%RH$, a una temperatura de 25°C.
- Tiempo de respuesta: <5 segundos, velocidad del aire debe ser de 1 m/s.

Parámetros eléctricos:

Voltaje de alimentación: 3.3V – 5V @ 2.5mA

Protocolo de comunicación: One wire.

3.3.2. Sensor de presión barométrica y temperatura BMP280

La presión atmosférica es la fuerza por unidad de superficie que ejerce el aire sobre la superficie terrestre. La presión atmosférica en un punto coincide numéricamente con el peso de una columna estática de aire de sección recta unitaria que se extiende desde ese punto hasta el límite superior de la atmósfera. Como la densidad del aire disminuye conforme aumenta la altura, no se puede calcular ese peso a menos que seamos capaces de expresar la variación de la densidad del aire ρ en función de la altitud z o de la presión p . Por ello, no resulta fácil hacer un cálculo exacto de la presión atmosférica sobre un lugar de la superficie terrestre. Además, tanto la temperatura como la presión del aire están variando continuamente, en una escala temporal como espacial, dificultando el cálculo.

Se puede obtener una medida de la presión atmosférica en un lugar determinado, pero con ella no se pueden obtener muchas conclusiones: es la variación de dicha presión a lo largo del tiempo lo que permite obtener una información útil que, unida a otros datos meteorológicos (temperatura atmosférica, humedad y vientos) da una imagen bastante acertada del tiempo atmosférico en dicho lugar e incluso un pronóstico a corto plazo del mismo.

El sensor BMP280 (BOSH, 2015) es un sensor de presión atmosférica de alta precisión. Es capaz de leer presión barométrica (absoluta) y temperatura. Por medio de cálculos matemáticos es capaz de detectar diferencia de alturas ya que la presión atmosférica o barométrica es inversamente proporcional a la altura sobre el nivel del mar, es decir, a medida que se sube, respecto al nivel del mar, decrece la presión atmosférica.

El sensor BMP280 tiene las siguientes características:

Rango de Medición: 300 – 1100hPA (Calibrado de fábrica).

Incluye medición de temperatura.

Voltaje de Alimentación. 1.8V – 3.6V Vdd / 1.62 – 3.6V Vddio.

Bajo consumo de energía: 5uA @ 1 muestra/seg.

Bajo ruido: 0.06hPa (Ultra low Power)

0.02hPa (Advanced Resolution Mode)

Interface: I2C

El sensor BMP280 consiste en un sensor piezo-resistivo, un convertidor análogo digital y una unidad de control con una memoria EEPROM, con una interfaz I2C. Los valores de temperatura y presión no están originalmente compensados. En la memoria EEPROM tiene almacenados 176 bits de datos para calibración, utilizados para compensar los distintos parámetros del sensor.

Este sensor está diseñado para conectarse directamente a un microcontrolador, vía protocolo I2C. En la siguiente figura se muestra el diagrama de conexión del sensor BMP280.

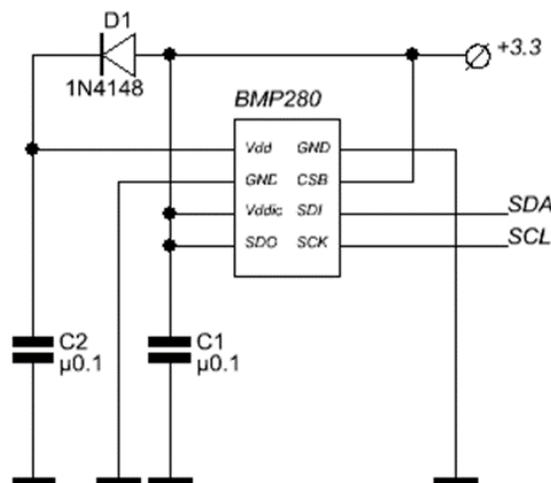


Figura 9. Esquema de conexiones del sensor BMP280.

Fuente. Elaboración propia de acuerdo al diseño basado en la hoja de datos del sensor BMP280.

3.4 Web Service UBIDOTS [1]

UBIDOTS es un servicio en la nube que nos permite almacenar e interpretar información de sensores en tiempo real, haciendo posible la creación de aplicaciones para el Internet de las Cosas. Este servicio se basa en lo que se denominan "Application Programming Interface" o APIs, a través de las cuales los objetos se conectan e interactúan con la red.

3.4.1. Protocolo MQTT en UBIDOTS

MQTT significa MQ Telemetry Transport. Se trata de un protocolo de mensajería de publicación / suscripción (Publish/Suscribe), extremadamente simple y ligero, diseñado para dispositivos con restricciones de recursos, tales como memoria o consumo de energía, entre otros, trabajando en redes de bajo ancho de banda, alta latencia o no confiables.

Los principios de diseño de este protocolo son minimizar el ancho de banda de la red y los requerimientos de recursos del dispositivo, al tiempo que también intenta asegurar la confiabilidad y cierto grado de seguridad de la entrega. Estos principios también se convierten en el ideal de protocolo del emergente mundo de los dispositivos conectados ("máquina a máquina" (M2M) o "Internet de las cosas"), y para aplicaciones móviles donde el ancho de banda y la energía de la batería están muy limitados.

MQTT es un protocolo extremadamente simple y ligero. La conexión a un servidor sólo necesita unos 80 bytes. El dispositivo permanece conectado todo el tiempo, cada "publicación" (Método Publish) de datos (datos desde el dispositivo a servidor) y la suscripción (Método Subscribe) de datos (datos del servidor al dispositivo) es de unos 20 bytes. Ambos ocurren casi instantáneamente.

Proporciona una forma escalable y rentable de conectar dispositivos a través de Internet. Es capaz de entregar mensajes en tiempo casi real y garantiza su entrega. Conecta miles de dispositivos, enviando actualizaciones instantáneas y notificaciones, que es donde MQTT realmente sobresale.

