



UNIVERSIDAD DEL AZUAY

FACULTAD DE CIENCIA Y TECNOLOGÍA

**ESCUELA DE INGENIERÍA EN MECÁNICA
AUTOMOTRIZ**

**ANÁLISIS DEL SISTEMA DE CONTROL DE PRESIÓN DE
NEUMÁTICOS TPMS EN LA MARCA FORD**

Trabajo de graduación previo a la obtención del título de Ingeniero Mecánico
Automotriz

Autor:

Andrés Israel Mora Castro

Director:

Efrén Esteban Fernández Palomeque

Cuenca – Ecuador

2013

DEDICATORIA

Para mis amados padres y querida mujer, por su comprensión, paciencia y porque nunca desfallecieron en los malos momentos de nuestras vidas.

AGRADECIMIENTOS

Agradezco a todos los ingenieros que me ayudaron con la realización de este trabajo. Especialmente a mi director de tesis Ing. Efrén Fernández por su colaboración y paciencia.

INDICE DE CONTENIDOS

DEDICATORIA.....	ii
AGRADECIMIENTOS.....	iii
INDICE DE CONTENIDOS.....	iv
INDICE DE FIGURAS.....	vii
INDICE DE TABLAS.....	viii
ÍNDICE DE ANEXOS.....	x
RESUMEN.....	xi
ABSTRACT.....	xii
INTRODUCCIÓN.....	1

CAPITULO I SISTEMA DE CONTROL DE PRESIÓN DE LOS NEUMÁTICOS

1.2 Ventajas y desventajas de los sistemas TPMS.....	3
1.2.1 Sistema TPMS directo.....	3
1.2.2 Sistema TPMS indirecto.....	4

CAPITULO II CONSTITUCIÓN DEL SISTEMA DE CONTROL DE PRESIÓN

2.1 Sensores de presión de neumáticos.....	7
2.2 Antena para vigilancia de la presión de los neumáticos.....	7
2.3 Unidad de control para vigilancia de la presión de los neumáticos.....	8

CAPITULO III CALIBRACIÓN DEL SISTEMA TPMS EN LA MARCA FORD

3.2	Funcionamiento y trabajo de los equipos de diagnostico.	12
3.2.1	La capacidad de auditar el vehículo antes de que se produzca un daño en el sensor.....	12
3.2.3	Soportar el reemplazo de sensores TPMS y la reprogramación del vehículo.	12
3.2.4	Software actualizable.....	12
3.3	Proceso de desmontaje de los sensores TPMS del neumático.....	13
3.4	Instalación y calibración de los sensores TPMS.....	13
3.4.1	Instalación de los sensores TPMS.....	13
3.4.2	Calibración de los sensores TPMS.....	15

CAPITULO IV SIMULACIÓN DEL ESQUEMA ELECTRÓNICO DEL SISTEMA TPMS

4.1	Componentes del esquema electrónico del sistema de control de presión de los neumáticos	20
4.1.1	ATMEGA 8.....	20
4.1.1.1	Características del ATMEGA 8.	20
4.1.1.2	Descripción de pines (Figura 4.1).....	22
4.1.2	LCD 2X16.	24
4.1.2.1	Descripción:	25
4.1.2.2	Características principales:.....	25
4.1.3	Sistema TX/RX.	25
4.1.3.1	Transmisor de radiofrecuencia de 433MHz.....	25
4.1.3.1.1	Especificaciones técnicas:.....	26
4.1.3.1.2	Características.	26
4.1.3.1.3	Descripción de pines (figura 4.3).....	26
4.1.3.1.4	Requisitos para la antena.	27
4.1.3.2	Receptor de radiofrecuencia de 433MHz.....	28
4.1.3.2.1	Especificaciones técnicas.....	29
4.1.3.2.2	Características.	29
4.1.3.2.3	Descripción de pines.....	29
4.1.4	Sensores de presión.	30
4.1.5	Regulador 7805.	30
4.1.6	Condensadores de 470uF 50V.	31
4.2	Diseño del esquema electrónico del sistema TPMS.	31
4.2.1	Funcionamiento del esquema electrónico del sistema TPMS.....	32
4.2.2	Código realizado en codevision AVR para el circuito visualizador.....	33

4.2.3 Código fuente para la rueda delantera izquierda.	35
4.3 Simulación del circuito electrónico.	36
4.3.1 Funcionamiento.	36
4.3.2 Código fuente para la simulación de los 4 sensores.....	36
CONCLUSIONES Y RECOMENDACIONES.....	40
BIBLIOGRAFIA.....	41
REFERENCIAS BIBLIOGRÁFICAS:.....	41
INDICE DE ANEXOS.....	43

INDICE DE FIGURAS

Figura 1.1: Símbolo oficial del sistema TPMS.....	3
Figura 1.2: Símbolo parte superior del sistema TPMS.....	3
Figura 2.1: Partes del sistema TPMS.....	6
Figura 2.2: Sensor de presión.....	7
Figura 2.3: unidad de control TPMS.....	8
Figura 3.1: Kit de servicio.....	11
Figura 3.2: Colocación del sensor en el vástago.....	14
Figura 3.3: Colocación del conjunto del sensor en la llanta.....	14
Figura 3.4: colocación de la tuerca.....	14
Figura 3.5: Apriete de la tuerca.....	15
Figura 3.6: Herramienta FORDTRIGGER.....	15
Figura 3.7: Centro de mensaje.....	16
Figura 3.8: Reiniciación del sensor.....	17
Figura 3.9: Entrenar el siguiente sensor.....	17
Figura 3.10: Calibración completada.....	18
Figura 4.1: ATMEGA 8.....	23
Figura 4.2: Transmisor de radiofrecuencia de 433 MHz.....	27
Figura 4.3: Pinaje del transmisor.....	27
Figura 4.4: Tipos de antenas.....	28
Figura 4.5: Receptor de radiofrecuencia de 433MHz.....	29

Figura 4.6: Pinaje del receptor.....	30
Figura 4.7: Circuito de alimentación de 5 voltios.....	31
Figura 4.8: Circuito visualizador.....	35
Figura 4.9: circuito de la rueda.....	36
Figura 4.10: Esquema de simulación de los 4 sensores.....	39

INDICE DE TABLAS

Tabla 1.1: Ventajas y desventajas de los dos sistemas TPMS.....5
Tabla 3.1: Cuadro sobre fallas del sistema TPMS.....9

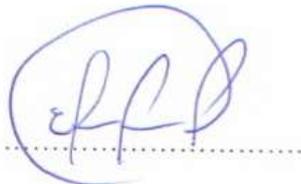
ÍNDICE DE ANEXOS

Anexo 1: Código para el circuito visualizador.....	43
Anexo 2: Código fuente para la rueda delantera izquierda.....	49
Anexo 3: Código fuente para la simulación de los 4 sensores.....	53

RESUMEN**ANÁLISIS DEL SISTEMA DE CONTROL DE PRESIÓN DE NEUMÁTICOS TPMS EN LA MARCA FORD**

En este trabajo de grado se realizó el análisis del sistema de control de presión de los neumáticos en los vehículos de la marca Ford y se desarrolló la programación y simulación del circuito electrónico del sistema TPMS. También se recopiló información sobre los componentes del sistema, su funcionamiento y los métodos de reseteo utilizados, con el objeto de identificar los parámetros de operatividad del sensor de presión como del módulo de control. Finalmente se elaboró una guía práctica para ejecutar el mantenimiento en este sistema, y la codificación de microprocesadores ATEMEGA 8 que forman parte del circuito visualizador como de los sensores integrados en las cuatro ruedas.

Palabras claves: ATEMEGA 8, reseteo, TPMS, microprocesador, radiofrecuencia.



Ing. Efrén Fernández.

Director de tesis



Mg. Mauricio Barros.

Presidente de Junta.



Andrés Mora.

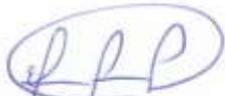
Alumno.

ABSTRACT

ANALYSIS OF TIRE PRESSURE MONITORING SYSTEM TPMS FOR FORD BRAND VEHICLES

In this work we analyzed the tire pressure monitoring control system on Ford vehicles, and developed the programming and simulation of the TPMS electronics circuit system. We also compiled information on the system components, its operation and reset methods used, in order to identify the parameters of the pressure sensor operation, and the control module. Finally we elaborated a practical service guide to run the maintenance of this system, and ATEMEGA 8 microprocessor coding, which is part of the circuit displayer, as well as the integrated sensors on all four wheels.

Keywords: ATEMEGA 8 reset, TPMS, microprocessor, Radio Frequency



Ing. Efrén Fernández
Thesis Director



Andrés Mora
Author



Mg. Mauricio Barros
President of the Board



UNIVERSIDAD DEL
AZUAY
DPTO. IDIOMAS



Translated by,
Lic. Lourdes Crespo

Andrés Israel Mora Castro

Trabajo de Graduación

Ing. Efrén Fernández Palomeque

Mayo, 2013

ANÁLISIS DEL SISTEMA DE CONTROL DE PRESIÓN DE NEUMÁTICOS TPMS EN LA MARCA FORD

INTRODUCCIÓN

Los talleres automotrices y usuarios en nuestro país desconocen la importancia del control de la presión en los neumáticos, siendo de gran importancia tanto para la eficiencia del combustible como en la duración de los neumáticos. En la actualidad la falta de conocimiento por parte de los usuarios en cuanto a las fallas de la presión en los neumáticos, genera 18,4 millones de CO₂ al año con respecto a los vehículos que tienen la presión adecuada en sus neumáticos. También se desperdicia 8 000 millones de litros de combustible y se desecha 55 millones de neumáticos por esta causa; además de significar una reducción del 20% de la vida útil del neumático, menor adherencia con la calzada, mayor riesgo de reventón del neumático y mayor distancia de frenado, según la pagina web diariomotor.com consultado el 5 de enero del 2013.

Para lograr la monitorización de presión de los neumáticos se utiliza sensores de presión los cuales emiten señales de radiofrecuencia a un módulo de control. El objetivo de la presente investigación, es realizar un análisis sobre el sistema de control de presión de los neumáticos que permita desarrollar la programación y simulación de este sistema.

CAPÍTULO I

SISTEMA DE CONTROL DE PRESIÓN DE LOS NEUMÁTICOS

1.1 Descripción del sistema de control de presión de los neumáticos TPMS

Un sistema de monitorización de presión de los neumáticos (TPMS) es un sistema electrónico que sirve para verificar la correcta presión de aire en los neumáticos de un automóvil, dando un aviso al conductor sobre el exceso o bajo nivel del inflado, y en el caso de la implementación de un sistema de inflado automático, comandar el sistema neumático que permita el proceso de llenado de los neumáticos.

