



Universidad
Nacional
de Loja

Universidad Nacional de Loja

Facultad de la Energía, las Industrias y los Recursos Naturales No
Renovables

Carrera Ingeniería en Mecánica Automotriz

Diseño e implementación de una red de comunicación vehículo a vehículo
(V2V), mediante el protocolo de comunicación ZigBee

Trabajo de Titulación previo a la
obtención del Título de Ingeniero
Mecánico Automotriz

AUTOR:

Aarón Antoniel Sarango Calva

DIRECTORA:

Ing. Génesis Jahel Vásquez Rodríguez, Mg.Sc.

Loja – Ecuador

2023

Certificación.

Loja, 17 de mayo de 2023

Ing. Génesis Jahel Vásquez Rodríguez, Mg.Sc.

DIRECTORA DEL TRABAJO DE TITULACIÓN

CERTIFICO:

Que he revisado y orientado todo el proceso de elaboración del Trabajo de Titulación denominado: **Diseño e implementación de una red de comunicación vehículo a vehículo (V2V), mediante el protocolo de comunicación ZigBee**, previo a la obtención del título de **Ingeniero Mecánico Automotriz**, de la autoría del estudiante **Aarón Antoniel Sarango Calva**, con **cédula de identidad** Nro. 115619249, una vez que el trabajo cumple con todos los requisitos exigidos por la Universidad Nacional de Loja, para el efecto, autorizo la presentación del mismo para su respectiva sustentación y defensa.

Ing. Génesis Jahel Vásquez Rodríguez, Mg.Sc.

DIRECTORA DEL TRABAJO DE TITULACIÓN

Autoría.

Yo, **Aarón Antoniel Sarango Calva**, declaro ser autor del presente Trabajo de Titulación y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos y acciones legales, por el contenido del mismo. Adicionalmente acepto y autorizo a la Universidad Nacional de Loja la publicación de mi Trabajo de Titulación en el Repositorio Digital Institucional – Biblioteca Virtual.

Firma

Cédula de Identidad: 1105619249

Fecha: 17 de mayo del 2023

Correo electrónico: aaron.sarango@unl.edu.ec/ aaron.sarango@gmail.com

Teléfono: 072612102/ 0986666687

Carta de autorización por parte del autor, para consulta, reproducción parcial o total y/o publicación electrónica del texto completo, del Trabajo de Titulación.

Yo, **Aarón Antoniel Sarango Calva**, declaro ser autor del Trabajo de Titulación denominado: **Diseño e implementación de una red de comunicación vehículo a vehículo (V2V), mediante el protocolo de comunicación ZigBee**, como requisito para optar por el título de **Ingeniero Mecánico Automotriz**, autorizo al sistema Bibliotecario de la Universidad Nacional de Loja para que, con fines académicos, muestre la producción intelectual de la Universidad, a través de la visibilidad de su contenido en el Repositorio Institucional.

Los usuarios pueden consultar el contenido de este trabajo en el Repositorio Institucional, en las redes de información del país y del exterior con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia del Trabajo de Titulación que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja, a los diecisiete días del mes de mayo del dos mil veintitrés.

Firma

Autor: Aarón Antoniel Sarango Calva.

Cédula: 1105619249

Dirección: Loja (Jipiro Alto)

Correo electrónico: aaron.sarango@unl.edu.ec/ aaron.sarango@gmail.com

Teléfono: 072612102/ 0986666687

DATOS COMPLEMENTARIOS:

Directora del Trabajo Titulación: Ing. Génesis Jahel Vásquez Rodríguez, Mg.Sc.

Dedicatoria.

Dedico este Trabajo de Titulación, en primer lugar, a Dios por la vida brindada. A mi familia, en especial a mi madre Lucrecia y mi hermano Anthony que siempre han estado presentes con su apoyo, motivación, paciencia y su amor, en todo momento, pero sobre todo durante este trayecto universitario.

Aarón Antoniel Sarango Calva

Agradecimiento.

Primeramente, agradezco a Dios por brindarme su gracia, la sabiduría y la inteligencia para terminar con éxito esta etapa universitaria. A mi madre infinitas gracias por su esfuerzo realizado para poder llegar a ser un profesional, agradezco a mi hermano por su ayuda, sus consejos brindados durante este ciclo de mi vida y por su ayuda en este proyecto investigativo.

Gracias a la ilustre Universidad Nacional de Loja por permitirme realizar mis estudios de tercer nivel en la Facultad de la Energía, las Industrias y los Recursos Naturales no Renovables. A los docentes de la carrera de Ingeniería Automotriz por sus conocimientos y enseñanzas impartidas, a mi tutora Ing. Génesis Vásquez, por orientarme y guiarme en este Trabajo de Titulación. Finalmente, agradecer a mi familia y amigos cercanos que me han brindado motivación para seguir adelante.

Aarón Antoniel Sarango Calva

Índice de contenidos.

Portada	i
Certificación	ii
Autoría	iii
Carta de autorización	iv
Dedicatoria	v
Agradecimiento	vi
Índice de contenidos	vii
Índice de tablas:.....	x
Índice de figuras:.....	x
Índice de anexos:.....	xi
1. Título	1
2. Resumen	2
2.1. Abstract.....	3
3. Introducción	4
4. Marco teórico	5
4.1. Sistemas de transporte inteligente (ITS).....	5
4.2. Redes vehicular Ad Hoc (VANET)	6
4.2.1. Comunicación de corto alcance (DSRC).....	7
4.2.2. WAVE	7
4.3. Tecnologías de comunicación empleados en el vehículo.....	8
4.3.1. Comunicación “vehículo a todo” (V2X)	8
4.3.2. Comunicación “vehículo a vehículo” (V2V).....	8
4.3.3. Comunicación “vehículo a infraestructura” (V2I).....	10
4.4. Redes inalámbricas	11
4.4.1. Definición de redes inalámbricas.....	11
4.4.2. Tipos de redes inalámbricas.....	12
4.4.2.1. WWAN (Wireless Wide Area Network)	12
4.4.2.2. WMAN (Wireless Metropolitan Area Network)	12

4.4.2.3.	WLAN (Wireless Local Area Network).....	12
4.4.2.4.	WPAN (Wireless Personal – Area Networks).....	13
4.5.	Protocolo de comunicación ZigBee.....	13
4.5.1.	Definición	13
4.5.2.	Topologías de red.....	14
4.5.3.	Aplicaciones.....	15
4.5.4.	Dispositivos que lo conforman	16
4.5.5.	Frecuencia de trabajo y rango de velocidades	16
4.5.6.	Seguridad en ZigBee.....	16
5.	Metodología.....	18
5.1	Equipos e instrumentos.....	18
5.1.1	Módulo Xbee	18
5.1.4.	Sensores utilizados.....	20
5.1.5.	Fritzing.....	21
5.1.6.	Cisco Packet Tracer	22
5.1.7.	XCTU.....	22
5.2	Protocolos de comunicación vehicular y tecnología V2V en el vehículo	22
5.3	Construcción del prototipo de comunicación V2V	22
5.3.1	Diseño de la red inalámbrica	23
5.3.2	Conexión de sensores a utilizar	24
5.4	Implementación de la red de comunicación y visualización de datos	32
5.4.1	Configuración de módulos Xbee	33
5.4.2	Programación en Arduino	35
5.4.3	Implementación del prototipo de comunicación en los vehículos	39
5.4.3.1	Visualización de datos	40
6.	Resultados.	42
6.1.	Protocolos de comunicación y tecnología V2V	42
6.2.	Diseño de la red de comunicación	44

6.3. Implementación y visualización de la red de comunicación creada.....	46
6.3.1. Implementación	47
6.3.2. Visualización en Ubidots	52
7. Discusión.	55
8. Conclusiones.	57
9. Recomendaciones.	58
10. Bibliografía.	60
11. Anexos.	65

Índice de tablas:

Tabla 1. Características de componentes de red ZigBee	16
Tabla 2. Características del módulo Xbee Pro S2B	19
Tabla 3. Descripción de sensores utilizados.....	21

Índice de figuras:

Figura 1. Sistema de transporte inteligente (ITS)	6
Figura 2. Tecnología de Comunicación V2V.....	8
Figura 3. Esquema del protocolo V2I	10
Figura 4. Relación entre tecnología V2I y V2V	11
Figura 5. Tipos de redes inalámbricas.....	13
Figura 6. Topologías de ZigBee	15
Figura 7. Módulo Xbee Pro S2B.....	18
Figura 8. Trama de datos UART	19
Figura 9. Adaptador Xbee Shield.....	20
Figura 10. Adaptador Xbee Explorer	20
Figura 11. Elementos utilizados para el diseño de la red inalámbrica V2V	23
Figura 12. Diagrama de flujo del diseño de la red	24
Figura 13. Circuito eléctrico de sensor de humedad y temperatura DHT11	25
Figura 14. Conexión del sensor DHT11 en el protoboard.	25
Figura 15. Circuito eléctrico de sensor de aceleración ADXL345	26
Figura 16. Conexión del sensor de aceleración ADXL345 en el protoboard	27
Figura 17. Circuito eléctrico de sensor de luminosidad BH1750	28
Figura 18. Conexión del sensor de luminosidad BH1750 en el protoboard	28
Figura 19. Circuito eléctrico de sensor de distancia TF Mini Plus	29
Figura 20. Conexión del sensor de distancia TF mini Plus en el protoboard.....	29
Figura 21. Circuito eléctrico del módulo GPS	30
Figura 22. Conexión del módulo GPS en el protoboard	30
Figura 23. Circuito eléctrico de modulo Xbee	31
Figura 24. Conexión del módulo Xbee en el protoboard	31
Figura 25. Circuito eléctrico de la pantalla LCD	32
Figura 26. Conexión de pantalla LCD por interfaz I2C.....	32
Figura 27. Adaptador de módulo Xbee	33
Figura 28. Descarga de firmware para Xbee Coordinador	33

Figura 29. Descarga de firmware para Xbee Router.....	34
Figura 30. Diagrama de flujo de la configuración de módulos Xbee	35
Figura 31. Montaje de sensores en Arduino esclavo.....	36
Figura 32. Montaje de sensores en Arduino maestro	36
Figura 33. Diagrama de flujo de programación en Arduino IDE	37
Figura 34. Diagrama de bloques de la programación y comunicación de la red	39
Figura 35. Conexión de adaptador Ethernet a Arduino Mega	40
Figura 36. Diagrama de flujo de configuración en Ubidots.....	41
Figura 37. Diseño de la red de comunicación V2V	44
Figura 38. Transmisión de datos del módulo Xbee router.....	45
Figura 39. Recepción de datos del módulo Xbee coordinador	45
Figura 40. Prueba de rango de cobertura a distancias menor a 500 m.....	46
Figura 41. Prueba de rango de cobertura a distancias mayores a 500 m.....	46
Figura 42. Implementación de elementos en el vehículo coordinador.....	47
Figura 43. Implementación de elementos en el vehículo esclavo	48
Figura 44. Advertencia de baja luminosidad.....	48
Figura 45. Advertencia de condiciones desfavorables	49
Figura 46. Advertencia de posible peligro	50
Figura 47. Advertencia de una distancia insegura	50
Figura 48. Vehículos entrando en una intersección	51
Figura 49. Advertencia en una intersección	51
Figura 50. Gráfica del sensor de distancia	52
Figura 51. Gráfica del sensor de aceleración	53
Figura 52. Gráfica del sensor de luminosidad.....	53
Figura 53. Gráfica del sensor de temperatura	54
Figura 54. Gráfica del sensor de humedad.....	54

Índice de anexos:

Anexo 1. Proceso de construcción del prototipo de comunicación V2V (Coordinador).....	65
Anexo 2. Vehículos utilizados para la comunicación V2V	65
Anexo 3. Condiciones de prueba para el sensor de luminosidad.....	65
Anexo 4. Condiciones de prueba para el sensor de aceleración	66
Anexo 5. Prueba de rango de cobertura a una distancia menor a 500 m	66

Anexo 6. Prueba de rango de cobertura a una distancia mayor a 500 m	67
Anexo 7. Caja para el prototipo (vehículo emisor)	67
Anexo 8. Caja para el prototipo (vehículo receptor).....	68
Anexo 9. Datasheet del sensor de aceleración (ADXL345)	68
Anexo 10. Datasheet del sensor de luminosidad (BHT1750FVI).....	68
Anexo 11. Datasheet del sensor de distancia Tf Mini Plus.....	68
Anexo 12. Datasheet del sensor de temperatura y humedad.....	68
Anexo 13. Datasheet del Módulo GPS	68
Anexo 14. Programación del bloque del vehículo emisor	68
Anexo 15. Programación del bloque del vehículo receptor	71
Anexo 16. Certificado de traducción	80

1. Título

Diseño e implementación de una red de comunicación vehículo a vehículo (V2V), mediante el protocolo de comunicación ZigBee.

2. Resumen.

En este proyecto de investigación se explora los protocolos de comunicación mayormente utilizados en la comunicación entre vehículos, además de los beneficios y las barreras por superar de la tecnología vehículo a vehículo (V2V).

El presente proyecto de investigación se enfocó en diseñar e implementar un prototipo de comunicación vehículo a vehículo, con el fin de que este sistema pueda ser el inicio de nuevos proyectos basados en esta innovadora comunicación para que en el futuro ayuden a prevenir y evitar accidentes automovilísticos.

En el prototipo implementado, el intercambio de datos se llevó a cabo mediante módulos Xbee, los cuales están basados en el protocolo de comunicación inalámbrico ZigBee. Para el diseño del prototipo, se ha empleado el software Cisco Packet Tracer, el cual permite simplificar y que sea fácil de comprender la arquitectura de la red de comunicación creada. Adicional, se empleó el programa Fritzing para el diseño de las conexiones eléctricas de los diferentes componentes que integran el sistema.

Para la instalación en el hardware, se utilizó una placa de pruebas, además de un microcontrolador Arduino Mega 2560, en el que se configuró y se caracterizó cada uno de los sensores de la red, además este microcontrolador procesa el algoritmo creado y a partir de este se puede emitir las advertencias necesarias para el conductor, en distintos escenarios de conducción en la vía.

Se llevó a cabo la implementación del prototipo de comunicación en dos vehículos particulares, y se sometió a un ensayo mediante pruebas de ruta dentro de la ciudad de Loja. Finalmente, para tener una apreciación de la información y de los valores obtenidos de los sensores, que están siendo procesados en la red de comunicación, se visualizará en una plataforma compatible con Arduino, denominada Ubidots.

Palabras claves: redes, comunicación, sensores, ZigBee, vehículo a vehículo (V2V).

2.1. Abstract.

This research project explores the communication protocols mostly used in vehicle-to-vehicle communication, as well as the benefits and barriers to overcome of vehicle-to-vehicle (V2V) technology.

The present research project focused on designing and implementing a prototype of vehicle-to-vehicle communication, so that this system could be the beginning of new projects based on this innovative communication to help prevent and avoid car accidents in the future.

In the implemented prototype, the data exchange was carried out through Xbee modules, which are based on the ZigBee wireless communication protocol. For the design of the prototype, the Cisco Packet Tracer software was used, which simplifies and makes it easy to understand the architecture of the communication network created. In addition, the Fritzing program was used to design the electrical connections of the different components that make up the system.

For the hardware installation, a test board was used, in addition to an Arduino Mega 2560 microcontroller, in which each of the sensors of the network was configured and characterized, in addition this microcontroller processes the algorithm created and from this it can issue the necessary warnings to the driver, in different scenarios of driving on the road.

The implementation of the communication prototype was carried out in two private vehicles, and it was tested by means of road tests within the city of Loja. Finally, in order to have an appreciation of the information and values obtained from the sensors, which are being processed in the communication network, it will be visualized in an Arduino compatible platform, called Ubidots.

Keywords: networks, communication, sensors, ZigBee, vehicle to vehicle (V2V).

3. Introducción.

Los siniestros automovilísticos ocasionan millones de muertes y de heridos en todo el mundo año tras año. Según la (OMS, 2022) estimula que se pierden 1,3 millones de vidas a consecuencias de colisiones automovilísticas. En el año 2019 en Ecuador según (ANT,2023), se registró 24.595 accidentes de tránsito, entre los cuales 19.999 han resultado heridos y 2.180 han fallecido. A nivel de localidad, en la ciudad de Loja, se han registrado 624 accidentes, de los cuales han existido 427 heridos, y 52 fallecidos. Motivo por el cual, la industria automotriz ha estado siempre enfocada en tratar de prevenir y disminuir estos accidentes fatales, para ello, se han creado e implementado diversos sistemas de seguridad pasiva o activa en el vehículo, e inclusive se sigue investigando más tecnologías que contribuyan a una mayor seguridad, una de estas tecnologías en exploración es la comunicación vehículo a vehículo (V2V), la cual posibilita la transmisión de información entre los vehículos mediante transmisiones inalámbricas, esto con el fin de prevenir accidentes o dar soluciones al problema del tráfico vehicular.

En este estudio en cuestión, se implementó un prototipo de comunicación vehículo a vehículo (V2V), mediante el protocolo ZigBee, el cual pretende alertar y prevenir al conductor de posibles peligros presentes en la conducción mediante mensajes de alerta. Donde la información que es recibida de un vehículo a otro, es de suma importancia para el conductor, puesto que le permite conocer las situaciones que le rodea al transitar por la vía, además le brinda un tiempo suficiente para reaccionar y evitar probables situaciones de riesgo.

Para dar el cumplimiento de los objetivos planteados en este proyecto de investigación, se han planteado tres etapas, de modo que se inició por una búsqueda bibliográfica, que respalde los protocolos de comunicación implementados en la comunicación vehículo a vehículo y las aplicaciones que se han desarrollado en relación a los sistemas de comunicación V2V. En la segunda etapa, se diseñó la red de comunicación vehículo a vehículo, en un software específico. En la etapa final, se implementó la red diseñada, y se probó el protocolo de comunicación en dos vehículos para poder visualizar los datos recibidos en una plataforma digital.

4. Marco teórico.

4.1. Sistemas de transporte inteligente (ITS)

El sistema de transporte inteligente, es un término en general que hace referencia a la unificación de tecnologías de comunicaciones, control y el procesamiento de información de redes vehiculares. Los beneficios que pueden generar a futuro son salvar vidas, prevenir accidentes automovilísticos, tiempo, dinero y ayudar a la conservación del medio ambiente. (Anand et al., 2017, p.21).

El ITS se basa en recopilación, integración, procesamiento y suministro de información sobre redes de transporte, esta información generada por parte del ITS en tiempo real, informa sobre las condiciones existentes en las carreteras, permitiendo a los viajeros, autoridades y a las agencias de carretera, que puedan tomar decisiones seguras y armonizadas en base a la información existente. (Anand et al., 2017, p.21).

ITS tiene algunas aplicaciones en las que puede ser empleada, como por ejemplo en París – Francia, se realizó una prueba operativa en el proyecto SCORE@F en el año 2010 al 2013, centrándose en los servicios cooperativos V2V y V2I (comunicación de vehículo a infraestructura), que aumenta la seguridad en la conducción. El campo de pruebas utilizado para las pruebas han sido autopistas con túneles, vías urbanas al suroeste de Paris, los resultados obtenidos son enviados a las empresas de normalización europea, como son ETSI (Instituto Europeo- de Normas de Telecomunicaciones) y CEN (Comité Europeo de Normalización). (Anand et al., 2017, p.33).

Asimismo, dentro de Sudamérica, en Brasil se avanza con un programa obligatorio de identificación de los vehículos, que estará basado en etiquetas electrónicas, con el fin de que cancelen impuestos o tarifas de registro de vehículos, peaje en las carreteras, entre otras, la tecnología fue implementada en este país en el año 2017. (Anand et al., 2017, p.34).

Así como estos países que se han nombrado, existen diversos países, que igualmente han desarrollado diferentes tecnologías ITS, entre los cuales se resalta España, Australia, China, Corea, Canadá, Estados Unidos, Taiwán, entre otros más. (Anand, 2017).

en la red CAN. (Li, 2015), la cual es una representación de bus serial, en la que los datos se encuentran circulando constantemente dentro de la red, para ser aprovechados por cualquier unidad de control que lo desee. (Ferrer y Domínguez, 2012, p.88). También el OBU puede integrarse en la red FlexRay, la cual es una red de alta velocidad, pero más costosa en relación a CAN. Mientras que la unidad de carretera (RSU), está separado de CAN. (Li, 2015).

Los estándares para las redes VANET, están dadas en Estados Unidos como comunicaciones de corto alcance (DSRC), basada por la normativa IEEE 802.11p, para países de Europa cuentan con otra normativa CEN/TC278 y para Japon la ISO/TC204. (Anand et al., 2017, p.16).

4.2.1. Comunicación de corto alcance (DSRC)

Son comunicaciones dedicada de corto alcance, que tiene una similitud y asociación con las comunicaciones WAVE, la comunicación DSRC, permite proporcionar altas transferencias de datos y baja latencia para redes vehiculares, dentro de un área pequeña. DSRC, se presenta en algunas aplicaciones como la información de tráfico, mensajes de seguridad (V2V), cobro de peajes, entre otras más. Este sistema en la actualidad no es compatible, por lo que la normalización tiene un papel importante para asegurar la interoperabilidad. (Anand et al., 2017, p.16).

La Comisión Federal de Comunicaciones de Estados Unidos (US-FCC), asignó un rango de frecuencia de 5,850 a 5,925 GHz, como un espectro de licencia libre que utiliza DSRC, para poder mejorar la seguridad vial. Además, DSRC proporciona una velocidad de datos de 27 Mbps (Morgan, 2010).

4.2.2. WAVE

Sistema que se encuentra enfocado en proporcionar servicios continuos e interoperables para ITS, se encuentra orientado principalmente en la comunicación vehículo a vehículo (V2V) y vehículo a infraestructura (V2I), para especificar tecnologías asociadas a WAVE, se suele utilizar el término DSRC, por lo que suelen ser términos que pueden estar asociados. (Anand et al., 2017, p.16).

El instituto de ingenieros eléctricos y electrónicos (IEEE), define la arquitectura del sistema WAVE, bajo IEEE 1609, en el cual se estandariza el conjunto de servicios e interfaces que permiten la comunicación inalámbrica segura, como el acceso físico para comunicaciones inalámbricas de alta velocidad, de aproximadamente 27 Mbps y de un corto alcance de un rango de hasta 1000 m, incluyendo una baja latencia en el entorno vehicular. (Shereen et al., 2013).

4.3. Tecnologías de comunicación empleados en el vehículo

Existen ciertos tipos de tecnologías de comunicación implementadas en el vehículo entre las cuales tenemos:

4.3.1. Comunicación “vehículo a todo” (V2X)

La comunicación vehículo a todo, es aquella que alberga todo tipo de comunicación en el vehículo, como las comunicaciones V2V, V2I, V2P, V2H, V2B, V2C siendo las comunicaciones V2V y V2I las más exploradas y utilizadas. Además de esto también se ha creado una variación de la comunicación V2X, la cual es la versión C-V2X, la cual incluye el uso del celular a la comunicación en el vehículo, esto gracias a la red 5G, en el que está en conexión con su alrededor, tanto la infraestructura como con las personas. (Huidobro, 2021)

4.3.2. Comunicación “vehículo a vehículo” (V2V)

La comunicación intervehicular V2V es un estándar que se viene estudiando desde tiempo atrás, se refiere al intercambio existente de datos o información, entre dos vehículos, esto con el fin de ayudar a mejorar el tránsito vehicular, como también reduciendo o evitando los accidentes automovilísticos. A un futuro se plantea que esta tecnología permita evitar secuestros y robos de identidad. La tecnología V2V, aplica el mismo concepto de *machine to machine* (M2M por sus siglas en inglés), en el que permite intercambiar información entre dos objetos, pero en este caso hará la interacción en vehículos. (Medina, 2019).

Figura 2.

Tecnología de Comunicación V2V



Nota. Adoptado de *Vehicle-to Vehicle communications Technology*, por NHTSA, 2014.

El objetivo del sistema de comunicación vehículo a vehículo, no está pensada en si utilizar los recursos propios para la seguridad de los ocupantes, más bien es una tecnología desde el punto de vista, en el que los vehículos compartan información sobre sí mismos, y luego esta información pueda ser analizada y procesada por una organización, que posteriormente se devuelven estos datos al resto de vehículos de la red. Entonces, se podrá informar o alertar al conductor para evitar posibles accidentes y evitar el tráfico vehicular. (Medina, 2019)

Esta tecnología, en lo que respecta a evitar accidentes de tránsito, se basa en algunos escenarios mencionados por (NHTSA, 2016). Estas se describirán a continuación:

- **Aviso de posible colisión frontal:** Esta función permitirá advertir al conductor sobre vehículos que se encuentran detenidos, o que reducen su velocidad de manera brusca, además advierte del riesgo de una colisión trasera inminente con un vehículo de adelante, en el mismo carril y dirección de viaje.
- **Alerta de colisión en intersección:** Esta tecnología permitirá dar una advertencia al conductor, cuando no es seguro entrar en una intersección, debido a que puede existir un potencial peligro de colisión entre uno o más vehículos.
- **Luz de freno electrónica de emergencia:** Advierte al conductor de frenadas bruscas en la fila de tráfico, esta función permitirá al vehículo transmitir una señal cuando su freno de emergencia este activado, al vehículo que viene atrás, recibe una advertencia de que existe un vehículo parado en la vía. Esta función es importante sobre todo cuando existen condiciones climáticas adversas.
- **Advertencia de no rebasar:** indica al conductor que un vehículo se aproxima en dirección opuesta, cuando intenta rebasar a un vehículo que circula más lento en una carretera de dos carriles, por lo que ayudaría a evitar choques en dirección opuestas.
- **Advertencia de punto ciego o cambio de carril:** previene a los conductores sobre la presencia de autos que se aproximan o se encuentran en su punto ciego en el carril adyacente, el sistema V2V advierte y previene posibles accidentes.

Se conoce que el sistema V2V está disponible para camionetas, autos, autobuses e incluso para motos, para el futuro se prevé que esta tecnología pueda llegar a implementarse en las bicicletas y peatones que permita mejorar los sistemas, a parte como sistema de seguridad, la información que se transmite no identifica al conductor o al vehículo. (NHTSA, 2017). Además, la comunicación vehículo a vehículo, tiene un alcance aproximado de 300 m, distancia que proporciona el suficiente tiempo de alertar al conductor de posibles accidentes. (NHTSA, 2014).

Asimismo, la entidad NHTSA (2016), menciona los gastos necesarios para implementar un sistema V2V en Estado Unidos, los cuales llegarían a costes de \$341 y \$350 por vehículo en 2020, probablemente para 2058 el costo disminuya de \$209 a \$227.

