

Universidad Católica Andrés Bello
Facultad de Ingeniería
Escuela Ingeniería de Telecomunicaciones

**DISEÑO DE UN SISTEMA DE TELEMETRÍA PARA LA
MONITORIZACIÓN DE PARÁMETROS EN LOS VEHÍCULOS
DE COMPETENCIAS SAE.**

REALIZADO POR: Goncalves De Gois, Armando. C.I.: 20.630.696
Maroun Meneses, Alejandro. C.I.: 20.246.682

TUTOR: Ing. Wilmer Pereira, PhD.

FECHA: Julio de 2014

DISEÑO DE UN SISTEMA DE TELEMETRÍA PARA LA MONITORIZACIÓN DE PARÁMETROS EN LOS VEHÍCULOS DE COMPETENCIAS SAE.

Goncalves, Armando

ajg.17.17@gmail.com

Maroun, Alejandro

alejandromaroun@gmail.com

El presente trabajo de investigación, tuvo como objetivo fundamental diseñar un sistema para la monitorización de parámetros en vehículos de competencia SAE, en este caso, velocidad, temperatura y nivel de combustible. Para ello, se hizo necesario utilizar sensores capaces de medir dichos parámetros, así como también la placa Arduino, y equipos de comunicación XBee, basados en el estándar IEEE 802.15.4, los cuales permiten establecer un enlace de corto alcance entre el vehículo y un puesto de control; dicho sistema se apoya sobre una plataforma de software sencilla en lenguaje JAVA, de fácil manejo para el usuario encargado de supervisar y monitorizar los parámetros. La intención de este sistema es elevar el nivel de desempeño durante carreras de competencia Formula y Baja SAE, donde se requiere que detrás del piloto que va conduciendo, se cuente con el seguimiento de todos los eventos que ocurren en el vehículo y de los posibles percances que se puedan presentar, con la finalidad de tener un tiempo de reacción cada vez menor y obtener mejores resultados en la competencia. En este sentido se realizaron pruebas estáticas y dinámicas. Las mismas permiten concluir entre otras cosas que la efectividad de las transmisiones de datos inalámbricas del Sistema de Telemetría, resultó ser significativamente dependiente del porcentaje de la línea de visión entre los dispositivos XBee. También se concluye que se pueden alcanzar velocidades de hasta 90 km/h sin que la comunicación se vea comprometida, ya que la efectividad se mantuvo por encima del 90% en espacios abiertos.

Descriptores: Sistema de Telemetría, monitorización, vehículos de competencia, XBee, SAE.

DEDICATORIA

Este Trabajo Especial de Grado, va dedicado en primer lugar a mis padres Ana Maria De Gois y Agustin Goncalves, los cuales me aconsejaron, y me apoyaron en todos los aspectos de mi vida además de brindarme la oportunidad de realizar mis estudios de ingeniería en la UCAB; a mi hermana y mi abuelo que siempre me acompañan. En igual medida le dedico este trabajo a mi novia Lisbeth Gómez quien siempre estuvo a mi lado ayudándome en todo y brindándome una sonrisa cuando más lo necesitaba.

En un último lugar y no menos importante a todos mis compañeros de la universidad, sin ellos este camino hubiese sido muy distinto.

Armando

A mi hermana Desirée, a quien le debo mucho por siempre acompañarme, por darme el cariño y los ánimos necesarios en momentos difíciles.

De igual forma, le dedico este trabajo a mi madre Margarita, quien me enseñó a cómo alcanzar las metas con dedicación, hábitos y valores, a nunca abandonar. Para la mejor, por estar siempre a mi lado en las peores situaciones.

A mi padre José Antonio, quien a pesar de no haberme acompañado en todo momento, me ha apoyado desde lejos.

A mis abuelos Tita, Salma, Jaoudat, Manolo y a mi familia en general, quienes han sido ejemplos de esfuerzo y constancia en mi vida, porque me han brindado su apoyo incondicional y por compartir conmigo buenos y malos momentos.

Alejandro

AGRADECIMIENTOS

En primer lugar a Dios por habernos permitido culminar la carrera siendo nuestra fortaleza frente a momentos duros y obstáculos.

A la ilustre Universidad Católica Andrés Bello por brindarnos los espacios necesarios para el desarrollo de las actividades académicas.

A nuestro tutor Ing. Wilmer Pereira por todo el apoyo y conocimiento brindado durante la elaboración de este Trabajo Especial de Grado.

A nuestros compañeros Cesar Chaurel y Francisco Da Silva en conjunto con todo el equipo de Baja SAE, quienes nos brindaron sus experiencias para la realización del sistema de telemetría.

A nuestros padrinos de promoción el Ing. Wilfredo Torres y a la Ing. María Gabriela Rodríguez, por toda la solidaridad y consejos durante la carrera universitaria, junto a los buenos momentos que compartimos durante todos estos años.

Por último, a toda la planta profesoral de la Escuela de Telecomunicaciones, quienes cumpliendo con su sagrado deber, nos dieron la oportunidad de alcanzar hoy nuestras metas.

ÍNDICE GENERAL

INTRODUCCIÓN	1
CAPÍTULO I: EL PROBLEMA	3
I.1. Planteamiento del Problema	3
I.2. Objetivos de la Investigación	5
I.3. Alcances y Limitaciones	5
I.4. Justificación de la Investigación	6
CAPÍTULO II: MARCO TEÓRICO	8
II.1. Antecedentes de la Investigación	8
II.2. Bases Teóricas Referenciales	10
II.2.1 Hardware	10
II.3. Protocolos de transmisión inalámbrica	17
II.3.1 IEEE 802.15: Redes Inalámbricas de Área Personal (WPAN)	18
II.4. Equipos XBee/RF	23
II.4.1 Redes Multinodo	24
II.4.2 Acceso al Medio	24
II.4.3 Direccionamiento	25
II.4.4 Comprobación de Error y Acuses de recibo	26
II.4.5 Encapsulación	26
II.4.6 Aplicación	26
II.4.7 Beneficios de XBee	27
II.4.8 Modos de comunicación	28
II.4.9 Sistema energético	31
II.4.10 Tipos de Módulos XBee	31
II.5. Software	39
II.5.1 Java	39
II.5.2 X-CTU	39
II.6. Sistemas de Telemetría	40
II.6.1 Elementos de un sistema de telemetría	41
II.7. Organización SAE internacional	42
II.7.1 Automóviles pertenecientes a la organización	42
CAPÍTULO III: MARCO METODOLÓGICO	44
III.1. Investigación documental y análisis de las fuentes	44
III.1.1 Parámetros de medición del Sistema de Telemetría	45
III.1.2 Selección del Hardware para el manejo de los datos	45
III.1.3 Selección del protocolo de transmisión a utilizar	45
III.1.4 Sistema de interpretación de datos (Interfaz)	46
III.2. Diseño de la Propuesta	46
III.2.1 Sensores:	46
III.2.2 Tarjetas modulares	48

III.2.3	Dispositivos de comunicación	48
III.3.	Prueba del Sistema y Cálculos.....	48
III.3.1	Pruebas estáticas	49
III.3.2	Pruebas Dinámicas:.....	50
III.4.	Análisis de Resultados.....	52
III.5.	Conclusiones y Recomendaciones:	52
III.6.	Elaboración del libro y presentación del Trabajo de Grado	52
CAPÍTULO IV: RESULTADOS.....		54
IV.1.	Configuración de los Módulos XBee	54
IV.2.	Etapas 1: Pruebas Estáticas:	55
IV.3.	Etapas 2: Pruebas Dinámicas	57
CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES.....		66
V.1.	Conclusiones.....	66
V.2.	Recomendaciones	68
Referencias Bibliográficas		69
ANEXO A.....		71
Fotografías durante Pruebas Experimentales		71
ANEXO B.....		74
Diagrama Sensor de Temperatura y Shield Arduino		74
ANEXO C		76
Código del módulo de procesamiento de datos Arduino		76
ANEXO D.....		78
Código de la Clase SerialTest de JAVA		78
ANEXO E		81
Código de la Clase UI (Interfaz) de JAVA		81

Índice de Tablas

Tabla 1: Características del Globalsat GPS module EM-506	17
Tabla 2: Comparación entre el modelo OSI y el estándar IEEE 802.15.4.....	23
Tabla 3: Características técnicas de los XBee.....	32
Tabla 4: Comparación entre XBee y XBee-Pro.....	34
Tabla 5: Configuración del XBee.	54
Tabla 6: Pruebas estáticas con línea de Visión (Outdoor).	55
Tabla 7: Pruebas estáticas sin línea de visión (Indoor).	56
Tabla 8: Pruebas Dinámicas a Velocidades Bajas (Máx. 30 km/h).	57
Tabla 9: Pruebas Dinámicas a Velocidades Intermedias (Máx. 60 km/h).	58
Tabla 10: Pruebas Dinámicas a Velocidades Altas (Máx. 90 km/h).....	58
Tabla 11: Pruebas Dinámicas (Radio de 300 m), Efectividad prom. por Velocidad.	59
Tabla 12: Pruebas Dinámicas a Velocidades Altas (Máx. 90 km/h).....	60
Tabla 13: Pruebas Dinámicas (Radio de 600m), Efectividad vs Línea de Visión.	61
Tabla 14: Pruebas Dinámicas Modo API (Radio de 600 m), Efectividad vs Línea de Visión.....	62
Tabla 15: Modo AT vs Modo API (Radio de 600 m), Velocidad Máx. 90 km/h.	63

Índice de Figuras

<i>Figura 1.</i> Tarjeta Arduino MEGA 2560.	11
<i>Figura 2.</i> IO Expansion Shield V2.1.	13
<i>Figura 3.</i> Sensor Savant FS-3400 BH (Flujo de Combustible).	14
<i>Figura 4.</i> Sensor LM-35 (Temperatura).	15
<i>Figura 5.</i> Encapsulación XBee.	26
<i>Figura 6.</i> Trama en modo API.....	30
<i>Figura 7.</i> Pines de Módulos XBee.....	31
<i>Figura 8.</i> Diagrama Transmisión de nodos punto a punto.....	33
<i>Figura 9.</i> Enrutamiento de los nodos XBee.....	35
<i>Figura 10.</i> Reflexiones de las ondas.	37
<i>Figura 11.</i> Zona de Fresnel.	38
<i>Figura 12.</i> Diseño del sistema de telemetría.....	47
<i>Figura 13.</i> Pruebas Estáticas.....	50
<i>Figura 14.</i> Pruebas dinámicas.....	51
<i>Figura 15.</i> Radio de las pruebas dinámicas.	51

<i>Figura 16.</i> Pruebas estáticas (Outdoor): Efectividad vs Distancia.	55
<i>Figura 17.</i> Pruebas estáticas sin línea de visión (Indoor).	56
<i>Figura 18.</i> Pruebas Dinámicas (Radio de 300 m): Efectividad vs Línea de Visión. ...	59
<i>Figura 19.</i> Pruebas Dinámicas (Radio 300 m): Velocidad vs Efectividad.	60
<i>Figura 20.</i> Pruebas Dinámicas a Velocidades Altas (Max 90 km/h).	61
<i>Figura 21.</i> Pruebas Dinámicas (Radio de 600m).	61
<i>Figura 22.</i> Pruebas Dinámicas Modo API (Radio de 600 m).	63
<i>Figura 23.</i> Modo AT vs Modo API (Radio 600m).	64
<i>Figura 24.</i> Interfaz del sistema de telemetría.	65

INTRODUCCIÓN

Los procesos y cambios continuos en las comunicaciones son de vital importancia para la evolución humana. A medida que en el mercado global se exige un desarrollo más alto y preciso, se inicia la puesta en marcha de procesos que permitan adquirir de forma óptima las mejores tecnologías de telemetría y así cumplir con los estándares de calidad que se tienen y se exigen hoy en día.

En este sentido, el hombre en su afán de buscar nuevas soluciones, ha sido capaz de desarrollar gracias a la tecnología, sistemas de medición más precisos. En el caso particular del campo de las comunicaciones a distancia, se ha venido desarrollando sistemas inalámbricos que hacen posible la transmisión de datos a gran distancia y velocidad. Para las competencias automovilísticas se utilizan sistemas de telemetría, que permiten monitorear las distintas variables o parámetros del vehículo en competición.

Es así como existen competencias como las de la organización SAE, las cuales involucra a las universidades de todo el mundo, a participar en una carrera de vehículos en donde se prueban los conocimientos y habilidades de cada grupo de jóvenes.

Mayormente cada equipo trabaja durante meses en el comportamiento mecánico del vehículo, en la seguridad del pasajero y de los alrededores del vehículo durante la carrera, entre otros. Vale destacar que en el ámbito estratégico, son necesarios distintos componentes que establezcan comunicación entre el piloto y el resto del equipo detrás de la pista de carreras. Es así pues, donde la presente investigación tiene como objetivo fundamental, diseñar el sistema de telemetría para la monitorización de parámetros en vehículos de competencias SAE con la finalidad de mejorar el rendimiento del vehículo y por ende, su desempeño en las pruebas de competición. Algunos de estos parámetros son, el nivel de combustible, la velocidad y la temperatura.

En este sentido, la telemetría como proceso altamente automatizado permite la medición y la obtención de datos a distancia para su posterior transmisión. Estos

datos ayudan a diagnosticar y evaluar las condiciones en las cuales se encuentra el vehículo en competencia, que facilitarán la toma de decisiones por parte del equipo de Formula o Baja SAE.

Así el trabajo de investigación quedó estructurado de la siguiente manera:

El capítulo I presenta los aspectos más importantes que constituyen la problemática de la investigación, los objetivos y la justificación de la investigación. Por su parte el capítulo II describe los antecedentes y los fundamentos teóricos en los cuales se sustentó el presente trabajo. Así mismo, el capítulo III contiene los aspectos metodológicos y técnicos requeridos para el desarrollo del trabajo. El capítulo IV contempla los análisis y los resultados a los cuales se llegó una vez realizadas las pruebas y experimentos. Finalmente se presenta el capítulo V, el cual contiene el cuerpo de conclusiones y recomendaciones derivados del proceso de análisis, una vez culminado el proceso de investigación.

CAPÍTULO I

EL PROBLEMA

I.1. Planteamiento del Problema

A lo largo de la historia, la evolución de los vehículos automotores ha sido uno de los hechos más predominantes a partir de su creación. El ser humano junto con el crecimiento de las sociedades, ha estado acompañado de la necesidad de transportarse de un lugar a otro. El avance de la industria automotriz no ha dejado de crecer ya que la tecnología ha estado en paralelo con la arquitectura de los vehículos. Pero más allá de cumplir con una necesidad, los carros se han convertido en una afición, en un hobby y en competencias deportivas.

Por otra parte, para las compañías fabricantes de vehículos y para sus clientes, es importante el completo funcionamiento del conjunto de partes que conforman el vehículo, en especial todo lo relacionado con el motor, suspensión, transmisión, aerodinámica y seguridad. Más allá de eso, también son importantes las pruebas que se realizan antes de ofrecer el vehículo para verificar el buen funcionamiento del mismo, ya sean de velocidad, pruebas de estabilidad, pruebas de seguridad, pruebas de tiempo, entre muchas más. Actualmente, dentro de los carros, existe una serie de sensores de tipo informativo que permiten al conductor conocer los parámetros importantes al momento de manejar, tales como: velocidad del vehículo, vueltas por minuto del motor, temperatura del motor, nivel de combustible, odómetro, entre otros dependiendo de la categoría del vehículo.

Desde esta misma perspectiva, se tiene que a nivel de competencia, los sensores juegan un papel crucial, ya que los requerimientos son mayores y a su vez existen mayores riesgos de que ocurran eventos inesperados. Vale la pena destacar entre otras, las competencias SAE, donde participan universidades de todo el mundo, con vehículos ensamblados y contruidos por estudiantes en su mayoría, para lo cual es de suma importancia la supervisión y monitorización sobre dichas máquinas de tal manera que los conductores corran las carreras con un nivel de protección adecuado.

Teniendo en cuenta que es posible que existan percances durante la competencia, es de gran valor que tanto los vehículos como el conductor, tengan a su disposición detrás de la pista el apoyo del resto del equipo en los *Pits*, donde se pueden resolver eventos de poca complejidad debido al tiempo, siendo este el factor más importante que promueve la competencia.

Es por ello que el tiempo de respuesta del equipo técnico entra en el juego estratégico de la carrera. Lamentablemente en los vehículos de competencia SAE, no existe comunicación entre lo que reportan los sensores y el grupo que está detrás de la pista, tampoco el equipo puede saber en dónde está el vehículo en determinado momento por no estar a la vista desde la parada o *Pits*. La única forma de que el equipo técnico se informe de los eventos es a través de la conversación con el piloto vía radio.

El problema radica en que el conductor se ve comprometido a mantener la conversación con el equipo, lo cual disminuye sus condiciones de manejo y de concentración, ya que debe invertir el tiempo en informar a su equipo vía radio sobre todos los hechos.

Estratégicamente, ciertos parámetros como el nivel de combustible, resultan indispensables durante la carrera, ya que si el vehículo se queda sin combustible, automáticamente el equipo queda descalificado, pero si el tanque está completamente lleno, reduce la velocidad del vehículo, esto sin contar que cada vez que se detenga a recargar el tanque de combustible, se pierde tiempo valioso. Lo ideal sería que desde el puesto del equipo, se encuentre un grupo de personas que se mantengan en comunicación, a través de un sistema de telemetría, que permita monitorizar los distintos parámetros relevantes. Esto permitirá entre otras cosas, evaluar todos los factores involucrados en el desarrollo de la carrera, lo que se traducirá en lograr mejores posiciones y clasificaciones en las mismas. En todo caso siempre resultará de gran utilidad el tener alertas que avisen cuando existe algún parámetro por encima o por debajo de lo normal.

Lo mencionado hasta ahora, permite sostener que resulta de vital importancia contar en el puesto de control dentro de los *Pits*, con la información del vehículo en

tiempo real, para idear la mejor estrategia ganadora en la competencia. Adicionalmente se tiene un respaldo del funcionamiento del vehículo en la pista, para así determinar su nivel de desempeño, las posibles fallas y mejores logros. Esto permitirá al equipo una ventaja significativa frente a sus oponentes.

I.2. Objetivos de la Investigación

Objetivo General:

Proponer un Sistema de Telemetría que permita la Monitorización de Parámetros en los Vehículos de Competencias SAE (STMPVC-SAE).

Objetivos Específicos:

1. Identificar los parámetros más importantes que van a conformar el sistema de medición a distancia.
2. Determinar el hardware para la adquisición de los datos en el vehículo.
3. Analizar la tecnología de transmisión inalámbrica más adecuada para la comunicación entre el vehículo y el puesto de control.
4. Desarrollar un software que permita la interpretación de los datos extraídos del vehículo.
5. Diseñar el Sistema de Telemetría para la Monitorización de Parámetros en Vehículos de Competencias SAE (STMPVC-SAE).
6. Demostrar el funcionamiento del sistema de telemetría propuesto mediante procesos de pruebas y cálculos respectivos.

I.3. Alcances y Limitaciones

La presente investigación tuvo por objeto diseñar un sistema de telemetría para la monitorización de al menos tres parámetros en vehículos de competición SAE (STMPVC-SAE), tomando en cuenta el diagnóstico de necesidades suministrado por los equipos de Fórmula y Baja SAE de la Universidad Católica Andrés Bello (UCAB).

Para ello, se hizo necesario la revisión de aspectos relacionados con los sistemas de telemetría, y más específicamente, en aquellos vinculados con el ámbito

automotor. En este sentido, el interés estuvo centrado en los vehículos de competencia SAE Internacional.

Así mismo, esta investigación incluye la identificación y caracterización de los parámetros de interés que fueron medidos a distancia, la determinación del hardware para la adquisición de datos provenientes del vehículo de competición, la selección de la tecnología más adecuada para la transmisión inalámbrica de dichos datos al puesto de control y el desarrollo de un software encargado de traducir los datos en información comprensible para los usuarios del sistema.