Es un protocolo abierto, estandarizado por OASIS e ISO (ISO / IEC 20922: 2016). Es ampliamente adoptado por muchos dispositivos y empresas de todo el mundo para el intercambio de datos con dispositivos restringidos o productos en el campo. MQTT mantiene el ancho de banda en un mínimo absoluto y se ocupa de las redes poco fiables sin esquemas complejos de corrección de errores o un enorme esfuerzo de implementación.

MQTT se basa en el paradigma de publish/suscribe. Esto facilita la difusión de mensajes de un editor a muchos suscriptores. Los mensajes de una gran cantidad de editores a algunos suscriptores también están cubiertos. Además, la conexión se construye desde el lado del cliente, lo que hace posible una comunicación bidireccional sin problemas de traducción NAT.

Se puede utilizar un nombre de usuario y una contraseña con un paquete MQTT en la Version 3.1 del protocolo. La encriptación a través de la red se puede manejar con SSL, independientemente del protocolo MQTT en sí mismo (vale la pena señalar que SSL no es el más ligero de los protocolos, y agrega una sobrecarga de red significativa). La seguridad adicional puede ser agregada por una aplicación que cifra los datos que envía y recibe, pero esto no es algo incorporado al protocolo, para mantenerlo simple y ligero.

3.4.2. Protocolo HTTP en UBIDOTS

De la misma manera en que las personas usan un navegador para visitar páginas web a través de URLs, botones y campos de texto, los dispositivos tienen su propia forma de interactuar con los sistemas web, necesitan una “página web especial”, con una estructura estandarizada y unos comandos preestablecidos. Éstas “páginas web para dispositivos” son lo que se conoce como un API REST, “Interfaz de Programación de Aplicaciones” (Application Programming Interface) y establece cómo diferentes componentes de un sistema de software deben interactuar entre ellos.

Estas API's pueden utilizar el protocolo HTTP, el mismo protocolo que usan los navegadores para comunicarse con las páginas web. El API de UBIDOTS implementa los principales cuatro métodos de HTTP:

GET Para la lectura de información

POST Para la creación de información

PUT Para la edición de información

DELETE Para el eliminado de información

Finalmente, un API también debe especificar el formato de los datos. Los más comunes son XML y JSON, siendo éste último el más popular, en gran parte por su simplicidad.

3.5 Red tipo malla (Mesh Network)

Una red de malla es una topología en la que cada nodo de la red está conectado a otros nodos alrededor de él. Cada nodo coopera en la transmisión de información.

Las redes de malla proporcionan tres beneficios importantes:

Enrutamiento. Con esta técnica, el mensaje se propaga a lo largo de una ruta saltando de nodo hasta que alcance su destino final.

Creación de redes ad-hoc. Se trata de un proceso automatizado que crea toda una red de nodos sobre la marcha, sin ninguna intervención humana.

Auto-corrección. Este proceso calcula automáticamente si uno o más nodos de la red están “desaparecidos” y reconfigura la red para reparar cualquier ruta rota.

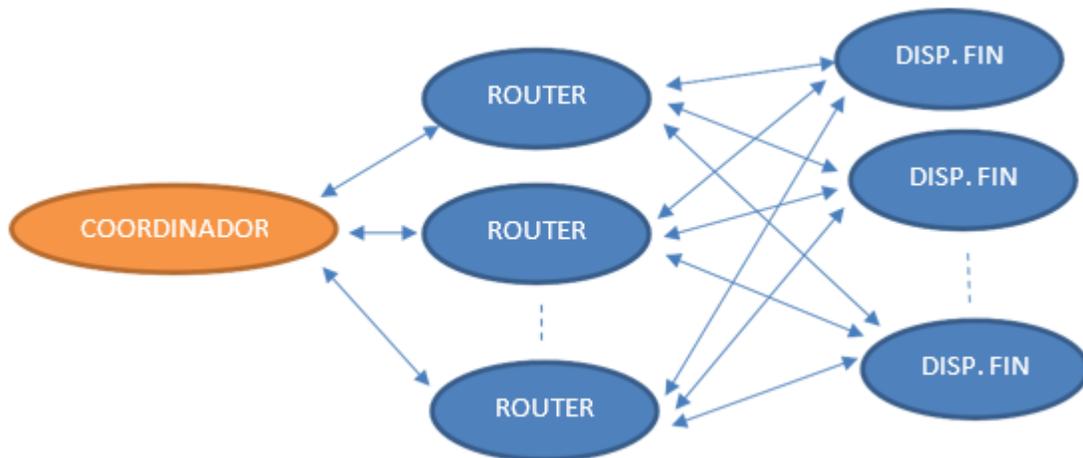


Figura 10. Arquitectura de la red tipo malla.

Fuente. Elaboración propia de acuerdo a la documentación referente a las redes tipo mesh.

Con redes de malla, la distancia entre dos nodos no importa, siempre y cuando haya suficientes nodos entre ellos para transmitir el mensaje. Cuando un nodo quiere comunicarse con otro, la red calcula automáticamente la mejor ruta. Una red de malla es confiable y ofrece redundancia. Si un nodo ya no puede funcionar, por ejemplo porque se ha eliminado de la red o porque una barrera bloquea su capacidad de comunicación, el resto de los nodos pueden comunicarse entre sí, ya sea directamente o a través de nodos intermediarios. ZigBee (DIGI, 2016) define tres tipos de dispositivos diferentes:

- a) Coordinador. Módulo MITT-SC17 Master.
- b) Enrutador (Router). Módulo MITT-SC17 Slave R.
- c) Dispositivo final. (End Device). Módulo MITT-SC17 E.

3.5.1. Módulo coordinador

Las redes ZigBee siempre disponen de un único dispositivo coordinador. Este dispositivo:

- Inicia la red, seleccionando el canal y PAN ID.
- Distribuye direcciones, permitiendo que routers y dispositivos finales se unan a la red. Ayuda en el enrutamiento de datos.
- Almacena los paquetes de datos inalámbricos para los dispositivos en modo “sleep” y dispositivos finales.
- Administra las otras funciones que definen la red, la protegen y la mantienen en operación. Este dispositivo no puede entrar a modo “sleep” y debe estar encendido en todo momento.

