

Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Ciudad de México

ESCUELA DE INGENIERÍA Y CIENCIAS  
DEPARTAMENTO DE MECATRÓNICA

INGENIERÍA EN TELECOMUNICACIONES Y SISTEMAS  
ELECTRÓNICOS



**TECNOLOGICO  
DE MONTERREY®**

**Mejora del dispositivo de corto alcance para comunicaciones V2X**

**Juan Gascón Repullés A01756117**

**Centro de Investigación en Microsistemas y Biodiseño**

Asesor:

Dr. Martin Rogelio Bustamante Bello

Sinodales:

Dr. Javier Izquierdo Reyes  
Dr. Edgar Santoyo Castelazo  
Dr. Renato Galluzi Aguilera

Ciudad de México a 2 de Junio del 2020

# Índice de figuras

1.1. Digrama de Gantt del proyecto . . . . .	8
2.1. Componentes primer prototipo . . . . .	11
2.2. Interfaz de corto alcance del nuevo prototipo . . . . .	15
2.3. Jerarquía de red en el dispositivo NRF24L01 . . . . .	16
2.4. Diagrama de comunicación básica NRF24L01 . . . . .	16
2.5. Diagrama de flujo del programa . . . . .	18
2.6. Diagrama de flujo del canal de control para el descubrimiento de dispositivos	19
2.7. Diagrama de flujo para la recepción de mensajes del canal de control . . . . .	20
2.8. Diagrama de flujo para el canal de datos de los RSU . . . . .	21
2.9. Diagrama de flujo para el canal de datos de los OBU . . . . .	23
2.10. Estructura del paquete ACK con payload . . . . .	23
2.11. Software CAD de diseño electrónico EAGLE . . . . .	25
2.12. Pantallas principales del programa EAGLE . . . . .	25
2.13. Vista <i>Schematic</i> de los componentes . . . . .	26
2.14. Vista <i>Schematic</i> del prototipo con NRF24L01 y alimentación . . . . .	27
2.15. Vista <i>BoardFile</i> del prototipo con NRF24L01 y alimentación . . . . .	28
2.16. Previsualización de la PCB . . . . .	29
2.17. Resumen características técnicas de la PCB . . . . .	29
2.18. Imagen del primer prototipo conexionado sobre protoboard . . . . .	30
2.19. Imagen del nuevo prototipo conexionado sobre la PCB . . . . .	31
2.20. Solución para la prevención de accidentes en intersecciones . . . . .	32
2.21. Representación de vías rápidas con más de un carril . . . . .	33
2.22. Comunicación objetivo entre vehículos próximos . . . . .	34
2.23. Ordenación de los vehículos tras pasar por cada RSU . . . . .	34
2.24. Reordenación de los vehículos tras pasar por nuevo RSU . . . . .	36
2.25. Comunicación básica Tx-Rx NRF24L01 con tasa 250 kb/s . . . . .	38
2.26. Comunicación básica Tx-Rx NRF24L01 con tasa 1 Mb/s . . . . .	38
2.27. Comunicación básica Tx-Rx NRF24L01 con tasa 2 Mb/s . . . . .	39
2.28. Comunicación Tx-Rx y Rx-Tx mediante ACK con payload . . . . .	40
2.29. Comparación latencia entre Xbee y NRF24L01 . . . . .	41

2.30. Prueba completa con duración de 50 segundos . . . . .	42
2.31. Vista de la prueba al prototipo entre 20 y 30 segundos . . . . .	43
2.32. Vista de la prueba al prototipo entre 24,8 y 25,8 segundos . . . . .	43
2.33. Vista de la prueba al prototipo entre 24,8 y 24,94 segundos . . . . .	44

# Índice tablas

2.1. Coste aproximado. . . . .	46
3.1. Tabla comparación características entre prototipos . . . . .	49

# Índice general

<b>Lista de figuras</b>	<b>2</b>
<b>Lista de tablas</b>	<b>3</b>
<b>1. Introducción</b>	<b>2</b>
1.1. Descripción de la empresa . . . . .	3
1.2. Antecedentes de la problemática (contexto) . . . . .	4
1.3. Justificación Técnica . . . . .	5
1.4. Objetivos . . . . .	6
1.5. Alcances y entregables del proyecto . . . . .	7
1.6. Impactos del proyecto . . . . .	7
1.7. Plan de actividades . . . . .	8
<b>2. Desarrollo</b>	<b>9</b>
2.1. Antecedentes de la solución . . . . .	9
2.1.1. Proyectos previos desarrollados en el CIMB/ADMAS . . . . .	9
2.1.2. Estándares internacionales: IEEE 802.11p . . . . .	12
2.1.3. Sistemas propios . . . . .	12
2.2. Desarrollo ingenieril de la solución . . . . .	13
2.2.1. Conclusiones prototipo anterior . . . . .	13
2.2.2. Requerimientos de la comunicación de corto alcance entre vehículos .	13
2.2.3. Mejora de la comunicación de corto alcance del prototipo . . . . .	14
2.2.4. Diseño del nuevo protocolo de comunicación de corto alcance . . . . .	17
2.2.5. Diseño de la PCB del prototipo . . . . .	24
2.2.6. Solución para escenarios reales . . . . .	31
2.3. Plan de pruebas realizado . . . . .	37
2.3.1. Pruebas al dispositivo NRF24L01 . . . . .	37
2.3.2. Pruebas comparativas entre Xbee y NRF24L01 . . . . .	40
2.3.3. Prueba del nuevo prototipo . . . . .	41
2.4. Análisis técnico de los resultados . . . . .	44
2.5. Recursos utilizados . . . . .	45

2.6.	Análisis económico del proyecto . . . . .	46
2.7.	Análisis de dilemas éticos . . . . .	46
2.7.1.	Conducción autónoma . . . . .	46
2.7.2.	Comunicación entre vehículos y entorno . . . . .	47
2.7.3.	Conclusiones dilema ético . . . . .	47
<b>3.</b>	<b>Conclusiones</b>	<b>49</b>
3.1.	Conclusiones técnicas del proyecto . . . . .	49
3.2.	Conclusiones de los miembros del proyecto . . . . .	50
3.2.1.	Conclusiones - Eloi Pascual . . . . .	50
3.2.2.	Conclusiones - Juan Gascón . . . . .	51
3.3.	Trabajo a futuro . . . . .	52
	<b>Bibliografía</b>	<b>53</b>
<b>4.</b>	<b>Anexos</b>	<b>55</b>
4.1.	AI: Poster . . . . .	56
4.2.	AII: Primer prototipo . . . . .	57
4.3.	AIII: Estándar ZigBee . . . . .	59
4.4.	AIV: ETSI TS 102 894-2 V1.2.1 . . . . .	60
4.5.	AV: Datasheet XBee . . . . .	61
4.6.	AVI: Datasheet NRF24L01 . . . . .	62
4.7.	AVII: Especificaciones XBee . . . . .	63
4.8.	AVIII: Especificaciones NRF24L01 . . . . .	64
4.9.	AIX: Datasheet ESP-32 DEVKIT V1 . . . . .	65
4.10.	AX: Código y pruebas realizadas . . . . .	66

En el presente documento nos disponemos a la exposición de la memoria relativa al trabajo Improvements in Short Range and cellular V2X communications, desarrollado en el CIMB (Computing Intelligence, Mechatronics and Biodesign) del Instituto Tecnológico y de Estudios Superiores de Monterrey durante el semestre Enero - Julio del año 2020.

Se realizará un análisis sobre el trabajo desempeñado, donde además se mostrarán los resultados finales y se incluirán otros apartados que brindarán de una visión de conjunto para una mayor comprensión del proyecto por parte del lector, con la intención de plasmar en un solo documento todo el trabajo realizado y el contexto que enmarca el desarrollo de este proyecto.

Aclaradas estas cuestiones previas, se brinda esta memoria que espera servir de guía de entendimiento del proyecto o de explicación para futuras modificaciones.

# Capítulo 1

## Introducción

Cuando imaginamos un futuro mejor, no podemos desligar este de una mejora en la calidad de vida de los humanos que habitarán el mundo. Esta mejora de la calidad de vida se puede ver reflejada en diferentes ámbitos y aspectos: una mejor salud, una menor cantidad de estrés, unas mejores condiciones laborales, etc. Se espera que la evolución de las tecnologías nos acerque, de una manera u otra, a estos futuros posibles y que mediante pequeñas mejoras en ámbitos cotidianos el estilo de vida de las personas sufra ligeros cambios que acaben resultando en una mejor calidad de vida. Uno de estos cambios que espera revolucionar nuestro día a día será la conducción autónoma.

La conducción autónoma ha sido tema de interés creciente durante los últimos años, siempre presente en noticias y telediarios debido a los rápidos avances que se están produciendo bien por marcas privadas, bien por equipo investigador universitario. Pero lo que está claro es que la conducción autónoma está cerca de convertirse en una realidad cotidiana.

Una conducción autónoma eficiente repercutirá directamente en la vida de las personas en aspectos que todavía no sabemos, pero podemos esperar mejoras significativas en diferentes ámbitos.

De acuerdo con BMW, la empresa alemana de fabricación de vehículos, de la conducción autónoma cabe esperar al menos estos tres resultados:

- Tráfico con cero emisiones
- Transporte asequible, siempre disponible o no discriminatorio
- Ciudades más humanizadas o ahorro de recursos

Como podemos ver, estos tres puntos están muy alineados con los valores deseados para la materia en cuestión (Proyecto Integrador en Desarrollo Sustentable) y este proyecto

pretende sumar esfuerzos en el ámbito de las comunicaciones y los vehículos autónomos.

El proyecto en cuestión pretende la creación de una comunicación entre vehículos y su entorno fiable, rápida, predecible y segura, que ayude en la detección de posibles colisiones y sea capaz de crear un mapa en tiempo real del tráfico y la situación en las carreteras.

De aquí nace el proyecto Short Range and Cellular V2X communications, que pretendía realizar los objetivos anteriormente comentados a través de una comunicación V2X (vehicles to X, es decir, de vehículo a cualquier otra cosa: infraestructura, peatones, otros vehículos, nube, etc.).

## 1.1. Descripción de la empresa

Todo el proyecto (la primera parte y esta segunda parte) se ha realizado en el CIMB.

*“El CIMB es una iniciativa generada por un grupo de profesores y alumnos del Tecnológico de Monterrey en Campus Ciudad de México en el año 2009. Este grupo, busca generar proyectos para la mejora de la calidad de vida de la sociedad mexicana y una mejora de la Experiencia del Usuario (UX) ante el uso de nuevas tecnologías. El laboratorio trabaja a través de la generación de proyectos de Postgrados y Pregrado sobre las distintas líneas de investigación de enfoque.”*

En concreto, dentro de los diferentes proyectos que se llevan a cabo en este laboratorio, el proyecto Short Range and Cellular V2X communications, y esta segunda parte Improvements in Short Range V2X communication pertenecían al proyecto ADMAS (Advanced Driver Monitoring for Assistance System). La descripción de este que podemos encontrar en su web es la siguiente:

*“El sistema pretende convertir un coche con nivel de autonomía 2, a un nivel 3 o 4, es decir convertir un automóvil común, en uno semi-autónomo. La semi-autonomía se logrará mediante sensores y algoritmos de inteligencia artificial que permitan tomar el control del vehículo cuando el conductor no se encuentre en condiciones de conducción segura.”*

En concreto, nuestro proyecto formaba parte del área de investigación de comunicación de ADMAS. La descripción ofrecida para esta área de investigación del programa en la web del proyecto ADMAS es la siguiente:

*“Se desarrolla la investigación, simulación e implementación de sistemas de comunicación entre los vehículos a través de protocolos y estándares internacionales, para la generación de una red comunicación inteligente de autos. Se desea el envío y recepción de datos críticos y no críticos en el estatus de los autos para una mejora en la movilidad vehículos.”*

## **1.2. Antecedentes de la problemática (contexto)**

Una vez entendido el grupo de trabajo que enmarca el proyecto, es más fácil entender el contexto de este proyecto en específico.

Para la realización de un vehículo semi-autónomo, objetivo del grupo ADMAS, se necesitan varias partes básicas que abarcan tanto cuestiones mecánicas, como eléctricas o de comunicaciones. Al igual que un coche necesita un motor que proporcione potencia y tracción a las ruedas para avanzar, una conducción autónoma necesita de un sistema que dote al vehículo de datos para poder tomar decisiones. Estas decisiones, tradicionalmente tomadas por el conductor, deben ser sustituidas por decisiones tomadas por el propio vehículo, de manera que este pueda operar de forma autónoma o semiautónoma simulando una conducción humana o incluso mejorándola.

Así pues, el vehículo autónomo necesitará de datos que le permitan tomar decisiones y que debe recabar de su entorno a través de receptores. Estos receptores pueden ser cámaras, sensores o sistemas de comunicación, que permitan al vehículo la obtención de los datos que le permitirán tomar una respuesta frente a un estímulo.

Para estas cuestiones, el grupo ADMAS ya trabajaba, entre otros proyectos, en la creación de visión artificial, que permitiese al vehículo identificar patrones de vía y circular por un circuito a través del procesado de imágenes obtenidas de cámaras. También se trabajaba en sistemas capaces de operar la dirección de un vehículo, para poder cambiar la dirección de este. Como vemos, el grupo dividía el objetivo principal de crear un vehículo autónomo en varias tareas de menor tamaño encargadas a grupos de varios colaboradores.

Entre esas tareas de menor tamaño, se encargó la constitución de un sistema de comunicación entre vehículos, ya que este sistema podría alertar de riesgo de colisión o corroborar datos obtenidos mediante otros sensores por otras vías. Esto hace que la comunicación entre los vehículos y su entorno (V2X) sea tan interesante. Una cámara puede reconocer un semáforo en rojo, pero si podemos tener una comunicación con el semáforo, siempre tendremos una doble verificación, dotando de mayor robustez y fiabilidad a la toma de decisiones del vehículo.

A esto, se le suman otros beneficios, como el hecho de poder advertir de un accidente si es avistado por la cámara y poder notificarlo en tiempo real y subir esta información a la nube (o a quién corresponda) junto con una ubicación exacta del lugar del accidente. También se podría comunicar con otros vehículos en caso de que el vehículo se hubiese salido de la vía o sufriese una avería, pudiendo así, aportar información que las cámaras no pueden captar y evitar accidentes. Otro ejemplo de ayuda en escenarios sin visión, sería el escenario de un cruce, donde el vehículo no capta la presencia de otros vehículos, pero podría captar una comunicación que procesara su ubicación y la dirección en la que se mueve y por tanto crear una imagen virtual de la posición del otro/otros vehículos.

Por tanto el V2X se consolidaba como un requerimiento para el correcto funcionamiento de un vehículo autónomo y por tanto se inició una línea de investigación, a través de varios proyectos, destinada a lograr una comunicación exitosa entre vehículos y entorno.

Tras varios proyectos, que posteriormente serán analizados con mayor detalle, el reto del actual proyecto era aterrizar las simulaciones y los trabajos teóricos anteriores en un prototipo funcional, capaz de dar un paso más hacia una comunicación V2X requerida para el proyecto ADMAS. Más concretamente, el de brindar a esta línea de investigación centrada en los estudios teóricos y las simulaciones, una vía experimental práctica más plausible, donde poder comprobar el comportamiento teórico desarrollado en los estudios anteriores.

### **1.3. Justificación Técnica**

Una vez entendidos los trabajos previos, podemos centrarnos en entender el reto tecnológico de la creación de un estándar de comunicación entre vehículos, y más concretamente, el reto que supone la mejora de un sistema de comunicación partiendo de un sistema ya establecido.

En este caso, partimos de la base de que no se encuentran sistemas establecidos internacionalmente o estándares que se hayan servido como referencia, ya que las tecnologías asociadas al V2X se encuentran en fase de desarrollo y sin acuerdo por parte de grandes marcas. Esto es debido a que muchos investigadores esperan a la llegada del 5G, tecnología que revolucionará el ámbito de las comunicaciones, pero más allá de esto, todos los estudios apuntan a que el 5G tiene grandes problemas de cobertura ya que al trabajar a altas frecuencias, su alcance se limita a metros y no a kilómetros. Esto generará desigualdad entre regiones que dispongan del 5G y las regiones que no lo hagan, y la llegada de esta tecnología no será igual en todas las regiones. Por ejemplo, regiones como la Selva Lacandona, en el estado de Chiapas, sigue sin tener cobertura móvil, o determinadas zonas en Oaxaca, siguen sin tener acceso a tecnología 4G. Por tanto, se trata de desarrollar un sistema capaz de comunicarse

entre los vehículos sin necesidad de cobertura móvil o de depender de tecnología externa a los vehículos.

Depender únicamente de tecnología que pueda ser instalada en un vehículo provocaría que lugares remotos, o simplemente puntos ciegos (como túneles subterráneos o zonas montañosas) pudiesen dejar de ser zonas de desconexión total en caso de accidentes o en caso de una futura conducción autónoma.

## 1.4. Objetivos

El objetivo general de este proyecto de un año es la realización un prototipo que permita la comunicación entre vehículos, integrando las siguientes características:

- **Comunicación de corto alcance:** Comunicación realizada mediante dispositivos de RF, en la que se requieren latencias entre mensajes pequeñas (menos de 100 ms) para que la comunicación sea efectiva y de utilidad.
- **Comunicación de largo alcance:** Comunicación realizada mediante la red celular. Únicamente se debe diseñar la interfaz para poder utilizar la red, ya que las características de la misma depende de los proveedores de estos servicios. Se cree que mejorará considerablemente con el despliegue del 5G.

A partir de las conclusiones obtenidas del primer prototipo y analizar sus debilidades, en esta segunda parte se va realizar específicamente:

- **Mejora de la comunicación de corto alcance:** Mediante la división de esta comunicación en dos canales diferenciados: control y datos.
  - **Canal de control:** Canal dedicado a dirigir y coordinar las comunicaciones. Se lleva a cabo con el dispositivo de RF Xbee, que nos permite conocer en todo momento la cantidad de vehículos a nuestro alrededor.
  - **Canal de datos:** Este canal se dedica al envío de datos en sí. Para ello se ha seleccionado el dispositivo NRF24L01, que nos permite comunicaciones con latencias entre mensajes bajas (menos de 10ms).
- **Diseño PCB del prototipo:** Diseño de la placa de circuito impreso del prototipo mediante el programa de modelado electrónico EAGLE. En el que se posibilita la agrupación de todos los componentes del prototipo en un mismo circuito.
- **Diseño de las pruebas en escenario real:** Diseño de las pruebas en escenario real para comprobar el desempeño del prototipo. En un principio se esperaba poder realizar el

mismo tipo de pruebas que se le realizó al primer prototipo, pero debido a la imposibilidad de realizar pruebas en el exterior con el confinamiento. Se van a explicar cómo se iban a llevar a cabo, por sí en un futuro es posible realizarlas.

## 1.5. Alcances y entregables del proyecto

Para este proyecto se van a realizar los siguientes entregables:

- Presentación técnica PC: Consiste en una presentación del proyecto desde un punto de vista más técnico frente a un tribunal, en el que se realizan correcciones para el entregable de Presentación ejecutiva PA.
- Presentación ejecutiva PA: También consiste en una presentación pero esta vez se realiza de una forma menos técnica.
- Poster: Descripción del proyecto en formato póster.
- Reporte final: Este reporte, en el que se explica todo lo relacionado con el proyecto.

## 1.6. Impactos del proyecto

Impacto de aprendizaje:

Este proyecto en comparación con los 2 trabajos anteriores se ha realizado desde el punto de vista más práctico que teórico. Debido a que en los dos trabajos anteriores se realizó el estudio de las distintas tecnologías con las que se podía llevar a cabo la solución a la comunicación entre vehículos. Por ello se decidió escoger una de estas tecnologías para realizar un prototipo y analizar los resultados obtenidos. Al consistir este trabajo en desarrollar un prototipo prácticamente desde cero, y no estar familiarizados con ello. Se aprendió:

- Aprender a dividir el proyecto en pequeñas tareas y definir objetivos en el tiempo, es decir a organizar el proyecto.
- Encontrar información relevante para el proyecto de cada uno de los componentes a partir de búsquedas en distintos foros y datasheet.
- Adaptación rápida a uso de tecnologías completamente nuevas para nosotros.

## 1.7. Plan de actividades

En esta sección se adjunta el diagrama de Gantt realizado para llevar a cabo el proyecto.