Finalmente, se planteó la demostración del funcionamiento del sistema de telemetría mediante procesos de pruebas y cálculos respectivos; dichas pruebas se realizaron en vehículos de uso particular, ya que para la presente fecha los vehículos de competencias SAE, no estaban disponibles. Uno de ellos, se encontraba fuera del país (para el caso de BAJA SAE) y el otro en proceso de reconstrucción (para el caso de Formula SAE). Esto último, junto a la dificultad de adquirir algunos dispositivos requeridos para el diseño del sistema de telemetría, por ser en su mayoría elementos importados, fueron las limitantes fundamentales de la presente investigación

I.4. Justificación de la Investigación

En la actualidad, todo vehículo terrestre, está compuesto por un sistema mecánico con una cantidad importante de variables para analizar, con lo cual la necesidad por registrar el comportamiento de todos los elementos es una obligación, para llevar a cabo el correcto funcionamiento del mismo y poder responder ante eventualidades lo más rápido posible. El grado de complejidad que implica su monitorización, crece con la cantidad de parámetros que se deseen manejar, dentro de este sistema mecánico se pueden nombrar de manera simplificada el motor, nivel de combustible y posición del vehículo.

Debido a que el vehículo necesita de un conductor que controle sus mandos para desplazarse por el terreno, se puede afirmar que se constituye un sistema formado por el hombre, auto y medio. Por lo tanto el auto deberá responder a las maniobras pretendidas por el conductor teniendo en cuenta la influencia del medio.

De estas ideas surgen algunas características importantes que el vehículo debe cumplir.

Las unidades de control de motor determinan la cantidad de combustible, el punto de ignición y otros parámetros monitorizando el motor a través de sensores.

Formula SAE engloba todos los aspectos de negocios incluyendo investigación, diseño, manufactura, prueba, desarrollo, mercadeo, gerencia y movilización de fondos. Formula SAE toma a los estudiantes y los coloca fuera de las aulas de clases en el mundo real. Este proyecto ha consistido en realizar un sistema de telemetría con el cual se puedan monitorizar distintos parámetros del comportamiento del carro como pueden ser velocidad, temperaturas, tensión de batería, nivel de combustible, etc.

La normativa de la competición obliga a realizar un vehículo nuevo cada año y adicionalmente, se pretende mejorarlo cada vez mediante la inclusión de piezas nuevas y a través de modificaciones en general del diseño. Con la telemetría se pretende facilitar el proceso de validación de dichas piezas y ayudar a detectar de manera rápida errores que pueden ser críticos para el motor como un excesivo calentamiento, rendimiento del combustible, vueltas por minuto.

CAPÍTULO II

MARCO TEÓRICO

II.1. Antecedentes de la Investigación

La revisión exhaustiva de trabajos vinculados con la presente investigación permitió establecer los siguientes como antecedentes más importantes:

Huertas y Véliz (2006), elaboraron un trabajo de investigación titulado "Sistema Monitor Remoto Interactivo de vehículos" (SMRI), que contempla el diseño y elaboración de un sistema de telemetría que permite obtener, en tiempo real, los parámetros proporcionados por el computador del vehículo, para ser utilizado por centros de servicio y mantenimiento automotriz. Este sistema está integrado por tres componentes: unidad remota, servidor y cliente. El proceso de monitorización se inicia cuando un cliente desea conocer los parámetros de un vehículo determinado, para lo cual se registra en el servidor. La unidad remota se reporta cada cierto tiempo y, al detectar que un cliente está registrado, se encarga de la extracción de los parámetros de la computadora del vehículo a través de la interfaz OBDII. La información obtenida se transmite de manera inalámbrica al servidor empleando la tecnología CDMA 2000 1X, el cual la retransmite al cliente.

Posteriormente, Vargas (2008), presentó el trabajo titulado "Sistema de Monitoreo y Control Remoto para una Central Micro-Hidráulica", el cual persigue el diseño e implementación de un SMyCR orientado a su aplicación en GD de pequeña escala. Para ello, se desarrolla una revisión del estado del arte entorno a sistemas de adquisición de datos y gestión de la energía. A partir de esto, se plantea una metodología para el diseño y construcción del SMyCR basada en recomendaciones extraídas de estándares internacionales y guías enfocadas al monitoreo y control remoto de unidades de pequeña escala y sistemas de automatización en general; se hace referencia a los aspectos relevantes en cuanto al uso de tecnologías de comunicación y control. El SMyCR se implementa en un prototipo de central micro-hidráulica (CMH), involucrando específicamente la construcción de dispositivos electrónicos y una interfaz gráfica capaz de establecer una comunicación con el

operador local de la CMH. La interfaz gráfica, desarrollada en la plataforma comercial LabVIEW, en operación conjunta con el gestor de bases de datos MySQL, permite el análisis en tiempo real de la evolución del estado de operación de la CMH. Se concluye que el diseño e implementación de un SMyCR para la CMH ha sido satisfactorio.

Hernández (2013), realizó una investigación titulada: “Sistema de telemetría para un vehículo de fórmula S.A.E”, la cual consistió en diseñar un sistema de telemetría para un vehículo de competencia Formula SAE. A través de este trabajo, el autor logró demostrar la capacidad de recibir datos en tiempo real de un automóvil en movimiento, sin necesidad de que el mismo se detuviera para la extracción de información almacenada en su recorrido. Para ello, utilizó módulos XBee Pro 868 lo cual permitió un alcance de hasta 30km en espacio libre con línea de visión. No obstante, sufrió varios problemas con la recepción correcta de los bits que se enviaban, lo cual tuvo que resolver disminuyendo la potencia de transmisión del módulo XBee ubicado en el carro. Se toma como antecedente esta investigación, puesto que la misma nos da información sobre un sistema de telemetría implementado con dispositivos de transmisión XBee, los cuales resultaron ser suficientemente robustos en las pruebas de campo realizadas; permitiendo de esta forma conocer un modelo de transmisión exitoso en el ámbito de las carreras automovilísticas.

Se obtuvo gracias a los antecedentes investigados, una idea de los dispositivos a utilizar, el estándar capaz de soportar las velocidades en la que se desplazan los vehículos y posibles tecnologías a evaluar para la realización del presente Trabajo Especial de Grado.

II.2. Bases Teóricas Referenciales

II.2.1 Hardware

1. Arduino:

Es una plataforma de electrónica abierta para la creación de prototipos basada en software y hardware flexibles, fáciles de usar. Se creó para artistas, diseñadores, aficionados y cualquiera interesado en crear entornos u objetos interactivos.

Arduino puede tomar información del entorno a través de sus pines de entrada de toda una gama de sensores (ya sea de una señal analógica o digital), y puede afectar aquello que le rodea controlando luces, motores y otros actuadores. El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino (basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing). (Oxer y Blemings, 2009).

Existe gran cantidad de tarjetas programables realizadas por el fabricante Arduino, entre las cuales se pueden destacar Arduino UNO y Arduino MEGA.

1.1. Arduino UNO: Es una placa electrónica basada en ATmega328. Tiene 14 pines digitales de entrada y salida (de los cuales 6 pueden ser usados como salidas PWM), 6 entradas analógicas, un oscilador de cerámica a 16MHz, un conector USB, un conector de alimentación, un header ICSP, y un botón de reinicio. Contiene todo lo necesario para apoyar el microcontrolador, basta con conectarlo con un cable USB, o alimentada con un adaptador o la batería para empezar de CA a CC.

1.2. Arduino MEGA: El Arduino Mega es una placa electrónica basada ATmega1280 (datasheet). Tiene 54 entradas/salidas digitales (de las cuales 14 proporcionan salida PWM), 16 entradas digitales, 4 UARTS (puertos serie por hardware), un cristal oscilador de 16MHz, conexión USB, entrada de corriente, conector ICSP y botón de reinicio; como se puede observar en la *Figura 1*. El Mega es compatible con la mayoría de shields diseñados para el Arduino Duemilanove o Diecimila.

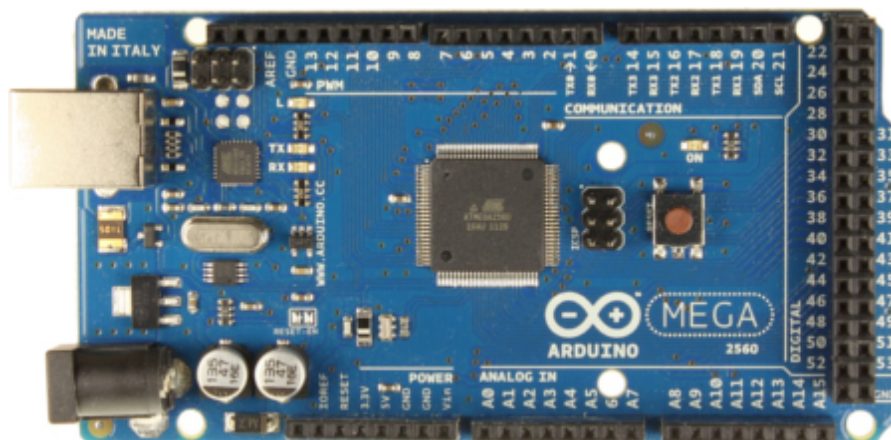


Figura 1. Tarjeta Arduino MEGA 2560.

Fuente: Arduino (2011).

El Arduino MEGA tiene 128KB de memoria flash para almacenar código (4KB son usados para el arranque del sistema (bootloader). El ATmega1280 tiene 8 KB de memoria SRAM. El ATmega1280 tiene 4KB de EEPROM.

La longitud y amplitud máxima de la placa Duemilanove es de 4 y 2.1 pulgadas respectivamente, con el conector USB y la conexión de alimentación sobresaliendo de estas dimensiones. Tres agujeros para fijación con tornillos permiten colocar la placa en superficies y cajas.

El Arduino Mega está diseñado para ser compatible con la mayoría de shields diseñados para el Diecimila o Duemilanove. Los pines digitales de 0 a 23 (y los pines AREF y GND adyacentes), las entradas analógicas de 0 a 5, los conectores de alimentación y los conectores ICPS están todos ubicados en posiciones equivalentes. Además el puerto serie principal está ubicado en los mismos pines (0 y 1), así como las interrupciones 0 y 1 (pines 2 y 3 respectivamente).

En cuanto a las entradas y salidas del dispositivo se tiene que Cada uno de los 54 pines digitales en el Duemilanove pueden utilizarse como entradas o como salidas usando las funciones `pinMode()`, `digitalWrite()`, y `digitalRead()`. Las E/S operan a 5 voltios. Cada pin puede proporcionar o recibir una intensidad máxima de 40mA y tiene una resistencia interna (desconectada por defecto) de 20-50kOhms. Además, algunos pines tienen funciones especializadas:

- Serie: 0 (RX) y 1 (TX), Serie 1: 19 (RX) y 18 (TX); Serie 2: 17 (RX) y 16 (TX); Serie 3: 15 (RX) y 14 (TX). Usado para recibir (RX) transmitir (TX) datos a través de puerto serie TTL. Los pines Serie: 0 (RX) y 1 (TX) están conectados a los pines correspondientes del chip FTDI USB-to-TTL.

- Interrupciones Externas: 2 (interrupción 0), 3 (interrupción 1), 18 (interrupción 5), 19 (interrupción 4), 20 (interrupción 3), y 21 (interrupción 2).. Estos pines se pueden configurar para lanzar una interrupción en un valor LOW(0V), en flancos de subida o bajada (cambio de LOW a HIGH(5V) o viceversa), o en cambios de valor. Ver la función attachInterrupt() para as detalles.

- PWM: de 0 a 13. Proporciona una salida PWM (Pulse Wave Modulation, modulación de onda por pulsos) de 8 bits de resolución (valores de 0 a 255) a través de la función analogWrite().

- SPI: 50 (SS), 51 (MOSI), 52 (MISO), 53 (SCK). Estos pines proporcionan comunicación SPI, que a pesar de que el hardware la proporcione actualmente no está incluido en el lenguaje Arduino.

- LED: 13. Hay un LED integrado en la placa conectado al pin digital 13, cuando este pin tiene un valor HIGH(5V) el LED se enciende y cuando este tiene un valor LOW(0V) este se apaga.

2. Shields:

Las Tarjetas Shields son placas que pueden ser conectadas encima de la placa Arduino extendiendo sus capacidades, estas siguen la misma filosofía que el conjunto original: son fáciles de montar y baratas de producir. (Ver *Figura 2*). (Oxer y Blemings, 2009)

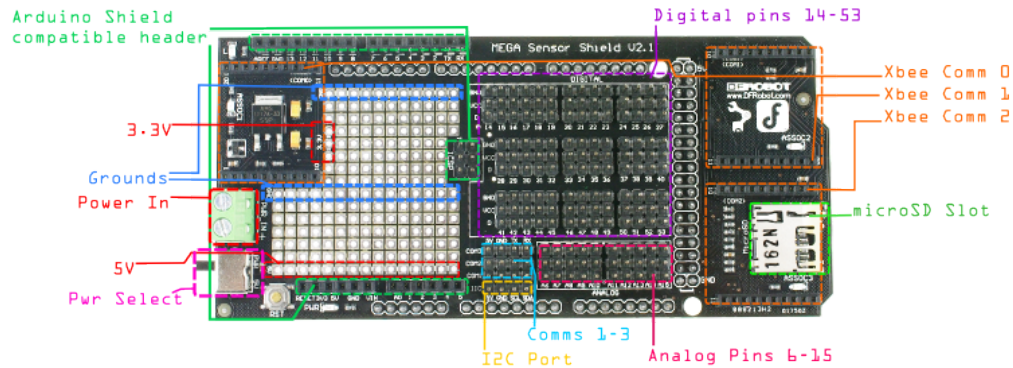


Figura 2. IO Expansion Shield V2.1.
Fuente: DFRobot (2012).

Estas extensiones de las placas electrónicas Arduino, permiten que los mismos obtengan una mayor cantidad de puertos, y extensiones las cuales ayudan a conectar distintos sensores de producción existentes en el mercado, ya sea que trabajen de forma analógica o digital.

Una de las ventajas más influyentes de los Shields, es la compatibilidad con los sensores inalámbricos XBee, la cual permite convertir al Arduino en modulo capaz de entrelazar comunicación de manera inalámbrica (a través de señales de radiofrecuencia), con cualquier dispositivo compatible con este sistema.

3. Sensores:

Los sensores convierten una magnitud física o química (generalmente no eléctrica) en una magnitud eléctrica, teniendo en cuenta magnitudes perturbadoras. Esta conversión se efectúa a menudo a través de fases intermedias no eléctricas (Zabler, 2002).

En el caso de los automóviles los sensores trabajan como “los órganos de percepción” del vehículo logrando medir parámetros como la aceleración, velocidad, temperatura, nivel de combustible entre otros. (Pallas, 2001).

Se conoce como parámetro al dato que se considera como imprescindible y orientativo para lograr evaluar o valorar una determinada situación. La medición bien conceptualizada de estos, permite planificar de manera más confiable, diferenciar con

mayor precisión y analizar las oportunidades, además de explicar los hechos acontecidos. (Bentley, 2000).

Los sensores aplicados en competencias automovilísticas como la fórmula 1 y en el caso de la presente investigación enmarcada en Fórmula y BAJA SAE, permiten a los miembros del equipo un estudio del vehículo en tiempo real, además de brindar un histórico de la evolución del automóvil en los aspectos monitoreados. A continuación se describen los tipos de sensores:

Sensor de Combustible: El sensor de combustible, está concebido como un dispositivo electrónico capaz de monitorizar y registrar datos sobre el movimiento de combustible en un vehículo, dando un registro exacto de cuando se llenó o quito combustible del tanque.

Además de contabilizar la cantidad de combustible que hay almacenado en el tanque, los datos pueden ser estudiados y determinar el consumo del motor en distintas actividades como lo son la aceleración y el frenado lo que permite a los equipos de la organización SAE una mejor planificación sobre la estrategia que tomar en la pista de carrera.

El Savant FS-3400 BH (ver *Figura 3*) es conocido en la actualidad como un dispositivo capaz de monitorear el flujo de gasolina que pasa a través de él, logrando de esta manera obtener la cantidad de combustible consumida por los vehículos, principalmente en el área de competición.



Figura 3. Sensor Savant FS-3400 BH (Flujo de Combustible).
Fuente: Micropik.

Entre las características del mismo se pueden destacar una posibilidad de monitorear de 2-30 litros por hora, una salida de tipo digital, un rango de trabajo de 2,4-26V con un amperaje máximo de 8Amperes.

Sensor de Temperatura: Por su parte los sensores de temperatura son dispositivos que transforman los cambios de temperatura en señales eléctricas que son procesados por el equipo eléctrico o electrónico.

Existen diversos tipos de sensores según la utilidad que se les dará, en el caso de los automóviles es necesario conocer la temperatura a la que trabaja el motor, y la temperatura del aceite, esto resulta esencial debido a que si el motor del automóvil se encuentra trabajando a mayor temperatura de lo debido es posible su daño total y descalificación de la competencia en el caso particular de la competencia SAE.

Entre la gran variedad de dispositivos que monitorean temperatura se puede destacar LM35 (Ver *Figura 4*), el cual permite la medición de la temperatura en un rango variable desde -55C hasta 155C.

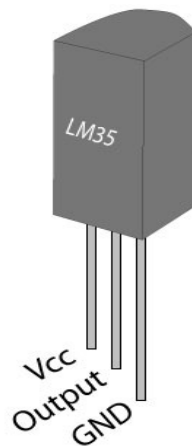


Figura 4. Sensor LM-35 (Temperatura).
Fuente: Texas Instruments (2013).

Posee la capacidad de trabajar en un rango variable de corriente directa (DC) desde los 4V a 30V, con una aceptación de hasta 60 μ A. Es ideal para monitorear entornos de reducido espacio; y a diferencia de otros sensores de temperatura posee un bajo costo.

Sensor de Velocidad: Es notable la necesidad de obtener información sobre el desplazamiento de un vehículo, y más aún cuando este se encuentra en competencias donde el rendimiento del automóvil se encuentra estrechamente ligado con la velocidad que el mismo puede conseguir.

Se denota como sensor de velocidad a cualquier dispositivo que logre procesar y convertir movimientos mecánicos en una señal eléctrica capaz de determinar el desplazamiento por tiempo de un automóvil. En la actualidad existen distintos tipos de sensores de velocidad pasando de dispositivos magnéticos en la caja de velocidad del vehículo, como monitores que miden el torque del auto.

En el caso de las competencias BAJA Y Formula SAE, resulta de importancia monitorear la velocidad a la cual se desplaza el prototipo en un lapso de tiempo, puesto que la velocidad está en completa dependencia con el funcionamiento del auto, el consumo de gasolina y el tiempo de vuelta en la competencia.

Entre los sensores de velocidad disponibles en el mercado se destaca el “Globalsat GPS module EM-506”; el cual como su nombre lo indica es módulo GPS. Entre sus características principales se puede mencionar su bajo costo, bajo consumo de energía y alta sensibilidad, ideal para un sistema de telemetría donde los recursos de energía y presupuesto son limitados.

Su alimentación es mediante corriente DC entre los 4,5V a 6,5V, con impedancia permitida de 2mA; operando a una frecuencia de 1.575GHz. Se pueden observar mayores características del módulo GPS en la TABLA 1.

TABLA 1
Características del Globalsat GPS module EM-506.

Sensitivity	Tracking : Autonomous acquisition :	-163dBm -160 dBm
Time-To-First-Fix ¹	Cold Start – Autonomous	< 35s <15s (with CGEE)
	Warm Start – Autonomous ²	< 35s < 15s(with CGEE)
	Hot Start – Autonomous ³	< 1s
Horizontal Position Accuracy ⁴	Autonomous	< 2.5m
Velocity Accuracy ⁵	Speed	< 0.01 m/s
	Heading	< 0.01 degrees
Reacquisition	0.1 second, average	
Update Rate	1 Sec / 5 Sec	
Maximum Altitude	< 18,000 meter	
Maximum Velocity	< 515 meter/ second	
Maximum Acceleration	< 4G	

Nota: Tomado de Globalsat Technology Corporation (2014).