Finalmente, la tecnología V2V presenta algunas barreras que pueden afectar a tener aceptación pública, que deben ser superadas, a continuación, se presenta los aspectos más importantes dados por la (NHTSA, 2016).

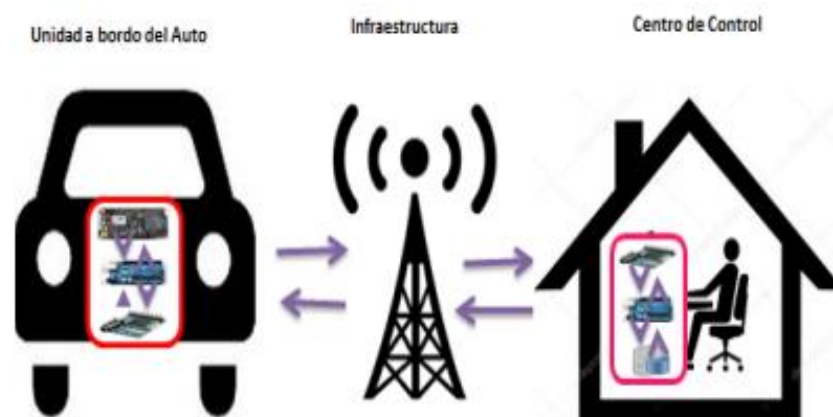
- **Falsos positivos:** sucede cuando se emite una advertencia al conductor, porque no existe un riesgo de seguridad inmediato que el conductor no haya tenido en cuenta. Estos falsos positivos pueden llegar a causar un asusto o una molestia al conductor por lo que no confiará en el sistema.
- **Privacidad:** si el conductor no se siente confiado y piensa que las comunicaciones V2V vigilen sus movimientos para fines privados o gubernamentales, va a evitar obtener un vehículo equipado con tecnología V2V.
- **Hackeo:** igualmente si el conductor no se siente seguro y teme que ladrones cibernéticos ingresen a los sistemas computarizados del vehículo, y dañen su vehículo o roben información. El consumidor no se va a sentir con la suficiente privacidad, por lo que no compraría un vehículo equipado con tecnología V2V.

4.3.3. Comunicación “vehículo a infraestructura” (V2I)

La comunicación V2I (vehículo con infraestructura), hace referencia al intercambio bidireccional de datos e información, entre vehículos, llámense estos autos, camionetas, autobuses, con semáforos, señalética en el carril y con más infraestructuras de los caminos inteligentes, por medio de conexión de redes inalámbricas. Su objetivo principal, es permitir reducir los accidentes, es decir que la seguridad aumente, evitando posibles lesiones, fallecimientos de los conductores u ocupantes el automóvil. (Siam, 2019).

Figura 3.

Esquema del protocolo V2I



Nota. Tomado de Seguimiento automático de trayectorias vehiculares usando el protocolo V2I, por Ramírez et al., 2018, Tecnología e innovación.

De esta tecnología se destaca dos ejemplos fundamentales y explicativos.

- **Semáforos inteligentes**

Los semáforos normales, cambian la luz en intervalos determinados, estos cambios de luz suelen variar o ajustarse dependiendo de la hora, por ejemplo, de horas pico. En cambio, los

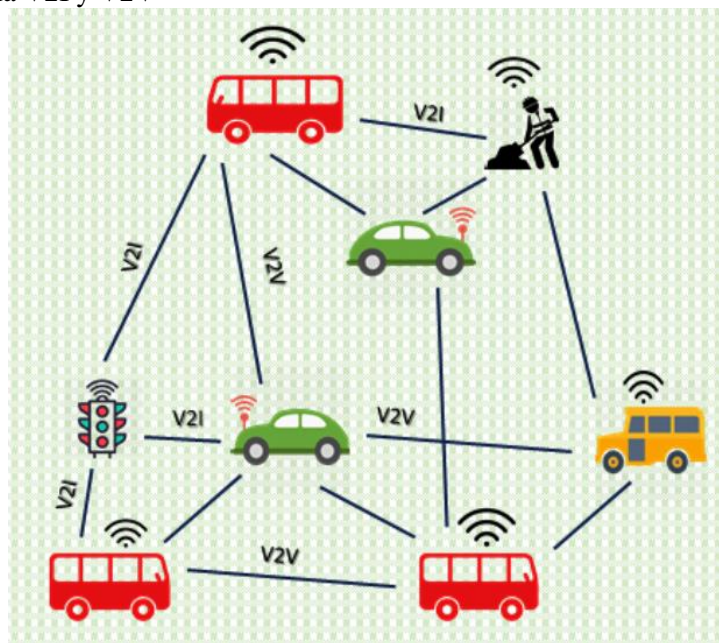
semáforos inteligentes, interactúan con el entorno, se encuentran equipados con sensores y cámaras, que detectan el número de automóviles que se encuentran esperando en cada carril, como el tiempo que llevan esperando, con esta información pueden calcular un tiempo adecuado para descongestionar cada carril. (Siam, 2019).

- **Estacionamientos inteligentes**

Estos estacionamientos inteligentes, permiten estacionamientos en las calles de ciudades, estaciones de trenes y aeropuertos. Permiten monitorear la calidad del aire, la iluminación, como también se comparte la información de los espacios disponibles de estacionamiento, ayuda a reducir las emisiones de los vehículos que circulan o están en ralentí buscando un lugar para poder estacionar. Estos estacionamientos ya se han visto implementados en países como Nueva Zelanda, Australia y Reino Unido. (Siam, 2019).

A la vez la tecnología V2V y V2I que se han descrito anteriormente se pueden relacionar, como se observa en la siguiente imagen, en la que se aprecia que un vehículo puede tener comunicación con una infraestructura y a la vez también puede comunicarse y transmitir información con otros vehículos.

Figura 4.
Relación entre tecnología V2I y V2V



Nota. Adaptado de Secure V2V and V2I Communication in Intelligent Transportation using Cloudlets, por Gupta et al., 2020, IEEE.

4.4. Redes inalámbricas

4.4.1. Definición de redes inalámbricas

Las redes inalámbricas se pueden definir como redes que hacen uso de ondas de radio, con una conexión de alta velocidad para comunicar o conectar uno o varios dispositivos, sin

tener que utilizar cableado, (Salazar, 2016), esto quiere decir que se ahorrará algunos costos. Esta tecnología ha permitido conectar dispositivos remotos, que pueden encontrarse a uno, dos metros o incluso a kilómetros de distancia, sin tener necesidad de romper huecos en las paredes para pasar conectores o cables. (Salazar, 2016)

Se puede conocer cuatro grupos diferentes, dependiendo del área de aplicación, como también del alcance de la señal:

4.4.2. Tipos de redes inalámbricas

4.4.2.1. WWAN (Wireless Wide Area Network)

Redes WWAN, en español conocidas como redes inalámbricas de área extensa, estas redes su cobertura, se extienden más allá de 50 km, las frecuencias por lo general utilizan frecuencias con licencia. (Salazar, 2016)

Estas redes mayormente son utilizadas para grandes áreas, como países y ciudades, gracias a los diferentes sistemas de satélites o ubicaciones con antena, que cuentan con un servidor de internet, se puede generar la comunicación de la red, gracias a los múltiples sistemas satelitales. Las tecnologías mayormente conocidas son la telefonía móvil y satélite. Estas redes son llamadas sistemas de segunda generación. (Sharma y Dhir, 2014).

4.4.2.2. WMAN (Wireless Metropolitan Area Network)

Las redes WMAN, traducidas sus siglas al español, como redes inalámbricas de área metropolitana. La tecnología más importante desarrollada en este tipo de red, es Wimax, la cual ofrece una tasa de transferencia de aproximadamente 70 Mbps, a distancias de 50 Km, compartiendo datos a miles de usuarios desde una estación base. (Coñapes, 2015)

Opera en dos bandas de frecuencias, como lo son de 2 a 11 GHz (banda con licencia), 10 a 66 GHz (banda sin licencia). Las redes WMAN son muy utilizadas en lo que respecta a las grandes compañías de telecomunicaciones, puesto que es optimizado mayormente para trabajar en espacios externos. (Salazar, 2016).

4.4.2.3. WLAN (Wireless Local Area Network)

Las redes WLAN, traducido al español como redes inalámbricas locales, que, son redes de baja cobertura, que permiten la comunicación de dos o más dispositivos a través de ondas de radio, en el que todos los dispositivos de la red se encuentran en un área definida, y alcanzan a cubrir distancias de 100 metros aproximadamente. (Salazar, 2016).

Así mismo estas redes tienen una velocidad de transmisión de datos de 11 Mbps hasta llegar a los 54 Mbps. Un ejemplo claro de las redes WLAN, es el Wi-Fi, el cual se encuentra

normalizado por la IEEE 802.11, estándar que reemplazaría los cables de la conexión alámbrica, a una conexión inalámbrica con Ethernet. (Salazar, 2016).

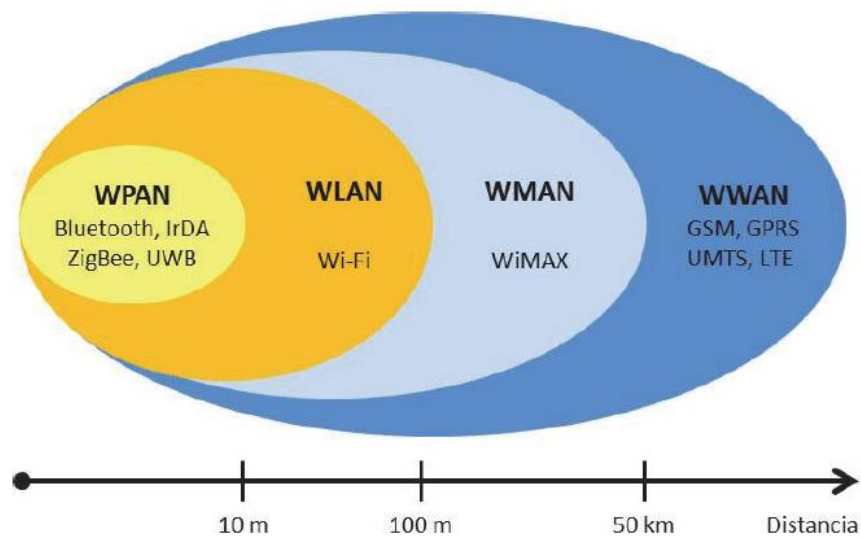
4.4.2.4. WPAN (Wireless Personal – Area Networks)

Las redes inalámbricas de tipo WPAN, también se conocen en español como redes de área personal, cubren distancias alrededor de 30 pies o de 10 metros, pueden ser implementadas en dispositivos portátiles de uso personal, como por ejemplo en teléfonos inteligentes, ordenadores portátiles, entre más. (Coñapes, 2015).

La red implica poca o ninguna infraestructura o conectividad directa con el mundo fuera del enlace. Lo que permite implementar soluciones pequeñas, económicas y de bajo consumo para una amplia gama de dispositivos. Tres organismos de estándares implementan las redes WPAN, como son Bluetooth, UWU, ZigBee. (Sharma y Dhir, 2014).

Finalmente, a continuación, se muestra una comparación entre los diferentes tipos de redes inalámbricas antes mencionadas.

Figura 5.
Tipos de redes inalámbricas



Nota. Adaptado de Redes Inalámbricas, por Salazar, 2016, Techpedia.

4.5. Protocolo de comunicación ZigBee

4.5.1. Definición

ZigBee es oficialmente un protocolo de red inalámbrica ligero que está diseñado para usarse con baja transmisión de datos de redes de sensores y control. (Salazar, 2016). La tecnología inalámbricas ZigBee, está basado en la normativa IEEE 802.15.4, es de corto alcance, de bajo consumo, no es costoso, también cuenta con velocidades bajas de transmisión. (Salazar, 2016).

Una de las funciones principales de ZigBee, es crear una topología de red jerárquica para que cierto número de dispositivos se comuniquen entre ellos, como también permite establecer características adicionales de comunicación, entre estas se destaca la autenticación, encriptación, de asociación y en los servicios de aplicación de la capa superior. (Vera, et al., 2017). La red ZigBee fue construida, específicamente para tratar la necesidad de una implementación de muy bajo costo, y la tasa de datos tiene un consumo de energía ultra bajo.

La tecnología inalámbrica ZigBee opera con velocidades máximas de hasta 250 Kbps, esto es suficiente para satisfacer las necesidades de los sensores y tratar con eficiente la automatización usando redes inalámbricas, (Salazar, 2016).

Asimismo, en el protocolo ZigBee se puede conectar más de 65.000 nodos a la red. (Elahi y Gschwender, 2009, p.33).

4.5.2. Topologías de red

- **Topología Estrella**

Se compone de un coordinador y varios dispositivos finales. En la topología estrella el coordinador inicia la configuración de la red al establecer un identificador de red (PAN ID), el cual no está siendo utilizado por un coordinador cercano, esto permite que funcione de forma independiente, que el identificador PAN sea único, aun si se encuentran redes cercanas al rango de cobertura. (Farahani, 2008, p.10).

- **Topología en Árbol**

La topología de tipo árbol, permite a los dispositivos finales puedan unirse unos con otros, mediante coordinadores o ruteadores, En esta topología, los routers o también conocidos como enrutadores, tienen dos funciones principales, puesto que un dispositivo ZigBee, puede unirse a la red a través de un router, permite que más nodos puedan participar dentro de la red de árbol, así también al incluir un enrutador, elimina la necesidad que el dispositivo final, se encuentre dentro del área de cobertura del coordinador de la red ZigBee. (Eady, 2007).

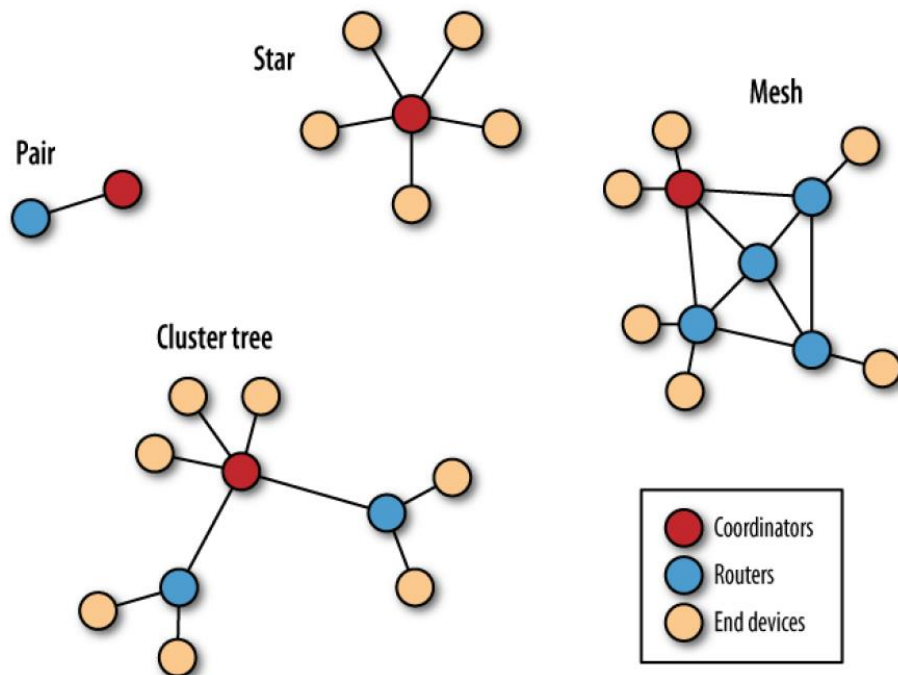
- **Topología de Malla**

En este tipo de topología, todos los dispositivos de la red están interconectados, consiguiendo una comunicación directa entre ellos. Debido, al no depender de un solo dispositivo para la gestión de la red, se gana fiabilidad con lo que respecta al resto de topologías. (Eady, 2007).

Con la configuración mallada, si existe un nodo que se encuentra en la ruta de un enlace de falla, se puede garantizar que la comunicación seguirá activa, puesto que se reorganizará las

rutas, y la información tomará otra ruta, para seguir con la comunicación en los dispositivos. (Eady, 2007). En la Figura 6 se presenta las topologías antes mencionadas.

Figura 6.
Topologías de ZigBee



Nota. La topología par, no se ha tomado en cuenta. Adaptado de Building Wireless Sensor Networks with ZigBee, Xbee, Arduino, and Processing, por Faludi, 2011, (p.28).

4.5.3. Aplicaciones

ZigBee, cuenta con un sinnúmero de aplicaciones dentro de la industria, como caseras. Tienen un decente requerimiento en la transmisión de datos, pero que demandan que sean seguras y confiables, con una tecnología sencilla y de baja potencia. Por lo cual el protocolo ZigBee ha sido diseñado para poder satisfacer estas necesidades, resultando económico y sencillo, además de ser seguro. Las principales aplicaciones que mencionan (Hasan et al., 2018), son:

Dentro del mundo automotriz se puede utilizar como complemento de las cámaras de tráfico. Además, “Agregar ZigBee Smart Energy a los PEV brindará a los fabricantes de automóviles y empresas de servicios públicos un lenguaje común para administrar la carga, el almacenamiento y el uso de energía en los PEV” según menciona Richard Scholer, presidente del Comité del Grupo de Trabajo de Comunicaciones de Vehículos Híbridos de SAE. (Pnewswire, 2011).

ZigBee se puede utilizar para interconectar ECU, sensores y actuadores en un automóvil con el fin de controlar las aplicaciones en el vehículo, monitoreo y lectura de estado. (Huang, et al., 2019).

4.5.4. Dispositivos que lo conforman

Los dispositivos que conforman una red Zigbee, estarán basados, según las funciones que puedan realizar, entre ellos se pueden encontrar tres principales los cuales se describirán en la Tabla 1, características descritas por (Elahi y Gschwender, 2009, p.34).

Tabla 1.
Características de componentes de red ZigBee

Dispositivo	Características
Coordinador	Inicia la red. Selecciona el canal a utilizar por la red. Procesa las peticiones de solicitud del router o de los dispositivos finales.
Router	Busca la mejor ruta hacia el destino, sobre el que quiere transmitir datos. Ocupado en topologías, como de árbol y de malla, permitiendo expandir la cobertura total de la red.
Dispositivo Final	Se une o sale de la red. No realiza procesos de enrutamiento

4.5.5. Frecuencia de trabajo y rango de velocidades

En el protocolo ZigBee, se encuentran diferentes tipos de frecuencias, mencionados por (Farahani, 2008, p.6). Exactamente son tres tipos de bandas.

- 868-868.6 MHz (banda de 868 MHz):
- 902-928 MHz (banda de 915 MHz)
- 2400-2483.5 MHz (banda de 2.4 GHz)

Las bandas de 868 MHz y 915 MHz, son usadas en Europa y América del Norte respectivamente. Mientras que la banda de 2.4 GHz es usada en todo el mundo. Por lo que esta banda también permite un ancho de banda grande y más canales. Con respecto al tamaño de las antenas en la banda de 2.4 GHz, son menores, con respecto a las otras dos bandas. (Farahani, 2008, p.6).

4.5.6. Seguridad en ZigBee

La seguridad en una red inalámbrica, tiene realmente un papel importante al momento de la transmisión de los datos, puesto que esta transferencia de información en una red inalámbrica puede ser recibida por cualquier dispositivo cercano, incluso un intruso. Por lo que se busca dentro de una red, varios elementos necesarios para garantizar una plena seguridad en la red. (Farahani, 2008, p.22).

- **Confidencialidad:** el cual debe de proporcionar privacidad, al mismo tiempo que protege la información para que no sea interceptada, incluso si un intruso trata de obstruir la información, está ya no tendría significado. (Elahi y Gschwender, 2009, p.155).
- **Autenticación:** los métodos de autenticación permiten verificar la identidad de una persona o computadora que está accediendo a la red. Ya que puede existir un dispositivo intruso que puede modificar, reenviar los mensajes, incluso si estos se encuentran encriptados. (Elahi y Gschwender, 2009, p.155).
- **Integridad:** permite que los datos se mantengan coherentes, evitando la manipulación de la información. Las limitaciones presentes en una red ZigBee en cuanto a seguridad, son los recursos limitados, los nodos funcionan con batería y tienen potencia computacional y tamaño de memoria. Por lo que es posible que los nodos no sean resistentes a la manipulación. (Farahani, 2008, p.23).

5. Metodología.

5.1 Equipos e instrumentos

A continuación, se muestra los equipos e instrumentos más importantes, utilizados en este proyecto investigativo.

5.1.1 Módulo Xbee

Los módulos Xbee, son dispositivos que son utilizados para enviar y recibir información o datos, de forma inalámbrica, basado en el protocolo IEEE 802.15.4. fabricados por DIGI, los cuales, funcionan dentro de varios tipos de topologías de red, lo pueden hacer de diversas maneras, es decir de coordinador, router y end device. (Faludi, 2011).

Los módulos están diseñados para operar mediante la interfaz UART permitiendo que cualquier microcontrolador o microprocesador, empiece a utilizar los servicios del protocolo ZigBee, respaldar además las necesidades únicas de las redes de sensores inalámbricos de baja potencia y con un costo bajo. Estos módulos trabajan dentro de la banda de secuencia ISM (bandas de radio industriales, científicas y médicas) de 2,4 GHz. (DIGI, 2022).

Figura 7.

Módulo Xbee Pro S2B



El módulo Xbee utilizado, estará trabajando en modo de operación transparente. Este modo de transmisión es el más sencillo, puesto que no requiere de una mayor configuración, viene en los módulos Xbee como modo de funcionamiento predeterminado. En este modo, los módulos simplemente actúan como un reemplazo de la línea serial por cable. (Eady, 2007).

Se destaca que el envío de datos en este modo de operación se lo realizará en forma de trama serie UART, en el que por lo general se transmite la información en formato ASCII. (Rohde y Schwarz, 2021). Esta trama consta de algunos elementos:

- Un bit de inicio, el cual indica que comienza la trama de datos.
- De 5 a 9 bits de datos, que contienen específicamente la información a transmitir.
- Un bit de paridad, este es opcional, pero permite detectar errores que pueden existir en la transmisión.

- Un bit por lo general indicara el final de la trama de datos.

Figura 8.

Trama de datos UART



Nota. Understanding UART, Rohde y Schwarz, 2021. <https://www.rohde-schwarz.com/>

Para realizar la comunicación en el presente proyecto de titulación, se ha utilizado el módulo Xbee S2B, en su versión PRO, con una entrada para su antena de tipo RPSMA, el módulo cuenta con ciertas características que se mencionan por parte de (DIGI, 2022), en la Tabla 2.

Tabla 2.

Características del módulo Xbee Pro S2B

Características del módulo Xbee	
Rango de visibilidad de RF para interiores	90 m (300 ft), variación internacional de 60 m (200 ft).
Rango de visibilidad directa de RF para exteriores	3200 m (2 miles), variación internacional de 1500 m (5000 ft).
Corriente límite de entrada	205 mA – 220 mA (@3,3V), variante internacional 117 mA – 132 mA (@3,3V)
Corriente límite de salida	47 mA – 62 mA (@3,3V)
Transmisión de potencia de salida	63 mW (18 dBm, variación internacional de 10mW (10 dBm)
Operación de banda de frecuencia	2,4 GHz
Voltaje de alimentación	2.7 V - 3.6 V
Rango de temperatura	-40 a 85 °C
Antena	Frecuencia de 2,4 GHz, ganancia de 10 DBi

Nota Zigbee RF Modules, DIGI,2022. <https://www.digi.com>

5.1.2. Adaptador Xbee Shield

La separación de los pines de los módulos Xbee, se encuentran bajo el sistema métrico (mm), mientras que el protoboard se basa en el sistema inglés (pulgadas). Por lo que existen adaptadores, para que los módulos Xbee puedan acoplarse al protoboard. Además, funciona a un voltaje de 3.3 V a 5 V (Fortuño, 2012).

Figura 9.
Adaptador Xbee Shield



5.1.3. Adaptador Xbee Explorer

Para poder leer los datos del módulo Xbee a la computadora, es necesario de un adaptador, es útil este adaptador Explorer para poder configurar el módulo según lo que se requiera realizar, es decir configurar el modo de operación que va a trabajar el Xbee, o si este debe de ser un coordinador, router o un objeto final, esto se hará en conjunto con el software X-CTU.

Figura 10.
Adaptador Xbee Explorer



5.1.4. Sensores utilizados

En la Tabla 3 se muestra las características de los sensores que se han utilizado dentro de la red de comunicación V2V. Las características descritas, han sido tomadas de sus respectivos Datasheets, que se los puede apreciar en la parte de Anexos (anexos 9-13).

Tabla 3.
Descripción de sensores utilizados

Descripción	Sensores utilizados				
	Sensor aceleración	Sensor luminosidad	Sensor de proximidad	Sensor de temperatura y humedad	Sensor GPS
Modelo del sensor	ADXL345	BH1750FVI	TF Mini Plus	DHT11	NEO 6M
Voltaje de operación	1.8 V a 3.6 V	2.4 V a 3.6 V	5 V \pm 0.5 V	3.0 V a 5.5 V	3.6 V a 5 V
Corriente de operación	25 a 130 μ A	120 μ A a 190 μ A	\leq 100 mA	0.5 mA a 2.5 mA	100 mA
Rango de Temperatura	-40 °C a 85 °C	-40 °C a 85°C	-20 °C a 60 °C	0 °C a 50 °C	-40 a 85 °C
Interfaz de comunicación	I2C	I2C	UART	Analógico	UART
# Pines	Cuatro pines	Cuatro pines	Cuatro pines	Tres pines	Cuatro pines
Información adicional	Se detecta la aceleración en sus tres ejes (X, Y, Z).	Rango detección de luz ambiental (1-65535 lx).	Rango de operación de 1 a 12 m.	Rango de medición para temp de 0-50°C y para la hum de 20 a 90 %.	Sensibilidad de navegación de -161 dBm y con una precisión de 0.1 m/s.

Adicional a estos elementos empleados, también se hizo uso de otros elementos secundarios como son: pantalla LCD, placa de pruebas (protoboard), leds, zumbadores, interruptores, cableado, entre otros. Para conectar todos los sensores y los elementos electrónicos se utilizó el microcontrolador Arduino Mega 2560, y el software Arduino IDE. Para el desarrollo de los algoritmos de programación, diseño e implementación se empleó los softwares Fritzing, Cisco Packet Tracer, y XCTU, que se los describirá a continuación.

5.1.5. Fritzing

Es un software de código abierto, el cual permite crear y diseñar circuitos electrónicos. Tiene una interfaz sencilla e intuitiva, se complementa con otras partes como Arduino y Processing. (Universidad Miguel Hernández, 2014).