3.5.2. Módulo Router

Un enrutador es un nodo ZigBee con todas las funciones. Este dispositivo:

- Puede unirse a redes existentes y enviar, recibir y enrutar información. El enrutamiento consiste en actuar como un mensajero para las comunicaciones entre otros dispositivos que están demasiado lejos para transmitir información por su cuenta.
- Puede almacenar en búfer paquetes de datos inalámbricos para dispositivos finales o en modo sleep. Puede permitir que otros routers y dispositivos finales se unan a la red.

3.5.3. Módulo final

Un dispositivo final es esencialmente una versión reducida de un enrutador. Este dispositivo:

- Puede unirse a las redes existentes y enviar y recibir información, pero no puede actuar como mensajero entre otros dispositivos.
- No se puede permitir que otros dispositivos se unan a la red.
- Utiliza hardware menos costoso y puede apagarse intermitentemente, ahorrando energía entrando temporalmente en un modo de suspensión.
- Siempre necesita un enrutador o el coordinador para ser su dispositivo “padre”. El “padre” ayuda a los dispositivos finales a unirse a la red y almacena mensajes para ellos cuando están en modo “sleep”.

Las redes ZigBee pueden tener cualquier número de dispositivos finales. De hecho, una red puede estar compuesta por un coordinador, múltiples dispositivos finales y cero routers.

3.6. Diseño físico del MTT-SS17

3.6.1. Módulo coordinador

El módulo coordinador fue implementado con base en la plataforma de Raspberry Pi Zero, con sistema operativo Linux, distribución Raspbian. El lenguaje de programación para el desarrollo de la aplicación es Python 2.7, a través del cual se coordinan las comunicaciones WiFi, utilizando protocolo http y las comunicaciones hacia los módulos router y finales, mediante el esquema de red tipo malla Xbee, a través del UART de los GPIO de las RBPi.

3.6.2 Módulos router y final

El hardware para ambos módulos es exactamente el mismo, solo cambia el firmware de configuración de los módulos Xbee. Se utiliza como procesador base el microcontrolador ATMEGA328 de Atmel, los sensores DHT22, de humedad y temperatura, el BMP280 de Temperatura y presión barométrica, y los módulos Xbee de la serie 2, para el envío de datos.

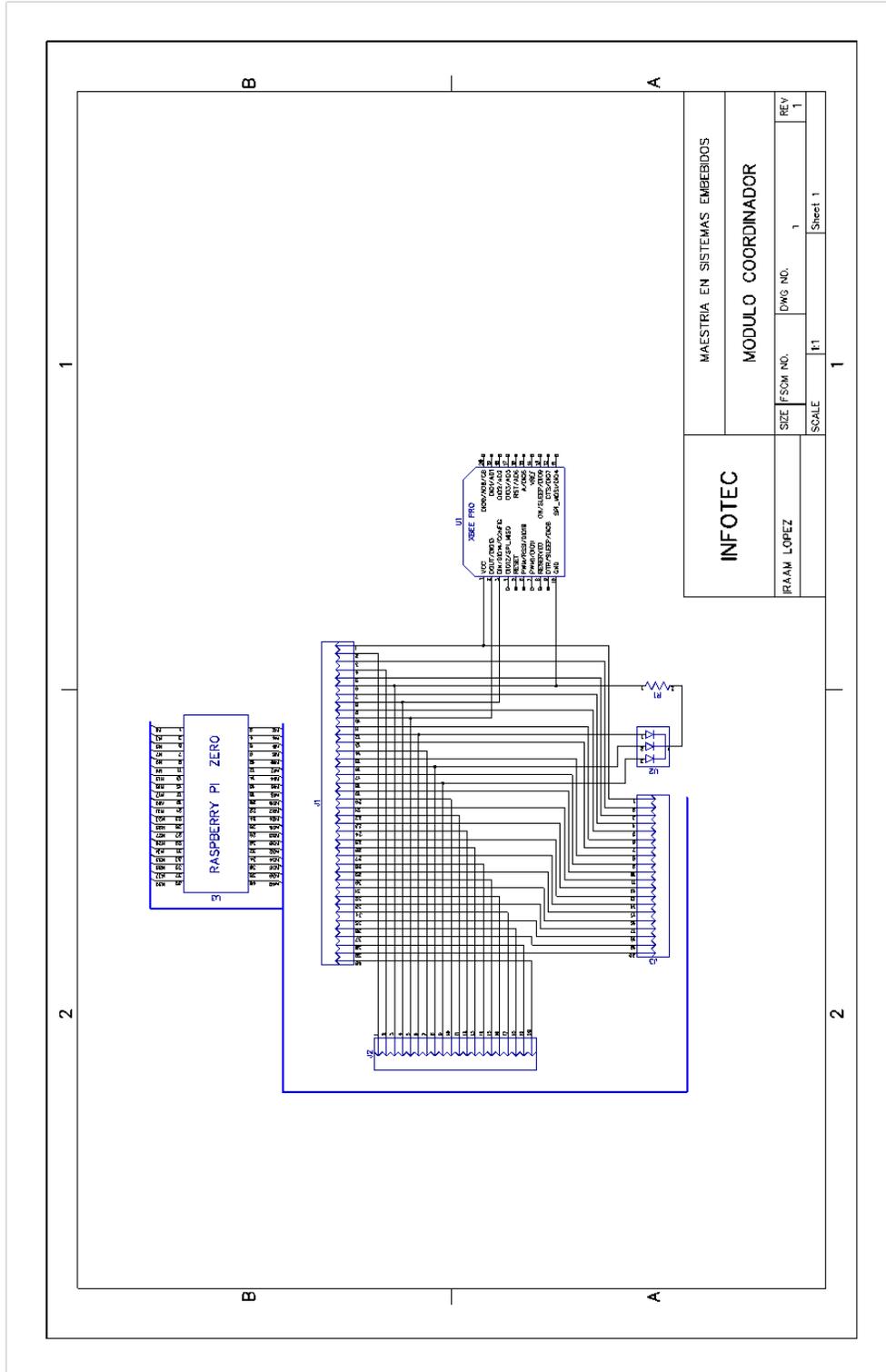
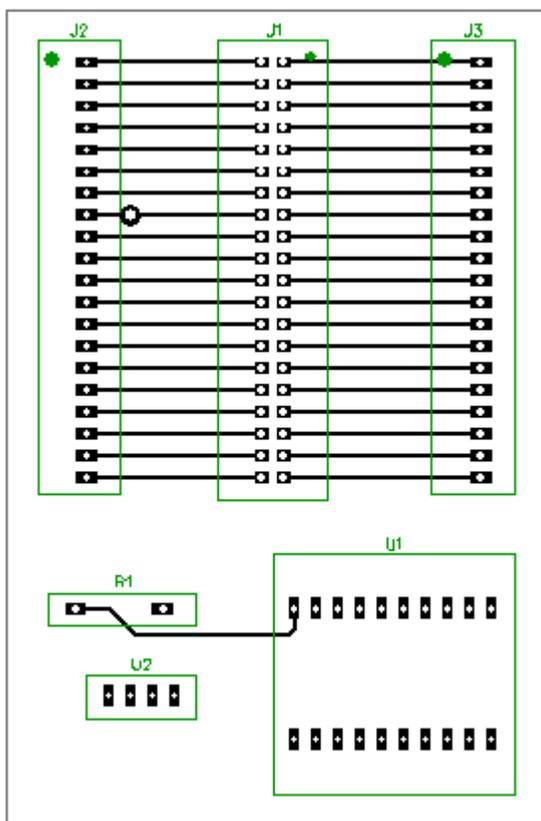