II.3. Protocolos de transmisión inalámbrica

Al momento de diseñar un sistema de comunicación, se deben estudiar ciertas variables y factores que afectan directamente al funcionamiento y a la calidad del sistema, tanto positivo como negativo, es decir, no sólo escoger el sistema de mejor calidad en la comunicación, sino también el de mejor costo, para tener una mejor productividad en todos los aspectos. Actualmente, la cantidad de tecnologías de telecomunicaciones, son las suficientes como para adaptarse de acuerdo a las distintas necesidades, específicamente en lo que respecta con cobertura, calidad de transmisión, capacidad de transmisión, entre otras, las cuales permiten a los usuarios implementar la más conveniente según sea el caso. Dentro de la serie de protocolos para las comunicaciones de corto alcance (desde 0 metros hasta 30 kilómetros) o también conocidos como los Estándares de Redes Locales y de Áreas Metropolitanas establecidos por la IEEE (Institute of Electrical and Electronics Engineers) en su familia de Protocolos 802.x, se encuentran muchas utilizadas a diario como: Wi-Fi, Bluetooth, WiMax, Ethernet, entre otros muchos más; de estos, para efectos del

proyecto, serán excluidos varios debido a los requerimientos del mismo, y mediante una tabla de comparación se estudiará el protocolo más adecuado.

II.3.1 IEEE 802.15: Redes Inalámbricas de Área Personal (WPAN)

Es un grupo de trabajo dentro de IEEE 802 especializado en redes inalámbricas de área personal (Wireless Personal Area Networks, WPAN). Se divide en cinco subgrupos, del 1 al 5.

Los estándares que desarrolla definen redes tipo PAN o HAN, centradas en las cortas distancias. Al igual que Bluetooth o ZigBee, el grupo de estándares 802.15 permite que dispositivos portátiles como PC, PDAs, teléfonos, sensores y actuadores utilizados en domótica, entre otros, puedan comunicarse e interoperar. Debido a que Bluetooth no puede coexistir con una red inalámbrica 802.11.x, se definió este estándar para permitir la interoperabilidad de las redes inalámbricas LAN con las redes tipo PAN o HAN. Para este protocolo, existieron grupos de trabajo (taskgroup), los cuales consisten en un personal encargado de desarrollar los distintos comportamientos específicos del estándar, así como las mejoras del mismo. A continuación se describen cada uno:

1. Taskgroup 1 (WPAN/Bluetooth)

IEEE 802.15.1-2002 desarrolla un estándar basado en la especificación 1.1 de Bluetooth. Incluye nivel físico (PHY) y control de acceso al medio (MAC). Se ha publicado una versión actualizada, IEEE 802.15.1-2005.

2. Taskgroup 2 (Coexistencia)

IEEE 802.15.2-2003 estudia los posibles problemas derivados de la coexistencia de WPAN's con otros dispositivos inalámbricos que utilicen las bandas de frecuencia no reguladas, tales como redes inalámbricas de área local (WLAN).

3. Taskgroup 3 (WPAN de alta velocidad)

IEEE 802.15.3-2003 es un estándar que define los niveles PHY y MAC para WPAN's de alta velocidad (11-55 Mbit/s). En este, surgieron varias versiones con modificaciones durante los años posteriores a su creación:

Versión 3a (PHY alternativa para WPAN de alta velocidad): La IEEE 802.15.3a intentó realizar mejoras al nivel físico de Ultra-WideBand para su uso en

aplicaciones que trabajen con elementos multimedia. Su aspecto más destacable fue la consolidación de veintitrés especificaciones de PHY para UWB en dos propuestas utilizando multiplexación por división de frecuencias ortogonal multibanda (Multi-Band Orthogonal Frequency Division Multiplexing, MB-OFDM) en UWB y UWB en secuencia directa (DS-UWB).

Versión 3b (Revisión MAC): Esta trabajó en el desarrollo de mejoras para la IEEE 802.15.3, refinando la implementación e interoperabilidad de MAC. Esto incluye optimizaciones menores que preserven la compatibilidad en todo caso, además de corrección de errores y ambigüedades así como aclaraciones.

Versión 3c (PHY alternativa de onda milimétrica): Se formó en marzo de 2005 y trabaja en el desarrollo de una PHY alternativa basada en ondas milimétricas para el estándar 802.15.3-2003. Esta WPAN operara en la nueva banda no regulada que se extiende en el rango de 57-64 GHz, que además no ha sido utilizada hasta la fecha, definida por FCC 47 CFR 15.255. Permite una coexistencia muy alta con todos los sistemas de microondas en la familia 802.15. Puede alcanzar tasas de transmisión muy elevadas, de más de 2 Gbit/s, ideal para aplicaciones de acceso a Internet de banda ancha y streaming (televisión digital, cine en casa, etc.).

4. Taskgroup 4 (WPAN de baja velocidad)

La IEEE 802.15.4-2003 o también conocida como “Low Rate WPAN”, trata las necesidades de sistemas con poca transmisión de datos pero vidas útiles muy altas con alimentación limitada (pilas o baterías) y de baja complejidad. La primera revisión fue aprobada en mayo de 2003. Tras la formación del grupo 4b en marzo de 2004, este grupo pasó a estado latente. Los protocolos ZigBee se basan en la especificación producida por este grupo de trabajo.

ZigBee es una de las modalidades de este taskgroup y utiliza la banda ISM para usos industriales, científicos y médicos; en concreto, 868 MHz en Europa, 915 en Estados Unidos y 2,4 GHz en todo el mundo. Sin embargo, a la hora de diseñar dispositivos, las empresas optarán prácticamente siempre por la banda de 2,4 GHz, por ser libre en todo el mundo. El desarrollo de la tecnología se centra en la sencillez y el bajo costo más que otras redes inalámbricas semejantes de la familia WPAN,

como por ejemplo Bluetooth. El nodo ZigBee más completo requiere en teoría cerca del 10% del hardware de un nodo Bluetooth o Wi-Fi típico; esta cifra puede alcanzar al 2% para los nodos más sencillos. No obstante, el tamaño del código en sí es bastante mayor y se acerca al 50% del tamaño del de Bluetooth.

Una red ZigBee puede constar de un máximo de 65535 nodos distribuidos en subredes de 255 nodos, frente a los ocho máximos de una subred (Piconet) Bluetooth.

Frente a Bluetooth, Zigbee requiere de un menor consumo eléctrico. En términos exactos, ZigBee tiene un consumo de 30 mA transmitiendo y de 3 μ A en reposo, frente a los 40 mA transmitiendo y 0,2 mA en reposo que tiene el Bluetooth. Este menor consumo se debe a que el sistema ZigBee se queda la mayor parte del tiempo dormido, mientras que en una comunicación Bluetooth esto no se puede dar, y siempre se está transmitiendo y/o recibiendo.

La cualidad que diferencia a ZigBee del común estándar base (IEEE 802.15.4), es que tiene la capacidad de enrutar, capaz de administrar distintos tipos de topología, de las cuales se puede mencionar la más famosa, la de malla. Ésta permite que si, en un momento dado, un nodo del camino falla y se cae, pueda seguir la comunicación entre todos los demás nodos debido a que se rehacen todos los caminos. La gestión de los caminos es tarea del coordinador (uno de los tipos de dispositivos dentro de la topología junto con los enrutadores y los terminales).

Durante los años consiguientes al surgimiento de este taskgroup, fueron surgiendo correcciones y modificaciones para mejoras relacionadas con compatibilidad:

- Versión 4a (PHY alternativa): El principal interés de este grupo es permitir comunicaciones y facilidades de localización de alta precisión, alta productividad agregada y necesidades energéticas extremadamente reducidas. Tomando en cuenta aspectos como la escalabilidad en la tasas de datos, distancia de transmisión, costos y consumo.

- Versión 4b (Revisiones y mejoras): En este grupo, se realizaron mejoras y aclaraciones específicas sobre IEEE 802.15.4-2003. Entre estos objetivos se encuentran la resolución de ambigüedades y reducción de complejidad innecesaria, el

incremento de la flexibilidad en el uso de claves de seguridad, las consideraciones para el uso de nuevos rangos de frecuencias disponibles y otros aspectos. Fue aprobada en junio de 2006 y se publicó en septiembre del mismo año como IEEE 802.15.4-2006.

- Versión 4c, Modificación de la capa física o PHY (PHYsical) para China: La IEEE 802.15.4c fue aprobada en 2008 y publicada en Enero de 2009. Esta modificación de las capas físicas añade nuevas especificaciones en el espectro de radiofrecuencia, para adaptarse a los cambios de normativas que hay en China que han abierto las bandas de 314-316 MHz, 430-434 MHz, y 779-787 MHz para el uso de PAN inalámbricas dentro de China.

- Versión 4d, Modificación de la capa física o PHY y de control de acceso al medio o MAC para Japón: El grupo de trabajo de IEEE 802.15.4d fue constituido para definir una modificación en el estándar existente 802.15.4 de 2006. La modificación contempla cambios tanto en la capa física como en la de control de acceso al medio, que son necesarios para soportar la asignación de una nueva frecuencia (950 MHz -956 MHz) en Japón, mientras coexisten con otros sistemas de protocolos en la frecuencia de banda.

- Versión 4e, Modificación de la capa de control de acceso al medio o MAC para aplicaciones industriales: Fue constituido para definir una modificación en el estándar existente 802.15.4 de 2006. La intención de esta modificación fue mejorar y agregar nuevas funcionalidades a la capa MAC, que básicamente consisten en mejorar el apoyo a los mercados industriales, permitir la compatibilidad con las modificaciones que se propusieron en el WPAN de China y las mejoras para añadir saltos de canal. Este grupo fue aprobado en el 2011.

- Versión 4f, Modificación en la capa física o PHY y la identificación por radiofrecuencia o RFID: Este grupo se estableció para definir nuevas capas físicas inalámbricas y mejoras con respecto al estándar de la capa MAC 802.15.4 del 2006 necesarias en las nuevas capas físicas para la identificación por frecuencia o RFID bidireccional.

- Versión 4g, PHY Modificación para Herramientas de Red Inteligentes o SUN: Finalmente, siendo el grupo más reciente, fue constituido para crear una nueva capa física que modifique 802.15.4 para proporcionar un estándar que facilite a gran escala aplicaciones de control de procesos como la utilidad de redes inteligentes capaces de soportar geográficamente diversas redes con una mínima infraestructura.

Aplicaciones de las Redes de Área Personal

El objetivo principal de las PANs (Personal Area Networks) es poder compartir de forma fácil la información entre todos los dispositivos inalámbricos. De allí que las PANs tengan distintas aplicaciones. A saber:

Sincronización de Dispositivos Personales: Esto permite sincronizar de forma automática los datos entre los dispositivos móviles inalámbricos en teléfonos móviles, computadoras portátiles, entre otros.

Conectividad Ad-Hoc: Permite la transferencia de archivos y otras informaciones hacia otros usuarios de dispositivos PAN.

Computadoras y periféricos sin cables: Dispositivos inalámbricos de uso diario como el ratón, teclado, impresoras, escáneres, entre otros.

Acceso inalámbrico localizado a una LAN o a internet: El mismo puede lograrse a través de un Access Point compatible con PANs. Y en muchos casos, permite navegar en internet y utilizar el correo electrónico.

Auriculares de Teléfonos Móviles.

Automatización del Hogar: Transferencia de comandos a dispositivos del hogar con compatibilidad PAN.

Telemedicina: Utilizar dispositivos médicos inalámbricos para beneficios de salud del usuario o enfermo.

Comparación del Modelo OSI

Debido a que las redes actuales funcionales de comunicación tienen como modelo de referencia al OSI, es necesario ver la comparación de este con el del estándar IEEE 802.15.4 (Ver TABLA 2)

TABLA 2
Comparación entre el modelo OSI y el estándar IEEE 802.15.4.

	Modelo ISO- OSI	Modelo IEEE 802.15
7	Capa de Aplicación	Capas Superiores
6	Capa de Presentación	
5	Capa de Sesión	
4	Capa de Transporte	
3	Capa de Red	
2	Capa de Enlace de Datos (DLL)	Control de Enlace Lógico (LLC)
		Control de Acceso al Medio (MAC)
1	Capa Física	Capa Física (PHY)

Nota: Tomado de Barriga, W y Zúñiga, F. (2006).

La norma IEEE 802 divide a la capa de Enlace de datos (DLL: Data Link Layer) en dos subcapas: la subcapa de acceso al medio (MAC: Media Access Control), y la de control de enlace lógico (LLC: Logical Link Control). La LLC es común en todos los estándares 802, mientras que la subcapa MAC depende del hardware y varía respecto a la implementación física de esta capa

II.4. Equipos XBee/RF

Son módulos de electrónica que permiten la comunicación inalámbrica entre uno o más dispositivos, los cuales fueron diseñados para cumplir con los estándares IEEE 802.15.4 y apoyar la necesidad de las redes de sensores inalámbricas de bajo costo y bajo consumo de energía. Los módulos requieren un mínimo de energía y son fiables en la entrega de datos entre dispositivos. (Hebel, M y Bricker, G., 2012).

En este sentido, suelen ser utilizados en proyectos como domótica, telemetría y control de sensores ya que cuentan con una fiabilidad muy aceptable. XBee utiliza un

protocolo muy implementado para las comunicaciones de datos que proveen características necesarias para una red robusta en una Red Inalámbrica de Sensores (WSN: Wireless Sensor Network); dichas características se refieren a, direccionamiento, confirmación de recepción, respuestas para asegurarse que la información entregada es válida. También XBee permite funcionalidades que van más allá de la comunicación de datos, tales como el uso en monitorización y control remoto de dispositivos. (Faludi, 2010).

II.4.1 Redes Multinodo

Muchos de los módulos inalámbricos para microcontroladores solamente envían y reciben data en la frecuencia proporcionada, esto depende del usuario final y de la aplicación a implementar, pero se realiza de esta forma para combatir los problemas de tipo de acceso al medio, verificación de entrega de datos, chequeo de error, y específicamente en redes multinodos, donde el nodo acepta y usa la data. Los dispositivos que usan un protocolo de red pueden facilitar el trabajo de un programador mediante el manejo de los parámetros antes mencionados.

II.4.2 Acceso al Medio

Se refiere a la capacidad que poseen los equipos para acceder al canal de transmisión, sabiendo que en este existen múltiples dispositivos que en determinado momento, pueden requerir el medio para enviar o recibir data simultáneamente. Los diferentes protocolos de red, manejan el acceso al medio de diferentes maneras, pero similar al comportamiento en las conversaciones de los seres humanos. Ethernet (IEEE 802.3) sobre un cable compartido, utiliza “CSMA/CD”, en donde todos los nodos (dispositivos de la red) tienen igual derecho para enviar data en la red (Multiple Access – MA); primero, un nodo escucha para asegurarse de que el medio (Cable CAT 5) está en uso. Si este lo está, el dispositivo debe esperar para la apertura de su transmisión; una vez disponible, intentará transmitir su data (Carrier Sense – CS). Mientras se envía la data, el nodo también monitorea que está en el medio y si existe otro nodo transmitiendo al mismo tiempo, la colisión de la información es detectada y ambos detendrán su transmisión, esperarán un tiempo predeterminado, e intentarán nuevamente la transmisión (Collision Detection – CD).

El protocolo de Wi-fi (IEEE 802.11), usa CSMA/CA para el acceso, el cual se diferencia principalmente del CSMA/CD, por no permitir a los nodos transmitir y escuchar al mismo tiempo y también, por utilizar medidas adicionales para evitar que las colisiones ocurran (Collision Avoidance – CA). Adicionalmente, la evitación de colisión ayuda a permitir la fluidez en situaciones donde dos transmisores inalámbricos están dentro del rango del otro, pero un tercer transmisor está fuera del rango del primer transmisor. El transmisor del medio, escucha transmisiones de ambos vecinos, pero los transmisores de los extremos, no se escuchan entre ellos. Existe una “invisibilidad” entre cada uno, este fenómeno es llamado en el ámbito de redes, el “Problema del nodo invisible” (Hidden Node Problem).

El protocolo de Token Ring (IEEE 802.5), asegura el acceso y previene las colisiones a través del paso de un “testigo” entre nodo y nodo en una topología de anillo, permitiéndole a cada uno, tener acceso exclusivo al medio por turnos. El protocolo USB utiliza una especie de votación, con la finalidad de saber si algún dispositivo USB requiere acceso para comunicarse, y de ser el caso, asignarle una ranura de tiempo (time slot) para comunicarse con el cliente.

II.4.3 Direccionamiento

A través de la asignación de identidad a cada uno de los dispositivos XBee en determinada red, es posible poder transmitir un flujo de información a determinado dispositivo (Punto a punto), o a un grupo entero de dispositivos (multipunto). En muchos de los casos, todos los nodos reciben el mensaje de “datos enviados”, pero si la data no contiene su dirección asignada programada previamente, únicamente el dispositivo con la dirección de destino correcta, será capaz de utilizar la información.

La dirección de destino del mensaje recibido es importante también, debido a que al momento de recibir el mensaje, el dispositivo necesita saber cual de todos envió la información. Adicionalmente, muchas de las veces, el emisor solicita confirmación de mensaje recibido (Acknowledgment), para lo cual se debe conocer la dirección del mismo.

II.4.4 Comprobación de Error y Acuses de recibo

Los dispositivos XBee, durante la transmisión, son capaces de agregar información adicional a los paquetes originales, con la finalidad de hacer pasar los datos a través de un proceso de verificación de errores al momento de recibirlos e incluso, el dispositivo receptor, en ciertos casos si la conexión lo requiere, envía un acuse de recibo al emisor.

II.4.5 Encapsulación

El mensaje (data a transmitir) es empaquetada con otra información necesaria por el protocolo para el control de los procesos, la cual está básicamente conformada por direcciones de origen y destino, valores de comprobaciones de error y otra información pertinente, requerida por el protocolo, con la finalidad de asegurar la correcta entrega del mensaje al otro extremo de la comunicación o nodo (ver *Figura 5.*)

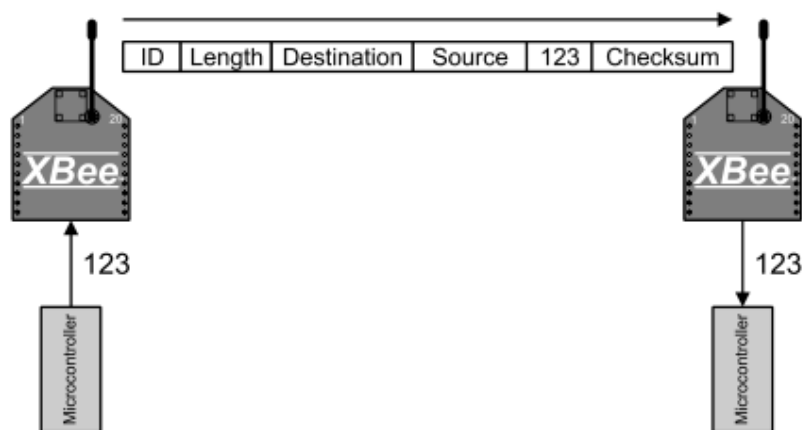


Figura 5. Encapsulación XBee.
Fuente: Hebel, M y Bricker, G (2012).

II.4.6 Aplicación

Para efectos del empaquetado de los datos, la transmisión punto a punto, comprobación de errores y acuses de recibo, son manejados por el protocolo de la Capa de Enlace. Estos, son los encargados de asegurarse que el nodo reciba la data libre de errores; depende de la aplicación de los datos, la cual definirá cual es el mensaje y cuáles serán las acciones a realizar, una vez recibida la secuencia de

paquetes. La aplicación está íntimamente enlazada con el software, ya que este, el cual fue previamente diseñado por un programador, para manejar los datos e interpretarlos de acuerdo a las necesidades y tareas del sistema.