Fritzing es un referente en el diseño electrónico libre, permite a diseñadores, investigadores o aficionados, crear sus propios prototipos basados en Arduino y crear los esquemas de circuitos impresos, para posteriormente fabricarlos. (Universidad Miguel Hernández, 2014).

5.1.6. Cisco Packet Tracer

Es una herramienta integral de aprendizaje y enseñanza de tecnología de redes, ofreciendo la combinación única de experiencias de simulación y visualización realista. Permite a los estudiantes y docentes diseñar crear, configurar y solucionar problemas de redes complejas mediante equipos virtuales. Finalmente impulsa a los estudiantes a explorar conceptos, realizar experimentos y comprender mejor la creación de redes. (Cisco, 2020).

Cisco Packet Tracer permite a los estudiantes el poder diseñar redes complejas y grandes, puesto que en la mayoría de los casos no es factibles realizar con hardware físico, debido a los grandes costes que derivan. (Tokio School, 2021).

5.1.7. XCTU

El software XCTU es una multiplataforma diseñada para que los desarrolladores puedan interactuar con los módulos RF de Digi, por medio de una interfaz gráfica de fácil utilidad. Así mismo se incluye herramientas que facilitan la configuración, instalación y lo que es la prueba y simulación de los módulos. (DIGI, 2022)

Este software es el oficial que permitirá trabajar con módulos XBee, y poder configurarlo y simularlos, así mismo se puede gestionar y configurar varios dispositivos de radiofrecuencia, e inclusive los conectados de forma remota, es decir por el aire. (Faludi, 2011, p.17).

5.2 Protocolos de comunicación vehicular y tecnología V2V en el vehículo

Para obtener información sobre los protocolos de red y de la tecnología desarrollada en la actualidad en los vehículos, se utilizó una metodología cualitativa, la cual permite analizar y recopilar información necesaria y pertinente. La información reunida se la obtuvo de fuentes confiables, de artículos científicos, de revistas IEEE (Asociación científica internacional de técnicos e ingenieros), revistas SAE (Sociedad de Ingenieros de Automoción), artículos científicos de Springer, la revista ScienceDirect, de diferentes libros y de páginas web confiables. Una vez obtenida la información se analiza y se selecciona las referencias más importantes.

5.3 Construcción del prototipo de comunicación V2V

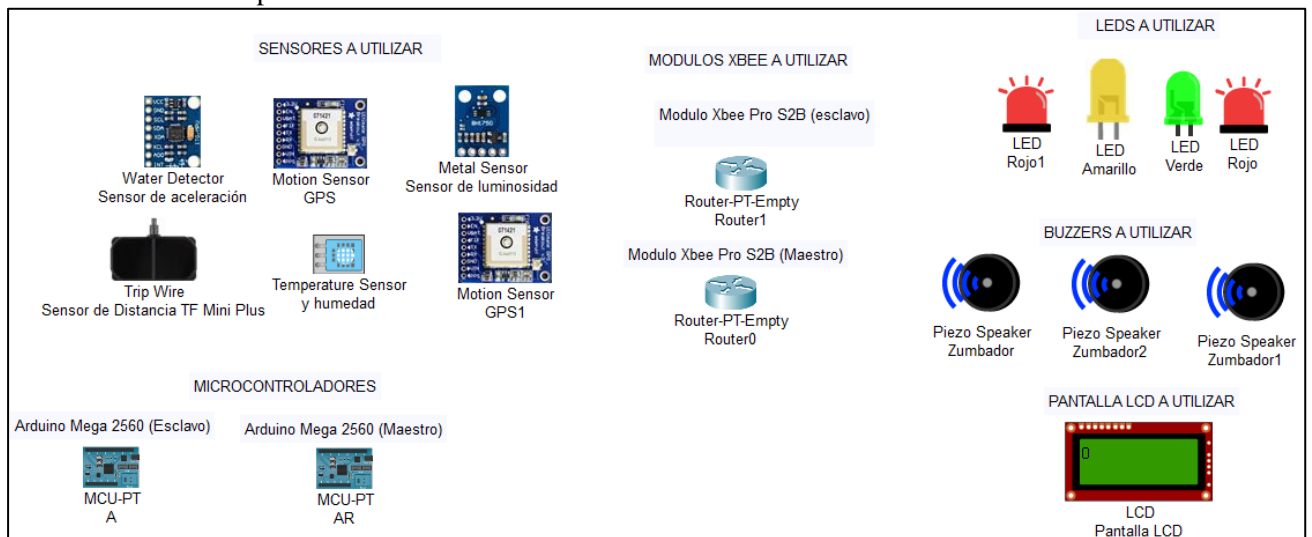
Para la construcción del prototipo de comunicación V2V, se desarrolló en dos fases, en la primera, consistió en el diseño la red de información, mientras que en la segunda se enfocó en el diseño de las conexiones de los sensores y elementos a utilizar.

5.3.1 Diseño de la red inalámbrica

Para el diseño de la red, se utilizó el software Cisco Packet Tracer, gracias a esta interfaz, se logró simular la red de transmisión de datos y verificar el intercambio de los paquetes de información. Para este proyecto de tesis se realizó solo lo que tiene que ver exclusivamente con diseño, además, se ha empleado una topología tipo árbol. La red consta de varios elementos como lo son sensores (sensor de distancia, sensor de aceleración, módulo GPS, sensor de temperatura, humedad y sensor de luminosidad), pantalla LCD, diodos led, zumbadores y módulos Xbee (Xbee Maestro, Xbee esclavo).

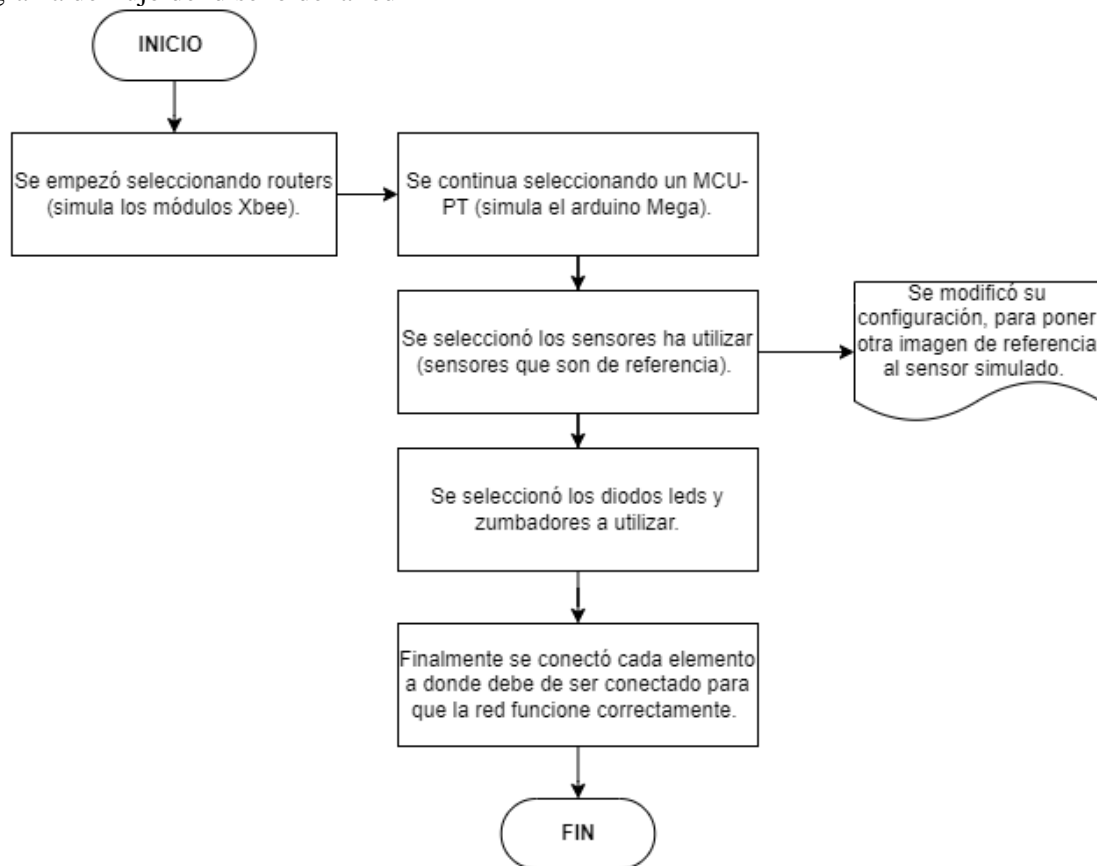
Para el diseño de la red en Cisco, se ha ocupado otros elementos para diseñar, puesto que los elementos que se necesitaban no existen en el programa, pero cada elemento está descrito, como se observa en la Figura 11.

Figura 11.
Elementos utilizados para el diseño de la red inalámbrica V2V



El diseño de la red se explica mediante un diagrama de flujo que se presenta a continuación

Figura 12.
Diagrama de flujo del diseño de la red



5.3.2 Conexión de sensores a utilizar

Con la ayuda del software Fritzing, se diseñó la conexión de cada uno de los componentes a utilizar, los cuales son: el sensor de temperatura y humedad, sensor de luminosidad, sensor de aceleración, módulo GPS, esta información es procesada mediante un Arduino Mega 2560 (esclavo). Igualmente se diseñó la conexión del sensor de distancia, y módulo GPS, los cuales se encuentran conectadas en el Arduino Mega 2560 (maestro). Adicional se realizó el diseño de las conexiones de la pantalla LCD y del módulo Xbee. No se realiza la conexión directamente de los sensores al módulo Xbee, puesto que el Xbee, solo permite la conexión de señales analógicas y digitales, y los sensores utilizados funcionan por medio de la interfaz I2C o UART.

- **Sensor de humedad y temperatura**

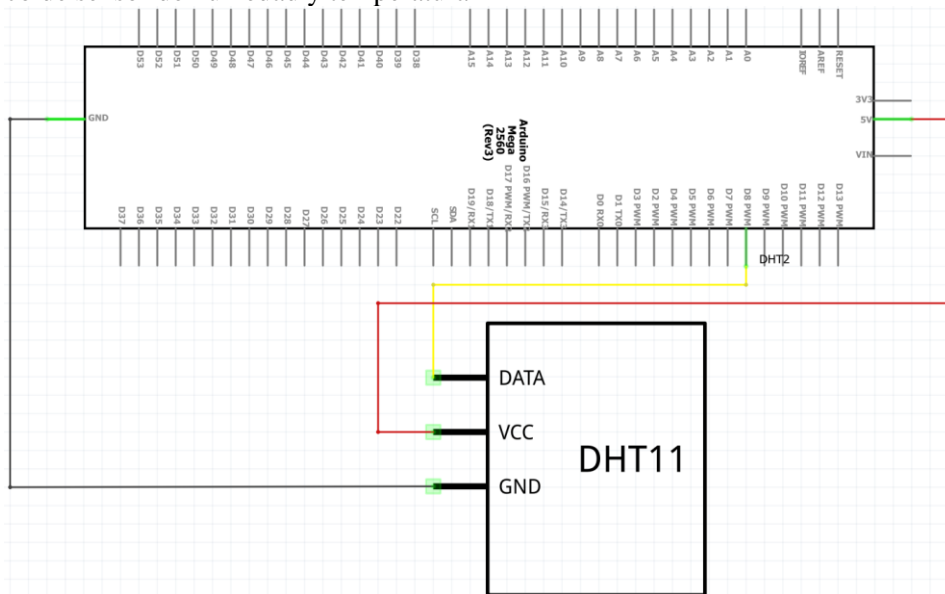
Para la obtención de la humedad y de la temperatura, se hizo uso del sensor DHT11. La información de la temperatura, como de la humedad son de importancia al momento de conducir, debido a que, si existe valores muy bajos o muy altos, provoca que el conductor pierda capacidad de reacción, se disminuya lo reflejos o se empiece a sentir fatiga o cansancio, es decir

el confort se ve afectado. Este sensor consta de tres pines los cuales están conectados de la siguiente forma:

- Un pin a corriente 5V
- Un pin a tierra
- Un pin conectado a una entrada digital (8), en Arduino Mega 2560 (esclavo).

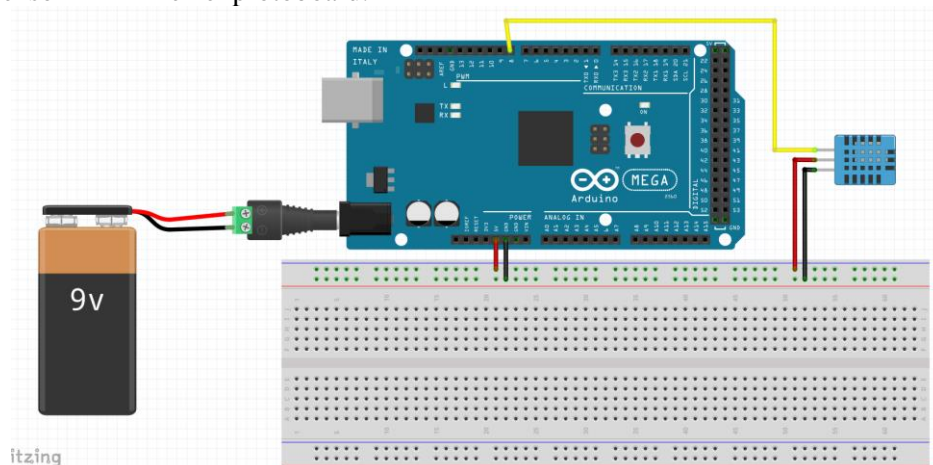
Para una mejor comprensión de la conexión se presenta un diagrama eléctrico realizado en el software Fritzing.

Figura 13.
Circuito eléctrico de sensor de humedad y temperatura DHT11



Asimismo, se ha realizado el conexionado en el protoboard como representación gráfica, igualmente en el software Fritzing, el cual se lo puede apreciar en la siguiente Figura 14.

Figura 14.
Conexión del sensor DHT11 en el protoboard.



- **Sensor de aceleración**

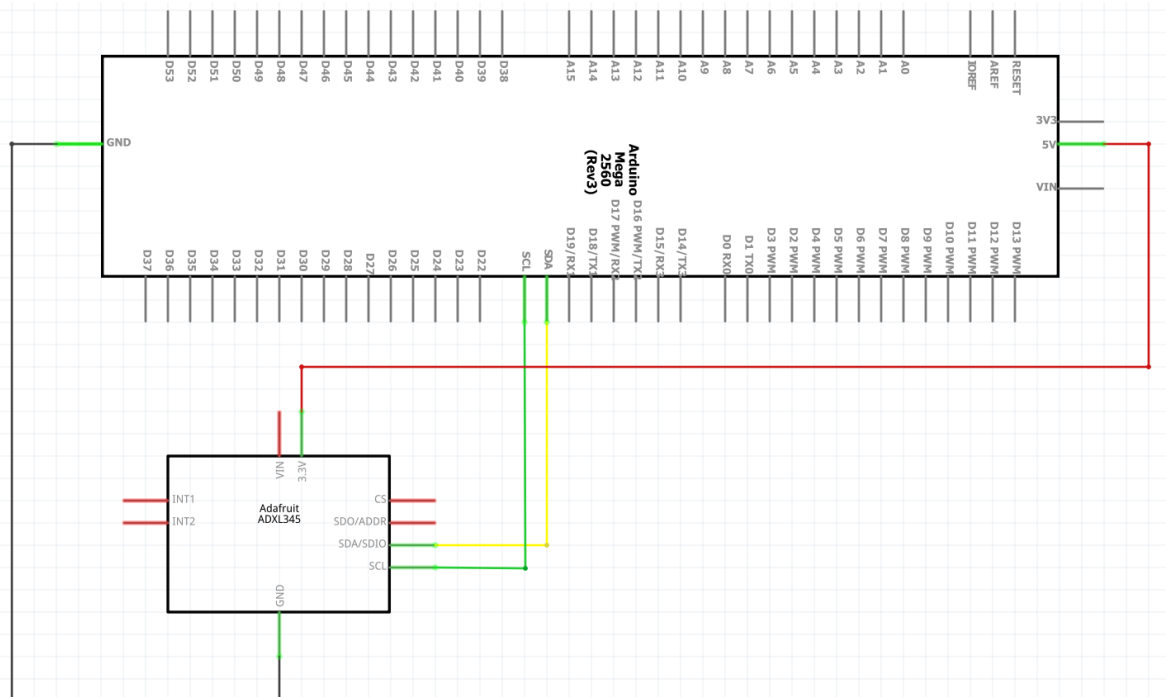
Para medir la aceleración y desaceleración del vehículo, se ha implementado el sensor de aceleración ADXL345. El sensor de aceleración es importante, en la presente investigación, puesto que, con la información obtenida de este sensor, se podrá alertar al conductor, para prevenir posibles accidentes, ya que, si existe un bache, derrumbe, falla geológica en la vía, el conductor comenzará a desacelerar, y con este dato de desaceleración, se le informará al otro conductor sobre posibles peligros en la vía.

Este sensor cuenta con ocho pines, de los cuales se ocuparán cuatro, los cuales se describen a continuación

- Un pin SDA conectado a SDA (20) en Arduino Mega 2560 (esclavo).
- Un pin SCL conectado a SCL (21) en Arduino Mega 2560 (esclavo).
- Un pin VCC conectado a corriente 5V.
- Un pin GND conectado a tierra.

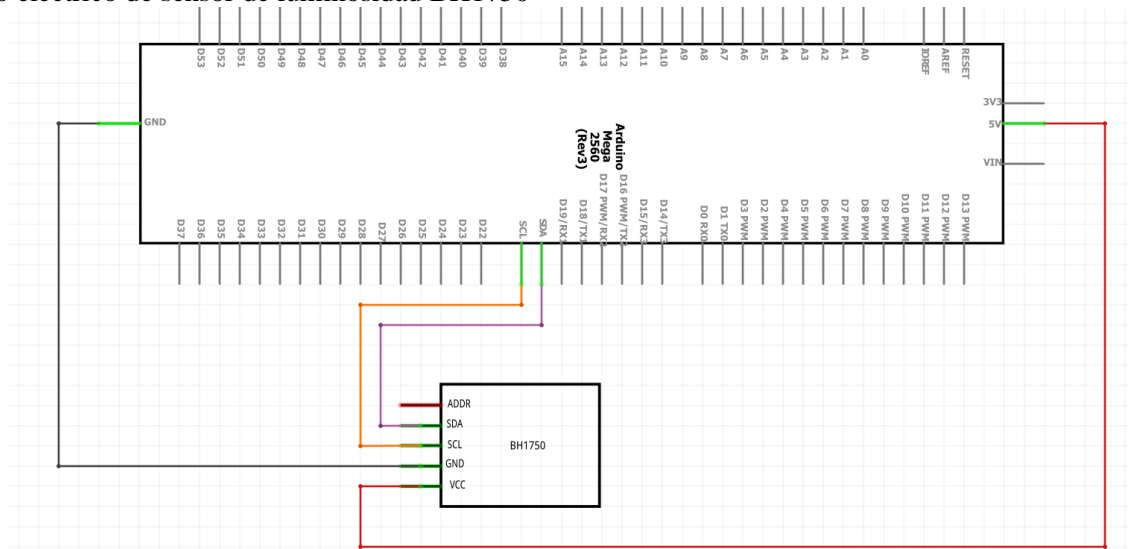
En la Figura 15 se presenta el diagrama eléctrico de la conexión pertinente del sensor de aceleración, diagrama realizado en el software Fritzing.

Figura 15.
Circuito eléctrico de sensor de aceleración ADXL345



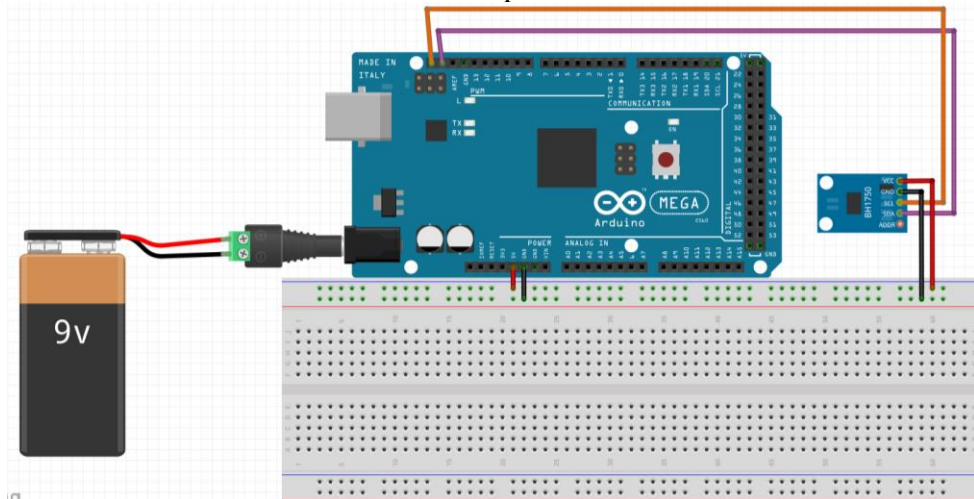
A continuación, en la Figura 16, se presenta la conexión del sensor de aceleración realizado en el software Fritzing.

Figura 17.
Circuito eléctrico de sensor de luminosidad BH1750



Posteriormente, se presenta la conexión del sensor de luminosidad realizado en el software Fritzing.

Figura 18.
Conexión del sensor de luminosidad BH1750 en el protoboard



- **Sensor de distancia**

Se ha utilizado el sensor de distancia TF Mini Plus este permite detectar objetos que se encuentran cercanos al elemento sensor, por lo que para este proyecto se ha utilizado para mantener una distancia segura entre los dos vehículos.

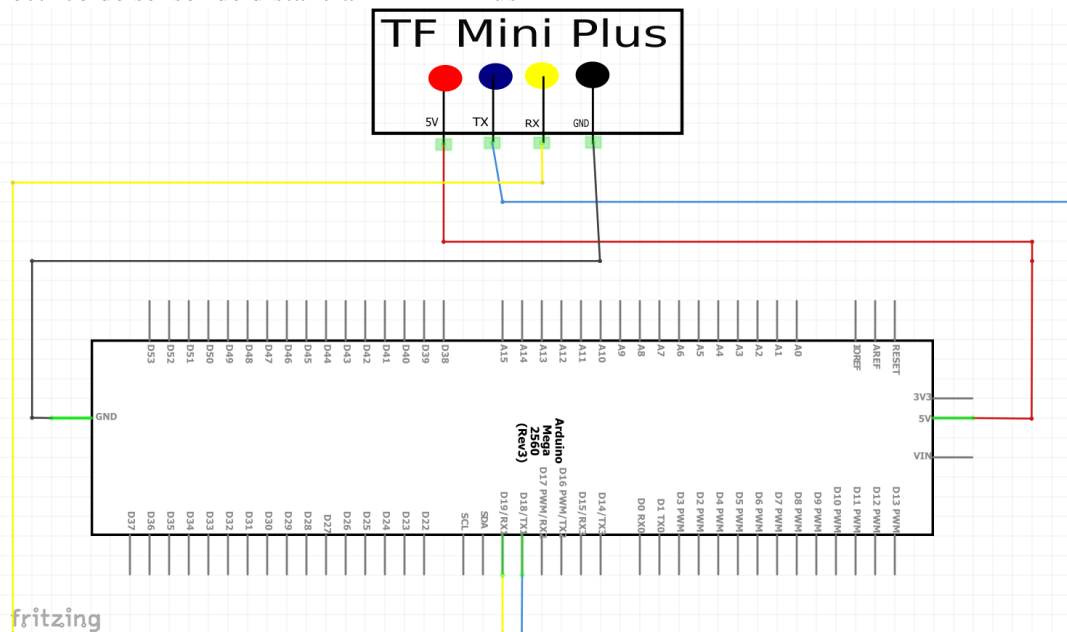
El sensor de distancia cuenta con cinco pines, a continuación, se describen los cuatro utilizados:

- Un pin RX conectado a RX en Arduino (19) en Mega 2560 (maestro).
- Un pin TX conectado a TX en Arduino (18) en Mega 2560 (maestro).

- Un pin VCC conectado a corriente 5V.
- Un pin GND conectado a tierra.

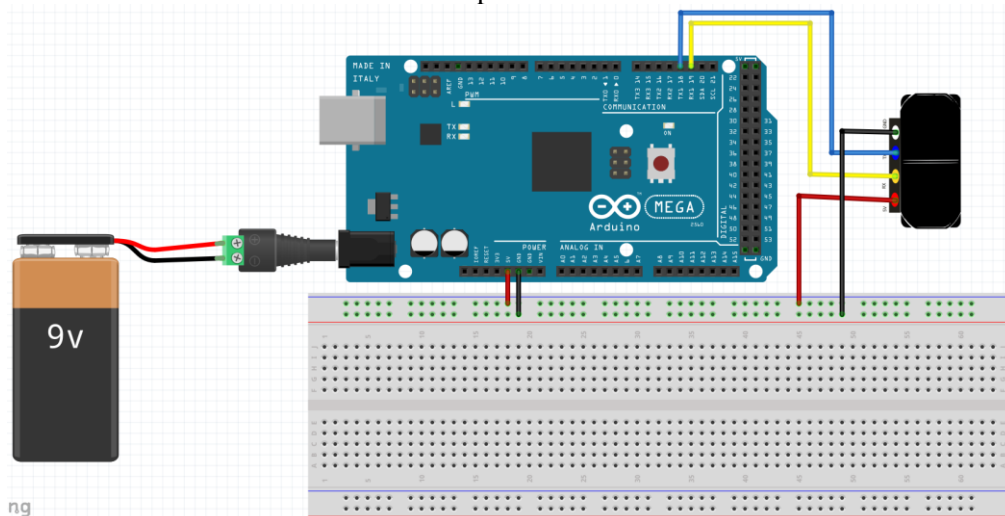
Se presenta un diagrama eléctrico de la conexión del sensor de distancia, realizado en el software Fritzing.

Figura 19.
Circuito eléctrico de sensor de distancia TF Mini Plus



Se ha realizado el conexionado en el protoboard del sensor de distancia, presentado en la Figura 20.