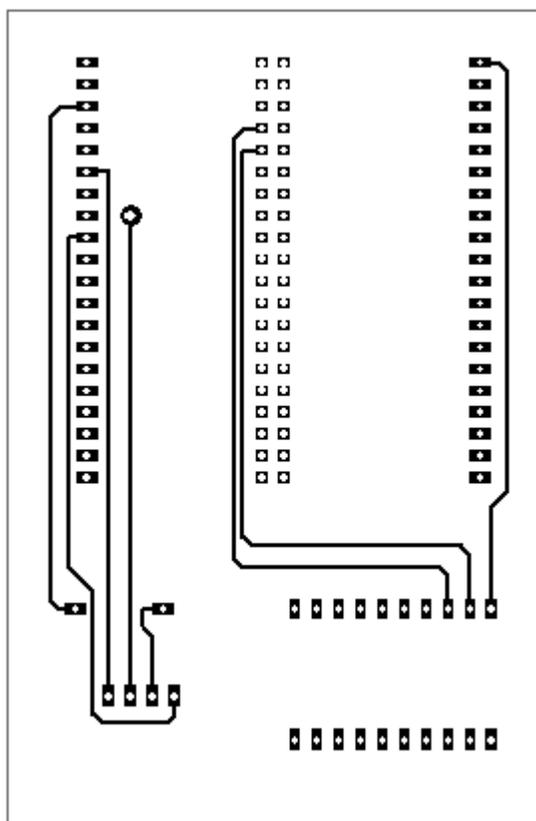
Figura 11. Diagrama esquemático del módulo coordinador.

Fuente: Elaboración propia de acuerdo a la documentación del diseño del sistema

MTT-SS17.



Top



Bottom

Figura 12. PCB del módulo coordinador.

Fuente: Elaboración propia de acuerdo al diseño del PCB para el sistema MTT-SS17.