*“La transmisión de datos de los sensores de las ruedas hacia la unidad de control se lleva a cabo de forma inalámbrica, por radiofrecuencia (gama de alta frecuencia AF). El intercambio de información de los periféricos en el vehículo se lleva a cabo a través de la red de área de controles (CAN-Bus) para sistemas de confort”.*¹

1.1.1 Tipos de alertas del sistema

El sistema de control de presión de los neumáticos normalmente proporciona las fallas de este sistema al conductor a través de uno de los dos tipos de luces de advertencia que se encuentran en el tablero de instrumentos del vehículo.

“El primero utiliza el símbolo oficial del TPMS, que es una sección transversal de un neumático con un signo de exclamación en el centro (figura 1.1).

¹WEBCUCSUR. Control de presión. [en línea]

<http://148.202.105.18/webcucsur/sites/default/files/Control%20autom%C3%A1tico%20de%20presi>
[consultado 20 de enero del 2013]



Figura 1.1: Símbolo oficial del sistema TPMS
(WEBCUCSUR. Control de presión)

Fuente: <http://www.discounttire.com/dtcs/infoTPMSArticle.do>. Consultado 20 de Enero 2013

La segunda es una imagen gráfica vista de la parte superior de un automóvil que indica que los neumáticos están bajos (figura 1.2)..²

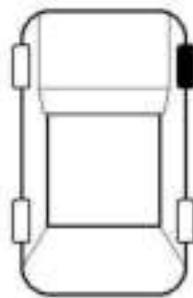


Figura 1.2: Símbolo parte superior del sistema TPMS
(DISCOUNTTIRE. Sistema TPMS)

Fuente: <http://www.discounttire.com/dtcs/infoTPMSArticle.do>. [Consultado 20 de Enero 2013]

1.2 Ventajas y desventajas de los sistemas TPMS

1.2.1 Sistema TPMS directo

Este sistema avisa al conductor sobre cualquier pérdida de presión en los neumáticos del vehículo. El sistema funciona con sensores situados en el interior de los neumáticos los cuales emiten señales de radiofrecuencia a un modulo en el interior del vehículo el cual es el responsable de la acumulación de los datos.

Entre las desventajas del sistema se enumeran los problemas prácticos de uso. Cada vez que se sustituyen los neumáticos, se debe informar al técnico que el vehículo cuenta con los sensores de presión en las ruedas, dado que éstos suelen dañarse sobre todo al remover las ruedas en los talleres de vulcanización. Además se debe agregar el costo de calibración del sistema TPMS en caso de cualquier cambio de las

²DISCOUNTTIRE. Sistema TPMS. [en línea] <http://www.discounttire.com/dtcs/infoTPMSArticle.do>
[consultado 20 de enero del 2013]

condiciones básicas (un sensor nuevo). También se tiene que calibrar y cargar nuevas condiciones básicas siempre que se reemplace los neumáticos. Se debe recordar que las llantas deben montarse en el mismo lugar que ocupaban anteriormente.

1.2.2 Sistema TPMS indirecto

El sistema TPMS indirecto es el más optado por los fabricantes de los vehículos. Su mayor ventaja es el bajo costo de implementación, su funcionamiento es por medio de unos sensores de velocidad, ubicados en las ruedas, que forman parte del sistema de frenado antibloqueo (ABS) y del sistema electrónico de control de estabilidad (ESC).

Los sistemas TPMS indirectos comparan la velocidad entre cada rueda del vehículo, detectando la presión baja en aquella que realiza mayor número de giros. Esto se debe al hecho de que el diámetro de un neumático con presión inferior es menor, por lo que al recorrer la misma distancia este neumático necesitará hacer más giros.

La mayor desventaja del sistema TPMS indirecto es la calibración de los neumáticos que deben hacerse de forma manual. Cada vez que se sustituyan o se aumente la presión de aire en los neumáticos el sistema debe calibrarse. La intención de los creadores de este sistema se basaba en su calibración en el momento en que la presión en cada rueda esté al mismo nivel (el recomendado por el fabricante).

El punto de referencia no es la presión correcta para las ruedas (no hay sensores de presión), sino el momento en que decida el técnico que la presión ya está a un nivel apropiado. El sistema aceptará la calibración que ponga el técnico o el propietario del vehículo, no informará que la presión en los neumáticos es el adecuado.

El botón de calibración puede usarse en cualquier momento que se considere apropiado, porque este sistema TPMS no está analizando el estado real anterior a la calibración. El sistema alertará al conductor únicamente en caso de una caída de presión de un 20 por ciento con respecto al nivel base, establecido por el técnico o propietario.

Otro problema es el momento de sustituir uno o más neumáticos. El sistema está perfectamente calibrado para el nivel de rigidez que presentaban los neumáticos al salir de la fábrica. Esto significa que, si se desea que el sistema avise de una pérdida de presión a nivel de un 20 por ciento, se deberá montar el mismo modelo de neumáticos que el predeterminado de fábrica.

El siguiente aspecto a considerar a la hora de utilizar los neumáticos con el sistema TPMS indirecto es el tiempo de respuesta. De acuerdo a las investigaciones realizadas por el Departamento de Transporte Estadounidense (DOT), al detectar la pérdida de presión en una de las ruedas, el conductor necesita aproximadamente 10 minutos para llegar a un lugar seguro.

*“El sistema TPMS indirecto, detectará la pérdida gradual de presión apenas tras recorrer unos cuantos kilómetros (porque analiza la presión comparando el comportamiento de los neumáticos en un determinado lapso de tiempo). En consecuencia, se perderá unos minutos muy valiosos (en el transcurso de los cuales la presión de inflado podrá haber llegado a cero). En total, al comparar el rendimiento de las ruedas, este sistema no será capaz de detectar una pérdida gradual de presión en todas las ruedas por el simple hecho de no disponer de un punto de referencia”.*³

TPMS DIRECTO	TPMS INDIRECTO	VENTAJAS	DESVENTAJAS
	X	El sistema es más económico.	
	X		Tiempo de advertencia deficiente.
X		Alerta exacta.	
	X		Calibración manual.
X			Costo de calibrar el sistema.
X			Sensores caros.

Tabla 1.1 ventajas y desventajas de los dos sistemas TPMS.

³ OPONEO. Sensores de presión. [En línea] <http://www.oponeo.es/articulo/sensores-de-presion-en-los-neumaticos-sistema-tpms> [consulta: 1 de febrero del 2013]

CAPÍTULO II

CONSTITUCIÓN DEL SISTEMA DE CONTROL DE PRESIÓN

El sistema TPMS se instala en los vehículos de la marca Ford para proporcionar una advertencia al conductor sobre una baja presión en cualquiera de las cuatro o cinco llantas del vehículo (cinco si el sistema TPMS está montado en la llanta de emergencia).

Los componentes de un sistema de control de presión de los neumáticos de la marca Ford (figura 2.1) consiste en:

1. Unos sensores de control de presión de neumáticos.
2. Una luz de advertencia del TPMS.
3. Un monitor de presión de los neumáticos.
4. Una antena TPMS.

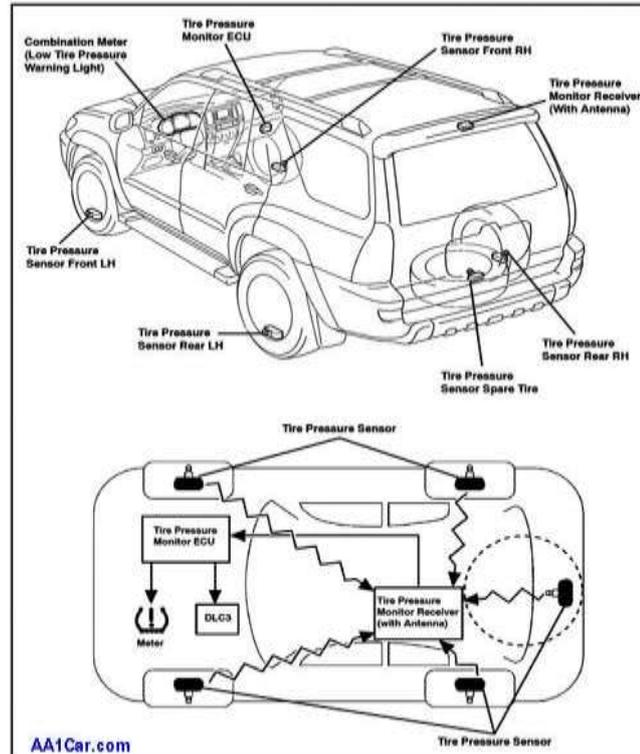


Figura 2.1: Partes del sistema TPMS
(AA1Car. Sistemas TPMS)

Fuente: http://www.aa1car.com/library/tire_monitors.htm [Consultado 20 de marzo de 2013]

2.1 Sensores de presión de neumáticos

Los sensores de presión en la marca Ford (figura 2.2) detectan las presiones de los neumáticos y transmiten estos datos junto con la identificación única de los TPMS al módulo de control del sistema. Estos sensores por medio de un tornillo van sujetos al vástago de la válvula y se pueden volver a utilizar en caso de sustituirse el neumático o llanta.

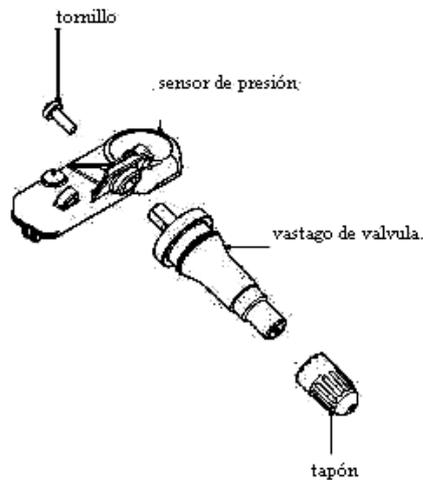


Figura 2.2: Sensor de presión.
(FORD. Sensores de presión)

Fuente: <http://www.fordparts.com> [Consultado 20 de marzo de 2013]

2.2 Antena para vigilancia de la presión de los neumáticos

Las antenas reciben las señales de radiofrecuencia que emiten los sensores de presión de los neumáticos y la transmiten a la unidad de control para el monitoreo de la presión de los neumáticos. Las señales de radiofrecuencia se filtran y se seleccionan en la unidad de control, con objeto de que sean procesados los datos que efectivamente provienen.

Las antenas transmisoras de los sensores de presión emiten la siguiente información:

- Código de identificación individual (código ID).
- Presión absoluta actual del neumático.
- Temperatura actual del aire en el neumático.
- Estado de la batería integrada.

- Transmisión fiable de los datos, la información necesaria sobre el estado, sincronización y gestión.