Figura 20.
Conexión del sensor de distancia TF mini Plus en el protoboard



- **Módulo GPS**

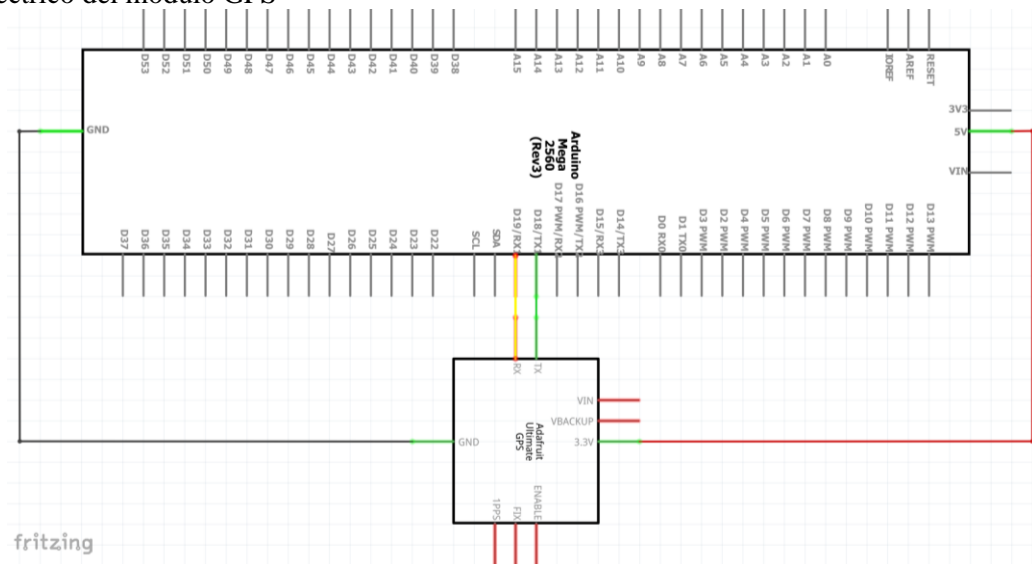
Con el fin de dar aviso al conductor al momento de entrar en una intersección, sobre posibles vehículos acercándose, se utilizó el módulo GPS, en el que se obtiene las coordenadas

(longitud y latitud), de los dos vehículos implicados en la intersección, se compara las coordenadas para saber que se aproxima un vehículo, por tal motivo cada automóvil contará con su propio GPS. El módulo GPS cuenta con cuatro pines:

- Un pin RX conectado a RX en Arduino (17) en Mega 2560 (esclavo o maestro).
- Un pin TX conectado a TX en Arduino (16) en Mega 2560 (esclavo o maestro).
- Un pin VCC conectado a corriente 5V.
- Un pin GND conectado a tierra.

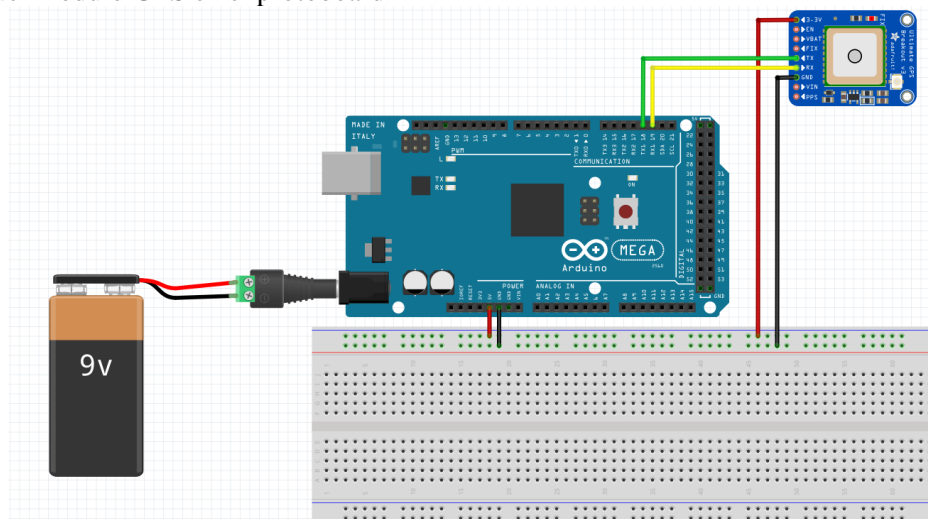
Se presenta un diagrama eléctrico de la conexión adecuada del módulo GPS, realizado en el software Fritzing.

Figura 21.
Circuito eléctrico del módulo GPS



Se ha realizado el conexionado en el protoboard del sensor de distancia, presentado en la Figura 22.

Figura 22.
Conexión del módulo GPS en el protoboard



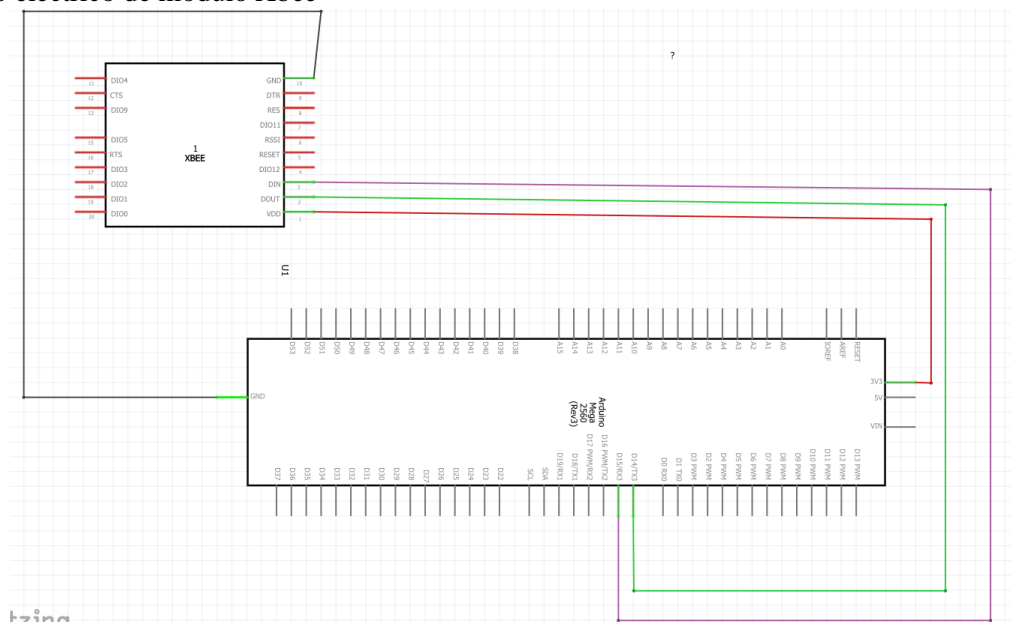
Además, en esta etapa se muestra las conexiones realizadas del módulo Xbee, como también de la pantalla LCD 20x4

- **Módulo Xbee**

En la siguiente Figura 23 se muestra el diagrama eléctrico de la conexión realizada de los módulos Xbee. La conexión de los módulos Xbee tanto maestro como esclavo, será la misma, pero conectados en diferentes Arduinos Mega 2560 y en diferentes puertos seriales.

Figura 23.

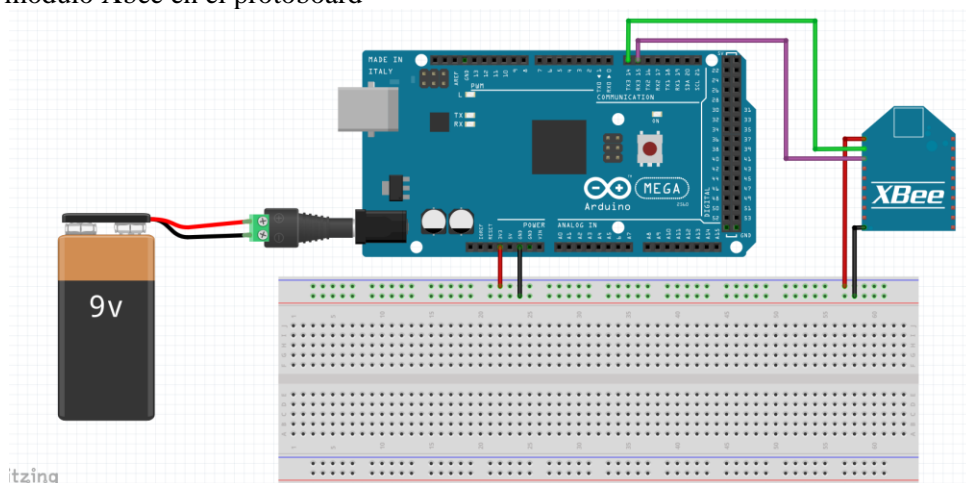
Circuito eléctrico de modulo Xbee



Además, se realizó el conexionado en el protoboard del módulo Xbee, presentado en la siguiente Figura 24.

Figura 24.

Conexión del módulo Xbee en el protoboard



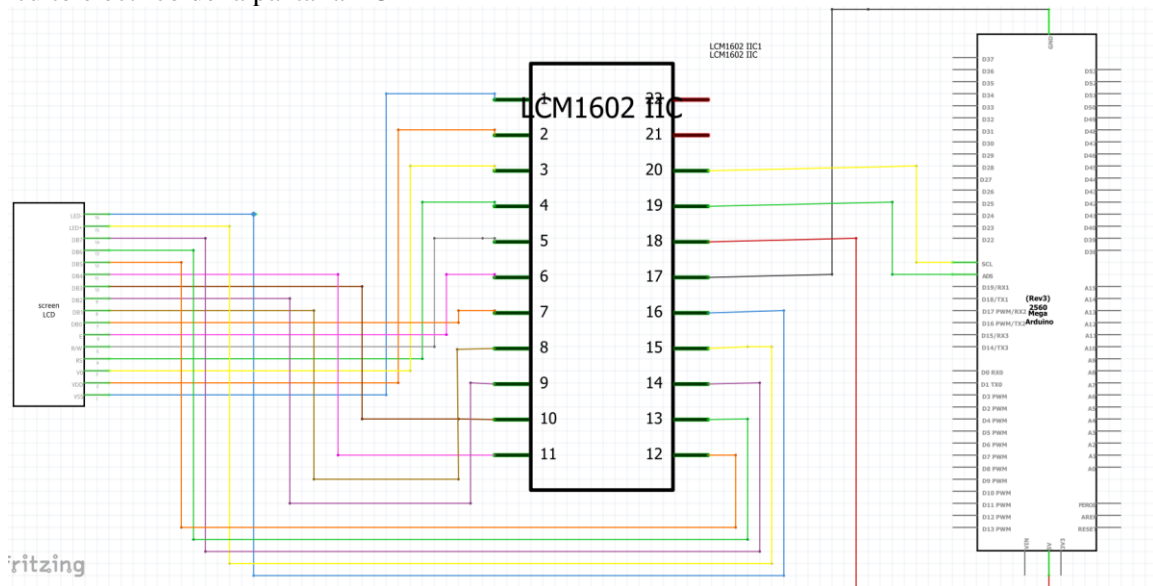
- **Pantalla LCD**

Se destaca que la pantalla cuenta con dieciséis pines, pero se ha conectado un módulo adaptador LCD a I2C, el cual reduce los dieciséis pines a cuatro pines, los cuales son:

- Un pin SDA conectado a SDA del Arduino Mega 2560 (maestro).
- Un pin SCL conectado a SCL del Arduino Mega 2560 (maestro).
- Un pin VCC conectado a corriente.
- Un pin GND conectado a tierra.

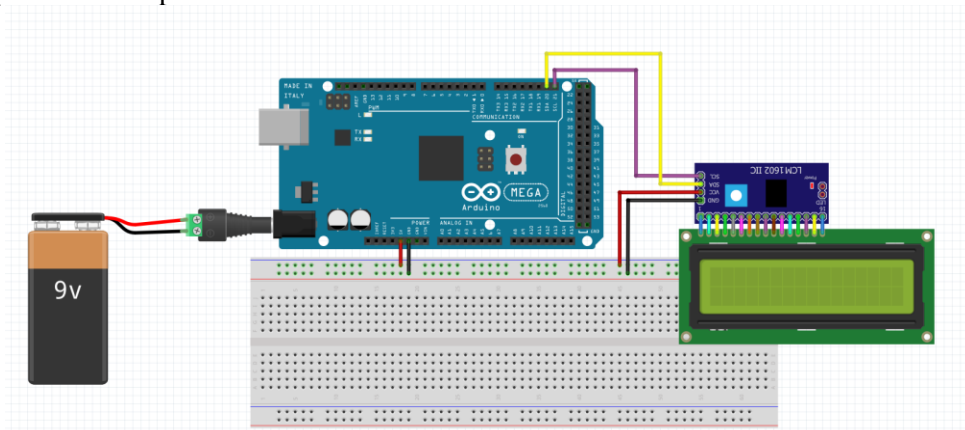
En la Figura 25, se presenta la conexión eléctrica de la pantalla LCD 16X2 que tiene similar configuración que la pantalla LCD 20x4, el esquema eléctrico fue realizado en el software Fritzing.

Figura 25.
Circuito eléctrico de la pantalla LCD



Se realiza la simulación de la conexión de la pantalla LCD junto al Arduino, realizada en el protoboard, la cual se puede observar en la siguiente Figura 26.

Figura 26.
Conexión de pantalla LCD por interfaz I2C



5.4 Implementación de la red de comunicación y visualización de datos

Se dividió esta sección en tres etapas, la primera que respecta a la configuración de los módulos Xbee, la segunda acerca de la programación realizada en Arduino, y en la tercera etapa

se realizó la implementación del prototipo de comunicación en los vehículos, junto con la visualización de los datos procesados en la red de comunicación inalámbrica.

5.4.1 Configuración de módulos Xbee

En esta etapa, se llevó a cabo la conexión y programación de dos módulos Xbee Pro S2B, para transferir los datos dentro de la red. En el que se configuró un módulo Xbee como maestro y otro módulo Xbee como esclavo.

Para realizar la configuración de los Xbee, primero se necesita tener instalado el software XCTU, además se utilizó el “Xbee Shield Explorer”, en el que se insertará el módulo Xbee, este adaptador permitirá conectar a un puerto USB a la computadora.

Figura 27.

Adaptador de módulo Xbee



Nota. En la figura, se observa al módulo Xbee, colocado sobre el adaptador USB.

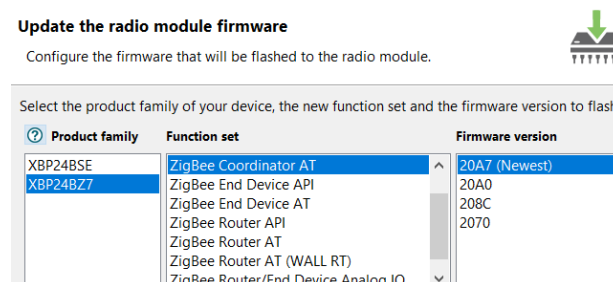
A continuación, se procedió a ejecutar el programa XCTU, y se dará un clic en descubrir los módulos conectados a la computadora, se solicita seleccionar parámetros básicos con los que va a trabajar el módulo, los cuales se deja como viene por *default*. Luego de esto se configuró cada módulo por separado.

- **Configuración de módulo Xbee como coordinador**

A continuación, como se observa en la Figura 28, hay que añadir un firmware para que el módulo funcione correctamente, en este caso el módulo utilizado es de la familia XBP24BZ7, la función que va a tener este módulo es de coordinador, para este proyecto de tesis, se utilizó el modo transparente y se escoge la versión más reciente del firmware.

Figura 28.

Descarga de firmware para Xbee Coordinador



Luego se desplegará una serie de configuraciones a realizar en el módulo Xbee, pero los datos que se modificaron son en la sección de *Networking* el “*PAN ID*”, el cual se ha elegido colocar *12345*. Además, en el *Destination Address Low*, el cual será las posibles direcciones que puede tener el módulo, por lo que para esta aplicación se digitó 0, puesto que solo se va a conectar con un módulo Xbee, En el *Node Identifier*, permite dar un nombre al módulo, para identificar de mejor manera, para esta ocasión se modificó con el nombre de “*Coordinador*”.

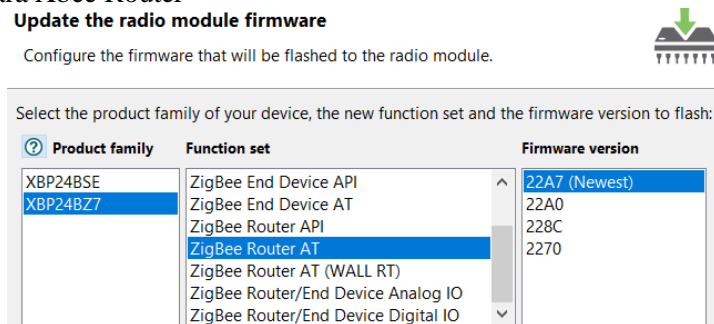
Finalmente se modifica valores adicionales, como la velocidad de baudios (Baud Rate), en el que para esta aplicación se ha utilizado 115200 baudios. Además de este valor se modifica la frecuencia de muestreo que se lo ha realizado cada 1000 ms o 3E8 en hexadecimal.

- **Configuración de módulo Xbee como esclavo**

Con lo que respecta al configurar el esclavo o router lo primero que se realizó, es añadir el firmware necesario para el enrutador, al igual que el módulo coordinador, es de la misma familia y trabaja igualmente en modo AT, este proceso realizado se lo puede observar mejor en la siguiente Figura 29.

Figura 29.

Descarga de firmware para Xbee Router

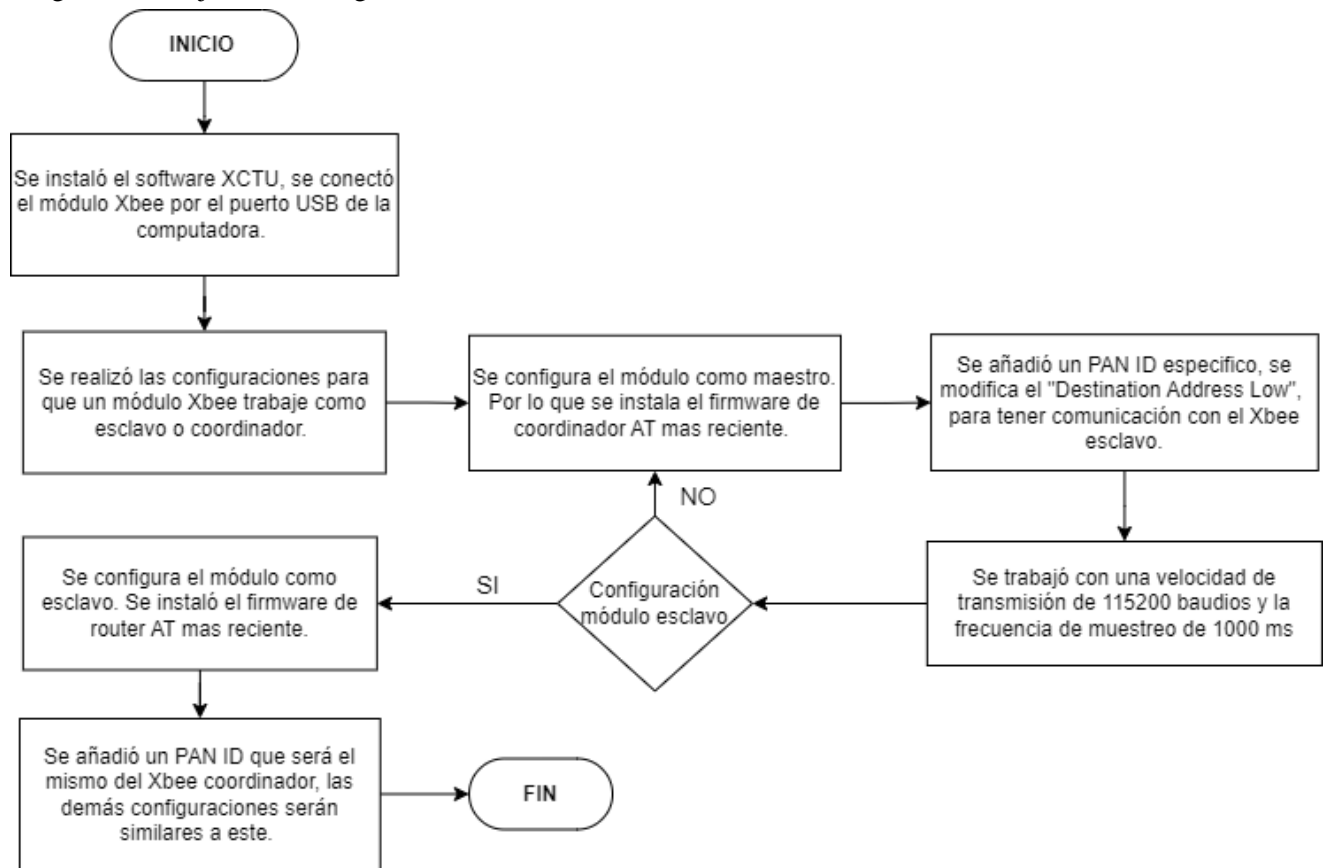


Las otras configuraciones realizadas al módulo Xbee router, serán similares a las configuraciones ya descritas anteriormente del coordinador. Salvo el nombre con el que se lo identificó será “Router”.

A continuación, se presenta un diagrama de flujo resumiendo la configuración de los módulos Xbee.

Figura 30.

Diagrama de flujo de la configuración de módulos Xbee



Lo que respecta a la parte de hardware, el módulo Xbee configurado como esclavo se encuentra conectado al puerto serial del Arduino Mega (esclavo), funciona como recepción de los datos por los sensores, para posteriormente poder ser transmitidos por radiofrecuencia en una banda de 2,4 GHz y en modo Transparente al módulo maestro.

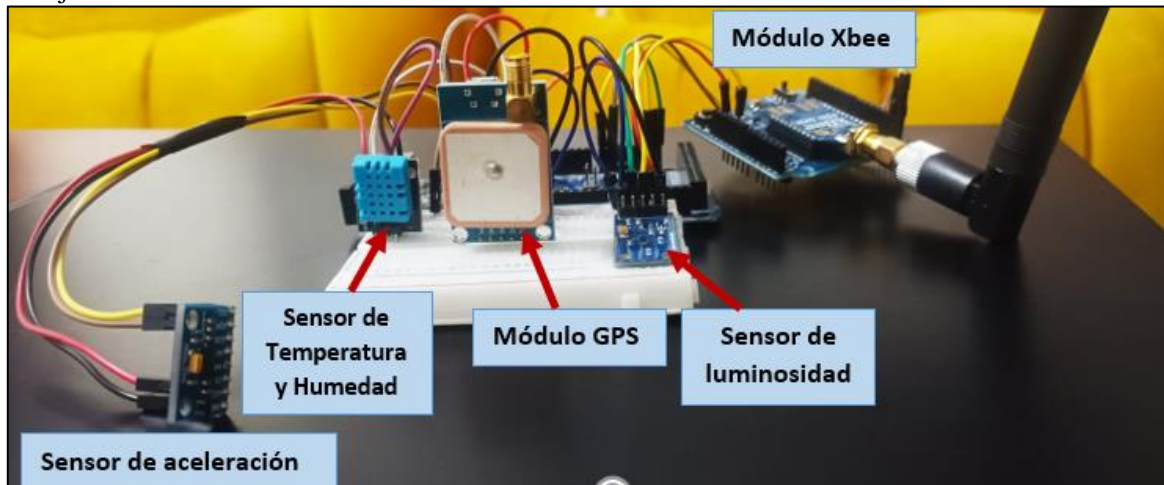
En cuanto al módulo Xbee maestro o en su defecto llamado también como coordinador, está conectado por puerto serial al Arduino Mega (maestro), es el encargado de recibir la información dada por radiofrecuencia por parte del módulo Xbee esclavo. Además, el coordinador para recibir información, analiza y verifica que el Xbee esclavo que envía información, tenga el mismo PAN ID con el que ha sido configurado el coordinador. Todo esto se realiza con la finalidad de que la información recibida sea confiable y segura.

5.4.2 Programación en Arduino

Una vez ejecutado el diseño de la red de comunicación y de las conexiones a realizar, se realizó la conexión con el Arduino, el cual cuenta con varios puertos -seriales y puertos de interfaz I2C, además de varios pines de entradas digitales y analógicas, puntos necesarios para llevar a cabo el prototipo de comunicación. A continuación, se observa en la Figura 31 el montaje de los sensores en el Arduino esclavo.

Figura 31.

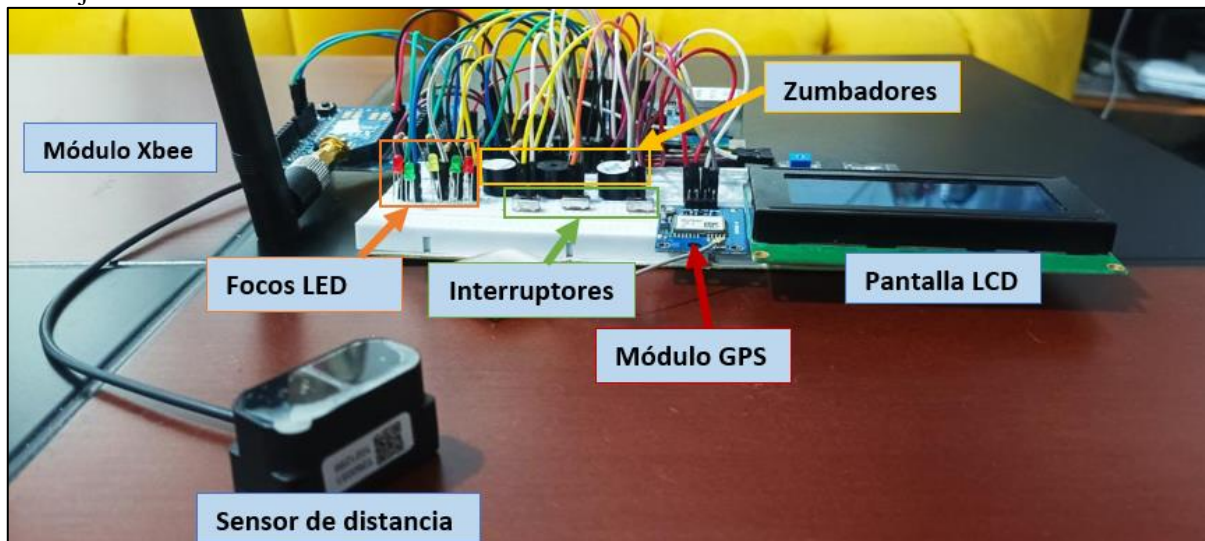
Montaje de sensores en Arduino esclavo



De igual manera, en la Figura 32 se observa el montaje de sensores en el Arduino maestro.