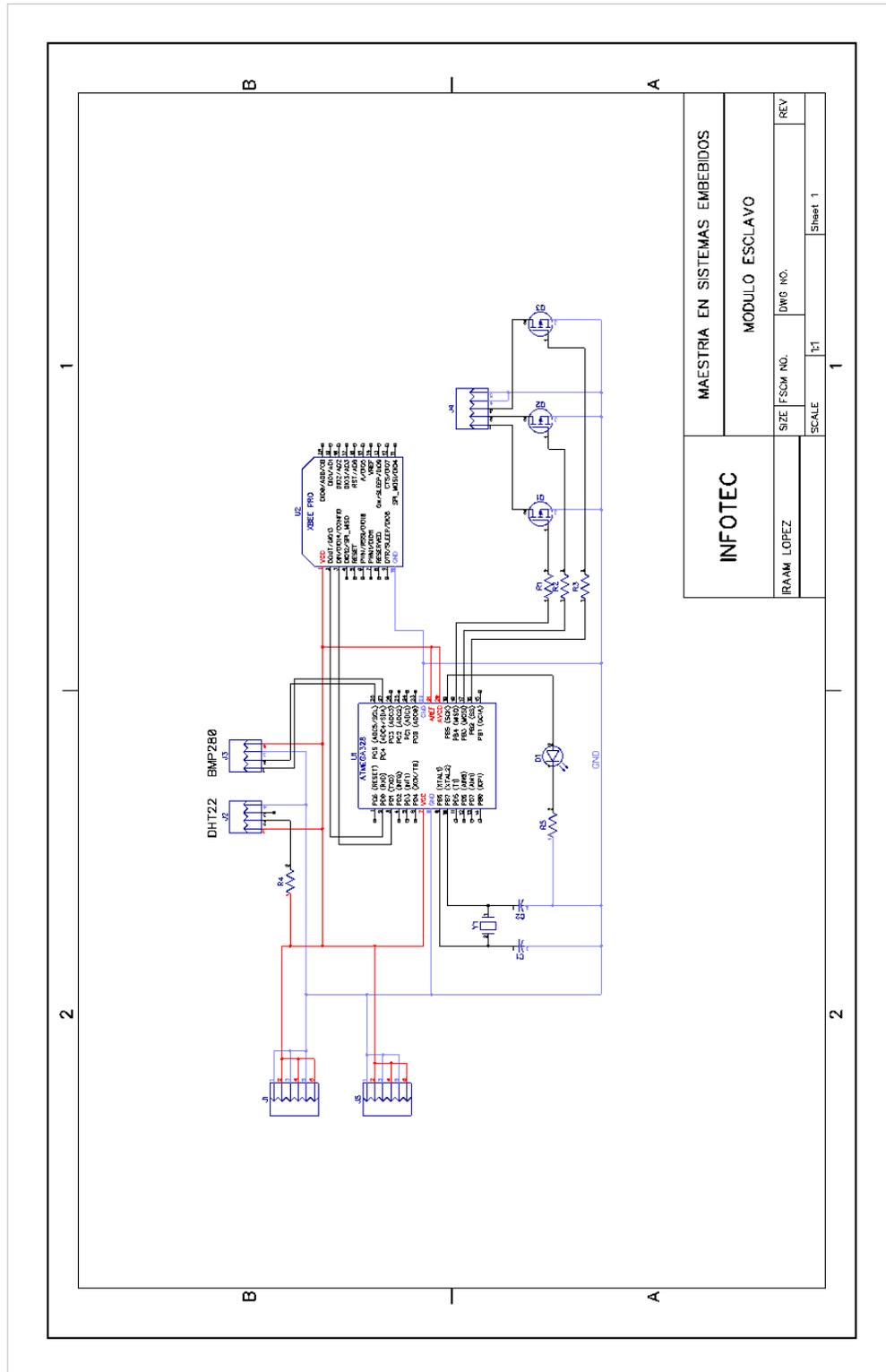
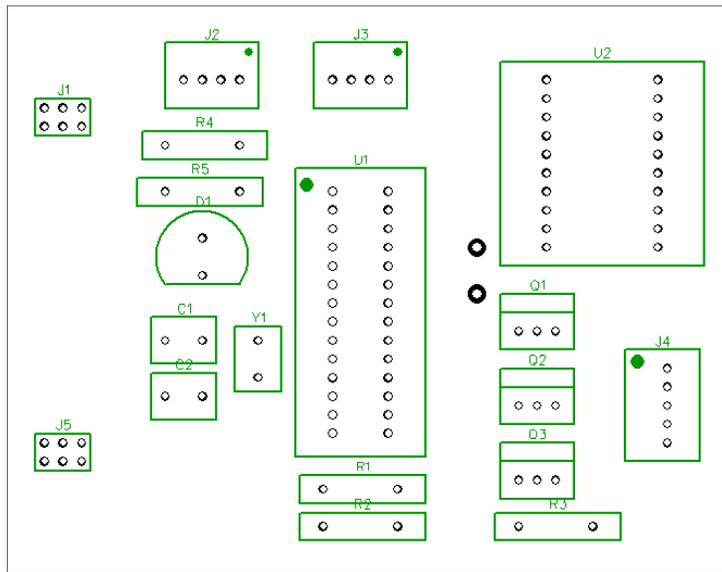


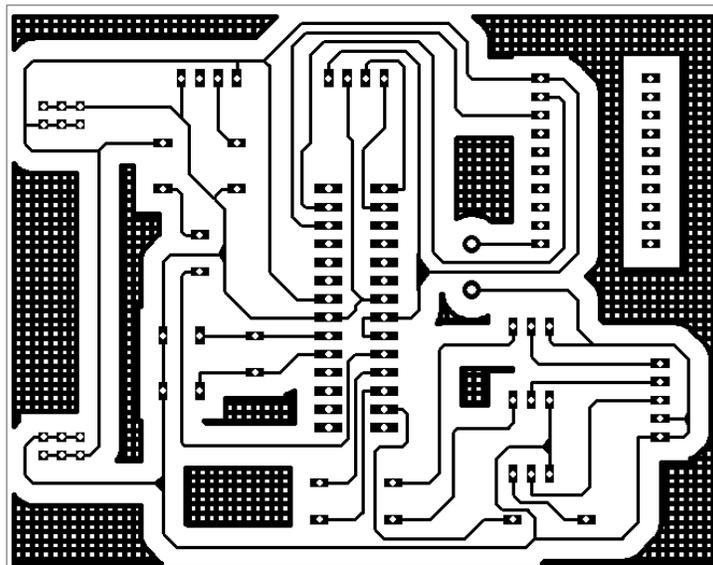
Figura 13. Diagrama esquemático del módulo router y/o final.

Fuente: Elaboración propia de acuerdo a la documentación del diseño del sistema

MTT-SS17.



Top.



Bottom.

Figura 14. PCB del módulo router y/o final.

Fuente. Elaboración propia de acuerdo al diseño del PCB para el sistema MTT-SS17.

3.7. Listas de Materiales (BOM)

A continuación se muestran las listas de componentes de los diferentes módulos del MTT-SS17. Solo se muestran los precios de los componentes, no se incluye ningún otro costo de producción o administrativo.

3.7.1. Lista de materiales del módulo coordinador

No	Descripción.	No. Parte.	Proveedor.	Cantidad	Precio Unitario USD	Precio Total. USD
1	Raspberry PI Zero	3400	Adafruit	1	10.00	10.00
2	Módulo Xbee S2	602-1967-ND	DigiKey	1	17.50	17.50
3	Regulador de voltaje 5.0V, 2A.	NA	Idesain	1	10.00	10.00
4	Resistencia 220 ohms	220 pr1/2	Steren	1	0.05	0.05
5	Conector header 40 pines	424-40-PIN-HEADER	Steren	3	2.50	7.50
6	LED Verde	NA	Ali Express	1	0.05	0.05
7	LED Rojo	NA	Ali Express	1	0.05	0.05
8	LED Amarillo	NA	Ali Express	1	0.05	0.05
9	PCB	NA	Gama Imp	1	12.50	12.50
10	Gabinete	GP-03	Steren	1	3.50	3.50
11	Tornillos		Tornillos Rangel	1	0.10	0.10
12	Tuercas		Tornillos Rangel	4	0.05	0.20
					TOTAL USD	61.50

Tabla 3. Lista de materiales BOM, módulo coordinador.