Figura 1.1: Diagrama de Gantt del proyecto

# Capítulo 2

## Desarrollo

### 2.1. Antecedentes de la solución

#### 2.1.1. Proyectos previos desarrollados en el CIMB/ADMAS

Como ya se ha evidenciado en los antecedentes de la problemática, el proyecto de comunicación entre vehículos ya tiene un largo recorrido en el CIMB y pretendía la creación de un sistema de comunicación V2X efectivo y eficiente que pudiera dar soporte a los sensores y sistemas de conducción autónoma que se encontraban en funcionamiento (o en desarrollo) en otras de las áreas de investigación de ADMAS.

A continuación, como se adelantó en los antecedentes de la problemática, se realizará un repaso de los trabajos previos relacionados con el presente y su importancia para el proyecto en líneas generales, así como sus resultados y conclusiones. Finalmente, se realizará un repaso más detallado de la primera parte de este proyecto para entender mejor qué problemática trata de resolver esta segunda parte del proyecto.

Como se ha comentado, empezamos por el proyecto *Comunicaciones Car to Car (C2C)* realizado en Noviembre de 2017 por Oscar Emanuel Vicente Mijangos, Esteban Hernan Alcocer Souza y Alfonso Ortíz Jiménez

En este proyecto el objetivo general era “*el diseño e implementación de una infraestructura de redes de telecomunicación enfocada a las comunicaciones automotrices la cual sirva como plataforma para desarrollar aplicaciones enfocadas a la seguridad, tráfico vial y comercial.*”

Y sus objetivos específicos eran:

1. Revisar los problemas de las comunicaciones inalámbricas C2X y C2C.

2. Investigar los antecedentes y protocolos de comunicación utilizados en las comunicaciones C2C.
3. Comparar las diferentes tecnologías utilizadas así como los dispositivos en existencia.
4. Simular el desempeño de la tecnología elegida en un entorno real simulado.
5. Probar el desempeño de la tecnología elegida a velocidades altas en tiempo real.
6. Desarrollar dos aplicaciones viales con base en la infraestructura propuesta.
7. Crear una infraestructura de comunicaciones Car to Car en México que esté estandarizada internacionalmente.

En el proyecto en cuestión se realizaron varias simulaciones que determinaban que las mejores soluciones para una comunicación C2C (car to car) exitosa eran la utilización del estándar 802.11p y el protocolo de comunicación ZigBee.

Proseguimos con el proyecto Comunicación Car to Car realizado en Noviembre de 2018 por Alec García Barba, Ángel Ortiz Jiménez, César Alejandro Hidalgo y Aldo Chávez Gallardo.

En resumen, este trabajo trataba de poner a prueba la utilización del estándar 802.11p y el protocolo Zigbee a través de simulaciones que constataran si estos eran aptos para su utilización en comunicaciones V2X. Así pues, realizar diferentes simulaciones de escenarios que pusieran a prueba estos protocolos en diferentes escenarios y determinar la mejor alternativa dependiendo de los resultados obtenidos.

Realizadas las simulaciones correspondientes, se determinó que ambos eran aptos y capaces de realizar una comunicación V2X, pero se remarcaba la optimización del estándar 802.11p para la comunicación entre vehículos, mientras que el protocolo ZigBee está diseñado para IoT, es decir, para la interconexión de sensores y dispositivos remotos, no para la comunicación entre vehículos.

Como continuación a estos dos proyectos, surgió el proyecto *Short Range and Cellular V2X communications*, que es la primera parte de este proyecto y fue desarrollada en el semestre Agosto - Diciembre del año 2020.

En esta primera parte, se proponía el diseño de un prototipo capaz de realizar una comunicación V2X que satisficiera los requerimientos técnicos necesarios y utilizando herramientas de desarrollo gratuitas que permitieran hacer de este prototipo una tecnología abierta

a futuras mejoras.

Para ello, se marcaba como objetivo establecer las bases del protocolo de comunicación y los mensajes que serían necesarios enviar en esta comunicación. También, crear un sistema de comunicación de largo alcance que permitiese subir datos a la nube.

Esta necesidad de construir un prototipo alejado de los proyectos anteriores, que se centraban en la realización de simulaciones, se atribuye en la primera parte de este proyecto, al hecho de que un estudio del INEGI evidenciaba que aproximadamente el 90 % de los accidentes que se registraban en México sucedían en cruces, y la incorporación de un dispositivo que fuera capaz de comunicarse en estos escenarios podría haber evitado 214973 accidentes y 386 decesos (datos de accidentes en intersecciones en 2018 del INEGI).

Por tanto, centrados en la solución de este escenario, se desarrolló un prototipo de comunicaciones entre vehículos que incluía mensajes recogidos en un estándar así como los componentes necesarios para la comunicación de corto y largo alcance.

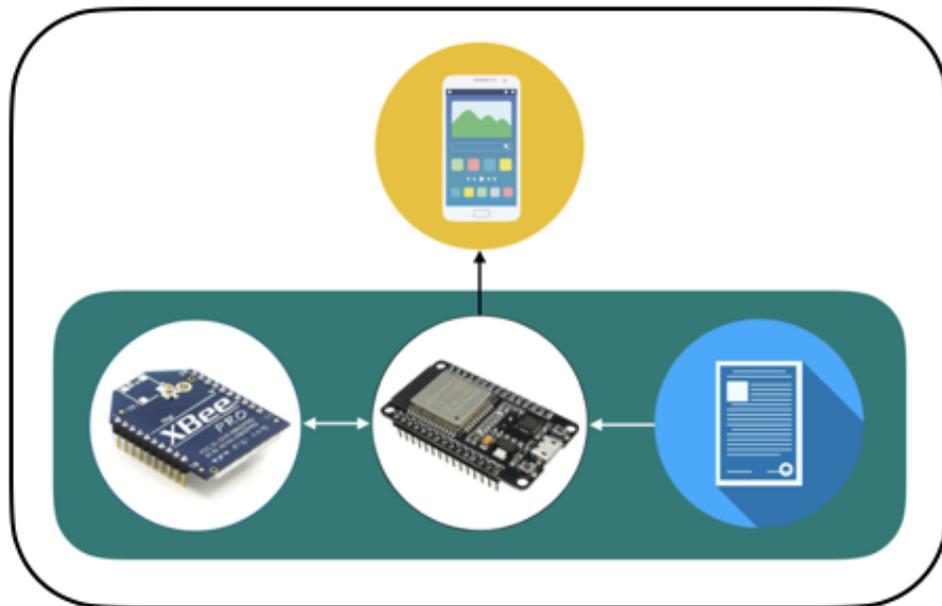


Figura 2.1: Componentes primer prototipo

La comunicación se llevaba a cabo a través del Xbee, debido a los informes de los proyectos anteriores y un dispositivo ESP-32 devkit v.1 servía de microcontrolador que controlara los mensajes y determinara cuales eran relevantes para subirlos a la nube. El prototipo fue catalogado como funcional tras pruebas en un entorno controlado y se le realizaron pruebas en un entorno semi-controlado.

Estas pruebas consistían en que tres vehículos equipados con el dispositivo (OBU, On Board Unit) realizaron un circuito cerrado en el aparcamiento del Campus Disney del campus del ITESM CCM. Además, se situó un dispositivo fijo fuera del circuito. Todos los dispositivos enviaron varios mensajes desde que se incorporaron y se registraron en hojas de datos con su TimeStamp de llegada correspondiente.

Tras el análisis de los datos, se evidenció que el alcance del dispositivo cumplía a la perfección los objetivos deseados. También el enlace de largo alcance cumplía con las expectativas previas, pero en cuanto a la latencia de los mensajes, esta resultó ser de unos 150 ms para una comunicación fiable y además se podían transmitir pocos caracteres y no permitía futuras mejoras por la limitación del propio dispositivo XBee.

Esto, provocaba que no se pudiese confiar en este dispositivo para una comunicación confiable entre vehículos, un paso vital para el desarrollo de los vehículos autónomos, por tanto a lo largo de este proyecto (y esta memoria) se evaluará y desarrollará una solución que supla estas carencias de los trabajos previamente realizados, especialmente del último prototipo desarrollado en la primera parte de este proyecto.

Los términos más importantes en los que se deberían centrar los esfuerzos sería en mejorar la comunicación, de manera que esta sea más rápida y fiable, para poder desarrollar un sistema más seguro, más eficaz y en el que poder confiar una comunicación de tal calibre. Así pues, se pretende dejar de usar una placa de prototipos y dotar al sistema de robustez y de facilidad para el transporte (por los problemas que suponía el movimiento del dispositivo).

### **2.1.2. Estándares internacionales: IEEE 802.11p**

En 2010, el IEEE aprobó el lanzamiento del estándar 802.11p, que pretendía convertirse en una suerte de sistema WiFi adaptado para comunicaciones V2X. Este estándar, se basa en el uso de dos canales especiales (una cifrada y una libre) en la banda de los 5.9 GHz, y es independiente de la llegada de la tecnología del 5G. A pesar de estar dotada del respaldo del IEEE, en los trabajos previos vimos que en las conclusiones de estos proyectos se afirmaba que 802.11p todavía no estaba listo para convertirse en el sistema a utilizar en este tipo de comunicaciones.

### **2.1.3. Sistemas propios**

Como ya se ha comentado a lo largo de esta memoria, existen, además de los estándares internacionales, soluciones propias ad-hoc de determinadas marcas de vehículos, las cuales han desarrollado un sistema de comunicación exclusivo para sus vehículos. Este es el caso de

Honda, el cual emite un aviso al conductor si hay riesgo de atropello a un peatón, Toyota, que avisa al conductor de excesiva velocidad al llegar a un cruce o Nissan, que también incorpora un sistema de aviso al llegar a intersecciones.

Se tratan pues, de soluciones específicas, que no son consideradas en este trabajo debido a la diferencia entre sus fabricantes (componentes, herramientas, presupuesto, etc.) y el grupo en el que se desarrolla este proyecto.

## **2.2. Desarrollo ingenieril de la solución**

### **2.2.1. Conclusiones prototipo anterior**

Una vez analizadas las conclusiones del prototipo realizado durante la primera parte de este proyecto, se concluyó:

*“Una vez analizadas las pruebas realizadas al dispositivo nos damos cuenta de que dispone de una latencia baja y de una pérdida entre paquetes prácticamente nulas, que son características muy interesantes para la comunicación de corto alcance entre vehículos. El problema viene conforme el número de nodos de la red comienza a incrementarse, lo que produce que comiencen a tener importancia la latencia y las pérdidas. El punto de inflexión a partir del cual la latencia y las pérdidas son insoportables se produce a partir de los 15/20 nodos en la red, lo que teniendo en cuenta el alcance del dispositivo hace que en muchas situaciones reales como atascos o circulación por ciudad se produzca dicho punto de inflexión superando los 20 nodos en la red.”*

Por lo que nos encontramos en que es necesaria una mejora en cuanto al sistema de comunicaciones de corto alcance del prototipo.

### **2.2.2. Requerimientos de la comunicación de corto alcance entre vehículos**

Para ello se realizó un análisis de las características más importantes que debe tener este tipo de comunicación, resaltando estas dos como esenciales:

- Saber en todo momento, la cantidad de vehículos (nodos) tenemos a nuestro alrededor para así poder comunicarnos con ellos si fuera necesario.
- Conseguir una latencia entre mensajes cuanto más baja mejor, poniendo como requerimiento que al menos ésta no fuera superior a los 100ms.

### 2.2.3. Mejora de la comunicación de corto alcance del prototipo

En cuanto a estas características el primer prototipo funcionaba relativamente bien, en cuanto a descubrir nodos en la red, pero su latencia aumentaba rápidamente conforme lo hacía también el número de nodos en la red. Haciendo prácticamente imposible la comunicación cuando estos superaban los 20.

Por lo tanto únicamente teníamos dos alternativas:

- Sustituir la tecnología encargada de esta comunicación (Xbee) en el prototipo por otra de las que encontramos en los trabajos anteriores. Esta opción es la más sencilla de llevar a cabo ya que sería únicamente adaptar el prototipo a esta nueva tecnología y realizar el mismo tipo de pruebas que se realizaron a la antigua tecnología. Pero tiene la desventaja de que se tienen que adquirir los equipos, es decir es un desembolso económico. No se conoce ninguna tecnología que nos proporcione baja latencia y descubrimiento de nodos. Si al final no se encuentra dicha tecnología, podríamos estar ante otro prototipo inservible a la hora de la comunicación de corto alcance.
- Realizar una interfaz de comunicaciones de corto alcance mixta, es decir con más de una tecnología. Esta opción nos permite continuar con el dispositivo Xbee que sabemos que funciona muy bien a la hora de descubrir nodos. Por ello nos tendríamos que centrar en buscar una tecnología que nos permita comunicaciones a una latencia lo más baja posible. Es una opción en la que a parte de buscar una nueva tecnología, tendremos que diseñar el nuevo protocolo de comunicación y adaptarlo al prototipo.

De estas dos opciones se ha elegido la segunda, es decir realizar una interfaz mixta. Por ello decidimos dividir en dos canales distintos (uno por interfaz), la comunicación de corto alcance, denominando estos:

- Canal de control: Es el encargado de descubrir nuevos nodos en la red y dirigir las comunicaciones.
- Canal de datos: Es el encargado de la comunicación como tal, es decir comunicarse a partir de la información recibida por el canal de control.

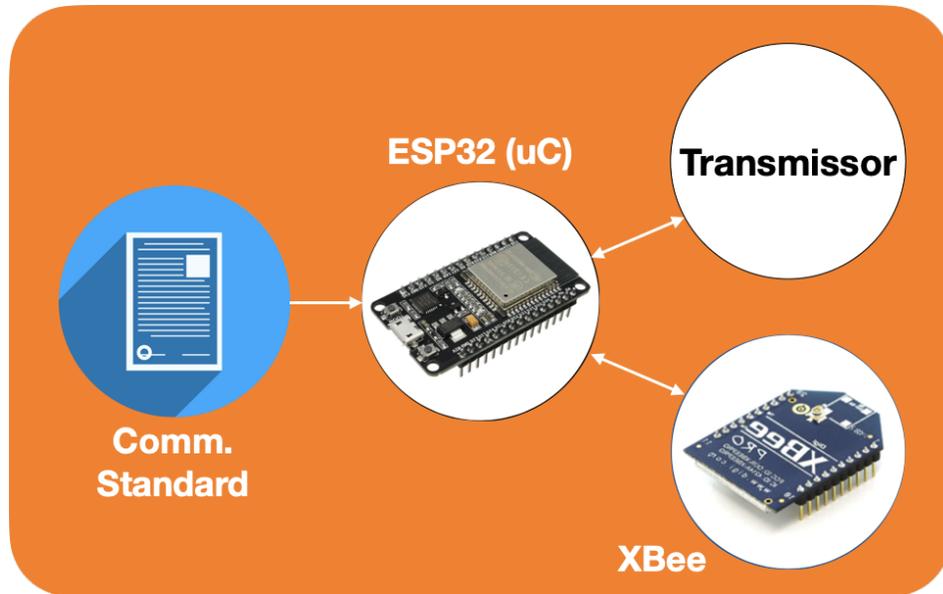


Figura 2.2: Interfaz de corto alcance del nuevo prototipo

Para la elección de la tecnología que llevará a cabo la función de envío y recepción de datos, revisamos las distintas alternativas de los proyectos anteriores entre las cuáles destacaba el 802.11p.

El principal problema con esta tecnología es que los equipos son bastante caros y difíciles de encontrar. Por lo que intentamos buscar alternativas que se asemejaran a ella.

Entre todas las que miramos nos llamó la atención el dispositivo NRF24L01, el cuál trabaja a la misma frecuencia que el 802.11p (2.4 GHz) y además eran equipos relativamente baratos y de los que ya disponíamos de ellos en el laboratorio.

### Dispositivo NRF24L01

En este apartado se quiere profundizar un poco más sobre este nuevo dispositivo, ya que tiene una serie de características muy favorables para las comunicaciones vehiculares, pero tenemos que tener en cuenta que tiene pequeñas peculiaridades a la hora de comunicarse con otros dispositivos.

Estas peculiaridades vienen dadas respecto a la configuración que se debe dar entre dispositivos para que estos se puedan comunicar. Principalmente porque estos dispositivos tienen una jerarquía de comunicación maestro-esclavo, es decir siempre tiene que haber un dispositivo que tenga el rol de maestro (transmisor) que se comunique con el resto de esclavos (receptor).

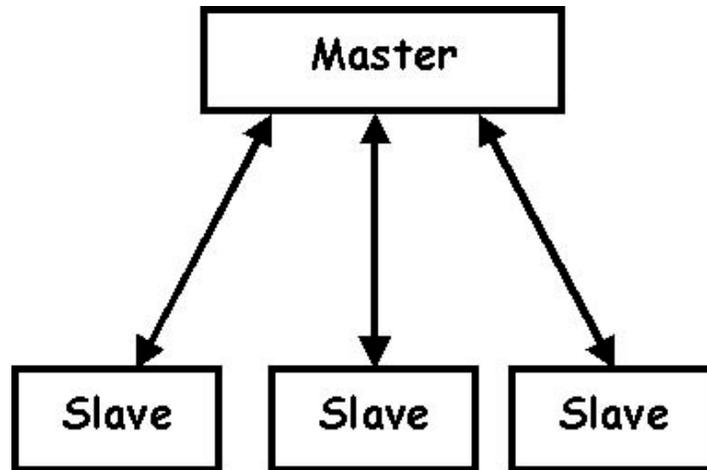


Figura 2.3: Jerarquía de red en el dispositivo NRF24L01

Siendo la comunicación básica del dispositivo, la que se puede ver en el siguiente diagrama, en el que se ve como a cada paquete con datos que envía el Maestro se recibe un mensaje de acuso de recibo (ACK) por parte del receptor. Haciendo que la comunicación sea fiable, y prácticamente anulando la pérdida de paquetes.

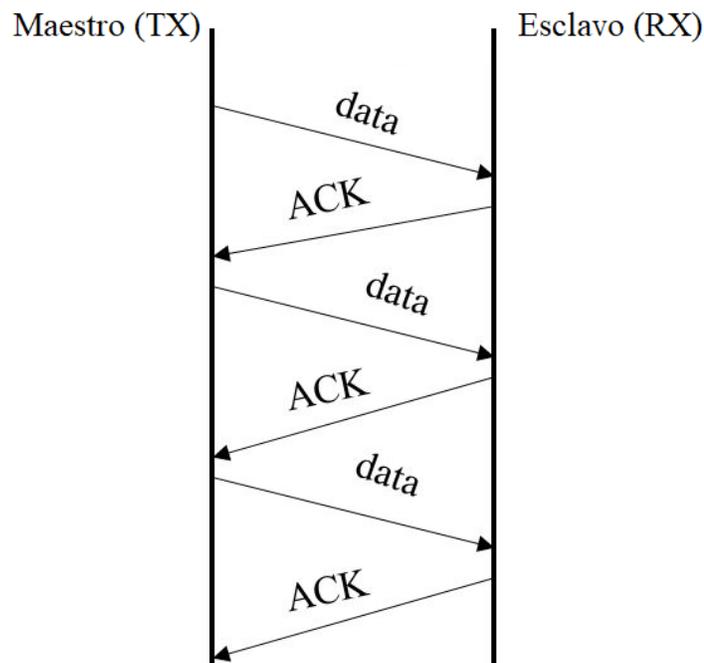


Figura 2.4: Diagrama de comunicación básica NRF24L01

Por tanto para el tipo de comunicación que queremos abarcar con este dispositivo, que son las comunicaciones vehiculares, en el que todos los nodos de la red tienen la misma jerarquía era un pequeño hándicap, ya que en todo momento tendríamos que ir cambiando la

configuración de todos los nodos para que todos pudieran enviar mensajes (o pasar al rol de maestro). Ese proceso de cambio de todos los dispositivos es costoso debido a:

- Tiempo requerido para el cambio de configuración en el nodo.
- Requiere de mucha sincronización y capacidad del canal de control para determinar en todo momento el rol de cada dispositivo.

Se decidió adaptar esta jerarquía al canal de datos, de tal forma que dividimos los nodos en maestros y esclavos:

- **Maestros:** son todos los dispositivos situados en un sitio fijo, o del cuál se conoce en todo momento su posición, como pueden ser un semáforo, una señal, etc. Es decir todos los dispositivos que contemplen la nomenclatura de RCU.
- **Esclavos:** el resto de dispositivos que circulan por la vía libremente, y que se van encontrando con sus distintos elementos fijos (RCU's). Es decir todos los dispositivos que contemplen la nomenclatura de OBU.

#### 2.2.4. Diseño del nuevo protocolo de comunicación de corto alcance

La comunicación de corto alcance se divide en dos canales, el de control y el de datos. Por ello primero se definió el protocolo de comunicación de cada uno de estos canales por separado.