2.3 Unidad de control para vigilancia de la presión de los neumáticos

“La unidad de control (figura 2.3) para vigilancia de la presión de los neumáticos analiza las señales de radiofrecuencia procedentes de las antenas del sistema, las cataloga por prioridades y transmite la información correspondiente al cuadro de instrumentos. Allí se visualizan los avisos a través de la pantalla del sistema para información del conductor”.⁴



Figura 2.3: unidad de control TPMS.
(EBAY. Módulo de control de Ford)

Fuente: <http://www.ebay.com/itm/11-FIESTA-TPMS-MODULE-/220988154446> [Consultado 20 de mayo de 2013]

⁴WEBCUCSUR.Unidad de control. [en línea]

<http://148.202.105.18/webcucsur/sites/default/files/Control%20autom%C3%A1tico%20de%20presi.>
[consultado 15 de mayo del 2013]

CAPÍTULO III

CALIBRACIÓN DEL SISTEMA TPMS EN LA MARCA FORD

3.1 Mantenimiento del sistema TPMS

Para dar un adecuado mantenimiento al sistema de monitorización de presión de los neumáticos se debe primero proseguir con la inspección del sistema y como primer paso se debe colocar el encendido en la posición ON y buscar en el tablero un símbolo de advertencia del sistema TPMS. Esta advertencia se debe encender tras dar el contacto y se apagará después de sensar las señales de los sensores de presión de los neumáticos. Si la luz de advertencia permanece encendida, entonces hay un problema en los sensores de presión de los neumáticos. Para reconocer la falla mediante la luz del tablero se debe guiar por medio de la siguiente tabla 3.1.

LUZ DE ADVERTENCIA DE BAJA PRESIÓN EN LOS NEUMÁTICOS	POSIBLE CAUSA	ACCIÓN REQUERIDA
Luz fija de advertencia	Bajo presión en el(los) neumático(s).	Revisar la presión de los neumáticos para confirmar que los neumáticos están inflados apropiadamente.
		Después de inflar los neumáticos con la presión recomendada por el fabricante, como se muestra en la Etiqueta del Neumático (localizada en la orilla de la puerta del conductor o en el

		Pilar-B), el vehículo debe ser conducido por lo menos a 20 mph (32 km/h) antes de que la luz se apague.
	Neumático de repuesto en uso	Su neumático de repuesto está en uso. Repare la rueda / el neumático y reinstálelo en el vehículo para restaurar la funcionalidad del sistema.
	Funcionamiento del TPMS.	Si los neumáticos están inflados apropiadamente, su neumático de repuesto no está en uso y la luz permanece encendida, comunicarse con su concesionario autorizado, tan pronto como sea posible.
Parpadeo de la Luz de Advertencia	El neumático de repuesto está en uso.	Su neumático de repuesto está en uso. Repare el neumático dañado y móntelo nuevamente en el vehículo para restaurar el funcionamiento del sistema.
	Falla del TPMS.	Si sus neumáticos está inflados apropiadamente , su neumático de repuesto no está en uso y la luz de advertencia del TPMS parpadea, entre en contacto con su concesionario autorizado, tan pronto como sea posible.

Tabla 3.1: Cuadro sobre fallas del sistema TPMS.
(FORD. Manual del propietario)

Fuente. http://www.forcenter.cl/assets/files/manuales/Ford_Edge_Manual_del_Propietario.pdf
pag 274 [consultado 20 de febrero del 2013]

En el siguiente paso se debe dar una inspección visual de los posibles daños de las válvulas de los sensores, después verificar si se encuentran en su posición las tapas de las válvulas y por último se debe controlar que cada neumático se encuentre con la correcta presión.

Después de la inspección del sistema se debe dar un mantenimiento preventivo a los sensores de presión, esto consta en verificar si el vástago de la válvula no presenta corrosión, el desgaste que produce la corrosión puede provocar que los componentes de aluminio se rompan, agrieten, haciendo al sensor TPMS inútil. La corrosión también puede causar la pérdida de aire, dando lugar a frecuentes advertencias de los neumáticos con baja presión. *“Para prevenir que la corrosión dañe el sistema se debe sustituir el kit de servicio (figura 3.1) cada vez que se cambia el neumático, esto aumentará la vida útil del sensor”*.⁵



Figura 3.1: Kit de servicio.

(TPMS DIRECT. Prevent TPMS sensor corrosion)

Fuente: http://www.tpmsdirect.com/TPMS_Corrosion_Prevention_a/743.htm [Consultado 20 de marzo 2013]

⁵TPMS DIRECT. Prevent TPMS sensor corrosion. [en línea]

http://www.tpmsdirect.com/TPMS_Corrosion_Prevention_a/743.htm [Consultado 20 de marzo 2013]

3.2 Funcionamiento y trabajo de los equipos de diagnostico

Una buena herramienta de diagnostico TPMS deberá tener las siguientes características:

3.2.1 La capacidad de auditar el vehículo antes de que se produzca un daño en el sensor

La herramienta debe activar y leer los datos de cada sensor TPMS. Esta herramienta necesita conocer todos los patrones de baja frecuencia de transmisión para activar los sensores de cada combinación de marca/modelo/año. La herramienta necesita comprobar si cada sensor está funcionando o no. Por último la herramienta debería decodificar la señal y ser capaz de variar su potencia de activación de acuerdo al sensor TPMS instalado. Para hacer esto, necesitará conocer todos los algoritmos de decodificación para todos los distintos sensores en el mercado.

La herramienta TPMS debe ser capaz de decodificar la transmisión del sensor y mostrar la información. Esto permitirá al técnico establecer si el sensor ha fallado o no está ajustado; si el puerto de presión en el sensor está bloqueado, si la pieza está funcionando a una temperatura demasiado alta, y otras informaciones de diagnóstico que pueden estar incluidas en los datos específicos del sensor (por ejemplo, si el sensor TPMS tiene batería baja).

3.2.3 Soportar el reemplazo de sensores TPMS y la reprogramación del vehículo

Si un sensor resulta estar defectuoso o dañado, necesitará ser reemplazado. Esto requiere la instalación de un nuevo sensor, y la reprogramación de la ECU con su identificador único y su posición en el vehículo. La herramienta también debe ser capaz de apagar la luz de advertencia del sistema TPMS.

3.2.4 Software actualizable

Nuevas variaciones del sistema TPMS están surgiendo a un ritmo cada vez más rápido. *“Es de vital importancia que la herramienta pueda ser actualizada utilizando un cable USB o similar a través de un PC con las últimas variantes del sistema*

*TPMS y otros atributos necesarios para continuar el mantenimiento de nuevos modelos*⁶.

3.3 Proceso de desmontaje de los sensores TPMS del neumático

“Para el proceso de desmontaje del sensor de presión se debe realizar los siguientes pasos:

- 1) Retirar la rueda del vehículo.*
- 2) Extraer la tapa del vástago de la válvula.*
- 3) Retirar la válvula.*
- 4) Extraer la tuerca del vástago de la válvula.*
- 5) Permitir que el sensor caiga en la rueda (esto evitará daños en el sensor mientras el neumático se retira de la rueda con una máquina de extracción de neumáticos).*
- 6) Desmontar el neumático.*
- 7) Recuperar el sensor TPMS de la rueda”.*⁷

3.4 Instalación y calibración de los sensores TPMS

3.4.1 Instalación de los sensores TPMS

Par la instalación del sensor de presión del neumático se debe seguir los siguientes pasos:

- 1) Insertar el tornillo en el orificio ranurado del sensor (figura 3.2).
- 2) Montar la válvula del tornillo (gire el vástago de la válvula 3 a 4 vueltas completas).

⁶BARTECUSA. Sensores de presión. [en línea] <http://www.bartecusa.com/espanol-mexicano/sensores-de-presion-de-neumaticos.html> [consultado 4 de marzo del 2013]

⁷ MAZDA FORUM. Desmontaje de sensores. [en línea] <http://www.mazdas247.com/forum/archive/index.php/t-78852.html> [consultado 26 de febrero del 2013]

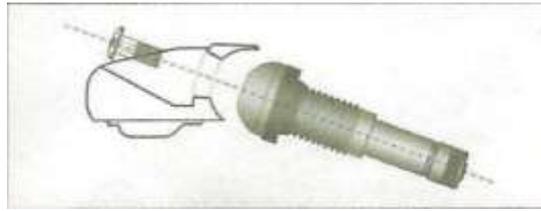


Figura 3.2: Colocación del sensor en el vástago.
(TPMS DIRECT. Montaje del sensor TPMS)

Fuente:http://www.tpmsdirect.com/Adjustable_Angle_Sensor_Mounting_Guidelines_a/647.htm
[Consultado 26 de febrero2013]

3) Deslice el vástago de la válvula a través del orificio de la válvula de la llanta (figura 3.3).

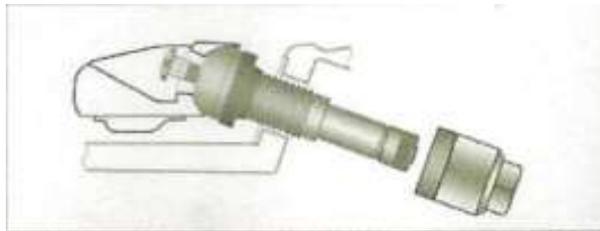


Figura 3.3: Colocación del conjunto del sensor en la llanta
(TPMS DIRECT. Montaje del sensor TPMS)

Fuente:http://www.tpmsdirect.com/Adjustable_Angle_Sensor_Mounting_Guidelines_a/647.htm
[Consultado 26 de febrero2013]

4) Inserte la tuerca sobre el vástago de la válvula (figura 3.4).

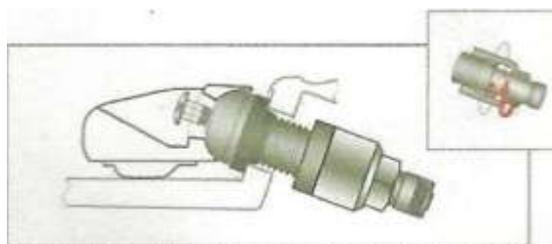


Figura 3.4: colocación de la tuerca.

(TPMS DIRECT. Montaje del sensor TPMS)

Fuente:http://www.tpmsdirect.com/Adjustable_Angle_Sensor_Mounting_Guidelines_a/647.htm
m [Consultado 26 de febrero2013]

5) Apriete la tuerca con el par final (figura 3.5). Siga apretando la tuerca con el par final (35 lbf.in o 4.0Nm)

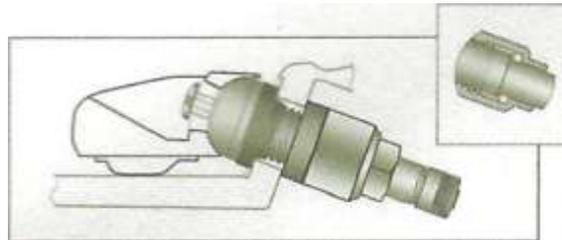


Figura 3.5: Apriete de la tuerca.
(TPMS DIRECT. Montaje del sensor TPMS)

Fuente: http://www.tpmsdirect.com/Adjustable_Angle_Sensor_Mounting_Guidelines_a/647.htm [Consultado 26 de enero de 2013]

3.4.2 Calibración de los sensores TPMS

Para calibrar los sensores TMS Ford se debe usar la herramienta de activación Fordtrigger (figura 3.6)



Figura 3.6: Herramienta fordtrigger.
(TPMS DIRECT. Recetado para vehículos ford)

Fuente: <http://tpmsdirect.wordpress.com/2012/02/28/how-to-reset-ford-tpms-sensors/> [Consultado 3 de enero de 2013]

Antes de empezar con el procedimiento, se debe asegurar de que los neumáticos tengan la presión adecuada de aire.