Fuente. Elaboración propia de acuerdo a información obtenida de proveedores de componentes electrónicos.

3.7.2. Lista de materiales del módulo router y/o final

No.	Descripción.	No. Parte.	Proveedor.	Cantidad	Precio Unitario. USD	Precio Total. USD
1	ATMEGA328	ATMEGA328	Adafruit	1	2.00	2.00
2	Módulo Xbee S2	602-1967-ND	DigiKey	1	17.50	17.50
3	Regulador de voltaje 5.0V, 3.3V, 2A.	NA	Idesain	1	10.00	10.00
4	Resistencia 220 ohms	220 pr1/2	Steren	6	0.05	0.30
5	Conector header 40 pines	424-40-PIN-HEADER	Steren	3	2.50	7.50
6	LED Verde	NA	Ali Express	1	0.05	0.05
7	LED Rojo	NA	Ali Express	1	0.05	0.05
8	LED Azul	NA	Ali Express	1	0.05	0.05
9	PCB	NA	Gama Imp	1	12.50	12.50
10	Sensor de Temp. y HR	DHT22	ML	1	4.50	4.50
11	Sensor de Temp. y Presión barométrica	BMP280	ML	1	6.00	6.00
13	IGBT 400V, 10 A	IRF740	ML	3	2.50	7.50
14	Conector de bloque 6 pin	NA	ML	1	5.00	5.00
15	Batería recargable con celda solar, 5000mA	NA	ML	1	13.00	13.00
10	Gabinete	GP-03	Steren	1	3.50	3.50
					TOTAL USD	89.45

Tabla 4. Lista de materiales BOM, módulo router y/o final.

Fuente. Elaboración propia de acuerdo a información obtenida de proveedores de componentes electrónicos.

3.8. Esquema de Mantenimiento

La finalidad del mantenimiento preventivo es: Encontrar y corregir los problemas menores antes de que estos provoquen fallas. El mantenimiento preventivo puede ser definido como una lista completa de actividades, todas ellas realizadas por usuarios, operadores, y mantenimiento para asegurar el correcto funcionamiento de la planta, edificios, máquinas, equipos, vehículos, etc.

El mantenimiento preventivo se diseña con la idea de prever y anticiparse a los fallos de las máquinas y equipos, utilizando para ello una serie de datos sobre los distintos sistemas y sub-sistemas e inclusive partes del mismo.

Bajo esa premisa se diseña el programa con frecuencias calendario o uso del equipo, para realizar cambios de sub-ensambles, cambio de partes, reparaciones, ajustes, etc., a maquinaria, equipos e instalaciones y que se considera importante realizar para evitar fallos.

Es importante trazar la estructura del diseño incluyendo en ello las componentes de conservación, confiabilidad, mantenibilidad, y un plan que fortalezca la capacidad de gestión de cada uno de los diversos estratos organizativos y empleados sin importar su localización geográfica, ubicando las responsabilidades para asegurar el cumplimiento.

Haciendo uso de los datos se hace una planeación esperando con ello evitar los paros y obtener con ello una alta efectividad del equipo, los conceptos de este mantenimiento se agrupan en dos categorías: preventivo y correctivo.

3.8.1. Mantenimiento preventivo

El mantenimiento preventivo se refiere a reemplazos, adaptaciones, restauraciones, inspecciones, evaluaciones, etc. hechas en períodos de tiempos por calendario o uso de los equipos. (Tiempos dirigidos). El mantenimiento preventivo podrá, en un futuro ser potencialmente mejorado, por medio de la incorporación de un programa de Mantenimiento Predictivo. Dentro del mantenimiento planeado se contempla el mantenimiento predictivo.

El Mantenimiento Correctivo se utilizará como la acción que emana de los programas de mantenimiento preventivo y predictivo (Tiempos dirigidos y Condiciones dirigidas de los equipos).

EL MTT-SS17 basa su operación en sistemas digitales que tienen una alta confiabilidad, por lo que el mantenimiento preventivo es mínimo. Con respecto a los sensores, estos requieren mayor mantenimiento preventivo debido a su menor tiempo de vida, comparada con el sistema digital de procesamiento de datos. Otro elemento que requiere atención es la batería recargable.

Los sensores, por estar expuestos al medio ambiente para poder realizar su función están expuestos a contaminantes como polvo, químicos, solventes, polen, etc. Los elementos de sensado, a su vez, tienen un tiempo de vida limitado. Debido a que la estabilidad de los sensores, se ven afectados con el tiempo, se deben calibrar los sensores cada año y sustituirse cada 2 años si está expuesto al aire libre, o 5 años si se usa en interiores.

El otro elemento a revisión dentro del programa de mantenimiento es la batería recargable, que debe cambiarse a los 3 años para asegurar su operación continua.

Es altamente recomendable limpiar con aire a presión la tarjeta del sistema digital cada vez que se hace el mantenimiento preventivo y calibración de los sensores.

3.8.2. Mantenimiento correctivo

- En caso de que el sistema deje de funcionar, se debe revisar lo siguiente.
- Hacer una inspección visual del sistema, buscando si el sistema tiene daños por golpes, quemaduras, humedad excesiva.
- Si todo se ve correcto, resetear el sistema.
- Si continua sin funcionar, asegurarse que existe el nivel de voltaje de alimentación correspondiente a la entrada del regulador de voltaje.
- Asegurarse que voltaje de operación (3.3 Volts) existe a la salida del regulador de voltaje. Si el voltaje es correcto, resetear el sistema digital y probar el funcionamiento.
- Si el sistema transmite, pero los datos son erróneos, revisar las conexiones y el estado de los sensores. Si el sistema no responde, reprogramar los procesadores digitales.



