



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA ELECTRÓNICA EN TELECOMUNICACIONES Y REDES

**“IMPLEMENTACIÓN DE UN PROTOTIPO ELECTRÓNICO
AUTOSUSTENTABLE GEO-LOCALIZADO CON NODOS
INALÁMBRICOS PARA LA UBICACIÓN DE MASCOTAS
MEDIANTE UNA RED DE SENSORES”**

Trabajo de titulación:

Tipo: PROPUESTA TECNOLÓGICA

Presentado para optar al grado académico de:

INGENIERO EN ELECTRÓNICA, TELECOMUNICACIONES Y REDES

AUTORA: DORIS JIMENA AVILA TACURI

TUTOR: ING. EDWIN VINICIO ALTAMIRANO SANTILLÁN

Riobamba-Ecuador

2019

©2019, Doris Jimena Avila Tacuri

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

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA EN ELECTRÓNICA, TELECOMUNICACIONES Y REDES

El Tribunal del Trabajo de Titulación certifica que: la investigación: “IMPLEMENTACIÓN DE UN PROTOTIPO ELECTRÓNICO AUTOSUSTENTABLE GEO-LOCALIZADO CON NODOS INALÁMBRICOS PARA LA UBICACIÓN DE MASCOTAS MEDIANTE UNA RED DE SENSORES.”, de responsabilidad de la señorita Doris Jimena Avila Tacuri, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Titulación, quedando autorizada su presentación.

NOMBRE	FIRMA	FECHA
Ing. Washington Luna. DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA	_____	_____
Ing. Patricio Romero DIRECTOR ESCUELA DE INGENIERÍA EN ELECTRÓNICA, TELECOMUNICACIONES Y REDES	_____	_____
Ing. Edwin Vinicio Altamirano Santillán DIRECTOR DE TRABAJO DE TITULACIÓN	_____	_____
Ing. Oswaldo Martínez MIEMBRO DEL TRIBUNAL	_____	_____

Yo, Doris Jimena Avila Tacuri soy responsable de las ideas, doctrinas y resultados expuestos en este Proyecto de Titulación y el patrimonio intelectual del Proyecto de Titulación pertenece a la Escuela Superior Politécnica de Chimborazo.

Doris Jimena Avila Tacuri

010379007-7

DEDICATORIA

Dedico este trabajo a Dios y la Virgen del Cisne por guiarme, cuidarme y haberme guiado por el camino correcto. A mis padres Adrián y Lupe por su amor, paciencia apoyo incondicional quienes con sus consejos y millones de palabras de aliento en los momento más duros de la carrera estuvieron siempre pendiente para no decaer, a mi hermano por su compañía y apoyo incondicional.

A todos mis amigos por ser como una familia en el transcurso de la carrera ya sea en los momentos más difíciles y alegres que se pasó en las aulas de clases, quienes siempre tendrán un espacio en mi corazón y en los recuerdos.

Doris Avila Tacuri

AGRADECIMIENTO

Un agradeciendo muy especial a Dios y a la Virgen del Cisne por darme la vida y esa fuerza y valor para seguir adelante a pesar de los malos momentos en el transcurso de mi carrera y así poder cumplir con una de mis metas.

El agradecimiento más grande es para mis padres y hermano por estar siempre pendiente de mí por su amor paciencia y por ser el motor fundamental de mi vida, y sobre todo por el apoyo y confianza que han puesto sobre mí.

Agradezco a todos mis amigos que estuvieron hasta el final de este camino. A la ESPOCH por abrirme las puertas y a mi escuela, Ingeniería Electrónica en Telecomunicaciones y Redes, a todos los docentes en especial al Ing. Edwin Altamirano Santillán por su guía y por la confianza para poder finalizar con éxito.

Doris Avila Tacuri.

TABLA DE CONTENIDO

RESUMEN	xiv
ABSTRACT	xv
INTRODUCCIÓN	1
CAPÍTULO I	
1. MARCO TEÓRICO.....	6
1.1. Mascota	6
<i>1.1.1. Problemática de las mascotas en Ecuador</i>	<i>6</i>
1.2. Sistemas de Identificación para Mascotas.....	7
<i>1.2.1. Placas o Etiquetas de Identificación.....</i>	<i>7</i>
<i>1.2.2. Microchips</i>	<i>8</i>
<i>1.2.3. Código QR</i>	<i>9</i>
<i>1.2.4. Collares con Rastreo GPS.....</i>	<i>9</i>
<i>1.2.5. Identificación por Radio Frecuencia (RFID).....</i>	<i>10</i>
1.3. Red de Sensores Inalámbricos (WSN).....	10
<i>1.3.1. Arquitectura de la Red WSN.....</i>	<i>11</i>
<i>1.3.2. Topología de la Red WSN.....</i>	<i>11</i>
<i>1.3.3. Análisis Comparativo de las Topologías de Red</i>	<i>14</i>
1.4. Tecnologías de Comunicación Inalámbrica de corto alcance.....	15
<i>1.4.1. Tecnología Bluetooth</i>	<i>16</i>
<i>1.4.2. Tecnología ZigBee.....</i>	<i>16</i>
<i>1.4.3. Tecnología Wi-Fi</i>	<i>16</i>
<i>1.4.4. Análisis Comparativo de Tecnologías de Comunicación Inalámbricas.....</i>	<i>17</i>
1.5. Tarjetas para Desarrollo de Hardware	18
<i>1.5.1. Arduino</i>	<i>18</i>
<i>1.5.2. Raspberry pi.....</i>	<i>19</i>
<i>1.5.3. NodeMCU.....</i>	<i>20</i>

1.5.4. Comparativa de las tarjetas de desarrollo	20
--	----

CAPITULO II

2. DISEÑO DEL PROTOTIPO INALÁMBRICO PARA GEO-LOCALIZACIÓN DE MASCOTAS.....	22
2.1. Requerimientos de <i>hardware</i> del prototipo de geo-localización.....	22
2.1.1. <i>Concepción general del prototipo</i>	22
2.1.2. <i>Arquitectura del Prototipo</i>	23
2.2. Selección de los dispositivos <i>hardware</i> para el prototipo	25
2.2.1. <i>Tarjetas de Desarrollo para el prototipo de geo-localización</i>	25
2.2.2. <i>NodeMCU ESP8266</i>	27
2.2.3. <i>Módulo Xbee S2C</i>	28
2.2.4. <i>Módulo GPS</i>	29
2.2.5. <i>Panel solar</i>	30
2.2.6. <i>Batería de Litio</i>	30
2.2.7. <i>Esquemas de conexión del prototipo para la geo-localización de mascotas</i>	31
2.3. <i>Software de desarrollo para el prototipo de geo-localización</i>	34
2.3.1. <i>Xctu</i>	34
2.3.2. <i>Arduino IDE</i>	34
2.3.3. <i>Ubidots</i>	34
2.3.4. <i>Programación de los módulos de comunicación</i>	34
2.3.5. <i>Requerimientos del Software del prototipo de geo-localización de mascotas</i>	37
2.3.6. <i>Diagramas de flujo del prototipo de geo-localización de mascotas</i>	37
2.4. <i>Visualización de los datos en la plataforma Ubidots</i>	40
2.5. <i>Diseño de ensamblaje para el prototipo de geo-localización</i>	42

CAPITULO III

3. PRUEBAS Y RESULTADOS.....	44
3.1. Selección de la topología de red para el prototipo.....	44
3.2. Pruebas para la comunicación de los nodos del prototipo.....	45
3.2.1. <i>Comunicación de los nodos</i>	45
3.2.2. <i>Alcance de comunicación de los nodos</i>	46

3.3.	Caracterización del nodo supervisión.....	47
3.3.1.	<i>Repetitividad del nodo de supervisión.....</i>	48
3.4.	Caracterización del nodo coordinador	49
3.4.1.	<i>Comunicación entre el nodo coordinador y la plataforma Ubidots.</i>	50
3.4.2.	<i>Prueba de almacenamiento.....</i>	51
3.5.	Funcionamiento del prototipo de geo-localización.	52
3.6.	Prueba de alimentación del prototipo de geo-localización	55
3.7.	Análisis económico del prototipo de geo-localización	56

CONCLUSIONES	57
---------------------------	-----------

RECOMENDACIONES	58
------------------------------	-----------

BIBLIOGRAFÍA

ANEXOS

ÍNDICE DE TABLAS

Tabla 1-1: Comparación de las topologías de red.....	14
Tabla 2-1: Características de transmisión para el estándar Wi-fi.....	17
Tabla 3-1: Cuadro comparativo entre las tecnologías de redes inalámbricas.....	17
Tabla 4-1: Comparación de las tarjetas de desarrollo Arduino, Raspberry, NodeMCU.....	21
Tabla 1-2: Especificaciones principales del Arduino UNO.....	26
Tabla 2-2: Especificaciones del Nodemcu ESP8266.....	27
Tabla 3-2: Especificaciones del módulo Xbee S2C.....	28
Tabla 4-2: Especificaciones GPS Ublox NEO-6M.....	29
Tabla 5-2: Especificaciones técnicas del panel solar.....	30
Tabla 1-3: Distancia para el funcionamiento del prototipo.....	47
Tabla 2-3: Valores obtenidos para la repetitividad del nodo de supervisión.....	48
Tabla 3-3: Cálculo para la alimentación con la batería.....	55
Tabla 4-3: Presupuesto económico del prototipo implementado.....	56

ÍNDICE DE FIGURAS

Figura 1-1: Placas de Identificación	8
Figura 2-1: Dispositivo Microchip	8
Figura 3-1: Código QR	9
Figura 4-1: Collar con rastreador GPS	10
Figura 5-1: Elementos de una red WSN	11
Figura 6-1: Topología Estrella.....	12
Figura 7-1: Topología Malla.....	13
Figura 8-1: Topología Árbol.....	13
Figura 9-1: Topología Bus.....	14
Figura 10-1: Tarjetas Arduino	19
Figura 11-1: Tarjetas Raspberry Pi.....	20
Figura 12-1: Tarjeta NodeMCU	20
Figura 1-2: Concepción general del prototipo.....	23
Figura 2-2: Diagrama de bloques del nodo supervisor	24
Figura 3-2: Diagrama de bloques del nodo propietario o coordinador	25
Figura 4-2: Arduino Uno	26
Figura 5-2: NodeMCU ESP8266.....	27
Figura 6-2: Xbee S2C y XBee explorer.....	28
Figura 7-2: GPS Ublox NEO-6M	29
Figura 8-2: Panel Solar	30
Figura 9-2: Pila Batería.....	31
Figura 10-2: Esquema de conexión del nodo de supervisión.....	31
Figura 11-2: Esquema de conexión del nodo coordinador	33
Figura 12-2: Configuración del nodo coordinador.	35
Figura 13-2: Configuración del nodo supervisor.....	36
Figura 14-2: Diagrama de flujo del nodo supervisor de datos.....	38
Figura 15-2: Diagrama de flujo del nodo coordinador.	40
Figura 16-2: Visualización de la variable creada en Ubidots.	41
Figura 17-2: Ubicación del nodo supervisor en Google Map.....	41
Figura 18-2: Diseño de la caja para el nodo coordinador.	42
Figura 19-2: Diseño de la caja para el nodo supervisor.....	43

Figura 1-3:	Topología Estrella.....	44
Figura 2-3:	Carga total y retardo entre la topología estrella y malla	45
Figura 3-3:	Comunicación entre los nodos	46
Figura 4-3:	Nodo de supervisión	47
Figura 5-3:	Datos de lectura de GPS	48
Figura 6-3:	Nodo coordinador	50
Figura 7-3:	Datos de comunicación mostrados en el IDE arduino	50
Figura 8-3:	Almacenamiento de datos en Ubidots.....	51
Figura 9-3:	Gráfico de la latitud y longitud almacenada en la plataforma	51
Figura 10-3:	Descarga de datos en un archivo Excel.....	52
Figura 11-3:	Ubicación de la mascota en Google Maps	52
Figura 12-3:	Prototipo acoplado en la mascota.....	53
Figura 13-3:	Delimitación del área para la ubicación de la mascota	54
Figura 14-3:	Mensaje de texto al propietario de la mascota	55

ÍNDICE DE ABREVIATURAS

PAE	Protección Animal Ecuador
APPA	American Pet Products Association
LOBA	Ley Orgánica de Bienestar Animal
RFID	Radio Frequency Identification
HDX	Half duplex o de media duplicidad
FDX	Full dúplex o de completa duplicidad
WSN	Wireless Sensor Network (Red de sensores inalámbricas)
WPAN	Wireless Personal Area Network, (Red Inalámbrica de Área Personal)
WLAN	Wireless Local Area Network (Red de Área Local Inalámbrica)
ISM	Industrial, Scientific & Medical
IEEE	Institute of Electrical and Electronics Engineers (Instituto de Ingeniería Eléctrica y Electrónica)
IoT	Internet of Things (Internet de las Cosas)
USB	Universal Serial Bus
HDMI	High Definition Multimedia Interface o HDMI (Interfaz Multimedia de Alta Definición)
GPS	Global Positioning System (Sistema de Posicionamiento Global)

RESUMEN

Se implementó un prototipo electrónico autosustentable con geo-localización mediante nodos inalámbricos y comunicación con el estándar Zigbee para la ubicación de mascotas en un área delimitada y una red tipo estrella para la comunicación entre los nodos. Consta de dos módulos: Nodo de Supervisión, que recolecta datos de latitud y longitud y Nodo Coordinador que se encarga del almacenamiento de los datos a la nube y permite la visualización en tiempo real de la ubicación. La información obtenida es almacenada en la plataforma llamada Ubidots, que permite saber la ubicación de la mascota y mediante sistema de mensajes cortos (SMS) alertar al dueño de la mascota se encuentra fuera del área delimitada. El prototipo es de fácil portabilidad, bajo costo y prestaciones adecuadas para el cual fue desarrollado. De las pruebas realizadas, el alcance de comunicación entre los dos nodos es de 100 metros con línea de vista, se establece que la comunicación es excelente de 1 a 50 metros, 51 a 70 alta, 71 a 90 media, 91 a 100 baja y de 101 a 150 no operacional. Se calculó el coeficiente de variación del prototipo donde tiene una variación de 0.2127% a 0.8587% en latitud y en longitud de 0.117% a 0.1085% donde el dispositivo es estable. El prototipo electrónico dispone de una alimentación autónoma de 11,29 horas continuas, sin la ayuda del panel solar luego de eso se cargará automáticamente cuando la batería lo requiera. Se concluye que el prototipo electrónico implementado cumple con los requerimientos planteados, y que es de ayuda para las personas que tienen una mascota. Se recomienda implementar y realizar estudios para reducir el tamaño del prototipo y aumentar sensores que permita medir otras variables.

Palabras claves: <GEO-LOCALIZACIÓN>, <UBICACIÓN DE MASCOTAS>, <PROTOTIPO ELECTRÓNICO>, <COMUNICACIÓN INALÁMBRICA>, <RED DE SENSORES INALÁMBRICOS (WSN)>, <SISTEMA DE ALERTA>.

ABSTRACT

A self-sustaining electronic prototype with geo-localization was implemented through wireless nodes and communication with the Zigbee standard for the location of pets in a delimited area and a star type network for communication between the nodes. It consists of two modules: Supervision Node, which collects latitude and longitude data and Coordinating Node that is responsible for storing the data to the cloud and allows real-time visualization of the location. The information obtained is stored on the platform called Ubidots, which allows to know the location of the pet and by means of a short message system (SMS) to alert the owner of the pet outside the defined area. The prototype is easy portability, low cost and adequate performance for which it was developed. Of the tests carried out, the communication reach between the two nodes is 100 meters with line of sight, it is established that the communication is excellent from 1 to 50 meters, 51 to 70 high, 71 to 90 medium, 91 to 100 low and from 101 to 150 non-operational. The coefficient of variation of the prototype was calculated where it has a variation of 0.2127% to 0.8587% in latitude and in length of 0.117% to 0.1085% where the device is stable. The electronic prototype has an autonomous power of 11.29 continuous hours, without the help of the solar panel after that it will be charged automatically when the battery requires it. It is concluded that the electronic prototype implemented meets the requirements, and that it is helpful for people who have a pet. It is recommended to implement and carry out studies to reduce the size of the prototype and increase sensors to measure other variables.

Keywords: <GEO-LOCALIZATION>, <LOCATION OF PETS>, <ELECTRONIC PROTOTYPE>, <WIRELESS COMMUNICATION>, <WIRELESS SENSORS NETWORK (WSN)>, <ALERT SYSTEM>.

INTRODUCCIÓN

ANTECEDENTES

Sin duda las mascotas ocupan un lugar muy importante en la vida de muchas familias, durante los años 2013 y 2014, en colaboración con el grupo activista Red Mascota Multimedia, se realizó un estudio estadístico, donde se presenta un panorama sobre los factores más comunes en el extravío de mascotas. Ningún estudio estadístico formal maneja una muestra de casos, por lo que no lo presentamos con una formulación detallada que lo avale. El problema es que en materia de mascotas, en México (y en general en América Latina), casi todo es especulativo. No hay datos, no hay números que se detalle sobre la estadística de canes extraviados o perdidos. De un número total de 273 casos, seguidos a través de un año, 115 eran mestizos (42%), mientras 158 parecían o eran de raza (58%). Un total de 110 tenían placa de identificación, 113 sólo collar, sin placa y 50 no traían ninguno. De los 140 que sí regresaron a casa, 90 eran mestizos (64%) mientras 50 eran de raza (36%). Los casos antes mencionados pueden responder a los escrúpulos de algunas personas y sus ganas de “quedárselo”, cuando lo encontraron o deshacerse en el peor de los casos (Colmenares, 2013).

En el Ecuador, no existe un registro de cuantas mascotas se encuentran desaparecidas, las estadísticas indican que 8 de cada 10 mascotas tienen hogar. En Quito se estima que existen 400 mil perros, pero de estos hay cerca de 280 mil (70%), que viven en la calle, explica Guillermo Altamirano, director de inspección de la Agencia Metropolitana de Control, quien sostiene que el 90% de estos animales tenían dueños, pero fueron abandonados (La Hora, 2013).

Carlos Realpe, coordinador de bienestar animal de Protección Animal Ecuador (PAE), dice que es importante ponerle la placa a la par del microchip y esterilizar para evitar pérdidas. Entre los métodos de localización se encuentran: Placas, microchips, códigos QR, e incluso registros de ADN que son utilizados hoy en día para registrar a las mascotas. Los dueños de animales de compañía adquieren un compromiso al adoptar o comprar un amigo canino o felino cuya responsabilidad se traduce en la identificación.

En el Ecuador, el registro e identificación de mascotas se hace por iniciativa de los dueños y fundaciones. En Protección Animal Ecuador se colocan en promedio 10 chips mensuales sin embargo, no todos los centros veterinarios ni hospitales cuentan con lectores (Castellanos Gabriela).

Una publicación en Solla Blog menciona que el seguimiento de mascotas tuvo un espacio en la Feria Internacional de Electrónicos Consumo CES 2015, en las Vegas (Estado Unidos) dieron a conocer el dispositivo Tagg que permite registrar y analizar la actividad física de la mascota, hacer el seguimiento vía GPS y enviar alertas de su ubicación y medir la temperatura ambiental a la que está expuesto y alertar ante posibles riesgos (Tagg y Wonderwoof, 2015, p. 7, p. 7).

FORMULACIÓN DEL PROBLEMA

¿Cómo implementar un prototipo electrónico autosustentable geo-localizado con nodos inalámbricos para la ubicación de mascotas mediante una red de sensores?

SISTEMATIZACIÓN DEL PROBLEMA

- ¿Cuáles son los métodos de localización de mascotas que se puede aplicar al momento de extravío?
- ¿Cuáles son los requerimientos que debe cumplir el prototipo utilizando tecnología inalámbrica de corto alcance?
- ¿Cuál es la topología y estructura de la red que cumple con los requerimientos definidos en la investigación?
- ¿Cuál es el diseño de los nodos y de los componentes hardware y software que se incorporaran al prototipo de red para que se adapten a los requerimientos de la investigación?
- ¿El prototipo de red implementado cumple con los requisitos establecidos establecido en localización indoor?

JUSTIFICACIÓN TEÓRICA

La semana nacional de identificación de mascotas se celebra el 16 al 22 de abril de cada año, sirve como recordatorio para asegurarnos de hacer todo lo posible para que nuestras mascotas puedan regresar fácilmente con sus dueños en caso de que se pierdan. Una de cada tres mascotas se pierde en algún punto de su vida, además de las etiquetas de identificación estándar para mascotas, otros métodos para identificar a las mascotas incluyen tatuaje permanente, microchip, etiquetas especiales como las etiquetas de identificación digitales y los dispositivos de rastreo (Becker, 2017).

Algunos motivos más comunes para que una mascota se alejara o se pierden se tiene algunos de los datos relevantes: 49 salieron corriendo tras un gato, rata, ardilla u otros perros dando 21%, 54 se asustaron con un claxon, escape, gritos o fuegos artificiales un 20%, 41 sencillamente empezaron a acelerar “sin razón” con un 18%, 46 se asustaron por otra persona que los provocó o los quiso alejar con 17% y un 44 sencillamente se perdieron de la vista dando 19% (Colmenares, 2013).

PetHub, es una compañía estadounidense que fabrica etiquetas de identificación, y a su vez ofrece algunos datos estadísticos sobre las mascotas perdidas: 1 de cada tres mascotas se pierden en algún momento de su vida y al menos del 2% de los gatos perdidos y sólo del 15 al 20% de los perros perdidos logran volver con sus familias según (Consejo Nacional de Estudio y Política de la Población de Mascotas). La mayoría de las mascotas que regresan a sus hogares tienen etiquetas de identificación, microchip o tatuaje y son el 80% de los dueños de mascotas creen que las etiquetas de identificación son crucialmente importantes, sin embargo, sólo el 33% informa que su mascota siempre porta una identificación (PetHub, s.f.).

Entre los métodos de identificación están las placas QR, al ser escaneados, estos códigos llevan a la persona a una página donde encuentra los datos del dueño. Loved Mascot, por ejemplo, es un servicio que permite identificar al animal de compañía con un código único. Los propietarios ingresan sus datos en una página web y, si un día se pierde, la persona que lo encuentre puede leer el QR con su teléfono y acceder a la información. Hay otros sistemas en Málaga, España se instauró un nuevo registro de mascotas. Se trata de una base de datos con identificación por ADN (Castellanos Gabriela).

En la actualidad no se ha desarrollado un sistema con las mismas características por tal motivo la investigación se centra en la implementación de un prototipo electrónico autosustentable geo-

localizado con nodos inalámbricos para la ubicación de mascotas mediante una red de sensores el cual considera una red inteligente que estará acorde a las líneas de investigación establecida en la Escuela Superior Politécnica de Chimborazo.

JUSTIFICACIÓN APLICATIVA

Debido al gran número de mascotas que desaparecen por diferentes motivos, con la información encontrada de que el porcentaje de mascotas desaparecidas aumenta. Surge la idea de crear un prototipo electrónico autosustentable geo-localizador con nodos inalámbricos para la ubicación de mascotas mediante una red de sensores, con esto se pretende tener la ubicación correcta de la mascota.

El desarrollo del prototipo a implementar está compuesto por un nodo de supervisión y un nodo coordinador. El nodo supervisor será el encargado de rastrear la ubicación mediante el sensor GPS, este estará ubicado en un chaleco para comodidad de la mascota, la información proporcionada por el sensor será enviada al nodo coordinador mediante comunicación inalámbrica de corto alcance a la plataforma de almacenamiento, el propietario tendrá la información en tiempo real para ubicar a la mascota. Además se delimita un área del domicilio del propietario, en caso de que su mascota se encuentre fuera de este entorno se le notificara con un mensaje de texto a su dispositivo móvil.

OBJETIVOS

OBJETIVO GENERAL

Implementar un prototipo electrónico autosustentable geo-localizado con nodos inalámbricos para la ubicación de mascotas mediante una red de sensores.

OBJETIVOS ESPECÍFICOS

- Analizar los métodos de localización de mascotas que se puede aplicar al momento de extravío.
- Establecer los requerimientos que debe cumplir el prototipo utilizando tecnología inalámbrica de corto alcance.
- Elegir la topología y estructura de red que cumple con los requerimientos definidos en la investigación.
- Seleccionar el diseño de los nodos y los componentes hardware y software que se incorporaran al prototipo de red para que se adapten a las necesidades de la investigación.
- Determinar si el prototipo de red implementado cumple con los requisitos establecidos en localización indoor.

CAPÍTULO I

1. MARCO TEÓRICO

En el siguiente capítulo se detalla información de las tecnologías y topologías existentes para llevar a cabo la implementación del dispositivo para la geo-localización de las mascotas, de acuerdo a los objetivos planteados. Además se indica las características de los elementos a utilizar, los mismos que son analizados para una mejor selección.

1.1. Mascota

Una mascota o animal de compañía se reduce en las designaciones como animal o cosa que sirve como talismán (Real Academia Española, 2017). La finalidad de una mascota o también llamados animales domésticos es llevar felicidad y buena suerte a los hogares que los acogen y así formar parte de la familia (Animalear, 2018).

En la mayor parte de los países occidentales, el número de hogares que acogen perros o gatos como mascotas ha crecido en las últimas décadas. En Estados Unidos en el año 2011 más de un tercio de los hogares tenían uno o más perros, y poco menos de un tercio tenía uno o más gatos. Cálculos más recientes realizados por la American Pet Products Association (APPA) estimaron que entre el 2015 y el 2016 más del 44% de los hogares estadounidenses contarían con al menos un perro de compañía. En la Unión Europea en el 2014 se encontró que poco más del 25% de los hogares tenía al menos un gato, y alrededor del 18% tenía al menos un perro en sus hogares (Díaz, 2017, p. 69, p. 56).

1.1.1. Problemática de las mascotas en Ecuador

Hoy en día existen muchas mascotas que se pierden lo que conlleva a que su propietario se le haga difícil su localización o que la mascota regrese a casa por sí sola, ocasionando que existan muchos perros callejeros sin un hogar; en la actualidad existen varias fundaciones de protección animal posicionadas en todo Ecuador que ayudan o dan un hogar a perros abandonados.

El 30 de octubre de 2014 (Yepez Desiree, 2015), un grupo de fundaciones como Protección Animal Ecuador (PAE), la Asociación Animalista Libera, Rescate Animal Ecuador, y varios representantes de la sociedad civil presentó el proyecto de Ley Orgánica de Bienestar Animal (LOBA) ante la Asamblea Nacional. LOBA abarca 5 áreas: violencia, salud pública, derechos de la naturaleza, bienestar animal y buen vivir. Quienes promueven LOBA dicen que no se trata de una propuesta punitiva sino de una ley necesaria para lograr el bienestar animal (El Telégrafo, 2014).