Para obtener los sensores en el modo de aprendizaje. Se debe seguir esta secuencia de encendido y pagado del *switch* (interruptor):

- 1) Comenzar con la llave en posición de apagado, pulsar y soltar el pedal del freno.
- 2) Girar la llave a la posición ON (no arrancar) y después del primer timbre cerrar el *switch*, se realiza dos veces consecutivas.
- 3) Girar la llave a la posición ON (no arrancar), pulsar y soltar el pedal de freno y girar la llave a la posición de apagado.
- 4) Girar la llave a la posición ON (no arrancar) y después de la primera campanada girar la llave a la posición de apagado. Este paso se realiza dos veces consecutivas.
- 5) Girar la llave a la posición ON (no arrancar). El claxon sonará una vez y el centro de mensajes mostrará el siguiente mensaje (figura 3.7).



Figura 3.7: Centro de mensaje.
(TPMS DIRECT. Receteado para vehículos ford)
Fuente: <http://tpmsdirect.wordpress.com/2012/02/28/how-to-reset-ford-tpms-sensors/>
[Consultado 3 de enero 2013]

- 6) Llevar la herramienta Fordtrigger hacia el sensor y mantenerla frente al vástago de la válvula y apuntar al sensor en el interior de la llanta y pulse el botón (figura 3.8) En aproximadamente 5 segundos, el sensor se reiniciara, el claxon sonará y su centro de mensajes le avisará para entrenar al siguiente sensor (figura 3.9).



Figura 3.8: Reiniciación del sensor.
(TPMS DIRECT. Receteado para vehículos Ford)

Fuente: <http://tpmsdirect.wordpress.com/2012/02/28/how-to-reset-ford-tpms-sensors/>
[Consultado 3 de enero de 2013]



Figura 3.9: Entrenar el siguiente sensor.
(TPMS DIRECT. Receteado para vehículos ford)

Fuente: <http://tpmsdirect.wordpress.com/2012/02/28/how-to-reset-ford-tpms-sensors/>
[Consultado 3 de enero de 2013]

Hacer lo mismo para la rueda RR y la rueda LR. Después de la LR se lleva a cabo el centro de mensajes mostrará lo siguiente (figura 3.10).



Figura 3.10: Calibración completada.
(TPMS DIRECT. Receteado para vehículos ford)

Fuente: <http://tpmsdirect.wordpress.com/2012/02/28/how-to-reset-ford-tpms-sensors/>
[Consultado 3 de enero de 2013]

Si la formación no se realiza correctamente, el claxon sonará dos veces y se debe repetir el proceso de formación.

Este proceso se tarda menos de 2 minutos.

Si su vehículo tiene botón de encendido:

- 1) Verificar que la presión de los neumáticos sea la correcta.
- 2) Asegurarse de que el botón de encendido está en OFF, luego presionar y soltar el pedal del freno.
- 3) Mantener pulsado el botón de encendido durante 2 segundos, soltar y después pulsar el botón una vez. El vehículo debe estar en OFF. Mantener pulsado el botón de encendido hacia abajo por dos segundos, soltar y a continuación pulsar el botón una vez.
- 4) Mantener pulsado el botón de encendido durante dos segundos más. El vehículo debe estar en modo ON.
- 5) Presionar y soltar el pedal del freno.

- 6) Pulsar el botón de encendido una vez que el vehículo está apagado.
- 7) Mantenga pulsado el botón de encendido durante 2 segundos, soltar y después pulsar el botón una vez. El vehículo debe estar en OFF. Mantener pulsado el botón de encendido hacia abajo por dos segundos, soltar y a continuación pulsar el botón una vez.
- 8) Mantener pulsado el botón de encendido durante dos segundos más. El vehículo debe estar en modo ON.
- 9) El claxon debe sonar, esto significa que el vehículo ha entrado en el modo de aprender de nuevo.
- 10) Colocar la parte superior de la herramienta FORDTRIGGER en la pared lateral del neumático del lado del conductor delante en la ubicación del vástago de la válvula.
- 11) Pulsar el botón de la herramienta FORDTRIGGER. El claxon del vehículo sonará significando que el sensor ha aprendido.
- 12) Una vez que el claxon suena, ir al neumático delantero del pasajero y repetir el proceso. Los neumáticos sólo deben tomar 30 segundos o menos.
- 13) Después de que el neumático delantero del lado del pasajero ha aprendido, continuar con el neumático trasero del lado del pasajero y terminar con el neumático trasero del lado del conductor.
- 14) Girar el encendido a OFF. Si el claxon no suena, los sensores han aprendido con éxito. Si el sonido del claxon es dos veces, entonces el proceso debe ser repetido porque hubo un fallo de funcionamiento.

Cuando los sensores son nuevos se realizan los mismos pasos anteriormente mencionados.

CAPITULO IV

SIMULACIÓN DEL ESQUEMA ELECTRÓNICO DEL SISTEMA TPMS

Se realizara un esquema electrónico del sistema de control de presión de los neumáticos para un vehículo de la marca Ford, que permita simular la caída de presión de cada uno de los neumáticos, cuyos datos se visualizaran en un cristal LCD.

4.1 Componentes del esquema electrónico del sistema de control de presión de los neumáticos

Los materiales que se van a utilizar para el esquema electrónico del sistema TPMS son los siguientes:

4.1.1 ATMEGA 8

El Atmega 8 es un microcontrolador de la familia Atmel, contiene una arquitectura tipo RISC (*Reduced Instruction Set Computing*), sus instrucciones se ejecutan en solo un ciclo de máquina, su estructura elimina la necesidad de usar componentes externos ya que internamente tiene: osciladores, USART, SPI, resistencias pull-up, modulación por anchura de pulso (PWM), convertidor ADC, comparador analógico y cronómetros, éstas son solo algunas de las características que se encuentran en este dispositivo. Su programación es posible realizarla en Lenguaje C, assembly o Basic. Todo su sistema está depositado en una sola pastilla.

4.1.1.1 Características del ATMEGA 8

“El Atmega 8 tiene las siguientes características:

- 1. Utiliza arquitectura RISC*
- 2. 130 instrucciones, las más simples se ejecutan en un solo ciclo de reloj*

3. *8 kbytes de Memoria Flash*
 - *ISP: Programable en sistema.*
 - *Capacidad Read-While-Write, (Lee mientras escribe). Duración: 1 000 ciclos de escritura/borrado.*
4. *512 bytes de memoria EEPROM.*
 - *Duración: 100 000 ciclos de escritura/lectura.*
5. *1 kbyte de memoria SRAM.*
6. *Seguro programable para la seguridad del software.*
7. *32 x 8 registros de trabajo de propósito general.*
8. *23 líneas de entrada/salida de propósito general.*
9. *USART (Universal Synchronous Asynchronous Receiver Transmitter) programable.*
10. *Reloj en tiempo real.*
11. *Interfaz serial SPI.*
12. *Vcc: 4.5 – 5.5 V*
13. *Frecuencia de funcionamiento 0 – 16 MHz.*
14. *3 timer/counter.*
15. *Interrupciones internas y externas.*
16. *Interfaz serial 2-wire.*
17. *6 canales ADC, donde 4 tienen 10 bits de exactitud, y 2 tienen 8 bits de exactitud.*
18. *1 Timer Watchdog programable con oscilador interno.*
19. *5 modos de ahorro de energía, seleccionado por software*⁸.

El microcontrolador AVR tiene y combina un amplio sistema de instrucciones, con 32 registros de propósito general. Las operaciones de la ALU entre registros son ejecutadas en el archivo de registro. Las operaciones son divididas en tres categorías: aritméticas, lógicas y funciones de bit. La arquitectura que resulta es más eficiente, y alcanza rendimientos de procesamiento de hasta diez veces más rápido que los microcontroladores con tecnología CISC.

⁸ATMEL CORPORATION. Microcontroller with 8 kbytes In-System Programmable Flash, Advance Information, Atmel Corporation, 2001, pag 1 [En línea] <http://www.atmel.com/images/doc1919.pdf> [consultado 16 de abril del 2013]

“La Memoria Flash ISP (Programable en sistema) permite que la memoria de programa del chip sea reprogramada en el propio sistema a través de la interfaz SPI o mediante un programador de memoria convencional. Combinando una tecnología RISC de 8 bits con un CPU con memoria Flash, el Atmega 8 resulta un poderoso microcontrolador que proporciona una alta flexibilidad en los diseños, a bajo costo aportando una solución muy efectiva para la mayoría de las aplicaciones de control. El Atmega 8 está apoyado por un completo juego de programas y sistemas de desarrollo, incluyendo: compiladores, ensambladores, simuladores, emuladores en circuito, y kits de evaluación”⁹

4.1.1.2. Descripción de pines (Figura 4.1)

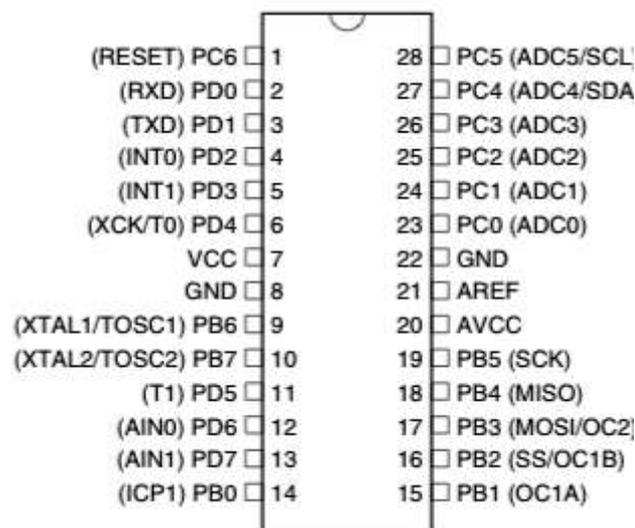


Figura 4.1: ATMEGA 8.

(MIGUEL GRASSI. Pines del ATMEGA 8)

Fuente: [http://www.miguelgrassi.com.ar/arduino/Arduino-To-Atmega8-Pins\[1\].png](http://www.miguelgrassi.com.ar/arduino/Arduino-To-Atmega8-Pins[1].png) fecha de consulta [5 de mayo de 2013]

1. Vcc:

Fuente de voltaje digital

2. GND.

Tierra de la fuente de voltaje digital.