Capítulo 4

Implementación del Sistema de monitoreo, telemetría y telecontrol MTT-SS17



Capítulo 4. Implementación del MTT-SS17

En este capítulo se muestran los prototipos de los diferentes módulos implementados físicamente.

4.1 Módulo coordinador

La implementación del módulo coordinador del sistema MTT-SS17 se realizó en primera instancia con una tarjeta raspberry pi zero y un módulo Xbee S2, únicamente alambrando las conexiones, para realizar las primeras pruebas de funcionamiento. El programa se desarrolló en programación Python que es un lenguaje de programación que viene nativo en el sistema operativo Raspbian stretch.

A continuación se muestran imágenes de los primeros prototipos.



Figura 15. Raspberry PI Zero y accesorios

Fuente. Elaboración propia, imagen obtenida durante la implementación del proyecto.



Figura 16. Raspberry PI Zero y Xbee.

Fuente. Elaboración propia, imagen obtenida durante la implementación del proyecto.

4.2. Módulos router y finales

El hardware para ambos módulos es exactamente el mismo, solo cambia el firmware de configuración de los módulos Xbee. Se utiliza como procesador base el microcontrolador ATMEGA328 de Atmel, los sensores DHT22, de humedad y temperatura, el BMP280 de Temperatura y presión barométrica, y los módulos Xbee de la serie 2, para el envío de datos.

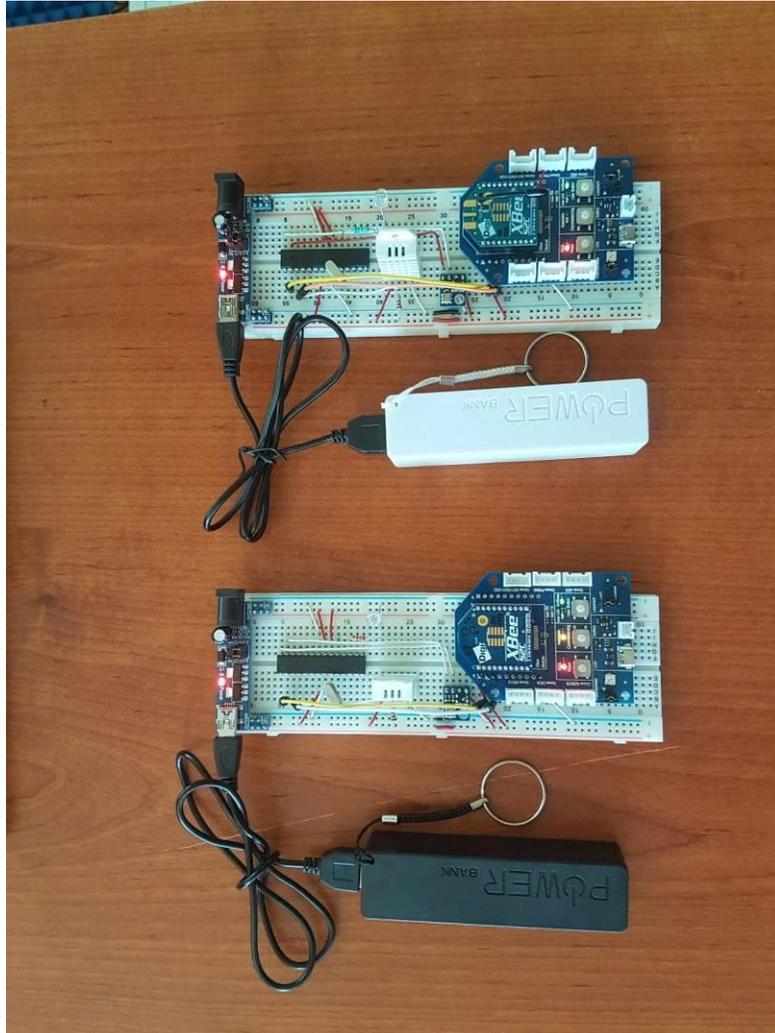


Figura 17. Módulos Router y Final implementados en protoboard.
Fuente. Elaboración propia, imagen obtenida durante la implementación del proyecto.



Capítulo 5

Resultados

Capítulo 5. Resultados

En este capítulo se muestran los resultados obtenidos de la implementación de los diferentes módulos del MTT-SS17. Se hace la conexión entre los módulos coordinador, router y final. El módulo coordinador recibe información de los módulos router y final, los cuales podemos observar si conectamos un monitor al módulo coordinador, y los reenvía al web service de UBIDOTS. A su vez recibe información del web service que el MTT-SS17 utiliza para controlar salidas digitales en los módulos router y finales. A continuación se muestran imágenes tanto de los datos en UBIDOTS como en el monitor del módulo coordinador.

5.1. Recepción de datos en el módulo coordinador

A continuación se muestra una imagen de los datos recibidos por la RBPi desde los módulos Router y finales implementados.