El Reglamento Nacional de tenencia de perros elaboró el Acuerdo Interministerial para la Tenencia Responsable de perros, en febrero del 2009 donde establece el reglamento los siguientes puntos. Las obligaciones citan que se debe otorgar condiciones de vida adecuadas a las características del animal, educar, socializar e interactuar con el perro en la comunidad, mantener únicamente el número de perros que las normas de bienestar animal permiten, entre otras. La prohibición apunta a no maltratar, golpear, abandonar, mantener en estado de aislamiento, encadenar, enjaular o por último envenenar masivamente perros propios o ajenos; al no cumplirse con este reglamento se acoge la respectiva sanción dependiendo del caso (Protección Animal Ecuador).

1.2. Sistemas de Identificación para Mascotas

Hoy en día existen muchas publicaciones en redes sociales y anuncios volantes requiriendo ayuda para encontrar mascotas perdidas, muchas de las veces estos canes extraviados no regresan con su dueño. Para prevenir este tipo de incidentes existen muchos métodos de identificación que va desde lo más sencillo como un collar de identificación a diversos sistemas de identificación que existe en el mercado. La identificación de mascotas es necesaria para tenerlas registradas de forma individual en cada uno de los hogares.

1.2.1. Placas o Etiquetas de Identificación

Unas de las prácticas más sencillas de tener identificado a las mascotas son las placas de identificación por lo general son de metal o plástico. Ubicada en el collar de la mascota donde se encuentra grabado los datos como: el nombre de la mascota, nombre de su dueño y número de teléfono de contacto. Una ventaja de esta identificación es su costo que no es muy elevado, la desventaja principal es que con el tiempo el grabado se va estropeando lo que hace imposible la lectura de los datos de la mascota y

asegurarse que el can tenga siempre su collar (Protección Animal Mundial, 2017). En la figura 1-1 se muestran placas de identificación de las mascotas.



Figura 1-1: Placas de Identificación

Realizado por: AVILA, Doris, 2019.

1.2.2. Microchips

Son dispositivos cilíndricos que miden 11,5 mm de largo por 2,1 mm de diámetro (Trovan), del tamaño de un grano de arroz, no tienen batería y están sellados en vidrio o polímero biocompatible que se implantan en los tejidos subcutáneos de las mascotas. Contienen cuatro elementos: condensador, antena, cable de conexión y una cubierta, como se muestra en la figura 2-1.

Se activa mediante una señal de radiofrecuencia de baja potencia emitida por los escáneres que se pasa sobre la zona, y las ondas de radio emitidas activan el microchip una vez activado el microchip transmite un número de identificación en la pantalla del escáner (AVMA(American Veterinary Medical Association), 2019).



Figura 2-1: Dispositivo Microchip

Realizado por: AVILA, Doris, 2019.

1.2.3. Código QR

Inventado en 1994 por Denso, una de las compañías más importantes del grupo Toyota y aprobado en junio de 2000 como una norma internacional ISO (ISO / IEC18004) (Maione et al., 2017, p. 2).

Conocido generalmente como código de barras bidimensional de forma cuadrada donde se almacena datos codificados, funciona a través de un collar que contiene un código QR en el cual se almacenan datos como: el nombre de la mascota, raza, número de teléfono de su propietario entre otros datos. Por medio de la cámara de los teléfonos celulares se da lectura de códigos QR y se accede a las direcciones URL de Internet automáticamente. En la figura 3-1 se muestra un código QR.



Figura 3-1: Código QR

Realizado por: AVILA, Doris, 2019.

1.2.4. Collares con Rastreo GPS

Este tipo de collares llevan incorporado un pequeño aparato que porta un receptor de señales GPS, lo cual permite saber en qué coordenadas se encuentra la mascota. De esta forma, el collar puede recoger las coordenadas geográficas de su posición y enviar vía telefónica o mediante un mensaje de texto al móvil o aplicación del propietario de la mascota.

Entre los inconvenientes que presentan estos dispositivos son: la utilización de una batería a la que debe constantemente cargarse, lo que conlleva sacarle su collar cada vez. También necesita estar suscripto a una empresa que brinda el servicio de rastreo satelital y el pago de su uso, entre otras. (Petfy, 2017). En la figura 4-1 se muestra el collar de rastreo GPS.



Figura 4-1: Collar con rastreador GPS

Fuente: <https://n9.cl/5Qw>

1.2.5. Identificación por Radio Frecuencia (RFID)

Las etiquetas con tecnología RFID (Radio Frequency Identification) son microchips utilizadas en mascotas en vez de tener implantado un microchip, RFID ayuda a localizar a mascotas dentro de un rango determinado. El rango de frecuencias de activación emitidas por el lector es de 120 y 134,2 kHz y el intercambio de información entre transponder a lector se escogerá dependiendo de dos sistemas de identificación de radiofrecuencia como es el Half duplex (HDX) o de media duplicidad y el Full duplex (FDX) o de completa duplicidad (Chip247).

1.3. Red de Sensores Inalámbricos (WSN)

La Red de sensores inalámbricos en los últimos tiempos ha tenido un rápido desarrollo, consiste en nodos sensores que se conectan entre sí para un bien en común, son de bajo costo y consumo de energía. El sistema WSN (Wireless Sensor Network) ha creado otras aplicaciones tales como el control y monitorización de entornos, automatización residencial y de edificios, el control industrial, la monitorización ambiental entre ellas la presión, humedad, sonido, entre otros. (Macías Villalba Álvaro, 2015, p. 3).

1.3.1. Arquitectura de la Red WSN

Esta basada en un conjunto de dispositivos llamados nodo sensor, encargado de recopilar información y enviar los datos a una estación base a través del Gateway, como se muestra en la figura 5-1 los elementos que intervienen en una red de sensores inalámbricos.

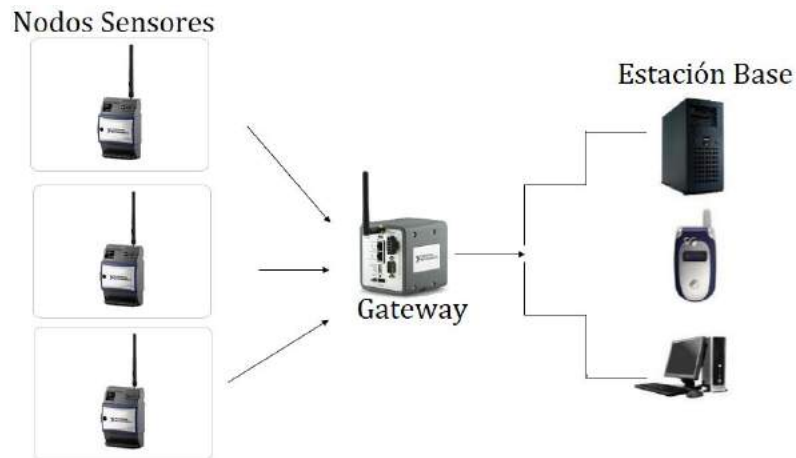


Figura 5-1: Elementos de una red WSN

Realizado por: AVILA, Doris, 2019.

Nodos Sensores: También llamados mota por ser pequeños y livianos, estos sensores recolectan distintos datos como la humedad, temperatura, ruido, entre otros. Estos datos realizan la comunicación entre el nodo sensor y la estación base pasando por medio del gateway.

Gateway: Intermediario para la interconexión entre los nodos sensores y una red TCP/IP.

Estación Base: Recolecta la información en una base de datos donde el usuario puede analizar y observar los valores obtenidos por los sensores. (Ortiz Tapia, p. 3).

1.3.2. Topología de la Red WSN

La conexión de la red tanto física como lógica de los dispositivos y nodos, interactúan entre si transmitiendo información. Entre las topologías existentes y las que utiliza una red WSN son las siguientes: estrella, mallas, bus y árbol las que pueden ser usadas de acuerdo a los requerimientos de los usuarios.

- **Topología en Estrella**

Como se observa en la figura 6-1, el nodo central de la red está en la mitad de los dispositivos, la información que envía el sistema puede ser de un solo sentido a los demás nodos sensores y a una distancia más o menos de 30 a 100 metros. Mediante la puerta de enlace se transmite la información en los nodos finales y no pueden compartir información entre ellos. Esta topología presenta una ventaja como es el bajo consumo de energía, y desventajas entre ellas su limitación de transmisión entre cada nodo y puerta de enlace. No presenta una vía de comunicación alternativa en caso de obstrucción de uno de los nodos, la información se perderá. (Macías Villalba Álvaro, 2015, p. 6).

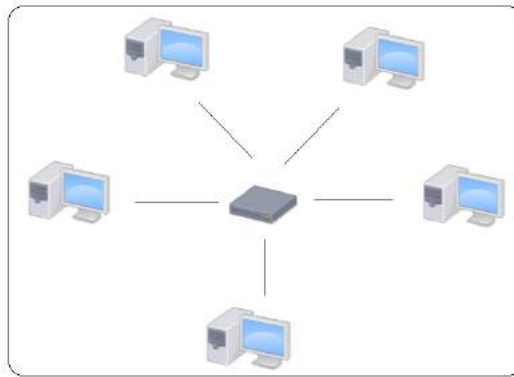


Figura 6-1: Topología Estrella

Realizado por: AVILA, Doris, 2019.

- **Topología en Malla**

Conocida como Mesh o Malla consiste en un sistema multisalto donde cada nodo puede enviar y recibir información de cualquier otro nodo, esta topología es una de las más utilizadas por ser muy fiable y flexible por donde el mensaje viaja en distintos caminos. Se observa en la figura 7-1 la conexión de la topología malla, que representa la forma de operar un sistema. Si un nodo falla la información no será afectada debido a la gran variedad de rutas que posee la red y es reestablecida automáticamente. (Macías Villalba Álvaro, 2015, p. 6).

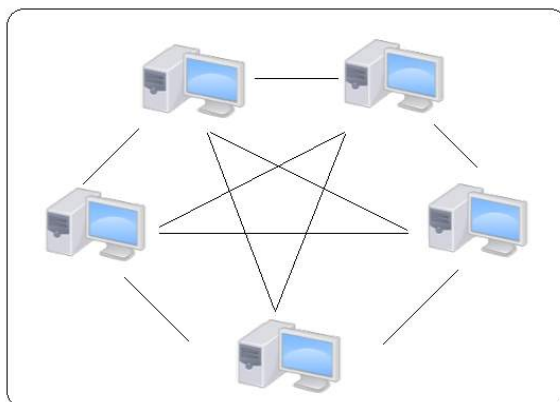


Figura 7-1: Topología Malla.

Realizado por: AVILA, Doris, 2019.

- **Topología en Árbol**

La comunicación es de forma jerárquica es decir la información se enviará de un nodo menos jerárquico a otro con mayor jerarquía cuyos datos serán enviados hasta el nodo central. Esta topología es más conocida como topología árbol. La principal característica de esta topología es que presenta una buena expansión de la red es decir mayor flexibilidad y cuyo inconveniente es el fallo en uno de sus nodos lo que implica que una rama de la topología se interrumpa y la comunicación no llegue a su destino. (Rodas Vásquez, 2013, p. 7). A continuación en la figura 8-1 la topología árbol.

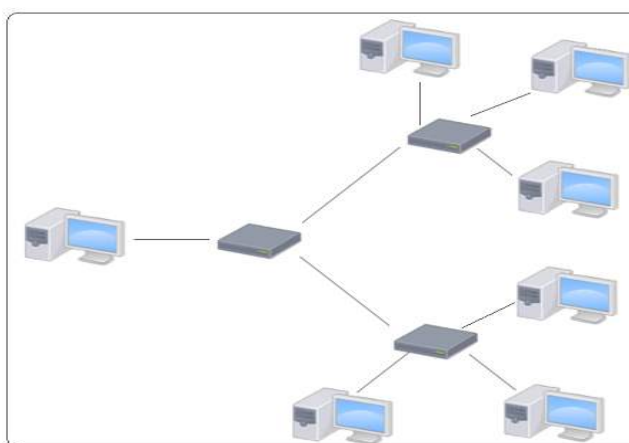


Figura 8-1: Topología Árbol.

Realizado por: AVILA, Doris, 2019.

- **Topología Bus**

La comunicación de esta topología es mediante la conexión de un único canal por tal motivo es llamado topología en bus, tiene facilidad en su implementación e inconvenientes en su red de enlace si este falla todos los datos se perderán y los nodos quedan incomunicados. En la figura 9-1 se muestra la topología bus. (Rodas Vásquez, 2013, p. 7).

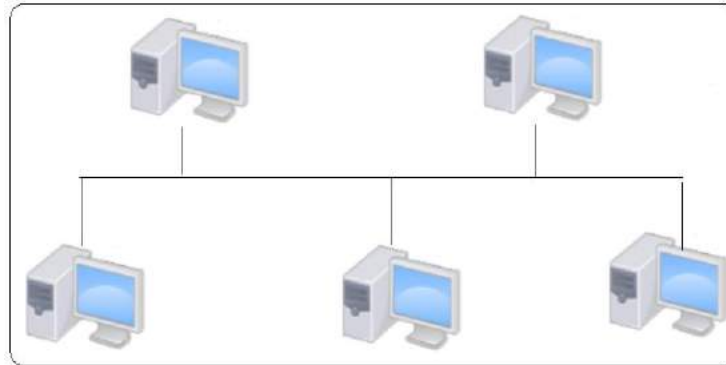


Figura 9-1: Topología Bus.

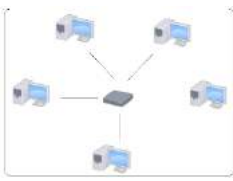
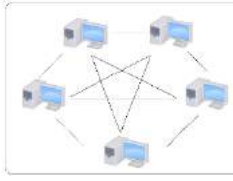
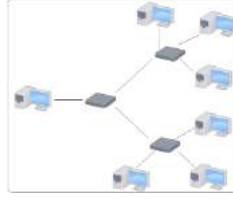

Realizado por: AVILA, Doris, 2019.

1.3.3. Análisis Comparativo de las Topologías de Red

En la tabla 1-1 se puntualiza varios ítems como el tráfico, cable, costo facilidad de poder añadir otros equipos y sus desventajas de las topologías de red que se ha detallado anteriormente.

Tabla 1-1: Comparación de las topologías de red.

TOPOLOGÍAS DE RED					
TOPOLOGÍA	TRÁFICO	CABLE	COSTO	FACILIDAD DE AÑADIR EQUIPOS.	DESVENTAJAS
	Es muy sencillo y fácil de controlar	Par Trenzado	Alto por la cantidad de cables.	Depende de los puertos.	Limitación de transmisión entre cada nodo y puerta de enlace

<p style="text-align: center;">ESTRELLA</p> 					
<p style="text-align: center;">MALLA</p> 	<p>Su velocidad puede variar.</p>	<p>Fibra óptica la más recomendable</p>	<p>Costo es elevado</p>	<p>Difícil cuando existen demasiados cables y ordenadores</p>	<p>Su elevado costo</p>
<p style="text-align: center;">ÁRBOL</p> 	<p>Mantiene poco tráfico al aumentar dicho tráfico la rapidez decae</p>	<p>Punto a punto</p>	<p>No es muy costoso</p>	<p>La expansión de la red permite tener mayor flexibilidad</p>	<p>Si uno de los nodos sufre un error queda aislado de los demás</p>
<p style="text-align: center;">BUS</p> 	<p>Fácil control es decir su transmisión es lenta</p>	<p>Es necesario cableado para su instalación</p>	<p>Más barato a comparación de los demás.</p>	<p>Implementación fácil.</p>	<p>Si el enlace falla todos los datos se perderán.</p>

Fuente: (Norber Barraza, 2013)

Realizado por: Avila, Doris; 2019

1.4. Tecnologías de Comunicación Inalámbrica de corto alcance

Para que exista la comunicación inalámbrica o también llamada comunicación sin cables, utiliza la modulación de ondas electromagnéticas que viajan a una gran velocidad por el espacio a distintas direcciones. Todo esto es posible y permite la transmisión de datos con una antena y energía proporcionada adicional (Regino Mejía Aura María, 2014).

Con el avance de la tecnología también ha sido necesario el desarrollo de las tecnologías inalámbricas para una mejor transmisión de datos; como lo es la tecnología ZigBee, Bluetooth o la más popular la tecnología Wi-Fi a continuación se detalla cada una.

1.4.1. Tecnología Bluetooth

Ericsson principal fundador de la tecnología Bluetooth creada en el año se 1994 con el apoyo de Intel, IBM, entre otras grandes empresas tecnológicas dan a conocer al mundo lo que hoy es Bluetooth Special Interest Group (Castellano Alvaro Richi, 2012, p. 3).

Bluetooth por formar parte de la Redes de Área Personal Inalámbricas (WPAN) trabaja bajo el estándar de IEEE asignándole la denominación 802.15.1, el cual permite la implementación de redes inalámbricas de corto alcance, este sistema opera a una banda ISM (Industrial, Scientific & Medical) de 2.4 GHz, máxima velocidad de transmisión es de 3 Mbps. Los rangos de frecuencia varían dependiendo de su país por ejemplo en Europa y Norte América trabajan a 2.400 hasta 2483.5 MHz (Corredor, Pedraza y Hernández, 2009, p. 73-84, p. 2).

1.4.2. Tecnología ZigBee

ZigBee Alliance consorcio internacional formado por más de 100 compañías populares como lo es Motorola, Samsung y Philips entre otras promueven y dan a conocer a la tecnología inalámbrica más conocida a nivel mundial como ZigBee, regulada bajo el estándar IEEE 802.15.4 hacia redes inalámbricas de área personal (WPAN) (Prieto Blázquez, p. 14).

Se caracteriza por brindar datos a velocidades alcanzadas entre 20Kbps y 250Kbps y cuenta con rangos de cobertura a 10 metros hasta 100 metros. En Europa dispone de bandas libres ISM (Industrial, Scientific & Medical) de 2,4 GHz a 868 MHz y en Norte América bandas de 915 MHz. ZigBee permite utilizar el protocolo asíncrono Half Duplex, esta tecnología puede incorporar la encriptación y su autenticación como medidas de seguridad para la red (Salgado Ignacio Vidri, 2012, p. 2).

1.4.3. Tecnología Wi-Fi

Esta tecnología surge con la idea y la necesidad de comunicación mediante la conexión inalámbrica y que pueda ser compatible con diverso dispositivos móviles. Wireless Fidelity una marca comercial

que hace referencia a la abreviación de Wi-Fi o conocida como WLAN soportan el estándar IEEE 802.11., estas proporciona un rango de acceso inalámbrico hasta 100 metros. Actualmente consta de algunos estándares el mismo que son detallados en la tabla 2-1:

Tabla 2-1: Características de transmisión para el estándar Wi-fi.

IEEE	Banda	Tasa de Transferencia
802.11.a	5GHz	54 Mbps
802.11.b	2,4 GHz	11 Mbps
802.11.g	2,4 GHz	54 Mbps
802.11n	2,4 GHz y 5GHz	600 Mbps
802.11.ac	6 GHz	1 Gbps – 500 Mbps




Fuente: (Salazar, p. 13)

Realizado por: AVILA, Doris, 2019

1.4.4. Análisis Comparativo de Tecnologías de Comunicación Inalámbricas

En comparación entre las tecnologías como muestra la tabla 3-1 se observa que Bluetooth ZigBee y Wi-Fi tienen el mismo rango de frecuencia de 2,4 GHz. De acuerdo al análisis del cuadro comparativo se selecciona la tecnología ZigBee por contener más nodos a comparación de los demás nos permite menos consumo de potencia en la transmisión de datos y mayor duración de la batería.

Tabla 3-1: Cuadro comparativo entre las tecnologías de redes inalámbricas.

	Bluetooth 	ZigBee 	Wi-Fi 
Modulación	FHSSS	DSSS	DSSS
Rango	10-100 metros	10-100 metros	50-100 metros
Banda Ancha	54 Mbps	250 kbps	1 Mbps
Topología de Red	Ad-Hoc, redes muy pequeñas.	Ad-Hoc punto a punto, estrella o malla.	Punto a Hub
Frecuencia de Operación	2.4 GHz	868 MHz en Europa 900-928 MHz Norte América	2.4 GHz y 5GHz

		2,4 GHz resto del mundo	
Complejidad	Alto	Bajo	Alto
Consumo de Potencia	Transmisión: 40mA Reposo: 0.2 mA	Transmisión: 30mA Reposo: 3 uA	Transmisión: 400mA Reposo: 20 mA
Duración de Baterías	1 semana	100 -1000 días	12 - 48 horas
Seguridad	Cifrado de 64 y 128 bits	128 AES, mayor seguridad en la capa de aplicación	
Numero de nodos por maestro	7	64000	32
Software y Hardware	Muy complejo	Sencillo	Complejo
Aplicaciones	Solución de cable, móviles, WPAN	Control y Monitorización	Web, e-mail y datos, WLAN
Ventajas	Interoperabilidad	Fiabilidad, Bajo consumo y Bajo coste	Velocidad y Flexibilidad

Fuente: (Ane Mariñelarena, 2013, p. 1),(Rodas Vázquez, 2013, p. 13)

Realizado por: AVILA, Doris, 2019

1.5. Tarjetas para Desarrollo de Hardware

También llamadas placas de desarrollo que hoy en día presentan diversas modificaciones en sus versiones y diseños, las que son utilizadas por múltiples sistemas estos cuentan con sus propios lenguajes de programación y sistema operativo, una de sus ventajas principales es que son compatibles con otros dispositivos.

1.5.1. *Arduino*

Una de las placas de desarrollo muy utilizadas por su bajo costo, código abierto y por las posibilidades de aplicación cuenta con la plataforma open-source con entradas y salidas, analógicas y digitales. ATMEL es el fabricante del microcontrolador AVR de la plataforma Arduino que posee una programación similar a C y C++, siendo su elemento principal de la placa. Existen múltiples modelos de Arduino con diferentes características como su nombre, formas, características y las distintas funciones. En la figura 10-1 se indica varios modelos de la tarjeta arduino (Machicao Marcelo, 2017, p. 1).


	Arduino Uno	Arduino Mega2560	Arduino Leonardo	Arduino Due	Arduino ADK	Arduino Nano	Arduino Pro Mini	Arduino Espora
								
Microcontrolador	ATmega328	ATmega2560	ATmega32u4	AT91SAM3X8E	ATmega2560	ATmega168 (versão 2.x) ou ATmega328 (versão 3.x)	ATmega168	ATmega32u4
Portas digitais	14	54	20	54	54	14	14	-
Portas PWM	6	15	7	12	15	6	6	-
Portas analógicas	6	16	12	12	16	8	8	-
Memória	32 K (0,5 K usado pelo bootloader)	256 K (8 K usados pelo bootloader)	32 K (4 K usados pelo bootloader)	512 K disponível para aplicações	256 K (8 K usados pelo bootloader)	16 K (ATmega168) ou 32K (ATmega328), 2 K usados pelo bootloader	16 K (2k usados pelo bootloader)	32 K (4 K usados pelo bootloader)
Clock	16 Mhz	16 Mhz	16 Mhz	84 Mhz	16 Mhz	16 Mhz	8 Mhz (modelo 3.3v) ou 16 Mhz (modelo 5v)	16 Mhz
Conexão	USB	USB	Micro USB	Micro USB	USB	USB Mini-B	Serial / Módulo USB externo	Micro USB
Conector para alimentação externa	Sim	Sim	Sim	Sim	Sim	Não	Não	Não
Tensão de operação	5v	5v	5v	3.3v	5v	5v	3.3v ou 5v, dependendo do modelo	5v
Corrente máxima portas E/S	40 mA	40 mA	40 mA	130 mA	40 mA	40 mA	40 mA	-
Alimentação	7 - 12 Vdc	7 - 12 Vdc	7 - 12 Vdc	7 - 12 Vdc	7 - 12 Vdc	7 - 12 Vdc	3.35 - 12 V (modelo 3.3v), ou 5 - 12 V (modelo 5v)	5v

Figura 10-1: Tarjetas Arduino

Fuente: <http://mrobotics.es/blog/wp-content/uploads/2016/04/Tabela-comparativa-630x449.png>

1.5.2. Raspberry pi

Es una placa reducida o SBC del tamaño de una tarjeta de crédito de bajo costo, fue diseñada por la fundación Raspberry en Reino Unido. Posee programas y lenguajes de programación como el Python y C++ que son compatibles con el Raspberry pi, entre los sistemas operativos puede utilizar el Windows 10 y GNU/Linux. Con un procesador Broadcom, memoria RAM, GPU, USB, HDMI, Ethernet y un conector para cámara (César y Sánchez, 2018, p. 23-24). En la Figura 11-1 se observa los modelos de Raspberry Pi.

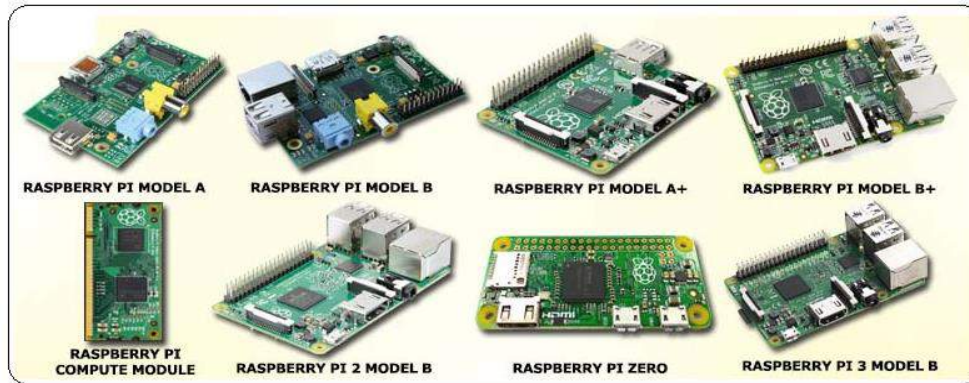


Figura 11-1: Tarjetas Raspberry Pi

Fuente: <https://fernandolopez6.wordpress.com/2017/01/23/modelos-raspberry-pi/>

1.5.3. NodeMCU

Desarrollado bajo el código abierto, el chip ESP8266 de la placa NodeMCU utiliza el LUA como lenguaje de programación para aplicaciones que requieran conexión a WiFi en un mínimo tiempo. Esta tarjeta de desarrollo tiene un procesamiento y almacenamiento que puede adaptarse sensores y aplicaciones a través de GPIOs con una baja duración en su ejecución. NodeMCU es compatible con Arduino y realizar aplicaciones en IoT (internet de las cosas) (ElectroniLab, 2018).