⁹ GORDON CHRISTIAN. Modulo didáctico para el microcontrolador ATMEGA8 [en línea] <http://bibdigital.epn.edu.ec/bitstream/15000/1955/1/CD-0193.pdf> pag 15 [consultado 1 de mayo del 2013]

3. RESET.

Entrada de RESET. Se genera un reset aplicando un nivel bajo de voltaje en este pin por un periodo de tiempo más largo que un ciclo de máquina, incluso si el reloj no está funcionando. Pulsos más cortos no generan un estado de reset.

4. XTAL 1.

Entrada al amplificador inversor del oscilador y entrada al circuito de operación interno del reloj.

5. XTAL2.

Salida del amplificador inversor del oscilador.

6. AVCC.

Este es un pin de fuente de voltaje para el Puerto A y sus ADCs (Conversores Analógico-Digitales). Si sus ADCs no son utilizados, este pin debe ser conectado a Vcc. Si sus ADCs sí son utilizados, este pin debe ser conectado a Vcc pero mediante un filtro pasa-bajos.

7. AGND.

Tierra analógica. Si el tablero presenta diferentes tierras, se debe conectarlas a este pin. En otro caso se conecta a GND.

8. AREF.

Este es el pin de referencia analógica para los ADC.

9. Puerto B.

“El puerto B es un puerto de entrada/salida bi-direccional de 8 bits, con resistencias internas de pull-up. Como salidas, los pines del puerto B tienen limitada la corriente de salida, si las resistencias pull-up están activadas. Los pines del puerto B tienen tres estados cuando la condición de reset llega a activarse, incluso si el reloj no está funcionando”¹⁰

Dependiendo de la configuración de las opciones de reloj, el pin PB6 puede ser usado como entrada al amplificador inversor del oscilador y entrada al circuito de operación interno del reloj.

Dependiendo de la configuración de las opciones de reloj, el pin PB7 puede ser usado como salida del amplificador inversor del oscilador.

¹⁰ATMEL CORPORATION. Microcontroller with 8 kbytes In-System Programmable Flash, Advance Information, Atmel Corporation, 2001, pag 4 [En línea] <http://www.atmel.com/images/doc1919.pdf> [consultado 16 de abril del 2013]

10. Puerto C / RESET.

“El puerto C es un puerto de entrada/salida bi-direccional de 8 bits, con resistencias internas de pull-up. Como salidas, los pines del puerto C tienen limitada la corriente de salida, si las resistencias pull-up están activadas.

Los pines del puerto C tienen tres estados cuando la condición de reset llega a activarse, incluso si el reloj no está funcionando”¹¹

PC6 es usado como entrada de Reset. Un nivel bajo de voltaje en este pin por un periodo de tiempo aproximado de 1.5 μ s generará un reset, incluso si el reloj no está funcionando. Pulsos más cortos no generan un estado de reset.

11. Puerto D.

“El puerto D es un puerto de entrada/salida bi-direccional de 8 bits, con resistencias internas de pull-up. Como salidas, los pines del puerto D tienen limitada la corriente de salida, si las resistencias pull-up están activadas.

Los pines del puerto D tienen tres estados cuando la condición de reset llega a activarse, incluso si el reloj no está funcionando”¹².

4.1.2 LCD 2X16

Las Pantallas de Cristal Líquido (LCD del inglés, *Liquid Cristal Display*), están altamente difundidas en la actualidad. Son muy útiles porque permiten mostrar información o datos de manera muy clara.

La magia de los LCD se debe a los “cristales líquidos”. En sí estas dos palabras suenan contradictorias, pero este material es la razón por la cual este dispositivo funciona.

¹¹ ATMEL CORPORATION. Microcontroller with 8 kbytes In-System Programmable Flash, Advance Information, Atmel Corporation, 2001, pag 5 [En línea]
<http://www.atmel.com/images/doc1919.pdf> [consultado 16 de abril del 2013]

¹² ATMEL CORPORATION. Microcontroller with 8 kbytes In-System Programmable Flash, Advance Information, Atmel Corporation, 2001, pag 5 [En línea]
<http://www.atmel.com/images/doc1919.pdf> [consultado 16 de abril del 2013]

4.1.2.1 Descripción

*“La pantalla de cristal liquido o LCD (Liquid Crystal Display) es un dispositivo controlado de visualización gráfico para la presentación de caracteres, símbolos o incluso dibujos (en algunos modelos), es este caso dispone de 2 filas de 16 caracteres cada una y cada carácter dispone de una matriz de 5x7 puntos (píxeles), aunque los hay de otro número de filas y caracteres. Este dispositivo está gobernado internamente por un microcontrolador y regula todos los parámetros de presentación, este modelo es el más comúnmente usado y esta información se basará en el manejo de este u otro LCD compatible”.*¹³

4.1.2.2 Características principales

1. Desplazamiento de los caracteres hacia la izquierda o la derecha.
2. Proporciona la dirección de la posición absoluta o relativa del carácter.
3. Memoria de 40 caracteres por línea de pantalla.
4. Movimiento del cursor y cambio de su aspecto.
5. Permite que el usuario pueda programar 8 caracteres.
6. Conexión a un procesador usando un interfaz de 4 u 8 bits.

4.1.3 Sistema TX/RX

4.1.3.1 Transmisor de radiofrecuencia de 433MHz

El módulo de radiofrecuencia de 433MHz (figura 4.2) es un transmisor de datos en UHF (*Ultra High Frequency*, ‘frecuencia ultra alta’). Al trabajar con el receptor de 433MHz conforma un sistema TX/RX el cual permite la implementación de enlaces de datos de radiofrecuencia de forma muy simple, alcanzando distancias de hasta 80 metros dentro de edificaciones o 350 metros en campo abierto cuando opera con la fuente de 12V.

¹³SCRIBD. Pantalla LCD. [en línea] <http://es.scribd.com/doc/44252680/LCD-16X2> [consultado 6 de mayo del 2013]

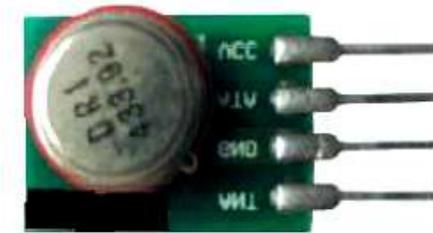


Figura 4.2: Transmisor de radiofrecuencia de 433 MHz.
(TECMIKRO. Radiofrecuencia: sistemas TX/RX a 433MHz)

Fuente: <http://www.programarpicenc.com/libro/cap15-a-radiofrecuencia-tx-rx-433mhz.html> Fecha de [consulta 6 de mayo de 2013]

4.1.3.1.1 Especificaciones técnicas

1. Señal de radiofrecuencia: Modulación ASK (Modulación por Desplazamiento de Amplitud).
2. Fuente de alimentación: 12V (también disponible en versiones de 3V y 5V).
3. Consumo de corriente: <16 mA.
4. Potencia de transmisión: 13 dBm.
5. Desviación de frecuencia: +- 75kHz.

4.1.3.1.2 Características

1. Alcance útil hasta 350 metros (12V), 230 metros (5V), 160 metros (3V).
2. Disponible en frecuencias de 433,92 MHz (433MHz) y 315.0 MHz.
3. Velocidades de transmisión hasta 20 kbps.

4.1.3.1.3 Descripción de pines (figura 4.3)

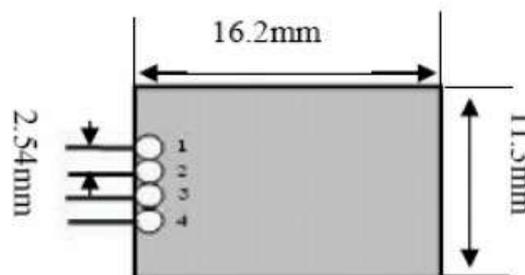


Figura 4.3: Pinaje del transmisor.
(TECMIKRO. Radiofrecuencia: sistemas TX/RX a 433MHz)

Fuente: <http://www.programarpicenc.com/libro/cap15-a-radiofrecuencia-tx-rx-433mhz.html> Fecha de [consulta 6 de mayo de 2013]

1. Vcc: Alimentación de CC regulada. Contenido máximo de rizado 0,1Vpp
2. DATA: Entrada de señal modulada. Acepta datos digitales serie a niveles de 0V a 5V.
3. GND: Referencia (tierra) del suministro de DC.
4. ANT: Entrada de 50 ohm de la antenna.

4.1.3.1.4 Requisitos para la antena

Se recomiendan y aprueban tres tipos de antena (figura 4.4).

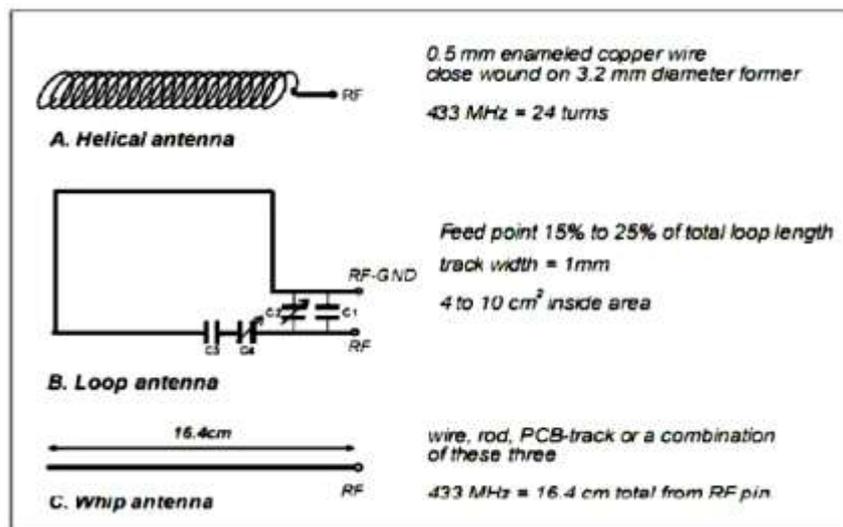


Figura 4.4: Tipos de antenas.

(TECMIKRO. Radiofrecuencia: sistemas TX/RX a 433MHz

Fuente: <http://www.programarpicenc.com/libro/cap15-a-radiofrecuencia-tx-rx-433mhz.html> Fecha de [consulta 6 de mayo de 2013]

1. Helicoidal: Bobina de alambre, conectada directamente al pin 2, circuito abierto en el otro extremo. Esta antena es muy eficiente, dado su pequeño tamaño (20 mm x 4 mm de diámetro). La antena helicoidal es una antena de alta Q, recortar la longitud del hilo o expandir la bobina para obtener resultados óptimos. Las antenas helicoidales se desintonizan mucho con la proximidad a otros objetos conductores.
2. Lazo: Un lazo de pista de PCB sintonizado por un condensador fijo o variable a tierra en el extremo "caliente" y alimentado desde el pin 2 en un punto a 20% desde el extremo de tierra. Los lazos tienen inmunidad a la desintonización por proximidad.