II.4.7 Beneficios de XBee

Los protocolos modernos como Wi-Fi (IEEE 802.11) y Bluetooth (IEEE 802.15.1), se comportan como sistemas que aseguran que los datos lleguen al destino correcto sin errores. El protocolo reduce en gran medida el trabajo del programador, en cuanto a entrega de información se refiere; algunos de los aspectos claves de los protocolos para llevar a cabo esta tarea son:

- Acceso al medio: Se refiere a distribuir correctamente el canal, para que no existan colisiones y errores en la transmisión cuando dos nodos de la red transmitan al mismo tiempo.
- Direccionamiento: Garantiza que solo el nodo correcto sea el que utilice la información recibida, permitiendo que los datos se envíen de un punto a otro, o incluso desde un nodo a todos los nodos de la red a través de un mensaje de difusión (broadcast).
- Detección de Error: Verifica los datos recibidos en el nodo correcto.
- Informes de recibo y Reintentos: Los informes de recibo implican los distintos mecanismos para reportar que la información se recibió en el nodo destino y los reintentos son aquellas retransmisiones luego de no haber recibido el informe de entrega (Acknowledgement) por parte del nodo destino, esto con la finalidad de asegurar la transmisión de la información de extremo a extremo.

El XBee utiliza el protocolo IEEE 802.15.4, el cual implementa todas las características antes mencionadas. Este protocolo es conocido como el que establece las Redes Inalámbricas de Área Personal de baja tasa de transmisión (LR-WPAN). Provee una tasa de transmisión de datos de más de 250kbps, entre los nodos, en una red CSMA/CA. Este protocolo no está diseñado para grandes segmentos de datos, como imágenes o videos, sino que provee la suficiente rapidez entre los nodos para el uso de monitorización y control de sistemas comúnmente relacionados con una Red Inalámbrica de Sensores (WSN).

En comparación con Bluetooth (IEEE 802.15.1), la LR-WPAN está diseñada como un protocolo mucho más simple, con bajas tasas de transmisión de datos (250kbps comparados con 1Mbps). Bluetooth fue diseñado como un reemplazo de los cables periféricos y es utilizado en comunicaciones entre dispositivos móviles (teléfonos, tablets, laptops), requiriendo seguridad de acceso y altas tasas de transmisión.

El XBee, utilizando el protocolo IEEE 802.15.4, incorpora lo siguiente para las comunicaciones y control en una WSN (Wireless Sensor Network):

- Clear Channel Assessment (CCA): Antes de transmitir, un nodo XBee escucha para verificar si la frecuencia seleccionada está ocupada.

- Direccionamiento: El XBee tiene dos opciones en este aspecto; una es la dirección MAC (serial único e irremplazable de 64 bits), y una dirección asignable de 16 bits, lo cual permitiría soportar 64000 direcciones distintas en una misma red.

- Verificación de errores y Informes de entrega (Acknowledgements): Los dispositivos XBee utilizan una suma de verificación (Checksum) enviada junto con la información, la cual a través de un algoritmo, obtiene una especie de resumen numérico. Si el checksum en el nodo destino es el mismo, los datos recibidos no contienen errores. Por otro lado, los informes de entrega, son enviados al nodo transmisor para indicarle la correcta recepción de datos; adicionalmente, se ejecutarán 3 reintentos de transmisión, en caso de no recibir los informes de entrega.

II.4.8 Modos de comunicación

Los dispositivos en general, poseen distintos modos de funcionamiento, ya sea para el ahorro de energía, requerimientos técnicos, entre otros, los cuales le facilitan al usuario, el uso adecuado para cada instancia de determinado sistema de comunicaciones. En el caso de los XBee, estos soportan dos modos: Transparente (AT) e Interface de aplicación para programación (API). Ambos tienen ventajas dependiendo del escenario.

- Modo AT: Conocido como modo transparente, en donde solo se envían los datos del mensaje. El protocolo de enlace entre los dos nodos es transparente al usuario final y los datos son enviados en "RAW", es decir, es como si entre dos

puertos seriales, en lugar de haber un cable, hubiera un medio radioeléctrico. Este modo permite transmisiones simples y recepciones de datos serial. Los comandos AT son utilizados para configurar el XBee remotamente, y permiten hacer modificaciones en la configuración del mismo.

A pesar de que la transmisión y recepción son datos en bruto, el mensaje por sí mismo, pasa a través de los nodos encapsulado con la información necesaria tal como direccionamiento y verificación de errores.

- Modo API: por sus siglas Application Programming Interface, es en el que el programador empaqueta los datos con la información requerida (dirección de destino, tipo de paquete, checksum). También, el nodo receptor, acepta los datos con la información requerida (dirección de origen, tipo de paquete, intensidad de la señal, y checksum. Las ventajas de este modo, básicamente se centran en que el usuario puede identificar al nodo de origen y destino del paquete, fragmentar los paquetes e identificar los fragmentos que llegan y comprobar que llegan en orden. Mientras la programación sea más extensa, este modo permitirá incrementar la flexibilidad y en algunos casos la fiabilidad del sistema.

Vale destacar, que al momento de comunicar dos o más nodos, no es necesario que estén en el mismo modo. Los datos pueden ser enviados en modo API y recibidos en modo AT o vice-versa. Los modos definen los enlaces de comunicaciones entre la PC o el controlador y el modem XBee, y no entre los módulos XBee. Los datos entre los módulos o nodos, siempre son enviados el mencionado protocolo de LR-WPAN.

A continuación se muestra la trama RF que se envía en modo API, con las cabeceras desglosadas (ver *Figura 6*).

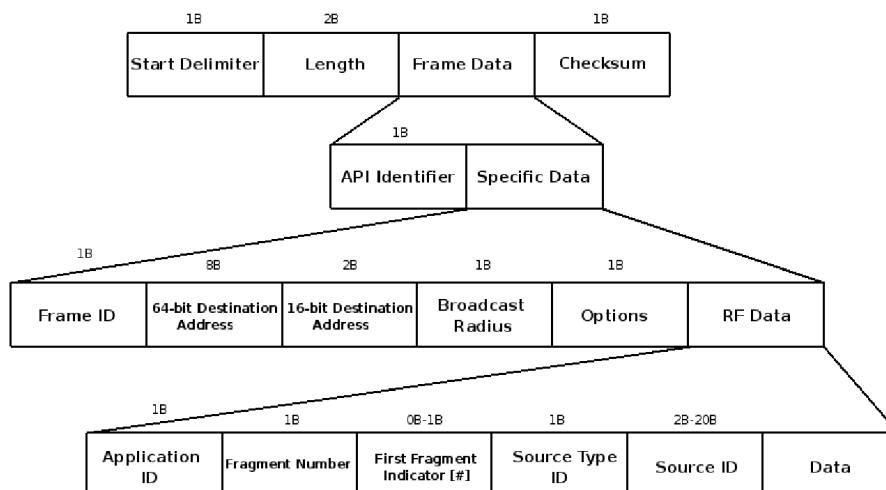


Figura 6. Trama en modo API.

Fuente: Hebel, M y Bricker, G (2012).

En la cabecera API (Application Header) se observan los siguientes campos:

- Application ID. Es un ID de aplicación que sirve para identificar un determinado paquete entre varios en el receptor.
- Fragmentnumber. El número de fragmento especifica la posición del fragmento en el paquete. El primer fragmento muestra el número total de fragmentos del paquete.
- FirstFragmentIndicator [#]. Es un campo opcional que se incluye en el primer fragmento para indicar que es el primer fragmento.
- Sourcetype ID: Especifica el tipo de identificador de origen elegido. Las posibilidades son: dirección MAC (64 bits), Identificador de red NA (16 bits) o Identificador de nodoNI (cadena de caracteres de 20 bytes máximo).
- Source ID. Especifica el identificador del nodo origen del paquete. Depende del campo sourcetype ID elegido.
- Data: Guarda los datos enviados en cada fragmento o paquete entero (en caso de no existir fragmentación) (puede ser de 8 a 92 Bytes).

Digi como fabricante, recomienda utilizar el modo API en caso de tener que enviar datos RF a múltiples destinos, o cuando un nodo recibe datos de diferentes orígenes, lo cual no es el caso.

Además, se ha optado por usar el modo API porque realiza tareas que facilitan la implementación de un nivel de red, como es el caso del direccionamiento o la fragmentación de paquetes. La única ventaja del modo transparente es que es más sencillo de implementar y se transmiten menos datos.

II.4.9 Sistema energético

Cada módulo tiene hasta 4 modos de funcionamiento:

- ON: modo normal de funcionamiento.
- Sleep: En este modo se detienen algunas funcionalidades del módulo y se pasa a un uso asíncrono, normalmente dirigido por eventos. En cada módulo funciona de una forma distinta y es específico de cada uno (programado por el fabricante).
- Hibernate: En este modo se detienen todas las funcionalidades del módulo y se pasa a un uso asíncrono, normalmente dirigido por eventos. En cada módulo funciona de una forma distinta y es específico de cada uno (programado por el fabricante).
- OFF: Mediante el uso de switches digitales controlados por el microcontrolador se apaga por completo el módulo.

II.4.10 Tipos de Módulos XBee

Existen varias versiones de los módulos, pero todos poseen los mismos pines de conexión. Las diferencias entre las versiones XBee, son comparables en cuanto a potencia de salida, tipo de antena, frecuencia de operación y capacidad sobre la red. El XBee tiene 20 pines distribuidos en dos filas paralelas, espaciados por 2 mm de distancia. (Ver *Figura 7* y TABLA 3).

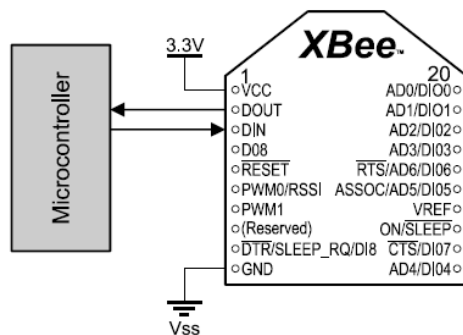


Figura 7. Pines de Módulos XBee
Fuente: Hebel, M y Bricker, G (2012).

DISEÑO DE UN SISTEMA DE TELEMETRÍA PARA LA MONITORIZACIÓN DE
PARÁMETROS EN LOS VEHÍCULOS DE COMPETENCIAS SAE.

TABLA 3
Características técnicas de los XBee.

Pin	Nombre	Tipo	Función
1	VCC	P	2.8 V to 3.4 V
2	DOUT	O	Salida de Datos Serial del XBee (datos recibidos)
3	DIN	I	Entrada de Datos Serial del XBee (datos a transmitir)
4	DO8	O	Salida de datos Digital 8
5	RESET	I	Reiniciar Modulo (low)
6	PWM0/	O	Salida de Modulación por Ancho de Pulsos
	RSSI	O	Indicador de Intensidad de Señal Recibida como señal PWM
7	PWM1	O	Salida de Modulación por Ancho de Pulsos
8	(Reservado)		
9	DTR	I	Data Terminal Ready: handshaking para actualizaciones de software (low)
	SLEEP_RQ	I	Solicitud de Modo Sleep: Un alto configura el XBee en modo sleep
	DI8	I	Salida Digital 8
10	GND	G	Tierra (Vss)
11	AD4	A	Entrada Analógico a Digital 4
	DIO4	IO	Entrada/Salida Digital 4
12	CTS	O	Libre para Enviar Salida handshaking al controlador (low)
	DIO7	IO	Entrada/Salida Digital 7
13	ON/SLEEP	O	Salida Digital. High = Modo Despierto, Low = Modo Sleep
14	VREF	A	Analógico a Voltaje de Referencia Digital
15	ASSOC	O	Indicación asociada cuando se une a una red
	AD5	A	Entrada Analógica a Digital 5
	DIO5	IO	Entrada/Salida Digital 5
16	RTS	I	Listo para enviar entrada del Handshaking (Low)
	AD6	A	Entrada Analógica a Digital 6
	IO6	IO	Entrada/Salida Digital 6
17-20	AD3-AD0	A	Entrada Analógica a Digital 3 a 0
	DIO3-DIO0	IO	Entrada/Salida Digital 3 a 0

Nota: Tomado de Hebel, M y Bricker, G (2012). Tipo de Pines: P = Energía, G = Tierra, I = Entrada, O = Salida, A = Entrada Analógica

1. XBee, XBee Pro, Módulos 802.15.4

El estilo 802.15.4 del XBee (También llamado Serie 1) permite las redes punto a punto, tal como se muestra en la *Figura 8*, y también redes punto a multipunto (un nodo a todos los nodos). Los XBee utilizan el protocolo de enlace de datos IEEE 802.15.4 para mover la data directamente entre dos o más dispositivos. Todos los nodos en la red usan la misma versión de software aunque las configuraciones entre ellos puede variar.

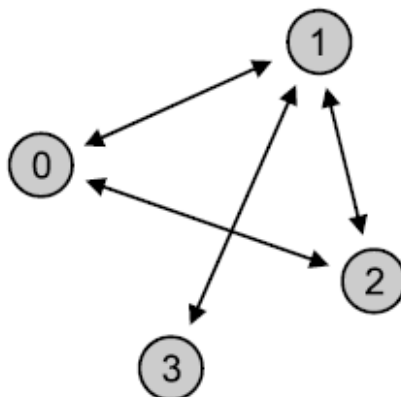


Figura 8. Diagrama Transmisión de nodos punto a punto
Fuente: Hebel, M y Bricker, G (2012).

El XBee y el XBee-Pro son muy parecidos en operación pero funcionan con grandes diferencias en tamaño y potencia. A pesar de que la estructura de los pines sea la misma, la superficie del XBee-Pro es un poco más larga. Las especificaciones de potencia y distancias, junto con otros detalles, se comparan en la TABLA 4.

TABLA 4
Comparación entre XBee y XBee-Pro.

Especificación	XBee	XBee-Pro
Voltaje de alimentación	2.8 VDC – 3.4 VDC	2.8 VDC – 3.4 VDC
Potencia RF	0 dBm, 1 mW	18 dBm, 60mW
Distancia en Espacios Abiertos - Outdoor (LOS)	90 m	1.6 km
Distancia en Espacios Cerrados - Indoor	30 m	90 m
Consumo de Corriente, Recepción	45 mA	50 mA
Consumo de Corriente, Transmisión	50 mA	215 mA
Consumo de Corriente, modo Sleep	< 10 μ A	<10 μ A
Transferencia de Datos	250 kbps	250 kbps
Frecuencia de operación, Canales	2.4 GHz, 16 Canales	2.4 GHz, 12 Canales
Sensibilidad del Receptor	-92 dBm	-100 dBm

Nota: Tomado de Hebel, M y Bricker, G (2012).

2. XBee-Pro 900 y 868

Los moduladores de 900 MHz (EEUU) y de 868 MHz operan a bajas frecuencias con una baja transferencia de datos (156 kbps) pero con mucho mayor alcance (hasta 10 km). Pero al momento de seleccionar, se debe tomar en cuenta las zonas en que puede ser utilizados de acuerdo a las normativas de cada país sobre el uso de las bandas del espacio radioeléctrico.

3. Los Módulos XBee ZB y XBee ZB-Pro

Los módulos XBee ZB extienden el alcance de la red mediante enrutamientos. Estos se auto establecen y forman redes para transferir datos por toda la red. Mientras

estos moduladores utilizan el protocolo IEEE 802.15.4 para la comunicación punto a punto, otros protocolos en capas más altas son los encargados de enrutar por la red.

Hay dos tipos de protocolos de redes malladas: ZigBee y DigiMesh. Los dispositivos pueden ser programados tanto en uno como en otro en los distintos modos AT o API. Dependiendo de la función del dispositivo, se le debe cargar la versión de software más adecuada.

En una red mallada, como una ZigBee, los dispositivos tienen diferentes niveles:

Coordinador: Establece y mantiene la red asignando direcciones para unir los dispositivos y asistir a la creación del enrutamiento.

Enrutador: Transfiere los datos entre nodos que no se pueden comunicar directamente debido a las distancias.

Terminal: Es el nodo que recolecta los datos, controla los dispositivos de la red y es el que comúnmente está conectado a los controladores, sensores y otros dispositivos para la interacción de la red. (Ver *Figura 9*).

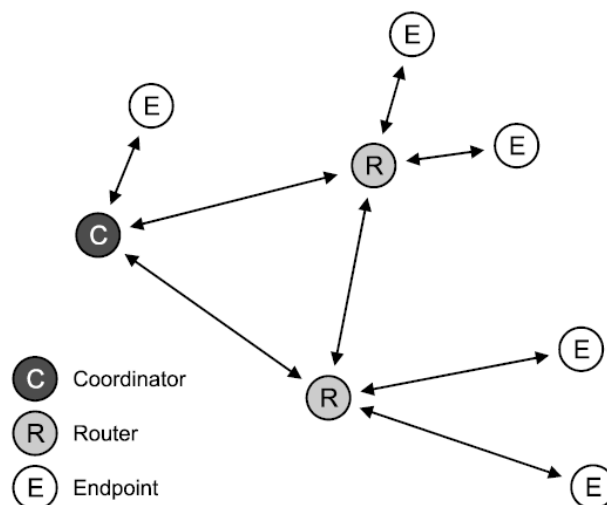


Figura 9. Enrutamiento de los nodos XBee

Fuente: Hebel, M y Bricker, G (2012).

4. Consideraciones del Alcance de Transmisiones RF

Mientras las especificaciones de distancias tales como 1.6 km suenan atractivas, existen muchos factores que pueden afectar este alcance, tales como la absorción, la

reflexión y la dispersión de ondas, los problemas de la línea de visión, tipo de antena y frecuencia. La sensibilidad del receptor permite recibir y usar señales por debajo de -100 dBm, lo cual significa 0.1 picowatts de potencia. La señal RF utiliza la Secuencia Directa del Espectro Disperso (Direct-Sequence Spread Spectrum), esparciendo la señal a lo largo del espectro de frecuencia, y por tanto, mejorando la relación señal a ruido (SNR):

El XBee tiene la capacidad de reportar el nivel RSSI (Receiver Signal Strength Indication) el cual permite monitorizar la intensidad de la señal recibida, y lo puede hacer de varias maneras:

- La salida PWM0/RSSI produce un ancho de pulso, proporcionando el indicador de nivel RSSI.
- En el modo comando, la unidad puede obtener el valor mediante el comando ATDB.
- Cuando se recibe en modo API, un byte en el paquete contiene el nivel RSSI.

Este parámetro resulta de gran utilidad al momento de recibir datos, a que si el nivel RSSI indicado en el pin, no es el mínimo requerido, el dispositivo XBee puede descartar uno o más paquetes de la secuencia para evitar los errores en la información.

5. Absorción y Reflexión

Las ondas de radiofrecuencias pueden ser absorbidas y reflejadas causando la pérdida de potencia a través de los objetos y por las distancias. El metal es un gran reflector de energía de RF, pero la mayoría de las superficies puede causar una reflexión de ondas, lo cual implica la pérdida de intensidad de la señal y en muchos casos, crear interferencia sobre otras ondas. La absorción puede ser un factor muy importante, ya que a veces puede reducir la potencia de la señal, pero en otras casos puede ser útil. Un microondas por ejemplo, calienta debido a que el agua absorbe la energía y hace la transferencia de la energía RF para calentar el alimento. De hecho, los microondas operan a 2,4 GHz, lo cual es muestra cuán susceptible es la absorción de energía RF es a esta frecuencia.

Mientras la señal puede que no sea afectada por una lluvia, fuertes tormentas pueden afectar la señal cuando está sea densa. En espacios cerrados, las paredes y otros objetos pueden causar la absorción y reflexión, limitando la distancia de transmisión para atravesar unas pocas paredes.

6. Problemas de la Línea de Visión

También conocido como Line-of-Sight (LOS), es un factor esencial para la fiabilidad de la señal. Si no hay obstáculos como árboles, edificios o paredes que absorban o reflejen la señal, la potencia en el receptor puede ser muy alta. Un aspecto más complicado es la reflexión y la interferencia deconstructiva de las ondas reflejadas debido a los objetos cercanos o incluso el suelo. Aún con línea de visión, las reflexiones del suelo u otras superficies, pueden causar que las ondas se reflejen, y a su vez, se crucen con otras ondas en el receptor, causando errores en la recepción. Mientras que el módulo XBee-Pro puede mandar transmisiones de 60 mW, solo un poco de esa potencia es dirigida al receptor. Las reflexiones de algunas de las otras ondas, llegarán más lentamente y desfasada al receptor, interfiriendo con la calidad de la señal, tal como se muestra en la *Figura 10*.