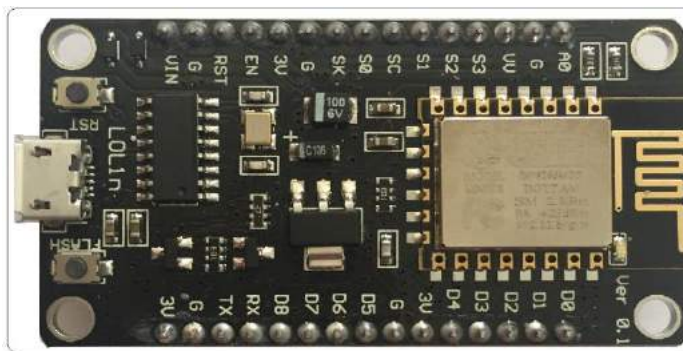


Figura 12-1: Tarjeta NodeMCU

Realizado por: AVILA, Doris, 2019

1.5.4. Comparativa de las tarjetas de desarrollo

En la Tabla 4-1 se muestran las características comparativas de las tarjetas de desarrollo que se ponen en análisis para el prototipo. Por su bajo costo y por poseer terminales analógicos se escoge el Arduino y el NodeMCU ya que una red de sensores inalámbricos no requiere de altas velocidades.

Tabla 4-1: Comparación de las tarjetas de desarrollo Arduino, Raspberry, NodeMCU.

Características	Arduino uno	Raspberry Pi B	NodeMCU
Procesador	ATMega 328	Arm11	ESP8266
Velocidad reloj	16 MHz	700 MHz	
Voltaje	5V	5V	5V
Voltaje Vin	7-12V		20 V
Voltaje (limite)	6-20V		
RAM	2 Kb	512 Mb	32Kb
USB	No dispone	2	
Audio	No dispone	HDMI, Analógico	
Video	No dispone	HDMI, Analógico	
Ethernet	No dispone	10/100	
Pines Entrada/Salida Digitales	14	28 GPIO	9 GPIO
Pines E analógicos	6		1
Corriente pin E/S	40mA	50mA	
Alto	68.6 mm	5.7 cm	
Ancho	53.4 mm	8.6 cmm	
Peso	25g	45 g	
Sistema operativo	No dispone	Linux	
Entorno	Arduino IDE	Linux, IDLE, Open Embedded, QEMU, Scratch box, Eclipse	
Costo	22 USD	25 USD	15 USD

Fuente: (Luis Posadas-Yagüe Jose Luis Poza Luján, 2014, p. 15)

Realizado por: AVILA, Doris, 2019

CAPITULO II

2. DISEÑO DEL PROTOTIPO INALÁMBRICO PARA GEO-LOCALIZACIÓN DE MASCOTAS

En este capítulo se describe el diseño del prototipo inalámbrico para geo-localización de mascotas. La concepción y los requerimientos para el prototipo a utilizarse. Se detallan los diagramas de bloques de cada nodo. Selección del software y hardware adecuado para el funcionamiento del prototipo. Muestra también los diagramas de conexión y los algoritmos de los nodos. Para la visualización de la información almacenada en el prototipo se lo realizo a través de la plataforma Ubidots.

2.1. Requerimientos de *hardware* del prototipo de geo-localización

Los requerimientos del prototipo para geo-localización de mascotas parten del objetivo general del trabajo de titulación, cada uno se detalla a continuación.

- El diseño del prototipo inalámbrico debe monitorear la longitud y latitud para la localización de las mascotas.
- El prototipo debe ser sostenible, eficiente, bajo costo y de larga duración.
- La información obtenida por el prototipo debe ser almacenada en tiempo real.
- Visualizar la ubicación de las mascotas a través de la aplicación Google Maps para su localización.

2.1.1. *Concepción general del prototipo*

Establecidos los requerimientos del prototipo, continuamos con la concepción general del prototipo inalámbrico de acuerdo a la Figura 1-2. Para la comunicación de la red de sensores se utiliza tecnología Zigbee.

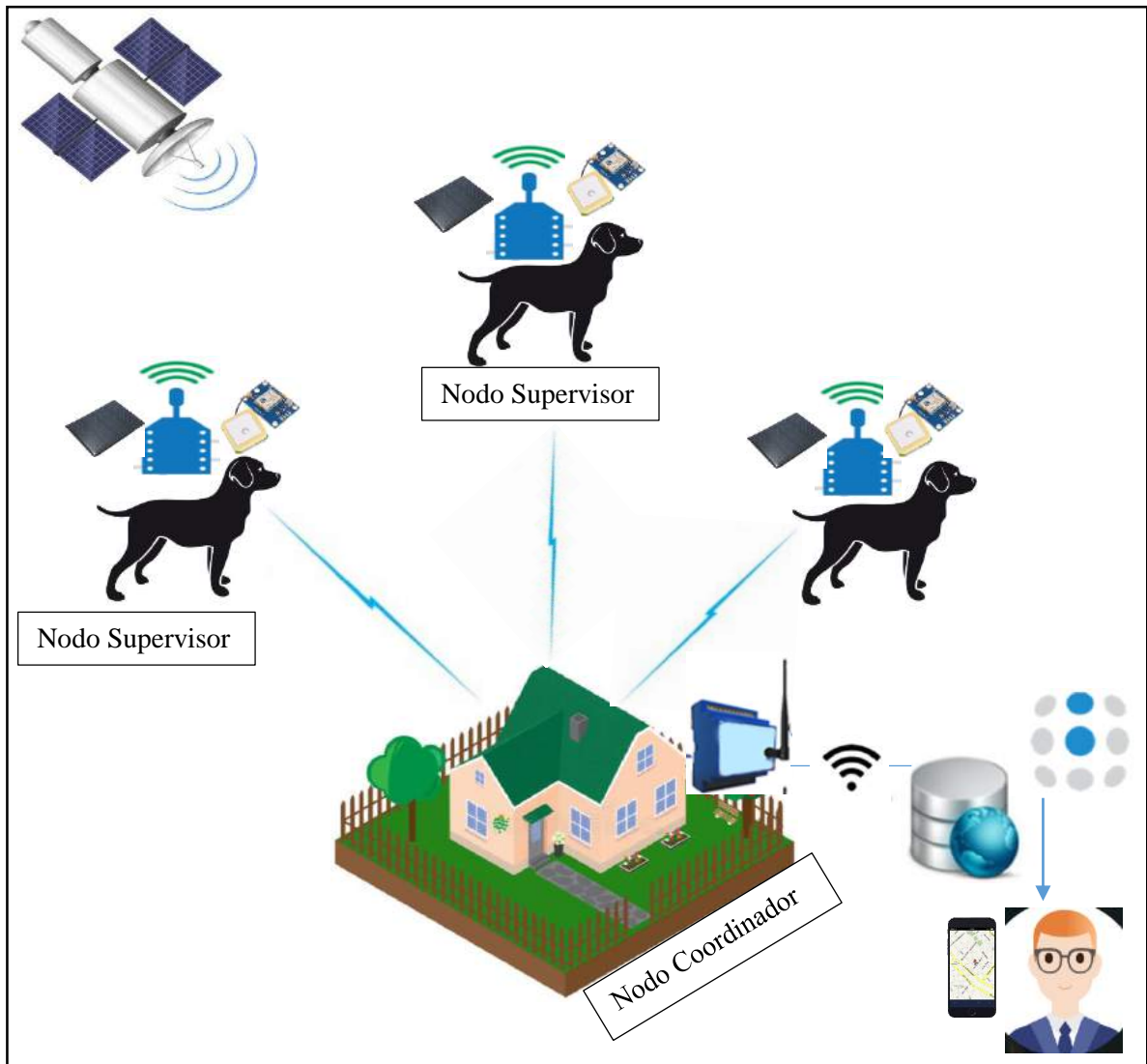


Figura 1-2: Concepción general del prototipo.

Realizado por: AVILA, Doris, 2019

2.1.2. *Arquitectura del Prototipo*

Consta de dos bloques tanto del nodo coordinador como el nodo de adquisición. En cada uno se muestra los diagramas de bloques y su funcionalidad.

- **Diagrama de bloques del nodo de supervisión**

La figura 2-2 indica el diagrama de bloques del nodo de supervisión, consta de 4 bloques:

1. El bloque de obtención de datos, se encarga de leer el puerto serial del GPS para obtener la latitud y longitud.
2. El bloque de procesamiento de la información está compuesto por una tarjeta de desarrollo arduino uno que utiliza salidas digitales para la obtención de los datos.
3. La fuente de alimentación del prototipo es mediante un panel solar recargable por una batería.
4. El bloque de transmisión zigbee envía los datos obtenidos del nodo supervisor al nodo propietario o coordinador.

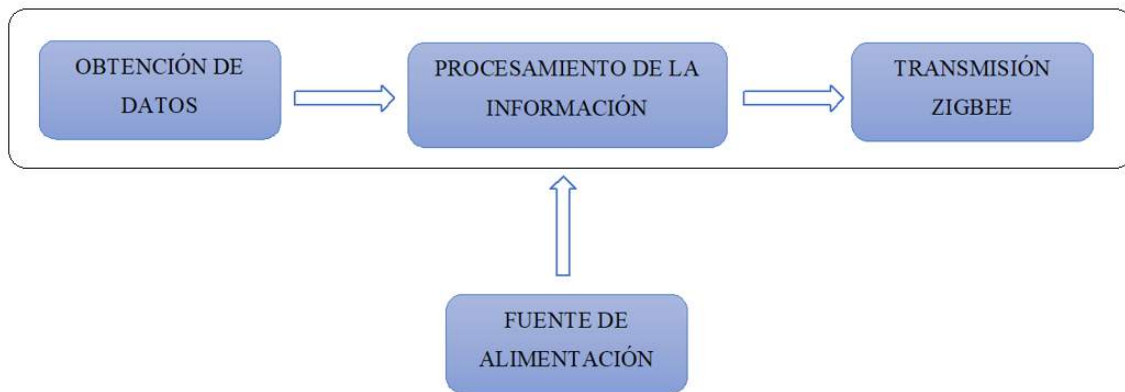


Figura 2-2: Diagrama de bloques del nodo supervisor

Realizado por: AVILA, Doris, 2019

○ **Diagrama de bloques del nodo propietario o coordinador**

La figura 3-2 se indica el diagrama de bloque del nodo propietario, constituido por 4 bloques.

1. El bloque recepción zigbee que se encarga de recibir las tramas transmitidas por los nodos de supervisión.
2. El bloque procesamiento de información está formado por un Nodemcu esp 8266 para enviar la información al bloque de visualización.
3. El bloque de fuente de alimentación proporciona energía a todo el bloque del nodo coordinador.
4. El bloque de visualización muestra la longitud y latitud, fecha, hora de la ubicación de la mascota a través de la aplicación Ubidots y el recorrido mediante Google Map.

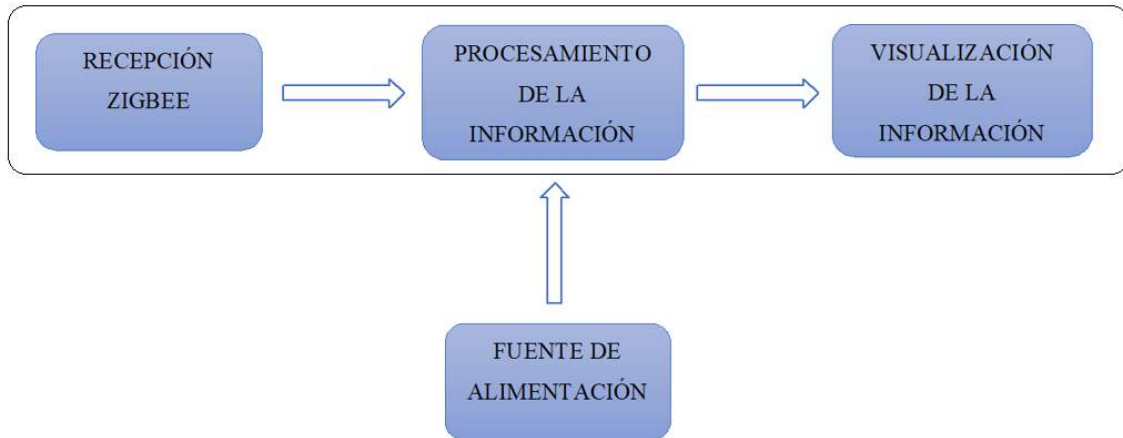


Figura 3-2: Diagrama de bloques del nodo propietario o coordinador

Realizado por: AVILA, Doris, 2019

2.2. Selección de los dispositivos *hardware* para el prototipo

A continuación, se definen las principales características y especificaciones técnicas de funcionamiento de los elementos que conforman el prototipo.

2.2.1. *Tarjetas de Desarrollo para el prototipo de geo-localización*

En la elección de las tarjetas de desarrollo que intervienen en el prototipo se toma en cuenta las siguientes características como su bajo costo, el consumo de energía, número de terminales disponibles y que sea de software libre para su funcionamiento con otros dispositivos.

- **Arduino Uno**

Arduino Uno R3 como se muestra en la figura 4-2 está basada en una placa de microcontrolador ATmega328P que cuenta con 14 puertos digitales de entrada y salida y velocidad de reloj de 16 MHz, lo que permite la selección de Arduino para el desarrollo del prototipo es que es de bajo costo, se puede disponer de varios terminales I/O, el tipo de comunicación que dispone y software es libre. En la tabla 1-2 se presenta las especificaciones de la tarjeta Arduino Uno R3.



Figura 4-2: Arduino Uno

Realizado por: AVILA, Doris, 2019

Tabla 1-2: Especificaciones principales del Arduino UNO

Características	Arduino Uno R3
Microcontrolador	ATmega328P
Tensión de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (límite)	6-20V
Pines digitales de E/S	14 (6 proporcionan salida PWM)
PWM Digital I/O pins	6
Clavijas de entrada analógicas	6
Corriente DC por pin de E/S	20mA
Corriente DC para 3.3V	50mA
Memoria flash	32KB (0.5KB utiliza el cargador para el arranque)
SRAM	2KB
EEPROM	1KB
Velocidad de reloj	16MHz
Led_Builtin	13
longitud	68.6mm
Anchura	53.4mm
Peso	25g

Realizado por: AVILA, Doris, 2019

Fuente: <https://store.arduino.cc/usa/arduino-uno-rev3>

2.2.2. NodeMCU ESP8266

Pequeña placa conocida como ESP8266 con un chip Wi-Fi de bajo costo como se muestra en la Figura 5-2. Ofrece entre las ventajas la incorporación de un regulador de tensión integrado, así como un puerto USB de programación el cual puede ser LUA o IDE de Arduino. En la tabla 2-2 se describe las especificaciones de la tarjeta Nodemcu.

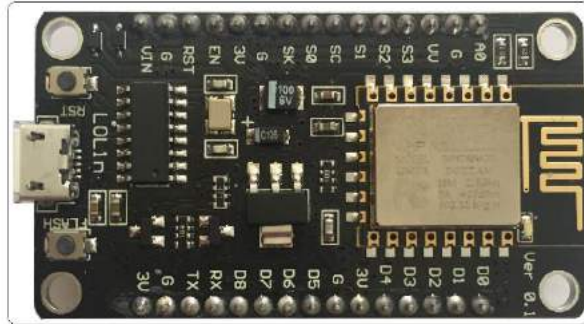


Figura 5-2: NodeMCU ESP8266

Realizado por: AVILA, Doris, 2019

Tabla 2-2: Especificaciones del Nodemcu ESP8266

Parámetros	Descripción
Procesador	ESP8266
Voltaje de Alimentación (USB)	5V DC
Voltaje de Entradas/Salidas	3.3V DC
Frecuencia de procesador	80MHz-160MHz
RAM	32KB
Data RAM	96KB
Memoria Flash Externa	4MB
Pines digitales GPIO	17
Antena	PCB
WiFi	802.11.b/g/n
Seguridad	WPAN/WPAN2
Corriente de operación	200 Ma
Protocolos	TCP, UDP

Realizado por: AVILA, Doris, 2019

2.2.3. Módulo Xbee S2C

Fabricados por Digi Interntional, Inc, son módulos de un bajo costo que permite la comunicación sin cables utilizando el protocolo de comunicación ZigBee (802.15.4) entre dispositivos electrónicos, como se observa en la Figura 6-2 este módulo puede organizar redes de conexión como son punto a punto, punto a multipunto, broadcast y malla (MCI Electronics, 2018). Trabaja a una frecuencia de 2.4 GHz, como se muestra en la tabla 3-2 las especificaciones del módulo S2C.



Figura 6-2: Xbee S2C y Xbee explorer

Realizado por: AVILA, Doris, 2019

Tabla 3-2: Especificaciones del módulo Xbee S2C

Especificaciones	Descripción
Frecuencia de operación	2.4 GHz
Velocidad de transmisión	250 kb/s
Corriente de Tx	28 a 31 mA
Potencia de Rx	33 a 45mA
Rango de señal	1200 metros
Sensibilidad	-100 a -102dBm
Voltaje	2.1 ~ 3.6 V
Técnicas de modulación	DSSS
Interfaz de comunicación	SPI, UART
Encriptación	128 bits
Memoria Flash	32 KB
Memoria RAM	2kB
Temperatura de operación	-40°C ~ 85°C

Realizado por: AVILA, Doris, 2019

2.2.4. Módulo GPS

Para la geo-localización se utiliza el módulo GPS Ublox NEO-6M con una antena de cerámica, de un tamaño pequeño y práctico como se muestra en la figura 7-2. El puerto serial UART permite la comunicación, un voltaje de alimentación que va desde los 3 a 5 voltios que permite medir los parámetros como latitud, longitud permitiendo usarla en las plataformas como Arduino, Rasperry Pi entre otras. En la tabla 4-2 se indica las especificaciones del módulo GPS.

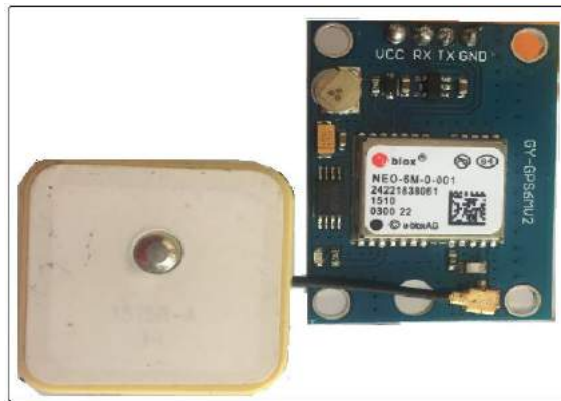


Figura 7-2: GPS Ublox NEO-6M

Realizado por: AVILA, Doris, 2019

Tabla 4-2: Especificaciones GPS Ublox NEO-6M

Especificaciones	Descripción
Voltaje de alimentación	3 – 5 VDC
Interface	Serial UART 5V
Sistema de coordenadas	WGS-84
Sensibilidad de captura	-148dBm
Sensibilidad de rastreo	-161 dBm
Máxima altura medible	18000
Máxima velocidad	515 m/s
Exactitud	1micro segundo
Batería de respaldo	MS621FE
Tamaño de la antena	25mm x 25mm
Tamaño del módulo	25mm x 35mm

Antena	Cerámica
--------	----------

Realizado por: AVILA, Doris, 2019

2.2.5. Panel solar

También llamado módulo fotovoltaico compuesto de silicio cristalino que se compone de células conectadas de silicio que proporciona energía alternativa como se muestra en la figura 8-2. El panel solar utiliza la energía solar para poder convertirla en electricidad, existen diversos modelos y tamaños en la tabla 7-2 que indica especificaciones técnicas del panel a utilizar en el prototipo.

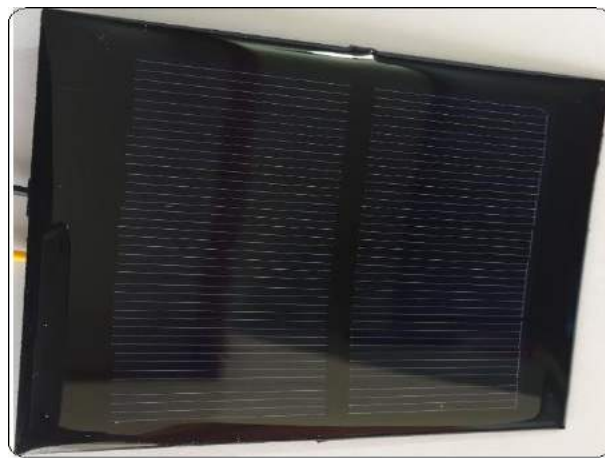


Figura 8-2: Panel Solar

Realizado por: AVILA, Doris, 2019

Tabla 5-2: Especificaciones técnicas del panel solar

Especificaciones	Descripción
Voltaje	5 V
Corriente nominal	100 mA
Dimensiones	70*70mm

Realizado por: AVILA, Doris 2019

2.2.6. Batería de Litio

Es una batería de litio, que se parece mucho a la pila tipo AA, pero tiene en la salida el voltaje 3,7 V y capacidad de 5000 mAh, cuenta con un terminal positivo y negativo como se observa en la figura 9-2.



Figura 9-2: Pila Batería

Realizado por: AVILA, Doris, 2019

2.2.7. Esquemas de conexión del prototipo para la geo-localización de mascotas

En este apartado se describe los dispositivos y las conexiones de cada nodo que está compuesto el prototipo de geo-localización para mascotas.

- Esquema de conexión del nodo de supervisión

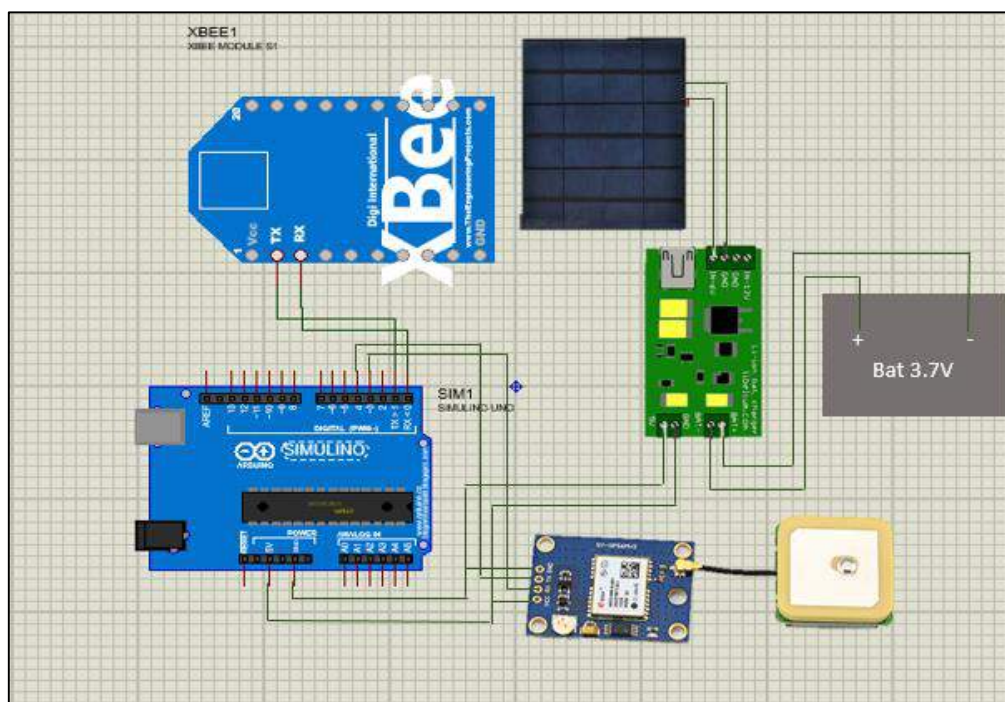


Figura 10-2: Esquema de conexión del nodo de supervisión

Realizado por: AVILA, Doris, 2019

El diagrama de conexión de este nodo se muestra en la Figura 10-2, sus componentes son: un arduino uno, un módulo Xbee s2c, un módulo GPS NEO-6M, un panel solar, un regulador y una batería. Cada componente se conecta como se indica a continuación:

- La tarjeta de desarrollo Arduino está conectado con el módulo Xbee mediante los terminales de TX y RX de cada uno, para la alimentación del Xbee se utiliza los 5 voltios y GND de la tarjeta de desarrollo arduino.
- Los terminales analógicos del arduino 3(RX) y 4(TX) están conectados con los terminales RX y TX del módulo GPS respectivamente. Los terminales de 5V y GND de la tarjeta de desarrollo arduino están conectados a los terminales de 5V y GND del GPS.
- El panel solar está conectado mediante los terminales (positivo y negativo) al convertidor de voltaje de la batería de litio.
- Mediante el convertidor de voltaje del panel solar, se abastece energía al nodo utilizando los terminales de 5V y GND que están conectados a los terminales de 5V y GND de la tarjeta de arduino Uno.