##### Canal de control

Para el diseño de este canal, puesto que se trata del descubrimiento de los vehículos que se encuentran a nuestro alrededor, utilizamos la misma metodología que utilizamos en el primer prototipo con el Xbee. Estos dispositivos están configurados de tal forma que envían cada cierto tiempo (aprox. 200ms) un mensaje de reconocimiento que es recibido por el resto de dispositivos que estén a su alcance.

El diagrama de flujo con los distintos pasos detalladamente explicados se encuentran en el reporte de la primera parte del proyecto. Por ello se va a realizar un pequeño resumen con los datos más relevantes para este segundo reporte.

Para poder llevar a cabo la tarea de poder enviar y recibir mensajes, se pensó en realizar dos tareas diferenciadas que trabajase en paralelo. Como bien es sabido los microcontroladores, trabajan de forma secuencial, recorriendo el programa cada una de las instrucciones en orden. Por ello se implementó un RTOS, que es un sistema operativo en tiempo real, que nos

permite crear diferentes tareas y que éstas las ejecute el microcontrolador simultáneamente a la vez.

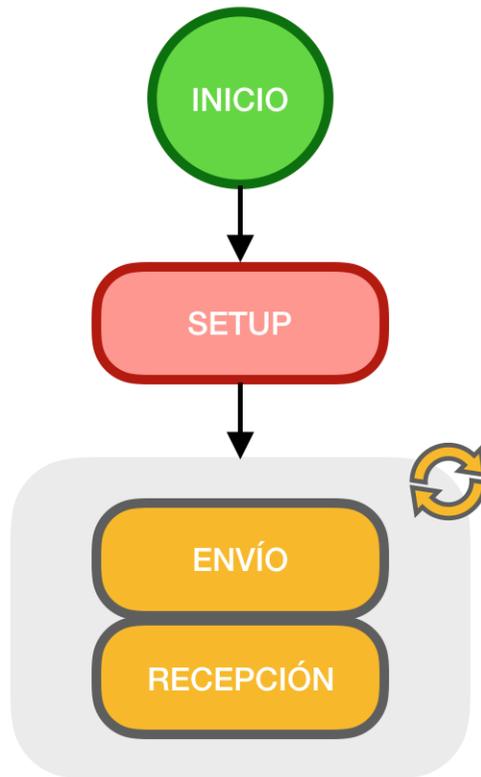


Figura 2.5: Diagrama de flujo del programa

Estas dos tareas, una vez son creadas se ejecutan simultáneamente a la vez, siendo estas:

- **Tarea de Envío:** esta tarea consiste en el envío periódico de mensajes de reconocimiento, para darte a conocer al resto de dispositivos.

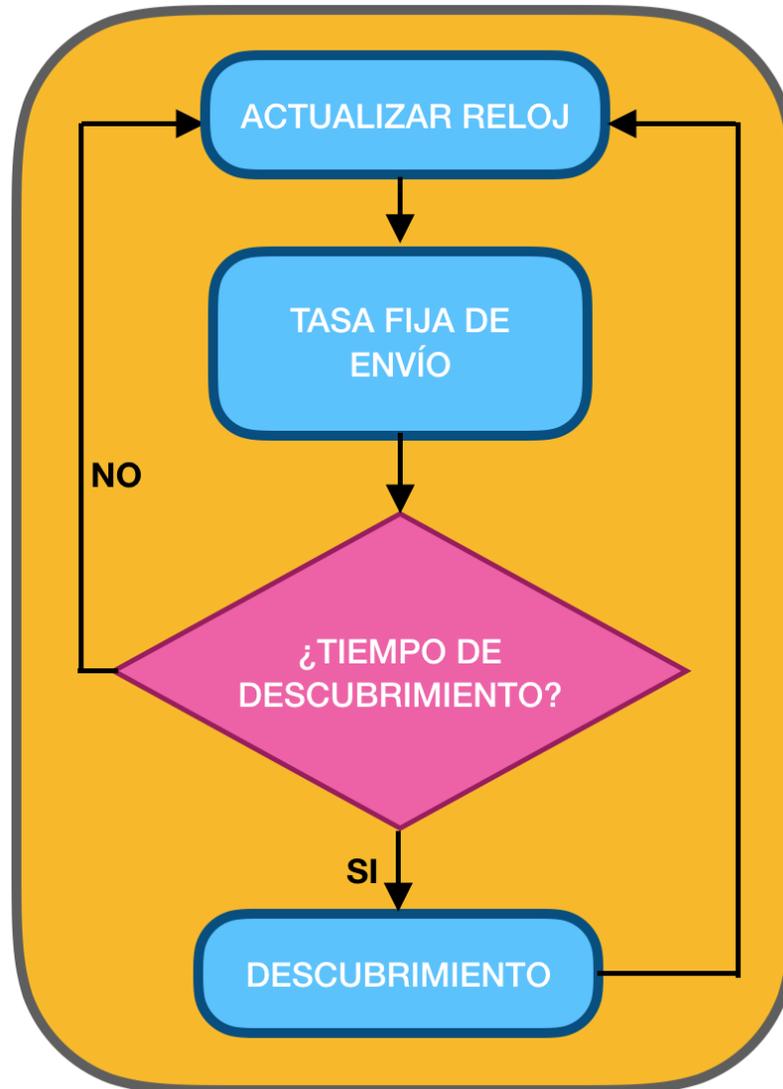


Figura 2.6: Diagrama de flujo del canal de control para el descubrimiento de dispositivos

- **Tarea de Recepción:** En esta tarea vamos recibiendo los mensajes que nos llegan del resto de dispositivos del canal de control, conforme recibimos un nuevo mensaje, extraemos la dirección del mensaje y pasamos al método “*Comunicación NRF24L01*”, que consiste en el envío de mensajes a través del canal de datos.

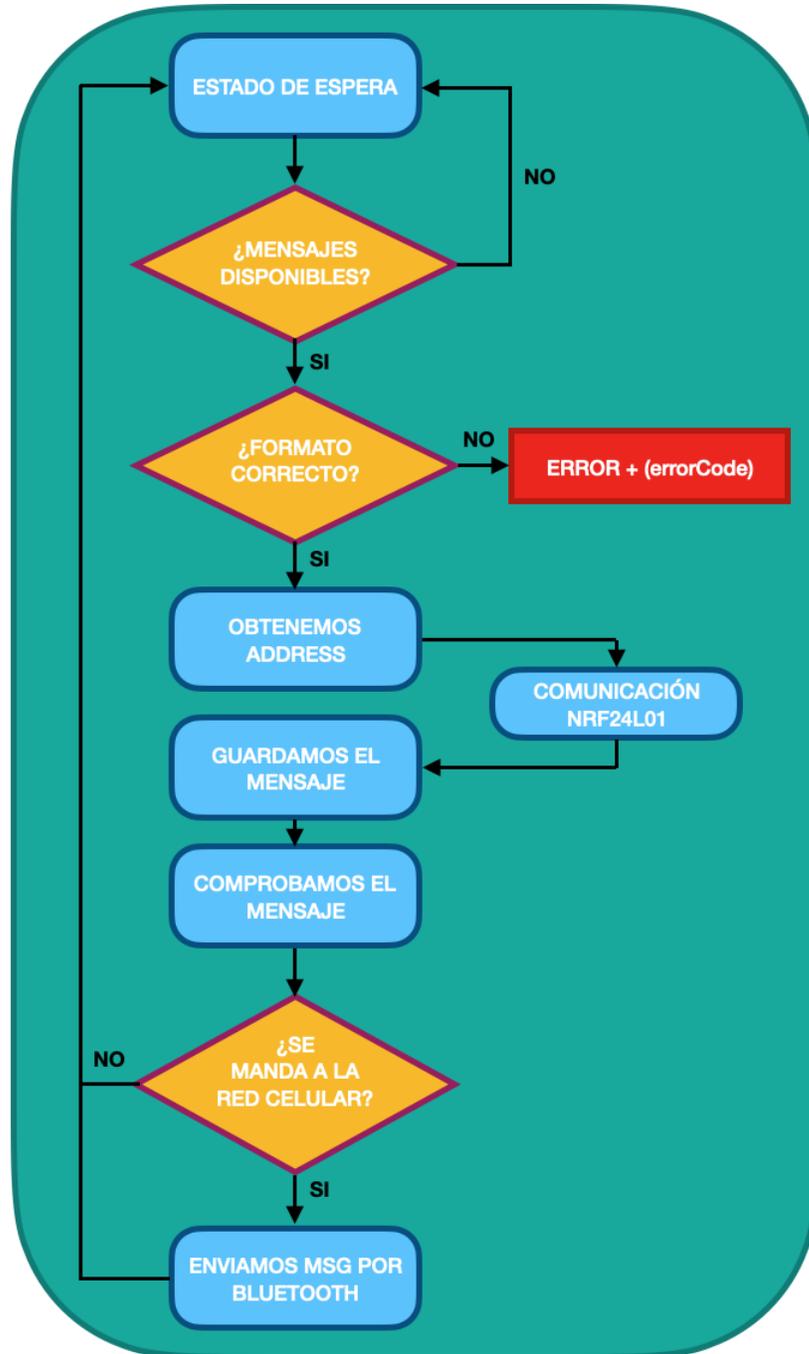


Figura 2.7: Diagrama de flujo para la recepción de mensajes del canal de control

### Canal de datos

Este canal es el que se dedica a la comunicación en sí con el resto de dispositivos. Como ya hemos visto, este canal está determinado por el canal de control, es decir hasta que el canal de control no cede el control al canal de datos, no procedemos a comunicarnos.

El protocolo de comunicación con este canal de datos varía en función de si es un maestro (RCU) o esclavo (OBU):

- **Rol de maestro (RCU):**

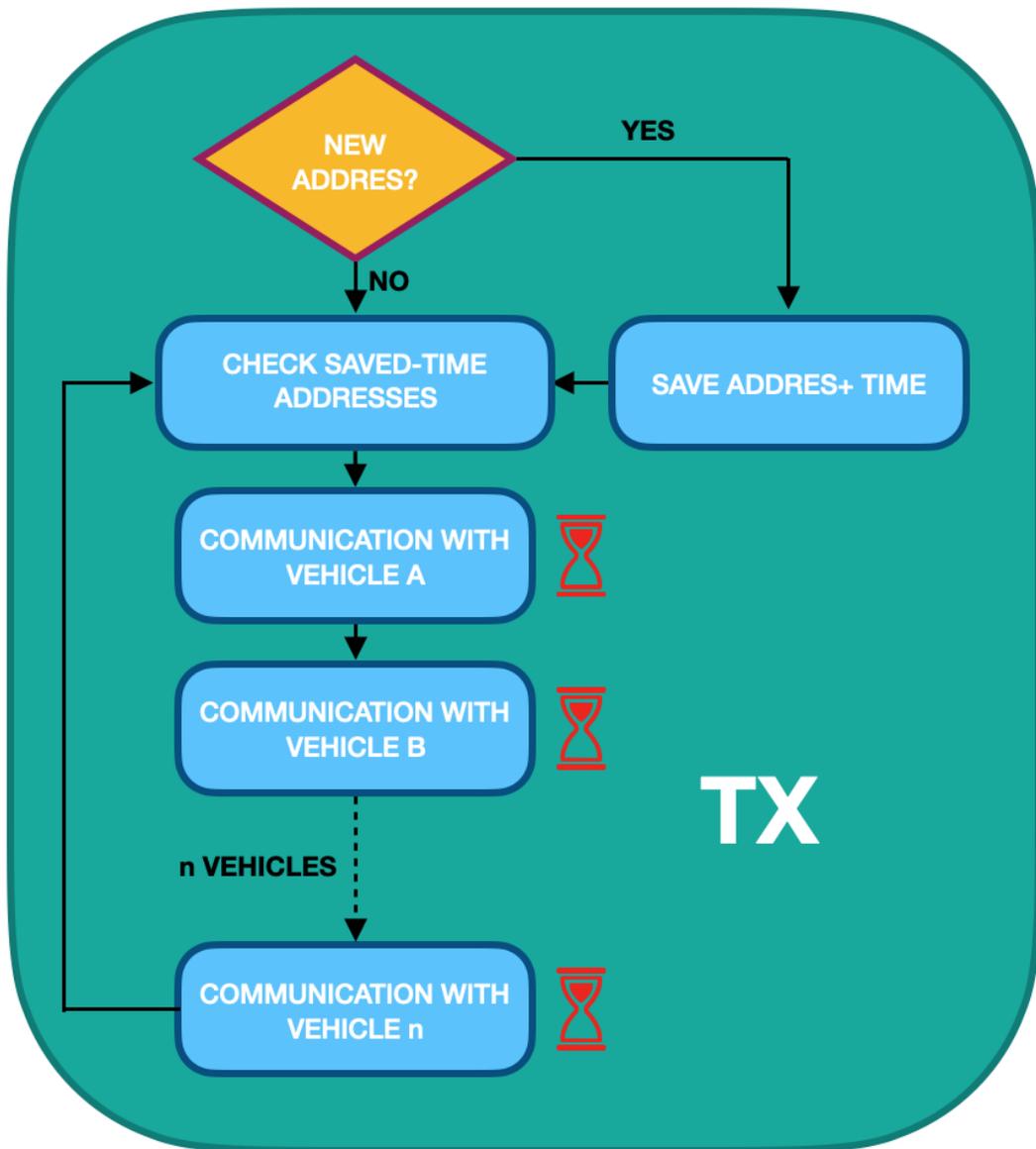


Figura 2.8: Diagrama de flujo para el canal de datos de los RSU

En este modo, una vez la red de control nos cede el control al canal de datos realizamos lo siguiente:

1. Comprobamos la dirección del vehículo que acaba de ser descubierto, es un nuevo vehículo o si por el contrario ya nos habíamos comunicado con el anteriormente. Esto lo realizamos comprobando si el vehículo se encuentra en la lista de vehículos con los que no hemos comunicado anteriormente. Si ésta dirección no se encuentra en la lista, agregamos la dirección junto al tiempo en el que hemos descubierto al vehículo. Si por el contrario esta dirección ya estaba en la lista, únicamente actualizamos el valor del tiempo.
2. Una vez ya tenemos la dirección en nuestra lista, pasamos a comprobar la cantidad de conexiones que siguen activas, es decir no se han quedado obsoletas. A lo que nos referimos con lo de obsoletas es si de una determinada dirección no hemos recibido ningún mensaje suyo en un periodo de 10 segundos desde la última vez. Por ello eliminamos de la lista todas aquellas direcciones con las que tengamos una diferencia mayor o igual a 10 segundos entre su valor de tiempo y el tiempo actual. De esta manera tenemos actualizada la lista actualizada con únicamente las direcciones que siguen activas.
3. Una vez tenemos la lista con las conexiones activas, procedemos a comunicarnos con cada una de ellas individualmente, enviando el mensaje oportuno y esperando la respuesta ACK de cada una de ellas. Pero esta respuesta ACK, no es una respuesta normal y corriente, sino como se explicará más adelante en el “*Rol de esclavo*”, además de certificar la recepción de mensaje, el esclavo añade información. Por tanto tomaremos esa información que ya son datos en tiempo real del vehículo y los enviamos por el puerto serie del microcontrolador para su procesamiento.

■ Rol de esclavo (OBU):

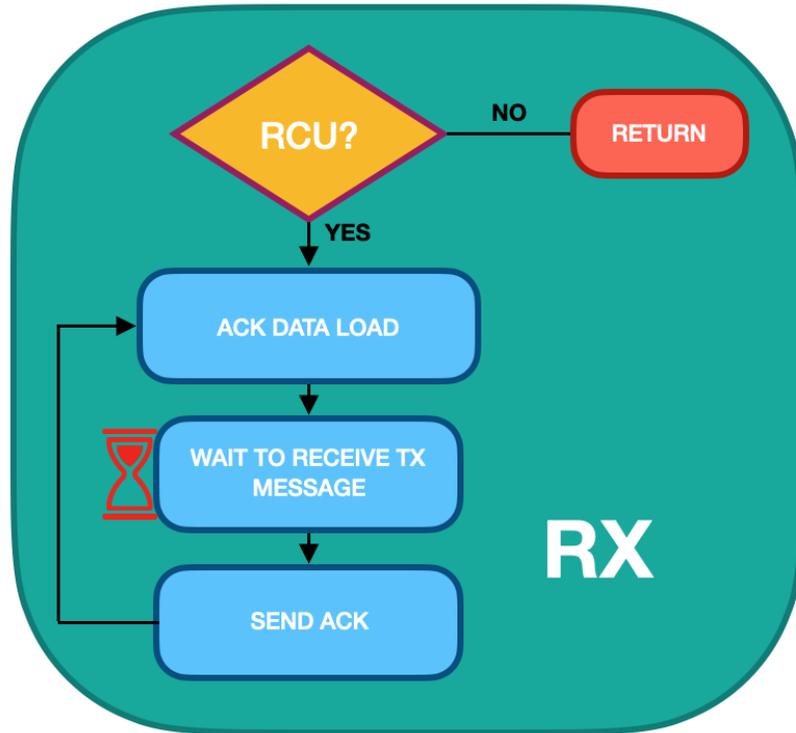


Figura 2.9: Diagrama de flujo para el canal de datos de los OBU

En el caso del esclavo, una vez hemos descubierto una dirección realizamos lo siguiente:

1. Comprobamos si se trata de un RSU o no. Esto lo podemos saber porque los RSU en su mensaje de reconocimiento envían un mensaje especial. Por lo tanto sabemos que si es así, estamos ante un RSU que va a proceder a comunicarse con nosotros. Sino salimos del método y retornamos el control al canal de control.
2. Una vez sabemos que es un RSU, precargamos el dato que queremos enviar en el campo Payload del mensaje ACK, creando un mensaje ACK con payload (datos) como se ve en la siguiente figura:



Figura 2.10: Estructura del paquete ACK con payload

3. Una vez ya tenemos preparado el ACK, esperamos a que el RSU pase a comunicarse con el dispositivo, y una vez recibamos el mensaje del RSU, nosotros enviamos el ACK para confirmar que hemos recibido el mensaje junto con la información que hemos añadido.
4. Una vez que ya hemos enviado el ACK, nos quedamos a la espera de recibir otro mensaje por parte del RSU.

### **2.2.5. Diseño de la PCB del prototipo**

El diseño de la PCB que juntara todos los componentes en un solo lugar y dotara de robustez y portabilidad al prototipo era uno de los objetivos específicos de este semestre, así pues, el diseño de esta, estaba supeditado a tener los componentes definitivos que componían el nuevo sistema de comunicación, y una vez estos estuvieron definidos, se diseñó una placa de circuito impreso con el objetivo de sustituir la actual tabla de pruebas, donde los cables están unidos por contacto. Además, de los componentes previos, se añadió además, una placa de alimentación, que mejoraba las prestaciones de nuestro prototipo en varios aspectos. En primer lugar, cada tecnología podía ser alimentada a su alimentación correspondiente, ya fuera esta 5V o 3.3V; además, la placa de alimentación lleva incorporado diferentes tomas de alimentación (USB y clavija Jack) y un interruptor que permite el encendido y el apagado de la alimentación mecánicamente.

### **Limitaciones derivadas del virus COVID-19**

Antes de empezar con la descripción de la PCB, remarcar que una de las limitantes en las que el COVID-19 ha afectado a este proyecto es en el hecho de no poder pedir la fabricación de esta placa. Esto es debido al alto coste que esto suponía si se fabricaba en España. Debido a la situación del momento, las aduanas y envíos internacionales sufrieron fuertes restricciones motivadas por el pánico frente a la propagación del virus y las limitaciones de movilidad impuestas. Esto ha provocado que la fabricación de PCBs, normalmente manufacturadas en China y enviadas posteriormente a su destino, no haya sido una opción viable. Por tanto todas las imágenes que se adjuntan en el presente documento serán planos teóricos de la PCB en base su diseño en software.

### **Diseño de la PCB y resultados comparativos**

Para el diseño de la placa, se utilizó el Software CAD de diseño electrónico EAGLE, con el que a través de ciertas librerías descargadas de diferentes lugares de internet se descargaron los componentes necesarios para su integración en la PCB.