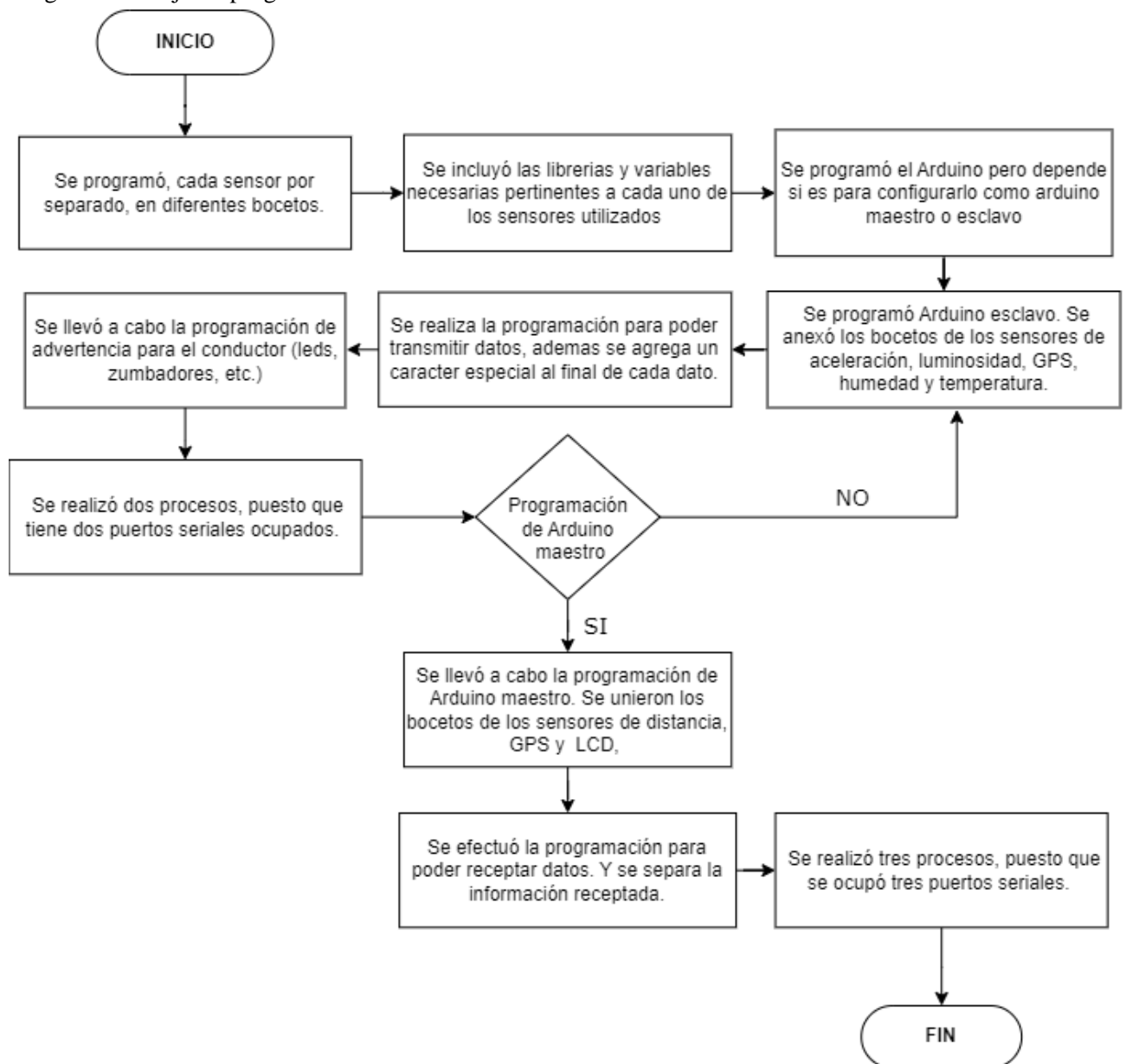
Figura 32.

Montaje de sensores en Arduino maestro



Además, se llevó a cabo toda la programación de la red de comunicación en el software Arduino IDE, el cual es un entorno de desarrollo integrado, conformado por un grupo de herramientas de programación, muy útil para la programación y utilización con la placa Arduino. Cada sensor programado tendrá sus unidades de medición. Es decir, para el sensor de temperatura y humedad será en grados centígrados y porcentaje respectivamente, para el sensor de aceleración su unidad en m/s^2 , el sensor de luminosidad en luxes, el sensor de distancia en centímetros y finalmente el módulo GPS se ha obtenido la latitud y longitud en formato DD (grados decimales).

Figura 33.
Diagrama de flujo de programación en Arduino IDE



Para comprender la programación y comunicación dentro de la red de comunicación vehicular creada, se ha desarrollado diagramas de bloques (Figura 34), los cuales se detallan a continuación:

Empezando con el bloque del vehículo emisor, el cual constará de un Arduino Mega 2560 (esclavo), al cual estarán conectados los sensores a utilizar, como es el sensor de luminosidad (BH1750), sensor de aceleración (ADXL345), que estarán enlazados mediante interfaz I2C, de igual forma se conecta el sensor de temperatura y humedad (DHT11), al ser un dispositivo digital, se lo conectará a uno de los pines digitales del Arduino. Finalmente se conecta el módulo GPS (6m), mediante puerto serial o conocido también como UART

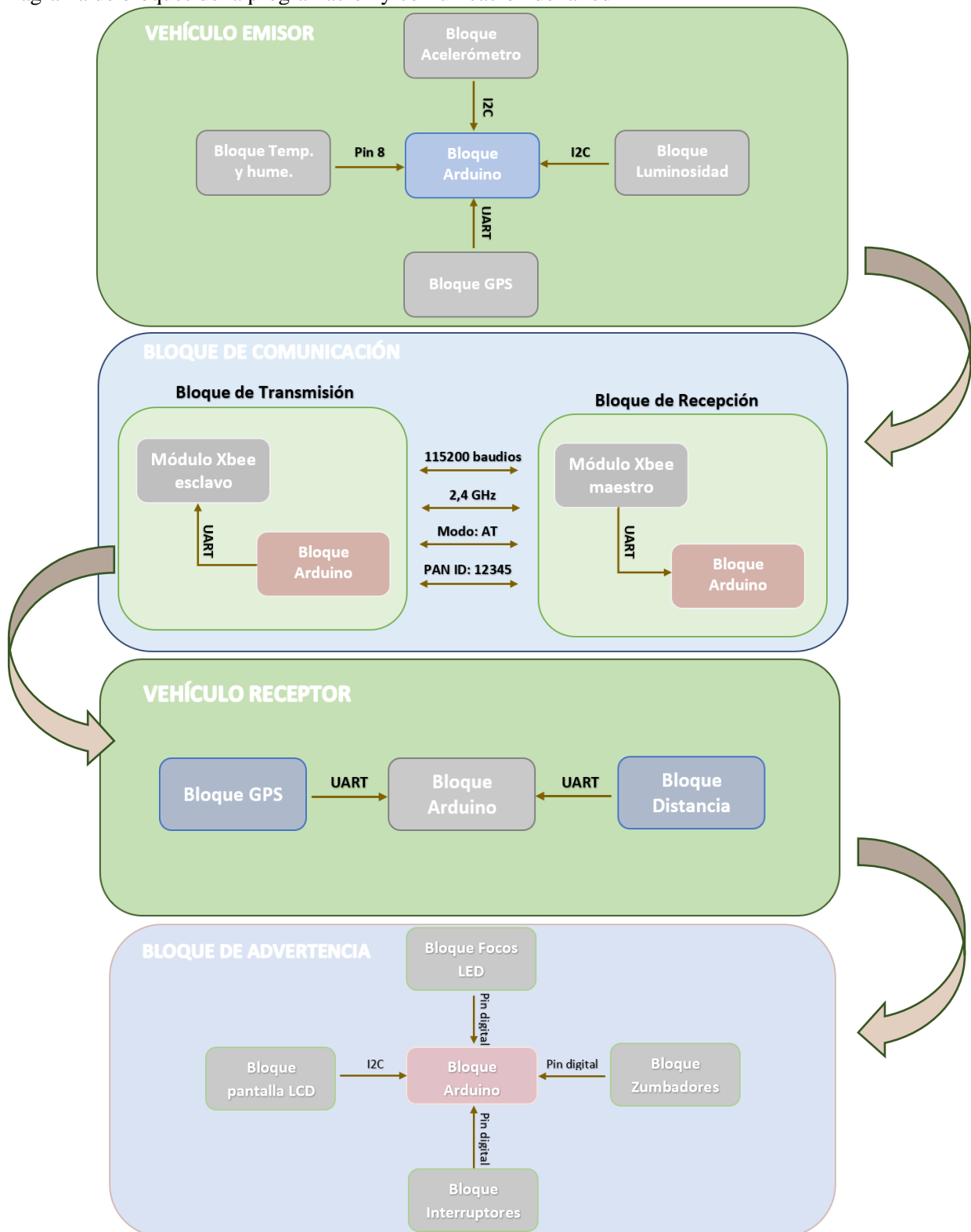
Continuando con el bloque de comunicación, se divide en dos sub-bloques. Empezando con el bloque de transmisión, en el cual el Arduino esclavo estará procesando los datos de los sensores conectados anteriormente. Para transmitir esta información se conecta el módulo Xbee Pro S2B, mediante el puerto serial #3 del Arduino. En la programación se imprime los datos en este puerto serial y además se imprimirá un carácter distinto para cada uno de los sensores. El módulo Xbee, se encontrará configurando como esclavo, en modo de trabajo transparente, y tendrá un PAN ID: "12345", además una velocidad de transmisión de 115 200 baudios y una frecuencia de emisión de 2,4 GHz. Enfatizar que la trama de datos que se está transmitiendo es tipo UART, con un bit de inicio, de 5 a 9 bits de datos, y un bit stop, además puede o no existir un bit de paridad. Mas información se aprecia en la sección 5.1.1

Para el segundo bloque secundario que será el de recepción, se utiliza un módulo Xbee configurado como coordinador, el cual estará receptando la información del Xbee esclavo. El Xbee coordinador estará conectado al Arduino (coordinador) mediante el puerto serial #1. Para que sea útil la información que se está recibiendo, primero se debe de leer la información del puerto serial, a continuación, se debe de separar los datos recibidos y finalmente se debe de transformar los datos que están llegando como cadena de caracteres en valores numéricos, para esto se debe de utilizar distintas funciones en la programación realizada en Arduino. La trama de datos que se recepta, es la misma, que la trama que se esta enviando, sin ninguna modificación.

Prosiguiendo con el bloque del vehículo receptor, constará del Arduino Mega 2560 (coordinador), el cual estará procesando la información del sensor de distancia y del módulo GPS, estos estarán conectados mediante el puerto serial #3 y el puerto serial #2 según corresponde.

Finalmente se describe el bloque de advertencia, el cual constará del Arduino Mega 2560 (coordinador), y de dispositivos de alerta como focos LED, zumbadores, interruptores, los cuales se encuentran conectados por pines digitales, y la pantalla LCD conectada por interfaz I2C. Se utilizará la información que está siendo procesada por el Arduino (coordinador), para emitir diversas precauciones al conductor. No obstante, estas advertencias estarán condicionadas según los valores estipulados para cada una de las funciones que cumplirá el prototipo de comunicación.

Figura 34.
Diagrama de bloques de la programación y comunicación de la red



5.4.3 Implementación del prototipo de comunicación en los vehículos

Se han utilizado dos automóviles para implementar el prototipo de comunicación, un vehículo tipo sedán, precisamente un Nissan-Sunny del año 1993 (esclavo), otro vehículo

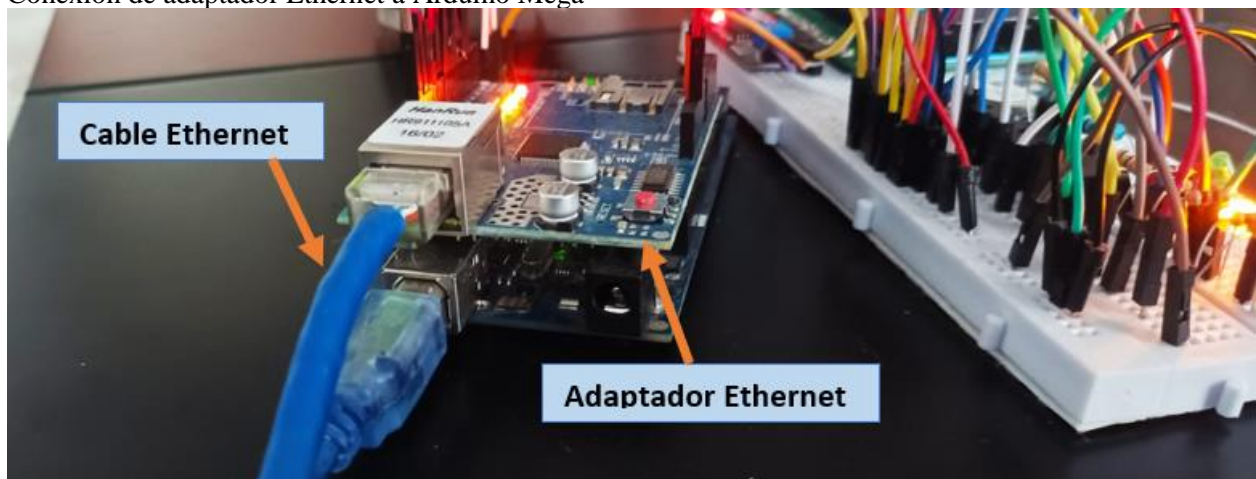
Chevrolet-Blazer del año 1997 (coordinador), no ha sido necesario escoger vehículos modernos, puesto que es un prototipo y los sensores utilizados están sobrepuestos en el vehículo. Para el vehículo coordinador el sensor de distancia se colocará en la parte delantera del automóvil, los demás elementos estarán ubicados sobre el tablero. Para el vehículo esclavo el sensor de aceleración estará montado en la parte lateral del tablero del vehículo, en una posición neutral es decir de 0.00 m/s^2 , obteniendo así una mejor sensibilidad al momento de frenar, los demás elementos y sensores estarán colocados sobre el tablero del auto, en este lugar ubicado no afecta en la obtención de datos de cada sensor. Por último, la corriente para alimentar los sensores y microcontroladores provendrá de la batería del propio vehículo, pero con un regulador de voltaje, regularizado a 6,5 V, para evitar que se llegue a dañar el microcontrolador Arduino.

5.4.3.1 Visualización de datos

La plataforma Ubidots es una herramienta, que permite subir datos a la nube, para poder comunicar esta plataforma con Arduino, se lo podrá realizar por Ethernet o por Wifi, para este proyecto se ha escogido utilizar Ethernet, por lo tanto, se utilizó un adaptador Ethernet, para poder conectar el cable Ethernet, este adaptador será compatible para Arduino Mega. Lo mencionado anteriormente se observa en la Figura 35.

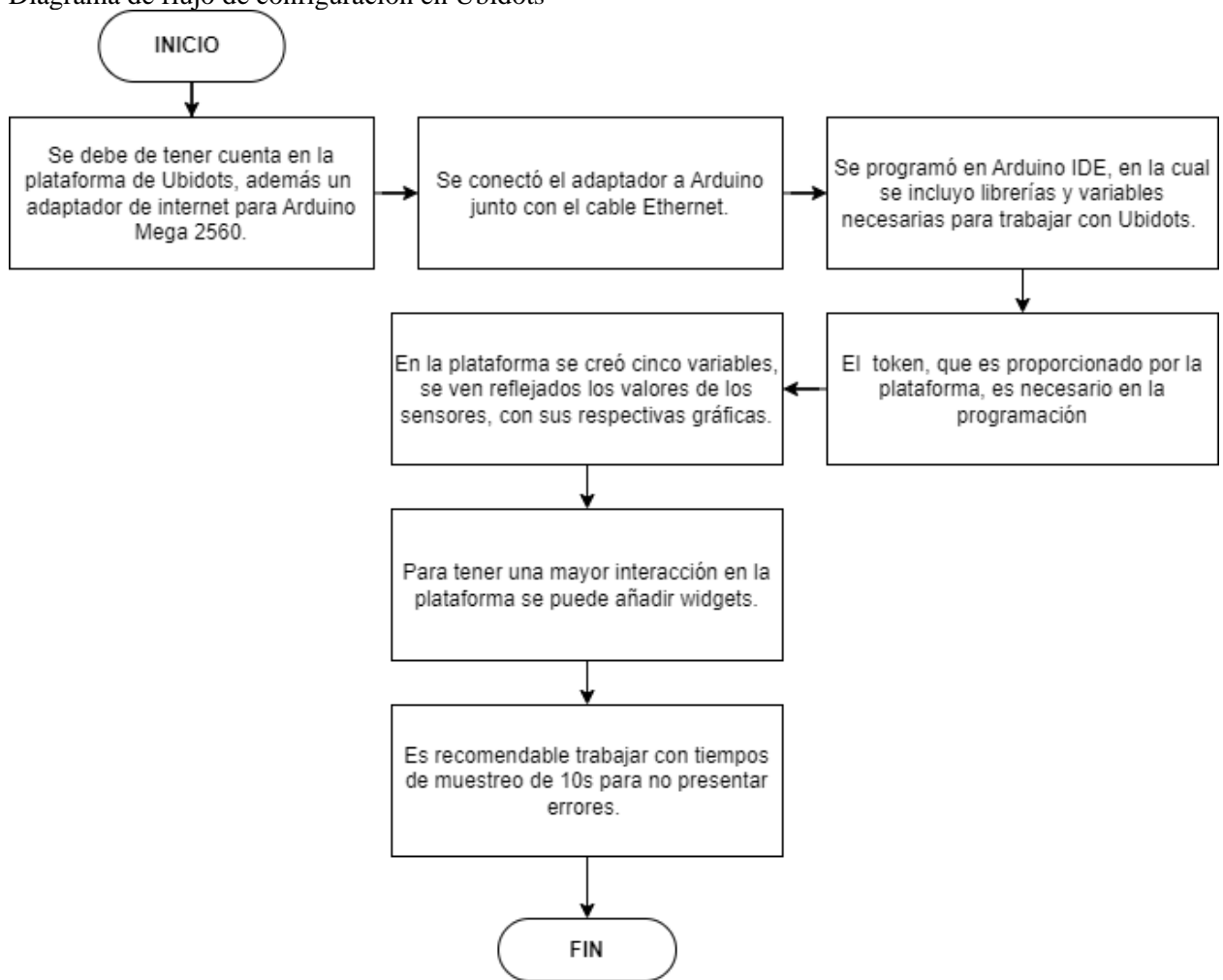
Figura 35.

Conexión de adaptador Ethernet a Arduino Mega



A continuación, se presenta un diagrama de flujo, en el que se resume el proceso de las conexiones para el uso de la plataforma Ubidots con Arduino.

Figura 36.
Diagrama de flujo de configuración en Ubidots



6. Resultados.

Los resultados que se han obtenido, se describen a continuación en tres secciones.

6.1. Protocolos de comunicación y tecnología V2V

De la investigación bibliográfica y la búsqueda de los últimos trabajos en torno a la comunicación V2V, se han obtenido algunos resultados importantes a tener en cuenta, que se presenta a continuación.

Dentro de la comunicación entre vehículos, se conoce a esta comunicación como una red vehicular Ad-Hoc (VANET), para llevar a cabo transmisiones correctas se emplean diferentes estándares y protocolos distintos, a continuación, se mencionan algunos de los protocolos de mayor interés:

- En las comunicaciones de corto alcance (DSRC), o también conocida de manera arbitraria como sistema de acceso inalámbrico en entornos vehiculares (WAVE), el propósito de esta es proporcionar altas transferencias de datos y baja latencia de comunicación, dentro de un área pequeña para redes vehiculares. (Anand et al., 2017, p.16).
- La pila del protocolo WAVE, está compuesta de diferentes capas, pero se destaca un protocolo en específico que se ha desarrollado, el WSMP, el cual permite mejorar la eficiencia de WAVE, posibilita que las aplicaciones controlen directamente los parámetros de la capa inferior, como la potencia de transmisión, la velocidad de datos, el número de canal y las direcciones del receptor, con esto se evita sobrecargar el sistema. (Li, 2015).
- El protocolo LTE (evolución a largo plazo), que es diseñado para manejar grandes cantidades de tráfico de datos a través de una red de conmutación de datos (Andrews et al., 2018, p.19). Ha sido puesto a prueba para comunicaciones vehiculares, según menciona (Trichias et al., 2012), que LTE puede llegar a cumplir con la mayoría de requisitos de las aplicaciones de sistemas de transporte inteligente (ITS), siempre y cuando no se alcance el límite de la capacidad de la red.
- También se ha utilizado el protocolo VCWC, el cual permite comunicación para advertir sobre colisiones vehiculares, a través de la difusión de mensajes de advertencia de emergencia, se indica el comportamiento anormal del vehículo, como el frenado de emergencia, velocidad desmedida, o salidas del carril repentinamente. (Willke et al., 2009).
- Según (Sallam y Mahmoud, 2015), se evaluaron protocolos de enrutamiento AOMDV (vector de distancia bajo demanda ad-hoc), y OLSR (protocolo de enrutamiento de estado de enlace optimizado), en escenarios VANET, en el cual los resultados de su simulación

muestran que OLSR es ligeramente mejor que AODV en el escenario ideal, Pero después de cierto umbral, AODV funciona mejor que OLSR.

Además, a continuación, se menciona las implementaciones e investigaciones realizadas en comunicación V2V

- En el año 2011, BMW, ya realizó investigaciones en sistemas de comunicación V2V, los ingenieros de BMW, efectuaron experimentos en un Sedan 6, en el que se demostró un sistema para la asistencia al conductor al giro a la izquierda en una intersección. Las estadísticas indican que del 30% al 60% de todas las lesiones que causan los accidentes automovilísticos y hasta un tercio de todas las muertes vehiculares ocurren en los cruces de carreteras. (Jurgen, 2012).
- La tecnología V2V, es una tecnología sumamente nueva, se ha implementado prototipos de comunicación V2V desde 2012 hasta el 2014, fue realizado en Estados Unidos, por NHTSA, junto con otras entidades, el prototipo fue montado en aproximadamente 2800 vehículos, con esto se pudo demostrar que esta tecnología, está lista para ser utilizada en el mundo real, teniendo en cuenta ciertas características a tener en cuenta. (NHTSA, 2017).
- Se han realizado pruebas con diferentes tipos de tecnologías o con algoritmos, basados en comunicación V2V, pero todos estos han sido prototipos para comprobar su veracidad y eficacia. Además, también la tecnología V2V está relacionada con los vehículos autónomos, por lo que en China se ha empezado a construir una zona experimental para experimentos de conducción autónoma, en la ciudad de Zhangzhou en el año de 2017. (Uhlemann, 2017).
- Toyota a finales de 2015 implementó un paquete de ITS, en tres modelos de sus vehículos en Japón, este paquete denominado ITS Connect, el cual es compatible con tecnologías V2V y V2I, con una frecuencia estandarizada en Japón de 760 MHz. Al expandir esta tecnología, Toyota espera ayudar a reducir la cantidad de accidentes que ocurren cerca de las intersecciones, en Japon representa aproximadamente el 40 % de todos los accidentes de tráfico. (Toyota, 2015).
- Para el año 2017, la industria automotriz Cadillac, informó que los vehículos tipo sedán CTS, equiparan tecnología V2V, que manejaran hasta 1000 mensajes por segundo, esta información será compartida y se podrá usar para alertar a los conductores sobre los próximos peligros potenciales, para un futuro conectado y seguro. (Cadillac, 2017).
- LA NHTSA, planteó una normativa en la cual los fabricantes de los vehículos debían de incluir la tecnología V2V en sus modelos de automóviles, esto se aplicaría desde el 2021 y para el año 2023 debería de ser obligatorio, permitiendo una comunicación masiva, entre

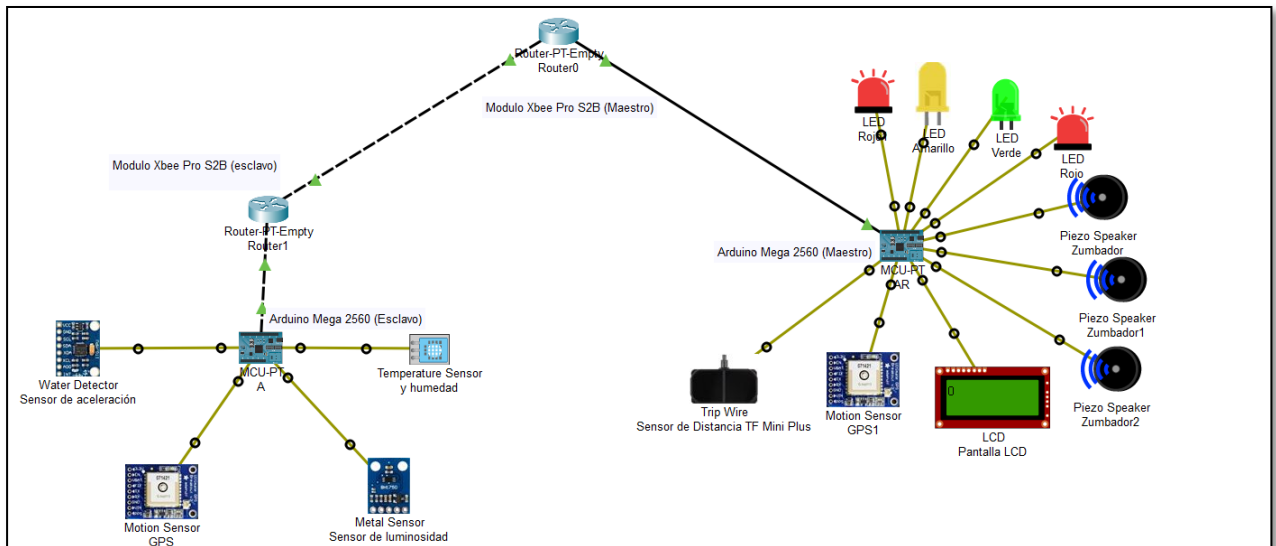
los vehículos, independiente de la marca del automóvil (NHTSA, 2016). Para el año que se llevó la presente investigación, no se encontró indicios que aquella normativa haya sido aprobada en EEUU, más bien en el portal Motortrend, se menciona que Lanctot, director de “Automotive Connected Mobility”, piensa que Estados Unidos y Europa han perdido un poco el rumbo con respecto a tecnología V2V, mientras que China sigue avanzando en desarrollar dicha tecnología. (Stevens y Marrs, 2022).

- En la actualidad no existe esta comunicación V2V implementada en masa en vehículos, puesto que esto depende de varios factores para llevarse a cabo, de los que se destaca falsos positivos, privacidad, problemas gubernamentales, hackeo, entre otros más.

6.2. Diseño de la red de comunicación

Al diseñar la red en Cisco Packet Tracer se aprecia en la Figura 37, el realizar el diseño de la red, es posible comprender como se llevará a cabo la transferencia de los datos en la red de comunicación. También permite observar cómo se realizarán las conexiones físicas de la red.