- **Esquema de conexión del nodo coordinador**

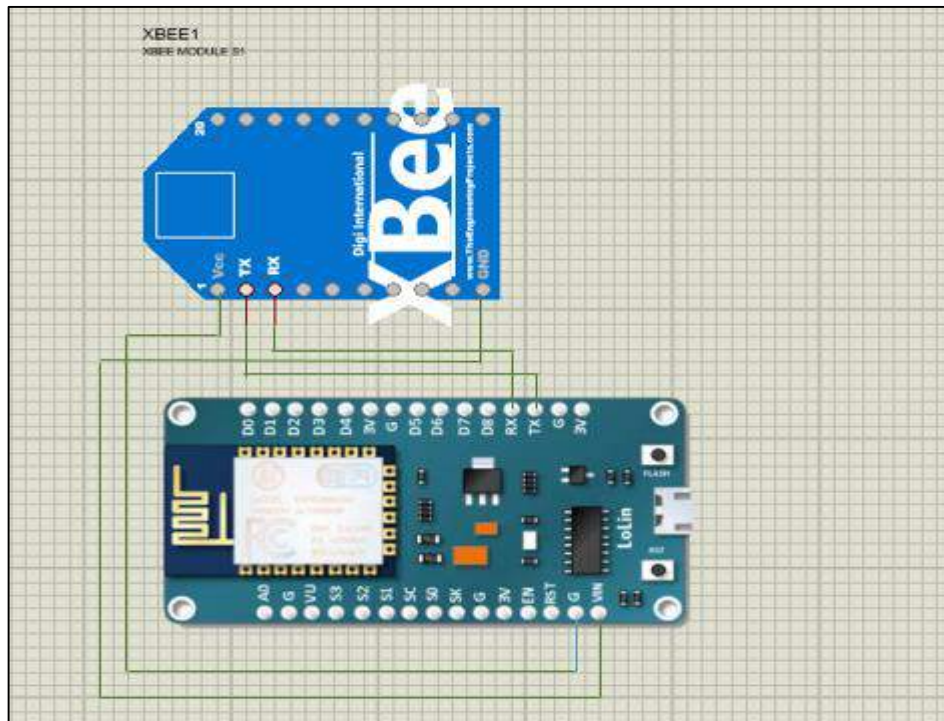


Figura 11-2: Esquema de conexión del nodo coordinador

Realizado por: AVILA, Doris, 2019

El diagrama de conexión del nodo coordinador que se muestra en la Figura 11-2, tiene los siguientes componentes: módulo Xbee s2c y la tarjeta de desarrollo Nodemcu ESP8266. Cada componente se conecta como se indica a continuación:

- El Nodemcu está conectado al módulo Xbee mediante los terminales de TX y RX que cada uno posee, para la alimentación se utiliza los 5V y GND que el Nodemcu ESP8266 suministra. Este nodo se encarga de almacenar los datos en el servidor donde los usuarios pueden visualizar en la red para su respectivo análisis.

2.3. Software de desarrollo para el prototipo de geo-localización

Para la configuración del prototipo se necesita implementar el software que ayude a manipular la parte física, a continuación se detalla las herramientas utilizadas para el respectivo diseño.

2.3.1. Xctu

Es una multiplataforma gratuita que puede ser usada en Windows, MacOs y Linux, este software permite configurar, actualizar firmware y modificar los módulos XBee. Xctu presenta algunas funcionalidades como la administrar y configurar varios dispositivos de RF, la más importante es que restaura sin novedades el firmware y además trabajan en modo de consola API y AT. Su principal ventaja es que incluye un conjunto de herramientas se vienen integradas y se puede ejecutar sin un módulo de RF además las descripciones y los módulos se pueden ver (Digi International Inc., 2015, p. 3).

2.3.2. Arduino IDE

IDE por sus siglas en ingles que significa Integrated Development Environment y en español Entorno de Desarrollo Integrado es un programa de código abierto. Para crear los programas que se ejecutaran y controlaran los nodos del prototipo se usó la versión IDE 1.8.8.

2.3.3. Ubidots

Es una plataforma creada sin un código enfocado en el internet de las cosas permitiendo la implementación y administración de varios dispositivos, disponiendo de la información en tiempo real para que el usuario pueda visualizar las entradas de los sensores a la nube. La plataforma de Ubidots es compatible con algunas opciones de hardware, y protocolos de conectividad, la comunicación, compatible con alguna tarjeta de desarrollo entre ellas está el arduino, rasperry pi

2.3.4. Programación de los módulos de comunicación

El *software* XCTU se utiliza para la configuración de los módulos Xbee, además este *software* nos permite comprobar el funcionamiento de la red de sensores.

- **Nodo coordinador**

Los parámetros que se configuraron en este nodo para que funcionen de manera correcta son los siguientes:

- Pan ID (ID): se estable un número para la autenticación de la red.
- Coordinador Enable (CE): indica el valor 1 para que el módulo trabaje como coordinador.
- Serial Number High y Serial Number Low (SH+SL): dirección de origen compuesta por 64 bits para el módulo Xbee.
- Destination Address High y Destination Address Low (DH+DL): dirección de destino establecida como 0000FFFF para transmitir un mensaje de broadcast a los nodos de la red de sensores.
- 16-bit Parent Address (MP): dirección de 16 bits establecida para cada nodo.
- Baud Rate (BD): es la velocidad de trasmisión para la red.
- Api Enable (AP): modo de transmisión.
- Api Output Mode (AO): se estable la comunicación en modo API. Los parámetros que no mencionan son configurados por defecto, la figura 12-2 indica la configuración de los parámetros anteriormente mencionados del nodo coordinador.



Figura 12-2: Configuración del nodo coordinador.

Realizado por: AVILA, Doris, 2019

- **Nodo Supervisor**

Los parámetros que se configuraron en este nodo para que funcionen de manera correcta son los siguientes:

- Pan ID (ID): se estable un número para la autenticación de la red.
- Coordinador Enable (CE): indica el valor 0 para que el modulo trabaje como dispositivo final.
- Serial Number High y Serial Number Low (SH+SL): dirección de origen compuesta por 64 bits para el módulo Xbee.
- Destination Address High y Destination Address Low (DH+DL): dirección de destino establecida como 00000000 para que la comunicación sea solo con el nodo coordinador.
- Channel Verification (JV): se pone el valor en 1 para afirmar que tanto el dispositivo final y el coordinador utilice el mismo canal.
- 16-bit Parent Address (MP): dirección de 16 bits establecida para cada nodo.
- Baud Rate (BD): es la velocidad de transmisión para la red.
- Api Enable (AP): se configura el modo de transmisión de forma transparente.
- Api Output Mode (AO): se estable la comunicación en modo API. Los parámetros que no mencionan son configurados por defecto, la figura 13-2 indica la configuración de los parámetros anteriormente mencionados del dispositivo final.

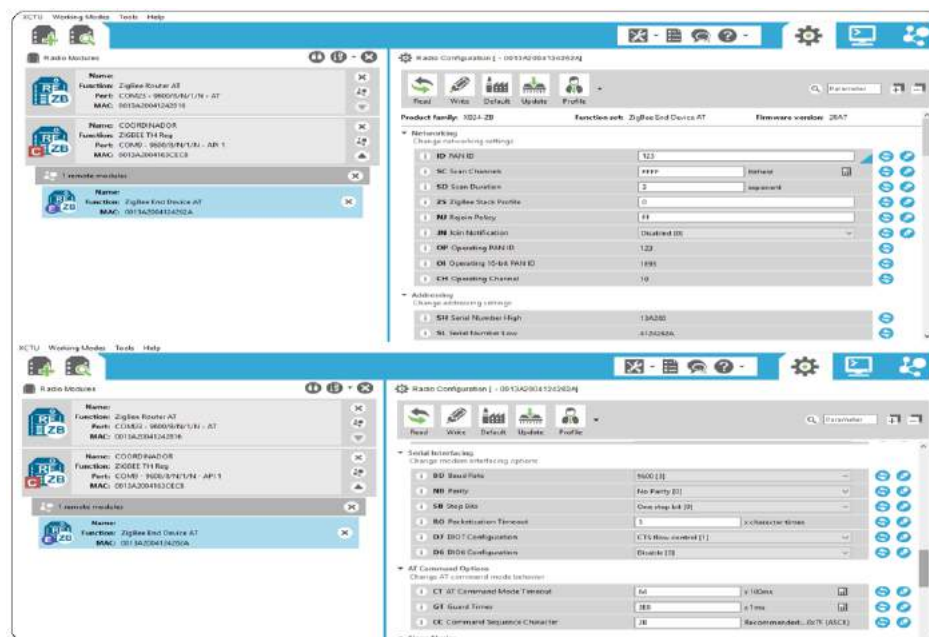


Figura 13-2: Configuración del nodo supervisor.

Realizado por: AVILA, Doris, 2019

2.3.5. *Requerimientos del Software del prototipo de geo-localización de mascotas*

Los nodos que conforman el prototipo para geo-localización de mascotas necesitan los requerimientos que se detallan a continuación:

- **Requerimientos del nodo supervisor**
- Adquirir los datos del módulo GPS en tiempo real, mostrándolos en grados, minutos y segundos de la latitud y longitud.
- Transmitir los valores del GPS al nodo coordinador a través de la comunicación zigbee.
- **Requerimientos del nodo coordinador**
- Recibir las tramas zigbee enviadas desde el nodo supervisor mediante la comunicación zigbee.
- Enviar los datos de la latitud y longitud al servidor para que el usuario, en este caso el propietario de la mascota pueda visualizar en cualquier momento mediante el acceso a internet.

2.3.6. *Diagramas de flujo del prototipo de geo-localización de mascotas*

Los diagramas de flujo para la programación de cada nodo del prototipo de geo-localización de mascotas se indican a continuación:

- **Diagrama de flujo del nodo supervisor de datos**

El diagrama de flujo para la programación del nodo supervisor se muestra en la figura 14-2. Cada paso de la programación se detalla de la siguiente manera:

Inicialización

- Declarar la librería **TinyGPS**, esta librería es útil para que trabaje el módulo GPS.

- Declarar la variables para leer la longitud y latitud del GPS: latitud, longitud ya que nos permite almacenar los datos del GPS.
- Inicializar el puerto serial para el GPS.

Bucle Repetitivo

- Si el puerto serial del GPS está disponible se leerán los datos.
- Se mostraran los datos de latitud y longitud obtenidos por el GPS.
- Enviar los datos obtenidos en forma de trama del nodo supervisor mediante la comunicación zigbee al nodo coordinador de la red.

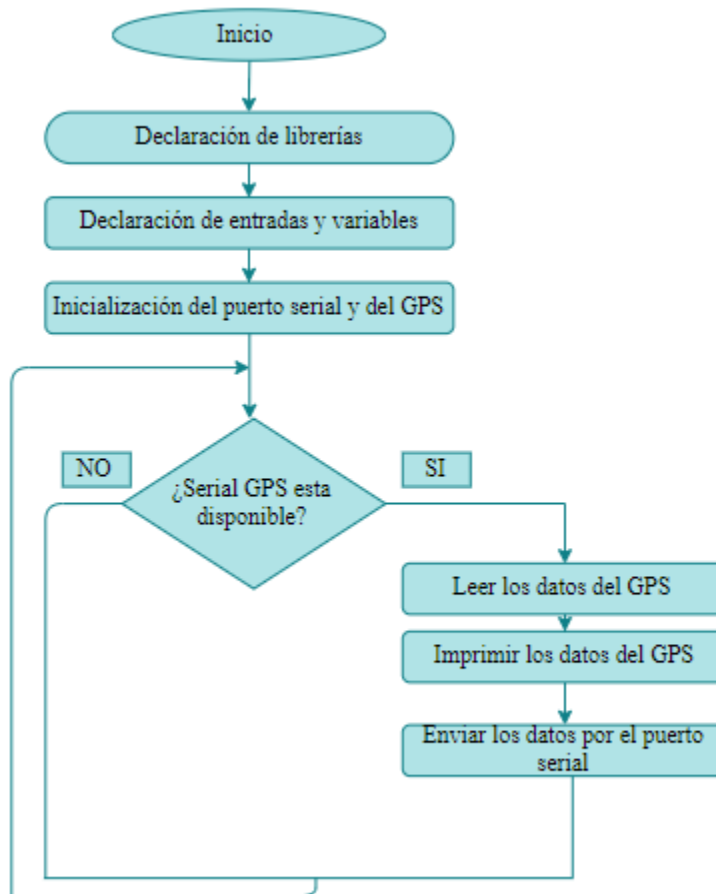


Figura 14-2: Diagrama de flujo del nodo supervisor de datos

Realizado por: AVILA, Doris, 2019

- **Diagrama de flujo del nodo coordinador**

El diagrama de flujo para la programación del nodo coordinador se muestra en la figura 15-2. Cada paso de la programación se indica a continuación:

Inicialización

- Declarar la librería ESP8266WiFi.h necesaria para establecer comunicación TCP entre la tarjeta de desarrollo Nodemcu y la red Wi-fi.
- Establecer el nombre y contraseña de la red para acceder a la red Wi-fi, también se debe indicar el ID para ingresar a la plataforma Ubidots.
- Iniciar el puerto serial para establecer la comunicación al módulo Xbee.

Bucle Repetitivo

- Verificar si llegan datos que son enviados por el nodo supervisor.
- Si los datos llegan en forma de trama se obtiene la información para almacenar en cada variable.
- Crear la conexión TCP para enviar los datos a la plataforma Ubidots, además se debe tomar en cuenta el tiempo que se tarda en transmitir la información, finalmente se cierra la conexión establecida.

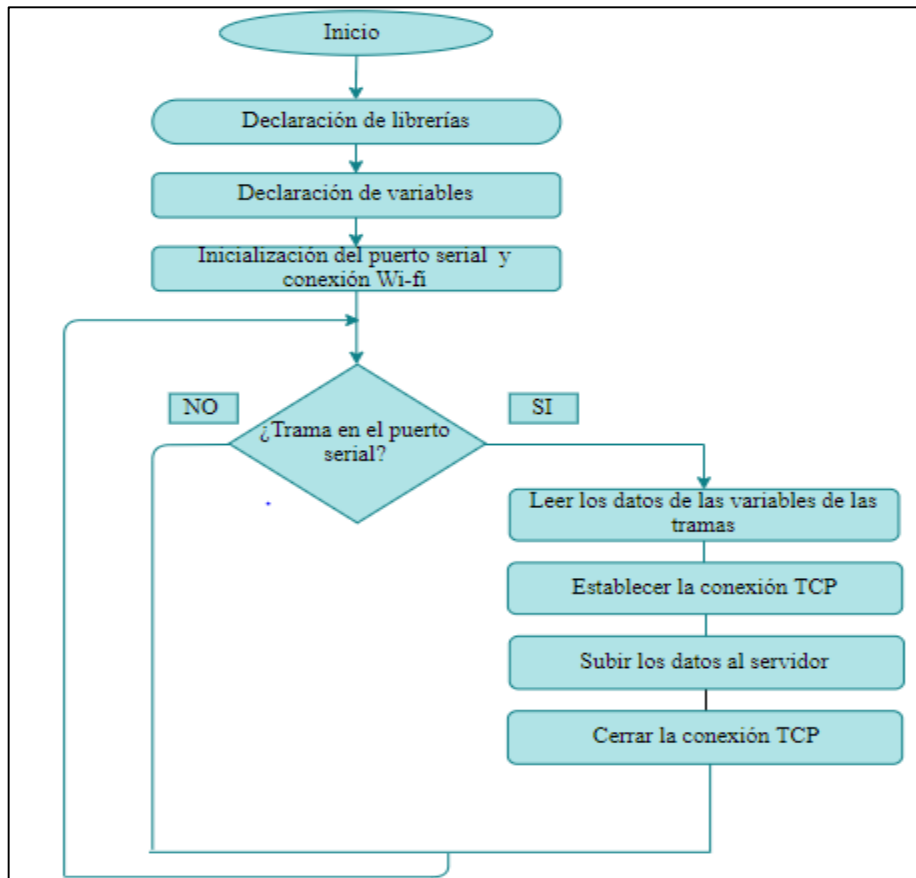


Figura 15-2: Diagrama de flujo del nodo coordinador.

Realizado por: AVILA, Doris, 2019.

2.4. Visualización de los datos en la plataforma Ubidots

Ubidots es una herramienta libre que nos permiten crear dispositivos y luego las variables a utilizar. Para almacenar los datos se creó un dispositivo llamado geo-localización y una variable denominada position como se indica en la figura 16-2, además hay que tener una cuenta donde Ubidots le genera un token para el Nodemcu Esp8266.



Figura 16-2: Visualización de la variable creada en Ubidots.

Realizado por: AVILA, Doris, 2019.

La plataforma almacena los datos y nos permite generar un histórico y gráficos de los datos de latitud y longitud enviados por el nodo supervisor. Nos permite generar la ubicación del nodo que va en la mascota mediante google maps, además se puede enviar un sms al teléfono acerca de su mascota y ver en qué lugar se encuentra, como se observa en la figura 17-2.

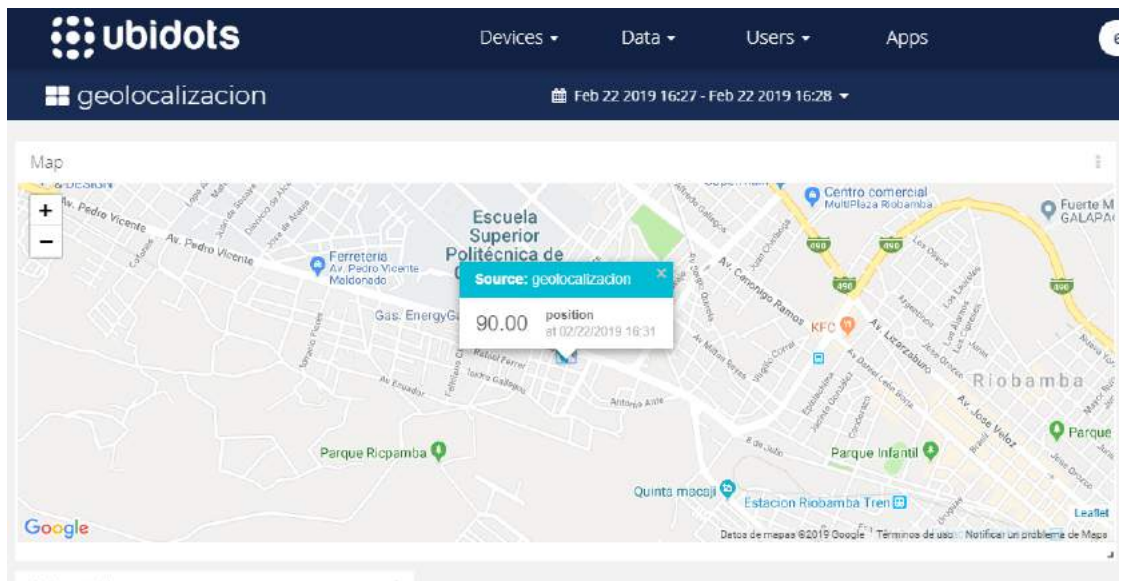


Figura 17-2: Ubicación del nodo supervisor en Google Map.

Realizado por: AVILA, Doris, 2019

2.5. Diseño de ensamblaje para el prototipo de geo-localización

Mediante el *software* AutoCAD se diseñó los modelos de las cajas que serán utilizadas para los dispositivos tanto como para el nodo supervisor y nodo coordinador. En la figura 18-2 se observa el diseño de la caja para el nodo coordinador con las dimensiones de ancho 10 cm de y de alto 4 cm, la imagen de color verde se ve en forma plana el modelo de la caja y el diseño de color blanco se indica la forma que tendrá el ensamblaje de la cajón para los dispositivos.

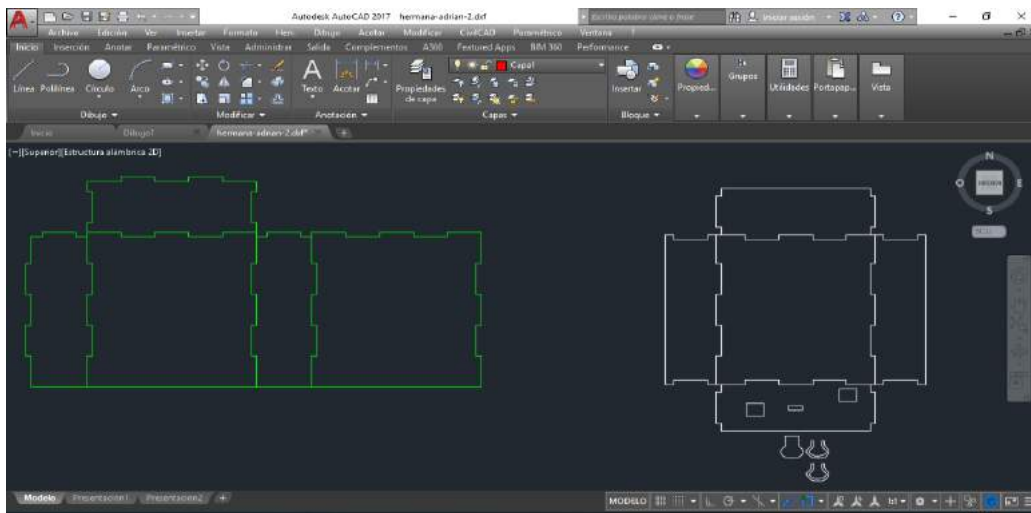


Figura 18-2: Diseño de la caja para el nodo coordinador.

Realizado por: AVILA, Doris, 2019

Para el nodo de supervisión el diseño se indica en la figura 19-2, en forma plana se ve el modelo de la caja que va a contener a los dispositivos y la parte roja será donde este el panel solar, las dimensiones son: de ancho 8.5 cm, largo 14.5 y alto 4 cm.

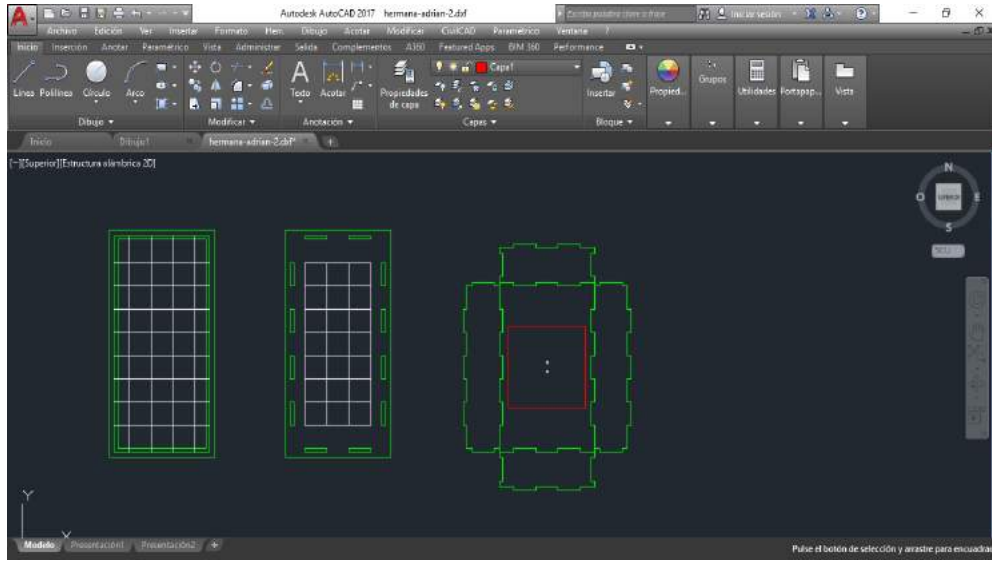


Figura 19-2: Diseño de la caja para el nodo supervisor.

Realizado por: AVILA, Doris, 2019

CAPITULO III

3. PRUEBAS Y RESULTADOS

En el siguiente capítulo se muestra las pruebas y los resultados adquiridos por los dispositivos de cada nodo que está compuesto el prototipo de geo-localización de mascotas. Para evaluar el prototipo se realizaron pruebas de funcionamiento y alimentación. Se comprobó que el prototipo sea estable. Finalmente se detalla el presupuesto económico del prototipo.

3.1. Selección de la topología de red para el prototipo.

Para la implementación de la red de sensores se utilizó la topología estrella simulada en el software Riverbed Modeler. La cual consta de un nodo coordinador y ocho dispositivos finales como se indica en la figura 1-3.

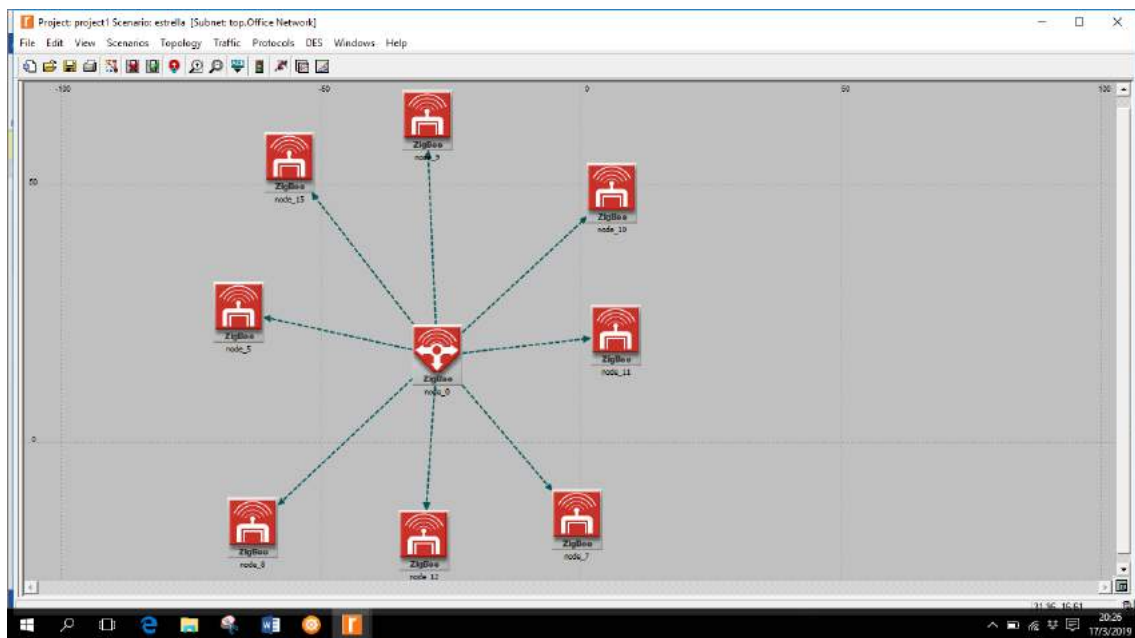


Figura 1-3: Topología Estrella

Realizado por: AVILA, Doris, 2019

La Figura 2-3 se muestra los parámetros de la carga y el retardo promedio de la topologías simuladas. Por ejemplo en la topología malla tiene un retardo de 0.0070 segundos en cambio la topología de

estrella tiene un retardo de 0.0066 segundos demostrando que esta topología tiene menor retardo. Además se indica que la topología estrella tiene una velocidad de carga de 52,000 bits/sec menor a la topología malla que tiene la velocidad de 39,000 bits/sec. Analizando estos parámetros se determina que la topología estrella es la más óptima para la implementación del prototipo de geo-localización de mascotas.