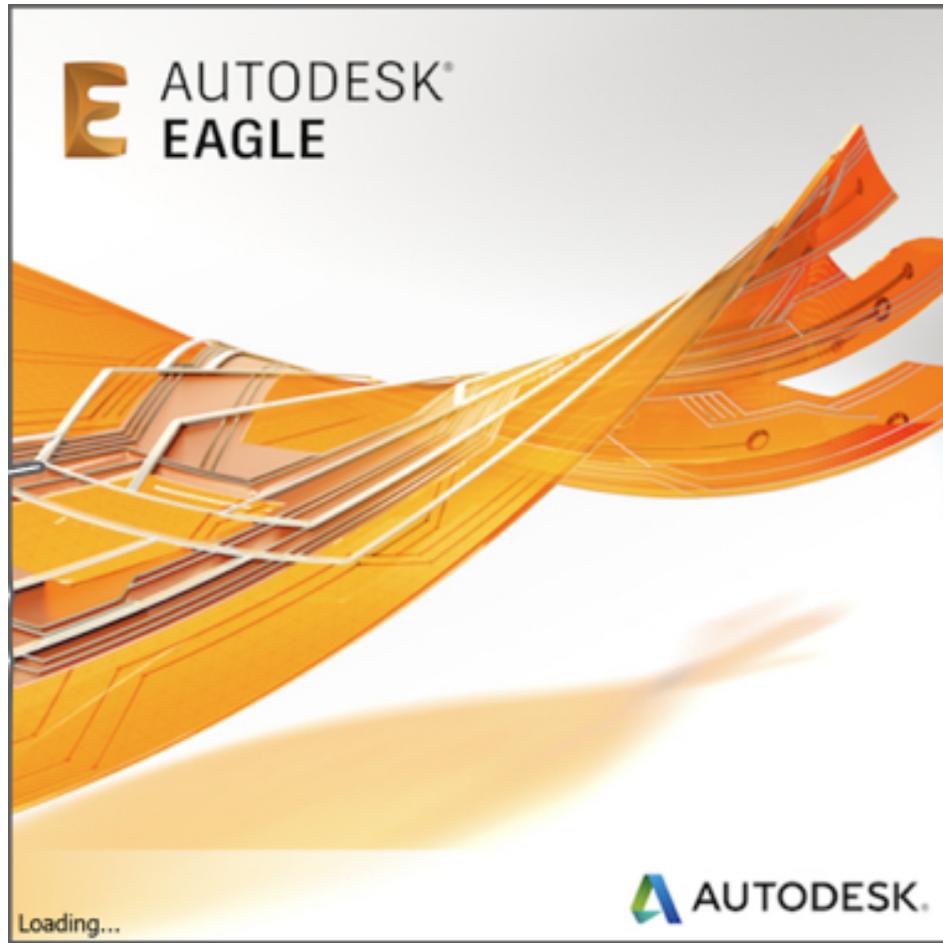


Figura 2.11: Software CAD de diseño electrónico EAGLE

Para una correcta aproximación a este software, explicaremos primero su funcionamiento en líneas generales, de manera que facilite la comprensión de las decisiones tomadas durante el diseño de la PCB. En la siguiente imagen, podemos ver las dos pantallas principales de EAGLE.

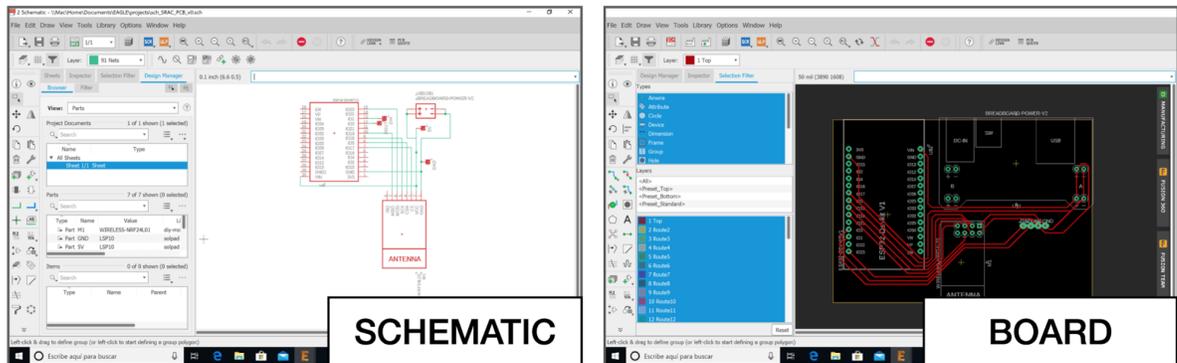


Figura 2.12: Pantallas principales del programa EAGLE

Cada una de estas pantallas, nos brinda una vista de uno de los dos documentos que se generarán para el diseño de una PCB. El fichero Schematic (.sch) y el fichero BoardFile (.brd). Cada uno de los componentes que se añadirán a la PCB tienen una vista asociada a su Schematic y una vista asociada a su BoardFile. Se pudieron encontrar librerías eléctricas (Schematic) y físicas (BoarFile) de todos los componentes utilizados excepto el adaptador del dispositivo XBee, para el cual se implementaron las conexiones eléctricas como pines de salida y las físicas como huellas diseñadas a medida para nuestra placa.

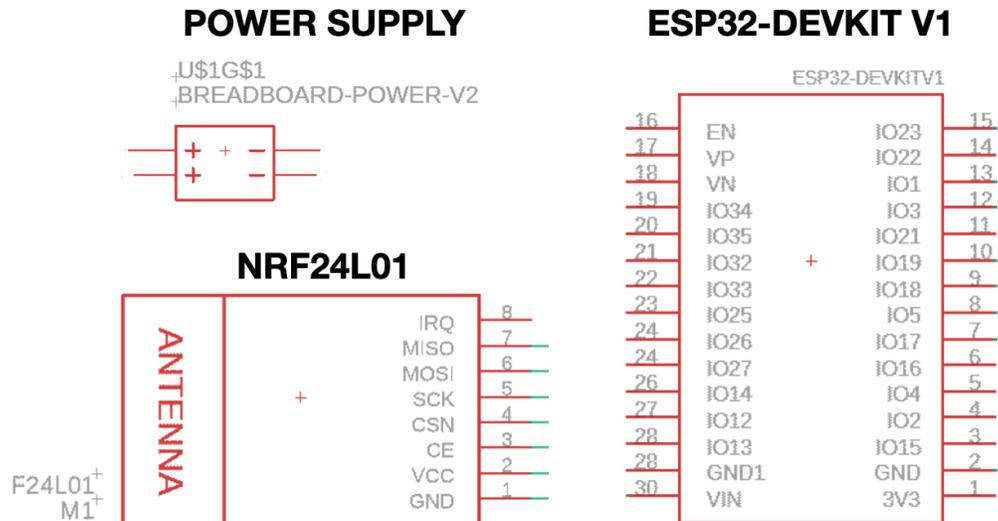


Figura 2.13: Vista *Schematic* de los componentes

En el fichero Schematic, tenemos la vista de los componentes eléctricos del sistema así como sus pines, y este funciona igual que cualquier programa de diseño eléctrico. Cada componente tiene sus pines etiquetados y estos son unidos a través de wires que conectan eléctricamente los pines seleccionados.

En el fichero .brd en cambio, disponemos de las vistas física de cada uno de los componentes. Es decir, los *footprint* (el tipo de huella correspondiente al patillaje y al encapsulado de los componentes) y su disposición física asociada. Aquí, las conexiones son realizadas a través de *paths*, los cuales son una especie de “camino” físico que conecta cada uno de los pines a través de sus *footprints*. Como ya se ha comentado anteriormente, para el adaptador del dispositivo Xbee no se encontraron librerías que contuviesen sus vistas correspondientes y, por tanto, se diseñó un *footprint* a medida que se correspondía con las dimensiones físicas de este dispositivo (disposición de los agujeros, radio de estos, *footprint* de cobre similar, etc.).

Una vez tenemos claros estos diseños y ficheros, cabe comprender que cada una de estas vistas o ficheros no son independientes. Es decir, nosotros diseñamos el sistema eléctrico desde la vista *Schematic* y realizamos las conexiones correspondientes y para que el programa acepte como correcto nuestro diseño, estas conexiones eléctricas deben verse reflejadas en el *BoardFile* mediante conexiones físicas.

En el caso del diseño que nos ocupa, para el prototipo se trasladaron las conexiones actuales montadas sobre una placa de prototipado, a un diseño eléctrico sobre el que se conectan eléctricamente los componentes igual que en el prototipo ya montado. La vista asociada de este Schematic en EAGLE es la que se puede ver en la imagen a continuación.

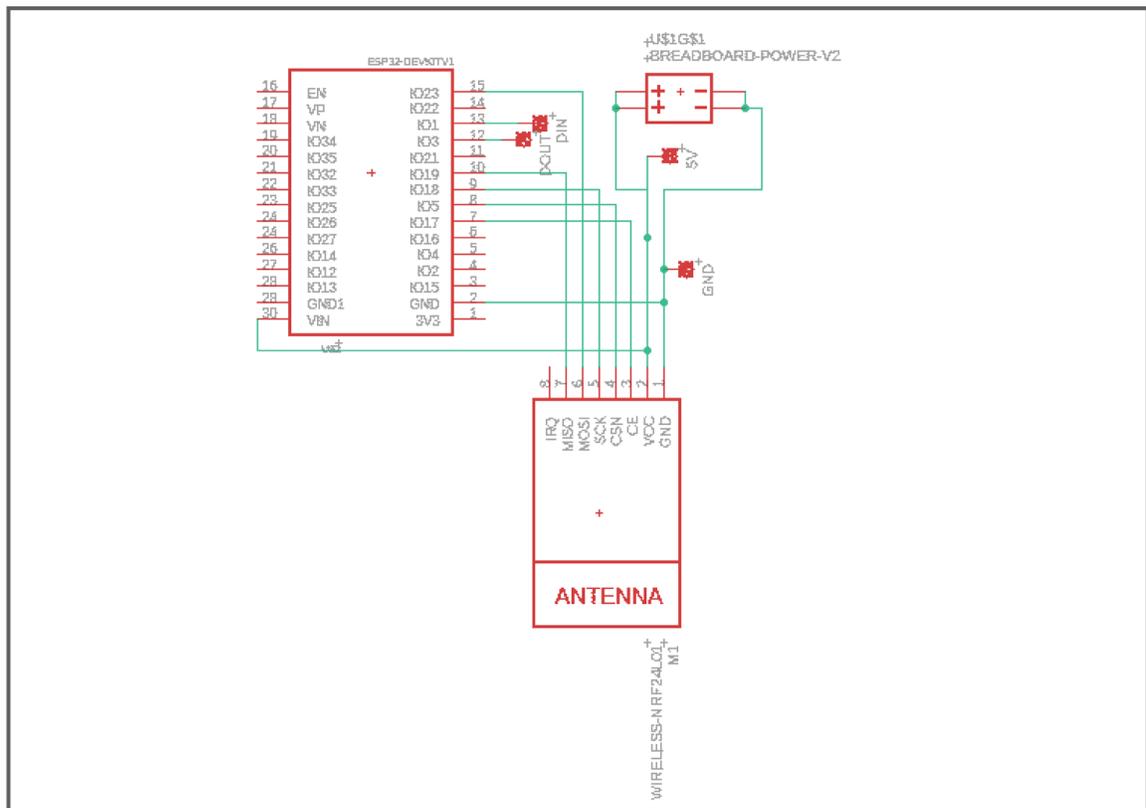


Figura 2.14: Vista *Schematic* del prototipo con NRF24L01 y alimentación

Como vemos, el adaptador del dispositivo Xbee se ha dejado en función de sus cuatro pines (DIN, DOUT, 5V, GND) y el resto de componentes se han interconectado siguiendo la lógica eléctrica del prototipo funcional existente previamente.

En cuanto al *BoardFile* y su vista asociada correspondiente, se diseñó siguiendo varios criterios especiales. En tamaño de los *paths* es superior al requerido (0.016 inches  $\approx$  0.41mm) y solo se usa una capa para todo el diseño, con la intención de abaratar costes e incluso que

se pueda fabricar esta placa de forma casera mediante métodos como el método de planchado sobre cobre. Además, se dispusieron los componentes físicos de manera que estos tuvieran un sentido práctico, es decir, los dispositivos con antenas se colocaron en un lugar de la placa que permitiese la extensión de sus antenas, y los dispositivos que necesitaban tomas de alimentación se colocaron en la parte contrario, de manera que todas las tomas de alimentación quedan en un mismo lugar y no interfieren con las antenas.

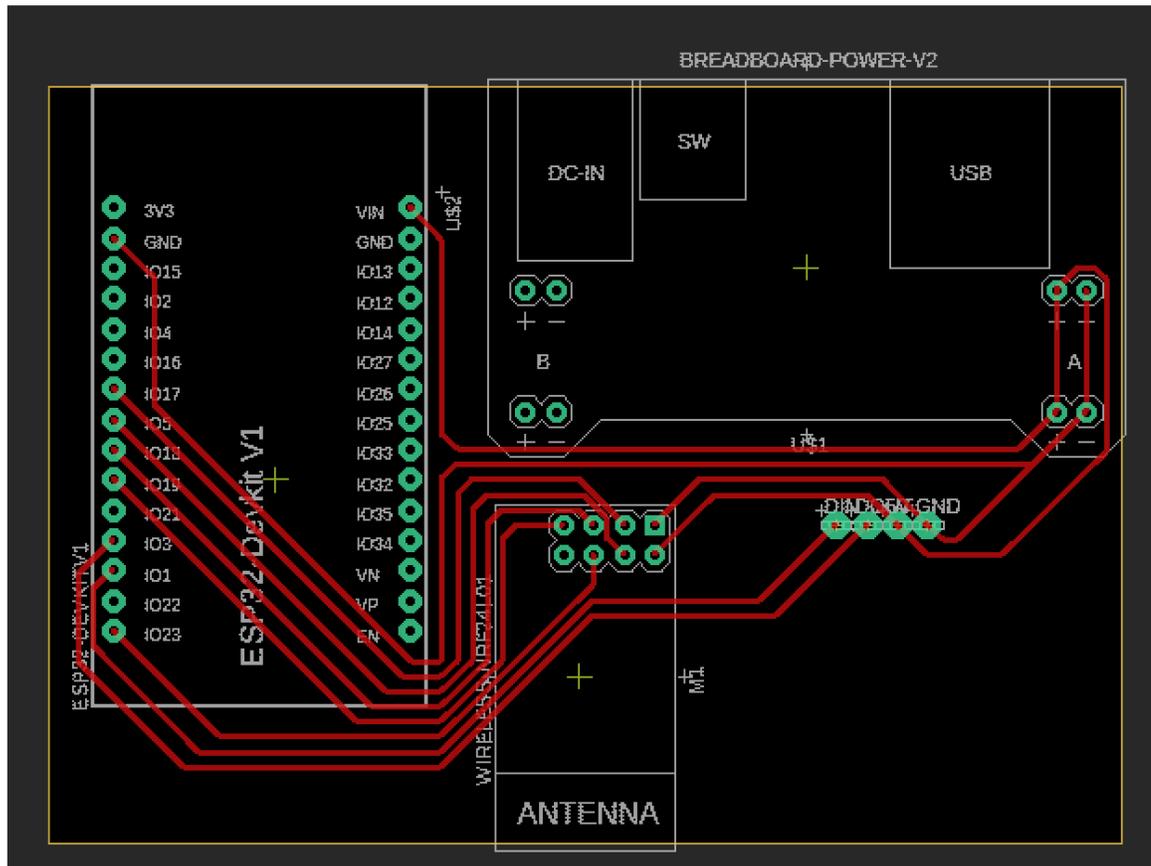


Figura 2.15: Vista *BoardFile* del prototipo con NRF24L01 y alimentación

Los paths son de color rojo, los límites de la PCB se corresponden con la línea amarilla y los *footprints*, lugares donde se colocarán los componentes, son las circunferencias de color verde. Abajo a la derecha se puede apreciar el espacio reservado para el adaptador del Xbee, con su *footprint* personalizado.

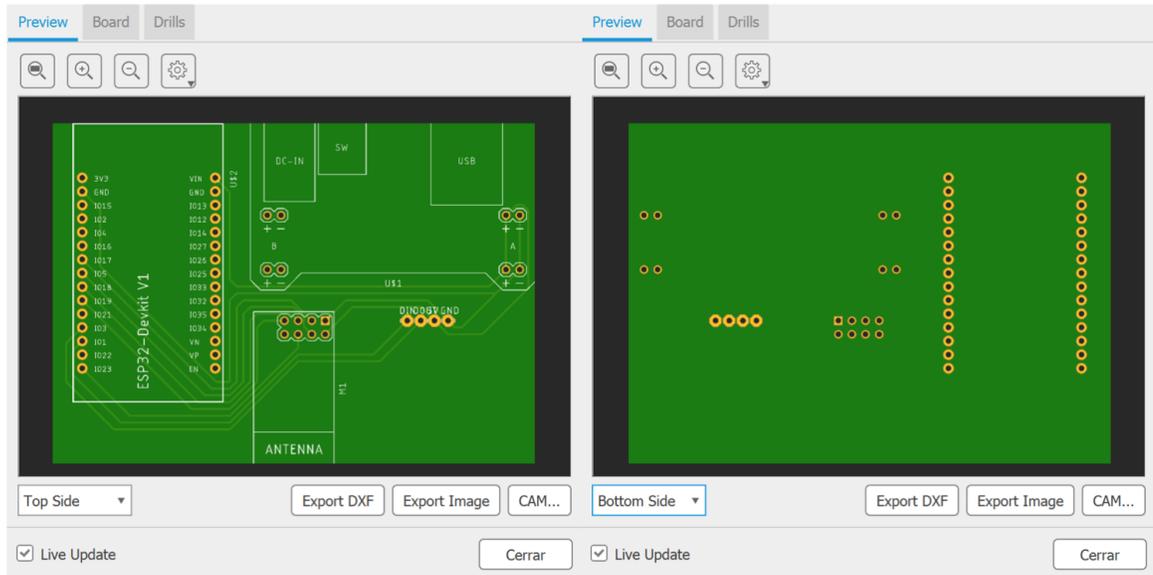


Figura 2.16: Previsualización de la PCB

Aquí podemos ver una previsualización de la PCB ofrecida por el programa, con sus perspectivas frontal y trasera (top & bottom), con la disposición del cobre y de los agujeros, disponible para ser exportada a varios formatos para su fabricación.

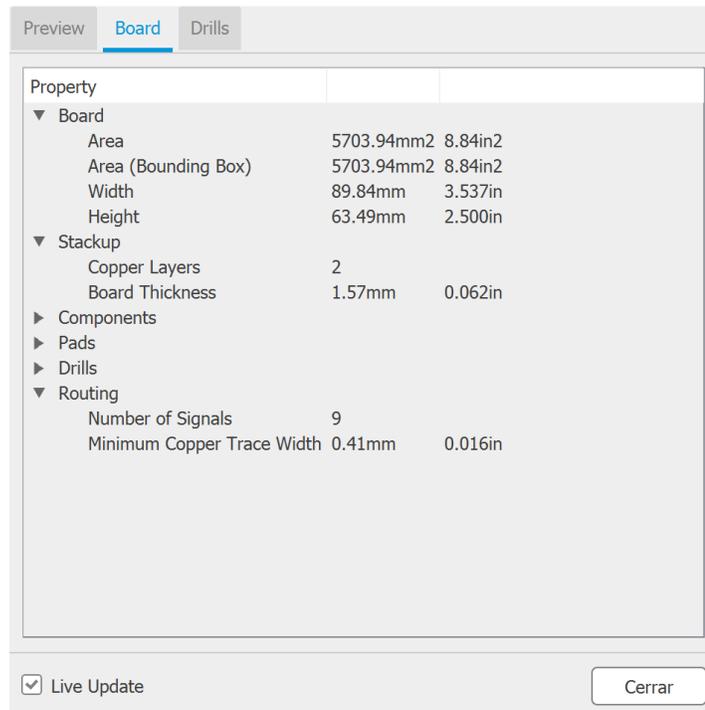


Figura 2.17: Resumen características técnicas de la PCB

Adjuntamos también una tabla resumen de sus características técnicas. Donde podemos ver las dimensiones físicas de la placa de circuito impreso terminada y características técnicas para su fabricación, como número de capas utilizadas (2 debido a que los agujeros están impresos tanto arriba como abajo) y el tamaño y la cantidad de *paths*.

Finalmente, para acabar con la placa de circuito impreso, adjuntamos imágenes que simulan el prototipo montado, con los componentes sobre la PCB para generar una visión general del prototipo y sus dimensiones, así como una mejor visualización de la disposición de los dispositivos y del prototipo en sí mismo y los criterios que se han usado para su diseño. También imágenes de cómo era el prototipo en la primera parte del proyecto para evidenciar cómo el uso de una PCB dota de robustez y portabilidad al sistema a pesar de que se usen más componentes.