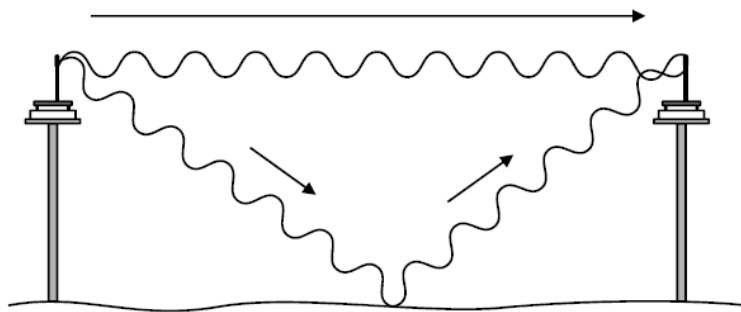


Figura 10. Reflexiones de las ondas.
Fuente: Hebel, M y Bricker, G (2012).

Para lograr la mejor transmisión, se necesita de un área libre entre el transmisor y el receptor, y esta puede ser representada gráficamente como un óvalo, también llamada Zona de Fresnel. (Ver *Figura 11*).

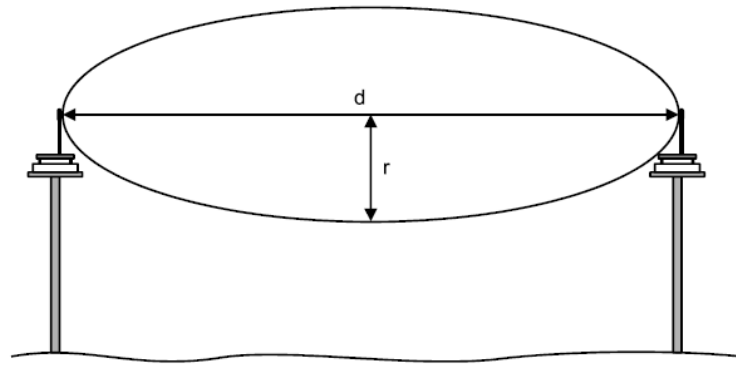


Figura 11. Zona de Fresnel.

Fuente: Hebel, M y Bricker, G (2012).

El radio de la zona puede ser calculado utilizando la siguiente ecuación:

$$r_m = 17.32 \sqrt{\frac{d_{Km}}{4f_{Ghz}}}$$

7. Frecuencia de Operación

Tal como se mostró en la tabla de comparación de especificaciones, los XBee tienen múltiples canales de frecuencia en los cuales pueden operar. Esto permite múltiples redes en la misma área para operar en una única frecuencia. También, le brinda al usuario la posibilidad de seleccionar la frecuencia que ocasione menor ruido que en otros canales, evitando interferencias como podría ocurrir con Redes Wifi, microondas o teléfonos inalámbricos sobre la banda de 2,4 GHz. A través de los comandos AT, se puede seleccionar el canal de operación.

8. Tipos de Antenas

En el ámbito de los sistemas de comunicaciones inalámbricas, la antena representa un componente crítico cuya implementación influye significativamente en el desempeño global del sistema. Esto se debe a que la antena es el elemento que convierte las señales eléctricas conducidas por las pistas de la tarjeta de circuito impreso (PCB) a señales que se propagan a través del espacio libre; por lo tanto actúa como una impedancia y un dispositivo de conversión.

Los XBee vienen en varios tipos de antenas, proporcionando distintos valores de ganancia. Los dos más populares son el de antena de chip y el de antena de látigo (cableada). Esta última tiene un 20% de mejor alcance.

II.5. Software

II.5.1 Java

Nacido en 1991, Java es un lenguaje de programación orientado a objetos. Fue creado gracias a la necesidad de Billy Joy, ingeniero de Sun Microsystems y su equipo que se encargaban de un proyecto “Green”, el cual consistía en desarrollar aplicaciones destinadas a móviles y otros periféricos.

Convencidos de las grandes ventajas que tiene la programación orientada a objetos optaron por desarrollar en C++, ya que conocían las ventajas del mismo. Luego de un tiempo trabajando en la aplicación se dieron cuenta de las fallas y dificultades que tenía este lenguaje en cuanto a la compatibilidad con distintas arquitecturas físicas (procesadores, tamaño de memoria, entre otros), así como también el nivel de adaptación de la interfaz física. (Groussard, 2012).

Observando las dificultades de programación en otros lenguajes como C++ optaron por el desarrollo de una nueva plataforma. James Gosling (conocido como el padre de Java) y Patrick Naughton empleados de SUN, fueron los encargados de la creación de este nuevo lenguaje.

Basados en las interesantes propuestas creadas por otros ambientes de desarrollo como C/C++, Mesa, Ada, entre otros, se logró crear Java una plataforma segura y fácilmente distribuida en numerosos periféricos y sistemas transportables interconectados.

Fue bautizado como Java, palabra de argot que significa café, debido a todas las tazas de café que se tomaron los desarrolladores para la creación de este lenguaje.

II.5.2 X-CTU

Es un software anunciado por “MaxStream” para configurar y probar módulos de radio MaxStream. (Digi International Inc., 2008). Resulta ser una solución para la configuración de los módulos MaxStream entre los cuales se destacan los Xbee y

XBee PRO. Este programa brinda la facilidad de realizar pruebas, observar los datos configurados en los dispositivos, así como también contar con una ventana terminal que permite el envío y recepción de datos, así como también se puede observar la comunicación efectuada entre los equipos RF; es de fácil instalación y resulta ser compatible con diversos sistemas operativos.

II.6. Sistemas de Telemetría

La telemetría es el proceso altamente automatizado de comunicaciones mediante el cual se efectúa la medición y otros datos recogidos en los puntos remotos o inaccesibles y se transmiten al equipo receptor para la monitorización. La palabra se deriva de las raíces griegas: tele (remotos) y metrón (medida). Los sistemas que necesitan instrucciones y datos externos para operar requieren la contrapartida de telemetría. (Carden, Russell y Robert, 2002).

El objetivo principal de un sistema de telemetría es recoger datos de un lugar remoto o inalcanzable y transmitirlos a un sitio donde puedan ser correctamente estudiados y analizados.

Al hablar sobre telemetría, se puede imaginar un canal inalámbrico de información, aunque no siempre es así; existen equipos de monitorización donde la información viaja a través de cableado.

La utilización de este sistema es muy común en las competencias automovilísticas. El primer intento de introducir la telemetría en la Fórmula 1 fue en 1980, cuando Karl Kempf diseñó un sistema que consistía en un mini-ordenador y un conjunto de sensores de captación de datos para controlar la suspensión electrónica. La telemetría permite que los datos registrados por multitud de sensores repartidos por todo el monoplaza y su mecánica sean conocidos en el momento por los ingenieros del equipo. Esto les puede permitir predecir el límite de utilización de un motor o el grado de seguridad activa de una determinada pieza. (Gómez de la Calle, 2010)

En la actualidad es muy diversificado el uso de esta tecnología, además de los monoplazas de carrera se implementa en aviones, barcos, misiles de guerras entre otros.

Para el caso de la presente investigación, el diseño de un sistema eficiente de telemetría requiere conocer los distintos tipos de sensores disponibles. De igual forma, conocer algunos aspectos de la competencia *SAE*, así como los automóviles que compiten en la misma.

II.6.1 Elementos de un sistema de telemetría

Según Gómez de la Calle, (2010) para la creación de un sistema de telemetría es necesario contar con los siguientes elementos:

1. Transductor: Un transductor es un equipo encargado de transformar un fenómeno físico cambiante a una señal eléctrica proporcional. Casi todos los cambios físicos a medir disponen de equipos (transductores) que pueden convertir éstas en señales eléctricas.
2. Unidad terminal: Es un dispositivo que modifica los datos medidos por el transductor de manera que puedan ser transmitidos como señal codificada utilizando algún tipo de canal de transmisión.
3. Medios de transmisión: Con relación a los canales o medios de transmisión los más comúnmente utilizados para la medición remota son:
 - Par Trenzado: Se trata de dos hilos de cobre entrelazados (por ejemplo cable telefónico).
 - Cable Coaxial: Se trata de dos conductores, donde uno es el eje central y el otro cubre al aislante del primero en forma de cubierta cilíndrica.
 - Radio: Se refiere a la transmisión de información mediante ondas electromagnéticas. Son los sistemas de radio comunes dedicados a un servicio específico.
 - Fibra Óptica: Se basa en un medio cristalino que permite la propagación de la luz, la cual no se dispersa sino que se mantiene dentro de la fibra por las características ópticas especiales de la misma con mínimas pérdidas de potencia.

Por lo general se envía más de una señal al mismo tiempo por el canal de transmisión. La codificación por modulación de impulsos, por la cual las ondas se transforman en una señal en código binario, se ha desarrollado en las últimas décadas gracias a los avances concurrenciosos en el campo de la computación digital y en la microelectrónica.

4. Receptor: es un dispositivo capaz de decodificar la señal recibida de la unidad remota y de mostrarla en algún formato adecuado para su análisis y almacenamiento.

II.7. Organización SAE internacional

Se le denomina *SAE International* (Society of Automotive Engineers International) a la organización científica y educativa sin fines de lucro dirigida a los profesionales en la ingeniería aeroespacial, automoción y todas las industrias comerciales especializadas en la construcción de vehículos. Fue llevada a cabo tomando en cuenta las necesidades de la humanidad como objetivo primordial.

Para ello, la asociación cuenta con más de 138.000 ingenieros y técnicos relacionados con la industria aeroespacial, industria automotriz y de vehículos comerciales. Las competencias de SAE International son el aprendizaje permanente y el desarrollo de normas de consenso voluntario. (SAE International, 2012)

II.7.1 Automóviles pertenecientes a la organización

Existen distintos campos en los que está enfocada la organización; por la parte automotriz encontramos dos grandes categorías:

1. Fórmula SAE

Es una competencia de diseño estudiantil organizado por SAE International, cuenta con una compañía encargada de desarrollar el diseño de pequeños autos de carrera al estilo fórmula, cuyo prototipo debe ser evaluado por su potencial como un elemento de producción. Cada alumno del equipo se encarga de diseñar y construir pruebas de un ejemplar basado en una serie de normas cuyo propósito se viene dado por la garantía del funcionamiento de eventos en el propio establecimiento y la promoción de la solución inteligente de los problemas.

La Fórmula SAE promueve carreras y la excelencia en la ingeniería, ya que abarca todos los aspectos de la industria automotriz, incluyendo la investigación, diseño, fabricación, pruebas, desarrollo, marketing, administración y finanzas. Además, lleva a los estudiantes fuera del aula permitiéndoles aplicar las teorías de los libros de texto a experiencias reales de trabajo. (SAE International, 2012).

2. Baja SAE

Dentro de esta misma perspectiva, se encuentra Baja SAE el cual consiste en tres concursos regionales que simulan los proyectos de diseño de ingeniería del mundo real y sus desafíos relacionados. Los estudiantes de ingeniería tienen la tarea de diseñar y construir un vehículo todo terreno que pueda vencer los obstáculos propios de terrenos difíciles e incluso al agua. Tiene como objetivo presentar a todos los miembros estudiantiles SAE con un ambicioso proyecto que involucra tareas de planificación y fabricación encontradas en la introducción de un nuevo producto para el mercado de consumo industrial.

Los equipos deben competir unos contra otros en cuanto a la fabricación y diseño del coche. Los estudiantes deben funcionar como un equipo, no sólo para diseñar, construir, probar, promover y competir con un vehículo dentro de los límites de las normas, sino también para generar apoyo financiero para su proyecto y gestionar sus prioridades educativas. (SAE International, 2012).

CAPÍTULO III

MARCO METODOLÓGICO

En este capítulo se exponen y se detallan las actividades que se desarrollaron para alcanzar los objetivos propuestos en la presente investigación. Para ello, se siguió el siguiente esquema:

1. Investigación documental y análisis de las fuentes
2. Diseño de la propuesta
3. Prueba del sistema y cálculos
4. Análisis de Resultados
5. Conclusiones y Recomendaciones
6. Elaboración del libro y presentación del Trabajo de Grado

III.1. Investigación documental y análisis de las fuentes

En esta fase se hizo una exhaustiva revisión, selección y posterior análisis de los elementos teóricos relacionados con el tema en estudio que dieron fundamento a la presente investigación, basándose principalmente en la telemetría de los vehículos, así como en los diferentes productos existentes en el mercado que permiten su implementación.

Esta fase de documentación contemplan principio obtener información sobre los sistemas de telemetría existentes, las tecnologías y los protocolos de transmisión inalámbrica. De la misma manera, se determinan los principales estándares involucrados en el diagnóstico de vehículos, los sensores que poseen los vehículos de competencias SAE, los dispositivos encargados de manipular la información de sensores en el vehículo (hardware) y los sistemas para la interpretación de datos, es decir, el software.

Además de haber realizado una exhaustiva revisión bibliográfica relacionada con la temática, se entrevista a varios estudiantes que han participado o que actualmente participan en las actividades de competencias SAE pertenecientes a la UCAB. Esto permite contar con una visión más amplia, tanto teórica como práctica, al momento de tomar decisiones con relación a cuáles son los elementos y sistemas

más adecuados para la realización del sistema de monitorización en los vehículos de BAJA y Formula SAE.

III.1.1 Parámetros de medición del Sistema de Telemetría

Tomando en cuenta la documentación y de acuerdo a las necesidades de cada una de las competencias de SAE en las que participa la Universidad Católica Andrés Bello (BAJA SAE, Formula SAE), se seleccionan distintos parámetros de medición en el vehículo.

Con base a las similitudes y necesidades de ambos equipos, se eligen parámetros fundamentales para la realización de un sistema general compatible con ambas disciplinas.

Conociendo previamente las pruebas a las cuales son sometidos los vehículos mencionados, se acuerda determinar los parámetros: combustible consumido, temperatura y velocidad a la cual se desplaza el vehículo en tiempo real.

III.1.2 Selección del Hardware para el manejo de los datos

Como parte del Sistema de Telemetría y conociendo las condiciones a las cuales son expuestos los vehículos de competencia, especialmente el todoterreno del Equipo BAJA SAE (el cual es sometido a movimientos bruscos, saltos, exposición al agua y barro, etc.), se valora la necesidad de contar con un Hardware que permite procesar los datos adquiridos por los sensores, y que a su vez, sea capaz de transmitir los mismos al dispositivo emisor de la comunicación inalámbrica.

Basándose en lo anterior, se decide seleccionar la tarjeta Arduino MEGA, la cual se adapta perfectamente a las condiciones requeridas en las competencias, puesto que la misma resulta, entre otras cosas, ser una solución fácilmente programable y de fácil acceso a los equipos involucrados en dichas competencias.

III.1.3 Selección del protocolo de transmisión a utilizar

Basado en la documentación, se comparan distintos protocolos de transmisión entre los cuales se destacan WiFi, WiMax, IEEE 802.15.4, ZigBee, entre otros; finalmente se opta por utilizar la tecnología 802.15.4 para la comunicación del sistema de telemetría, debido a que esta ofrece significativas ventajas en el ámbito donde se desarrollan las competencias, y de los recursos y exigencias de las mismas.

Dicha tecnología ofrece ventajas tales como: alta movilidad, bajo costo, bajo consumo de energía y fácil implementación.

Posteriormente, se comparan los distintos dispositivos disponibles en el mercado que utilizan un protocolo de transmisión basado en tecnología del protocolo 802.15.4, seleccionando como el más idóneo para los requerimientos de esta investigación el dispositivo. (Ver TABLA 4).

III.1.4 Sistema de interpretación de datos (Interfaz)

Para el desarrollo y correcta aplicación de un software gráfico o interfaz, se investiga sobre los distintos lenguajes y aplicaciones que se pueden utilizar para mostrar al usuario los valores arrojados por los sensores de manera clara y precisa, sin necesidad de la instalación de múltiples programas para su visualización.

Con base en estos requerimientos se realiza una interfaz basada en el lenguaje JAVA, la cual permite al usuario que desee observar el comportamiento de los sistemas del automóvil de competencia, conocer los datos monitoreados arrojados por los sensores en tiempo real de una manera gráfica y sencilla.

De esta forma, se alcanzan los objetivos específicos 1, 2 y 3.

III.2. Diseño de la Propuesta

En esta fase, se describen todos aquellos procedimientos y materiales relacionados con el diseño del STMPVC-SAE.

III.2.1 Sensores:

Determinados ya los parámetros, se procede a la selección de los sensores requeridos para su medición, tomando en cuenta la compatibilidad de los mismos con el módulo encargado de procesar los datos (Tarjeta Arduino MEGA).

El diagrama de diseño que se muestra a continuación (*Figura 12*), Contempla el proceso de la comunicación entre los diferentes dispositivos involucrados en el Sistema de Telemetría.

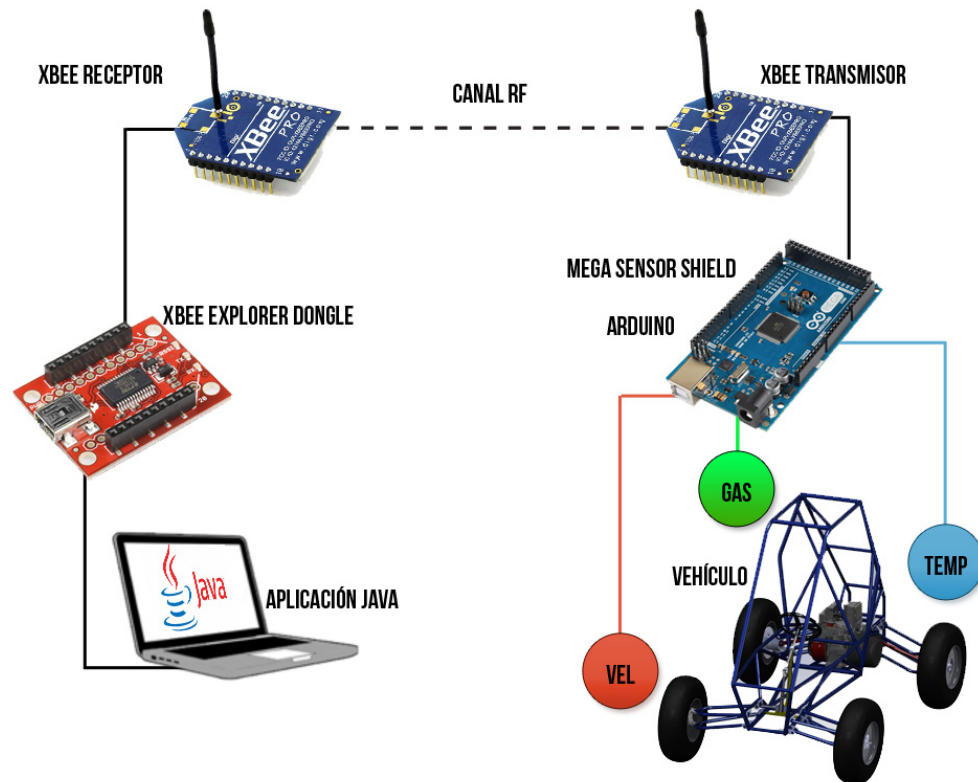


Figura 12. Diseño del sistema de telemetría.
Fuente: Propia.

Vale la pena destacar que para el correcto diseño del sistema anteriormente mencionado, se toma en cuenta el tipo de dato arrojado por cada sensor, es decir, digital o analógico, y en función de ello se programa la tarjeta Arduino MEGA, para el procesamiento de la data suministrada por los dispositivos de monitoreo en el vehículo de competición.