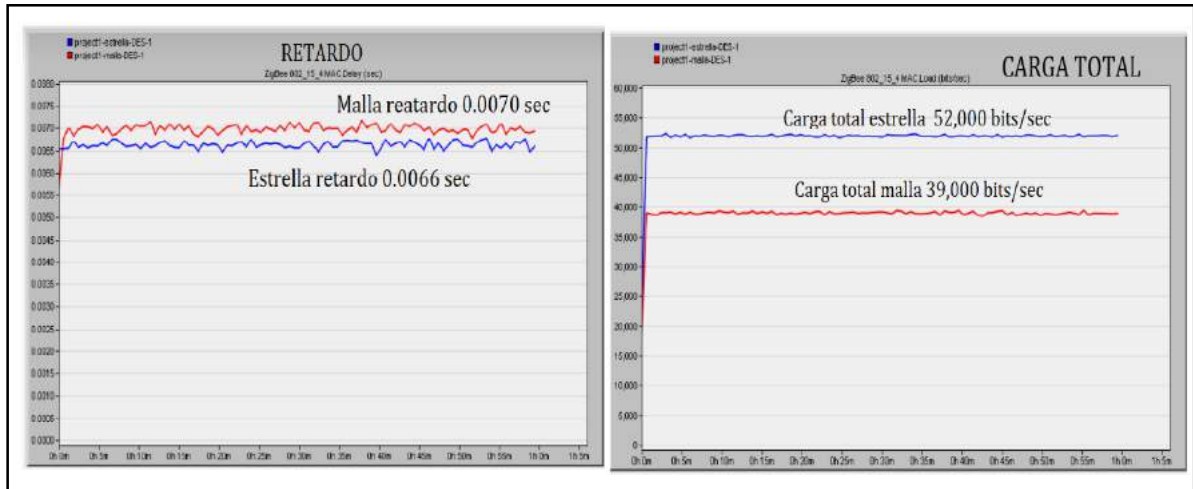


Figura 2-3: Carga total y retardo entre la topología estrella y malla

Realizado por: AVILA, Doris, 2019

3.2. Pruebas para la comunicación de los nodos del prototipo

Esta prueba es de gran importancia porque se comprueba que los datos que adquiere el nodo supervisor lleguen al nodo coordinador. Para comprobar que esto suceda se realizó la comunicación entre nodos y el alcance de comunicación de los nodos.

3.2.1. Comunicación de los nodos

Se envió 10 datos del nodo supervisor para verificar que lleguen en forma de trama al nodo coordinador en un lapso de 4 minutos.

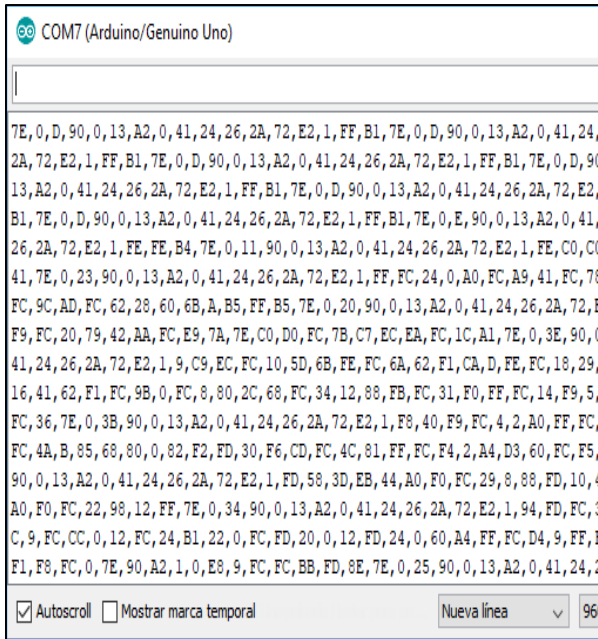


Figura 3-3: Comunicación entre los nodos

Realizado por: AVILA, Doris, 2019

La Figura 3-3 muestra las tramas que son recibidas por el nodo coordinador, verificando la integridad de los datos entre los nodos del prototipo.

3.2.2. Alcance de comunicación de los nodos

Esta prueba nos ayuda a comprobar la distancia máxima entre nodos. La tabla 1-3 indica la distancia para el funcionamiento del prototipo. Para determinar el alcance, los módulos se iban separando en intervalos de 5 metros hasta que la señal se pierda o no sea receptada. A la señal se le clasifico de la siguiente manera:

- 1 a 50m (Excelente): No hay pérdidas de paquetes y la comunicación se realiza sin ningún problema.
- 51 a 70m (Alta): Los paquetes se transmiten de forma estable.
- 71 a 90m (Media): La comunicación presenta retardos, pero aun así se envían los paquetes.
- 91 a 100m (Bajo): Los datos no son íntegros son alterados y se pierden paquetes.
- 101 a 150m (No operacional): La comunicación entre nodos no existe.

Tabla 1-3: Distancia para el funcionamiento del prototipo

Distancia (m)	Calificación de la señal
1-50	Excelente
51-70	Alta
71-90	Media
91-100	Bajo
101-150	No operacional

Realizado por: AVILA, Doris, 2019.

De acuerdo a la calificación de la señal se concluye que el alcance máximo de la comunicación entre el nodo coordinador y supervisor es de 100 metros para la localización indoor.

3.3. Caracterización del nodo supervisión

La caracterización del nodo de supervisión se indica en la figura 4-3 donde se observa a lado izquierdo la parte interna con sus diferentes dispositivos y a la derecha su parte ensamblada.

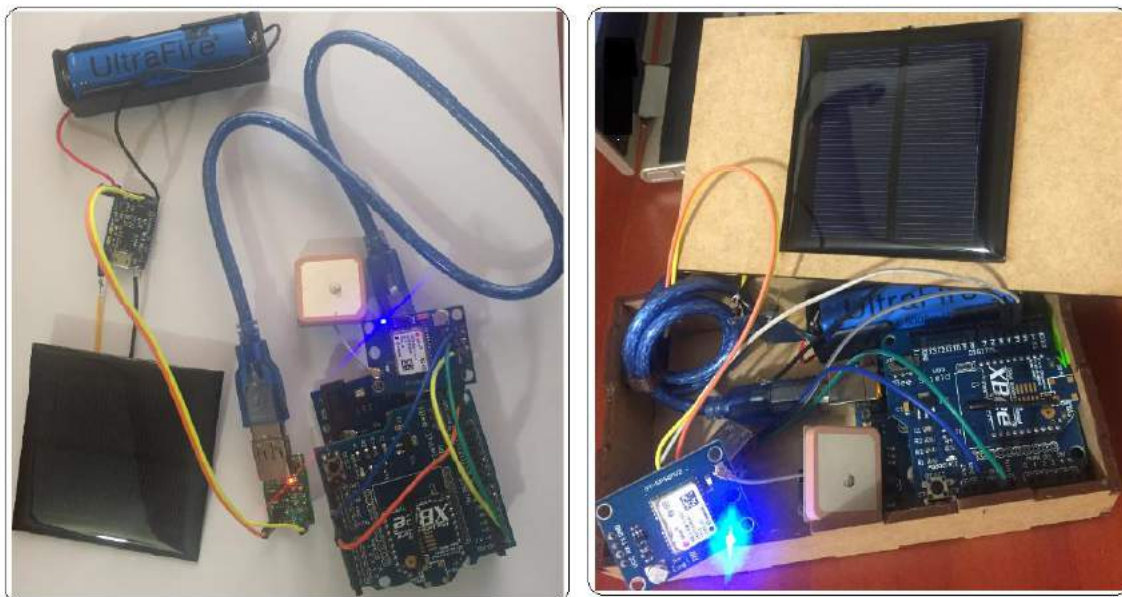


Figura 4-3: Nodo de supervisión

Realizado por: AVILA, Doris, 2019

Se comprobó su funcionamiento realizando pruebas de lectura del GPS, las mismas que son de gran utilidad para enviar los datos al siguiente nodo, como se muestra en la figura 5-3.

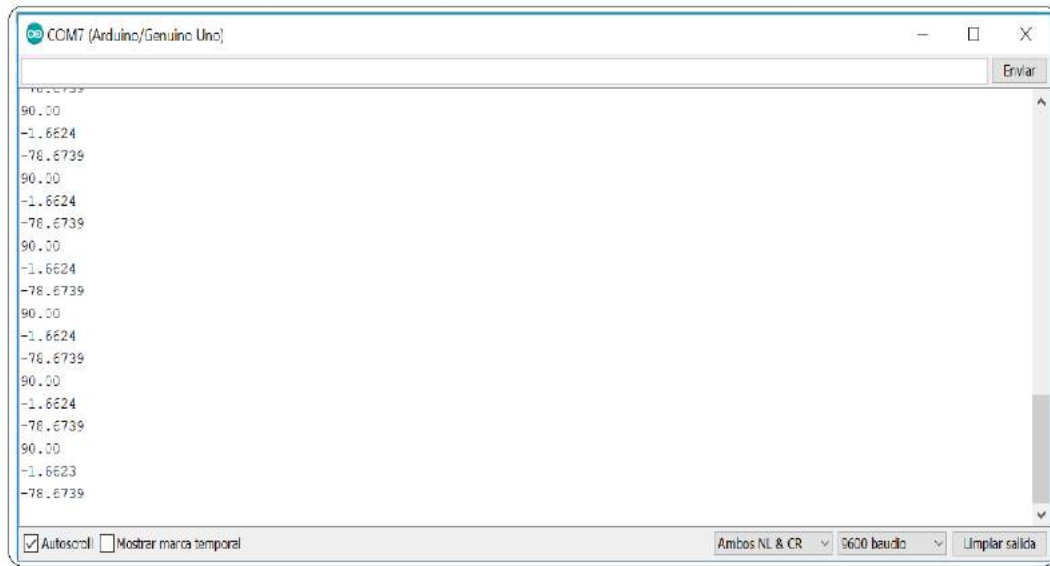


Figura 5-3: Datos de lectura de GPS

Realizado: AVILA, Doris, 2019

3.3.1. Repetitividad del nodo de supervisión

El objetivo fundamental de esta prueba es encontrar la estabilidad del prototipo, para lo cual se tomaron 20 muestras de la latitud y longitud del sensor en un lapso de 5 minutos, en un intervalo de 15 segundos, cada dato es almacenado en la plataforma Ubidots.

Tabla 2-3: Valores obtenidos para la repetitividad del nodo de supervisión

Numero de muestra	Tiempo	Latitud	Longitud	Latitud	Longitud	Latitud	Longitud
		(° ' ")	(° ' ")	Decimal	Decimal	Decimal	Decimal
1	17:10:15	1°39'43.92"	78°40'26.04"	1,6622	78,6739	1,6622	78,6739
2	15:10:30	1°39'44.28"	78°40'26.40"	1,6623	78,674	1,6622	78,6739
3	15:10:45	1°39'44.64"	78°40'26.04"	1,6624	78,6739	1,6622	78,6739
4	15:11:00	1°39'43.45"	78°40'26.04"	1,662	78,6739	1,6621	78,6738
5	15:11:15	1°39'45.36"	78°40'25.68"	1,6626	78,6738	1,6621	78,6738
6	15:11:30	1°39'45.36"	78°40'25.68"	1,6626	78,6738	1,6621	78,6738
7	15:11:45	1°39'46.08"	78°40'25.32"	1,6628	78,6737	1,6621	78,6738
8	15:12:00	1°39'46.44"	78°40'25.32"	1,6629	78,6737	1,6621	78,6738
9	15:12:15	1°39'46.44"	78°40'25.32"	1,6629	78,6737	1,6623	78,6739
10	15:12:30	1°39'46.44"	78°40'25.32"	1,6629	78,6737	1,6623	78,6737
11	15:12:45	1°39'46.44"	78°40'25.32"	1,6629	78,6737	1,6623	78,6737

12	15:13:00	1°39'46.44"	78°40'25.32"	1,6629	78,6737	1,6623	78,6737
13	15:13:15	1°39'46.44"	78°40'25.32"	1,6629	78,6737	1,6624	78,6739
14	15:13:30	1°39'46.80"	78°40'25.32"	1,663	78,6737	1,6624	78,6737
15	15:13:45	1°39'46.80"	78°40'25.32"	1,663	78,6737	1,6625	78,6737
16	15:14:00	1°39'46.80"	78°40'25.32"	1,663	78,6737	1,6625	78,6738
17	15:14:15	1°39'46.80"	78°40'25.32"	1,663	78,6737	1,6621	78,6739
18	15:14:30	1°39'45.36"	78°40'25.68"	1,6626	78,6738	1,6623	78,6737
19	15:14:45	1°39'45.36"	78°40'25.68"	1,6626	78,6738	1,6625	78,6737
20	15:15:00	1°39'45.36"	78°40'25.68"	1,6626	78,6738	1,6622	78,6737
Media				1,662705	78,67377	1,66226	78,67379
Desviación estándar				0,00029465	0,00009233	0,000142902	0,0000852
Coeficiente de variación				0.2127%	0.117%	0.8587%	0.1085%

Realizado por: AVILA, Doris, 2019.

La tabla 2-3 indica los valores obtenidos por el sensor de geo-localización, se realiza un análisis estadístico descriptivo en Excel donde se encontrara la media, la desviación estándar y el coeficiente de variación, demostrando que los datos obtenidos por el prototipo tienen una variación entre 0.2127% a 0.8587% en latitud y en longitud 0.117% a 0.1085% demostrando que son valores menores a 5 % según datos de (INE-CHILE, 2016, p. 4), concluyendo que el prototipo implementado es estable.

3.4. Caracterización del nodo coordinador

La figura 6-3 indica el ensamblaje del nodo coordinador. Este módulo es el nodo central de la red de sensores para la geo-localización de mascotas porque su objetivo principal es el de crear la red, al igual que hace la recepción de las tramas del nodo supervisor para almacenar los datos en la plataforma Ubidots. Su funcionamiento es comprobado en las pruebas siguientes.

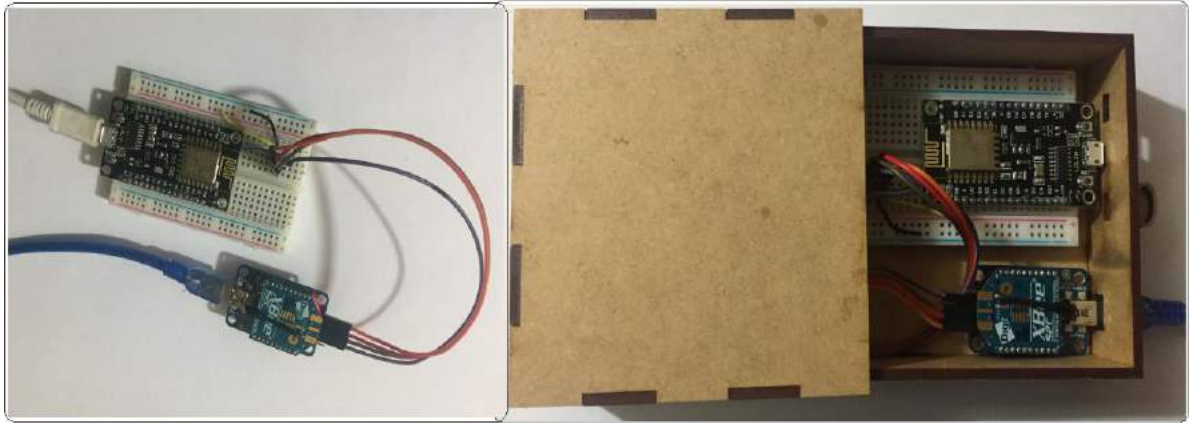


Figura 6-3: Nodo coordinador

Realizado por: AVILA, Doris, 2019

3.4.1. Comunicación entre el nodo coordinador y la plataforma Ubidots.

Esta prueba se realizó para comprobar la integridad de la información. En la figura 7-3 se muestran los datos leídos en el IDE del Arduino y del módulo ESP8266, verificando así que todos los datos son obtenidos en tiempo real.

```
COM17

Making request to Ubidots:

POST /api/v1.6/devices/geolocalizacion HTTP/1.1
Host: industrial.api.ubidots.com
User-Agent: ESP8266/1.0
X-Auth-Token: BBFF-G1J1D0SdpKpRxxH8vvGwEbOSi24NTe
Connection: close
Content-Type: application/json
Content-Length: 72

{"position":{"value":90.0000,"context":{"lat":-1.6622,"lng":-78.6739}}

Ubidots' Server response:

HTTP/1.1 200 OK

 Autoscroll  Mostrar marca temporal
```

Figura 7-3: Datos de comunicación mostrados en el IDE arduino

Realizado por: AVILA, Doris, 2019

3.4.2. Prueba de almacenamiento

La prueba de almacenamiento es para que la información obtenida por el sensor sea almacenada en la plataforma gratuita Ubidots. Para la cual se debe tener comunicación inalámbrica Wi-fi con el módulo Nodemcu esp8266 este se encarga de transmitir los datos al servidor. La figura 8-3 muestra el almacenamiento de los datos en Ubidots.

DATE	VALUE	CONTEXT	ACTIONS
New data available			
2019-02-22 16:02:50 -05:00	90.00	("lat":-1.66,"lng":-78.67)	
2019-02-22 16:02:49 -05:00	90.00	("lat":-1.66,"lng":-78.67)	
2019-02-22 16:02:47 -05:00	90.00	("lat":-1.66,"lng":-78.67)	
2019-02-22 16:02:46 -05:00	90.00	("lat":-1.66,"lng":-78.67)	
2019-02-22 16:02:45 -05:00	90.00	("lat":-1.66,"lng":-78.67)	
2019-02-22 16:02:44 -05:00	90.00	("lat":-1.66,"lng":-78.67)	
2019-02-22 16:02:43 -05:00	90.00	("lat":-1.66,"lng":-78.67)	
2019-02-22 16:02:42 -05:00	90.00	("lat":-1.66,"lng":-78.67)	
2019-02-22 16:02:41 -05:00	90.00	("lat":-1.66,"lng":-78.67)	
2019-02-22 16:02:40 -05:00	90.00	("lat":-1.66,"lng":-78.67)	

Figura 8-3: Almacenamiento de datos en Ubidots

Realizado por: AVILA, Doris, 2019

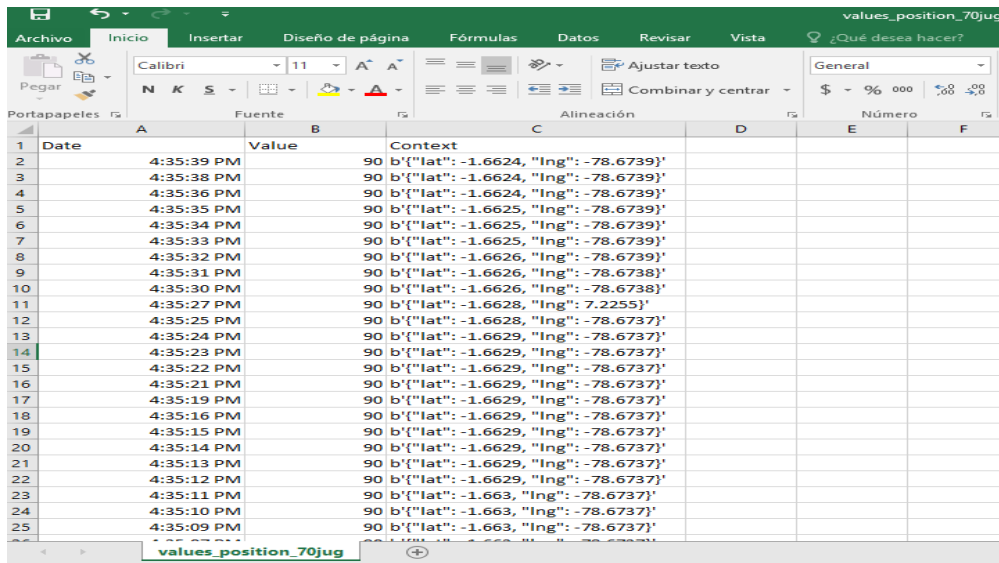
La información es almacenada de forma correcta en el servidor Ubidots, al igual que puede ser descargada en formato .xml, que permite generar reportes gráficos de esta información de acuerdo a la Figura 9-3.



Figura 9-3: Gráfico de la latitud y longitud almacenada en la plataforma

Realizado por: AVILA, Doris, 2019

La figura 10-3 indica los datos extraídos de la plataforma en archivo Excel.



	A	B	C	D	E	F
1	Date	Value	Context			
2	4:35:39 PM		90 b'{"lat": -1.6624, "Ing": -78.6739}'			
3	4:35:38 PM		90 b'{"lat": -1.6624, "Ing": -78.6739}'			
4	4:35:36 PM		90 b'{"lat": -1.6624, "Ing": -78.6739}'			
5	4:35:35 PM		90 b'{"lat": -1.6625, "Ing": -78.6739}'			
6	4:35:34 PM		90 b'{"lat": -1.6625, "Ing": -78.6739}'			
7	4:35:33 PM		90 b'{"lat": -1.6625, "Ing": -78.6739}'			
8	4:35:32 PM		90 b'{"lat": -1.6626, "Ing": -78.6739}'			
9	4:35:31 PM		90 b'{"lat": -1.6626, "Ing": -78.6738}'			
10	4:35:30 PM		90 b'{"lat": -1.6626, "Ing": -78.6738}'			
11	4:35:27 PM		90 b'{"lat": -1.6628, "Ing": 7.2255}'			
12	4:35:25 PM		90 b'{"lat": -1.6628, "Ing": -78.6737}'			
13	4:35:24 PM		90 b'{"lat": -1.6629, "Ing": -78.6737}'			
14	4:35:23 PM		90 b'{"lat": -1.6629, "Ing": -78.6737}'			
15	4:35:22 PM		90 b'{"lat": -1.6629, "Ing": -78.6737}'			
16	4:35:21 PM		90 b'{"lat": -1.6629, "Ing": -78.6737}'			
17	4:35:19 PM		90 b'{"lat": -1.6629, "Ing": -78.6737}'			
18	4:35:16 PM		90 b'{"lat": -1.6629, "Ing": -78.6737}'			
19	4:35:15 PM		90 b'{"lat": -1.6629, "Ing": -78.6737}'			
20	4:35:14 PM		90 b'{"lat": -1.6629, "Ing": -78.6737}'			
21	4:35:13 PM		90 b'{"lat": -1.6629, "Ing": -78.6737}'			
22	4:35:12 PM		90 b'{"lat": -1.6629, "Ing": -78.6737}'			
23	4:35:11 PM		90 b'{"lat": -1.663, "Ing": -78.6737}'			
24	4:35:10 PM		90 b'{"lat": -1.663, "Ing": -78.6737}'			
25	4:35:09 PM		90 b'{"lat": -1.663, "Ing": -78.6737}'			

Figura 10-3: Descarga de datos en un archivo Excel.

Realizado por: AVILA, Doris, 2019.

3.5. Funcionamiento del prototipo de geo-localización.

Posteriormente de validar las pruebas necesarias de cada nodo del prototipo de geo-localización de mascotas, se procede a evaluar el funcionamiento de todo el prototipo es decir la comunicación de la red y el almacenamiento de la información cumpliendo con los requerimientos mencionados.

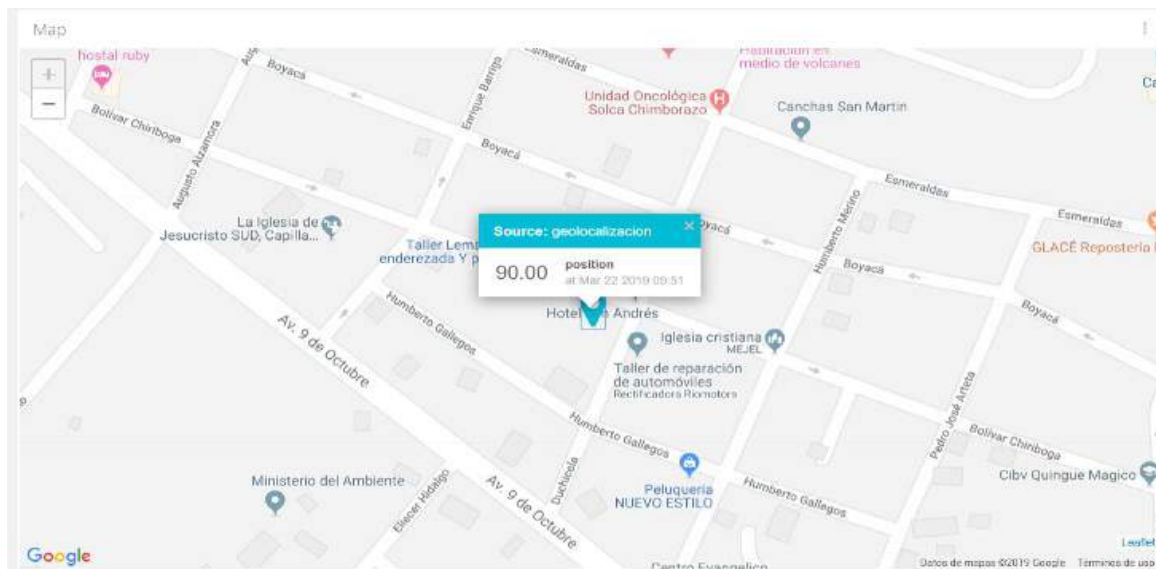


Figura 11-3: Ubicación de la mascota en Google Maps

Realizado por: AVILA, Doris, 2019.

En la figura 11-3 se indica la ubicación de la mascota con fecha y hora dentro del domicilio haciendo uso de la aplicación Google Maps, con esto se demuestra que el prototipo muestra la localización en tiempo real.



Figura 12-3: Prototipo acoplado en la mascota.

Realizado por: AVILA, Doris, 2019.

Se observa en la figura 12-3 que la mascota tiene acoplado el nodo supervisor, el que está sujeto a un chaleco para mayor comodidad, este nos proporcionará los datos de latitud y longitud del sensor GPS para el almacenamiento en el servidor.



Figura 13-3: Delimitación del área para la ubicación de la mascota

Realizado por: AVILA, Doris, 2019.

Para demostrar la localización indoor se delimito un área específica haciendo referencia el domicilio del propietario de la mascota como se muestra en la figura13-3 esto ayudara a tener definido el área y si sale de este lugar le llegara un mensaje de texto con la alerta como se indica a continuación.



Figura 14-3: Mensaje de texto al propietario de la mascota

Realizado por: AVILA, Doris, 2019.

Si la mascota sale del área establecida por el propietario se le notificara mediante un mensaje de texto a su dispositivo móvil de acuerdo a la figura 14-3.

3.6. Prueba de alimentación del prototipo de geo-localización

El objetivo principal de esta prueba es para determinar que el prototipo posea una alimentación continua, por lo que se ha considerado utilizar una batería de li-on con capacidad de 5000mA/h con un voltaje de salida de 3.7V. Se calculó la alimentación del prototipo sumando las corrientes de cada dispositivo del nodo supervisor, para así hallar el consumo máximo de la corriente, teniendo como resultado un consumo de corriente de 310 mA según la tabla 3-3. Por lo que se concluye que con la batería se va alimentar un tiempo de 11,29 horas aproximadamente continuas sin la ayuda del panel solar.