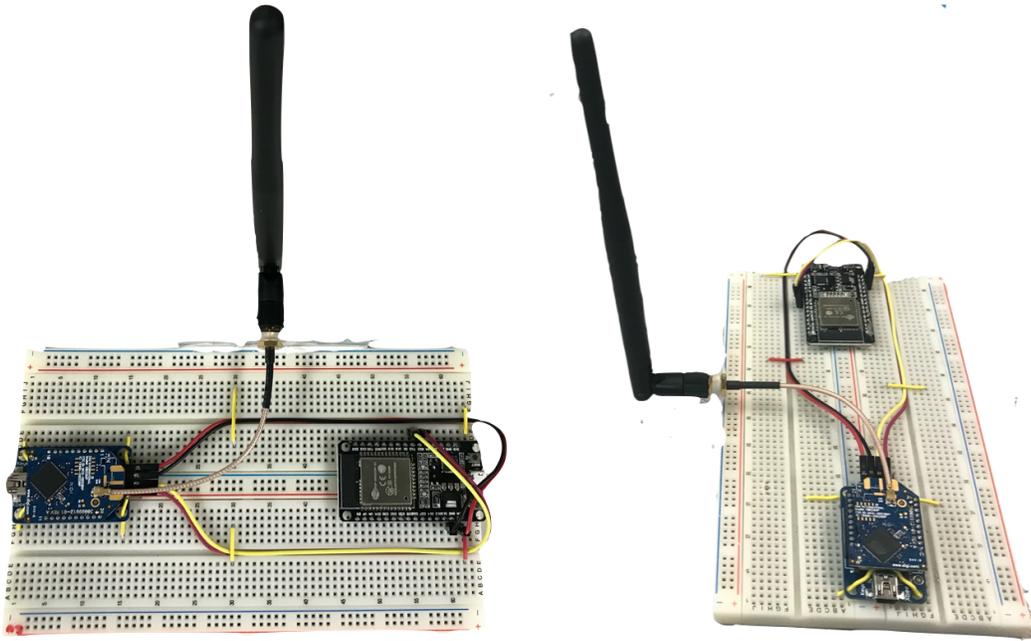


Figura 2.18: Imagen del primer prototipo conectado sobre protoboard

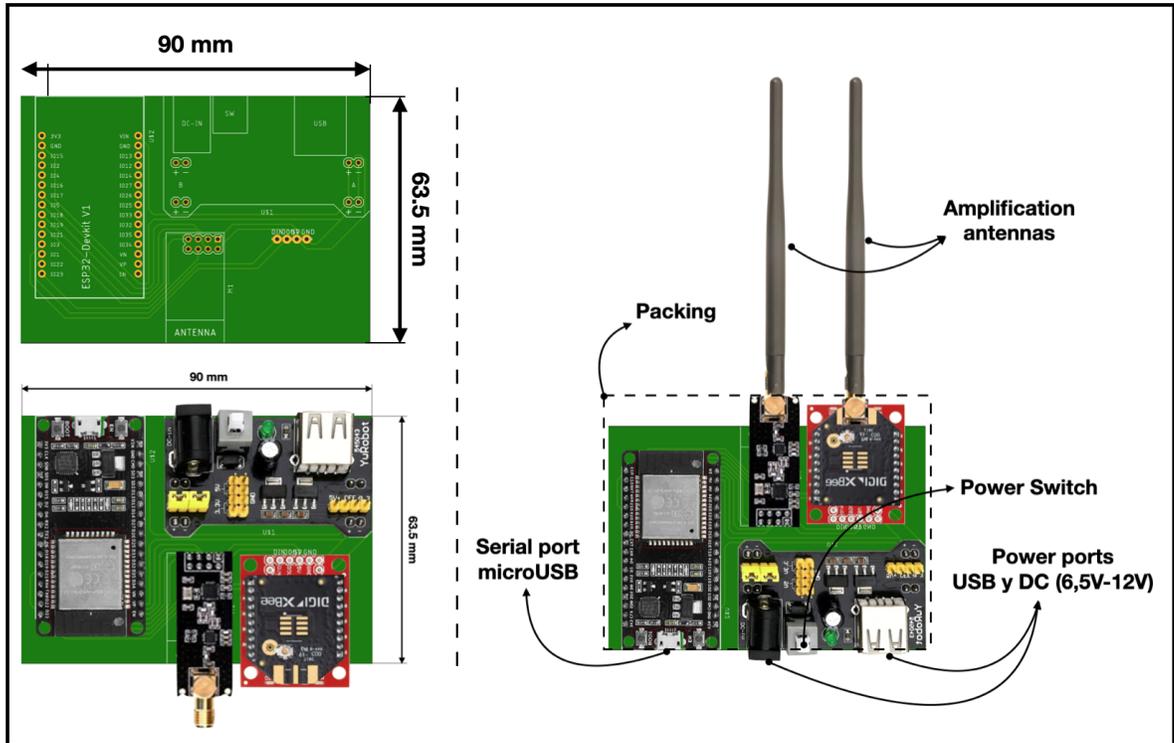


Figura 2.19: Imagen del nuevo prototipo conectado sobre la PCB

### 2.2.6. Solución para escenarios reales

Cómo se explicó en el anterior reporte uno de los aspectos importante es el uso de este dispositivo para la prevención y detección de accidentes para bajar la tasa de los mismos. En ese reporte se realizó un estudio a partir de los datos proporcionados por INEGI (Instituto Nacional de Estadística y Geografía), se determinó que en el 2018 más del 90 % de todos los accidentes en México se produjeron en intersecciones de vía y alcances frontales.

Por tanto para esta segunda parte del proyecto nos centramos también en determinar cómo este prototipo se podría utilizar para disminuir los accidentes en estas dos situaciones:

#### Solución para intersección vial

Para esta situación, tenemos que tener claro que una intersección vial *“hace referencia a aquellos elementos de la infraestructura vial y de transporte donde se cruzan dos o más caminos”*. Por lo que se deduce que son puntos conocidos, en los que suele haber señalización o elementos fijos que los identifican.

Puesto que disponen de puntos fijos, se puede alojar un RSU en cada una de las intersecciones en las que mayor número de accidentes se produzcan. Por lo que conforme los OBU

se acercaran a estos puntos serían detectados por los RSU y sería posible la comunicación entre estos. Esto se explica de una forma mucho más gráfica en con el siguiente diagrama.

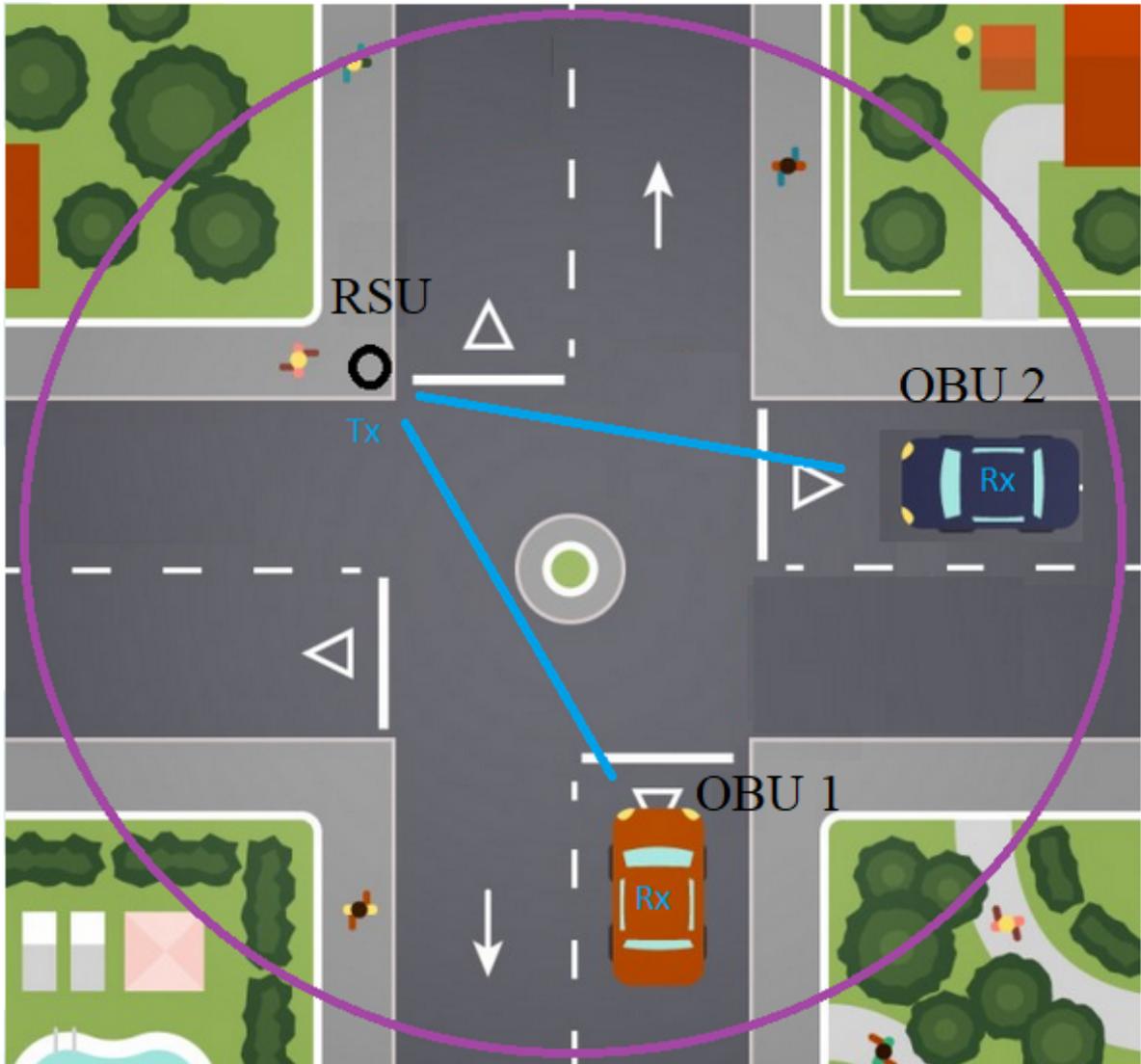


Figura 2.20: Solución para la prevención de accidentes en intersecciones

Como podemos observar en la imagen, el círculo de color morado representa el canal de control del RSU y las líneas azules simulan la comunicación por el canal de datos. Siguiendo el siguiente procedimiento:

1. Primero el OBU 1 se aproxima a la intersección, hasta que entra en el rango de alcance del canal del control del RSU.
2. En cuanto esto ocurre, el RSU, recibe el mensaje de reconocimiento del OBU1 y pasa a comunicarse con él mediante el canal de datos.

3. Durante este proceso un segundo vehículo, el OBU 2, entra dentro del rango de alcance del canal de control del RSU, por lo que el RSU es consciente de que tiene otra dirección con la que comunicarse.
4. En cuanto tiene las dos direcciones almacenadas, pasa a comunicarse con cada una de ellas, hasta que estos dos vehículos abandonan la intersección y salen del rango de alcance del canal de control del RSU. Una vez pasados 10 segundos sin recibir mensajes de estas direcciones, el RSU es consciente de que éstas están obsoletas y procede a su eliminación.

Si analizamos este caso con únicamente dos OBU's y un RSU y teniendo en cuenta, como veremos en las pruebas realizadas, que el tiempo de transmisión entre distintas direcciones por el canal de control es menor a 10ms. Deducimos que tendríamos más de 100 iteraciones entre ambos OBU y el RSU, o 50 iteraciones entre RSU y cada OBU.

Si tenemos en cuenta que como máximo la latencia entre mensajes no debe ser mayor a 200ms, se puede determinar que la cantidad máxima de nodos que pueden ser soportados por canal de datos es de 20 nodos.

Por lo que creemos de forma teórica, aunque con sus limitaciones, es capaz de realizar una comunicación efectiva ante este tipo de situaciones.

### **Solución para alcances frontales.**

A diferencia del anterior, para la caracterización de los accidentes debidos a alcances frontales o trasero-frontal, nos centramos en vías en las cuales existe más de un carril y tienen altas velocidades. Esto es debido a que a mayor velocidad, mayor probabilidad de que el accidente tenga unas consecuencias más graves para los ocupantes del vehículo. Por tanto, vías rápidas de dos o más carriles por las cuales circulan un grupo de vehículos.

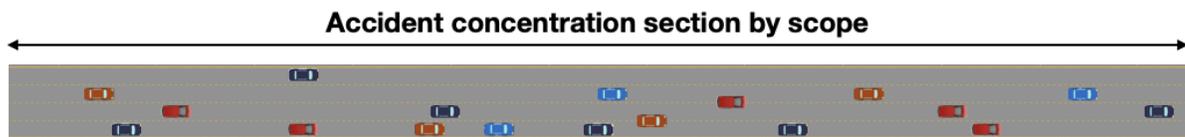


Figura 2.21: Representación de vías rápidas con más de un carril

Una vez comprendido esto, podemos pensar en cuál sería la comunicación objetivo para estos vehículos. Entendemos que para evitar el mayor número de accidentes, estos vehículos deberían de poder tener contacto con los vehículos de delante y atrás suyo, con los cuales tiene mayor posibilidad de colisionar. Conocer su posición, su velocidad o su aceleración o trayectoria podría potencialmente evitar un accidente.

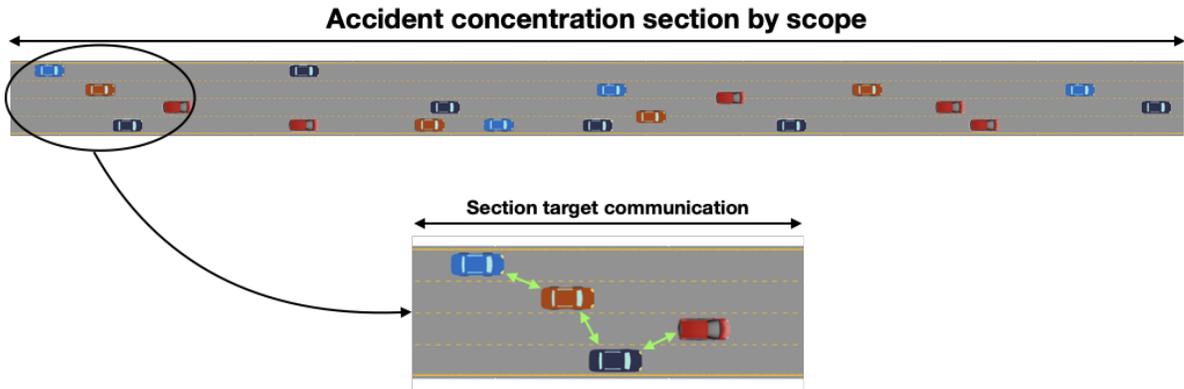


Figura 2.22: Comunicación objetivo entre vehículos próximos

Por tanto, para conseguir esto con el sistema de comunicación diseñado, debemos pensar en cómo conseguir una comunicación con los vehículos adyacentes, y la solución escogida es realizar un sistema similar al realizado en las intersecciones. Establecer comunicación con el canal de control con un RSU, de manera que nos comuniquemos con él y éste sea capaz de reenviar la información relevante al vehículo que corresponda.

Esto plantea dos retos principales: el primero, conseguir la latencia necesaria para que el RSU se pueda comunicar con diferentes vehículos y el segundo, hacer que el RSU sea capaz de saber con quién comunicarse en cada momento. El primer problema de latencia se resuelve utilizando el nuevo canal de datos y el dispositivo NRF24L01. En cambio, el segundo problema se solucionará utilizando el canal de control.

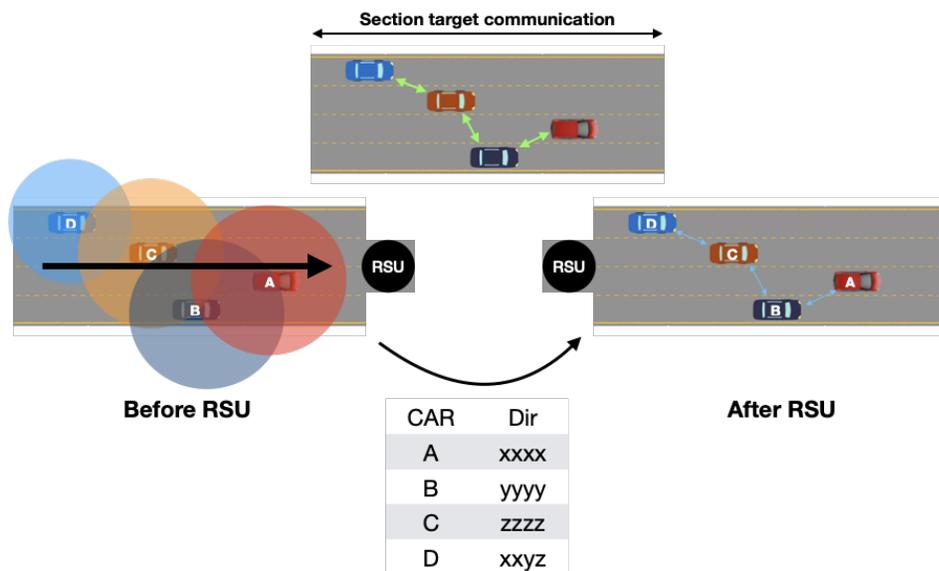


Figura 2.23: Ordenación de los vehículos tras pasar por cada RSU

Para conseguir la comunicación deseada, el RSU debe conocer la posición relativa

de cada vehículo, es decir, el RSU necesita saber qué vehículo está situado delante o detrás de qué vehículo. En la imagen, vemos la comunicación objetivo para un grupo de vehículos. Para alcanzarla, se coloca un RSU y a través del canal de control de estos vehículos, estos establecen la primera comunicación con el RSU (lo descubren). A partir de este momento, los vehículos se comunicarán con él a través del canal de datos. Como vemos, el alcance del canal de control es limitado y el RSU recibirá primero los mensajes del canal de control del vehículo A, en segundo lugar del vehículo B, posteriormente el C y continuando con este sistema, el de todos los vehículos que vayan atravesándolo.

Gracias a esto, el RSU será capaz de construir una tabla o vector, con las direcciones de los vehículos y la posición que estos vehículos entre sí. La posición en este vector será la que determinará a quién son reenviados los datos.

Para tener una representación más visual, imaginemos un escenario como el de la imagen, donde los vehículos A, B, C y D han sido descubiertos en el orden correcto por el RSU y sus direcciones han sido almacenadas en el vector correspondiente. Todos los vehículos están transmitiendo a través del canal de datos información de su posición y su velocidad al RSU. Por tanto el RSU, está recibiendo información de todos los vehículos.

El RSU, reenviará la información del vehículo A y/o del vehículo C al vehículo B, de manera que el vehículo B sabrá la posición y la velocidad de estos dos vehículos, y así podrá prever colisiones. Para el vehículo C, el RSU reenviará la información de los vehículos B y D en el momento en que estos le transmiten datos.

Como vemos, con este sistema somos capaces de hacer llegar información relevante al vehículo correspondiente, pero solo durante un momento determinado en el que los vehículos tiene esa posición relativa. Si estos vehículos cambiaran su posición relativa, por ejemplo, realizando un adelantamiento, estos reenvíos de información se convertirían en algo inútil ya que la información no sería relevante. Para solucionar esto se colocan RSU periódicamente a lo largo de la vía, de manera que un RSU diferente pueda tener un vector diferente (y por tanto actualizado) de la posición de los vehículos.

Siguiendo con el ejemplo anterior, imaginemos que estos vehículos se han desplazado, y nos encontramos en el RSU 2 y la posición de los vehículos ha cambiado y los vehículos C y D han adelantado (o están adelantando al vehículo B). Tendríamos una nueva comunicación objetivo.