Figura 37.
Diseño de la red de comunicación V2V

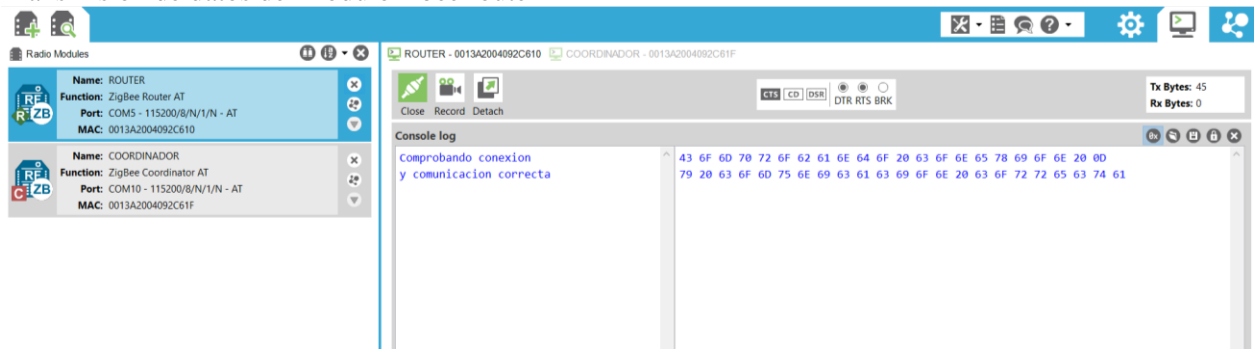


Nota. En la presente Figura se aprecia el diseño de la red de comunicación V2V, realizado en el software Cisco Packet Tracer

Antes de ser implementada la red creada, en los vehículos, se realizó simulaciones para comprobar el correcto funcionamiento de los sensores y de la comunicación en la red. Por lo cual se empezó simulando los módulos Xbee en el programa XCTU, al ser simulación, se enviará un mensaje corto del módulo esclavo al módulo maestro, para comprobar una comunicación correcta entre módulos. En la Figura 38, se observa el mensaje que envía el router o esclavo hacia el coordinador.

Figura 38.

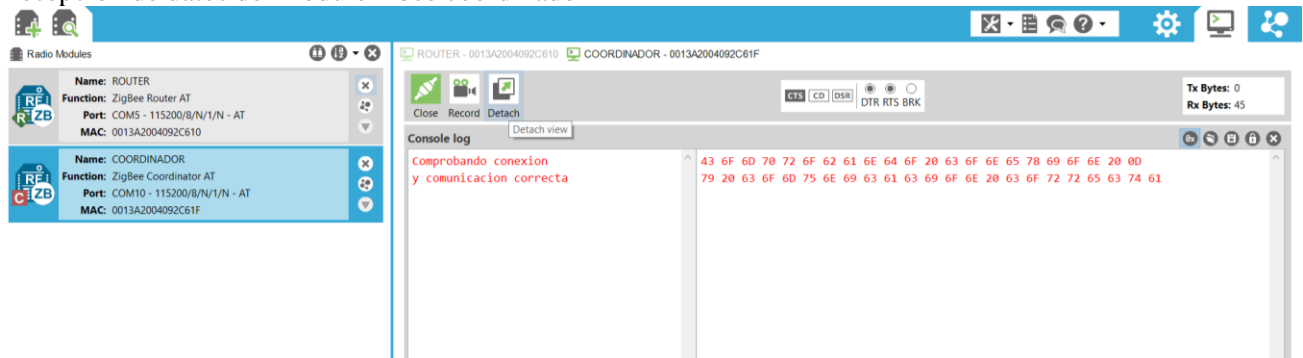
Transmisión de datos del módulo Xbee router



En la siguiente Figura 39, se observa el mensaje que ha sido recibido por el coordinador.

Figura 39.

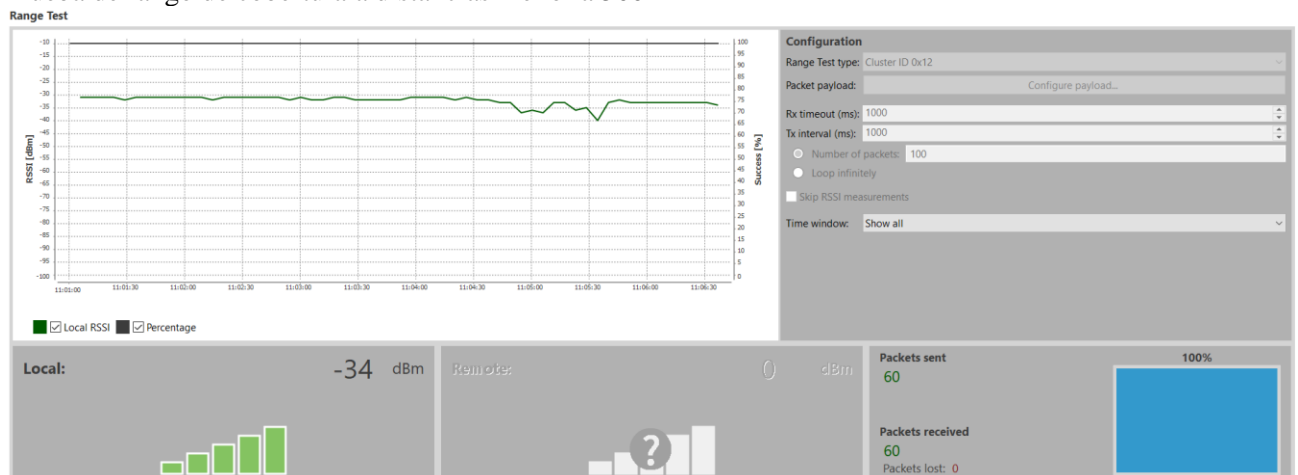
Recepción de datos del módulo Xbee coordinador



Además, se realizó una prueba de rango de cobertura, es decir el software XCTU, permite realizar un *test range*, el cual permite conocer la potencia de los módulos Xbee, a ciertas distancias entre los módulos Xbee. En modo de trabajo AT, antes de los 500 m, se ha comprobado una buena potencia y transmisión de datos, en la Figura 40 se puede observar que existe una buena transferencia de datos, teniendo una transmisión de datos en 34 dBm (decibelios de milivoltio), o también resaltado como 100 % de transmisión de datos correctos. Resaltar que la gráfica básicamente es lineal, aunque existen unas ciertas variaciones, que pueden deberse a cuanto más se alejen los módulos Xbee, existen interferencias en las señales.

Figura 40.

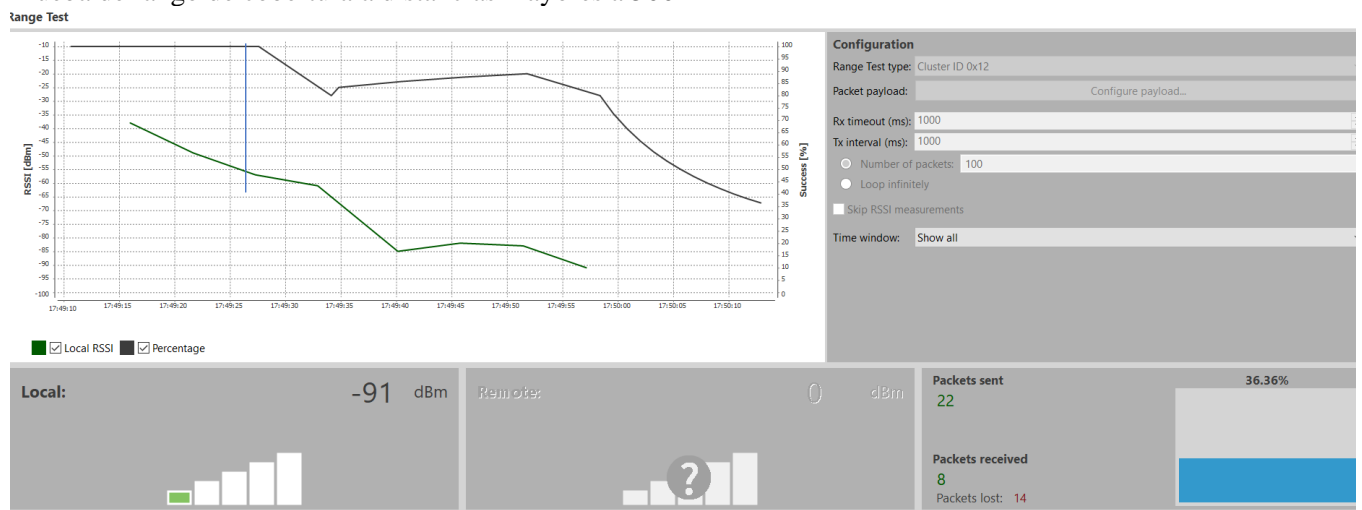
Prueba de rango de cobertura a distancias menor a 500 m



Después de 500 m, se empieza a perder potencia y se deja de recibir datos correctos. En la Figura 41 se aprecia una deficiente transmisión de datos, puesto que la transmisión empieza siendo correcta, pero luego, decae la señal, siendo hasta 58 dBm una transmisión al 100 %, luego de esto se empieza a perder paquetes de información, llegando a tener una transmisión de datos de - 91 dBm, siendo un porcentaje sumamente bajo de 36 % aproximadamente.

Figura 41.

Prueba de rango de cobertura a distancias mayores a 500 m



Una vez conectado todos los elementos y que la comunicación inalámbrica sea exitosa, se comprueba que el Arduino esclavo y el Arduino maestro están transfiriendo los datos de manera correcta.

6.3. Implementación y visualización de la red de comunicación creada

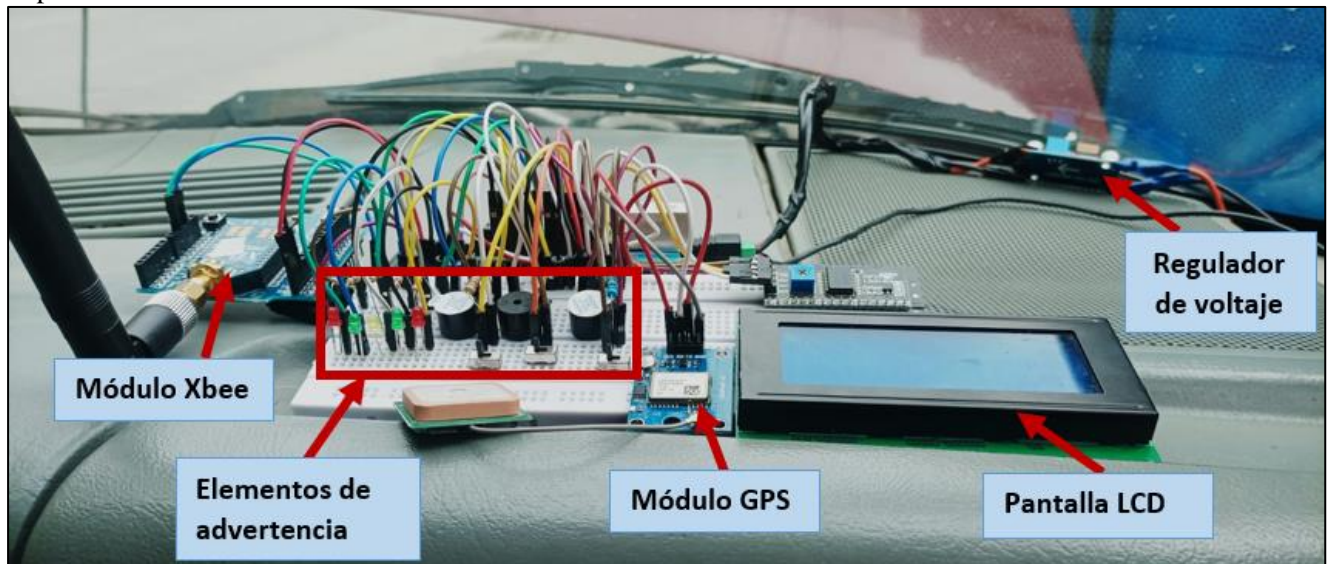
En este punto se presentará los resultados de la implementación del prototipo de la red de comunicación V2V en los vehículos, y de la visualización de los datos en la plataforma de Ubidots.

6.3.1. Implementación

Las pruebas fueron realizadas en dos vehículos, en un vehículo se implementó el Arduino maestro y en el otro el Arduino esclavo. Además, las pruebas fueron realizadas en vías reales de la ciudad de Loja, no se planteó una ruta en específica, por lo tanto, el prototipo de comunicación V2V ha sido probado en diferentes sectores de la ciudad, para comprobar su efectividad y conectividad. En la siguiente Figura 42, se observa los sensores implementados en el vehículo que va a recibir los datos (coordinador).

Figura 42.

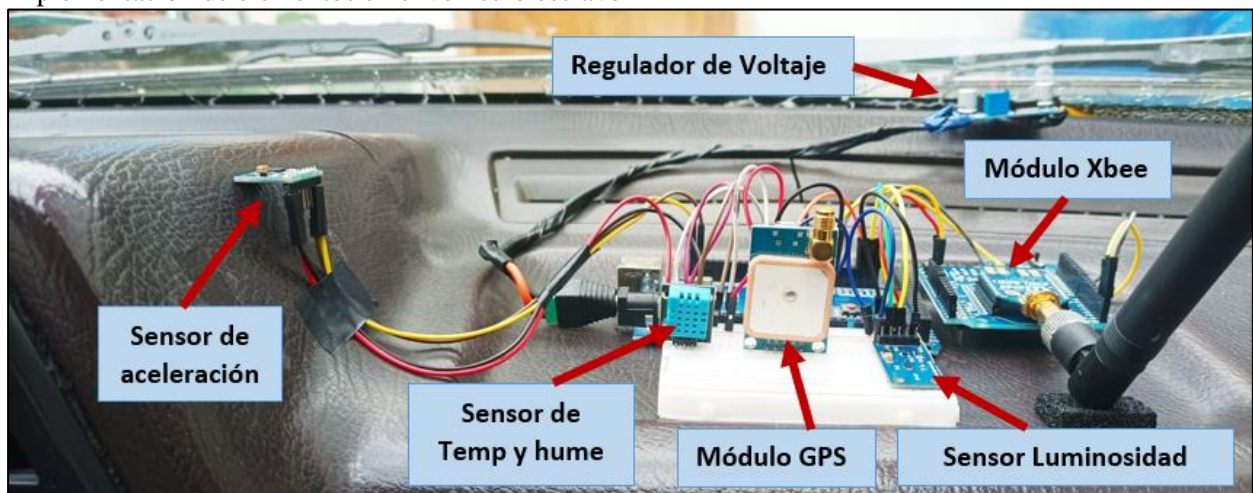
Implementación de elementos en el vehículo coordinador



Nota. La Figura se ha instalado la red de comunicación para el vehículo que va a recibir y procesará la información del vehículo esclavo. Importante señalar que el sensor de distancia no se encuentra visible en la imagen, debido a que esta ubicado en la parte delantera del vehículo.

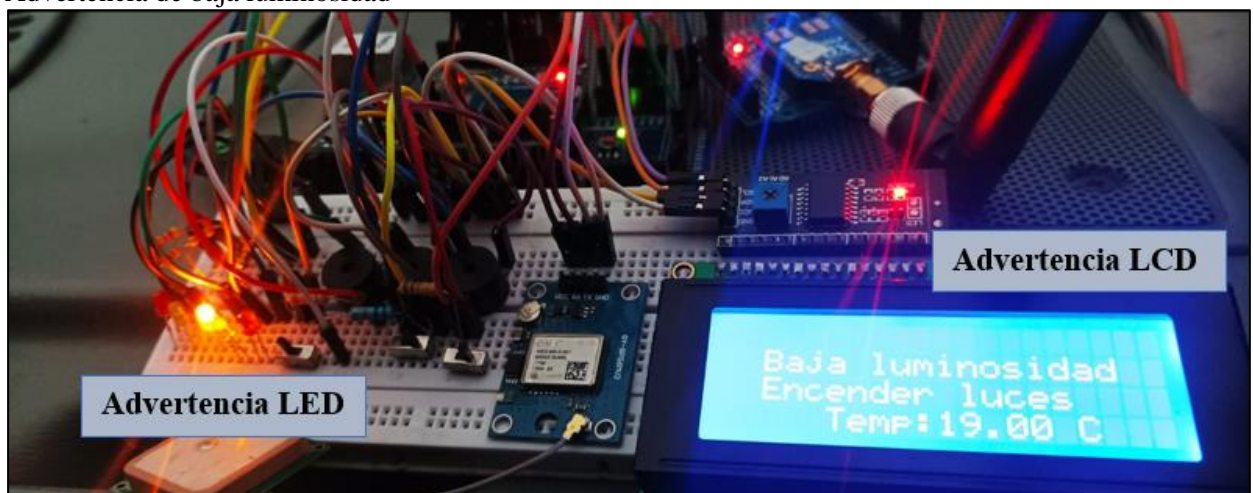
La Figura 43, se observa la configuración de la red de comunicación para el vehículo que enviará información al vehículo coordinador, finalmente se puede destacar que el sensor de aceleración, para obtener mayores precisiones se lo puede colocar en posición con el centro de masa del vehículo.

Figura 43.
Implementación de elementos en el vehículo esclavo



A continuación, se detalla algunas funciones que cumple la red de comunicación creada, la primera función, cuando existe una baja luminosidad, el vehículo esclavo envía los datos que está obteniendo del sensor de luminosidad, hacia el vehículo coordinador, el cual procesa los datos y estará programado según (Instituto Ecuatoriano de Normalización [INEN], 2015), en la normativa NTE INEN 1155, que recomienda una intensidad luminosa del faro delantero aproximadamente de 135 luxes, al ser menor a este valor, se encenderá una advertencia, la cual constará de un mensaje en la pantalla LCD, advirtiendo una baja luminosidad. como también se observará un led encendido, esto con el fin de alertar al conductor, que encienda las luces, para tener una mejor visibilidad.

Figura 44.
Advertencia de baja luminosidad

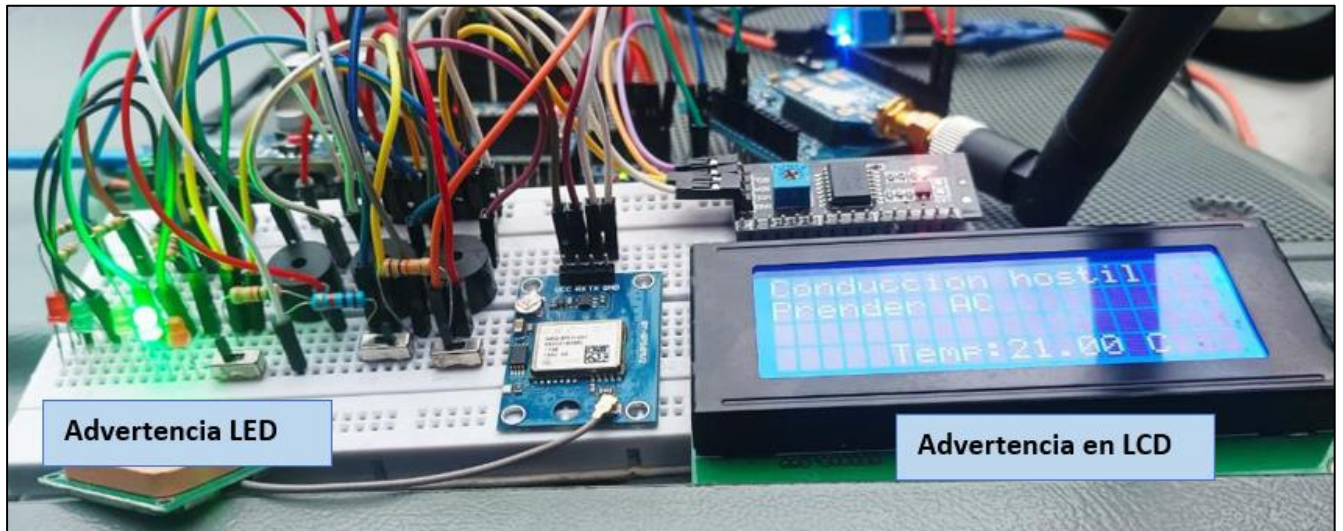


La segunda función consiste en alertar al conductor, cuando existe una temperatura excesiva o temperatura baja, pero complementado con la humedad relativa existente, un automóvil enviará la información adquirida por el sensor de temperatura y humedad, hacia otro automóvil, este último se encuentra programado según la referencia de (Zhou Qiao, 2013), en

el cual se menciona niveles de confort aceptables para la temperatura de 23°C a 28°C, y de 30 % a 70% rango confortable para la humedad, por lo tanto, se ha programado cuando exista dos condiciones, una temperatura mayor a 28°C y una humedad menor a 30 %, cuando se cumpla estas condiciones se enviará una alerta mencionando que existen condiciones desfavorables, e igualmente se enviara una alerta cuando se cumplan dos condiciones una de temperatura menor a 23 °C y una humedad mayor a 70 %.

Figura 45.

Advertencia de condiciones desfavorables

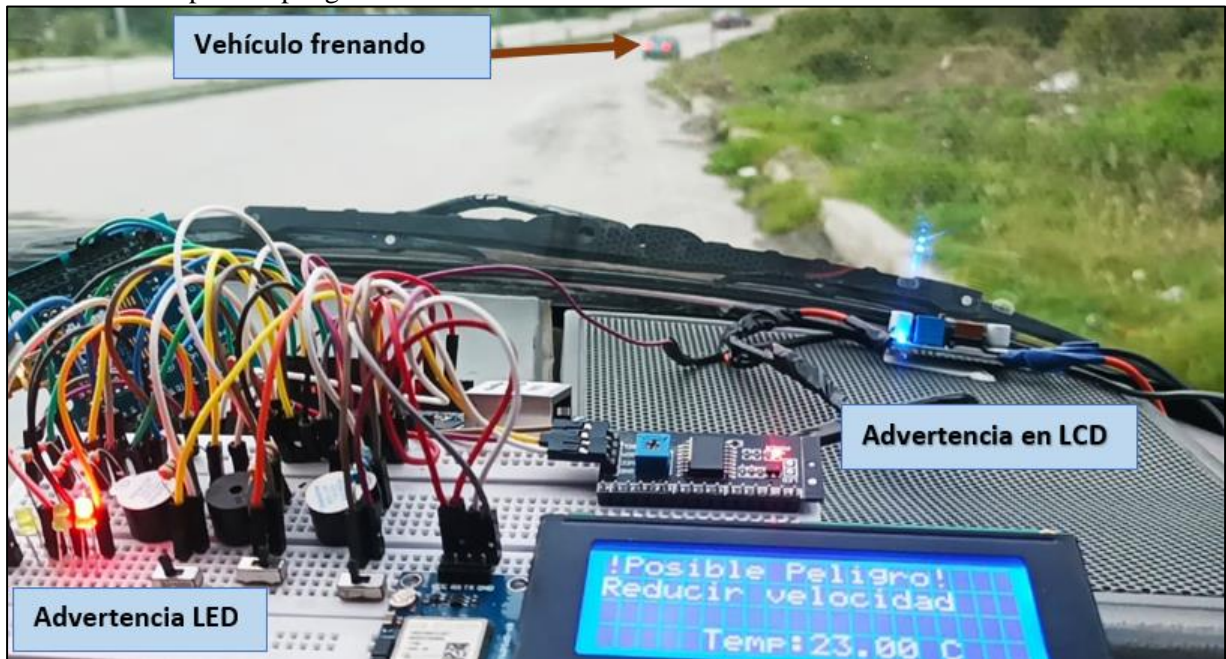


Nota. Imagen del circuito de conexión de los sensores de temperatura y humedad, indicando las condiciones en las que se encuentra el vehículo.

Continuando con la tercera función, la cual permite alertar al conductor cuando existe un posible peligro en la vía, por lo que existirá una desaceleración brusca por el conductor. Por lo tanto, un vehículo enviará información obtenida del sensor de aceleración, a otro vehículo el cual se encontrará programado según lo mencionado por (Deligianni et al., 2017), los cuales describen que existen diferentes umbrales de desaceleración según varias literaturas, en una de estas referencias se describe una desaceleración aproximadamente desde 0.2 g hasta 0.4 g. Pero del estudio que han realizado describen que a 0.3 g, es una desaceleración de conducción normal, por lo tanto, de esta información obtenida, para esta investigación se ha decidido utilizar una desaceleración de 0.4 g (3.9 m/s^2).

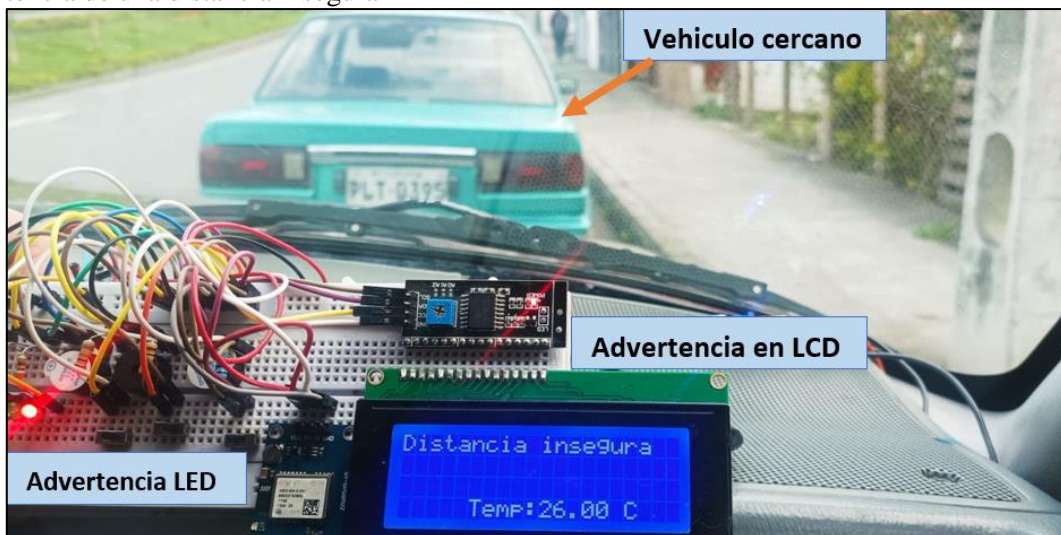
Para cuando reciba un dato con una aceleración menor a -3.9 m/s^2 , se envié una alerta, esta prevención se verá reflejada a través de una pantalla LCD, además de encenderse un zumbador y un led.