Tabla 3-3: Cálculo para la alimentación con la batería

Nodo	Elementos	Corriente de operación
Supervisión	Arduino Uno	50 mA
	Sensor GPS	150mA
	Módulo Xbee	110mA

Consumo de corriente=	310mA
Capacidad de la batería=	5000mA
Duración en horas=	(capacidad de la batería/consumo del dispositivo) *0,7 (5000mA/310mA) *0,7 11,29 horas

Realizado por: AVILA, Doris, 2019.

3.7. Análisis económico del prototipo de geo-localización

La Tabla 4-3 se detalla el presupuesto económico de los dispositivos utilizados en la implementación del prototipo de geo-localización para las mascotas. Algunos de los dispositivos incluyen el costo de envío.

Tabla 4-3: Presupuesto económico del prototipo implementado

Nodo	Dispositivos	Cantidad	Costo (USD)
Supervisión	Tarjeta de desarrollo Arduino Uno	1	13,00
	Sensor GPS	1	18,00
	Shield para Arduino	1	11,00
	Módulo XBee S2C	1	48,00
	Panel solar	1	7,00
	Convertidor de voltaje	1	10,00
	Pila batería 3.7V	1	14,00
	Ensamblaje	1	40,00
	Cables de conexión	1	2,00
Total del nodo supervisor=			163,00
Coordinador	NodeMCU-ESP8266	1	15,00
	Módulo XBee S2C	1	48,00
	Explorer Xbee	1	15,00
	Cables	1	2,00
	Ensamblaje	1	3,00
Total del nodo coordinador =			83,00
Costo total del prototipo de geo-localización =			246,00

Realizado por: AVILA, Doris, 2019

El prototipo implementado tiene un costo total de \$246,00 dólares como se muestra en la tabla 4-3, el cual indica que el nodo de adquisición es mucho mayor su costo de implementación.

CONCLUSIONES

- Se implementó un prototipo electrónico autosustentable de geo-localización, con un estándar de comunicación inalámbrico Zigbee. Su estructura está compuesta por dos nodos: el de supervisión y coordinación que permite la ubicación de mascotas mediante una red de sensores.
- Se determinó que la aplicación adecuada para la localización en esta investigación fue el de Google map, que permite visualizar las coordenadas de latitud y longitud; estableciendo que el prototipo es estable por la prueba de repetitividad dando tiene una variación de 0.2127% a 0.8587% en latitud y en longitud de 0.117% a 0.1085% dando como resultado que el dispositivo es estable de coeficiente de variación tomadas de la muestra.
- Luego de analizar la topología y estructura de red implementada en esta investigación se concluye que la topología estrella es la más óptima por el número de dispositivos finales, y si un nodo falla no se pierde la información porque busca el nodo más próximo.
- La prueba de alcance para la comunicación RF en lugares indoor es de 100 metros con línea de vista de los análisis obtenidos se establece que la comunicación es excelente de 1 a 50 metros, 51 a 70 alta, 71 a 90 media, 91 a 100 baja y de 101 a 150 no operacional.
- El mensaje de alerta enviado mediante sms, alerta al dueño que la mascota se encuentra fuera del área delimitado. Teniendo una respuesta oportuna y la toma de decisión más adecuada.
- El dispositivo se encuentra colocado en un chaleco en la parte superior de la mascota, protegiendo así la integridad de los dispositivos electrónicos y manteniendo su correcto funcionamiento.
- Por las características tanto del microcontrolador nodeMCU, módulo Zigbee, la plataforma arduino y la programación IDE, fueron los requisitos más adecuados para el diseño e implementación del prototipo.
- El prototipo implementado dispone de una alimentación autónoma de 11,29 horas continuas de funcionamiento sin el uso del panel solar, pero se volverá a cargar automáticamente cuando la batería lo requiera con la ayuda del panel.

RECOMENDACIONES

- Para el caso de la comunicación inalámbrica Zigbee, se recomienda implementar otra tecnología como GPRS, microondas, entre otras; donde se puede tener mayor ganancia y alcance
- Para mejorar futuras implementaciones adaptar otro tipo de sensores que permitan la medición de diferentes variables como el caso del pulso, temperatura, entre otros de la mascota.
- Mejorar a futuro el material de construcción para que sea adaptable a cualquier tipo de clima, situación geografía o variaciones de tiempo.
- Para un monitoreo de la ubicación de la mascota se deberá usar una aplicación para dispositivos móvil que señale la trayectoria que realiza la mascota y la ubicación donde se encuentra en caso de extravío.

BIBLIOGRAFÍA

ANE MARIÑELARENA, ZigBee vs Bluetooth vs wifi.docx.[en línea]. 2013. [Consulta: 25 enero 2019]. Disponible en: <https://es.scribd.com/document/146071105/ZigBee-vs-Bluetooth-vs-wifi-docx>.

ANIMALEAR, ¿Qué es una mascota? - Animalear.com. [en línea]. 2018. [Consulta: 21 noviembre 2018]. Disponible en: <https://animalear.com/b/que-es-una-mascota-2>.

AVMA, Microchipping of Animals. [en línea]. 2019. [Consulta: 20 marzo 2019]. Disponible en: <https://www.avma.org/KB/Resources/Reference/Pages/Microchipping-of-Animals-Backgrounder.aspx>.

BECKER, 4 Datos Impactantes Sobre las Mascotas Perdidas. [en línea]. 2017. [Consulta: 10 octubre 2018]. Disponible en: <https://mascotas.mercola.com/sitios/mascotas/archivo/2017/04/17/semana-de-etiquetas-de-identificacion-animal.aspx>.

CASTELLANO ALVARO RICHI, Bluetooth. Introducción a su Funcionamiento. - [en línea]. 2012. [Consulta: 23 enero 2019]. Disponible en: <https://docplayer.es/47057681-Bluetooth-introduccion-a-su-funcionamiento.html>.

CASTELLANOS GABRIELA, La identificación de mascotas previene las pérdidas. [en línea]. [Consulta: 10 octubre 2018]. Disponible en: <https://www.elcomercio.com/narices-frias/beneficios-identificacion-mascotas-ecuador-perdidas.html>.

CÉSAR, C.I. y SÁNCHEZ, A., Presentado por. [en línea]. S.l.: 2018. [Consulta: 5 febrero 2019]. Disponible en: <http://acis.org.co/archivos/Conferencias/2018/Conferencia2202.pdf>.

CHIP247, Identificación electrónica en animales - Chip247.com - Productos de Identificación electrónica para animales. [en línea]. [sin fecha]. [Consulta: 9 enero 2019]. Disponible en: https://www.chip247.com/identificacion_electronica.php.

COLMENARES, P., RED ANIMALIA. [en línea]. 2013. [Consulta: 10 octubre 2018]. Disponible en: <https://redanimalia.com/2013/04/04/perros-perdidos/>.

CORREDOR, Ó.F., PEDRAZA, L.F. y HERNÁNDEZ, C.A., Bluetooth Technology: Alternative To Cellular Networks of Voice and Data. *Revista Visión Electronica*, 2009. p. 73-84.

DÍAZ, M., ¿QUÉ ES UNA MASCOTA? OBJETOS Y MIEMBROS DE LA FAMILIA WHAT IS A PET? OBJECTS AND FAMILY MEMBERS. *Revista Ajayu de Psicología* [en línea], 2017. p. 69. [Consulta: 22 noviembre 2018]. ISSN 2077-2161. Disponible en: http://www.scielo.org.bo/pdf/rap/v15n1/v15n1_a04.pdf.

DIGI INTERNATIONAL INC., XCTU Documentation. [en línea], 2015. [Consulta: 19 febrero 2019]. Disponible en: ftp://ftpl.digi.com/support/documentation/90001458_B.pdf.

EL TELÉGRAFO, Ley LOBA abarca 5 áreas del bienestar animal (Infografía). [en línea]. 2014. [Consulta: 22 noviembre 2018]. Disponible en: <https://www.eltelegrafo.com.ec/noticias/sociedad/1/ley-loba-abarca-5-areas-del-bienestar-animal-infografia>.

ELECTRONILAB, NodeMCU – Board de desarrollo con módulo ESP8266 WiFi y Lua Electronilab. [en línea]. 2018. [Consulta: 6 febrero 2019]. Disponible en: <https://electronilab.co/tienda/nodemcu-board-de-desarrollo-con-esp8266-wifi-y-lua/>.

INE-CHILE, ANEXO ESTADÍSTICO 1 : COEFICIENTES DE VARIACIÓN Y ERROR ASOCIADO AL ESTIMADOR ENCUESTA NACIONAL DE EMPLEO (ENE) INSTITUTO NACIONAL DE ESTADÍSTICAS (INE) [en línea]. S.l.: 2016. [Consulta: 2 abril 2019]. Disponible en: http://historico.ine.cl/ene/files/pdf/anexo_estadistico_coeficiente_de_variacion_y_error_asociado_1_estimador_v20160429.pdf.

LA HORA, El 90% de los perros callejeros tenía dueño. [en línea]. 2013. [Consulta: 10 octubre 2018]. Disponible en: <https://lahora.com.ec/noticia/1101496739/noticia>.

LUIS POSADAS-YAGÜE JOSE LUIS POZA LUJÁN, J., Red de sensores-Internet de las cosas. [en línea]. S.l.: 2014. [Consulta: 6 febrero 2019]. Disponible en:

<https://riunet.upv.es/bitstream/handle/10251/55288/GONZÁLEZ - Red de Sensores y el Internet de las cosas.pdf>.

MACHICAO MARCELO, C., CARACTERÍSTICAS DE LAS PLACAS ARDUINO. [en línea]. S.l.: 2017. [Consulta: 30 enero 2019]. Disponible en: <http://www.uajms.edu.bo/revistas/wp-content/uploads/2017/12/Art1-bit@bitdic2017.pdf>.

MACÍAS VILLALBA ÁLVARO, Aplicación de una red inalámbrica de sensores para detección de presencia y conteo. [en línea]. Sevilla : 2015. [Consulta: 9 enero 2019]. Disponible en: <http://bibing.us.es/proyectos/abreproy/90408/fichero/Aplicación+de+una+red+inalámbrica+de+sensores+para+detección+de+presencia+y+conteo.Final.pdf+>.

MAIONE, C., TURRA, C., FERNANDES, E.A.D.N., BACCHI, M.A., BARBOSA, F. y BARBOSA, R.M., Finding the Most Significant Elements for the Classification of Organic Orange Leaves: A Data Mining Approach. *Analytical Letters*, 2017. ISSN 1532236X. DOI 10.1080/00032719.2017.1286667.

MCI ELECTRONICS, XBee ZB S2C TH (con Antena) | MCI Electronics. [en línea]. 2018. [Consulta: 18 febrero 2019]. Disponible en: <https://www.mcielectronics.cl/shop/product/xbec-zb-s2c-th-wire-antenna-21554>.

NORBER BARRAZA, Topología y tipología. [en línea]. 2013. [Consulta: 17 enero 2019]. Disponible en: <https://es.slideshare.net/norberbarraza/topologia-y-tipologia>.

ORTIZ TAPIA, F., REDES DE SENSORES INALÁMBRICOS WIRELESS SENSOR NETWORKS (WSN). [en línea]. S.l.: [Consulta: 10 enero 2019]. Disponible en: http://profesores.elo.utfsm.cl/~tarredondo/info/networks/Presentacion_sensores.pdf.

PETFY, 5 tipos de identificación y rastreo para mascotas. Desde Dispositivos - Tienda Petfy. [en línea]. 2017. [Consulta: 9 enero 2019]. Disponible en: <https://tiendapetfy.cl/blogs/aprendiendo-con-milo/6-tipos-de-identificacion-y-rastreo-para-mascotas-desde-dispositivos-gps-identifican-qr-bluetooth-microchip-subcutaneo-y-otros>.

PRIETO BLÁZQUEZ, J., Introducción a los sistemas de comunicación inalámbricos. [en línea]. [Consulta: 22 enero 2019]. Disponible en: https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_

oviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_(Modulo_1).pdf.

PROTECCIÓN ANIMAL ECUADOR, Legislación Vigente - PAE. [en línea]. [Consulta: 22 noviembre 2018]. Disponible en: <http://www.pae.ec/derecho-animal/legislacion-vigente/>.

PROTECCIÓN ANIMAL MUNDIAL, La importancia de contar con una placa de identificación para tu mascota | World Animal Protection Costa Rica. [en línea]. 2017. [Consulta: 29 noviembre 2018]. Disponible en: <https://www.worldanimalprotection.cr/noticias/la-importancia-de-contar-con-una-placa-de-identificacion-para-tu-mascota>.

REAL ACADEMIA ESPAÑOLA, mascota | Definición de mascota - Diccionario de la lengua española - Edición del Tricentenario. [en línea]. 2017. [Consulta: 21 noviembre 2018]. Disponible en: <http://dle.rae.es/?id=OW11ptF>.

REGINO MEJÍA AURA MARÍA, Tecnologías de comunicación inalámbrica: Bluetooth, Wifi, in by Aura Maria Regino Mejía on Prezi. [en línea]. 2014. [Consulta: 20 enero 2019]. Disponible en: <https://prezi.com/i7w7kwq6vgiy/tecnologias-de-comunicacion-inalambrica-bluetooth-wifi-in/>.

RODAS VÁSQUEZ, U., Tesis para optar el Título de Ingeniero Electrónico, que presenta el bachiller. [en línea]. Perú: 2013. [Consulta: 15 enero 2019]. Disponible en: http://tesis.pucp.edu.pe/repositorio/bitstream/handle/123456789/5425/RODAS_URPI_DISEÑO_RED_INALAMBRICA_SENSORES_EMPRESA_PERU.pdf?sequence=1&isAllowed=y.

SALAZAR, J., REDES INALÁMBRICAS. [en línea]. S.l.: 2016. [Consulta: 25 enero 2019]. Disponible en: <http://www.techpedia.eu>.

SALGADO IGNACIO VIDRI, ZigBee y sus aplicaciones -[en línea]. 2012. [Consulta: 22 enero 2019]. Disponible en: <https://docplayer.es/10724850-Zigbee-y-sus-aplicaciones.html>.

TAGG Y WONDERWOOF, I.C., Ahora es más fácil monitorear las mascotas desde un dispositivo | SOLLA. [en línea]. 2015. [Consulta: 10 octubre 2018]. Disponible en: <https://www.solla.com/content/ahora-es-m-s-f-cil-monitorear-las-mascotas-desde-un-dispositivo>.

TROVAN, Microchips para animales :: Trovan. [en línea]. [Consulta: 7 enero 2019]. Disponible en:

<https://www.trovan.com/sp/lineas-de-productos/trovanunique-animales/microchips-para-animales#tab-8>.

YEPEZ DESIREE, Los animalistas están inconformes con posibles cambios a ley LOBA | El Comercio. [en línea]. 2015. [Consulta: 22 noviembre 2018]. Disponible en: <https://www.elcomercio.com/tendencias/loba-ley-cambios-animales-asamblea.html>.

ANEXOS

Anexo A: Hojas técnicas de los dispositivos del prototipo de geo-localización



Product Overview

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the ATmega8U2 programmed as a USB-to-serial converter.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

Index

Technical Specifications	Page 2
How to use Arduino Programming Environment, Basic Tutorials	Page 6
Terms & Conditions	Page 7
Environmental Policies half sqm of green via Impatto Zero®	Page 7



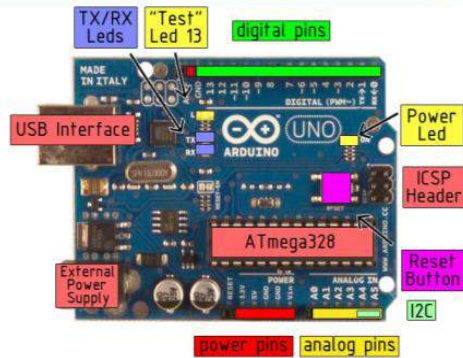
Technical Specification

EAGLE files: [arduino-ultimate-uno-arduino.gp](#) Schematic: [arduino-uno-schematic.pdf](#)

Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

the board



Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- 5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- 3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- GND.** Ground pins.

Memory

The ATmega328 has 32 KB of flash memory for storing code (of which 0.5 KB is used for the bootloader); it has also 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.



The Uno has 6 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the [analogReference\(\)](#) function. Additionally, some pins have specialized functionality:

- IC: 4 (SDA) and 5 (SCL).** Support I²C (TWI) communication using the [Wire library](#).

There are a couple of other pins on the board:

- AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#)
- Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and ATmega328 ports](#).

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega8U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '8U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an *.inf file is required.

The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Uno's digital pins.

The ATmega328 also support I²C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I²C bus; see the [documentation](#) for details. To use the SPI communication, please see the ATmega328 datasheet.

Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno w/ ATmega328" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega328 on the Arduino Uno comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [IC header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega8U2 firmware source code is available. The ATmega8U2 is loaded with a DFU bootloader, which can be activated by connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2. You can then use [Atmega's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader).

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega523 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

USB Overcurrent Protection

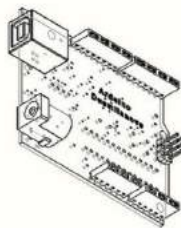
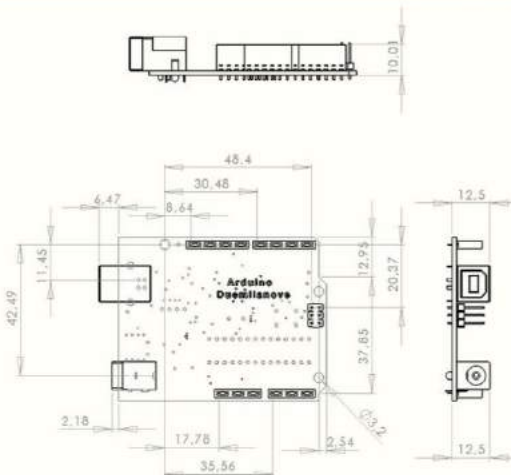
The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics

The maximum length and width of this Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.



Dimensioned Drawing



How to use Arduino



Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the [Arduino programming language](#) (based on [Wiring](#)) and the Arduino development environment (based on [Processing](#)). Arduino projects can be stand-alone or they can communicate with software on running on a computer (e.g. [Flash](#), [Processing](#), [MaxMSP](#)).

Arduino is a cross-platform program. You'll have to follow different instructions for your personal OS. Check on the [Arduino site](#) for the latest instructions. <http://arduino.cc/en/Guide/HomePage>

Linux Install

Windows Install

Mac Install

Once you have downloaded/unzipped the arduino IDE, you can Plug the Arduino to your PC via USB cable.

Blink led

Now you're actually ready to 'burn' your first program on the arduino board. To select "blink led", the physical translation of the well known programming 'hello world', select

File>Sketchbook> Arduino-0017>Examples> Digital>Blink.

Once you have your sketch you'll see something very close to the screenshot on the right.

In Tools>Board select

Now you have to go to Tools>SerialPort and select the right serial port, the one arduino is attached to.



Terms & Conditions



- Warranty**
 - The producer warrants that its products will conform to the Specifications. This warranty lasts for one (1) years from the date of the sale. The producer shall not be liable for any defects that are caused by neglect, misuse or mismanagement by the Customer, including improper installation or handling, or for any products that have been altered or modified in any way by a Customer. Moreover, the producer shall not be liable for any defects that result from Customer's design, specifications or instructions for such products. Testing and other quality control techniques are used to the extent the producer deems necessary.
 - If any products fail to conform to the warranty set forth above, the producer's sole liability shall be to replace such products. The producer's liability shall be limited to products that are returned by the producer within the warranty period. If the producer wishes to replace such products, the producer shall have a reasonable time to replacements. Replaced products shall be warranted for a new full warranty period.
 - EXCEPT AS SET FORTH ABOVE, PRODUCTS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS." THE PRODUCER DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR PARTICULAR PURPOSES.
 - Customer agrees that prior to using any systems that include the producer's products, Customer will test such systems and the functionality of the products as used in such systems. The producer may provide technical, application or design advice, quality characteristics, reliability data or other services. Customer acknowledges and agrees that providing these services shall not support or otherwise alter the producer's warranties, as set forth above, and no additional obligations or liabilities shall arise from the producer providing such services.
 - The Arduino™ products are not authorized for use in safety-critical applications where a failure of the product would reasonably be expected to cause serious personal injury or death. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Arduino™ products are neither designed nor intended for use in military or aerospace applications or environments and for automotive applications or environments. Customer acknowledges and agrees that any such use of Arduino™ products which is solely at the Customer's risk, and that Customer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.
 - Customer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products and any use of Arduino™ products in Customer's applications, notwithstanding any applications-related information or support that may be provided by the producer.
- Indemnification**

The Customer acknowledges and agrees to defend, indemnify and hold harmless the producer from and against any and all third-party losses, damages, liabilities and expenses it incurs in the event directly caused by: (a) an actual breach by a Customer of the representation and warranties made under the terms and conditions or (b) the gross negligence or willful misconduct by the Customer.
- Consequential Damages Waiver**

In no event the producer shall be liable to the Customer or any third parties for any special, collateral, indirect, punitive, incidental, consequential or exemplary damages in connection with or arising out of the products provided hereunder, regardless of whether the producer has been advised of the possibility of such damages. This section will survive the termination of the warranty period.
- Changes to specifications**

The producer may make changes to specifications and product descriptions at any time, without notice. The Customer must rely on the absence or characteristics of any features or instructions marked "reserved" or "unreleased." The producer reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the web site or Materials is subject to change without notice. Do not treat a design with this information.



Environmental Policies



The producer of Arduino™ has joined the Impatto Zero® policy of LifeGate.it. For each Arduino board produced is created / looked after half squared km of Costa Rica's forests.



NEO-6 series

Versatile u-blox 6 GPS modules

Highlights

- UART, USB, DDC (PC compliant) and SPI interfaces
- Available in Crystal and TCXO versions
- Onboard RTC crystal for faster warm and hot starts
- 1.8 V and 3.0 V variants



NEO-6
12.2 x 16.0 x 2.4 mm

Features

- u-blox 6 position engine
 - Navigate down to -162 dBm and -168 dBm (coldstart)
 - Faster acquisition with AssistNow Autonomous
 - Configurable power management
 - Hybrid (GPS/GNSS) engine (WAAS, EGNOS, MSAS)
 - Anti-jamming technology
- Simple integration with u-blox wireless modules
- A-GPS: AssistNow Online and AssistNow Offline services, OMA SUR compliant
- Backward compatible (hardware and firmware) with migration from NEO-5 family or NEO-4S
- LCC package for reliable and cost-effective manufacturing
- Compatible with u-blox GPS Solution for Android
- Based on GPS chips qualified according to AEC-Q100
- Manufactured in ISO/TS 16949 certified production sites
- Qualified according to ISO 16750

Product description

The NEO-6 module series brings the high performance of the u-blox 6 position engine to the miniature NEO form factor. u-blox has been designed with low power consumption and low costs in mind. Intelligent power management is a breakthrough for low-power applications. These features combine a high level of integration capability with flexible connectivity options in a small package. This makes them perfectly suited for mass-market end products with strict size and cost requirements. The DDC interface provides connectivity and enables synergies with u-blox LCC and USA wireless modules. All NEO-6 modules are manufactured in ISO/TS 16949 certified sites. Each module is tested and inspected during production. The modules are qualified according to ISO 16750 - Environmental conditions and electrical testing for electrical and electronic equipment for road vehicles.

Product selector

Model	Type	Supply	Interfaces	Features
	<ul style="list-style-type: none"> • Standalone GPS • Standalone (G/GNSS) • Timing & Real Data • Dual-frequency 	<ul style="list-style-type: none"> • 1.8V-2.0V • 2.7V-3.0V 	<ul style="list-style-type: none"> • UART • USB • SPI • DDC (PC compliant) 	<ul style="list-style-type: none"> • Programmable (through I2C or SPI) • Dual-band • RTC crystal • Antenna supply and regulator • Configurable pins • Temperature • External interrupt / Voltage
NEO-6G	•	•	•	•
NEO-6M	•	•	•	•
NEO-6Q	•	•	•	•

◦ - requires external components and integration on application processor

C = Crystal / T = TCXO

GPS locate, communicate, accelerate

Receiver performance data

Receiver type	50-channel u-blox 6 engine GPS L1 C/A code SBAS: WAAS, EGNOS, MSAS	
Navigation update rate	up to 5 Hz	
Accuracy ¹	Position	2.5 m CEP 2.0 m CEP
Acquisition ²	NEO-6M/6Q NEO-6M	
	Cold start	75 s / 77 s
	Acq. start ³	1 s / < 1 s
	Hot start	1 s / 1 s
Sensitivity ⁴	NEO-6M/6Q NEO-6M	
	Tracking	-162 dBm / -167 dBm
	Cold start	-168 dBm / -167 dBm
	Hot start	-157 dBm / -156 dBm

¹ u-blox 6 - 16 dBm
² Standard acquisition time, connection speed and lock time
³ Determined with a good antenna system

Electrical data

Power supply	2.7 V - 3.0 V (NEO-6M/6Q) 1.75 V - 2.0 V (NEO-6G)
Power consumption	111 mW @ 8 Hz (continuous) 33 mW @ 3.0 V Power Save Mode (3 Hz) 65 mW @ 3.0 V (coldstart) 22 mW @ 1.8 V Power Save Mode (1 Hz)
Backup power	1.4 V - 3.0 V, 22 µA
Supported antennas	Active and passive

Interfaces

Serial interfaces	1 UART 1 USB V2.0 (full speed) 12 Mbps 1 DDC (PC compliant) 1 SPI
Digital I/O	Configurable I2C/I2C 1 EXTINT input for wake-up
Serial and I/O	Voltages: 2.7 - 3.0 V (NEO-6M/6Q/6S) 1.8 - 2.0 V (NEO-6G)
Timeout	Configurable: 0.25 Hz to 1 kHz
Protocols	NMEA, USB binary, RTCM

Head Notes
u-blox reserves the right to change specifications without notice. Models, prices, logos and other information may be subject to change without notice. © 2012 u-blox AG. All rights reserved. u-blox is a registered trademark of u-blox AG. All other trademarks are the property of their respective owners.
The information contained herein is provided "as is". u-blox and its affiliates assume no liability for errors or omissions in the information contained herein. u-blox is not responsible for any damage or loss of data that may result from the use of the information contained herein. For more information, please contact your u-blox distributor or u-blox AG.