En este sentido, para el sensor que mide el nivel de combustible se propone el uso de un dispositivo capaz de obtener el flujo de líquido que pasa desde el tanque de gasolina al motor. Luego a través de cálculos de sustracción, es posible conocer la cantidad de gasolina disponible. Este procedimiento se ajusta a la normativa legal establecida por la Organización SAE para los vehículos de competición, la cual prohíbe alterar o introducir cualquier tipo de objeto dentro del tanque de gasolina. Por tal motivo se hizo imposible la utilización de algún dispositivo o sensor, en el interior del mismo.

Por su parte, para medir la velocidad, se sugiere la utilización e implementación de un módulo de GPS, el cual es capaz de captar a través de los satélites de posición, la velocidad a la que se desplaza el automóvil de competición.

En cuanto al sensor de temperatura, se utiliza un dispositivo eléctrico; este permite una vez realizados los cálculos, obtener la temperatura del aire a su alrededor.

III.2.2 Tarjetas modulares

Para la realización de un sistema de telemetría que se acople fácilmente a las necesidades de los equipos en competición, es necesario la utilización de una tarjeta modular programable, sencilla de configurar y que además posea en el mercado una extensa variedad de sensores compatibles.

Para el caso particular de este trabajo especial de grado, se utiliza la Tarjeta Arduino MEGA, la cual es capaz de soportar los dispositivos de comunicación inalámbrica XBee, así como múltiples dispositivos sensoriales con data digital y analógica.

III.2.3 Dispositivos de comunicación

Se escogen los dispositivos XBee capaz de efectuar la transmisión de datos a distancia y en movimiento. El modelo utilizado es el XBee Pro Serie 1 de 60mW de potencia, el cual trabaja en la banda de radiofrecuencia entre 2400-2483,5MHz.

Tomando en cuenta la documentación previa, se analizan y evalúan diversos dispositivos de transmisión por radiofrecuencia disponibles en el mercado, permitiendo seleccionar los módulos RF basados en el estándar IEEE 802.15.4. Los mismos se destacan por su gran alcance y poco consumo de energía ideal para la realización de un sistema limitado en potencia.

III.3. Prueba del Sistema y Cálculos

Durante esta fase se contempla la demostración del funcionamiento del STMPVC-SAE mediante procesos de pruebas y cálculos respectivos, para la comprobación del funcionamiento del dispositivo y diseño elegido.

Se desea estudiar el comportamiento de los módulos XBee en sus diferentes modos de operación (API y AT); de esta manera se determina cuál de ellos resulta ser el más eficaz para un sistema de telemetría que está expuesto a distintas condiciones

de operación. Por ende se realizan pruebas de ambos modos de operación y se registra la data recibida para obtener la fiabilidad de ambas configuraciones.

Debido a la multiplicidad de situaciones y condiciones a las cuales son sometidos los equipos SAE de la Universidad Católica Andrés Bello, se requieren dispositivos que se adapten a distintas velocidades y logren transmitir correctamente los parámetros monitoreados. Para ello, es necesario realizar distintas pruebas que permitirán determinar el correcto funcionamiento del sistema diseñado. Las pruebas o experimentos diseñados se llevan a cabo de la siguiente manera:

III.3.1 Pruebas estáticas

Con base en la metodología experimental descrita por Gutiérrez (2005), se realizan en primera instancia pruebas estáticas del sistema con la finalidad de obtener una primera lectura de los parámetros sin que la señal sufra ningún tipo de distorsión frente a variables relacionadas con la movilidad.

En este sentido, se plantean una serie de pruebas que van desde comprobar las condiciones básicas de operación del sensor de temperatura (Lm35), el correcto funcionamiento del software diseñado, hasta llegar a la comprobación del envío efectivo de paquetes mediante los módulos de transmisión inalámbrica.

Con esta primera fase experimental se estudia el comportamiento de la señal, la pérdida de paquetes, y la fidelidad del sistema, teniendo en cuenta distintas distancias entre los módulos de RF (emisor y receptor).

Habida cuenta de ello, se comienza con una distancia de 80m en una transmisión de 2 minutos, de 3 secuencias numéricas, donde cada una de dichas series se incrementa de forma unitaria, en múltiplos de tres, y múltiplos de cinco respectivamente; simulando de esta manera la interacción que existe en las competencias de Formula y Baja SAE de los tres tipos de parámetros a medir (velocidad, nivel de combustible y temperatura).

Posteriormente se incrementa la distancia a 300 y 600m, repitiéndose el proceso anteriormente descrito. Dichas pruebas se realizan en un espacio abierto con dispositivos interactuando a frecuencias a 2,4GHz (señales WiFi), para comprobar si se presentan o no interferencias significativas en los resultados.

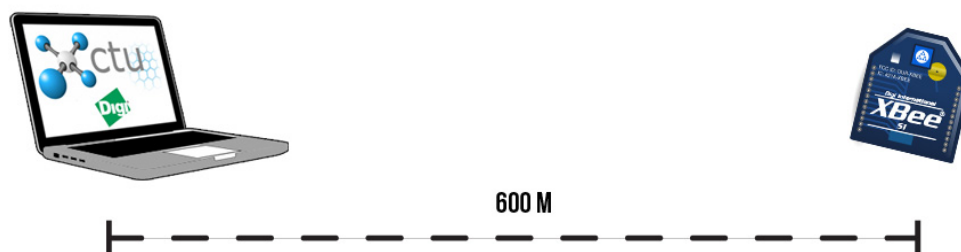


Figura 13. Pruebas Estáticas.

Fuente: Propia.

III.3.2 Pruebas Dinámicas:

En esta fase del experimento se lleva a cabo todo lo relacionado con la movilidad o dinámica del sistema de telemetría, sometiéndolo a distintas pruebas, simulando en la medida de lo posible, el entorno donde podrían competir los vehículos de SAE.

En este sentido, resulta importante contar con un sistema robusto ante las interferencias y variantes del ambiente en el tiempo, cuando hay variaciones en la distancia entre los módulos XBee o en la velocidad del móvil. Considerando lo anteriormente descrito se realiza un análisis de la interferencia sufrida por la señal en estos entornos, la cantidad de Bytes perdidos y el tiempo de respuesta de los equipos, entre otros.

Se somete a variación tanto la velocidad del móvil como la distancia entre los dispositivos XBee del sistema de telemetría; de la misma manera que en los experimentos anteriores, se dispone de un espacio abierto sin obstáculos relevantes. Esta prueba se realiza de forma repetida hasta obtener un comportamiento medio del sistema. Vale destacar que se toma en cuenta el *porcentaje de línea de visión*, como el porcentaje de tiempo donde existe visibilidad entre los módulos RF (emisor y receptor) con respecto a la duración total del experimento. (Ver *Figura 14*).

Como se observa en la *Figura 15* se coloca ambos dispositivos de transmisión en radios de 300 y 600m, con una variación de velocidad máxima que viene dada por

40, 60 y 80 km/h, las cuales se clasificaran en velocidades bajas, medias y altas respectivamente.

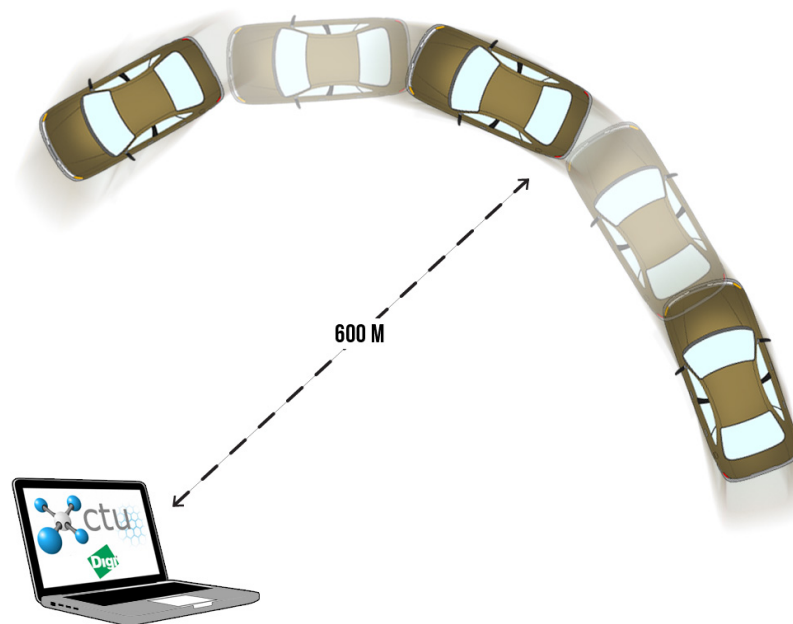


Figura 14. Pruebas dinámicas.
Fuente: Propia.

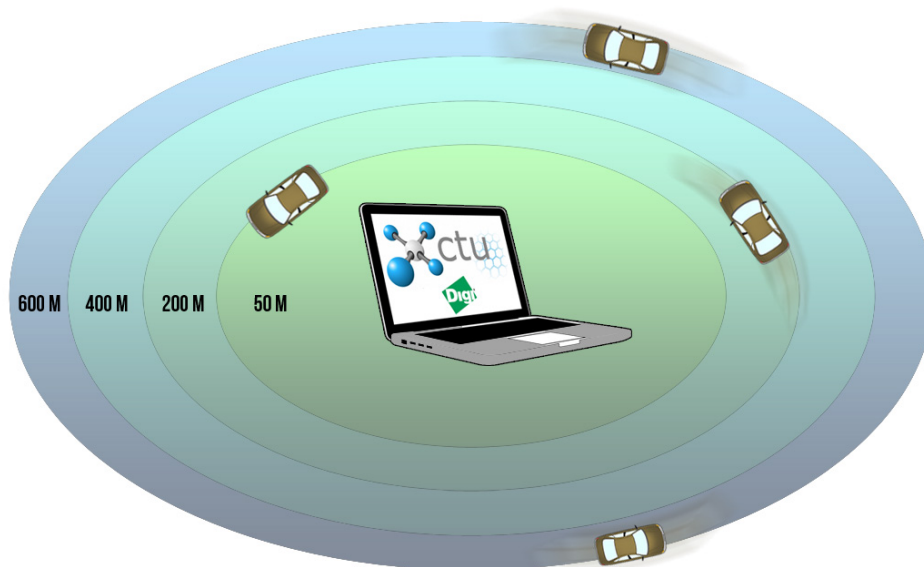


Figura 15. Radio de las pruebas dinámicas.
Fuente: Propia.

III.4. Análisis de Resultados

Esta fase abarca todo lo referente a los resultados obtenidos, los cuales se exponen utilizando distintas herramientas que permiten su mejor observación, interpretación y análisis.

De igual manera, se estudia todo lo referente a los resultados finales obtenidos en los experimentos, los cuales se basan en una serie de pruebas estáticas y dinámicas relacionadas al funcionamiento del sistema de telemetría propuesto. Dichas deducciones serán sometidas al análisis e interpretación en atención a los objetivos planteados en la investigación.

III.5. Conclusiones y Recomendaciones:

Esta fase abarca los aspectos finales de la investigación, las conclusiones se basan en establecer los parámetros finales sobre lo observado al final de la experimentación y se encuentra relacionada directamente con cualquiera de la variables indicadas en el planteamiento del problema de un trabajo investigativo. Para ello, se recogerán los hallazgos más importantes del proceso de investigación, tomando en cuenta los resultados obtenidos.

Así mismo, las recomendaciones se establecen con elemento germinativo de la investigación, ya que en ellas se dan sugerencias tomando en cuenta el alcance de la misma y los resultados obtenidos. De esta manera, futuros investigadores podrán entre otras cosas, hacer uso de dichas recomendaciones, utilizar la presente investigación como antecedente o punto de partida para próximas investigaciones vinculadas con la temática tratada.

III.6. Elaboración del libro y presentación del Trabajo de Grado

Esta última fase implica la organización, tabulación, revisión y transcripción de la información, así como la elaboración de la presentación de la investigación.

Se realiza un tomo con todos los requerimientos exigidos en la normativa para la presentación de los Trabajos Especiales de Grado. En este sentido el trabajo se estructura por capítulos, los cuales contienen cada uno de los pasos que se lleva a cabo para la culminación de la investigación, en este caso para el diseño del sistema

de telemetría para los vehículos de Formula y Baja SAE de la Universidad Católica Andrés Bello.

El trabajo investigativo en su totalidad, se somete a evaluación por parte de docentes especializados en el tema tratado, de manera que puedan emitir su juicio acerca del mismo. Una vez culminado el proceso evaluativo y aprobatorio, el trabajo escrito es empastado atendiendo las normas. Posteriormente, este trabajo es llevado a los anaqueles de la biblioteca central de la Universidad Católica Andrés Bello (UCAB) donde podrá ser consultado por otros investigadores interesados en el tema.

CAPÍTULO IV

RESULTADOS

Durante este capítulo se presentan los datos obtenidos de las distintas pruebas, así como también un análisis detallado de lo ocurrido en cada experimento.

IV.1. Configuración de los Módulos XBee

Durante la fase de pruebas, se utilizaron los módulos XBee con la siguiente configuración en el modo AT (modo transparente):

TABLA 5
Configuración del XBee.

PRUEBAS MODO AT		
Alias	RECEPTOR	EMISOR
Shield	Xbee Explorer USB	Arduino Mega Shield
Modelo	Xbee Pro Serie 1 60mW	
Frecuencia	2,4GHz	
Funtion Set	802.15.4	
Familia	XBP24	
Firmware Version	10e6	
Canal	C	
PAN id	6032	
Destination Address High	0	0
Destination Address Low	0	1
MAC MODE	802.15.4 + MaxStreamheader w/ACKS [0]	
16-bit Source Address	1	0
Serial Number High	13A200	13A200
Serial Number Low	407A6AA7	407A68BB
Coordinator Enable	EndDevice [0]	
Power Level	Highest [4]	
Sleep Mode	No Sleep [0]	
Interface Data Rate	9600 [3]	
Parity	No parity [0]	
API Enable	API Disable [0]	

Nota: Realizado con datos propios.

Se configuraron los dispositivos de transmisión en modo AT (modo transparente) mediante el software X-CTU; como se puede observar en la TABLA 5, se utiliza la mayor potencia disponible por el XBee Pro Serie 1 garantizando un comportamiento óptimo en las pruebas.

IV.2. Etapa 1: Pruebas Estáticas:

En esta fase se realizaron experimentos vinculados a la efectividad del sistema de telemetría en un entorno estático, donde se someten los módulos de radiofrecuencia a ciertas distancias del módulo receptor.

Para ello se tomó en cuenta la línea de visión que existía entre los XBee. Se clasifico en dos grandes experimentos (en espacios abiertos y en espacios cerrados), los cuales son:

TABLA 6
Pruebas estáticas con línea de Visión (Outdoor).

Experimento	1.a	1.b
Distancia Máx. (m)	300	600
Efectividad (%)	100%	100%

Nota: Realizado con datos propios.

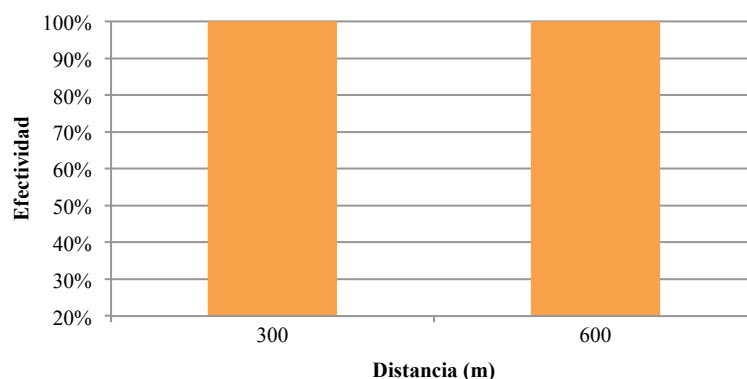


Figura 16. Pruebas estáticas (Outdoor): Efectividad vs Distancia.
Fuente: Propia.

Se observa en la TABLA 6 y en la *Figura 16*, los resultados de las pruebas realizadas en un espacio abierto (Outdoor), donde no existía ningún tipo de obstáculos que interfirieran con la línea de visión.

Nótese que los dispositivos fueron ubicados a distancias de 300 y 600 metros (experimento 1.a y experimento 2.a respectivamente) donde no existió ningún tipo de error en la comunicación, sin importar la alineación cardinal de las antenas de los XBee. Así mismo, el aumento de la distancia no produjo ningún cambio, observándose de igual forma un 100% en el porcentaje de efectividad.

TABLA 7
Pruebas estáticas sin línea de visión (Indoor).

Experimento	2.a	2.b
Distancia Máx. (m)	25	100
Efectividad (%)	97%	43%

Nota: Realizado con datos propios.

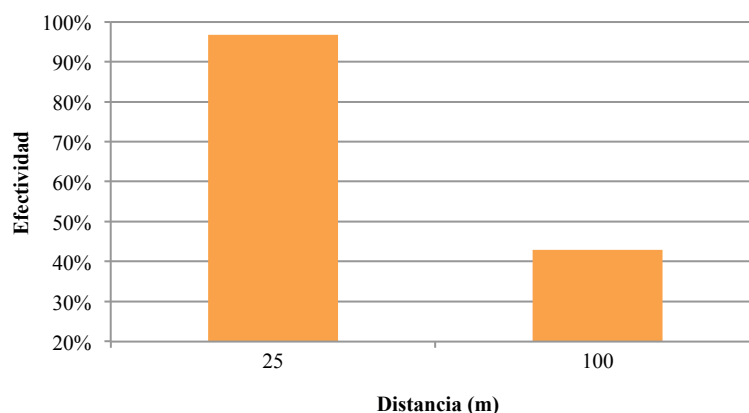


Figura 17. Pruebas estáticas sin línea de visión (Indoor).
Fuente: Propia.

En cuanto a los experimentos 2.a y 2.b en espacios cerrados (Indoor), los módulos XBee se ubicaron en un ambiente con obstáculos que impiden tener línea de visión. Las distancias entre los dispositivos, estuvieron en el orden de 25 y 100 m. (Ver TABLA 7).

La *Figura 17* permite apreciar la pérdida significativa del porcentaje de efectividad al momento de aumentar la distancia de 25 a 100 metros, es decir, la misma se redujo a un 40%; esto se debe entre otras cosas, a la atenuación, la absorción y la dispersión sufrida por la señal al impactar contra los objetos que obstaculizan su trayectoria.

Con base a lo anterior, es importante destacar que el porcentaje de efectividad es mayor cuando hay línea de visión entre los dispositivos RF. Al comparar los experimentos 1.b y 2.b, se puede apreciar que aún cuando la distancia entre los dispositivos en 1.b fue 6 veces mayor que en el 2.b, el primero resulto poseer un porcentaje de efectividad del 100%, a diferencia de la prueba 2.b que arroja grandes pérdidas en la transmisión a solo 100m del receptor.

IV.3. Etapa 2: Pruebas Dinámicas

En esta etapa se realizaron pruebas en movimiento, para identificar el comportamiento del sistema de telemetría en situaciones donde el vehículo se encuentra en desplazamiento. (Ver fotografías en el Anexo A).

Las pruebas se trabajaron en un rango de distancia de 0 a 600 m, y se clasificaron los experimentos por velocidades máximas. El primer grupo de pruebas (3.a, 3.b, 3.c, 3.d y 3.e), fueron desarrolladas a una velocidad máxima de 30km/h, denominadas como *velocidades bajas*. Seguidamente para las pruebas (3.f, 3.g, 3.h, 3.i y 3.j) se tomaron los datos a una velocidad máxima de 60 km/h (*velocidades intermedias*) y por último, las correspondientes a las *velocidades altas* (3.k, 3.l, 3.m, 3.n y 3.o), fueron llevadas a cabo a velocidades no mayores de 90 km/h.

TABLA 8
Pruebas Dinámicas a Velocidades Bajas (Máx. 30 km/h).

Experimento	3.a	3.b	3.c	3.d	3.e
Velocidad Máx. (km/h)	30				
Distancia Máx. (m)	300				
Efectividad (%)	69%	85%	87%	86%	99%
Línea de Visión (%)	42%	69%	68%	69%	89%

Nota: Realizado con datos propios.