Figura 46.
Advertencia de posible peligro



La cuarta función que brinda, alertar al conductor que existe una distancia insegura entre un vehículo y otro, un auto enviará información obtenida del sensor de distancia, hacia otro auto, este se encontrará programado según lo estipulado en el (Decreto Ejecutivo 1196, 2012), el cual menciona que en áreas intracantoniales, se deberá de preservar una distancia prudencial mínima de tres metros con respecto al vehículo al que antecedan en el mismo carril. Por lo que, al tener valores iguales o menores a tres metros de distancia, se enviará una alerta al conductor esto a través de la visualización de un mensaje de advertencia en la pantalla LCD, además para mayor precaución se encenderá un led y se escuchará un zumbador.

Figura 47.
Advertencia de una distancia insegura



Por último, la función, de alertar al conductor cuando está entrando en una intersección de posible colisión con un vehículo, para esto un vehículo enviará su ubicación hacia otro vehículo, este último estará programado para comparar su ubicación y la ubicación recibida. Al comparar las ubicaciones y ser muy similares, se enviará una alerta que se reflejará en una pantalla LCD, además de encender un led y un zumbador. Esta advertencia empezará a tenerla el conductor con bastante anterioridad antes de llegar a la intersección, es decir a mitad de la cuadra de la calle. En la siguiente Figura 48 se observa la situación mencionada.

Figura 48.

Vehículos entrando en una intersección



En la Figura 49 se aprecia la advertencia que se recibe

Figura 49.

Advertencia en una intersección



Finalmente, para complementar el prototipo se simuló por medio de interruptores las acciones que debería realizar el conductor para dejar de recibir advertencias. Por ejemplo,

cuando se alerte de luminosidad baja, por medio de un interruptor se simuló como el encendido de las luces, por lo que la advertencia desaparecerá. Y cuando las condiciones de temperatura y humedad no son las adecuadas, por medio de otro interruptor se simuló el encendido del aire acondicionado, para mejorar las condiciones de conducción.

6.3.2. Visualización en Ubidots

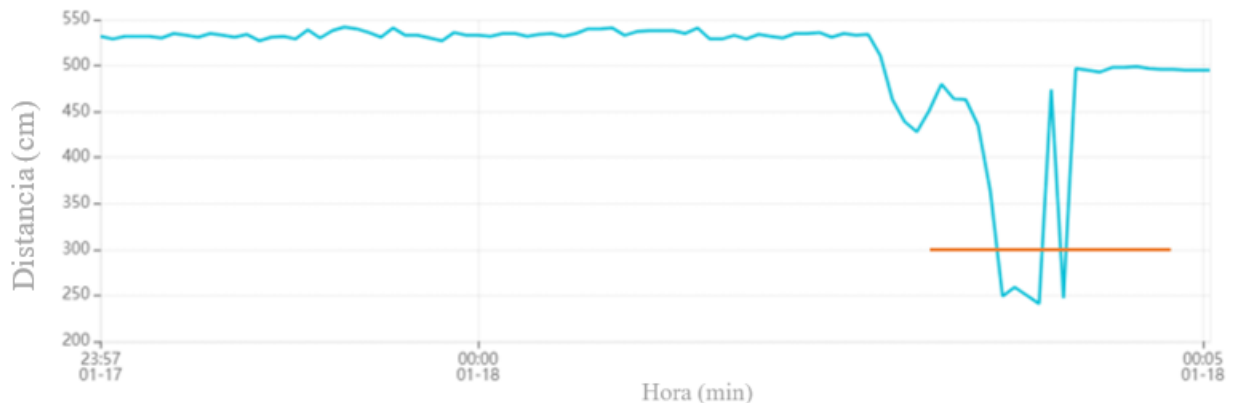
En lo que respecta a la plataforma de Ubidots se logró observar los datos, que están siendo transmitidos y procesados dentro de la red de comunicación. A continuación, se presentan los resultados de las gráficas obtenidas.

- **Datos del sensor de distancia**

En la Figura 50, se aprecia los valores del sensor de distancia, se destaca en la gráfica, que se mantiene casi lineal hasta cierto punto, hasta que empieza a acercarse a distancias menores de 300 cm, lo que indica que el vehículo se encuentra muy cercano a un vehículo o un obstáculo, por lo tanto, en estos valores, se enviará una advertencia al conductor que está transitando a una distancia insegura.

Figura 50.

Gráfica del sensor de distancia

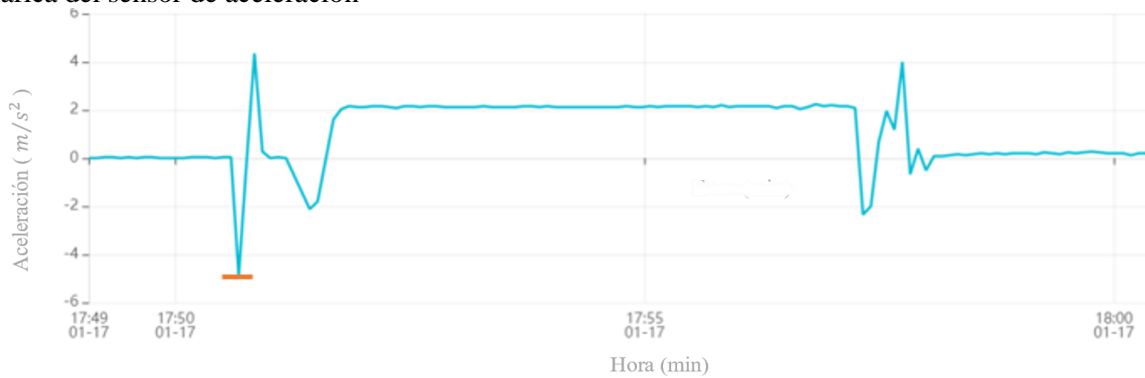


- **Datos del sensor de aceleración**

En la Figura 51 se observa los valores del sensor de aceleración, se destaca un punto que indica que ha existido una desaceleración la cual es mayor a $-3.9 m/s^2$. En este punto, se considera una frenada brusca, por lo que en el prototipo de comunicación se enviará una advertencia al conductor de posibles riesgos en la vía.

Figura 51.

Gráfica del sensor de aceleración

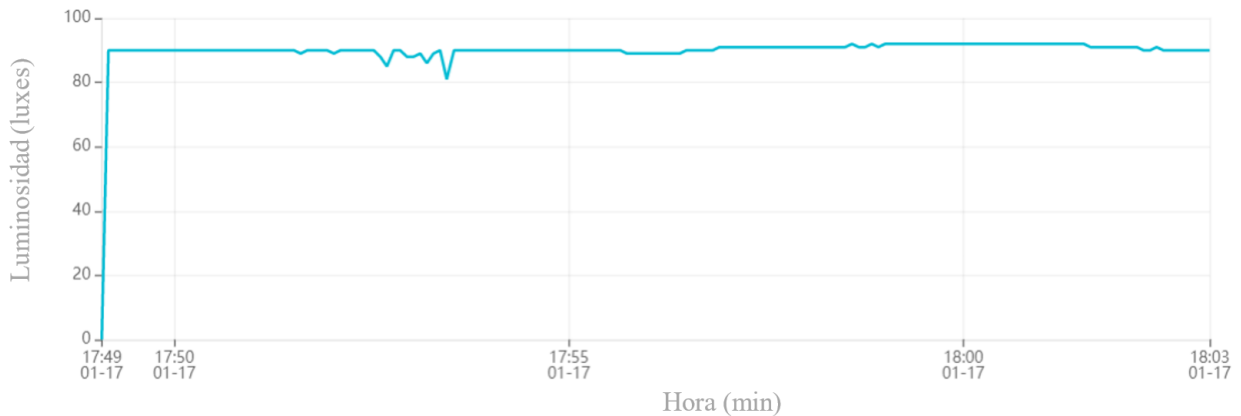


- **Datos del sensor de luminosidad**

En Figura 52, muestra los valores del sensor de luminosidad, los datos han sido tomado al atardecer, por lo que se puede apreciar valores casi lineales, en 90 luxes aproximadamente, las irregularidades presentes, se deben cuando el vehículo pasa por debajo de techos de casas o debajo de árboles. Al ser un valor menor al que ha sido programado, se encenderá una advertencia para que el conductor pueda encender las luces de su vehículo.

Figura 52.

Gráfica del sensor de luminosidad

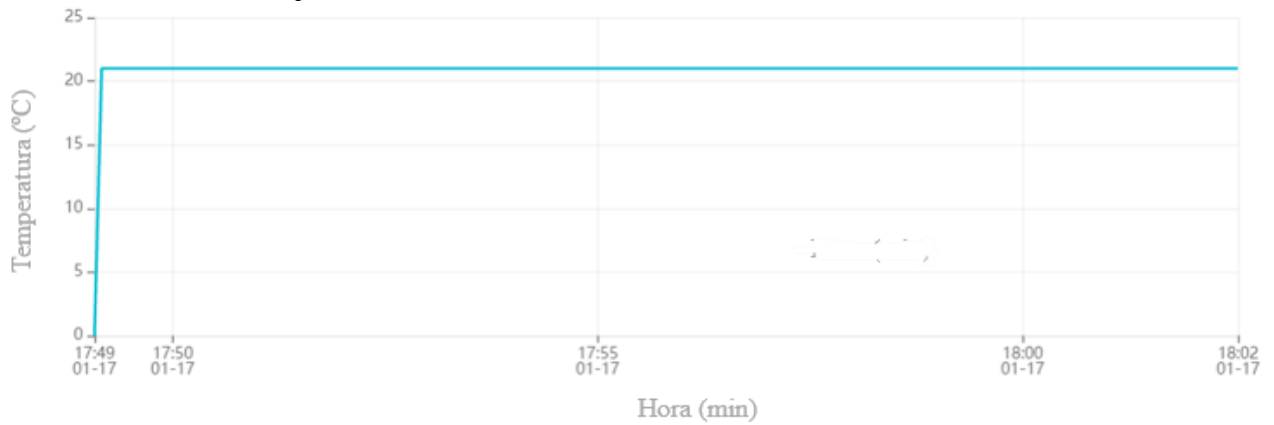


- **Datos del sensor de temperatura y humedad**

En la Figura 53 se pueden ver los valores registrados por el sensor de temperatura y humedad, pero en esta gráfica solo se muestran los datos de temperatura, muestra una tendencia lineal cercana a 21 °C. Importante destacar que la temperatura está en rangos desfavorables para una conducción con condiciones óptimas.

Figura 53.

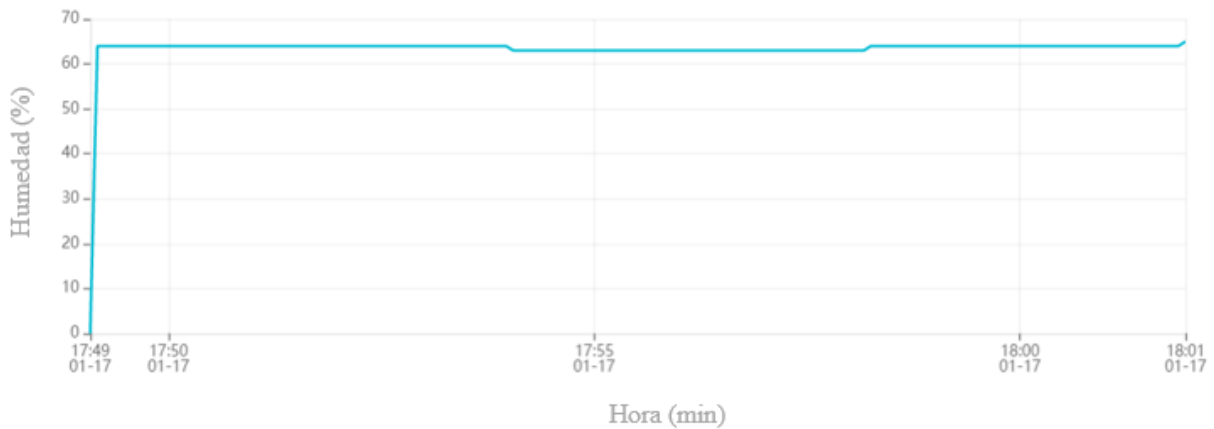
Gráfica del sensor de temperatura



La Figura 54, se presentan los valores del sensor de humedad, se observa una humedad lineal, con breves variaciones, aproximadamente una humedad de 63 %, valor adecuado para tener una agradable conducción.

Figura 54.

Gráfica del sensor de humedad



Además, Ubidots permite mostrar la ubicación de donde se están recibiendo los datos, por lo que se ha mostrado la ubicación del lugar en el que se estuvo realizando las pruebas de comunicación vehicular.

7. Discusión.

El prototipo de comunicación vehículo a vehículo (V2V) ha sido creado con el objetivo de prevenir y reducir el número de accidentes automovilísticos, lo que se traduciría a una reducción en el número de fallecidos y heridos producidos por estos siniestros. Según la (NHTSA, 2016), con la implementación de esta tecnología V2V, se podría evitar aproximadamente el 80 % de los accidentes vehiculares que involucran a conductores sin impedimentos.

En la actualidad las investigaciones con respecto a la tecnología de comunicación V2V, han aumentado, ya se ha planteado varios prototipos y sistemas a prueba que han estado en funcionamiento. En 2011 según (Jurgen, 2012), BMW empezó realizando pruebas V2V en un vehículo tipo sedán 6. Asimismo, en 2014 la entidad (NHTSA, 2017), realizó pruebas en más de dos mil vehículos. También, para 2015 (Toyota, 2015), incluyó en su paquete de ITS, en tres modelos de sus vehículos, este paquete era compatible con tecnología V2V. Finalmente se destaca la participación de (Cadillac, 2017), que para el año 2017 equipó con tecnología V2V a los vehículos tipo sedán.

La comunicación inalámbrica del protocolo ZigBee, según menciona la empresa (DIGI, 2022), ha sido utilizado por lo general en aplicaciones relacionada a los hogares inteligentes, aplicaciones de IOT (internet de las cosas), aplicaciones control y supervisión, en dispositivos médicos, entre otras más. ZigBee permite la comunicación inalámbrica a un bajo coste y bajo consumo, lo que permite tener una gran utilidad en la actualidad. No obstante, el protocolo de comunicación ZigBee en la actualidad como menciona (Huang, 2019), está siendo utilizado en la industria automotriz para interconectar ECU, sensores y actuadores del vehículo, con el objetivo de monitorear, controlar y leer el estado de las aplicaciones del vehículo.

El sistema de comunicación vehicular V2V implementado mediante el protocolo ZigBee, ha sido de gran utilidad, ha cumplido con lo requerido, además la información enviada como recibida, ha sido correcta, en ciertos momentos de las pruebas realizadas se presentó ciertos errores o información no recibida, que puede ser a causa del movimiento y velocidad del vehículo, como de la interferencia de señales en el ambiente. (Revankar et al., 2018), realizaron investigaciones de comunicación V2V mediante ZigBee, en el que su sistema cumple varias funciones, de las que se destaca dos, la primera que mide la distancia recorrida e informa el tiempo transcurrido, desde que se aplica el freno, hasta la detención del vehículo, considerando velocidades variables, la segunda función, es que mide la distancia entre dos vehículos, y avisará en caso que exista posibilidad de colisión. De igual forma, (Deepa et al., 2020),

realizaron indagaciones de la comunicación V2V con ZigBee, en su sistema creado, han logrado la comunicación entre dos vehículos mediante ZigBee, como también la medición de distancia entre estos vehículos mediante un sensor ultrasónico, el que permitirá alertar de que un vehículo viene antes.

8. Conclusiones.

Se realizó una investigación acerca de los diferentes protocolos de comunicación empleados en la comunicación vehicular, los cuales buscan mejorar la efectividad de la comunicación vehículo a vehículo, reducir la latencia y con una velocidad de transmisión de información adecuada.

De igual forma se investigó sobre la tecnología V2V en la actualidad, la cual se encuentra progresando cada día más para brindar seguridad y evitar accidentes de tránsito. Aunque todavía no se ha implementado masivamente en los vehículos y existen barreras por superar, ya se han realizado prototipos y algunas empresas automovilísticas han empezado a implementar este sistema en alguno de sus modelos de vehículos.

Se diseñó un sistema que permitió la comunicación e intercambio de información entre dos vehículos, a través del protocolo ZigBee de forma efectiva, implementando una red de sensores utilizando la placa de desarrollo Arduino.

Se implementó el sistema de comunicación desarrollado en dos vehículos convencionales y se comprobó la eficiencia de la red mediante pruebas de ruta, los datos obtenidos por los sensores se mostraron mediante la plataforma Ubidots. No obstante, se ha encontrado algunas irregularidades momentáneas de transmisión de datos, que pueden estar relacionados al modo de operación utilizado por los módulos Xbee, por interferencia de señales, como por ejemplo redes telefónicas, presentes en el ambiente, el movimiento y la velocidad de los vehículos.

El protocolo de comunicación ZigBee, presenta un reducido consumo de energía, su construcción es simple y sencilla, tiene un bajo coste y opera con una banda de frecuencia mundial de 2,4 GHz, pero también presenta algunas desventajas como la tasa de transferencia baja y el rango de transmisión de datos es relativamente baja.

La plataforma Ubidots, es una aplicación bastante útil a la hora de enviar datos de Arduino, su visualización es muy amigable y muy sencilla de entender, además es fácil de utilizar, no requiere de demasiada programación para empezar a recibir información, por último, permite obtener gráficas en tiempo real de los datos que se están recibiendo.

9. Recomendaciones.

Para proyectos más complejos, en el que se necesite conectar más dispositivos de recepción y se requiera mayor rango de transmisión de información, se sugiere que los módulos Xbee, funcionen en modo de operación API, puesto que este permite configurar los dispositivos Xbee remotos como locales, a diferencia del modo transparente que se debe entrar al modo comando y estar actualizando los dispositivos. Además, el modo API permite transmitir datos fiables, seguros, rápidos y permite que el rango de comunicación sea mayor

Si se requiere añadir más sensores que funcionen bajo la interfaz I2C o UART, a la red de comunicación, es recomendable utilizar otro microcontrolador diferente a Arduino, o en su caso realizar un conexionado especial en el Arduino Mega para conseguir conectar más dispositivos a la red. Teniendo en cuenta la capacidad máxima del microcontrolador (8 bits), puesto que, si sobrepasa, la información se verá truncada.

Si se desea mejorar el prototipo de comunicación V2V se puede ocupar la información proveniente de sensores propios del vehículo, aunque esto requerirá de nuevas configuraciones y programaciones, además se puede añadir nuevas condiciones para cada una de las funciones que proporciona el prototipo para que sea más eficiente y apegado a la realidad. No obstante, se debe de tener en cuenta que se está trabajando con diferentes protocolos de comunicación, uno que es el CAN y el otro Zigbee.

Con respecto a la función de advertencia en las intersecciones, se sugiere añadir más condiciones a cumplirse. Como por ejemplo añadir predicción de ruta, también se puede utilizar un dispositivo con cámara ubicado en la esquina de la intersección, que se esté comunicando con el vehículo emisor y el vehículo receptor, así se podrá mejorar la advertencia que se proporcionara al conductor.

El prototipo de comunicación vehicular construido, ha sido sometido a pruebas de ruta en la zona urbana de la ciudad de Loja. Por lo tanto, para proyectos futuros se recomienda realizar pruebas en trayectos suburbanos, con el fin de validar que la red sea lo suficientemente robusta para tener una comunicación correcta y confiable.

Para investigaciones futuras en este prototipo de comunicación vehículo a vehículo, se sugiere que se pueda gestionar un asistente al conductor, para que automáticamente cuando llegue una advertencia, se controle los actuadores del vehículo, según correspondan a la alerta recibida.

Igualmente, para investigaciones futuras, en el que se requiera realizar comunicaciones en más de dos vehículos, se sugiere configurar la red de comunicación a modo malla, el cual

permitirá que todos los nodos estén entrelazados y comunicados, por lo que cada vehículo que se comunique actuara como un nodo.

Para el envío de datos a la plataforma de Ubidots, se podría utilizar un método diferente para transmitir los datos que no sea por medio de cable LAN, se recomienda utilizar la transferencia de datos vía WiFi, consiguiendo la eliminación de cables, sin la necesidad de estar cerca de un router.

Comprobar la conexión y la alimentación de corriente correcta de los dispositivos que se utilizará para obtener los datos, dado que si existe una deficiente alimentación o una mala conexión puede obtenerse datos erróneos, además la comunicación de la red será demasiado inestable.

Tener en cuenta el tiempo de muestreo de cada sensor utilizado en el prototipo de comunicación, puesto que, si no se trabaja con el tiempo de muestreo correcto es probable que la transmisión de datos sea errónea.

10. Bibliografía.

- Abbasi, I y Shahid A. (2018). *A Review of Vehicle to Vehicle Communication Protocols for VANETs in the Urban Environment*.
- Anand, P., Naveen, C., Alfred, D. y Seungmin, R. (2016). *Intelligent Vehicular Networks and Communications. Fundamentals, Architectures and Solutions*. (1ra ed.) Elsevier.
- Andrews, J., Humphreys, T., Bhat, C., Heath, R., Narula, L., Choi, C. y Li, J. (2018). Evaluation of Routing Protocols for Vehicular Ad hoc Networks (VANETs) in Connected Transportation Systems (Informe tecnico Nro. 135). D-STOP. <https://ctr.utexas.edu/wp-content/uploads/135.pdf>
- ANT. (2023). Visor de Siniestralidad Nacional. <https://www.ant.gob.ec/visor-de-siniestralidad-estadisticas/>
- Arizaga, J., Alonso, M., Álvarez, R. y Castañeda J. (2018). Redes VANET Vehicular Ad-Hoc Networks, la conectividad de los autos. Primera parte. https://www.researchgate.net/publication/331471365_Redets_VANET_Vehicular_Ad-Hoc_Networks_la_conectividad_de_los_autos_Primer_a_parte
- Cadillac. (2017). *Press Releases: V2V Safety Technology Now Standard on Cadillac CTS Sedans*.
https://media.gm.ca/dld/content/Pages/news/ca/en/2017/Mar/0315_V2V/_jcr_content/rightpar/download/file.res/0315_V2V%20Cadillac%20V2V%20Technology%20En.pdf
- Chen, W. (2015). *Vehicular Communications and Networks*. Elsevier. <https://doi.org/10.1016/C2013-0-16482-3>
- CISCO. (20 de abril de 2020). Cisco Packet Tracer 8.2 Frequently Asked Questions. <https://www.netacad.com/courses/packet-tracer/faq>
- Coñapes, S. (31 de marzo de 2015). *Tipos De Redes inalámbricas*. Google Sites. <https://sites.google.com/site/redesinalambricas3/tipos-de-redes-inalambricas>
- Decreto Ejecutivo 1196 [Ministerio de Obras Públicas]. En el cual se establecen leyes de transporte terrestre tránsito y seguridad vial. Registro Oficial 731 de 25 de junio de 2012 (Ecuador). <https://www.obraspublicas.gob.ec/wp-content/uploads/downloads/2015/03/Decreto-Ejecutivo-No.-1196-de-11-06-2012-REGLAMENTO-A-LA-LEY-DE-TRANSPORTE-TERRESTRE-TRANSITO-Y-SEGURIDAD-VIA.pdf>
- Deepa, N., Hemavarthini, M., Monisha, S., Monisha, T. y Susithra. (2020). Vehicle to Vehicle Communication Using Zigbee. *International Journal of Advanced Research Trends in*

- Deligianni, S., Quddus, M., Morris, A., Anvuur, A. y Reed S. (2017). Analyzing and Modelling Drivers' Deceleration Behaviour from Normal Driving. *Transportation Research Record*, 2663 (1), 134-141. <https://doi.org/10.3141/2663-17>.
- DIGI, (2022). Zigbee RF Modules: XBEE2, XBEEPRO2, PRO S2B. <https://www.digi.com/resources/documentation/digidocs/pdfs/90000976.pdf>
- Eady, F. (Hands-On ZigBee: Implementing 802.15.4 with Microcontrollers. (1ra ed.). Newnes.
- Elahi, A. y Gschwender, A. (2009). Zigbee wireless sensor and control network. (1ra ed.). Prentice Hall.
- Faludi, R. (2011). *Building Wireless Sensor Networks with ZigBee, Xbee, Arduino, and Processing*. (1ra ed.). O'Reilly Media.
- Farahani, S. (2008). ZigBee Wireless Networks and Transceivers. (1ra ed.). Elsevier. <https://doi.org/10.1016/B978-0-7506-8393-7.X0001-5>
- Fesvial y Lumileds. (2021). ¿Qué importancia damos a la iluminación en el vehículo? <https://revista.dgt.es/Galerias/motor/noticias/2021/04ABRIL/ESTUDIO-ILUMINACION-VEHICULOS.pdf>
- Fortuño, A. (2012). Desarrollo e implementación de una red de sensores Zigbee mediante el dispositivo Xbee de Digi. [Tesis de grado, Departamento de Ingeniería Eléctrica Electrónica y Automática, Universitat Rovira i Virgili]. <http://deeea.urv.cat/public/PROPOSTES/pub/pdf/1957pub.pdf>
- Gupta, M., Benson, J., Patwa F y Sandhu, R. Secure V2V and V2I Communication in Intelligent Transportation Using Cloudlets. *IEEE Transactions on Services Computing*, 15 (4), 1912-1925. <https://doi.org/10.1109/TSC.2020.3025993>
- Hasan, F., Siddique, N. y Chakraborty, S. (2018). Intelligent Transportation Systems 802.11-based Vehicular Communications. (2daed.). Springer. DOI: 10.1007/978-3-319-64057-0
<https://www.ant.gob.ec/visor-de-siniestralidad-estadisticas/>
<https://www.who.int/es/news-room/fact-sheets/detail/road-traffic-injuries>
- Huang, J., Zhao, Y., Zhou Y. y Xing C. (2019). *In-Vehicle Networking: Protocols, Challenges, and Solutions*, in *IEEE Network*. *IEEE Network*, 33(1), 92-98. doi: 10.1109/MNET.2018.1700448.
- Huidobro, J. (2021). *El vehículo conectado*. ACTA, 1-20. <https://www.acta.es/recursos/revista-digital-manuales-formativos/651-098>

- Instituto Ecuatoriano de Normalización [INEN]. (2015). Vehículos automotores. Dispositivos para mantener o mejorar la visibilidad. Requisitos (NTE INEN 1155). <https://normas-apa.org/referencias/citar-normas-tecnicas-o-de-calidad/>
- Jurgen, R. (Ed.). (2015). V2V/V2I Communications for Improved Road Safety and Efficiency. SAE International. DOI: 10.4271/PT-154
- Li, J. (2015). *An Overview of the DSRC/WAVE Technology*. 7th International ICST Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness.
- Medina, A. (2019). *Comunicaciones V2V con esquemas de seguridad I.A. Contenido de la plática*. ResearchGate. https://www.researchgate.net/publication/334039048_Comunicaciones_V2V_con_esquemas_de_seguridad_IA_Contentido_de_la_platica
- MITMA. (21 de enero de 2020). *Sistemas inteligentes de transporte*. Ministerio de Transportes, Movilidad y Agenda Urbana. <https://www.mitma.gob.es/transporte-terrestre/sistemas-inteligentes-de-transporte/sistemas-inteligentes-de-transporte-its>
- Morgan, Y. (2010). *Managing DSRC and WAVE Standards Operations in a V2V Scenario*. *International Journal of Vehicular Technology*, Hindawi. 1-18. <https://doi.org/10.1155/2010/797405>
- NHTSA. (16 de agosto de 2016). *Federal Motor Vehicle Safety Standards; V2V Communications*, https://www.nhtsa.gov/sites/nhtsa.gov/files/documents/v2v_nprm_web_version.pdf
- NHTSA. (2014). “*VEHICLE-TO-VEHICLE COMMUNICATIONS: READINESS OF V2V TECHNOLOGY FOR APPLICATION*”, https://www.nhtsa.gov/sites/nhtsa.gov/files/documents/v2v_fact_sheet_101414_v2a.pdf
- NHTSA. (24 de marzo de 2017). *Comunicación Vehículo a Vehículo*, <https://www.nhtsa.gov/es/tecnologia-e-innovacion/comunicacion-vehiculo-vehiculo>
- OMS. (2022). *Traumatismos causados por el tránsito*. <https://www.who.int/>
- Prnewswire. (27 de Junio de 2011). *ZigBee Alliance and Society of Automotive Engineers Accelerates Electric Vehicle Connection to the Smart Grid*. Cisión. <https://www.prnewswire.com/news-releases/zigbee-alliance-and-society-of-automotive-engineers-accelerates-electric-vehicle-connection-to-the-smart-grid-114712274.html>
- Ramirez, I., Trejo, F. y Isaza, C. (2018). Seguimiento automático de trayectorias vehiculares usando el protocolo V2I. *Tecnología e Innovación*, 5 (15), 1-8.