Package

24 pin LCC (Leadless Chip Carrier), 12.2 x 16.0 x 2.4 mm, 1.6 g



Environmental data, quality & reliability

Operating temp: -40°C to 85°C
Storage temp: -40°C to 85°C

RoHS compliant (lead free)

Qualified according to ISO 16750

Manufactured in ISO/TS 16949 certified production sites

Support products

u-blox 6 Evaluation Kit:
Easy-to-use kit to get familiar with u-blox 6 positioning technology, evaluate functionality, and measure GPS performance.
IUC Kit: u-blox 6 Evaluation Kit with TCXO, suitable for NEO-6S, NEO-6Q
DUC Kit: u-blox 6 Evaluation Kit with crystal, suitable for NEO-6M

Ordering information

NEO-6G: u-blox 6 GPS Module, 1.8V, TCXO, 12x16mm, 250 package
NEO-6M: u-blox 6 GPS Module, 1.8V/3.0V, 250 package
NEO-6Q: u-blox 6 GPS Module, TCXO, 12x16mm, 250 package
Available as sample and tape on reel (250 pieces)

Contact us

HD Switzerland
+41 44 722 7444
info@u-blox.com

China
+86 10 88 188 345
info_cn@u-blox.com

EMEA
+41 44 722 7444
info_emea@u-blox.com

Japan
+81 3 5779 8950
info_jp@u-blox.com

Americas
+1 703 483 5100
info_us@u-blox.com

APAC - Singapore
+65 6734 2011
info_ap@u-blox.com

Taiwan
+886 2 2037 1020
info_tw@u-blox.com

1.4 Block diagram

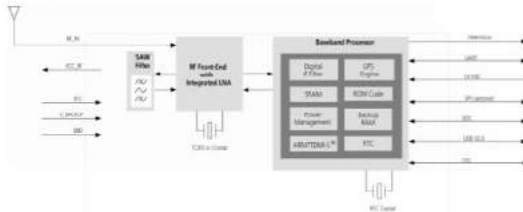


Figure 1: Block diagram (For available options refer to the product features table in section 1.2).

1.5 Assisted GPS (A-GPS)

Supply of aiding information like ephemeris, almanac, rough last position and time and satellite status and an optional time synchronization signal will reduce time to first fix significantly and improve the acquisition sensitivity. All NEO-6 modules support the u-blox AssistNow Online and AssistNow Offline A-GPS services* and are OMA SUR compliant.

1.6 SuperSense Indoor GPS

All NEO-6 modules come with SuperSense, providing improved acquisition/reacquisition and tracking sensitivity. SuperSense enables high performance tracking and navigation even in difficult signal environments such as urban canyons or indoor locations.

1.7 KickStart / Oscillators

An available feature is Kick-Start. This functionality uses a TCXO to accelerate weak signal acquisition, enabling faster start and reacquisition times. KickStart is available with the NEO-6Q and NEO-6S.

1.8 Protocols and interfaces

Protocol	Type
NMEA	Input/output, ASCII, 0183, 2,3 (compatible to 3.0)
UBX	Input/output, binary, u-blox proprietary

Table 3: Available protocols

Both protocols are available on UART, USB, and DDC. For specification of the various protocols see the u-blox 6 Receiver Description including Protocol Specification [2].

NEO-6 modules support a number of peripheral interfaces for serial communication. The embedded firmware uses these interfaces according to their respective protocol specifications. For specific applications, the firmware also supports the connection of external memories.



DIGI XBEE® S2C DIGIMESH® 2.4

Embedded RF modules deliver wireless connectivity using the innovative and easy-to-deploy Digimesh protocol

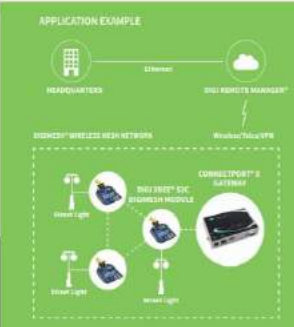
Digi XBee Digimesh 2.4 embedded RF modules provide wireless connectivity to electronic devices using a globally deployable 2.4 GHz transceiver. These modules use the Digimesh networking protocol. This innovative, peer-to-peer mesh network offers users added network stability through self-healing, dense network operation, extending the operational life of battery dependent networks.

Digi XBee modules are ideal for low-power applications. Digi XBee PRO modules are power-amplified versions of Digi XBee modules for extended-range applications. Products in the Digi XBee family are easy to use, share a common hardware footprint, and are fully interoperable with other

Digi XBee products utilizing the same technology. They are available in a variety of different protocols to suit different applications, enabling users to substitute one Digi XBee module for another with minimal development time and risk.

The updated Digi XBee S2C Digimesh module is built with the SiliconLabs EHS37 SoC and offers improved power consumption, support for over-the-air firmware updates, and provides an upgrade path to IEEE 802.15.4 or ZigBee® mesh protocols if desired.

- BENEFITS**
- No configuration needed for out-of-the-box RF communications
 - Common Digi XBee footprint for a variety of RF modules
 - Simple topology to deploy and grow
 - 2.4 GHz for worldwide deployment
 - Low-power Digi XBee or extended-range XBee-PRO (pin-over-the-air compatible)
 - Multiple antenna options
 - Outdoor RF line-of-sight range up to 2 miles (3.2 km)* with high-gain antennas
 - DigiMesh peer-to-peer mesh networking protocol
 - Supports self-healing and discovery for network stability



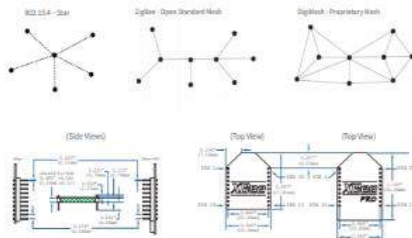
SPECIFICATIONS	Digi XBee® S2C Digimesh® 2.4	Digi XBee-PRO® S2C Digimesh® 2.4
PERFORMANCE		
TRANSMITTER CHIPSET	SiliconLabs EHS37 SoC	SiliconLabs EHS37 SoC
DATA RATE	RF 250 kbps, Serial up to 1 Mbps	RF 250 kbps, Serial up to 1 Mbps
INDOOR/OUTDOOR RANGE*	Up to 200 ft (60 m)	Up to 2000 ft (600 m)
OUTDOOR RF LINE-OF-SIGHT RANGE**	Up to 1000 ft (300 m)	Up to 2 miles (3200 m)
TRANSMIT POWER	3.3 mW (4.0 dBm) / 6.3 mW (8.0 dBm) boost mode	42 mW (16.0 dBm)
RECEIVER SENSITIVITY (typical)	-100 dBm / -102 dBm boost mode	-102 dBm
FEATURES		
SERIAL DATA INTERFACE	UART, SPI	UART, SPI
CONFIGURATION METHOD	API or AT commands, local or over-the-air (OTA)	API or AT commands, local or over-the-air (OTA)
FREQUENCY BAND	ISM 2.4 GHz	ISM 2.4 GHz
FORM FACTOR	Through-hole, Surface Mount	Through-hole, Surface Mount
HARDWARE	S2C	S2C
ADC INPUTS	1x 10-bit ADC Inputs	1x 10-bit ADC Inputs
DIGITAL I/O	1x	1x
ANTENNA OPTIONS	Through-hole PCB Antenna, U.FL Connector, RF24M Connector, or Integrated Mini-DPI-16 Pad, PCB Antenna, or U.FL Connector	Through-hole PCB Antenna, U.FL Connector, RF24M Connector, or Integrated Mini-DPI-16 Pad, PCB Antenna, or U.FL Connector
OPERATING TEMPERATURE	-40°C to +107°C	-40°C to +107°C
DIMENSIONS & WEIGHT	Through-hole: 0.960" x 1.000" (24.14 x 25.40 mm) SMT: 0.960" x 1.000" (24.14 x 25.40 mm)	Through-hole: 0.960" x 1.250" (24.14 x 31.75 mm) SMT: 0.960" x 1.250" (24.14 x 31.75 mm)
NETWORKING AND SECURITY		
PROTOCOL	Digi XBee Digimesh 2.4 (proprietary IEEE 802.15.4 based mesh protocol)	Digi XBee Digimesh 2.4 (proprietary IEEE 802.15.4 based mesh protocol)
UPDATABLE TO IEEE 802.15.4 PROTOCOL	Yes	Yes
UPDATABLE TO ZIGBEE PROTOCOL	Yes	Yes
INTERFERENCE IMMUNITY	DFS (Direct Sequence Spread Spectrum)	DFS (Direct Sequence Spread Spectrum)
CRYPTO KEY	128-bit AES	128-bit AES
RELIABLE PACKET DELIVERY	Reliable Acknowledgments	Reliable Acknowledgments
IDS	Pin ID and address checker (in and outgoing) (optional)	Pin ID and address checker (in and outgoing) (optional)
CHANNELS	16 channels	16 channels
POWER REQUIREMENTS		
SUPPLY VOLTAGE	2.2 to 3.0V	2.7 to 3.0V
TRANSMIT CURRENT	15 mA @ 3.3VDC / 45 mA boost mode	120 mA @ 3.3VDC
RECEIVE CURRENT	20 mA @ 3.3VDC / 35 mA boost mode	15 mA @ 3.3VDC
POWER-DOWN CURRENT	<1 µA @ 27°C	<1 µA @ 27°C
REGULATOR APPROVALS		
FCC, IC (NORTH AMERICA)	Yes	Yes
ETSI (EUROPE)	Yes	Yes
SRM (AUSTRALIA AND NEW ZEALAND)	Yes	Yes
TELEC (JAPAN)	Yes	No (Coming soon)

*Range figures estimate are based on free air format with limited sources of interference. Actual range will vary based on transmitting power, orientation of transmitter and receiver, height of receiving antenna, height of receiving antenna, weather conditions, interference sources in the area, and terrain between receiver and transmitter, including indoor and outdoor structures such as walls, trees, buildings, hills, and mountains.



PART NUMBERS	DESCRIPTION
SMT MODULES	
SD42MPS-00	Digi XBee, S2C Digimesh 2.4, SMT, PCB antenna
SDPACMPS-00	Digi XBee-PRO, S2C Digimesh 2.4, SMT, PCB antenna
SD42MUS-00	Digi XBee, S2C Digimesh 2.4, SMT, U.FL
SDPACMUS-00	Digi XBee-PRO, S2C Digimesh 2.4, SMT, U.FL
SD42MBS-00	Digi XBee, S2C Digimesh 2.4, SMT, RF Pad
SDPACMBS-00	Digi XBee-PRO, S2C Digimesh 2.4, SMT, RF Pad
THROUGH-HOLE MODULES	
SD42MPT-00	Digi XBee, S2C Digimesh 2.4, Th, PCB antenna
SDPACMPT-00	Digi XBee-PRO, S2C Digimesh 2.4, Th, PCB antenna
SD42MUT-00	Digi XBee, S2C Digimesh 2.4, Th, U.FL
SDPACMUT-00	Digi XBee-PRO, S2C Digimesh 2.4, Th, U.FL
SD42MST-00	Digi XBee, S2C Digimesh 2.4, Th, RF50M
SDPACMST-00	Digi XBee-PRO, S2C Digimesh 2.4, Th, RF50M
SD42MWT-00	Digi XBee, S2C Digimesh 2.4, Th, MRE antenna
SDPACMWT-00	Digi XBee-PRO, S2C Digimesh 2.4, Th, MRE antenna
KIT	
SK-WEM	Digi XBee S2C Digimesh Development Kit with two SD42MPT-00 modules and one SD42MPS-00 module

Note: International users select the standard module. The PRO module is for use in North America only.



DIGI SERVICE AND SUPPORT | You can purchase with confidence knowing that Digi is always available to serve you with expert technical support and our industry leading warranty. For detailed information visit www.digi.com/support.

© 2008-2018 Digi International Inc. All rights reserved.
All trademarks are the property of their respective owners.

5310462
14/15/18

DIGI INTERNATIONAL WORLDWIDE HQ
877-812-3444 / 912-812-3444 | www.digi.com

DIGI INTERNATIONAL GERMANY
+49-69-540-426-0

DIGI INTERNATIONAL JAPAN
+81-3-5428-5262 | www.digi-japan.jp

DIGI INTERNATIONAL SINGAPORE
+65-6213-5381

DIGI INTERNATIONAL CHINA
+86-22-59492290 | www.digi.com.cn



ESP8266EX Datasheet

Version 4.3

Espressif Systems IOT Team
<http://bbs.espressif.com/>
 Copyright © 2015

Disclaimer and Copyright Notice

Information in this document, including URL references, is subject to change without notice. THIS DOCUMENT IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE. All liability, including liability for infringement of any proprietary rights, relating to use of information in this document is disclaimed. No licenses express or implied, by estoppel or otherwise, to any intellectual property rights are granted herein.

The Wi-Fi Alliance Member Logo is a trademark of the WiFi Alliance.

All trade names, trademarks and registered trademarks mentioned in this document are property of their respective owners, and are hereby acknowledged.

Copyright © 2015 Espressif Systems. All rights reserved.

Table of Contents

1. General Overview	6
1.1. Introduction	6
1.2. Features	7
1.3. Parameters	7
1.4. Ultra Low Power Technology	9
1.5. Major Applications	9
2. Hardware Overview	11
2.1. Pin Definitions	11
2.2. Electrical Characteristics	13
2.3. Power Consumption	13
2.4. Receiver Sensitivity	14
2.5. MCU	15
2.6. Memory Organization	15
2.6.1. Internal SRAM and ROM	15
2.6.2. External SPI Flash	15
2.7. AHB and AHB Blocks	16
3. Pins and Definitions	17
3.1. GPIO	17
3.1.1. General Purpose Input/Output Interface (GPIO)	17

3.2. Secure Digital Input/Output Interface (SDIO)	18
3.3. Serial Peripheral Interface (SPI/HSPI)	18
3.3.1. General SPI (Master/Slave)	18
3.3.2. SDIO / SPI (Slave)	19
3.3.3. HSPI (Master/Slave)	19
3.4. Inter-integrated Circuit Interface (I2C)	19
3.5. I2S	20
3.6. Universal Asynchronous Receiver Transmitter (UART)	20
3.7. Pulse-Width Modulation (PWM)	21
3.8. IR Remote Control	22
3.9. ADC (Analog-to-digital Converter)	22
3.10. LED Light and Button	24
4. Firmware & Software Development Kit	26
4.1. Features	26
5. Power Management	27
6. Clock Management	28
6.1. High Frequency Clock	28
6.2. External Reference Requirements	29
7. Radio	29
7.1. Channel Frequencies	30
7.2. 2.4 GHz Receiver	30
7.3. 2.4 GHz Transmitter	30

1. General Overview

1.1. Introduction

Espressif Systems' Smart Connectivity Platform (ESCP) is a set of high performance, high integration wireless SOCs, designed for space and power constrained mobile platform designers. It provides unsurpassed ability to embed WiFi capabilities within other systems, or to function as a standalone application, with the lowest cost, and minimal space requirement.

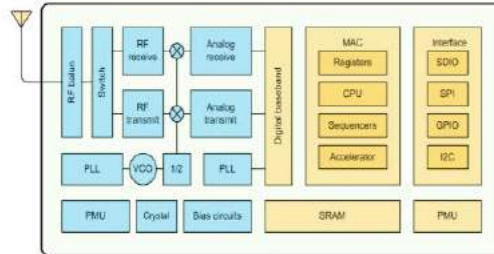


Figure 1 ESP8266EX Block Diagram

ESP8266EX offers a complete and self-contained WiFi networking solution; it can be used to host the application or to offload WiFi networking functions from another application processor.

When ESP8266EX hosts the application, it boots up directly from an external flash. It has integrated cache to improve the performance of the system in such applications.

Alternately, serving as a WiFi adaptor, wireless internet access can be added to any micro controller-based design with simple connectivity (SPI/SDIO or I2C/UART interface).

ESP8266EX is among the most integrated WiFi chip in the industry; it integrates the antenna switches, RF balun, power amplifier, low noise receive amplifier, filters, power management modules, it requires minimal external circuitry, and the entire solution, including front-end module, is designed to occupy minimal PCB area.

ESP8266EX also integrates an enhanced version of Tensilica's L106 Diamond series 32-bit processor, with on-chip SRAM, besides the WiFi functionalities. ESP8266EX is often integrated with external sensors and other application specific devices through its GPIOs; sample codes for such applications are provided in the software development kit (SDK).

Espressif Systems' Smart Connectivity Platform (ESCP) demonstrates sophisticated system-level features include fast sleep/wake context switching for energy-efficient VoIP, adaptive radio biasing for low-power operation, advance signal processing, and spur cancellation and radio co-existence features for common cellular, Bluetooth, DDR, LVDS, LCD interference mitigation.

1.2. Features

- 802.11 b/g/n
- Integrated low power 32-bit MCU
- Integrated 10-bit ADC
- Integrated TCP/IP protocol stack
- Integrated TR switch, balun, LNA, power amplifier and matching network
- Integrated PLL, regulators, and power management units
- Supports antenna diversity
- WiFi 2.4 GHz, support WPA/WPA2
- Support STA/AP/STA+AP operation modes
- Support Smart Link Function for both Android and iOS devices
- SDIO 2.0, (H) SPI, UART, I2C, I2S, IR Remote Control, PWM, GPIO
- STBC, 1x1 MIMO, 2x1 MIMO
- A-MPDU & A-MSDU aggregation & 0.4μs guard interval
- Deep sleep power <10uA, Power down leakage current < 5uA
- Wake up and transmit packets in < 2ms
- Standby power consumption of < 1.0mW (DTIM3)
- +20 dBm output power in 802.11b mode
- Operating temperature range -40C ~ 125C
- FCC, CE, TELEC, WiFi Alliance, and SRRC certified

1.3. Parameters

Table 1 Parameters

Categories	Items	Values
WiFi Parameters	Certificates	FCC/CE/TELEC/SRRC
	WiFi Protocols	802.11 b/g/n
	Frequency Range	2.4G-2.5G (2400M-2483.5M)
	Tx Power	802.11 b: +20 dBm
		802.11 g: +17 dBm 802.11 n: +14 dBm
	Rx Sensitivity	802.11 b: -91 dBm (11 Mbps)
		802.11 g: -75 dBm (54 Mbps) 802.11 n: -72 dBm (MCS7)
Types of Antenna	PCB Trace, External, IPEX Connector, Ceramic Chip	
Hardware Parameters	Peripheral Bus	UART/SDIO/SPI/I2C/I2S/IR Remote Control GPIO/PWM
	Operating Voltage	3.0-3.6V
	Operating Current	Average value: 80mA
	Operating Temperature Range	-40°-125°
	Ambient Temperature Range	Normal temperature
	Package Size	5x5mm
	External Interface	N/A
Software Parameters	WiFi mode	station/softAP/SoftAP+station
	Security	WPA/WPA2
	Encryption	WEP/TKIP/AES
	Firmware Upgrade	UART Download / OTA (via network)
	Software Development	Supports Cloud Server Development / SDK for custom firmware development
Network Protocols	IPv4, TCP/UDP/HTTP/FTP	

User Configuration	AT Instruction Set, Cloud Server, Android/iOS App
--------------------	---

1.4. Ultra Low Power Technology

ESP8266EX has been designed for mobile, wearable electronics and Internet of Things applications with the aim of achieving the lowest power consumption with a combination of several proprietary techniques. The power saving architecture operates mainly in 3 modes: active mode, sleep mode and deep sleep mode.

By using advance power management techniques and logic to power-down functions not required and to control switching between sleep and active modes, ESP8266EX consumes about than 60uA in deep sleep mode (with RTC clock still running) and less than 1.0mA (DTIM=3) or less than 0.5mA (DTIM=10) to stay connected to the access point.

When in sleep mode, only the calibrated real-time clock and watchdog remains active. The real-time clock can be programmed to wake up the ESP8266EX at any required interval.

The ESP8266EX can be programmed to wake up when a specified condition is detected. This minimal wake-up time feature of the ESP8266EX can be utilized by mobile device SOCs, allowing them to remain in the low-power standby mode until WiFi is needed.

In order to satisfy the power demand of mobile and wearable electronics, ESP8266EX can be programmed to reduce the output power of the PA to fit various application profiles, by trading off range for power consumption.

1.5. Major Applications

Major fields of ESP8266EX applications to Internet-of-Things include:

- Home Appliances
- Home Automation
- Smart Plug and lights
- Mesh Network
- Industrial Wireless Control
- Baby Monitors
- IP Cameras
- Sensor Networks
- Wearable Electronics

2. Hardware Overview

2.1. Pin Definitions

The pin assignments for 32-pin QFN package is illustrated in Fig.2.

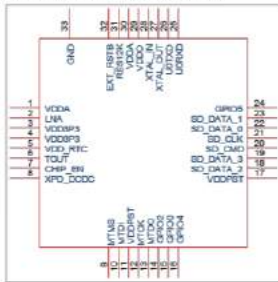


Figure 2 Pin Assignments

Table 2 below presents an overview on the general pin attributes and the functions of each pin.

Table 2 Pin Definitions

Pin	Name	Type	Function
1	VDDA	P	Analog Power 3.0~3.6V
2	LNA	I/O	RF Antenna Interface. Chip Output Impedance=50Ω No matching required but we recommend that the n-type matching network is retained.
3	VDD3P3	P	Amplifier Power 3.0~3.6V
4	VDD3P3	P	Amplifier Power 3.0~3.6V
5	VDD_RTC	P	NC (1.1V)

6	TOUT	I	ADC Pin (note: an internal pin of the chip) can be used to check the power voltage of VDD3P3 (Pin 3 and Pin4) or the input voltage of TOUT (Pin 6). These two functions cannot be used simultaneously.
7	CHIP_EN	I	Chip Enable. High: On, chip works properly; Low: Off, small current
8	XPD_DCDC	I/O	Deep Sleep Wakeup: GPIO1&
9	MTMS	I/O	GPIO14; HSPI_CLK
10	MTDI	I/O	GPIO12; HSPI_MISO
11	VDDPST	P	Digital/I/O Power Supply (1.8V~3.3V)
12	MTCK	I/O	GPIO13; HSPI_MOSI; UART0_CTS
13	MTDO	I/O	GPIO15; HSPI_CS; UART0_RTS
14	GPIO2	I/O	UART Tx during flash programming; GPIO2
15	GPIO0	I/O	GPIO0; SPI_CS2
16	GPIO4	I/O	GPIO4
17	VDDPST	P	Digital/I/O Power Supply (1.8V~3.3V)
18	SDIO_DATA_2	I/O	Connect to SD_D2 (Series R: 200Ω); SPIHD; HSPFD; GPIO9
19	SDIO_DATA_3	I/O	Connect to SD_D3 (Series R: 200Ω); SPIWP; HSPWP; GPIO10
20	SDIO_CMD	I/O	Connect to SD_CMD (Series R: 200Ω); SPI_CS0; GPIO11
21	SDIO_CLK	I/O	Connect to SD_CLK (Series R: 200Ω); SPI_CLK; GPIO6
22	SDIO_DATA_0	I/O	Connect to SD_D0 (Series R: 200Ω); SPI_MSI0; GPIO7
23	SDIO_DATA_1	I/O	Connect to SD_D1 (Series R: 200Ω); SPI_MOSI; GPIO8
24	GPIO5	I/O	GPIO5
25	U0RXD	I/O	UART Rx during flash programming; GPIO3
26	U0TXD	I/O	UART Tx during flash programming; GPIO1; SPI_CS1
27	XTAL_OUT	I/O	Connect to crystal oscillator output, can be used to provide BT clock input
28	XTAL_IN	I/O	Connect to crystal oscillator input
29	VDDD	P	Analog Power 3.0V~3.6V
30	VDDA	P	Analog Power 3.0V~3.6V
31	RES12K	I	Serial connection with a 12 kΩ resistor and connect to the ground
32	EXT_RSTB	I	External reset signal (Low voltage level: Active)

Note: GPIO2, GPIO0, MTDO can be configurable as 3-bit SDIO mode.

2.2. Electrical Characteristics

Table 3 ESP8266EX Electrical Characteristics

Parameters	Conditions	Min	Typical	Max	Unit
Storage Temperature Range		-40	Normal	125	°C
Maximum Soldering Temperature	IPC/JEDEC J-STD-020			260	°C
Working Voltage Value		3.0	3.3	3.6	V
I/O	V _I /V _{IN}	-0.3/0.75V _{IO}		0.25V _{IO} /3.8	V
	V _{OL} /V _{OH}	N/0.8V _{IO}		0.1V _{OH} /N	
	I _{max}			12	mA
Electrostatic Discharge (HBM)	TAMB=25°C			2	KV
Electrostatic Discharge (CDM)	TAMB=25°C			0.5	KV

2.3. Power Consumption

The following current consumption is based on 3.3V supply, and 25°C ambient, using internal regulators. Measurements are done at antenna port without SAW filter. All the transmitter's measurements are based on 90% duty cycle, continuous transmit mode.

Table 4 Description on Power Consumption

Parameters	Min	Typical	Max	Unit
Tx 802.11b, CCK 11Mbps, P_OUT=+17dBm		170		mA
Tx 802.11g, OFDM 54Mbps, P_OUT=+15dBm		140		mA
Tx 802.11n, MCS7, P_OUT=+13dBm		120		mA
Rx 802.11b, 1024 bytes packet length, -80dBm		50		mA
Rx 802.11g, 1024 bytes packet length, -70dBm		56		mA
Rx 802.11n, 1024 bytes packet length, -65dBm		56		mA
Modem Sleep①		15		mA
Light-Sleep②		0.9		mA
Deep-Sleep③		10		µA
Power Off		0.5		µA

- ①: Modem-Sleep requires the CPU to be working, as in PWM or I2S applications. According to 802.11 standards (like U-APSD), it saves power to shut down the WiFi Modem circuit while maintaining a WiFi connection with no data transmission. E.g. in DTIM3, to maintain a sleep 300ms-wake 3ms cycle to receive AP's Beacon packages, the current is about 15mA
- ②: During Light-Sleep, the CPU may be suspended in applications like WiFi switch. Without data transmission, the WiFi Modem circuit can be turned off and CPU suspended to save power according to the 802.11 standard (U-APSD). E.g. in DTIM3, to maintain a sleep 300ms-wake 3ms cycle to receive AP's Beacon packages, the current is about 0.9mA.
- ③: Deep Sleep does not require WiFi connection to be maintained. For application with long time lags between data transmission, e.g. a temperature sensor that checks the temperature every 100s, sleep 300s and waking up to connect to the AP (taking about 0.3~1s), the overall average current is less than 1mA.