TABLA 9
Pruebas Dinámicas a Velocidades Intermedias (Máx. 60 km/h).

Experimento	3.f	3.g	3.h	3.i	3.j
Velocidad Máx. (km/h)	60				
Distancia Máx. (m)	300				
Efectividad (%)	70%	91%	85%	96%	97%
Línea de Visión (%)	41%	72%	71%	91%	90%

Nota: Realizado con datos propios.

TABLA 10
Pruebas Dinámicas a Velocidades Altas (Máx. 90 km/h).

Experimento	3.k	3.l	3.m	3.n	3.o
Velocidad Máx. (km/h)	90				
Distancia Máx. (m)	300				
Efectividad (%)	67%	96%	82%	96%	98%
Línea de Visión (%)	40%	91%	72%	92%	91%

Nota: Realizado con datos propios.

Como se puede apreciar en la TABLA 8, TABLA 9 y TABLA 10 se muestran los resultados de las primeras pruebas realizados con una distancia entre los dispositivos no mayor a 300 m. Los experimentos fueron llevados a cabo haciendo variaciones en la velocidad y la distancia. Dichos resultados permiten evidenciar que aquellos que presentan pocos obstáculos entre los XBee, poseen un mayor porcentaje de línea de visión entre los dispositivos.

A continuación se presenta la representación gráfica de los resultados obtenidos en los experimentos anteriormente descritos:

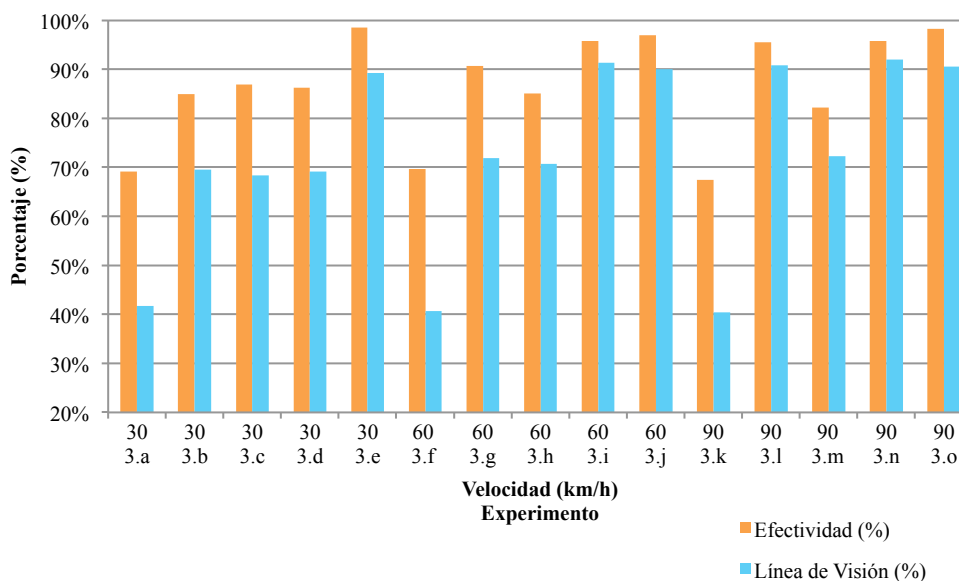


Figura 18. Pruebas Dinámicas (Radio de 300 m): Efectividad vs Línea de Visión.
Fuente: Propia.

En la *Figura 18* se presentan los porcentajes de efectividad en la transmisión obtenidos en las distintas pruebas realizadas a una velocidad máxima (30, 60 y 90 km/h) y dentro de un radio de distancia entre los modulo RF de 300 metros.

El análisis macro de la misma, permite determinar que el porcentaje de línea de visión incide significativamente en el porcentaje de efectividad de las transmisiones del sistema, es decir, a mayor porcentaje de línea de visión, mayor es el porcentaje de efectividad, lo que se traduce en un menor porcentaje de pérdidas en la comunicación.

TABLA 11
Pruebas Dinámicas (Radio de 300 m), Efectividad prom. por Velocidad.

Velocidad Máx. (km/h)	30	60	90
Efectividad (%)	91%	88%	88%

Nota: Realizado con datos propios.

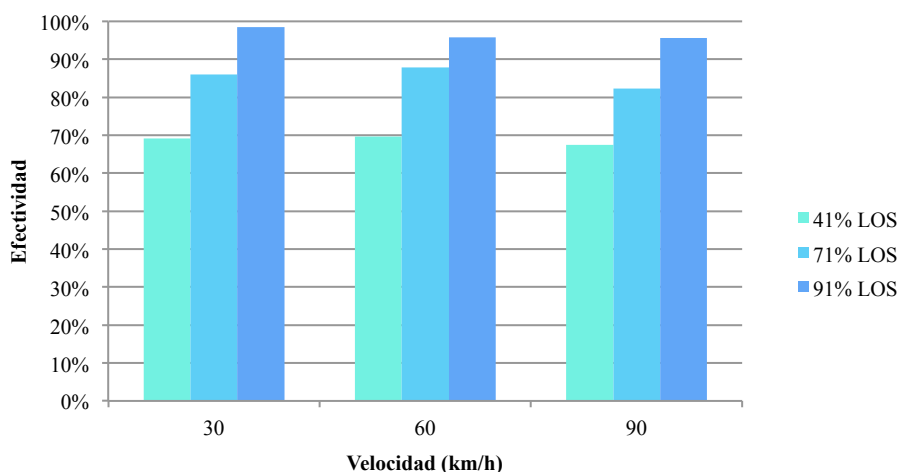


Figura 19. Pruebas Dinámicas (Radio 300 m): Velocidad vs Efectividad.
Fuente: Propia

La TABLA 11 y la *Figura 19* presentan el promedio del porcentaje de efectividad obtenido con respecto a las 3 velocidades máximas de cada experimento.

Vale destacar que la *Figura 19* muestra los datos clasificados por rangos de porcentaje de línea de visión, es decir, pruebas con un porcentaje de línea de visión entre 50-65, con un porcentaje de línea de visión entre 65-80 y pruebas con un porcentaje de línea de visión entre 80-100. Los datos de esta última fueron extraídos mediante el promedio de las Tablas 8, 9 y 10.

El análisis permite sostener que a mayor velocidad, el porcentaje de efectividad de las transmisiones, tiende a disminuir gradualmente. Sin embargo, aunque la velocidad no resulta ser un factor muy relevante para determinar el porcentaje de efectividad de la transmisión, esta le agrega ciertas pérdidas al sistema que se deben considerar.

TABLA 12
Pruebas Dinámicas a Velocidades Altas (Máx. 90 km/h).

Velocidad Máx. (km/h)	90					
Distancia Máx. (m)	300					
Efectividad (%)	67%	82%	98%	96%	96%	98%
Línea de Visión (%)	40%	72%	91%	91%	92%	93%

Nota: Realizado con datos propios.

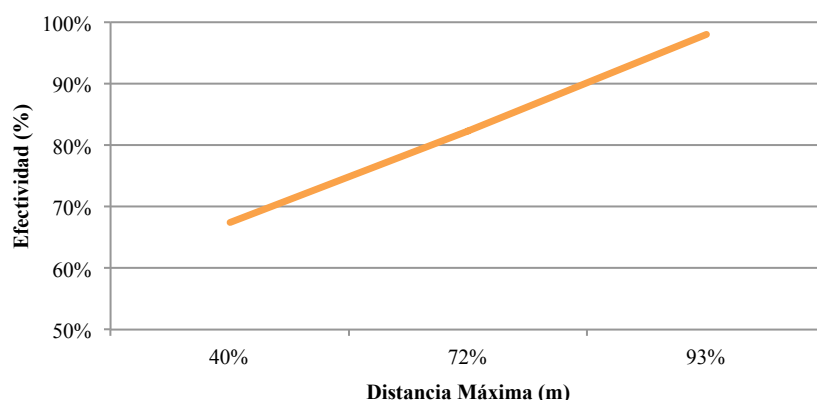


Figura 20. Pruebas Dinámicas a Velocidades Altas (Max 90 km/h).
Fuente: Propia.

Los resultados que se observan en la TABLA 12 y la *Figura 20* evidencian que a mayor porcentaje de línea de visión, mayor es el porcentaje de efectividad. Estas pruebas fueron realizadas a *velocidades altas* (90 km/h).

TABLA 13
Pruebas Dinámicas (Radio de 600m), Efectividad vs Línea de Visión.

Velocidad Máx. (km/h)	30	60	90
Efectividad (%)	99%	98%	98%
Línea de Visión (%)	89%	89%	93%

Nota: Realizado con datos propios.

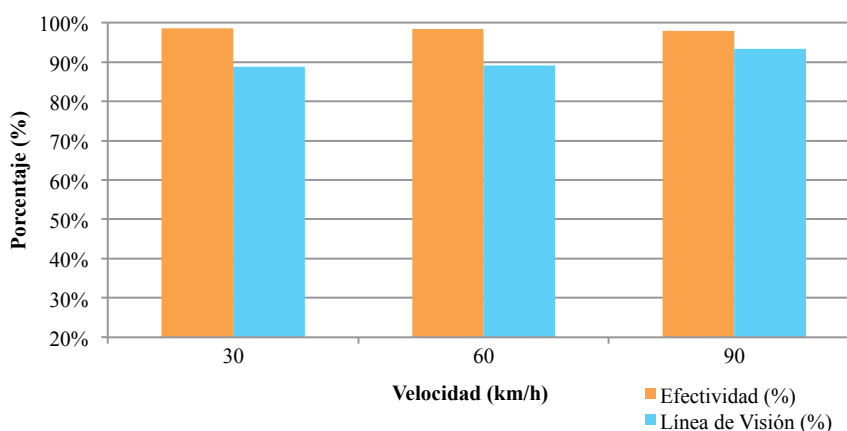


Figura 21. Pruebas Dinámicas (Radio de 600m).
Fuente: Propia.

Los resultados obtenidos en esta prueba, permiten sostener que a pesar de que se desarrolló a distintas velocidades, estas resultaron no ser totalmente determinantes para el porcentaje de efectividad del sistema. En este sentido, se tiene que la distancia se incrementó a 600 m obteniéndose un porcentaje de efectividad superior al 85% en todas las pruebas realizadas, incluso a altas velocidades. (Ver TABLA 13 y *Figura 21*)

Con la finalidad de contar con un sistema más robusto y de mayor fiabilidad, los XBee brindan el modo API, el cual consiste en suministrar datos adicionales enviados como cabecera en la transmisión, tales como: dirección de destino, dirección de origen, checksum (Verificación de errores), entre otros; los cuales son de gran utilidad para las redes de varios nodos, en donde se necesita dirigir la información a un dispositivo en específico, y en donde la retransmisión en caso de errores sea de gran necesidad. Por lo antes expuesto, se consideró pertinente, realizar pruebas en modo API.

A continuación se muestran los resultados obtenidos en las pruebas, luego de haber configurado el XBee en modo API (Application Programming Interface).

TABLA 14
Pruebas Dinámicas Modo API (Radio de 600 m), Efectividad vs Línea de Visión

Experimento	4.a	4.b	4.c
Velocidad Máx. (km/h)	30	60	90
Efectividad (%)	88%	87%	87%
Línea de Visión (%)	92%	91%	90%

Nota: Realizado con datos propios.

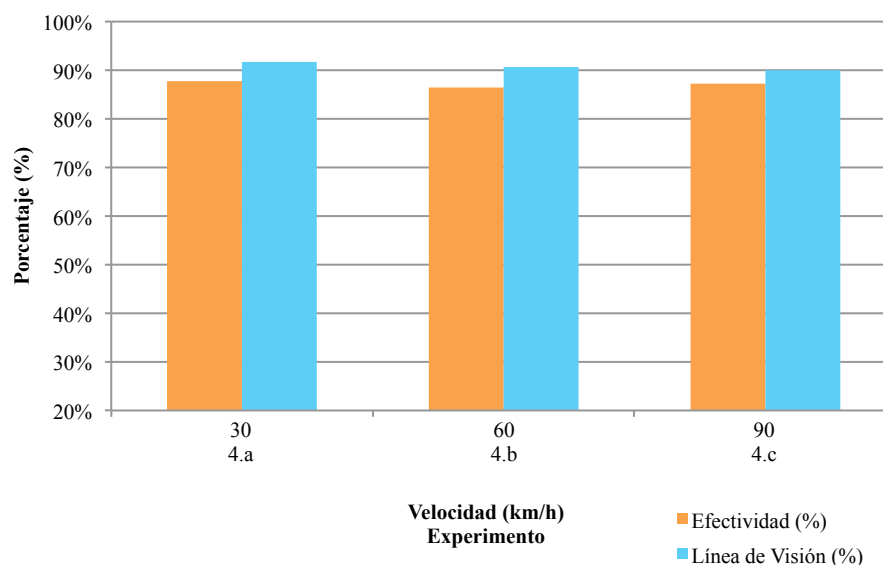


Figura 22. Pruebas Dinámicas Modo API (Radio de 600 m).
Fuente: Propia.

Los experimentos 4.a, 4.b y 4.c se efectuaron tomando en cuenta los tres rangos principales de velocidad que alcanzan los prototipos SAE (velocidades bajas, intermedias, altas). Nótese que tuvieron un porcentaje de efectividad de transmisión mayor a 80, pero al igual que la configuración AT, se vieron afectados significativamente por la línea de visión. (Ver TABLA 14).

Los resultados obtenidos en esta prueba, demuestran que para este tipo de sistemas, el modo API no exhibe grandes beneficios a la hora de determinar el porcentaje de efectividad y el porcentaje de línea de visión. (Ver *Figura 23*)

TABLA 15
Modo AT vs Modo API (Radio de 600 m), Velocidad Máx. 90 km/h.

Modo	Modo AT	Modo API
Efectividad (%)	98%	87%
Línea de Visión (%)	93%	90%

Nota: Realizado con datos propios.

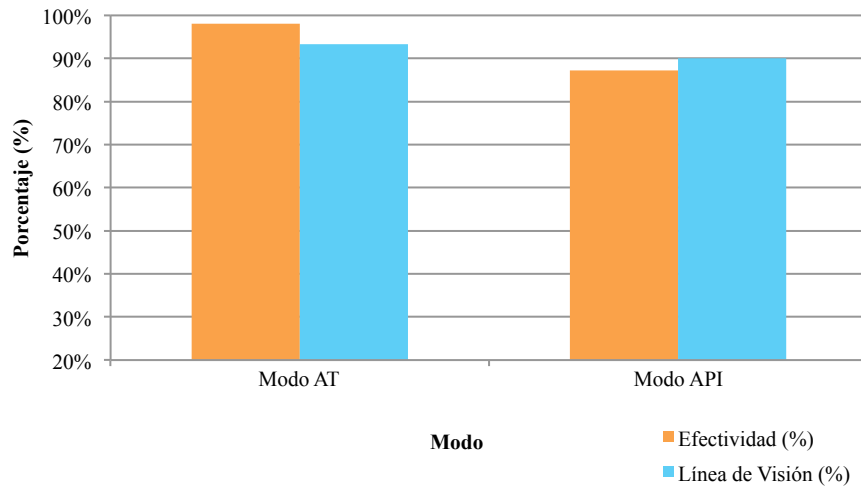


Figura 23. Modo AT vs Modo API (Radio 600m).
Fuente: Propia.

La *Figura 23* permite comparar los comportamientos de los modos AT y API a 600 m. Los resultados obtenidos permiten apreciar que el porcentaje de efectividad de transmisión es mayor en la configuración del modo AT.

Vale la pena destacar que en modo AT (transparente), se puede configurar el módulo a través de simples comandos AT, mientras que, en el modo API es algo más complicado, debido a que no existe el modo datos y modo comandos. En consecuencia, para el modo API se requiere la utilización de un protocolo, el cual permite comunicarse con los módulos mediante unas tramas con su cabecera, datos, final de trama.

Sabiendo que para este tipo de sistemas de telemetría se utiliza una conexión punto a punto para la monitorización del vehículo, no es necesario el envío de información complementaria tales como: dirección de origen, dirección de destino; la cual retardaría la transmisión de datos. Por lo tanto, se prefiere la utilización del modo AT sobre el modo API.

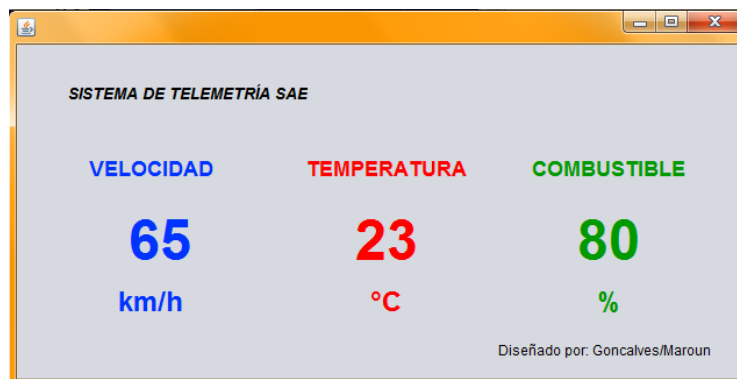


Figura 24. Interfaz del sistema de telemetría.
Fuente: Propia.

En la *Figura 24* se observa la interfaz realizada para la visualización del sistema de telemetría para la monitorización de parámetros en los vehículos de competencias SAE. En este sentido se pueden apreciar los parámetros más importantes que conforman el sistema de medición a distancia (velocidad en km/h, temperatura en grados centígrados y el nivel de combustible determinado a través del porcentaje de gasolina que posee el automóvil). El diagrama de conexión del sensor utilizado (Temperatura) a la Expansion Shield de Arduino, puede verse en el Anexo B.

Para ello se optó por la realización de una interfaz sencilla, que permite ser manipulada o utilizada por cualquier persona, sin necesidad de poseer conocimientos técnicos acerca de la programación. Con base en esta idea, el desarrollo de esta interfaz se programó en Java, un lenguaje universal, orientado a objetos, de fácil adaptación para todos aquellos que necesiten ajustar, agregar o modificar algún parámetro. (Ver Códigos de Programación en Arduino y JAVA en Anexos C,D y E).

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

V.1. Conclusiones

A continuación se presentan las conclusiones obtenidas una vez culminado el trabajo de investigación sobre el diseño del sistema de telemetría en los automóviles de competencia Baja y Formula SAE:

1. El objetivo fundamental de un sistema de telemetría es recoger datos de un lugar remoto o inalcanzable y transmitirlos a un sitio donde puedan ser correctamente estudiados y analizados, con la finalidad de tomar decisiones oportunas e inmediatas con relación a los datos obtenidos.

2. Actualmente la telemetría ha ocupado una gran importancia en las competencias automovilísticas como la Formula 1 o Nascar, donde no solo se monitorea y controla las funciones principales del automóvil, sino también con el piloto de carreras, estudiando el comportamiento que este maneja con el automóvil desde los pits. La telemetría brinda la posibilidad de ver en tiempo real el comportamiento de casi todos los componentes del vehículo durante la competencia mediante sistemas de transmisión que son monitoreados en equipos especializados

3. Para el desarrollo de los procesos de medición, se hace necesario la utilización e implementación de sensores, encargados de producir una señal proporcional al valor de una propiedad mediante una correlación predefinida, sin alterar el sistema medido. En este sentido, quedaron identificados como parámetros más importantes la temperatura, la velocidad y el flujo de combustible para conformar el Sistema de medición a distancia.

4. La metodología que se utilizó para escoger los componentes del sistema de Telemetría propuesto (STMPVC-SAE), se basó en un estudio del ambiente de competencias y de los propios vehículos. Desde el análisis de la distancia necesaria para desarrollar el sistema de transmisión adecuadamente, hasta el canal en el cual se puede transmitir según los entes reguladores, incluyendo la tecnología que

se puede utilizar y finalmente que tipo de transmisor cumple con los requisitos para obtener la transmisión deseada.