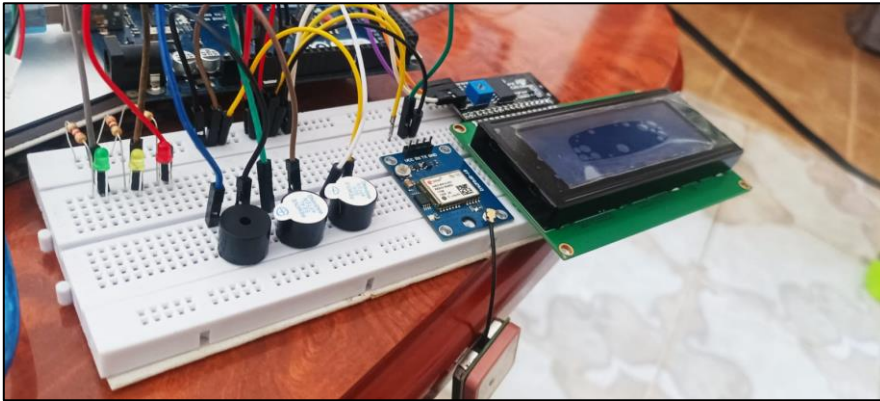
- https://www.ecorfan.org/bolivia/researchjournals/Tecnologia_e_innovacion/vol5num15/Revista_de_Tecnolog%C3%ADa_e_Innovaci%C3%B3n_V5_N15_1.pdf
- Revankar, P., Raut, K. y Kudikala M. (2018). Vehicle to vehicle communication using zigbee. *International Journal of Scientific & Engineering Research (IJSER)*, 9(2), 1-5, <https://www.ijser.org/researchpaper/Vehicle-to-vehicle-communication-using-zigbee.pdf>
- Rohde y Schwarz. (2021), Understanding UART, https://www.rohde-schwarz.com/lat/productos/prueba-y-medicion/essentials-test-equipment/digital-oscilloscopes/entendiendo-el-uart_254524.html
- Salazar, J. (2016). *Redes inalámbricas*. Techpedia https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01_R_ES.pdf
- Sallam, G. y Mahmoud, A. (2015). (Performance Evaluation of OLSR and AODV in VANET Cloud Computing Using Fading Model with SUMO and NS3. [2015 International Conference on Cloud Computing (ICCC)]. <https://doi.org/10.1109/CLOUDCOMP.2015.7149649>
- Sharma, K. y Dhir, Neha. (2014). *A Study of Wireless Networks: WLANs, WPANs, WMANs, and WWANs with Comparison*. *International Journal of Computer Science and Information Technologies (IJCSIT)*, 5 (6), 1-4. <https://ijcsit.com/docs/Volume%205/vol5issue06/ijcsit20140506200.pdf>
- Shereen, A., Ariffin, S y Faisal, N. (2013). Overview of Wireless Access in Vehicular Environment (WAVE) Protocols and Standards. *Indian Journal of Science and Technology*, 6 (7), 1-8. <https://dx.doi.org/10.17485/ijst/2013/v6i7.18>
- Siam, A. (27 de septiembre de 2019). *Conozca la comunicación de vehículo a infraestructura*. GEOTAB. <https://www.geotab.com>
- Stevens, T. y Marrs, T. (06 de diciembre de 2022). How V2V and V2X Technology Could Change the Auto World. Motortrend. <https://www.motortrend.com/features/what-is-v2v-v2x-technology/>
- Tokio School, (13 de marzo de 2021). Principales funciones de Cisco Packet Tracer. <https://www.tokioschool.com/noticias/cisco-packet-tracer/>
- Toyota. (30 de septiembre de 2015). Toyota Bringing Advanced ITS Technology to Mass-market Models. <https://global.toyota/en/detail/9676551>
- Trichias, K., Van den Berg, H., Heijenck, G., Jongh, J. y Litjens, R. (2012). Modeling and Evaluation of LTE in Intelligent Transportation Systems. Netherlands Organization for

<https://ris.utwente.nl/ws/files/5367233/Trichias12modeling.pdf>

- Uhlemann, E. (2017). The United States and Europe Advances Vehicle-to-Vehicle Deployment. *IEEE Vehicular Technology Magazine*, 12 (2), 18-22.
<https://doi.org/10.1109/MVT.2017.2680660>
- Universidad Miguel Hernández. (30 de mayo de 2014). Fritzing: un programa ‘open source’ para el diseño electrónico. Oficina de Software y Hardware Libre.
<https://oshl.umh.es/2014/05/30/fritzing-un-programa-open-source-para-el-diseno-electronico/>
- Vera, C., Barbosa, J. y Pabón, D. (2017). La Tecnología ZigBee estudio de las características de la capa física. *Scientia Et Technica*, 22(3), 238-245.
<https://www.redalyc.org/pdf/849/84954626002.pdf>
- Willke, T., Tientrakool, P. y Maxemchuk, N. (2009). A Survey of Inter-Vehicle Communication Protocols and Their Applications. *IEEE Communications Surveys & Tutorials*, 11 (2), 3-20, <https://doi.org/10.1109/SURV.2009.090202>.
- Zhou, Q. (2013). Thermal Comfort in Vehicles. [Tesis de grado, Facultad de Ingeniería y Desarrollo Sostenible. Universidad de Gävle].
<http://www.divaportal.se/smash/get/diva2:741222/FULLTEXT01.pdf>

11. Anexos.

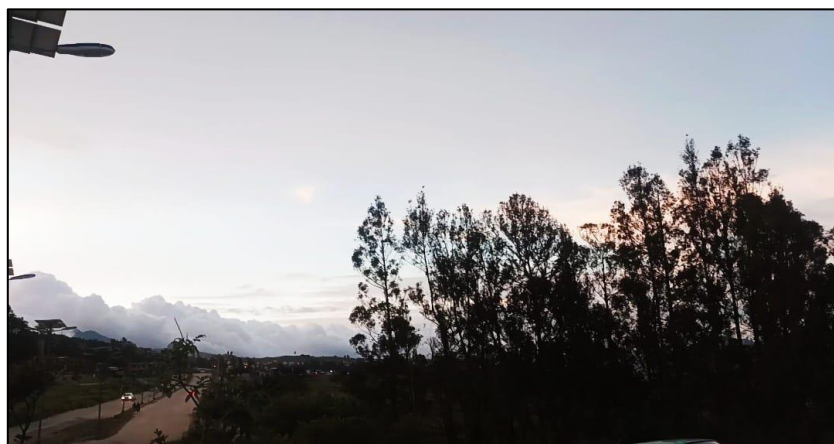
Anexo 1. Proceso de construcción del prototipo de comunicación V2V (Coordinador)



Anexo 2. Vehículos utilizados para la comunicación V2V



Anexo 3. Condiciones de prueba para el sensor de luminosidad



Nota. Se observa en la imagen, que está atardeciendo el día, por lo que la luminosidad disminuirá.

Anexo 4. Condiciones de prueba para el sensor de aceleración



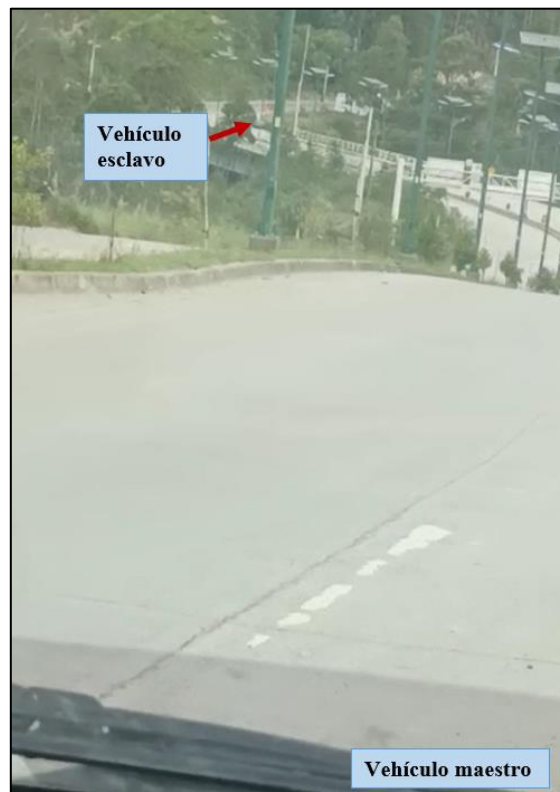
Nota. En la imagen, se observa que la carretera presenta un mal estado, con fallas geológicas, por lo que el vehículo esclavo frenará significativamente, y esto alertará al vehículo coordinador de que existe una falla más adelante

Anexo 5. Prueba de rango de cobertura a una distancia menor a 500 m



Nota. En la presente figura, se observa la prueba de ruta, que se ha realizado para comprobar el rango de cobertura de los módulos Xbee, además se aprecia que los vehículos se encuentran a una distancia aproximadamente de 30 m.

Anexo 6. Prueba de rango de cobertura a una distancia mayor a 500 m



Nota. En la presente Figura se muestra la prueba realizada para valorar el rango de cobertura de los módulos Xbee, a una distancia mayor a 500 m, en la imagen no se puede observar claramente el vehículo esclavo, puesto que es una distancia grande.

Anexo 7. Caja para el prototipo (vehículo emisor)



Nota. Se ha utilizado una caja para encapsular el prototipo, que ira montado en el vehículo emisor

Anexo 8. Caja para el prototipo (vehículo receptor)



Nota. Se ha utilizado una caja para encapsular el prototipo, que irá montado en el vehículo receptor, en el que se aprecia una pantalla LCD, y a la derecha el sensor de distancia.

Anexo 9. Datasheet del sensor de aceleración (ADXL345)

<https://pdf1.alldatasheet.com/datasheet-pdf/view/254714/AD/ADXL345.html>

Anexo 10. Datasheet del sensor de luminosidad (BH1750FVI)

<https://www.mouser.com/datasheet/2/348/bh1750fvi-e-186247.pdf>

Anexo 11. Datasheet del sensor de distancia Tf Mini Plus

https://cdn.sparkfun.com/assets/1/4/2/1/9/TFmini_Plus_A02_Product_Manual_EN.pdf

Anexo 12. Datasheet del sensor de temperatura y humedad

<https://www.digikey.at/htmldatasheets/production/2071184/0/0/1/dht11-humidity-temp-sensor.html>

Anexo 13. Datasheet del Módulo GPS

<https://datasheetspdf.com/pdf-file/866235/u-blox/NEO-6M/1>

Anexo 14. Programación del bloque del vehículo emisor

```
//Sensor BH1750 luminosidad
#include<BH1750.h>
#include<Wire.h>
int Sensorluz = 0x23;
//Declaracion del sensor luminosidad
```



```

BH1750 sensor;
//Declaraciones para sensor de aceleracion
#include <Adafruit_Sensor.h>
#include <Adafruit_ADXL345_U.h>
int ADXLAddress = 0x53;
Adafruit_ADXL345_Unified accel = Adafruit_ADXL345_Unified();
//Declracion de variables para sensor DHT temperatura
#include <DHT.h>
#include <DHT_U.h>
#define DHTPin 8
#define DHTTYPE  DHT11
DHT dht11(DHTPin, DHTTYPE);
//Tiempos en millis para cada sensor
unsigned long tiempo = 0; //Sensor aceleracion
long intervalo = 200;
unsigned long tiempo1 = 0; //Sensor temperatura y humedad
long intervalo1 = 3000;
unsigned long tiempo2 = 0; //Sensor de luminosidad
long intervalo2 = 4000;
//Declaraciones para modulo GPS
#include <TinyGPS.h>
float lat, lon;
TinyGPS gps; // Se crea GPS
void setup()
{
  Serial.begin(115200); //Inicializacion de puerto serial
  Serial.println("The GPS Signal:");
  Serial1.begin(9600); // Conexión del módulo GPS
  Serial3.begin(115200);
  Wire.begin();
  if (!accel.begin())
  {
    Wire.beginTransmission(ADXLAddress);
    Wire.endTransmission();
  }
}

```

```

Serial.println(" Sensor inactivo");
while (1); }
//Inicializa los sensores
sensor.begin();
dht11.begin();}
void loop()
{
//Recibir datos de modulo GPS por Serial1
while (Serial1.available()) { // Se lee lo existente en el puerto serial 1
if (gps.encode(Serial1.read())) //
{
gps.f_get_position(&lat, &lon); // Obteniendo la longitud y latitud
//Latitud
Serial.print(lat, 6);
Serial3.print(lat, 6); //se imprime la informacion a enviar
Serial.print("# ");
Serial3.print("# "); //se imprime la un caracter especial
//Longitud
Serial.print(lon, 6);
Serial.println("$ ");
Serial3.print(lon, 6);
Serial3.println("$ ");} }
//Sensor de aceleración
unsigned long fmillis = millis();
while ((fmillis - tiempo) >= intervalo) {
tiempo = fmillis;
sensors_event_t event;
accel.getEvent (&event);
Serial.print(event.acceleration.x);
Serial.println("");
Serial3.print(event.acceleration.x);
Serial3.println("");
//Sensor de temperatura y humedad}
unsigned long fmillis1 = millis();

```

```

while ((fmillis1 - tiempo1) >= intervalo1) {
  tiempo1 = fmillis1;
  int TempeC = dht11.readTemperature (); //Declaracion para temperatura
  int Hume = dht11.readHumidity(); //Declaracion para humedad
  //Temperatura
  Serial.print(TempeC);
  Serial.println("*");
  Serial3.print(TempeC);
  Serial3.println("*");
  //HUMEDAD RELATIVA//
  //Humedad %
  Serial.print(Hume);
  Serial.println("% ");
  Serial3.print(Hume);
  Serial3.println("% "); }
//Sensor de luminosidad
unsigned long fmillis2 = millis();
while ((fmillis2 - tiempo2) >= intervalo2) {
  tiempo2 = fmillis2;
  unsigned int lux = sensor.readLightLevel();
  Serial.print(lux);
  Serial.println(",");
  Serial3.print(lux);
  Serial3.println(",");}}

```

Anexo 15. Programación del bloque del vehículo receptor

```

//Declaracion de procesos
void processA();
void processB();
void processC();
//Pin de los pulsadores
const int PulsPin = 10;
const int PulsPin1 = 11;
const int PulsPin2 = 13;

```

```

//Variable del estado del pulsador
int EstadoPuls = 0;
int EstadoPuls1 = 0;
int EstadoPuls2 = 0;
//Variable para los timbres o zumbadores
int timbre = 7;
int timbre1 = 8;
int timbre2 = 9;
//Variable de temperatura
float temp;
float hume;
int Tempera;
//Matriz para recibir datos
char data[64];
uint8_t idx = 0;
//Pantalla LCD
#include <LiquidCrystal_I2C.h> // Se incluye pantalla LCD por I2C
LiquidCrystal_I2C lcd(0x27, 20, 4); // La direccion para la pantalla LCD 0x27
//Variable del módulo GPS
#include <TinyGPS.h>
//long lat,lon; //
float lat, lon;
float long1, lat1;
TinyGPS gps;
//Tiempo utilizado en millis para GPS
unsigned long tiempo = 0;
long intervalo = 100;
//Tempo utilizado en millis para luminosidad
unsigned long tiempo1 = 0;
long intervalo1 = 000;
//Tiempo temperatura y humedad
unsigned long tiempo2 = 0;
long intervalo2 = 7000;
void setup()

```

```

{
//Declaracion de pines de entrada y salida (leds,zumbadores, interruptor)
pinMode(PulsPin, INPUT);
pinMode(PulsPin1, INPUT);
pinMode(PulsPin2, INPUT);
pinMode (2, OUTPUT);
pinMode (3, OUTPUT);
pinMode (4, OUTPUT);
pinMode (5, OUTPUT);
pinMode (6, OUTPUT);
pinMode(timbre, OUTPUT);
pinMode(timbre1, OUTPUT);
pinMode(timbre2, OUTPUT);
//Iniciar la pantalla LCD
lcd.init();
//Encender la luz de fondo.
lcd.backlight();
lcd.clear();
//Inicializacion de Seriales
Serial.begin(115200);
Serial.println(" Señal de GPS:");
Serial2.begin(9600); // modulo GPS
Serial1.begin(115200); //recepcion de datos
Serial3.begin(115200); //sensor de distancia }
void loop()
{
processA(); //Recibir informacion de Arduino esclavo a Arduino maestro
processB(); //Datos de modulo gps
processC(); //Recolección de datos de sensor de distancia TF mini Plus
//Comparación de ubicaciones
float LONGITUD;
LONGITUD = (lon - (long1));
float LATITUD = (lat - (lat1));
Serial.println(LONGITUD, 7);
}

```

```

unsigned long fmillis = millis();
while ((fmillis - tiempo) >= intervalo) {
    tiempo = fmillis;
    EstadoPuls1 = digitalRead(PulsPin1);
    if (EstadoPuls1 == HIGH) {
        if ((LONGITUD >= -0.0003000) && (LONGITUD <= 0.0003001) && (LONGITUD !=
lon ) && (LONGITUD != -(long1)))) {
            digitalWrite(2, HIGH);
            digitalWrite(timbre1, HIGH);
            lcd.setCursor(0, 1);
            lcd.print(" Cuidado");
            lcd.setCursor(0, 0);
            lcd.print("vehiculo acercandose");} }
    if (EstadoPuls1 == HIGH) {
        if (( 0.0003000 <= LONGITUD >= -0.0003001) && (LONGITUD = lon ) &&
(LONGITUD = -(long1)))) {
            digitalWrite(2, LOW);
            digitalWrite(timbre1, LOW); } }
    if (EstadoPuls1 == LOW) {
        digitalWrite(2, LOW);
        digitalWrite(timbre1, LOW);
        lcd.setCursor(0, 0);
        lcd.print("          "); } }
//Programación para recibir datos de un arduino a otro Arduino
void processA()
{
    //Sensor de luminosidad
    while (Serial1.available() > 0) // Se comprueba lo que existe en el puerto serial1
    {
        char c = Serial1.read(); // se añade una variable y dentro de esta se lee lo que existe en el
puerto serial 1
        if (c == ',') // se realiza una condición, si es igual "," este es el caracter especial
para el sensor de luminosidad
        {

```

```

data[idx] = '\0';          // se crea una matriz de datos
float lum = atoi(data);    // se crea una variable float, en la cual contendra los datos del
sensor de luminosidad

Serial.print(" lum : ");
Serial.println(lum);
idx = 0;
unsigned long fmillis1 = millis();
while ((fmillis1 - tiempo1) >= intervalo1) {
    tiempo1 = fmillis1;
    EstadoPuls = digitalRead(PulsPin);
    if (EstadoPuls == HIGH) {
        if (lum <= 130) {
            digitalWrite (3, HIGH);
            lcd.setCursor(0, 1);
            lcd.print("Baja luminosidad");
            lcd.setCursor(0, 2);
            lcd.print("Encender luces"); } }
    if ((EstadoPuls == HIGH) && (lum >= 131)) {
        digitalWrite(3, LOW);
        lcd.setCursor(0, 1);
        lcd.print("          ");
        lcd.setCursor(0, 2);
        lcd.print("          "); }
    if (EstadoPuls == LOW) {
        digitalWrite(3, LOW);
        lcd.setCursor(0, 1);
        lcd.print("          "); }}}

//Sensor de aceleración
else if (c == '/') // caracter especial para el sensor de aceleracion
{
    data[idx] = '\0';
    float ace = atof(data);
    Serial.print(" ace : ");
    Serial.println(ace);
}

```

```

idx = 0;
if (ace <= -3.9) {
  lcd.setCursor(0, 0);
  lcd.print("!Posible Peligro!");
  delay(400);
  lcd.setCursor(0, 1);
  lcd.print("Reducir velocidad");
  digitalWrite (timbre1, HIGH);
  digitalWrite(2, HIGH); }
if (ace >= -4.0) {
  digitalWrite (timbre1 , LOW);
  digitalWrite(2, LOW); }}
//Sensor de temperatura
else if (c == '*') //caracter especial para datos de temperatura
  data[idx] = '\0';
  float temp = atoi(data);
  Tempera = int(temp);
  Serial.print(" Tempe : ");
  Serial.println(temp);
  lcd.setCursor(5, 3);
  lcd.print("Temp: ");
  lcd.setCursor(10, 3);
  lcd.print(temp);
  lcd.setCursor(15, 3);
  lcd.print(" C");
  idx = 0; }
//Sensor de Humedad
else if (c == '%') //caracter especial para los datos de humedad
{
  data[idx] = '\0';
  float hume = atoi(data);
  Serial.print(" Humed: ");
  Serial.println(hume);
  idx = 0;

```



```

unsigned long fmillis2 = millis();
while ((fmillis2 - tiempo2) >= intervalo2) {
    tiempo2 = fmillis2;
    EstadoPuls2 = digitalRead(PulsPin2);
    if (EstadoPuls2 == HIGH) {
        if ((hume >= 60 ) && (Tempera <= 23 )) {
            digitalWrite (4, HIGH);
            lcd.setCursor(0, 1);
            lcd.print("Conduccion hostil");
            lcd.setCursor(0, 2);
            lcd.print("Encender AC"); }
        if ((hume <= 30) && (Tempera >= 28)) {
            digitalWrite (4, HIGH);
            lcd.setCursor(0, 1);
            lcd.print("Conduccion hostil");
            lcd.setCursor(0, 2);
            lcd.print("Encender AC");
        }
    }
    if (EstadoPuls2 == LOW) {
        digitalWrite(4, LOW);
        lcd.setCursor(0, 2);
        lcd.print("          "); } }
//Modulo GPS de módulo esclavo
//Longitud
else if (c == '$') //carácter especial para el módulo GPS
{
    data[idx] = '\0';
    long1 = atof(data);
    Serial.print(" long1 : ");
    Serial.println(long1, 6);
    idx = 0;}
//Latitud
else if (c == '#')

```

```

{
  data[idx] = '\0';
  lat1 = atof(data);
  Serial.print(" lat1 : ");
  Serial.println(lat1, 6);
  idx = 0;}
else
{
  data[idx++] = c;}}}
void processB()
{
  while (Serial2.available() > 0) { // //se lee el puerto serial 2
    if (gps.encode(Serial2.read())) //
    {
      gps.f_get_position(&lat, &lon); // Obteniendo latitud y longitud
      //Latitud
      Serial.print("latitud: ");
      Serial.println(lat, 6);
      //Longitud
      Serial.print("longitud: ");
      Serial.println(lon, 6);}}}
void processC()
{
  //Sensor de distancia
  int distancia = 0;
  int strength = 0;
  getTFminiData(&distancia, &strength);
  if (distancia)
  {
    Serial.print("distancia: ");
    Serial.println(distancia);
    if (distancia <= 150) {
      lcd.setCursor(0, 0);
      lcd.print("Distancia insegura");

```

```

    digitalWrite (timbre, HIGH);
    digitalWrite(5, HIGH);}
else if (distancia >= 151) {
    digitalWrite (timbre, LOW);
    digitalWrite (5, LOW);
    lcd.setCursor(0, 0);
    lcd.print("          ");}}
void getTFminiData(int* distancia, int* strength) // obtenido de https://github.com/
{
    static char i = 0;
    char j = 0;
    int checksum = 0;
    static int rx[9];
    while (Serial3.available())
    {
        rx[i] = Serial3.read();
        if (rx[0] != 0x59)
        {
            i = 0;}
        else if (i == 1 && rx[1] != 0x59)
        {
            i = 0;}
        else if (i == 8)
        {
            for (j = 0; j < 8; j++)
            {
                checksum += rx[j];}
            if (rx[8] == (checksum % 256))
            {
                *distancia = rx[2] + rx[3] * 256;
                *strength = rx[4] + rx[5] * 256;}
            i = 0;}
        else
        {

```

i++;}}

Anexo 16. Certificado de traducción



Lic. Mónica Guarnizo Torres.
SECRETARIA DE "BRENTWOOD LANGUAGE CENTER"

CERTIFICA:

Que el documento aquí compuesto es fiel traducción del idioma español al idioma inglés del trabajo de titulación denominado "DISEÑO E IMPLEMENTACIÓN DE UNA RED DE COMUNICACIÓN VEHÍCULO A VEHÍCULO (V2V), MEDIANTE EL PROTOCOLO DE COMUNICACIÓN ZIGBEE", del estudiante AARÓN ANTONIEL SARANGO CALVA, con cédula de identidad No. 1105619249, egresado de la carrera de Ingeniería en Mecánica Automotriz.

Lo certifica en honor a la verdad y autoriza al interesado hacer uso del presente en lo que a sus intereses convenga.

Loja, 05 de abril de 2024

Lic. Mónica Guarnizo Torres
SECRETARIA DE B.L.C.