3. **Látigo:** Este es un alambre, varilla, pista de la placa o combinación conectada directamente a la patilla 2. La longitud total óptima es 15,5cm (1/4 de longitud de onda a 433MHz). Mantenga el extremo del circuito abierto (“caliente”) lejos de componentes de metal para evitar graves pérdidas de sintonía. Los látigos son sensibles a los planos de tierra y se beneficiarán de los radiales internos a tierra de ¼ de longitud de onda, si el producto es pequeño y está revestido de plástico.

La elección de la antena y su posición controlan directamente el alcance del sistema TX/RX. Mantenerlo libre de otro metal en el sistema, particularmente el extremo caliente. La mejor posición de lejos, es ubicarlo sobre la parte superior del producto. Esto a menudo no es deseable por razones prácticas / ergonómicas, por lo que se debe alcanzar un compromiso. Si se debe usar una antena interna, tratar de mantenerla alejada de otros componentes metálicos, en particular los grandes, como transformadores, baterías, planos de tierra y pistas en circuito impreso. El espacio alrededor de la antena es tan importante como la propia antena.

4.1.3.2 Receptor de radiofrecuencia de 433MHz

Este es un receptor miniatura de datos en UHF, permite la implementación sencilla de enlaces TX/RX inalámbricos de datos a velocidades de hasta 4,8kbps y distancias de hasta 40 metros dentro de edificios o 110 metros en campo abierto.

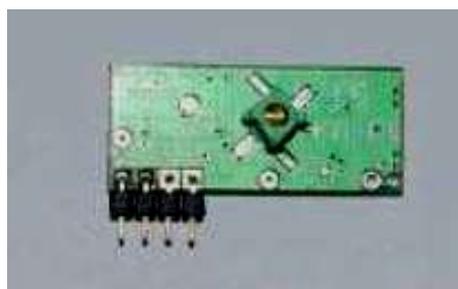


Figura 4.5: Receptor de radiofrecuencia de 433MHz.
(TECMIKRO. Radiofrecuencia: sistemas TX/RX a 433MHz)

Fuente: <http://www.programarpicenc.com/libro/cap15-a-radiofrecuencia-tx-rx-433mhz.html> Fecha de [consulta 6 de mayo de 2013]

4.1.3.2.1 Especificaciones técnicas

1. Conversión individual ASK súper regenerativa.
2. Fuente de alimentación: 5V
3. Consumo de corriente: 2,2mA

4.1.3.2.2 Características

1. Velocidades de hasta 4,8kbps.
2. Alcance utilizable de hasta 110 metros.
3. Versiones disponibles en 433,92 MHz (433MHz) y 315MHz.
4. Versiones disponibles: regulado y no regulado.
5. Adopción de inductancia ajustable para estabilizar la frecuencia.
6. Rápido tiempo de establecimiento de datos.

4.1.3.2.3 Descripción de pines

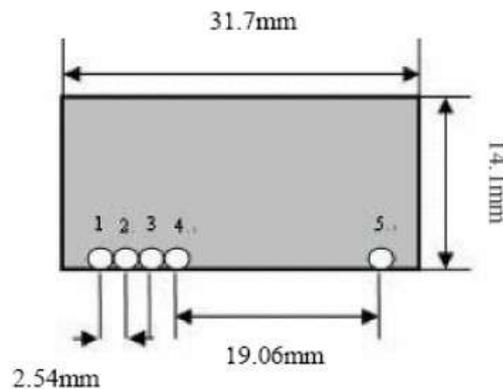


Figura 4.6: Pinaje del receptor.

(TECMIKRO. Radiofrecuencia: sistemas TX/RX a 433MHz)

Fuente: <http://www.programarpicenc.com/libro/cap15-a-radiofrecuencia-tx-rx-433mhz.html> Fecha de [consulta 6 de mayo de 2013]

- GND (pin 1): Tierra del suministro de DC
- DATA (pin 2): Salida de datos digitales. Puede ser utilizado para alimentar decodificadores externos. Los datos son datos verdaderos, es decir, como se ingresan al transmisor. Internamente conectado al pin 3.
- DATA (pin 3): Salida de datos digitales. Puede ser utilizado para alimentar decodificadores externos. Los datos son datos verdaderos, es decir, como se ingresan al transmisor. Internamente conectado al pin 2.

- VCC (pin 4): Suministro de 5vcc regulada, conectado internamente al pin 5.
Contenido máximo de rizado 0,1Vpp.
- ANT (pin 5): Entrada de antena de 50 ohm.

4.1.4 Sensores de presión

Los sensores de presión están en contacto con la presión de los neumáticos, estos sensores convierten la señal física en una magnitud eléctrica. Esta magnitud eléctrica es enviada al sistema electrónico en el cual se procesa según la programación dada. Estos sensores mandan señales en el rango de 0 a 5 voltios.

4.1.5 Regulador 7805

La familia 78xx y 79xx son una gama de integrados dedicados a la regulación de voltaje, hay muchas variables: regulables, fijos, de tensión positiva o negativa. El más común, y el que mayormente se usa es el famoso 7805, que es un regulador de tensión positiva de 5 Volts a 1A. Se sabe que el buen funcionamiento del firmware que grabemos en el PIC está sujeto, no sólo a la buena programación que se haya hecho a la hora de diseñarlo, sino que también a una alimentación fija, constante y regulada a la hora de ejecutarlo. Entonces la manera más segura, económica y sencilla de obtener ese voltaje, es la utilización de un integrado regulador de voltaje. El regulador 7805 se lo utiliza para dar a nuestro sistema los 5 voltios que requiere. En la figura 4.7 se muestra el circuito necesario para obtener este voltaje.

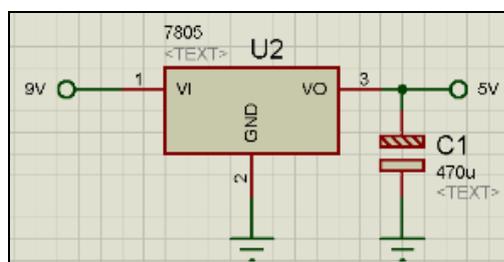


Figura 4.7: Circuito de alimentación de 5 voltios.

4.1.6 Condensadores de 470uF 50V

“Un condensador es un dispositivo capaz de almacenar energía en forma de campo eléctrico. Está formado por dos armaduras metálicas paralelas (generalmente de aluminio) separadas por un material dieléctrico. Tiene una serie de características tales como capacidad, tensión de trabajo, tolerancia y polaridad”¹⁴.

4.2 Diseño del esquema electrónico del sistema TPMS

Par el diseño del esquema electrónico del sistema de control de presión de neumáticos se necesita de los programas Code Vision AVR V2.05.0 y del programa ISIS 7 Professional. Estos programas servirán para la programación del ATMEGA 8 y la simulación del sistema TPMS.

“El programa Code Vision AVR V2.05.0 es un compilador desarrollado por Pavel Haiduc para los microcontroladores AVR de 8 bits, desde los tinyAVR hasta los XMEGA. Su principal ventaja es que provee librerías integradas para controlar sus periféricos internos y también dispositivos externos como LCDs, GLCDS, RTCs, sensores de temperatura, memorias SD, etc. En este sentido se le podría comparar con los compiladores C de CCS o Mikroe para los PICmicro.

El compilador CodeVisionAVR no tiene un simulador del programa creado. En su lugar genera un buen archivo COFF que puede ser utilizado por otros programas como Armel Studio 6 o Proteus”¹⁵

“El programa ISIS 7 Professional es una compilación de programas de diseño y simulación electrónica, desarrollado por Labcenter Electronics que consta de los

¹⁴ ABELDG. Condensadores. [en línea]

<http://perso.wanadoo.es/abeldg/documentacion/condensadores.pdf> [consultado 8 de mayo del 2013]

¹⁵ CURSOMICROS. Copiladores Code Vision [en línea]

AVR <http://www.cursomicros.com/avr/compiladores/compiladores-para-avr.html#compilador-codevision> [consultado 8 de mayo del 2013]

dos programas principales: *Ares e Isis, y los módulos VSM y Electra*¹⁶.

El Programa ISIS, *Intelligent Schematic Input System* (Sistema de Enrutado de Esquemas Inteligente) permite diseñar el plano eléctrico del circuito que se desea realizar con componentes muy variados, desde simples resistencias, hasta alguno que otro microprocesador o microcontrolador, incluyendo fuentes de alimentación, generadores de señales y muchos otros componentes con prestaciones diferentes. Los diseños realizados en Isis pueden ser simulados en tiempo real, mediante el módulo VSM, asociado directamente con ISIS.

Para el esquema real se necesita los siguientes componentes:

1. 5 microcontroladores ATMEGA 8.
2. 4 sensores de presión.
3. 5 transmisores.
4. 5 receptores.
5. 1 LCD 2X16.
6. 4 pilas baterías de 9V.
7. 5 reguladores 7805.
8. 5 condensadores de 470uF 50V.

4.2.1 Funcionamiento del esquema electrónico del sistema TPMS

El circuito visualizador (Figura 4.8) que se encuentra ubicado en el interior del vehículo envía a través del puerto serie y por medio del transmisor de radiofrecuencia el número 1, este número es recibido por los módulos de radiofrecuencia de recepción de cada uno de los circuitos que se encuentran en cada rueda (figura 4.9). El microcontrolador de cada rueda comienza a preguntar qué número a recibido, como en este caso es el número 1 el microcontrolador de la rueda delantera izquierda envía el dato de presión que adquirió a través del puerto analógico 0. Después el circuito visualizador envía el número 2, entonces el

¹⁶ IDENTI. INFO. Proteus. [en línea] http://identi.info/downloads/92859277/Proteus-7_8-y-7_7-Full-Para-Windows-7.html [consultado 7 de mayo del 2013]

microcontrolador de la rueda delantera derecha envía el dato de presión para que sea visualizado, después el circuito visualizador envía el número 3, y el circuito de la rueda trasera izquierda envía su dato de presión y por último el circuito visualizador envía el número 4 para que el circuito de la llanta trasera derecha envíe su dato de presión, después se repite el proceso con la rueda delantera izquierda.

4.2.2 Código realizado en codevision AVR para el circuito visualizador

Para el código del circuito visualizador debemos seguir los siguientes pasos.