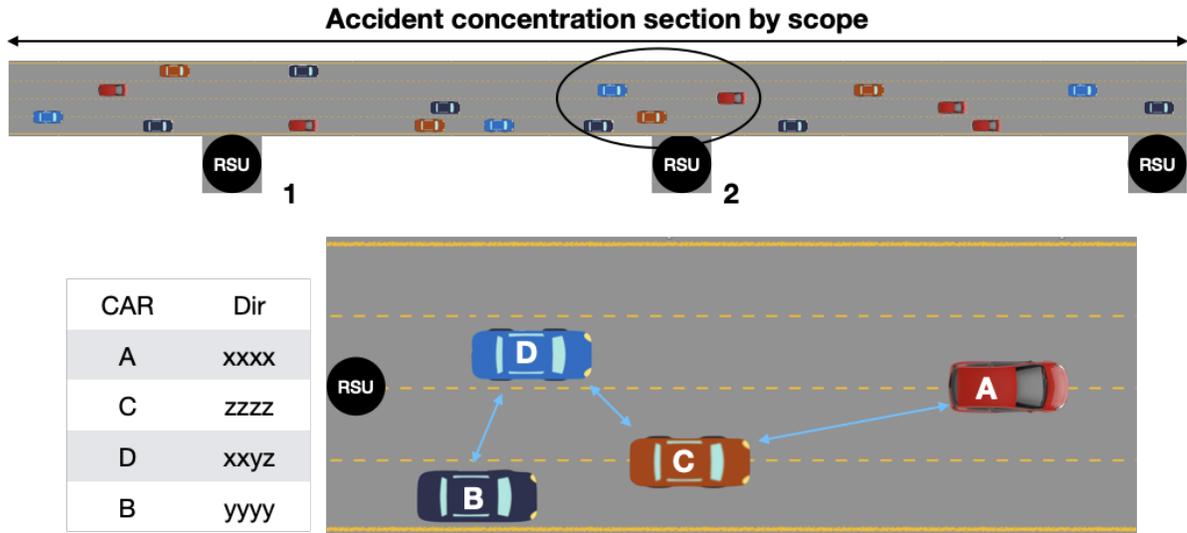


Figura 2.24: Reordenación de los vehículos tras pasar por nuevo RSU

Como vemos, la situación de los coches ha cambiado, pero también el orden en que el nuevo RSU ha descubierto los vehículos, y por tanto, el vector donde el RSU apunta la posición de los vehículos. En este caso, el vehículo C, recibiría la información transmitida por el vehículo A y por el vehículo D. Colocando RSU a lo largo de la vía, se iría actualizando la posición de los vehículos cada vez que se llegara a un RSU.

La situación de estos RSU así como la distancia entre ellos dependerá de la vía. Estos deberían ponerse tramos en los que podemos encontrar concentración de accidentes o incluso a lo largo de toda la vía si se puede. También dependerá de la velocidad de esta y del número de carriles, ya que esto influye directamente en el número de adelantamientos que se producirán a lo largo de la vía.

Este sistema brinda una solución a este tipo de alcances, pero tiene ciertas limitaciones ya que no hay un sistema capaz de resolver estos escenarios a la perfección. Sin realizar las pruebas en escenarios no controlados, podemos prever que el alcance de los NRF24L01 será determinante (300 metros de radio nos daría un alcance máximo de comunicación de unos 600 metros desde que se descubre el RSU). También se debería de incluir una unidad (tampoco muy compleja) que procesara los mensajes recibidos del RSU para poder determinar en qué casos hay riesgos de que se produzca un accidente por alcance.

A pesar de las limitaciones de ambos escenarios, consideramos que el sistema brinda una solución innovadora a los dos escenarios que más accidentes dejan en las carreteras de México.

## 2.3. Plan de pruebas realizado

El plan de pruebas que se ha realizado es el siguiente:

### 2.3.1. Pruebas al dispositivo NRF24L01

Estas pruebas comprenden, todas las pruebas que se realizaron al nuevo dispositivo para determinar su correcto desempeño como interfaz para el canal de datos en la comunicación de corto alcance del prototipo.

#### Comunicación Básica TX - RX

La primera prueba que se realizó fue comprobar la latencia mínima que era capaz de ofrecernos el dispositivo en la configuración de comunicación más básica. Que comprende la comunicación entre un maestro (TX) y esclavo (RX) con ACK.

Las pruebas se han realizado siguiendo la siguiente configuración en los NRF24L01:

- Envío de mensajes por parte del TX con la longitud máxima (32 bytes).
- Acuso de recibo por parte del RX (ACK), sin envío de información.
- Delay de un milisegundo entre el envío de mensajes.
- Esta prueba se ha realizado con cada una de las tasas de transmisión disponibles en el dispositivo (250 kbps, 1 Mbps y 2 Mbps).

Obteniendo los siguientes resultados:

- Prueba para tasa 250 kb/s:



Figura 2.25: Comunicación básica Tx-Rx NRF24L01 con tasa 250 kb/s

**Data rate conseguido: 60.452 kb/s**

- Prueba para tasa 1 Mb/s:

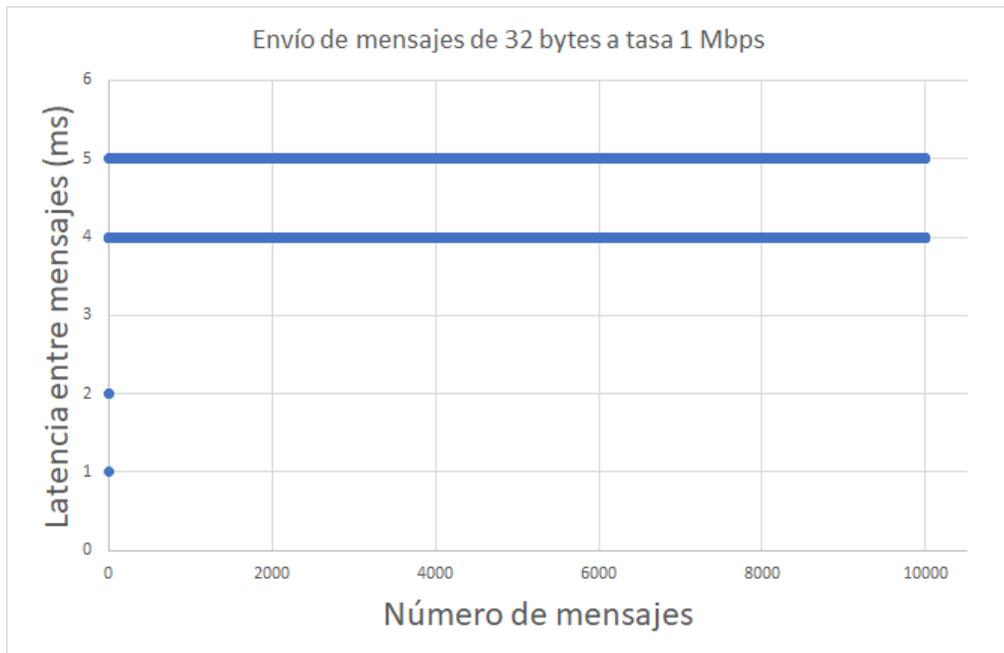


Figura 2.26: Comunicación básica Tx-Rx NRF24L01 con tasa 1 Mb/s

**Data rate conseguido:** 60.454 kb/s

- Prueba para tasa 2 Mb/s:

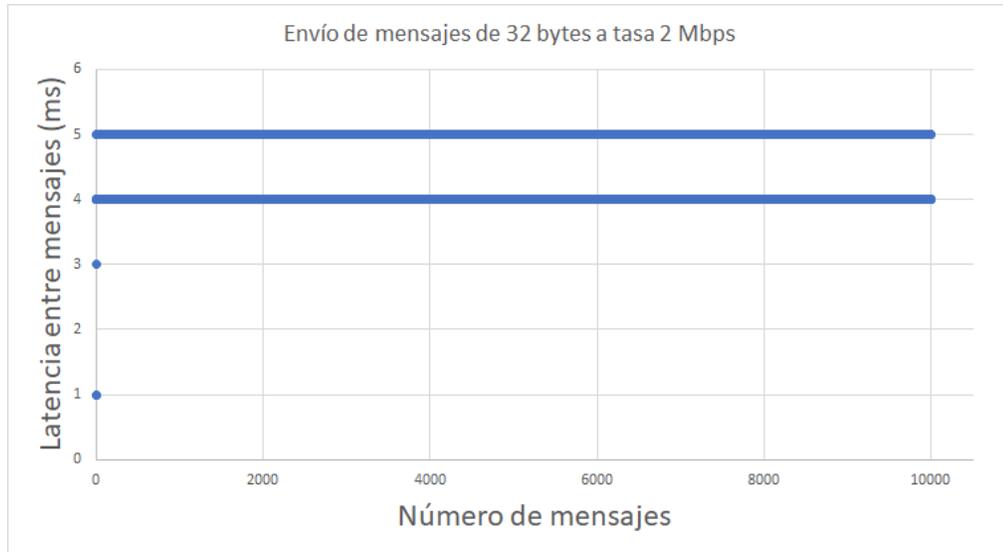


Figura 2.27: Comunicación básica Tx-Rx NRF24L01 con tasa 2 Mb/s

**Data rate conseguido:** 60.464 kb/s

**Resultados:** se puede observar que la latencia entre mensajes es exactamente la misma en las tres tasas de transmisión. Esto es debido a que estas tasas de transmisión son más relativas al tipo de protocolo que conlleva la comunicación, por ejemplo para conseguir una tasa de 2 Mbps que indica el datasheet, debemos de realizar comunicaciones sin ACK entre otros muchos parámetros. Pero este tipo de comunicaciones no es interesante para el proyecto, ya que perdemos la fiabilidad en las comunicaciones.

### Comunicación TX - RX y RX - TX mediante ACK payload

Una vez comprobada la baja latencia entre mensajes con el NRF24L01 procedemos a realizar la prueba con el envío de mensajes entre TX y RX pero esta vez con mensajes ACK con datos, es decir mensajes ACK con payload ( Fig. 2.10).

Las pruebas se han realizado siguiendo la siguiente configuración en los NRF24L01:

- Envío de mensajes por parte del TX (RSU) del tamaño de 1 byte, ya que los mensajes que se envían según el estándar europeo ETSI 102 894-2 son un total de 132, por lo que podemos indicar cuál de ellos deseamos que nos envíe el RX (OBU) enviando únicamente un byte.

- Acuso de recibo por parte del RX (ACK), añadiendo los datos que van siendo solicitados por el TX (RSU).
- Delay de un milisegundo entre el envío de mensajes por parte del TX (OBU).
- Puesto que la tasa de transmisión elegida no supone ningún cambio en la latencia entre mensajes, se ha realizado una única prueba a la tasa de transmisión utilizada por defecto de 250 kbps.

Obteniendo los siguientes resultados:

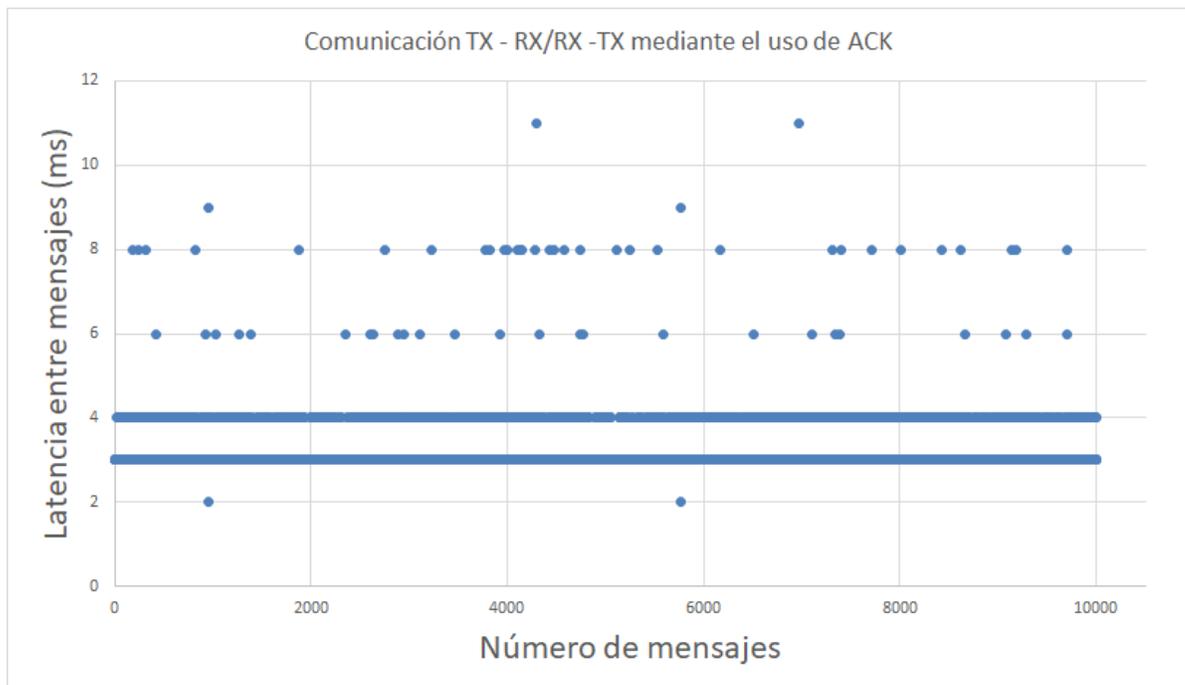


Figura 2.28: Comunicación Tx-Rx y Rx-Tx mediante ACK con payload

Resultados: Podemos observar una ligera dispersión respecto a la latencia entre mensajes, pero no hay que darle demasiada importancia ya que si miramos el tiempo total en que se realiza la prueba (33 segundos) podemos determinar que la velocidad media de los paquetes ha sido de 3,3 ms, es decir muy por debajo del valor máximo mostrado en el gráfico (11 ms). Y que la gran mayoría de los mensajes se encuentra en la franja de los 3 y 4 ms.

### 2.3.2. Pruebas comparativas entre Xbee y NRF24L01

Con esta prueba lo que queremos es realizar una comparativa entre los dos canales de datos de los prototipos. En el caso del primer prototipo se utilizaba el Xbee y para este segundo prototipo hemos utilizado el NRF24L01. Para ello se ha realizado la comparación

centrándonos en la latencia en una comunicación en la que se ven involucrados 4 dispositivos.

Obteniendo los siguientes resultados:

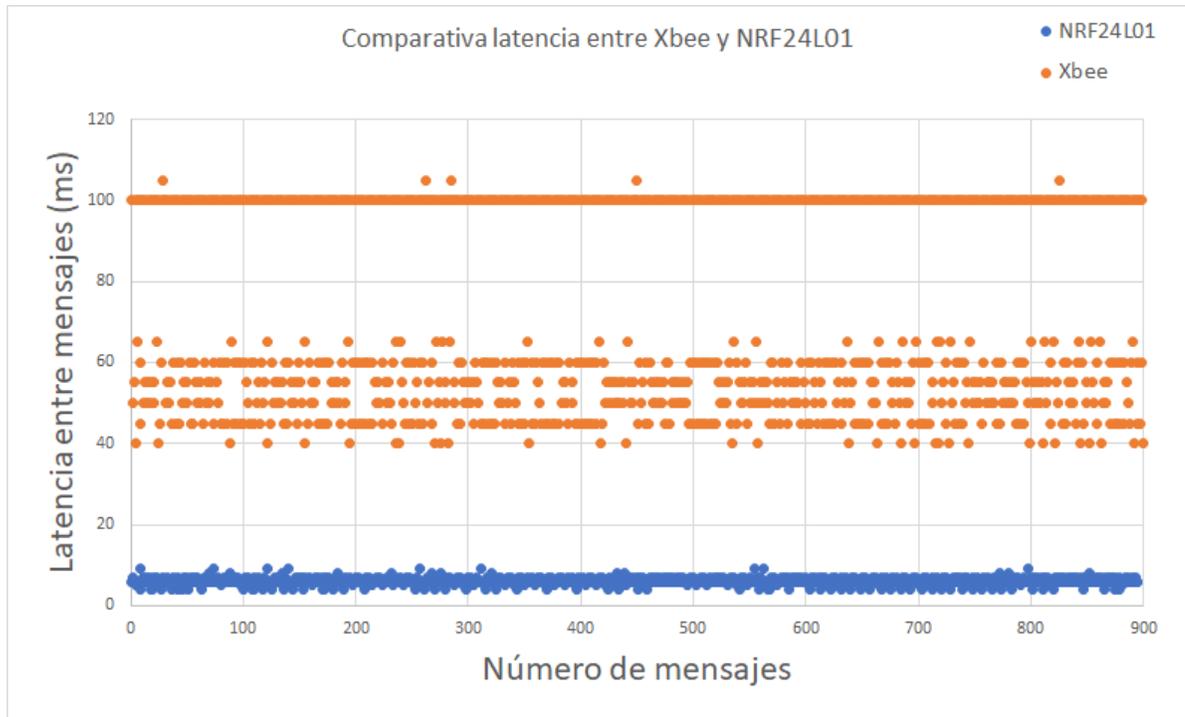


Figura 2.29: Comparación latencia entre Xbee y NRF24L01

Resultados: Se puede comprobar fácilmente que el NRF24L01 (Azul) tiene una latencia mucho menor que el Xbee (Naranja), aproximadamente unas 10 veces menor.

### 2.3.3. Prueba del nuevo prototipo

Una vez ya hemos comprobado la ventaja de usar el NRF24L01, queremos realizar una prueba para comprobar el correcto funcionamiento del prototipo al completo. Es decir con una fusión correcta del canal de datos y del canal de control. Para ello se ha realizado la comunicación entre dos prototipos, ya que debido a que no disponemos de fuentes de alimentación, ese era el máximo de dispositivos que podíamos alimentar correctamente con los recursos de los que se disponía.

La configuración de los prototipos es la siguiente:

- Un prototipo configurado como RSU y el otro como OBU.
- El RSU y el OBU envían un mensaje de reconocimiento por el canal de control cada segundo.

- El RSU realiza el envío de 10 mensajes consecutivos por el canal de control cuando reconoce al OBU.
- El OBU responde con un mensaje ACK con payload a cada uno de los mensajes enviados por el RSU.
- La longitud de los mensajes que se envían son de 16 bytes, ya que creemos que es la longitud de mensaje más adecuada (longitud media mensajes estandarizado: 12 bytes).

Esto se ha llevado a cabo iniciando ambos prototipos (RSU y OBU) durante 50 segundos aproximadamente y monitorizando los mensajes que se recibían por parte del RSU.

Ya que mostrando la prueba completa de 50 segundos en una misma gráfica no tiene mucho sentido ya que es imposible distinguir los mensajes y periodos. Por lo que a partir de la gráfica de la prueba completa, hemos ido haciendo zoom en los periodos que nos han parecido interesantes:

- Prueba completa (50 segundos).



Figura 2.30: Prueba completa con duración de 50 segundos

- Realizamos zoom en la franja de 20 a 30 segundos.

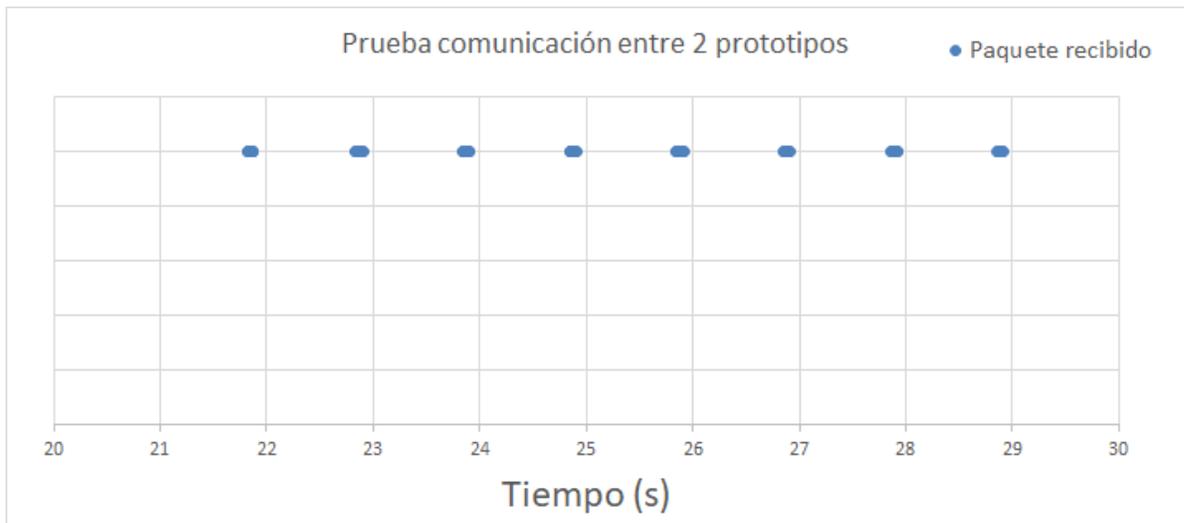


Figura 2.31: Vista de la prueba al prototipo entre 20 y 30 segundos

- Realizamos zoom en la franja de 24,8 a 25,8 segundos.



Figura 2.32: Vista de la prueba al prototipo entre 24,8 y 25,8 segundos

- Realizamos zoom en la franja de 24,8 a 24,94 segundos.