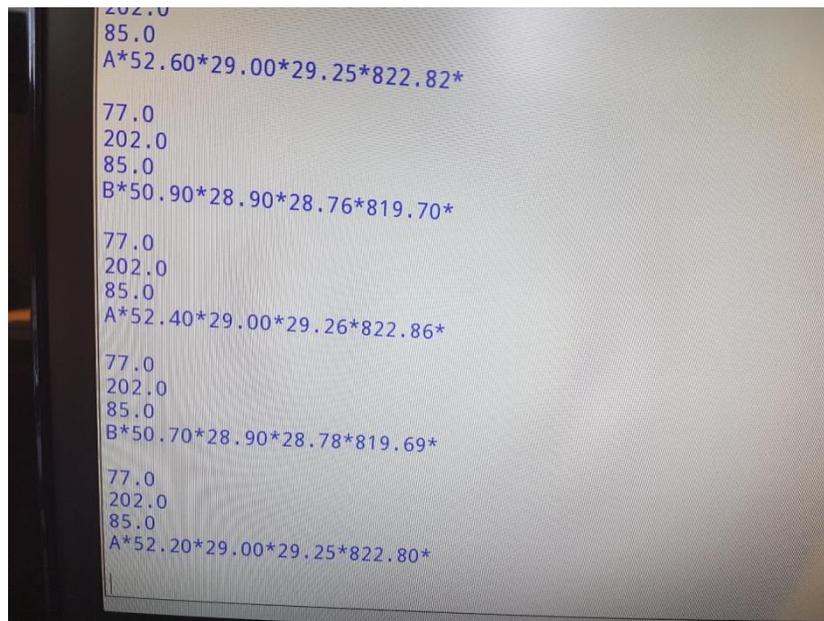


Figura 18. Datos recibidos por la RBPi Zero enviados desde los módulos Router y Final.

Fuente. Elaboración propia, imagen obtenida durante la implementación del proyecto.

5.2. Recepción de datos en el web service de UBIDOTS



Figura 19. Datos recibidos en el web service de UBIDOTS, enviados desde el módulo coordinador.

Fuente. Elaboración propia, imagen obtenida durante la implementación del proyecto.

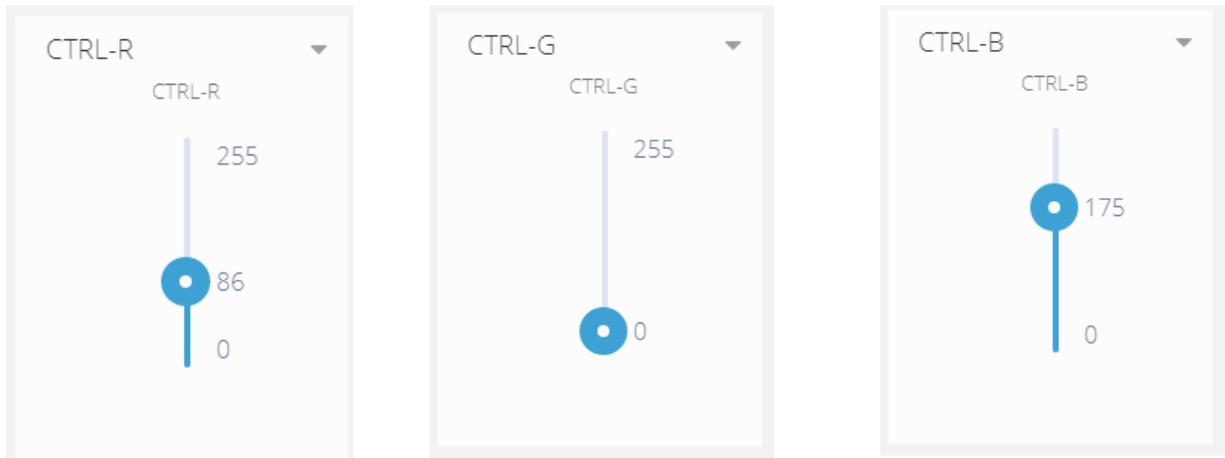


Figura 20. Controles implementados en el web service de UBIDOTS.

Fuente. Elaboración propia, imagen obtenida durante la implementación del proyecto.



Conclusiones



Conclusiones

Se diseñó un sistema de monitoreo, telemetría y telecontrol, con capacidad para medir variables ambientales, tales como temperatura, humedad relativa y presión barométrica, con posibles aplicaciones en la medición del impacto en zonas urbanas, por ejemplo.

Fue posible enviar los datos recabados por la red de sensores hacia la plataforma UBIDOTS, vía WiFi, para su registro y análisis, basados en un hardware robusto y a un precio competitivo respecto a productos similares. Además, en este proyecto fue posible realizar la recepción de datos desde el web service de UBIDOTS hacia los módulos router y final, donde se pudieron realizar acciones de control a través de los puertos digitales de dichos módulos.

Durante el desarrollo del proyecto se presentaron retos a superar. El realizar proyectos de IoT conlleva no solo diseñar hardware de procesamiento y sensores, sino que involucra entender el esquema de red de la www (internet), direccionamientos, protocolos, capas físicas y lógicas, servidores, clientes, puertos de acceso, también involucra el análisis de consumos de energía de los módulos de hardware, comunicaciones inalámbricas, etc. La complejidad del desarrollo de IoT es alta, pero se puede facilitar el desarrollo de prototipos utilizando librerías de software y esquemas de hardware libres, que aunque con ciertas limitaciones, permiten acelerar el proceso de prototipado de los proyectos. Llevar un prototipo a producto final conlleva una complejidad mayor.

Bibliografía

- AOSONG ELECTRONICS CO. LTD. (s.f.). *DHT22 Digital HR and temperature sensor/module*.
- BOSH. (2015). *BMP280 DIGITAL PRESSURE SENSOR*. Germany: BOSH SENSORTEC.
- GRUPO ENEL. (18 de nov de 2016). *Endesa Educa*. Obtenido de <http://www.endesaeduca.com/>
- INC., A. (2015). ATMEL 8-Bit microcontroller with 4/8/16/32 kbytes, In System Programmable Flash. San José, California, Unites States.
- INC., D. I. (December de 2016). XCTU. *CONFIGURATION AND TEST UTILITY*. Minnetonka, Minnesota, Unites States.
- Raspberry PI Foundation. (2017). *www.raspberrypi.org*. Obtenido de <https://www.raspberrypi.org/>
- UBIDOTS. (16 de Diciembre de 2016). <https://ubidots.com/>. Obtenido de <https://ubidots.com/docs/api/index.html#rest-api-reference>