5. Se diseñó un sistema de telemetría capaz de monitorizar parámetros mediante el uso de la Tarjeta Arduino MEGA, la cual destaca por su gran compatibilidad con los sensores en el mercado y cuenta con un lenguaje sencillo de programar. Así mismo, se escogió el estándar IEEE 802.15.4 y a los Equipos XBee PRO como el sistema de transmisión inalámbrica y los equipos más adecuados para el STMPVC-SAE, por su alto desempeño frente a condiciones de movimiento y a la notable efectividad en las transmisiones requeridas por los ambientes de competencias SAE.

6. Otro elemento que vale la pena señalar, tiene que ver con el reto que significó trabajar con la comunicación inalámbrica al tener un punto fijo y otro en movimiento, ya que la velocidad es un factor de difícil manejo dentro de las leyes que rigen este tipo de comunicaciones. En este sentido, se estableció una distancia máxima de transmisión de 600 m, la cual se necesitaba para las pruebas.

7. Se realizaron pruebas de tipo estáticas y dinámicas en distintos ambientes (Indoor y Outdoor), con la finalidad de demostrar el funcionamiento del estándar de comunicación adoptado por el sistema de telemetría. La efectividad de las transmisiones de datos inalámbricas del Sistema de Telemetría, resultaron ser significativamente dependientes del porcentaje de la línea de visión entre los dispositivos XBee (emisor y receptor). Así mismo se pudo evidenciar que a través de las pruebas, que no hay diferencias significativas entre los escenarios estáticos y dinámicos, alcanzando velocidades de hasta 90 km/h sin que la comunicación e información transmitida entre los componentes del STMPVC-SAE se vea comprometida (Efectividad por encima del 90%).

8. De igual manera se desarrolló una interfaz abierta sobre JAVA, capaz de ser ajustada según las necesidades del usuario y los requerimientos de la competencia, creando de esta forma un sistema escalable en el tiempo, y capaz de visualizar un resumen de los parámetros seleccionados para ser medidos durante las competencias de Formula y Baja SAE.

V.2. Recomendaciones

Los resultados obtenidos permiten sugerir las siguientes recomendaciones:

Se sugiere el estudio de la alimentación de energía alterna como fuente de corriente para el Sistema de Telemetría mediante la implementación de paneles solares en la parte superior de los vehículos, el cual garantizaría un flujo continuo de energía, y a su vez, se estaría colaborando con el cuidado del medio ambiente.

De la misma manera, se le recomienda a futuros investigadores, la posibilidad de desarrollar un programa capaz de almacenar la data obtenida por el sistema de monitorización, de forma tal que los equipos en competición puedan estar al tanto del desempeño que ha registrado el carro a lo largo de la competencia y a su vez, generar estadísticas del funcionamiento y rendimiento del vehículo con respecto al tiempo.

Sería importante contar con la presencia de los vehículos de competencia Formula y Baja SAE al momento de realizar las pruebas. Esto permitiría ubicar en el vehículo los sensores y probar el sistema de telemetría propuesto en el campo bajo condiciones reales.

Por último sería recomendable la instalación del componente de telemetría en el vehículo de competencia SAE y la ubicación de los módulos RF en la parte superior del mismo, ya que como se pudo observar en los resultados, la línea de visión representa el factor más importante a la hora de lograr una transmisión exitosa, efectiva y confiable.

Referencias Bibliográficas

- Arduino. (2011). *Arduino Mega 2560*. Recuperado el 12 de Febrero de 2014, de Arduino: <http://arduino.cc/en/Main/ArduinoBoardMega2560>
- Barriga, W., & Zúñiga, F. (Noviembre de 2006). Tecnologías Inalámbricas de corto alcance: ZigBee y Bluetooth. *Trabajo de Grado* . Ecuador.
- Bentley, J. (2000). *Sistemas de Medición. Principios y Aplicaciones* (Segunda edición ed.). Mexico D.F.: Editorial CECSA.
- Carden, F., Jedlicka, R., & Henry, R. (2002). *Telemetry Systems Engineering*. Norwood: Editorial Artech House, Inc.
- DFRobot. (19 de Junio de 2012). *Mega IO Expansion Shield V2(SKU:DFR0165)*. Recuperado el 4 de Junio de 2014, de DFRobot: [http://www.dfrobot.com/wiki/index.php/Mega_IO_Expansion_Shield_V2\(SKU:DFR0165\)](http://www.dfrobot.com/wiki/index.php/Mega_IO_Expansion_Shield_V2(SKU:DFR0165))
- Digi International Inc. (11 de Febrero de 2008). *XBee Znet 2.5/XBee-PRO Znet 2.5 OEM RF Modules*. Recuperado el 16 de Marzo de 2014, de http://www0.egr.uh.edu/courses/ece4437/labsupport/xbec%20zigbee/digi%20xbec%20manual%2090000866_c.pdf
- Faludi, R. (2010). *Building Wireless Sensor Networks*. Sebastopol: Editorial O'Reilly Media Inc.
- Globalsat Technology Corporation. (Mayo de 2014). *Globalsat GPS Module*. Recuperado el 7 de Junio de 2014, de USGlobalsat: http://www.usglobalsat.com/store/download/717/EM506_um.pdf
- Gómez de la Calle, M. (2010). Comunicaciones en la Fórmula 1: Telemetría. *Trabajo de Grado* . Madrid.
- Groussard, T. (2012). *Java 7 los fundamentos del lenguaje Java*. Barcelona: Ediciones ENI.
- Gutiérrez, A. (2005). *Metodología Experimental*. Ciudad de Mexico: Limusa S.A.

- Hebel, M., & Bricker, G. (08 de Octubre de 2012). *Getting Started with XBee RF Modules*. Recuperado el 3 de Abril de 2014, de Parallax, Inc.: <http://www.parallax.com/sites/default/files/downloads/122-32450-XBeeTutorial-v1.0.1.pdf>
- Hernández, J. (Septiembre de 2013). Sistema de Telemetría para un Vehículo de Formula SAE. *Trabajo de Grado*. Barcelona.
- Huertas, E., & Véliz, F. (13 de Octubre de 2006). Sistema Monitor Remoto Interactivo de vehículos (SMRI). *Trabajo de Grado*. Caracas, DC, Venezuela.
- Micropik. (s.f.). *Savant FS-3400AH Diesel Flow Sensor*. Recuperado el 8 de Abril de 2014, de Micropik: <http://www.micropik.com/PDF/FS-3400AH.pdf>
- Oxer, J., & Blemings, H. (2009). *Practical Arduino*. New York: Editorial TIA.
- Pallas, R. (2001). *Sensores y Acondicionadores de señal* (Tercera edición ed.). Mexico D.F.: Editorial Alfaomega.
- SAE International. (Septiembre de 2010). *SAE*. Recuperado el 14 de Junio de 2014, de http://students.sae.org/cds/formulaseries/rules/2014_fsae_rules.pdf
- Stallings, W. (2004). *Comunicaciones y Redes de Computadores* (Séptima edición ed.). Madrid: Editorial Pearson Educación, S.A.
- Tanenbaum, A. (2003). *Redes de Computadoras* (Cuarta edición ed.). Mexico D.F.: Editorial Pearson Education, Inc.
- Texas Instruments. (Octubre de 2013). LM35 Precision Centigrade Temperature Sensors. Dallas, TX, USA.
- Vargas, M. (Agosto de 2008). Sistema de Monitoreo y Control Remoto para una Central Micro-Hidráulica. *Trabajo de Grado*. Santiago de Chile.
- Zabler, E. (2002). *Los Sensores en el Automóvil*. Stuttgart: Editorial Robert Bosch GmbH.

ANEXO A
Fotografías durante Pruebas Experimentales

DISEÑO DE UN SISTEMA DE TELEMETRÍA PARA LA MONITORIZACIÓN DE
PARÁMETROS EN LOS VEHÍCULOS DE COMPETENCIAS SAE.

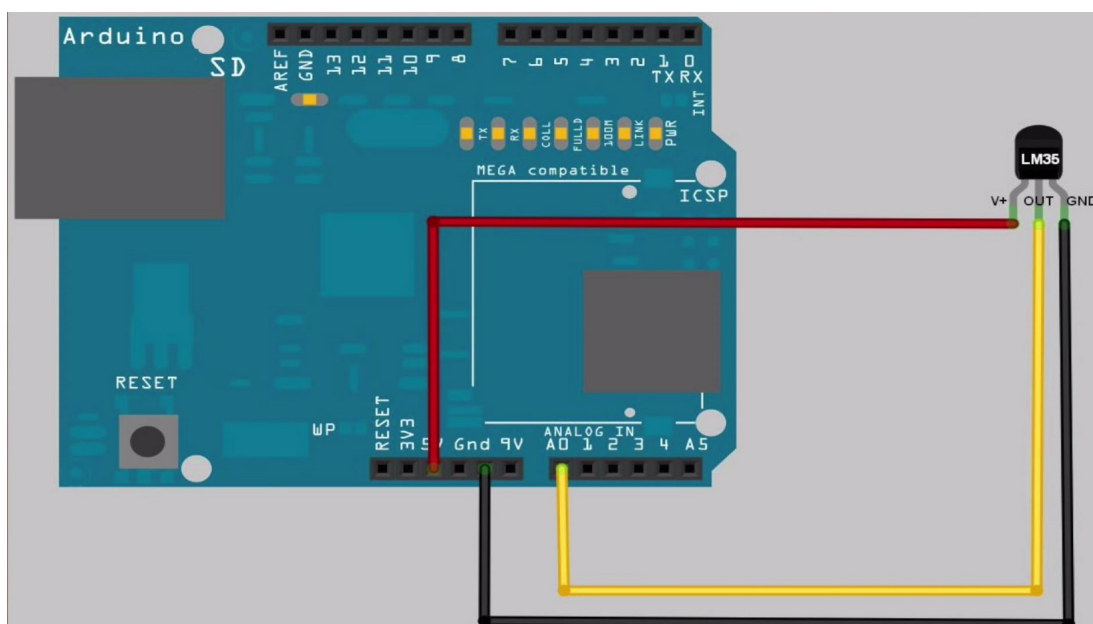


DISEÑO DE UN SISTEMA DE TELEMETRÍA PARA LA MONITORIZACIÓN DE PARÁMETROS EN LOS VEHÍCULOS DE COMPETENCIAS SAE.



ANEXO B

Diagrama Sensor de Temperatura y Shield Arduino



ANEXO C

Código del módulo de procesamiento de datos Arduino

```
intvel = 0;
intvelo;
int temp;
int gas;
inttank = 15; // Se introduce la cantidad de litros que lleva el
tanque
intrandomvel, randomgas;
intfueltank;

voidsetup() // Se ejecuta cada vez que el Arduino se inicia
{
  Serial.begin(9600);    //Inicia comunicación serial
  fueltank=30-tank;
  fueltank=tank+fueltank;
}

void loop()
{
  if (vel>=30)
  vel=15;
  if (vel<1)
  vel=15;

  temp= (5.0 * analogRead(0)*100.0)/1023.0;
  delay (500);

  if (fueltank<=1)
  fueltank=30;

  fueltank=fueltank-0.00001;

  vel=vel+1;
  velo=vel+39;
  Serial.println(velo);
  delay(1000);
  temp=temp*0.4;
  Serial.println(temp);
  delay(1000);
  gas=fueltank+70;
  Serial.println(gas);
  delay(1000);
}
```

ANEXO D

Código de la Clase SerialTest de JAVA

```
package SerialTest;

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.io.OutputStream;
import gnu.io.CommPortIdentifier;
import gnu.io.SerialPort;
import gnu.io.SerialPortEvent;
import gnu.io.SerialPortEventListener;
import java.io.IOException;
import java.util.Enumeration;

public class SerialTest implements SerialPortEventListener {
    SerialPort serialPort;
    private static final String PORT_NAMES[] = {
        "COM4"

    };

    private BufferedReader input;
    private OutputStream output;
    private static final int TIME_OUT = 2000;
    private static final int DATA_RATE = 9600;

    public String num = "nada";

    public void initialize() {

        System.setProperty("gnu.io.rxtx.SerialPorts",
"COM4");

        CommPortIdentifier portId = null;
        Enumeration portEnum =
CommPortIdentifier.getPortIdentifiers();

        while (portEnum.hasMoreElements()) {
            CommPortIdentifier currPortId =
(CommPortIdentifier) portEnum.nextElement();
            for (String portName : PORT_NAMES) {
                if (currPortId.getName().equals(portName)) {
                    portId = currPortId;
                    break;
                }
            }
        }
        if (portId == null) {
            System.out.println("No se encontrÃ³ el puerto
COM.");
            return;
        }

        try {
            serialPort = (SerialPort)
portId.open(this.getClass().getName(),
TIME_OUT);
```

```
        serialPort.setSerialPortParams(DATA_RATE,
                                         SerialPort.DATABITS_8,
                                         SerialPort.STOPBITS_1,
                                         SerialPort.PARITY_NONE);

        input = new BufferedReader(new
InputStreamReader(serialPort.getInputStream()));
        output = serialPort.getOutputStream();

        serialPort.addEventListener(this);
        serialPort.notifyOnDataAvailable(true);
    } catch (Exception e) {
        System.err.println(e.toString());
    }
}

public synchronized void close() {
    if (serialPort != null) {
        serialPort.removeEventListener();
        serialPort.close();
    }
}

public synchronized void serialEvent(SerialPortEvent oEvent) {
    if (oEvent.getEventType() ==
SerialPortEvent.DATA_AVAILABLE) {
        try {
            String inputLine=input.readLine();
            System.out.println(inputLine);
            num = inputLine;
            num();
            UI.g=inputLine;
        } catch (Exception e) {
            System.err.println(e.toString());
        }
    }
}

public String num () {
    return num;
}

public static void main(String[] args) throws Exception {
    SerialTest main = new SerialTest();
    main.initialize();
    Thread t=new Thread() {
        public void run() {
            try {Thread.sleep(1000000);} catch
(InterruptedIOException ie) {}
        }
    };
    t.start();
    System.out.println("Iniciado");
}
}
```

ANEXO E

Código de la Clase UI (Interfaz) de JAVA

```
package SerialTest;

import java.util.concurrent.TimeUnit;
import java.util.logging.Level;
import java.util.logging.Logger;

public class UI extends javax.swing.JFrame {
    private String inputLine;
    static public String g;

    public UI() {
        initComponents();
        automatico a= new automatico();
        a.start();
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated
Code">//GEN-BEGIN: initComponents
    private void initComponents() {

        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
        vel = new javax.swing.JLabel();
        temp = new javax.swing.JLabel();
        gas = new javax.swing.JLabel();
        jLabel8 = new javax.swing.JLabel();
        jLabel5 = new javax.swing.JLabel();
        jLabel6 = new javax.swing.JLabel();
        jLabel7 = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jLabel1.setFont(new java.awt.Font("Lucida Grande", 3, 14));
        // NOI18N
        jLabel1.setText("SISTEMA DE TELEMETRÍA SAE");

        jLabel2.setFont(new java.awt.Font("Lucida Grande", 1, 18));
        // NOI18N
        jLabel2.setForeground(new java.awt.Color(0, 51, 255));
        jLabel2.setText("VELOCIDAD");

        jLabel3.setFont(new java.awt.Font("Lucida Grande", 1, 18));
        // NOI18N
        jLabel3.setForeground(new java.awt.Color(255, 0, 0));
        jLabel3.setText("TEMPERATURA");

        jLabel4.setFont(new java.awt.Font("Lucida Grande", 1, 18));
        // NOI18N
        jLabel4.setForeground(new java.awt.Color(0, 153, 0));
        jLabel4.setText("COMBUSTIBLE");
```

```
        vel.setFont(new java.awt.Font("Lucida Grande", 1, 48)); //
NOI18N        vel.setForeground(new java.awt.Color(0, 51, 255));

vel.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
vel.setText("65");

        temp.setFont(new java.awt.Font("Lucida Grande", 1, 48)); //
NOI18N        temp.setForeground(new java.awt.Color(255, 0, 0));

temp.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
temp.setText("23");

        gas.setFont(new java.awt.Font("Lucida Grande", 1, 48)); //
NOI18N        gas.setForeground(new java.awt.Color(0, 153, 0));

gas.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
gas.setText("80");

        jLabel8.setText("Diseñado por: Goncalves/Maroun");

        jLabel5.setFont(new java.awt.Font("Lucida Grande", 1, 24));
// NOI18N        jLabel5.setForeground(new java.awt.Color(0, 51, 255));
jLabel5.setText("km/h");

        jLabel6.setFont(new java.awt.Font("Lucida Grande", 1, 24));
// NOI18N        jLabel6.setForeground(new java.awt.Color(255, 0, 0));
jLabel6.setText("°C");

        jLabel7.setFont(new java.awt.Font("Lucida Grande", 1, 24));
// NOI18N        jLabel7.setForeground(new java.awt.Color(0, 153, 0));
jLabel7.setText("%");

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignme
nt.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(62, 62, 62)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignme
nt.CENTER)
                    .addComponent(jLabel2)
```

```
        .addComponent(jLabel5))
        .addGap(61, 61, 61))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(vel,
javax.swing.GroupLayout.PREFERRED_SIZE, 158,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(28, 28, 28)))

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignme
nt.CENTER)
        .addComponent(temp,
javax.swing.GroupLayout.PREFERRED_SIZE, 143,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel3)
        .addComponent(jLabel6))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
69, Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignme
nt.CENTER)
        .addComponent(jLabel4)
        .addComponent(gas,
javax.swing.GroupLayout.PREFERRED_SIZE, 134,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel7))
        .addGap(47, 47, 47))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
        .addComponent(jLabel8)
        .addGap(26, 26, 26))
        .addGroup(layout.createSequentialGroup()
        .addGap(44, 44, 44)
        .addComponent(jLabel11)

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
        .addGroup(layout.createSequentialGroup()
        .addGap(32, 32, 32)
        .addComponent(jLabel11)
        .addGap(43, 43, 43)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignme
nt.BASELINE)
        .addComponent(jLabel12)
```

```
        .addComponent(jLabel3)
        .addComponent(jLabel4))
        .addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(vel)
        .addComponent(temp)
        .addComponent(gas))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel5)
        .addComponent(jLabel6)
        .addComponent(jLabel7))
        .addGap(18, 18, 18)
        .addComponent(jLabel8)
        .addContainerGap(17, Short.MAX_VALUE))
    );

    pack();
} // </editor-fold> // GEN-END: initComponents

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available,
stay with the default look and feel.
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {
        java.util.logging.Logger.getLogger(UI.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {
        java.util.logging.Logger.getLogger(UI.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {
```

```
java.util.logging.Logger.getLogger(UI.class.getName()).log(java.util
.logging.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(UI.class.getName()).log(java.util
.logging.Level.SEVERE, null, ex);
    }
//</editor-fold>

/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {

        new UI().setVisible(true);

    }
});
}

public class automatico extends Thread{

    public void run(){
        SerialTest t = new SerialTest();
        t.initialize();

        while (1==1){
            try {
                TimeUnit.SECONDS.sleep(1);
            } catch (InterruptedException ex) {

Logger.getLogger(UI.class.getName()).log(Level.SEVERE, null, ex);
            }

            int foo = Integer.parseInt(g);
            float tanque = Integer.parseInt(g);
            float temperatura = Integer.parseInt(g);
            float velo = Integer.parseInt(g);

            if (foo<=40){
                temperatura=(temperatura*(10/4));
                String temp1;
                temp1 = String.valueOf(temperatura);
                temp.setText(temp1);
            }
            if ((foo>40) && (foo<=69)){
                velo=(10/3)*(velo-39);
                String velo2;
                velo2 = String.valueOf(velo);
                vel.setText(velo2);
            }
            if (foo>70){
                tanque=((tanque-70)*10)/3;
                String tank;
                tank = String.valueOf(tanque);
                gas.setText(tank);
            }
        }
    }
}
```

```
    }

    }

}
// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JLabel gas;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel temp;
private javax.swing.JLabel vel;
// End of variables declaration//GEN-END:variables
}
```