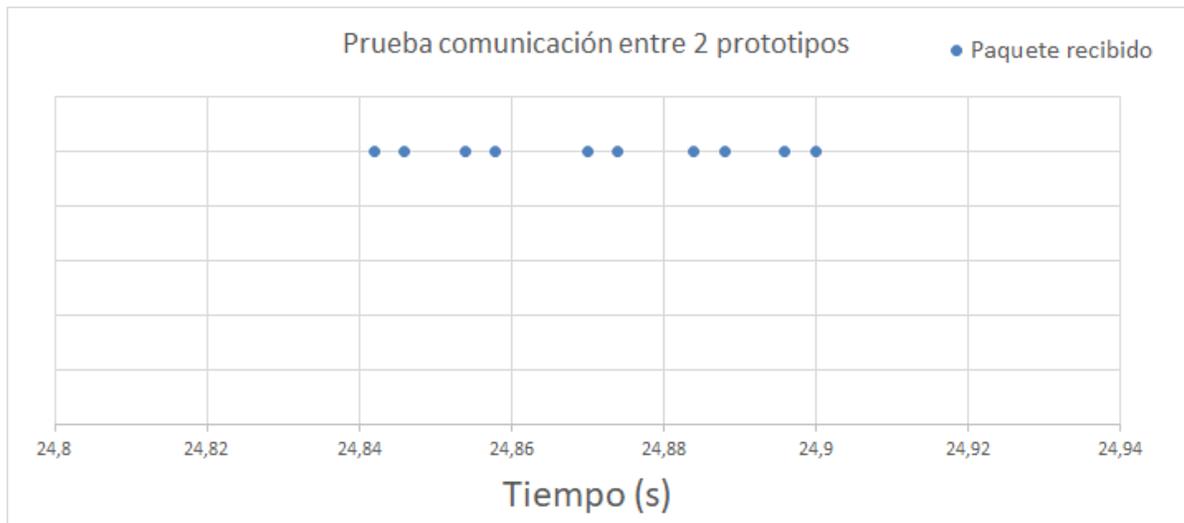


Figura 2.33: Vista de la prueba al prototipo entre 24,8 y 24,94 segundos

Resultados: Con las diferentes gráficas podemos comprobar el correcto funcionamiento de ambos prototipos (RSU y OBU). El RSU se comunica con el OBU cuando este es detectado, por el canal de control, enviando 10 mensajes consecutivos al OBU. Una vez ha enviado esos 10 mensajes, el RSU espera hasta que el OBU vuelva a ser detectado por el canal de control, repitiendo el proceso. Teniendo en cuenta que los 10 mensajes son recibidos en menos de 60 ms, contamos que la latencia media entre mensajes es de 6 ms.

## 2.4. Análisis técnico de los resultados

En la primera parte de las pruebas en las que se probó el nuevo dispositivo NRF24L01 se quería conseguir una latencia entre mensajes baja (menos de 50ms) y comprobar la fiabilidad de la tecnología para el canal de datos del prototipo. Este dispositivo ha demostrado que es capaz de conseguir latencias muy bajas (en torno a 4 y 5 ms), además de que con los mensajes de acuso de recibo (ACK) se realiza una comunicación fiable en la que no perdemos paquetes.

En la siguiente prueba se comprueba la capacidad del sistema de comunicar en ambas direcciones, sin modificar el modo de los dispositivos, para ello se utiliza el envío de mensajes ACK con datos por parte del receptor. Si vamos a los resultados podemos determinar que este tipo de comunicación es totalmente factible ya que, la latencia entre mensajes es baja (aprox 4 ms), con lo que es un protocolo que nos resulta muy interesante para el prototipo ya que nos aseguramos comunicaciones rápidas y fiables entre prototipos.

En la tercera y última de las pruebas se ha comprobado el correcto funcionamiento de los prototipos al completo, simulando a un RSU y un OBU. Si nos vamos a los resultados observamos paso a paso que se realiza correctamente la comunicación y que las latencias también son bastante bajas (6 ms de media). Por lo que con esta prueba podemos comprobar que los dos canales están bien implementados y que funcionan de la manera adecuada. Hubiera sido interesante poder añadir más nodos, pero tenemos el problema de que la alimentación proporcionada por el microcontrolador no es suficiente como para alimentar ambos dispositivos a la vez (Xbee y NRF24L01). Aún así el prototipo está preparado para la comunicación con más de un nodo.

## 2.5. Recursos utilizados

Los materiales necesarios para la realización de un prototipo son los siguientes:

- Microcontrolador ESP-32.
- Dispositivo de RF NRF24L01.
- Dispositivo de RF Xbee-PRO-S3B.
- Antenas omnidireccionales 5 dBi.
- Cables de conexión.
- Protoboard donde realizar las conexiones a falta de la PCB.

Los materiales complementarios al prototipo para poder llevar a cabo el proyecto:

- Herramienta vMicro para el IDE Visual Studio 2019.
- Software CAD EAGLE para el diseño de la PCB.
- Compilador Arduino.
- Cables micro-USB para programación y depuración del microcontrolador.
- Materiales y entorno de trabajo del CIMB (Centro de Investigación de Microsistemas y Biodiseño) del campus TEC Ciudad de México.

Cantidad	Producto	Precio	Total
1	XBP09-DMUIT-156	650\$	650\$
1	NRF24L01	160\$	160\$
1	ESP32-DEVKITV1	159\$	159\$
1	PCB	250\$	250\$
1	Fuente alimentación	30\$	30\$
2	Mano de obra	100\$	200\$
Coste			1289\$
1	Ganancia	100 %	1289\$
Total			2578\$
1	Impuestos	16 %	412.48\$
Total con impuestos			3150.48\$

Cuadro 2.1: Coste aproximado.

## 2.6. Análisis económico del proyecto

## 2.7. Análisis de dilemas éticos

En el ámbito de las comunicaciones entre vehículos, podemos analizar el dilema ético que suponen centrándonos en un punto de vista más general, es decir, centrados en la conducción autónoma o en un punto de vista más específico, y centrarnos en los datos que se utilizan para este tipo de comunicación. A continuación explicaremos ambos, empezando por la conducción autónoma.

### 2.7.1. Conducción autónoma

Como ya sabemos, la conducción autónoma tiene varios dilemas éticos asociados todavía sin resolver. El ejemplo al que más acostumbrados estamos sea, quizás, el ejemplo en el que el vehículo debe tomar una decisión en el que tenga que escoger entre salvar qué vidas: un niño o el conductor del vehículo; una colisión frontal con otro vehículo o descarrilar el vehículo. Como sabemos, no hay una solución correcta a este problema, pero no es el único que nos encontramos en el ámbito de la conducción autónoma.

Esta revolucionaria tecnología, se anuncia como la mejor manera de reducir los accidentes en la carretera, pero este aumento de seguridad, también va ligado a una aumento de la confianza. El vehículo tiene muchos menos accidentes, pero viaja más rápido y debido a esto, los accidentes que ocurren suelen ser de mayor calibre y, por tanto, más peligrosos para los pasajeros. Especialmente en los años en que esta tecnología sea implantada, los coches autónomos deberán convivir con los vehículos tradicionales, y en estos casos un nuevo dilema

de quién es el culpable en un accidente, se abre camino.

Por último, analizaremos también esta etapa de implantación, donde los coches autónomos destruirán empleos como conductores, chóferes o taxistas y además serán un indicador social. La aparición de este tipo de vehículos creará una brecha social relacionada con el acceso a la tecnología. Al principio, solo las grandes fortunas tendrán acceso y solo en países de alto desarrollo tecnológico que hayan podido invertir en el despliegue de toda la infraestructura necesaria para este tipo de vehículos. Así pues, tras estar durante años disponible solo para una élite rica, los vehículos autónomos se espera que produzcan una gran mejora en la calidad de la vida a largo plazo de las clases sociales menos altas (transporte urbano más barato, nada discriminatorio, disponible 24 horas al día, etc.), pero estas tendrán que sufrir en su primeras etapas de implantación la brecha social que surgirá con esta nueva tecnología.

### **2.7.2. Comunicación entre vehículos y entorno**

Como ya se analizó en la primera parte de este trabajo, el mayor dilema asociado a la comunicación entre vehículos reside en el tratamiento de los datos.

En este tipo de comunicaciones, muchas veces los protocolos deben tratar datos como el tipo de vehículo o incluso el color de este, para optimizar el reconocimiento de las cámaras y sensores e incluso ciertos mensajes del protocolo detallan número de ocupantes del vehículo y sus posiciones. Esto, se realiza por si se provoca un accidente, poder advertir a los profesionales sanitarios, o por si en un futuro, los coches tienen que tomar una decisión de recibir un impacto frontal o lateral, etc. y por tanto, son datos importantes que el vehículo almacena y comunica con su entorno. También datos como la ubicación del vehículo y la dirección que este lleva, son claves para una comunicación entre vehículos exitosa y que pueda prevenir accidentes.

Pero todos estos datos deben ser almacenados y analizados para la prevención de accidentes y a la vez afectan directamente a la privacidad de los usuarios que utilizan el sistema. Un mal tratamiento de estos datos podría traducirse en la deducción de información muy sensible para el usuario, como el vehículo que lleva, su dirección, el colegio donde estudian sus hijos, su lugar de trabajo o incluso sus aficiones.

### **2.7.3. Conclusiones dilema ético**

Todos estos dilemas se espera que ocurrirán en el tiempo en que esta tecnología empieza a implantarse, y que poco a poco desaparecerán o menguarán hasta el punto en que no sean tan relevantes como para marcar una diferencia significativa. Por tanto, pese a estos

dilemas y los problemas futuros, el coche autónomo brindará soluciones que mejorarán con creces la vida de las personas.

# Capítulo 3

## Conclusiones

### 3.1. Conclusiones técnicas del proyecto

El objetivo general de este proyecto era mejorar la interfaz de comunicación de corto alcance del primer prototipo. Por lo que creemos que con este segundo prototipo lo hemos podido llevar cabo, consiguiendo una latencia entre mensajes mucho menor a las conseguidas con el primer prototipo, que es una de las características principales de este tipo de comunicaciones. Esta no es la única de las características que ha mejorado con este segundo prototipo ya que si nos vamos a comparar los datos del datasheet de ambos tenemos:

Short Range Comm	Prototype 1	Prototype 2
Maximum Data rate	250 kb/s	2 Mb/s
Message latency	100 ms	9ms
ACK	No	Yes
Max. nodes per second	PCB	10
Proto board	No	Yes
Max range with visibility	1 km	200 m

Cuadro 3.1: Tabla comparación características entre prototipos

Con el nuevo prototipo conseguimos mejorar las características técnicas del primer prototipo en diferentes ámbitos (latencia, número de nodos, inclusión de sistema de ACKs, etc.). La inclusión del dispositivo NRF24L01 aporta grandes prestaciones al prototipo básico anterior y la protoboard lo dota de mayor robustez y portabilidad.

El único parámetro que nos es mejorado por el segundo prototipo es el alcance de comunicación, pero esta característica creemos que no es relevante ya que cuanto mayor alcance tenemos, mayor es el número de nodos en la red. Conforme aumenta el número de nodos en

la red, también aumenta la latencia, haciendo que a partir de cierto número de ellos la comunicación sea inviable. Esto es lo que nos ocurría con el primer prototipo, por lo que creemos que un alcance de unos 100-200 metros es más que suficiente, ya que hay que recordar que este tipo de enlaces son de corto alcance.

Por tanto, concluimos que en cuanto al rendimiento del dispositivo, este ha aumentado mucho respecto al prototipo anterior y las mejoras permiten además, que se siga trabajando sobre ellas en un futuro.

## **3.2. Conclusiones de los miembros del proyecto**

### **3.2.1. Conclusiones - Eloi Pascual**

Aún todos estos problemas asociados sin resolver, la llegada el coche autónomo se espera que cause grandes impactos positivos en nuestra sociedad. Rutas optimizadas que ayuden a reducir combustible, potenciación del vehículo eléctrico, vehículos públicos baratos y no discriminatorios o una reducción de accidentes son algunas de las ventajas directas que podemos prever de la conducción autónoma.

Escenarios más optimistas esbozan futuros donde los coches privados quedarán relegados a usos muy específicos y los vehículos serán sobretodo sociales y estarán en movimiento las 24 horas del día 7 días a la semana, provocando la desaparición de los espacios de aparcamiento que podrían ser reconvertidos en áreas verdes o aprovechadas para diferentes usos.

Pese a todos los problemas asociados y dilemas planteados, que seguro serán muchos sobretodo en la implantación, se espera que la llegada de los vehículos autónomos acabe transformando nuestro mundo en un lugar más eficiente, seguro, ecológico, sostenible y justo.

En cuanto a nuestro desarrollo personal en el proyecto, pese a las limitaciones asociadas a las circunstancias excepcionales que nos han tocado afrontar, creemos que hemos estado a la altura de la situación y a pesar del confinamiento, el flujo de trabajo ha sido el esperado. Se han cambiado los objetivos y se ha modificado el proyecto respecto a ellos y finalmente estamos muy satisfechos por haber logrado los resultados obtenidos a pesar de las limitaciones técnicas que implican estar separados físicamente de los miembros del equipo y de los ambientes de trabajo.

Este proyecto intenta aportar un grano de arena en cuanto a la realización del vehículo autónomo se refiere, y esperamos, que este proyecto pueda ser retomado y mejorado y que se sigan aportando granos de arena hasta que lo consigamos.

### 3.2.2. Conclusiones - Juan Gascón

Durante este proyecto se han visto y documentado muchos casos sobre cómo la comunicación entre vehículos puede tener un impacto relevante en salvar vidas. Esto es muy importante, ya que en muchos países, la muerte al volante supone la causa más importante de mortalidad por causas no naturales. Avanzar en el ámbito del V2X es igual a avanzar en la seguridad del vehículo autónomo. Avanzar en un camino que puede suponer a largo plazo, salvar vidas humanas.

Gracias a la conducción autónoma se podrá avanzar hacia un mundo más sostenible y más justo, donde decrezcan las desigualdades sociales y avancemos hacia una sociedad más moderna.

Es posible que nuestro trabajo consiga aportar algo novedoso en este tema, ya que ha sido realizado a través de herramientas de bajo coste y accesibles, para que el avance del proyecto no se vea limitado por motivos económicos o de accesibilidad. Mejoramos el rendimiento del prototipo anterior aumentando de manera marginal el costo del prototipo y esperaríamos que el proyecto pudiera seguir en esta dirección.

Consideramos que pese a las limitaciones sufridas por la situación global, el reparto de tarea y carga ha sido el adecuado y que hemos sorteado las dificultades técnicas que hemos encontrado con solvencia. Agradecer a mi compañero y a todo el equipo el apoyo brindado en todo momento y aspirar a mejorar en aquellas cosas que este proyecto no haya alcanzado.

Hemos aprendido muchísimo con el proyecto y esperamos que este no se quede aquí, que siga evolucionando y mejorando de la mano de futuros alumnos interesados en el ámbito de las comunicaciones y el V2X. A todos ellos, mucha suerte.

### 3.3. Trabajo a futuro

Debido a las condiciones actuales, parte del trabajo se ha adaptado para permitir adaptarse a la situación. Por tanto el trabajo a futuro lo podríamos dividir en dos partes principales:

- **Trabajo a realizar para finalizar el segundo prototipo:** En este caso se necesitaría realizar los experimentos que se desarrollaron de manera teórica a lo largo de esta memoria y que permitiesen conocer el rendimiento real del dispositivo en un entorno no controlado (un escenario con vehículos y interferencias) que no se ha podido simular desde nuestras casas.
- **Trabajo a realizar para la continuación y mejora del proyecto en general:** Si nos centramos en la continuación del proyecto en general, una vez se ha conseguido un prototipo que nos permite una comunicación de corto alcance relativamente efectiva, la continuación debería centrarse en el tratamiento de estos datos (como puede ser para la prevención de accidentes) o en el resto de comunicaciones que debe tener un vehículo (medio y largo alcance). No creemos que tenga especial relevancia el continuar mejorando la interfaz de corto alcance, ya que creemos que ésta se solucionará por completo con la llegada del 5G en los próximos años.

Por tanto creemos que habría trabajo que todavía debería ser realizado y el proyecto tiene todavía recorrido para futuros proyectos que puedan conseguir los objetivos descritos anteriormente. Con este fin, se aportaran todos los datos recabados, así como instrucciones y guías de uso del prototipo en un repositorio de github, que contendrá esto, más todos los códigos, ficheros, etc. relevantes del proyecto.

# Bibliografía

Alec García Barba Ángel Ortiz Jiménez César Alejandro Hidalgo Aldo Chávez Gallardo. (2018). Comunicación Car to Car. Ciudad de México: Propia.

Andrewrapp, A. user. (2016, 31 diciembre). xbee-arduino. Recuperado 1 noviembre, 2019, de <https://github.com/andrewrapp/xbee-arduino>

arduinofanboy. (2018). ESP32 BLE + Android + Arduino IDE = AWESOME. agosto 20,2019, de instructables Sitio web: <https://www.instructables.com/id/ESP32-BLE-Android-App-Arduino-IDE-AWESOME/>

Digi. (2018). XBee-PRO 900HP/XSC RF Modules. agosto 20,219, de Digi Sitio web: <https://www.digi.com/resources/documentation/digidocs/pdfs/90002173.pdf>

ETSI TS. (2014). ETSI TS 102 894-2. Francia: ETSI TS

Jorge Javier Anaya Catalán. (2016). Sistemas de comunicaciones V2X para vehículos inteligentes como soporte para sistemas cooperativos. Madrid: Propia.

Juan Gascón Repullés Eloi Pascual Belda. (2019). Short Range and Cellular V2X Communications. Ciudad de México: Propia.

Nordic Semiconductor. (2008). nrf24l01+ Single Chip Transceiver 2.4GHz . Trondheim: Propia.

Oscar Emanuel Vicente Mijangos Esteban Hernan Alcocer Souza Alfonso Ortíz Jiménez. (2017). Comunicaciones Car to Car (C2C) Avance de segundo término. Ciudad de México: Propia.

randomnerstutorials. (2018). esp32 dual core arduino ide. agosto 20, 2019, de randomnerstutorials Sitio web: <https://randomnerdtutorials.com/esp32-dual-core-arduino-ide/>

Wasmote - Open Source Sensor Node for the Internet of Things — NB-IoT, Cat-M, ZigBee, Sigfox, LoRaWAN, 3G / 4G / GPRS — Libelium — Libelium. (s.f.). Recuperado 30 octubre, 2019, de <http://www.libelium.com/products/wasmote/>

# **Capítulo 4**

## **Anexos**

# 4.1. AI: Poster



## Improvements in Short Range V2X Communication



Eloi Pascual Belda A01756116 ITSE, Juan Gascón Repullés A01756117 ITSE  
 Asesores: Dr. Javier Izquierdo Reyes y Dr. Martín Rogelio Bustamante Bello  
 Enero-Junio 2020

UNIVERSITAT POLITÈCNICA DE VALÈNCIA  
 Ingeniería en sistemas y servicios de telecomunicación

**Introducción**

De acuerdo con la Universidad de Delaware los vehículos autónomos consumen bastante menos combustible que los automóviles convencionales, en concreto entre un 19 y un 22%. Además, la optimización de la conducción a través de la tecnología hará posible que los usuarios lleguen a su destino entre un 26 y un 30% más rápido que en los automóviles conducidos por humanos.

La comunicación entre vehículos es un paso más en esta dirección

**Problemática**

La latencia entre mensajes y la limitación de caracteres hacían imposible un comportamiento eficiente y fiable por parte del dispositivo

**Objetivos**

General:  
 Mejorar el sistema de comunicaciones actual en cuanto a transmisión y portabilidad

Específicos:  
 - Diseñar una PCB que integre el sistema  
 - Desarrollar un canal de datos y un canal de control  
 - Diseño de escenarios de test

**Sistema propuesto**

El nuevo sistema se basa en el prototipo inicial, pero incluye un nuevo componente (el NRF24L01) que permite una comunicación más rápida y la creación de dos canales de comunicaciones separados

**Nuevo dispositivo**

Elementos escogidos para el prototipo

- Microcontrolador: ESP-32 DEVKIT V1.
- Comunicación por el canal de control: Xbee Series3.
- Comunicación por el canal de datos: NRF24L01.
- Standard de mensajes: basado en ETSI TS 102 894-2.

**Nuevo sistema de comunicación y tablas comparativas**

El nuevo diseño integra el canal de control en el dispositivo Xbee, mientras que se ha incorporado un nuevo dispositivo más rápido para el canal de datos: el NRF24L01

La incorporación del NRF24L01 a nuestro prototipo, nos permite el uso del sistema de ACKs.