1. Colocar las librerías para el ATMEGA 8, la librería para pasar de entero a asqui y la librería para los tiempos.
2. Colocar la sección de variables para recoger los datos de presión a través del puerto serial, la variable para visualizar los datos en el LCD y la variable que se necesita para esperar un tiempo en la programación.
3. Se desarrolla la subrutina que realiza el parpadeo cuando la presión esta fuera de 35 o 36 psi.
 - Se debe ubicar al cursor en la posición X e Y.
 - Se transforma el dato de entero a asqui.
 - Se pone en el LCD el valor del dato.
 - Se debe agregar en el LCD las letras psi.
 - Se debe colocar en la programación un tiempo de espera.
 - Se debe colocar de nuevo al cursor del LCD en la posición X e Y.
 - Se programa para que el LCD se ponga todos los espacios en blanco para que titile.
 - Se debe colocar en la programación un tiempo de espera.
 - Se debe colocar de nuevo al cursor del LCD en la posición X e Y.
 - Se pone en el LCD el valor del dato.
 - Se debe agregar en el LCD las letras psi.
4. Se desarrolla la subrutina que muestra los valores de 35 o 36 psi.
 - Se debe ubicar al cursor en la posición X e Y.
 - Se transforma el dato de entero a asqui.
 - Se pone en el LCD el valor del dato.
 - Se debe agregar en el LCD las letras psi.

5. Se desarrolla el programa principal del microcontrolador.
 - Se programa para que el programa llame a la subrutina de la configuración inicial.
 - Enviar a través del puerto serial el numero 1.
 - Programar para que el microcontrolador reciba el valor transmitido por el sensor de cada rueda.
 - Se debe programar para que pregunte si la variable dato es mayor a 36 o menor a 35 psi.
 - Si es menor o mayor se llama a subrutina muestra para titilar.
 - Se debe programar para que pregunte si la variable dato no es mayor a 36 o menor a 35 psi.
 - Si no es menor o mayor se llama a subrutina muestra 1 para que quede fijo en el LCD.
 - Lo mismo se hace para las ruedas 2,3,4 enviando por el puerto serial el numero 2,3,4.

El código para la programación del circuito visualizador se puede revisar en el anexo 1.

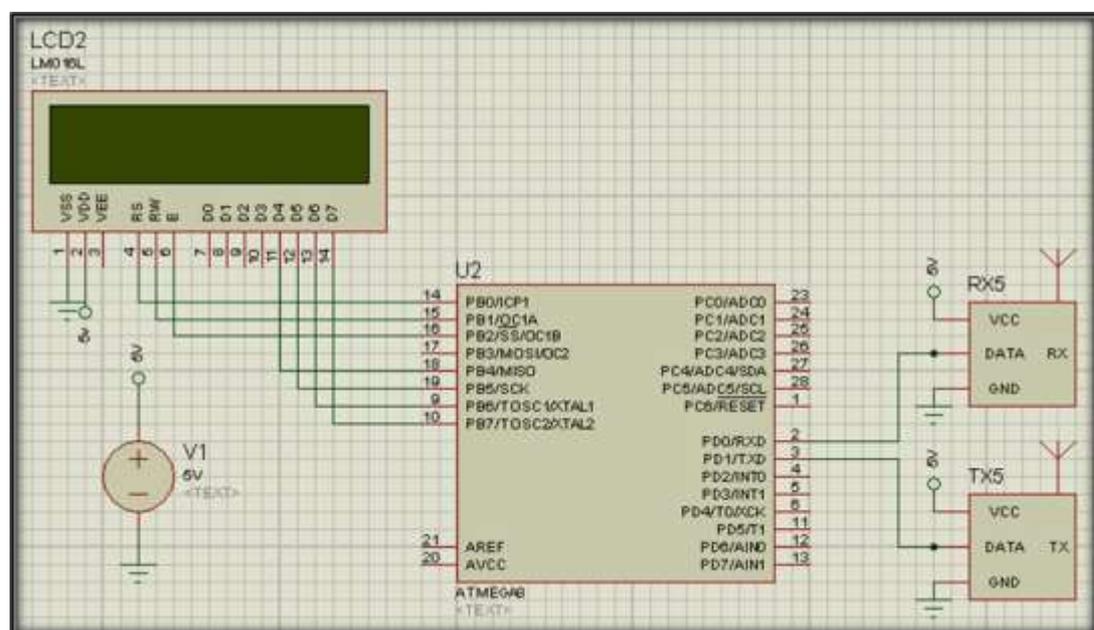


Figura 4.8: Circuito visualizador.

4.2.3 Código fuente para la rueda delantera izquierda

El código que se necesita debe transformar el valor del sensor y enviar al circuito visualizador. Para esto se debe seguir los siguientes pasos.

1. Colocar las librerías que se mencionaron anteriormente.
2. Se coloca las variables para recoger y enviar datos del puerto serie.
3. Se coloca la subrutina para convertir los datos analógicos en digitales.
4. Se desarrolla el programa principal del microcontrolador.
 - Se programa para que el programa llame a la subrutina de la configuración inicial.
 - Se programa para que el programa llame a la configuración inicial del microcontrolador.
 - Se programa para que pregunte si el dato es igual al dato adquirido en el puerto.
 - Se debe programar para que pregunte si el dato es igual a 1.
 - Se programa para que adquiera el dato del puerto analógico 0.
 - Se debe enviar el dato a través del puerto serie.

Este código sirve para las demás ruedas del sistema. Se debe cambiar el número en dato==2, dato==3, dato==4 y de que puerto adquiere el dato.

El código para la programación de cada rueda se puede revisar en el anexo 2.

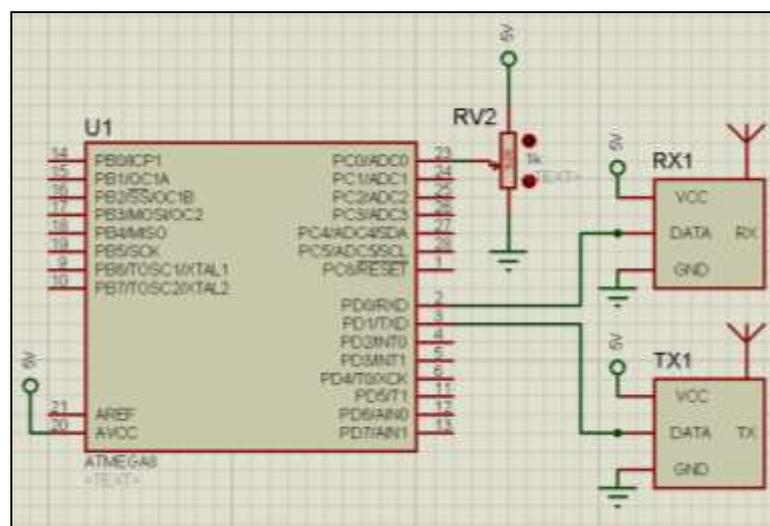


Figura 4.9: circuito de la rueda.

4.3 Simulación del circuito electrónico

Para la simulación del sistema (Figura 4.10) se recurrió a un circuito que permita simular en el programa ISIS 7 PROFESIONAL, ya que la transmisión por radiofrecuencia no permite simular el programa.

4.3.1 Funcionamiento

El ATMEGA 8 por el puerto analógico 0 (pin 23) recoge el valor del primer sensor, este valor será convertido en un valor de presión de 0 a 60 psi y será visualizado en la ubicación 0,0 del LCD. En caso que la presión sea menor a 35 o mayor a 36 este valor será visto en el LCD de manera titilante, si el valor del sensor está entre 35 y 36 psi el valor se mostrara fijo, lo mismo hace con el segundo puerto analógico 1 (pin 24), después con el puerto analógico 2 (pin 25) y termina con el puerto analógico 3 (pin 26), repite el procedimiento desde el puerto analógico 0.

4.3.2 Código fuente para la simulación de los 4 sensores

Esta programación nos ayudara a simular el sistema TPMS. Para esto se debe seguir los siguientes pasos.

1. Colocar las librerías para el ATMEGA 8, la librería para pasar de entero a asqui y la librería para los tiempos.
2. Colocar la sección de variables para recoger los datos de presión a través del puerto serial, la variable para visualizar los datos en el LCD y la variable que se necesita para esperar un tiempo en la programación.
3. Se coloca la subrutina para convertir los datos de análogo a digital.
4. Se programa la subrutina que inicializa al microcontrolador.
5. Se desarrolla la subrutina que realiza el parpadeo cuando la presión esta fuera de 35 o 36 psi.
 - Se debe ubicar al cursor en la posición X e Y.
 - Se transforma el dato de entero a asqui.

- Se pone en el LCD el valor del dato.
 - Se debe agregar en el LCD las letras psi.
 - Se debe colocar en la programación un tiempo de espera.
 - Se debe colocar de nuevo al cursor del LCD en la posición X e Y.
 - Se programa para que el LCD se ponga todos los espacios en blanco para que titile.
 - Se debe colocar en la programación un tiempo de espera.
 - Se debe colocar de nuevo al cursor del LCD en la posición X e Y.
 - Se pone en el LCD el valor del dato.
 - Se debe agregar en el LCD las letras psi.
6. Se desarrolla la subrutina que muestra los valores de 35 o 36 psi.
- Se debe ubicar al cursor en la posición X e Y.
 - Se transforma el dato de entero a asqui.
 - Se pone en el LCD el valor del dato.
 - Se debe agregar en el LCD las letras psi.
7. Se desarrolla el programa principal del microcontrolador.
- Se llama a la subrutina de la configuración para inicializar el microcontrolador.
 - En la variable dato se debe guardar el valor digital del puerto analógico 0.
 - Se debe programar para que pregunte si el dato es menor a 35 o mayor a 36 psi.
 - Si el dato es menor o mayor llama a la subrutina muestra para titilar
 - Si el dato es igual a 35 o 36 se llama a subrutina muestra 1 para mostrar el valor en la ubicación del LCD 0.0
 - En la variable dato se guarda el valor digital del puerto analógico 1.
 - Se debe programar para que pregunte si el dato es menor a 35 o mayor a 36 psi.
 - Si es menor o mayor se llama a subrutina muestra para titilar en el LCD en la ubicación 11.0
 - Si es igual a 35 o 36 se llama a subrutina muestra 1 para mostrar el valor en la ubicación del LCD 11.0

- En la variable dato se guarda el valor digital del puerto analógico 2.
- Se pregunta si e dato es menor a 35 o mayor a 36 psi.
- Si es menor o mayor se debe llamar a subrutina muestra para titilar en el LCD en la ubicación 0.1
- Si es igual a 35 o 36 psi llama a subrutina muestra 1 para mostrar el valor en el LCD en la ubicación 0,1
- En la variable dato se guarda el valor digital del puerto analógico 3
- Se bebe preguntar si el dato es menor a 35 o mayor a 36 psi.
- Si es menor o mayor se llama a la subrutina muestra para titilar en el LCD en la ubicación 11,1
- Si es igual a 35 o 36 llama a subrutina muestra 1 para mostrar el valor en el LCD en la ubicación 11.1

El código fuente para la simulación de los cuatro sensores se puede revisar en el anexo 3.