2.4. Receiver Sensitivity

The following are measured under room temperature conditions with 3.3V and 1.1V power supplies.

Table 5 Receiver Sensitivity

Parameters	Min	Typical	Max	Unit
Input frequency	2412		2484	MHz
Input impedance		50		Ω
Input reflection			-10	dB
Output power of PA for 72.2Mbps	15.5	16.5	17.5	dBm
Output power of PA for 11b mode	19.5	20.5	21.5	dBm
Sensitivity				
DSSS, 1Mbps		-98		dBm
CCK, 11Mbps		-91		dBm
6Mbps (1/2 BPSK)		-93		dBm
54Mbps (3/4 64 QAM)		-75		dBm
HT20, MCS7 (65Mbps, 72.2Mbps)		-72		dBm
Adjacent Channel Rejection				
OFDM, 6Mbps		37		dB
OFDM, 54Mbps		21		dB
HT20, MCS0		37		dB
HT20, MCS7		20		dB

2.5. MCU

ESP8266EX is embedded with Tensilica L106 32-bit micro controller (MCU), which features extra low power consumption and 16-bit RSIC. The CPU clock speed is 80MHz. It can also reach a maximum value of 160MHz. Real Time Operation System (RTOS) is enabled. Currently, only 20% of MIPS has been occupied by the WiFi stack, the rest can all be used for user application programming and development. The following interfaces can be used to connect to the MCU embedded in ESP8266EX:

- Programmable RAM/ROM interfaces (iBus), which can be connected with memory controller, and can also be used to visit external flash;
- Data RAM interface (dBus), which can be connected with memory controller;
- AHB interface, can be used to visit the register.

2.6. Memory Organization

2.6.1. Internal SRAM and ROM

ESP8266EX WiFi SoC is embedded with memory controller, including SRAM and ROM. MCU can visit the memory units through iBus, dBus, and AHB interfaces. All memory units can be visited upon request, while a memory arbiter will decide the running sequence according to the time when these requests are received by the processor.

According to our current version of SDK provided, SRAM space that is available to users is assigned as below:

- **RAM size < 36kB**, that is to say, when ESP8266EX is working under the station mode and is connected to the router, programmable space accessible to user in heap and data section is around 36kB.)
- There is no programmable ROM in the SoC, therefore, user program must be stored in an external SPI flash.

2.6.2. External SPI Flash

An external SPI flash is used together with ESP8266EX to store user programs. Theoretically speaking, up to 16 Mbyte memory capacity can be supported.

Suggested SPI Flash memory capacity:

- OTA is disabled: the minimum flash memory that can be supported is 512 kbyte;
 - OTA is enabled: the minimum flash memory that can be supported is 1 Mbyte.
- Several SPI modes can be supported, including Standard SPI, Dual SPI, DIO SPI, QIO SPI, and Quad SPI.

3. Pins and Definitions

The chipset encapsulates variable analog and data transmission I/Os, descriptions and definitions of which are explained below in detail.

3.1. GPIO

3.1.1. General Purpose Input/Output Interface (GPIO)

There are up to 17 GPIO pins. They can be assigned to various functions by the firmware. Each GPIO can be configured with internal pull-up (except XPD_DDCDC, which is configured with internal pull-down), input available for sampling by a software register, input triggering an edge or level CPU interrupt, input triggering a level wakeup interrupt, open-drain or push-pull output driver, or output source from a software register, or a sigma-delta PWM DAC.

These pins are multiplexed with other functions such as I2C, I2S, UART, PWM, IR Remote Control, etc. Data I/O soldering pad is bidirectional and tri-state that include data input and output controlling buffer. Besides, I/O can be set as a specific state and remains like this. For example, if you intend to lower the power consumption of the chip, all data input and output enable signals can be set as remaining low power state. You can transport some specific state into the I/O. When the I/O is not powered by external circuits, the I/O will remain to be the state that it was used the last time. Some positive feedback is generated by the state-remaining function of the pins, therefore, if the external driving power must be stronger than the positive feedback. Even so, the driving power that is needed is within 5uA.

Table 6 Pin Definitions of GPIOs

Variables	Symbol	Min	Max	Unit
Input Low Voltage	V_{IL}	0.3	$0.25 \times V_{DD}$	V
Input High Voltage	V_{IH}	$0.75 \times V_{DD}$	3.3	V
Input Leakage Current	I_L		50	nA
Output Low Voltage	V_{OL}		$0.1 \times V_{DD}$	V
Output High Voltage	V_{OH}	$0.8 \times V_{DD}$		V
Input Pin Resistance Value	C_{pad}		2	pF
VDDIO	V_{DD}	1.8	3.3	V
Maximum Driving Power	I_{max}		12	mA
Temperature	T_{amb}	-40	125	°C

All digital I/O pins are protected from over-voltage with a snap-back circuit connected between the pad and ground. The snap back voltage is typically about 6V, and the holding voltage is 5.8V. This

provides protection from over-voltages and ESD. The output devices are also protected from reversed voltages with diodes.

3.2. Secure Digital Input/Output Interface (SDIO)

One Slave SDIO has been defined by ESP8266EX, the definitions of which are described in Table 7 below. 4bit 25MHz SDIO v1.1 and 4bit 50MHz SDIO v2.0 are supported.

Table 7 Pin Definitions of SDIOs

Pin Name	Pin Num	IO	Function Name
SDIO_CLK	21	IO6	SDIO_CLK
SDIO_DATA0	22	IO7	SDIO_DATA0
SDIO_DATA1	23	IO8	SDIO_DATA1
SDIO_DATA_2	18	IO9	SDIO_DATA_2
SDIO_DATA_3	19	IO10	SDIO_DATA_3
SDIO_CMD	20	IO11	SDIO_CMD

3.3. Serial Peripheral Interface (SPI/HSPI)

Currently, one general Slave/Master SPI, one Slave SDIO/SPI, and one general Slave/Master HSPI have been defined by ESP8266EX. Functions of all these pins can be implemented via hardware. The pin definitions are described below:

3.3.1. General SPI (Master/Slave)

Table 8 Pin Definitions of General SPIs

Pin Name	Pin Num	IO	Function Name
SDIO_CLK	21	IO6	SPICLK
SDIO_DATA0	22	IO7	SPIQ/MISO
SDIO_DATA1	23	IO8	SPIQ/MOSI
SDIO_DATA_2	18	IO9	SPIHD
SDIO_DATA_3	19	IO10	SPIWP
SDIO_CMD	20	IO11	SPICSS0
U0TXD	26	IO1	SPICSS1
GPIO0	15	IO0	SPICSS2

3.3.2. SDIO / SPI (Slave)

Table 9 Pin Definitions of SDIO / SPI (Slave)

Pin Name	Pin Num	IO	Function Name
SDIO_CLK	21	IO6	SPI_SLAVE_CLK
SDIO_DATA0	22	IO7	SPI_SLAVE_MISO
SDIO_DATA1	23	IO8	SPI_SLAVE_INT
SDIO_DATA_2	18	IO9	NC
SDIO_DATA_3	19	IO10	SPI_SLAVE_CS
SDIO_CMD	20	IO11	SPI_SLAVE_MOSI

3.3.3. HSPI (Master/Slave)

Table 10 Pin Definitions of HSPI (Master/Slave)

Pin Name	Pin Num	IO	Function Name
MTMS	9	IO14	HSPICLK
MTDI	10	IO12	HSPQ/MISO
MTCK	12	IO13	HSPID/MOSI
MTDO	13	IO15	HSPICSS

Note:

- SPI mode can be implemented via software programming. The clock frequency can reach up to a maximum value of 80MHz.
- Function of Slave SDIO/SPI interface can be implemented via hardware, and linked list DMA (Direct Memory Access) is supported, software overheads are smaller. However, there is no linked list DMA on general SPI and HSPI, and the software overheads are larger, therefore, the data transmitting speed will be restrained by software processing speed.

3.4. Inter-integrated Circuit Interface (I2C)

One I2C, which is mainly used to connect with micro controller and other peripheral equipment such as sensors, is defined by ESP8266EX. The present pin definition of I2C is as defined below:

Table 11 Pin Definitions of I2C

Pin Name	Pin Num	IO	Function Name
MTMS	9	IO14	I2C_SCL
GPIO2	14	IO2	I2C_SDA

Both I2C-Master and I2C-Slave are supported. I2C interface functionality can be realized via software programming, the clock frequency can be up to around 100KHz at most. It should be noted that I2C clock frequency should be higher than the slowest clock frequency of the slave device.

3.5. I2S

Currently one I2S data input interface and one I2S data output interface are defined. I2S interface is mainly used in applications such as data collection, processing, and transmission of audio data, as well as the input and output of serial data. For example, LED lights (WS2812 series) are supported. The pin definition of I2S is as defined below:

Table 12 Pin Definitions of I2S

I2S Data Input:			
Pin Name	Pin Num	IO	Function Name
MTDI	10	IO12	I2SI_DATA
MTCK	12	IO13	I2SI_BCK
MTMS	9	IO14	I2SI_WS
I2S Data Output:			
Pin Name	Pin Num	IO	Function Name
MTDO	13	IO15	I2SO_BCK
U0RXD	25	IO3	I2SO_DATA
GPIO2	14	IO2	I2SO_WS

I2S functionality can be realized via software programming, the GPIOs that will be used are multiplexed, and linked list DMA is supported.

3.6. Universal Asynchronous Receiver Transmitter (UART)

Two UART interfaces, UART0 and UART1, have been defined by ESP8266EX, the definitions are as below:

Table 13 Pin Definitions of UART Interfaces

Pin Type	Pin Name	Pin Num	IO	Function Name
UART0	U0RXD	25	IO3	U0RXD
	U0TXD	26	IO1	U0TXD
	MTDO	13	IO15	U0RTS
	MTCK	12	IO13	U0CTS
UART1	GPIO2	14	IO2	U1TXD
	SD_D1	23	IO8	U1RXD

Data transfers to/from UART interfaces can be implemented via hardware. The data transmission speed via UART interfaces can reach 115200*40 (4.5Mbps).

UART0 can be for communication. It supports fluid control. Since UART1 features only data transmit signal (Tx), it is usually used for printing log.

Notes: By default, UART0 will output some printed information when the device is powered on and is booting up. The baud rate of the printed information is closely related to the frequency of the external crystal oscillator. If the frequency of the crystal oscillator is 40MHz, then the baud rate for printing is 115200; if the frequency of the crystal oscillator is 26MHz, then the baud rate for printing is 74880. If the printed information exerts any influence on the functionality of your device, you'd better block the printing during the power-on period by changing (U0TXD, U0RXD) to (MTDO, MTCK).

3.7. Pulse-Width Modulation (PWM)

Four PWM output interfaces have been defined by ESP8266EX. They can be extended by users themselves. The present pin definitions of the PWM interfaces are defined as below:

Table 14 Pin Definitions of PWM Interfaces

Pin Name	Pin Num	IO	Function Name
MTDI	10	IO12	PWM0
MTDO	13	IO15	PWM1
MTMS	9	IO14	PWM2
GPIO4	16	IO4	PWM3

The functionality of PWM interfaces can be implemented via software programming. For example, in the LED smart light demo, the function of PWM is realized by interruption of the timer, the minimum resolution can reach as much as 44 ns. PWM frequency range is adjustable from 1000 us to 10000 us.

i.e., between 100Hz and 1KHz. When the PWM frequency is at 1 KHz, the duty ratio will reach 1/22727, and over 14 bit resolution will be achieved at 1KHz refresh rate.

3.8. IR Remote Control

Currently, only one Infrared remote control interface is defined, the pin definition is as below:

Table 14 Pin Definition of IR Remote Control

Pin Name	Pin Num	IO	Function Name
MTMS	9	IO12	IR Tx
GPIO5	24	IO5	IR Rx

The functionality of Infrared remote control interface can be implemented via software programming. NEC coding, modulation, and demodulation are used by this interface. The frequency of modulated carrier signal is 38KHz, while the duty ratio of the square wave is 1/3. The length of data transmission, which is around 1m, is determined by two factors: one is the maximum value of rated current, the other is internal current-limiting resistance value in the infrared receiver. The larger the resistance value, the lower the current, so is the power, and vice versa. The transmission angle is between 15° and 30°, and is mainly determined by the radiation direction of the infrared receiver.

Notes: Among the eight interfaces mentioned above, most of them can be multiplexed. Pin definitions that can be defined is not limited to the eight ones herein mentioned, customers can self customise the functions of the pins according to their specific application scenarios. Functions of these pins can be implemented via software programming and hardware.

3.9. ADC (Analog-to-digital Converter)

ESP8266EX is embedded with a 10-bit precision SARADC. Currently, TOUT (Pin6) is defined as ADC interface, the definition of which is described below:

Pin Name	Pin Num	Function Name
TOUT	6	ADC Interface

Table 16 Pin Definition of ADC

The following two applications can be implemented using ADC (Pin6). However, these two applications cannot be implemented concurrently.

- Test the power supply voltage of VDD3P3 (Pin 3 and Pin 4).

The function used to test the power supply voltage on PA_VDD pin is: `uint16 system_get_vdd33(void)`

- Test the input voltage of TOUT (Pin 6).

4. Firmware & Software Development Kit

The application and firmware is executed in on-chip ROM and SRAM, which loads the instructions during wake-up, through the SDIO interface, from the external flash.

The firmware implements TCP/IP, the full 802.11 b/g/n/e/ri WLAN MAC protocol and WiFi Direct specification. It supports not only basic service set (BSS) operations under the distributed control function (DCF) but also P2P group operation compliant with the latest WiFi P2P protocol. Low level protocol functions are handled automatically by ESP8266:

- RTS/CTS
- acknowledgement
- fragmentation and defragmentation
- aggregation
- frame encapsulation (802.11h/RFC 1042)
- automatic beacon monitoring / scanning, and
- P2P WiFi direct

Passive or active scanning, as well as P2P discovery procedure is performed autonomously once initiated by the appropriate command. Power management is handled with minimum host interaction to minimize active duty period.

4.1. Features

The SDK includes the following library functions:

- 802.11 b/g/n/d/e/v/r support;
- WiFi Direct (P2P) support;
- P2P Discovery, P2P Group Owner mode, P2P Power Management
- Infrastructure BSS Station mode / P2P mode / softAP mode support;
- Hardware accelerators for CCMP (CBC-MAC, counter mode), TKIP (MIC, RC4), WAPI (SMS4), WEP (RC4), CRC;
- WPA/WPA2 PSK, and WPS driver;
- Additional 802.11i security features such as pre-authentication, and TSN;
- Open interface for various upper layer authentication schemes over EAP such as TLS, PEAP, LEAP, SIM, AKA, or customer specific;
- 802.11n support (2.4GHz);
- Supports MIMO 1x1 and 2x1, STBC, A-MPDU and A-MSDU aggregation and 0.4us guard interval;

- WMM power save U-APSD;
- Multiple queue management to fully utilize traffic prioritization defined by 802.11e standard;
- UMA compliant and certified;
- 802.1h/RFC1042 frame encapsulation;
- Scattered DMA for optimal CPU off load on Zero Copy data transfer operations;
- Antenna diversity and selection (software managed hardware);
- Clock/power gating combined with 802.11-compliant power management dynamically adapted to current connection condition providing minimal power consumption;
- Adaptive rate fallback algorithm sets the optimum transmission rate and Tx power based on actual SNR and packet loss information;
- Automatic retransmission and response on MAC to avoid packet discarding on slow host environment;
- Seamless roaming support;
- Configurable packet traffic arbitration (PTA) with dedicated slave processor based design provides flexible and exact timing Bluetooth co-existence support for a wide range of Bluetooth Chip vendors;
- Dual and single antenna Bluetooth co-existence support with optional simultaneous receive (WiFi/Bluetooth) capability.

5. Power Management

The chip can be put into the following states:

- **OFF**: CHIP_PD pin is low. The RTC is disabled. All registers are cleared.
- **DEEP_SLEEP**: Only RTC is powered on – the rest of the chip is powered off. Recovery memory of RTC can keep basic WiFi connecting information.
- **SLEEP**: Only the RTC is operating. The crystal oscillator is disabled. Any wakeup events (MAC, host, RTC timer, external interrupts) will put the chip into the WAKEUP state.
- **WAKEUP**: In this state, the system goes from the sleep states to the PWR state. The crystal oscillator and PLLs are enabled.
- **ON**: the high speed clock is operational and sent to each block enabled by the clock control register. Lower level clock gating is implemented at the block level, including the CPU, which can be gated off using the WAIT instruction, while the system is on.

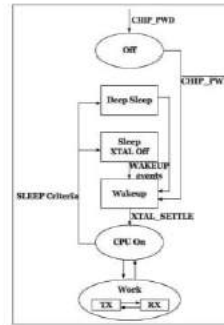


Figure 3 Illustration of Power Management

6. Clock Management

6.1. High Frequency Clock

The high frequency clock on ESP8266EX is used to drive both transmit and receive mixers. This clock is generated from the internal crystal oscillator and an external crystal. The crystal frequency can range from 26MHz to 52MHz.

While internal calibration of the crystal oscillator ensures that a wide range of crystals can be used, in general, the quality of the crystal is still a factor to consider, to have reasonable phase noise that is required for good performance. When the crystal selected is sub-optimal due to large frequency drifts or poor Q factor, the maximum throughput and sensitivity of the WiFi system is degraded. Please refer to the application notes on how the frequency offset can be measured.

Table 18 High Frequency Clock

Parameter	Symbol	Min	Max	Unit
Frequency	FXO	26	52	MHz
Loading capacitance	CL		32	pF
Motional capacitance	CM	2	5	pF
Series resistance	RS	0	65	Ω
Frequency tolerance	ΔFXO	-15	15	ppm
Frequency vs temperature (-25°C ~ 75°C)	ΔFXO_Temp	-15	15	ppm

6.2. External Reference Requirements

For an externally generated clock, the frequency can range from 26MHz to 52MHz can be used. For good performance of the radio, the following characteristics are expected of the clock:

Table 19 External Clock Reference

Parameter	Symbol	Min	Max	Unit
Clock amplitude	VXO	0.2	1	V _{pp}
External clock accuracy	ΔFXO_EXT	-15	15	ppm
Phase noise @1kHz offset, 40MHz clock			-120	dBc/Hz
Phase noise @10kHz offset, 40MHz clock			-130	dBc/Hz
Phase noise @100kHz offset, 40MHz clock			-138	dBc/Hz

7. Radio

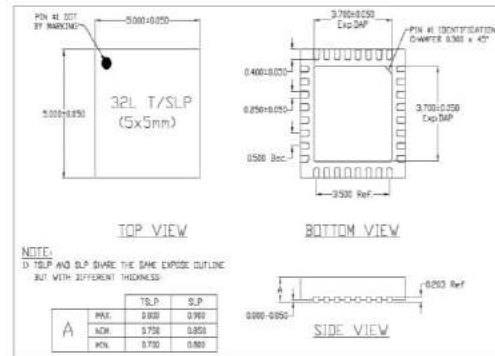
The ESP8266EX radio consists of the following main blocks:

- 2.4GHz receiver
- 2.4GHz transmitter
- High speed clock generators and crystal oscillator
- Real time clock
- Bias and regulators
- Power management

- inductor.
- varactor, and
- loop filter

The clock generator has built-in calibration and self test circuits. Quadrature clock phases and phase noise are optimized on-chip with patented calibration algorithms to ensure the best receiver and transmitter performance.

8. Appendix: QFN32 Package Size



NOTE:
① TSLP AND SLP SHARE THE SAME EXPOSED OUTLINE BUT WITH DIFFERENT THICKNESS

	TSLP	SLP
MAX.	0.80	0.90
MIN.	0.75	0.85
MIN.	0.70	0.80

Anexo B: Código Nodo Supervisión

```
#include <SoftwareSerial.h>//incluimos SoftwareSerial
#include <TinyGPS.h>//incluimos TinyGPS

TinyGPS gps;//Declaramos el objeto gps
SoftwareSerial serialgps(4,3);//Declaramos el pin 4 Tx y 3 Rx
SoftwareSerial zb(8,9);//Declaramos el pin 4 Tx y 3 Rx
//Declaramos la variables para la obtención de datos

unsigned long chars;
unsigned short sentences, failed_checksum;
float latitud,longitud,p;
float codigo=90.00;
void setup()
{
  Serial.begin(9600);//Iniciamos el puerto serie
  serialgps.begin(9600);//Iniciamos el puerto serie del gps
}
void loop()
{
  while(serialgps.available())
  {
    int c = serialgps.read();
    if(gps.encode(c)) {
      gps.f_get_position(&latitud, &longitud);
      //Serial.print(codigo);
      //Serial.print(latitud);
      imprimir();
      delay(500);
    }
  }
}
void imprimir(){
  Serial.print(codigo);
  Serial.print("\n");
  Serial.print(latitud,4);
  Serial.print("\n");
  Serial.print(longitud,4);
  Serial.print("\n");
}
```

Anexo C: Código nodo adquisición

```
1
/*****
 * Libraries included
 *****/

#include <ESP8266WiFi.h>

/*****
 * Constants and objects
 *****/
char const * SSID_NAME = "Ilbay1"; // Put here your SSID name
char const * SSID_PASS = "Le0n@5518"; // Put here your password

char* TOKEN = "BBFF-G1JlD0SdpKpRxwH8vvGwEbOSi24Nte"; // Put here your TOKEN

char* DEVICE_LABEL = "geolocalizacion"; // Your Device label

/* Put here your variable's labels*/
char const * VARIABLE_LABEL_1 = "position";

/* HTTP Settings */
//char const * HTTPSERVER = "http://things.ubidots.com";
char const * HTTPSERVER = "industrial.api.ubidots.com";
const int HTTPPORT = 80;
char const * USER_AGENT = "ESP8266";
char const * VERSION = "1.0";

WiFiClient clientUbi;

/*****
 * Auxiliar Functions
 *****/

void SendToUbidots(char* payload) {

    int contentLength = strlen(payload);

    /* Connecting the client */
    clientUbi.connect(HTTPSERVER, HTTPPORT);

    if (clientUbi.connected()) {
        /* Builds the request POST - Please reference this link to know all the request's structures https://ubidots.com/docs/api/ */
        clientUbi.print(F("POST /api/v1.6/devices/"));
        clientUbi.print(DEVICE_LABEL);
        clientUbi.print(F(" HTTP/1.1\r\n"));
        clientUbi.print(F("Host: "));
        clientUbi.print(HTTPSERVER);
        clientUbi.print(F("\r\n"));
        clientUbi.print(F("User-Agent: "));
        clientUbi.print(USER_AGENT);
        clientUbi.print(F("/"));
        clientUbi.print(VERSION);
        clientUbi.print(F("\r\n"));
    }
}
```

```
clientUbi.print(F("X-Auth-Token: "));
clientUbi.print(TOKEN);
clientUbi.print(F("\r\n"));
clientUbi.print(F("Connection: close\r\n"));
clientUbi.print(F("Content-Type: application/json\r\n"));
clientUbi.print(F("Content-Length: "));
clientUbi.print(contentLength);
clientUbi.print(F("\r\n\r\n"));
clientUbi.print(payload);
clientUbi.print(F("\r\n"));

Serial.println(F("Making request to Ubidots:\n"));
Serial.print("POST /api/v1.6/devices/");
Serial.print(DEVICE_LABEL);
Serial.print(" HTTP/1.1\r\n");
Serial.print("Host: ");
Serial.print(HTTPSERVER);
Serial.print("\r\n");
Serial.print("User-Agent: ");
Serial.print(USER_AGENT);
Serial.print("/");
Serial.print(VERSION);
Serial.print("\r\n");
Serial.print("X-Auth-Token: ");
Serial.print(TOKEN);
Serial.print("\r\n");
```

```

    Serial.print("Connection: close\r\n");
    Serial.print("Content-Type: application/json\r\n");
    Serial.print("Content-Length: ");
    Serial.print(contentLength);
    Serial.print("\r\n\r\n");
    Serial.print(payload);
    Serial.print("\r\n");
} else {
    Serial.println("Connection Failed ubidots - Try Again");
}

/* Reach timeout when the server is unavailable */
int timeout = 0;
while (!clientUbi.available() && timeout < 5000) {
    timeout++;
    delay(1);
    if (timeout >= 5000) {
        Serial.println(F("Error, max timeout reached"));
        break;
    }
}

/* Reads the response from the server */
Serial.println(F("\nUbidots' Server response:\n"));
while (clientUbi.available()) {
    char c = clientUbi.read();

    Serial.print(c); // Uncomment this line to visualize the response on the Serial Monitor
}

/* Disconnecting the client */
clientUbi.stop();
}

/*****
 * Main Functions
 *****/

void setup() {
    Serial.begin(9600);
    /* Connects to AP */
    WiFi.begin(SSID_NAME, SSID_PASS);

    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }

    WiFi.setAutoReconnect(true);
    Serial.println(F("WiFi connected"));
    Serial.println(F("IP address: "));
    Serial.println(WiFi.localIP());
}

```

```

    if (clientUbi.connect(HTTPSERVER, HTTPPORT)) {
        Serial.println("connected to Ubidots cloud");
    } else {
        Serial.println("could not connect to Ubidots cloud");
    }
}

void loop() {

    // Space to store values to send
    char payload[200];
    char str_val_1[30];
    char str_lat[30];
    char str_lng[30];

    /*----- Simulates the values of the sensors -----*/
    if (Serial.available()>0) {
        float uno=Serial.parseFloat();
        if(uno>0) {
            float sensor_value_1=uno;
            float lat=(Serial.parseFloat());
            float lng=(Serial.parseFloat());
            //float sensor_value_1 = random(0, 1000)*1.0;
            //float lat = -1.337259;
            //float lng = -78.640528;

            /*----- Transforms the values of the sensors to char type -----*/

            /* 4 is minimum width, 2 is precision; float value is copied onto str_val*/
            dtostrf(sensor_value_1, 4, 2, str_val_1);
            dtostrf(lat, 4, 2, str_lat);
            dtostrf(lng, 4, 2, str_lng);

            // Important: Avoid to send a very long char as it is very memory space costly, send small char arrays
            sprintf(payload, "{\");
            sprintf(payload, "%s\":"value\":"%s", payload, VARIABLE_LABEL_1, str_val_1);
            sprintf(payload, "%s,\"context\":"lat\":"%s, \"lng\":"%s}", payload, str_lat, str_lng);
            sprintf(payload, "%s", payload);

            /* Calls the Ubidots Function POST */
            SendToUbidots(payload);
        }
        //delay(5000);
    }
}

```