Es muy útil incluir datos en el campo de payload del ACK y esto permite que el sistema intercambie información utilizando un mensaje principal como solicitud y ACK como respuesta.

Short Range Comm.	Prototype 1	Prototype 2
Maximum Data rate	250 kbps	2Mbps
Message latency	100 ms	9 ms
ACK	✗	✓
Max nodes per second	10	100
Proto board	✗	✓
Max range with visibility	1 km	200 m

**Herramientas utilizadas**

**Trabajo futuro**

Testear el sistema en los escenarios desarrollados y obtener información del OBD-bus

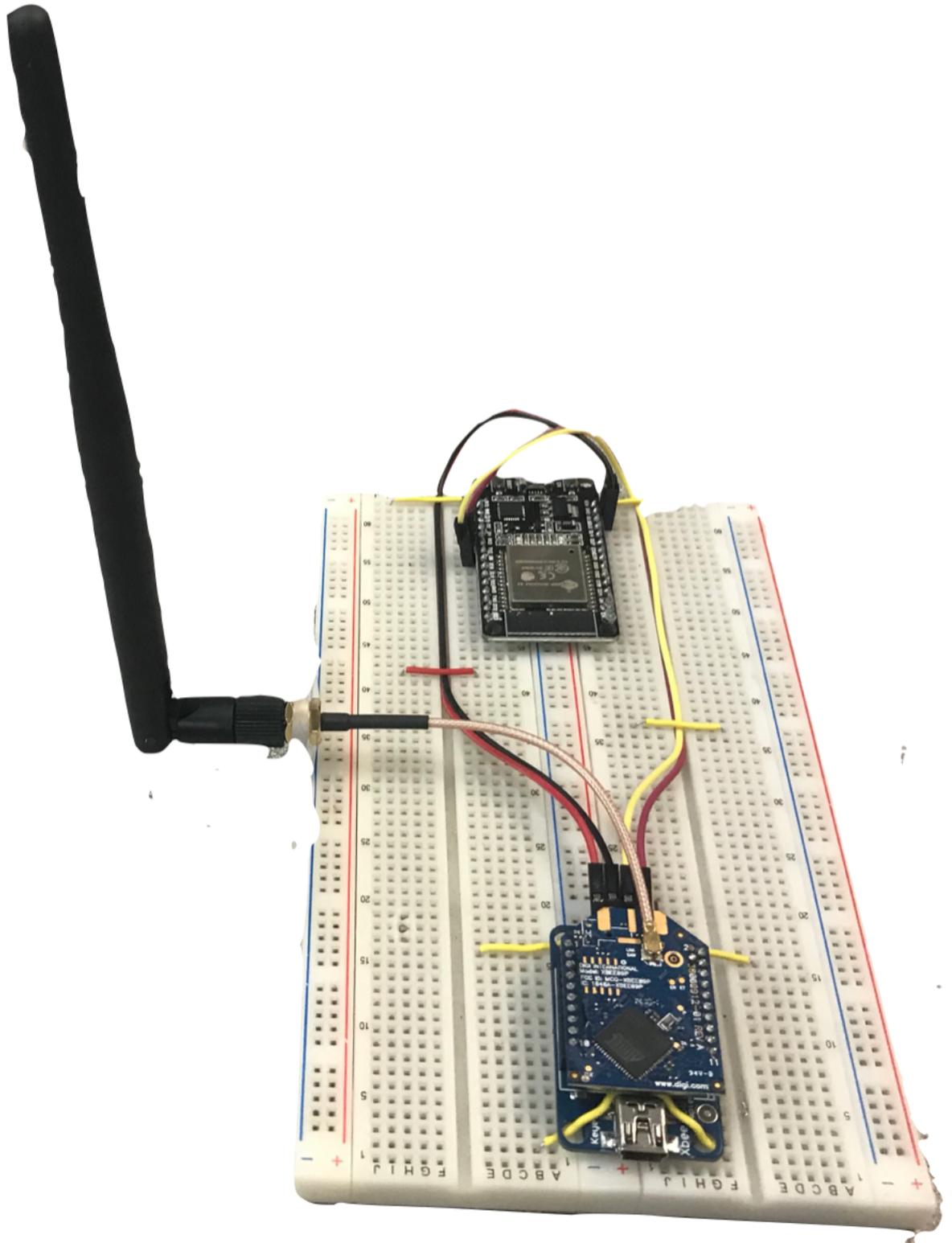
**Conclusiones**

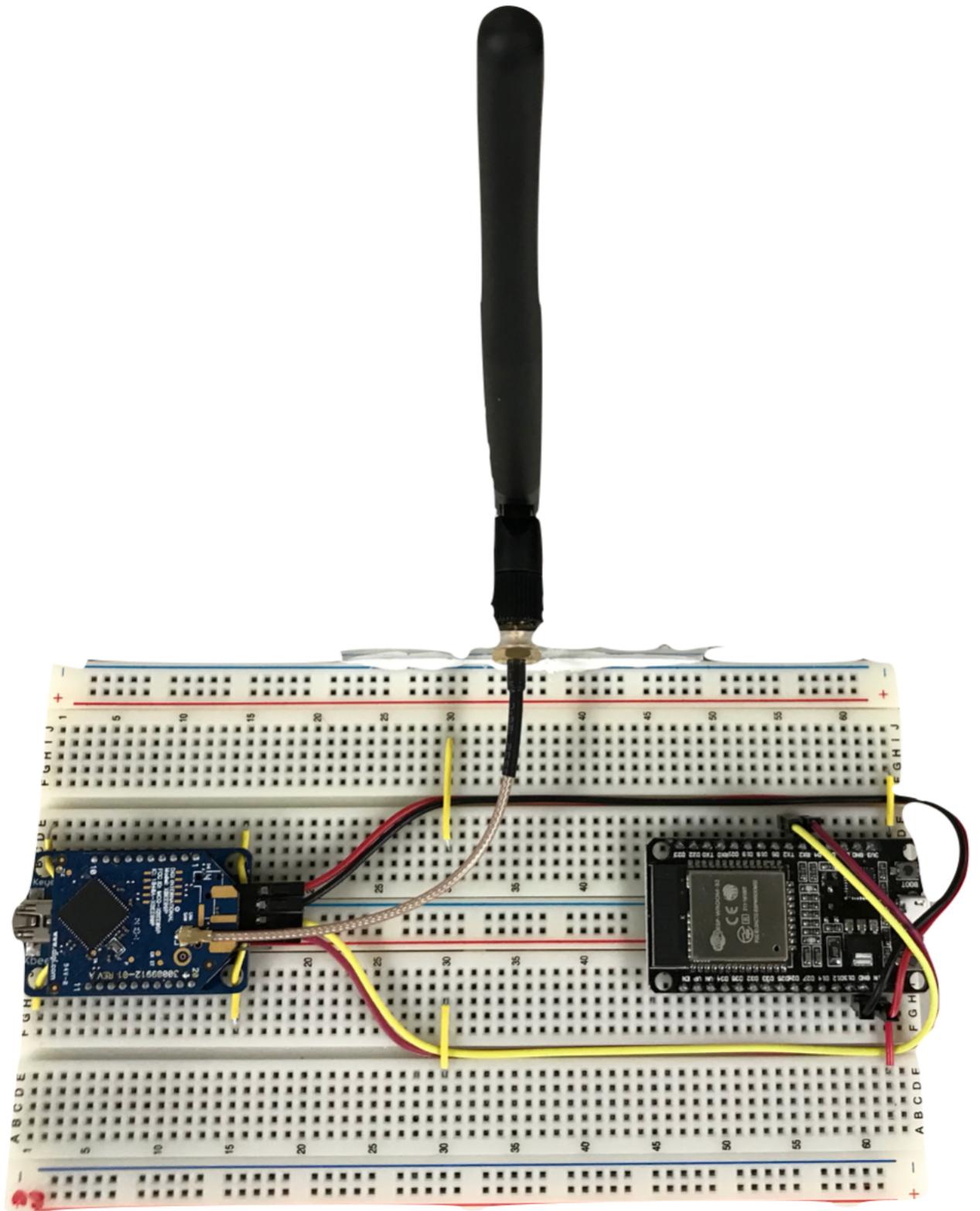
El nuevo dispositivo es casi 10 veces más rápido que el primero. También incorpora el sistema ACK que mejora el rendimiento general.

Además, la nueva comunicación de dos canales (datos y control) permite al prototipo resolver situaciones más complejas con una mejor velocidad de transmisión.



## 4.2. AII: Primer prototipo





### 4.3. AIII: Estándar ZigBee



1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16

## ZigBee Specification

ZigBee Document 05-3474-21	
August 5, 2015	
Sponsored by: ZigBee Alliance	
Accepted by	ZigBee Alliance Board of Directors
Abstract	The ZigBee Specification describes the infrastructure and services available to applications operating on the ZigBee platform.
Keywords	ZigBee, Stack, Network, Application, Profile, Framework, Device Description, Binding, Security

17  
18

August 5, 2015

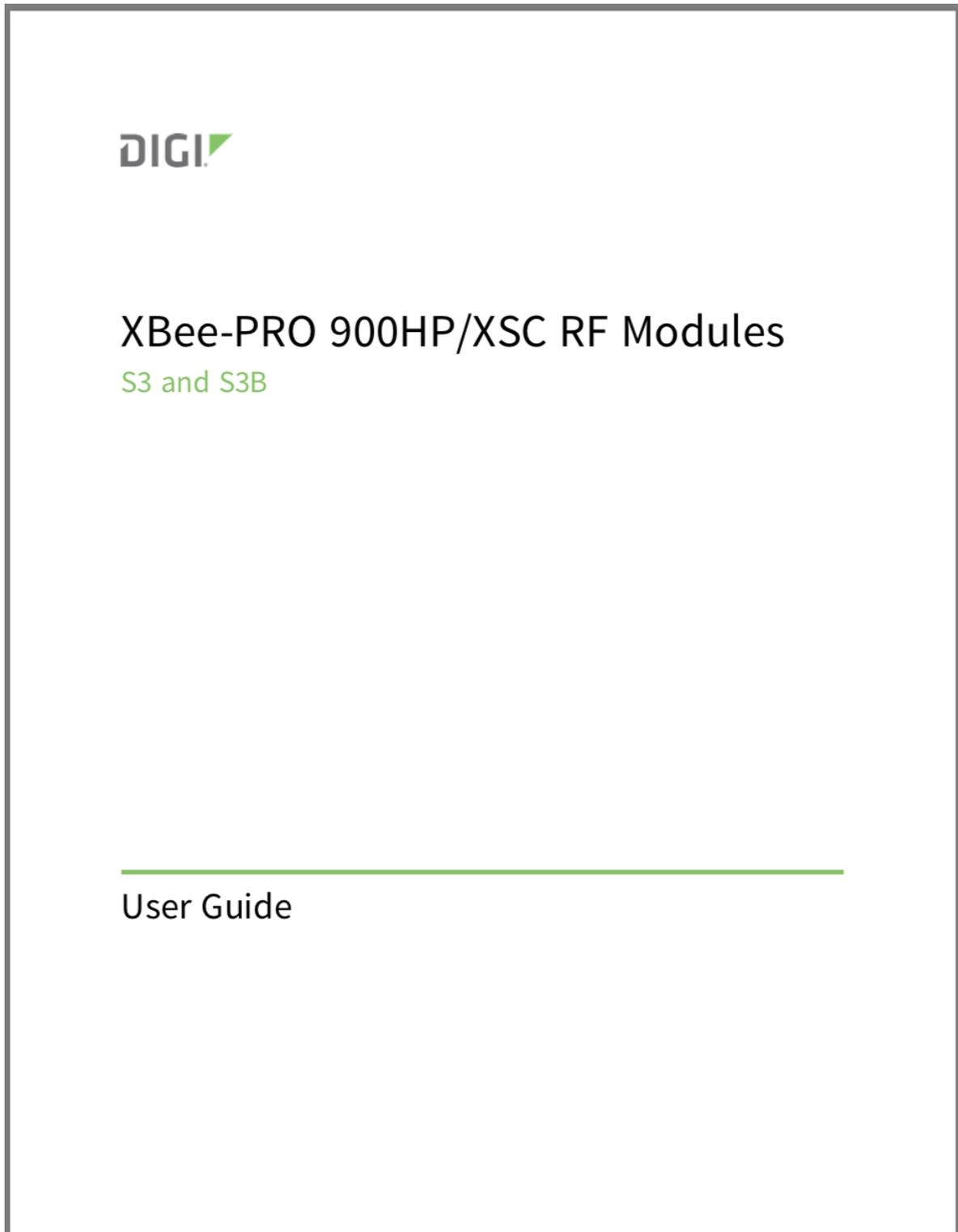
#### 4.4. AIV: ETSI TS 102 894-2 V1.2.1

ETSI TS 102 894-2 V1.2.1 (2014-09)



**Intelligent Transport Systems (ITS);  
Users and applications requirements;  
Part 2: Applications and facilities layer  
common data dictionary**

## 4.5. AV: Datasheet XBee



## 4.6. AVI: Datasheet NRF24L01



# nRF24L01+

## Single Chip 2.4GHz Transceiver

### Preliminary Product Specification v1.0

#### Key Features

- Worldwide 2.4GHz ISM band operation
- 250kbps, 1Mbps and 2Mbps on air data rates
- Ultra low power operation
- 11.3mA TX at 0dBm output power
- 13.5mA RX at 2Mbps air data rate
- 900nA in power down
- 26µA in standby-I
- On chip voltage regulator
- 1.9 to 3.6V supply range
- Enhanced ShockBurst™
- Automatic packet handling
- Auto packet transaction handling
- 6 data pipe MultiCeiver™
- Drop-in compatibility with nRF24L01
- On-air compatible in 250kbps and 1Mbps with nRF2401A, nRF2402, nRF24E1 and nRF24E2
- Low cost BOM
- ±60ppm 16MHz crystal
- 5V tolerant inputs
- Compact 20-pin 4x4mm QFN package

#### Applications

- Wireless PC Peripherals
- Mouse, keyboards and remotes
- 3-in-1 desktop bundles
- Advanced Media center remote controls
- VoIP headsets
- Game controllers
- Sports watches and sensors
- RF remote controls for consumer electronics
- Home and commercial automation
- Ultra low power sensor networks
- Active RFID
- Asset tracking systems
- Toys

## 4.7. AVII: Especificaciones XBee

SPECIFICATIONS	Digi XBee-PRO® 900HP	Programmable Digi XBee-PRO® 900HP
<b>HARDWARE</b>		
PROCESSOR	ADF7023 transceiver, Cortex-M3 EFM32G230 @ 28 MHz; Programmable includes: Freescale MC9S08QE32	
FREQUENCY BAND	902 to 928 MHz, software selectable channel mask for interference immunity	
ANTENNA OPTIONS	Wire, U.FL and RPSMSA	
<b>PERFORMANCE</b>		
RF DATA RATE	10 Kbps or 200 Kbps	
INDOOR/URBAN RANGE*	10 Kbps: up to 2000 ft (610 m); 200 Kbps: up to 1000 ft (305 m)	
OUTDOOR/ LINE-OF-SIGHT RANGE*	10 Kbps: up to 9 miles (15.5 km); 200 Kbps: up to 4 miles (6.5 km) (with 2.1dB dipole antennas)	
TRANSMIT POWER	Up to 24 dBm (250 mW) software selectable	
RECEIVER SENSITIVITY	-101 dBm @ 200 Kbps, -110 dBm @ 10 Kbps	
<b>FEATURES</b>		
DATA INTERFACE	UART (3V), SPI	
GPIO	Up to 15 Digital I/O, 4 10-bit ADC inputs, 2 PWM outputs	
NETWORKING TOPOLOGIES	DigiMesh, Repeater, Point-to-Point, Point-to-Multipoint, Peer-to-Peer	
SPREAD SPECTRUM	FHSS (Software Selectable Channels)	
<b>PROGRAMMABILITY</b>		
MEMORY	N/A	32 KB Flash / 2 KB RAM
CPU/CLOCK SPEED	N/A	HCS08 / Up to 50.33 MHz
<b>POWER</b>		
SUPPLY VOLTAGE	2.1 to 3.6 VDC	2.4 to 3.6 VDC
TRANSMIT CURRENT	215 mA	229 mA
RECEIVE CURRENT	29 mA	44 mA
SLEEP CURRENT	2.5 uA	3 uA
<b>REGULATORY APPROVALS</b>		
FCC (USA)	MCQ-XB900HP	
IC (CANADA)	1846A-XB900HP	
C-TICK (AUSTRALIA)	Yes	
ANATEL (BRAZIL)	Yes	
IDA (SINGAPORE)	Yes	
<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>(top view)</p> </div> <div style="text-align: center;"> <p>(side views)</p> </div> </div>		

## 4.8. AVIII: Especificaciones NRF24L01

PRELIMINARY PRODUCT SPECIFICATION



Single chip 2.4 GHz Transceiver

**nRF24L01**

### FEATURES

- True single chip GFSK transceiver
- Complete OSI Link Layer in hardware
- Enhanced ShockBurst™
- Auto ACK & retransmit
- Address and CRC computation
- On the air data rate 1 or 2Mbps
- Digital interface (SPI) speed 0-8 Mbps
- 125 RF channel operation
- Short switching time enable frequency hopping
- Fully RF compatible with nRF24XX
- 5V tolerant signal input pads
- 20-pin package (QFN20 4x4mm)
- Uses ultra low cost +/- 60 ppm crystal
- Uses low cost chip inductors and 2-layer PCB
- Power supply range: 1.9 to 3.6 V

### APPLICATIONS

- Wireless mouse, keyboard, joystick
- Keyless entry
- Wireless data communication
- Alarm and security systems
- Home automation
- Surveillance
- Automotive
- Telemetry
- Intelligent sports equipment
- Industrial sensors
- Toys

### GENERAL DESCRIPTION

nRF24L01 is a single chip radio transceiver for the world wide 2.4 - 2.5 GHz ISM band. The transceiver consists of a fully integrated frequency synthesizer, a power amplifier, a crystal oscillator, a demodulator, modulator and Enhanced ShockBurst™ protocol engine. Output power, frequency channels, and protocol setup are easily programmable through a SPI interface. Current consumption is very low, only 9.0mA at an output power of -6dBm and 12.3mA in RX mode. Built-in Power Down and Standby modes makes power saving easily realizable.

### QUICK REFERENCE DATA

Parameter	Value	Unit
Minimum supply voltage	1.9	V
Maximum output power	0	dBm
Maximum data rate	2000	kbps
Supply current in TX mode @ 0dBm output power	11.3	mA
Supply current in RX mode @ 2000 kbps	12.3	mA
Temperature range	-40 to +85	°C
Sensitivity @ 1000 kbps	-85	dBm
Supply current in Power Down mode	900	nA

Table 1 nRF24L01 quick reference data

## 4.9. AIX: Datasheet ESP-32 DEVKIT V1

# ESP32 Series Datasheet

### Including:

ESP32-D0WD  
ESP32-D0WDQ6  
ESP32-D2WD  
ESP32-S0WD



Version 3.1  
Espressif Systems  
Copyright © 2019

## 4.10. AX: Código y pruebas realizadas

elpasbel / v2x Private

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Security Insights Settings

No description, website, or topics provided. Edit

Manage topics

41 commits 1 branch 0 releases

Branch: master New pull request Create new file Upload files Find file Clone or download

juangas Add files via upload Latest commit 82100a4 17 days ago

Pruebas	Add files via upload	17 days ago
RX_APImode	Tx_Rx_Bluetooth	2 months ago
SerialToSerialBT	Tx_RX_Bluetooth_WebServer	2 months ago
TX_APImode	Add files via upload	2 months ago
Tx_Rx	Añadidas mas pruebas	last month
Tx_Rx_Con_Mensajes	Corregido error de no envío	22 days ago
App_bluetooth_Web.aia	Tx_RX_Bluetooth_WebServer	2 months ago
Falta por hacer.txt	Add files via upload	last month
ts_10289402v010201p.pdf	Add files via upload	2 months ago

Todos los códigos nombrados a lo largo de la memoria están disponibles en el siguiente repositorio de github: <https://github.com/elpasbel/v2x>

Aparte, también encontramos el catálogo de los mensajes así como los programas utilizados para la realización de las pruebas.

En este repositorio están alojadas también las hojas de cálculo originales con los datos reales, desde las cuales se obtuvieron las gráficas mostradas a lo largo de la memoria. Estos datos se corresponden a las pruebas descritas a lo largo de la caracterización del dispositivo.