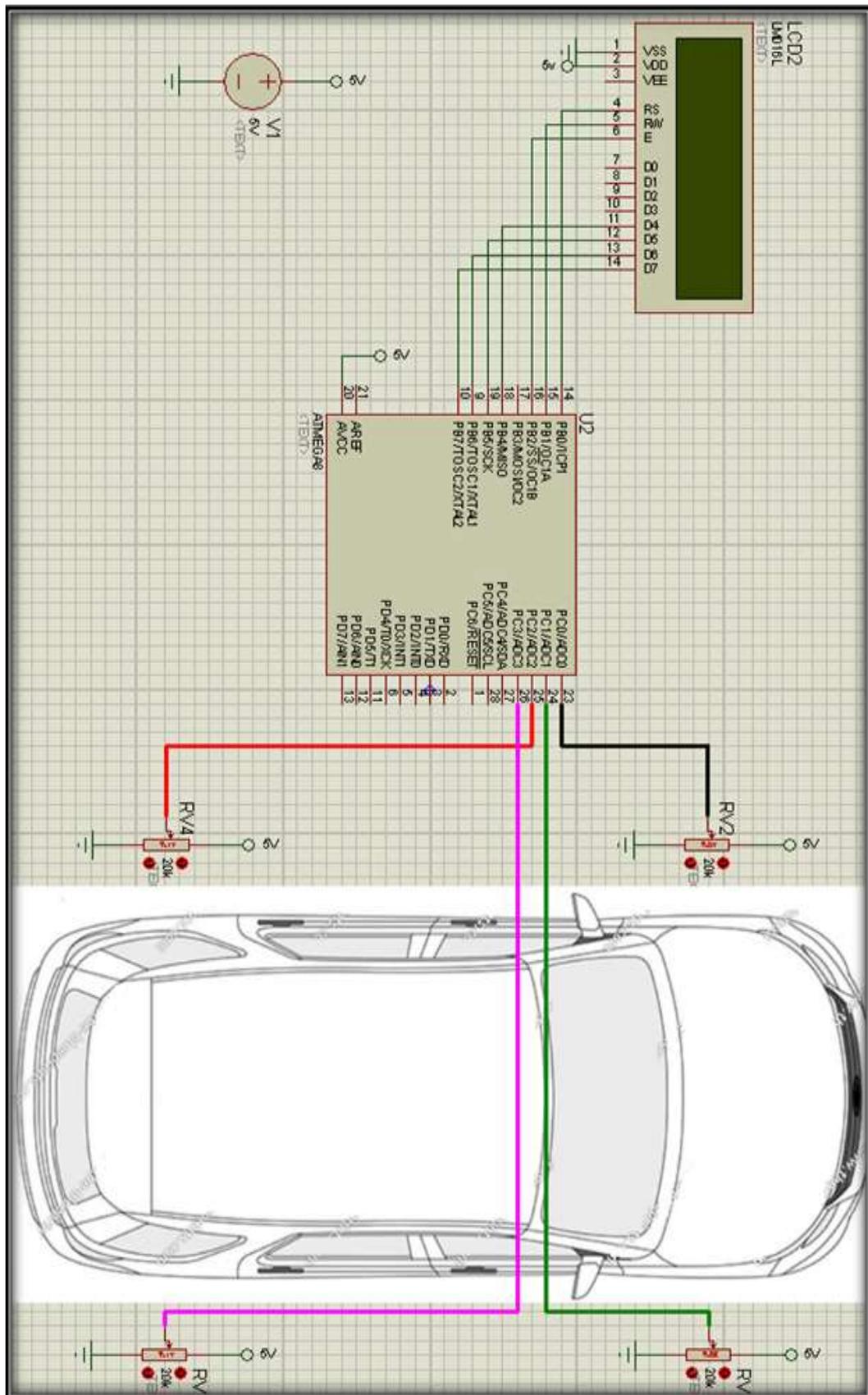


Figura 4.10: Esquema de simulación de los 4 sensores.

CONCLUSIONES Y RECOMENDACIONES

Al realizar el análisis comparativo de los sistemas TPMS se llegó a la finalidad de que el sistema TPMS directo es el más adecuado ya que tiene mayor precisión al procesar los datos de las presiones de los neumáticos.

Al realizar los procedimientos de calibración del sistema de monitoreo de presión directo se concluyó que los pasos son más fáciles de realizar teniendo la herramienta adecuada y la correcta guía de procedimientos a seguir.

La programación del sistema fue desarrollada en el programa codevision AVR para que al grabarla en el microcontrolador pueda funcionar correctamente haciendo que el sistema TPMS creado de los datos exactos de la presión de las llantas en tiempo real.

En cuanto a la simulación del sistema se hicieron unas modificaciones para que el programa ISIS 7 PROFESIONAL pueda simular el sistema ya que no se pudo establecer una conexión por medio de radiofrecuencia en este programa.

Para que el microcontrolador pueda adquirir el dato de presión, se debió multiplicar los 256 que es el tamaño de los datos que soporta el ATMEGA 8 por 60 y esto dividir para 256. Esto nos da valores de 0 a 60 que es lo que necesitamos para nuestro sistema.

Se recomienda utilizar la programación desarrollada en este documento cuando se realice físicamente el sistema TPMS y realizar estudios de costos y eficiencia en el vehículo ya que la parte electrónica del sistema no alcanza un valor alto. Se deberá tener en cuenta que mientras más económico sea el sensor más barato será el sistema.

BIBLIOGRAFÍA

REFERENCIAS BIBLIOGRÁFICAS

- ALONSO, José Manuel. **Sistemas de seguridad y confortabilidad** 4ta edición. Editorial Paraninfo. Madrid. 2008.
- ALONSO José Miguel, “**Técnicas del automóvil**”, Editorial Paraninfo, Madrid, 2006.
- ANGULO USATEGUI, José María. **microcontroladores DSPIC. Diseño práctico de aplicaciones**. McGraw-Hill Professional. Interamericana de España, SA. Madrid. 2006.
- BOSCH, Robert. **Manual de la técnica del automóvil 4ta generación**. Editorial GmbH. Plochingen. 2005.
- CARLOS REYES, “**Programación de microcontroladores**”, Graficas Ayerve, Ecuador 2004.
- RIBBENS, William. **Electrónica automotriz**. LIMUSA, México, 2008.
- ROMERO YESA, Susana, *Microcontroladores PIC diseño práctico de aplicaciones segunda parte PIC 16F87X, (2da edición)*. Concesión Fernández Madrid, España, 2004.
- RUEDA SANTANDER, Jesús. **Técnico en mecánica & electrónica automotriz, Volumen 1**. Diseli, 2003.
- SHEPARD, Steven. **Radio frecuencym identification**. McGraw-Hill Professional. 2004.
- VALDEZ PERES, Fernando. **Microcontroladores: fundamentos y aplicaciones con PIC**. Editorial Carles Parcerisas Civit. 2007.

REFERENCIAS ELECTRÓNICAS

- ATMEL: **AVR Atmega8**; <http://www.atmel.com/images/doc2486.pdf>
(Fecha de consulta 8 de febrero 2013).
- GÓRDON Christian .**Módulo didáctico para el microcontrolador ATMEGA8**. <http://bibdigital.epn.edu.ec/bitstream/15000/1955/1/CD-0193.pdf> pag 14 (fecha de consulta 1 de mayo de 2013).
- DATASHEET. **Atmega 8**; <http://www.datasheetcatalog.com> (Fecha de consulta 10 de febrero 2013)
- DIARIO MOTOR; ÁLVAREZ Sergio; **La incorrecta presión de los neumáticos aumenta la contaminación y disminuye la seguridad**;
<http://www.diariomotor.com/2008/04/30/la-incorrecta-presion-de-los-neumaticos-aumenta-la-contaminacion-y-disminuye-la-seguridad/> 30 de abril 2008 (fecha de consulta 5 de enero 2013)
- INFANTE José Ángel; **Sistema de control de presión de neumáticos en Renault Laguna**; <http://www.euskalnet.net/jinfante/presneum.html>. (fecha de consulta 5 de enero 2013)
- MECANIQUE; **MicroCode Studio**. <http://www.mecanique.co.uk/code-studio/index.html>
(Fecha de consulta 8 de febrero 2013)
- PNEUS ONLINE; **Presión de neumáticos**; Francia; (fecha de consulta 8 de enero 2013) <http://www.neumaticos-pneus-online.es/presion-consejos.html>
- TECMIKRO; **Radiofrecuencia: sistema TX/RX a 433MHz**;
<http://www.programarpicenc.com/libro/cap15-a-radiofrecuencia-tx-rx-433mhz.html> (Fecha de consulta 6 de mayo de 2013).
- TPMS DIRECT; **Recetado para vehículos Ford**; 28 febrero 2012; 3 de enero 2013; <http://tpmsdirect.wordpress.com/2012/02/28/how-to-reset-ford-tpms-sensors/>
(fecha de consulta 3 de enero del 2013).
- TPMS DIRECT; **Registro específico para Ford**;
http://www.tpmsdirect.com/FORDTRIGGER_TPMS_Instructions_a/729.htm (fecha de consulta 10 de enero 2013)
- TPMS DIRECT; **Prevent TPMS sensor corrosión**;
http://www.tpmsdirect.com/TPMS_Corrosion_Prevention_a/743.htm (fecha de consulta 20 de marzo de 2013).

ANEXOS

Anexos 1: Código para el circuito visualizador

```

/*****

```

```

This program was produced by the
CodeWizardAVR V2.05.0 Professional
Automatic Program Generator
© Copyright 1998-2010 Pavel Haiduc, HP InfoTech s.r.l.
http://www.hpinfotech.com

```

```

Project : Visualiza la presión de las llantas.
Version : 1.0
Date   : 22/04/2013
Author : Andres Mora.
Company : UDA
Comments: Circuito principal en el interior del vehículo.

```

```

Chip type      : ATmega8
Program type   : Application
AVR Core Clock frequency: 8,000000 MHz
Memory model   : Small
External RAM size : 0
Data Stack size : 256
*****/

```

```

/*-----

```

SECCION DE LIBRERIAS

```

-----*/
#include <mega8.h>    // LIBRERIA DEL ATMEGA 8
#include <stdlib.h>   //LIBRERIA PARA PASAR DE ENTERO A ASQUI
#include <delay.h>    //LIBRERIA PARA TIEMPOS

```

```

/*-----

```

SECCION DE VARIABLES

```

-----*/
unsigned char dato;    //VARIABLE PARA RECOGER DATOS DE PRESION ATRAVES DEL
PUERTO SERIAL.
unsigned char mensaje[5]; //VARIABLE PARA VISUALIZAR DATOS EN EL LCD.
unsigned char tiempo=500; //VARIABLE PARA ESPERAR UN TIEMPO.

```

```

/*-----

```

SUBROUTINA DE CONFIGUACION INICIAL DEL MICROCONTROLADOR PARA TRANSMITIR Y RECIBIR SERIALMENTE

```

-----*/

```

```

// Alphanumeric LCD Module functions

```

```

#include <alcd.h>

```

```

#ifndef RXB8
#define RXB8 1
#endif

```

```

#ifndef TXB8
#define TXB8 0
#endif

```

```

#ifndef UPE
#define UPE 2
#endif

#ifndef DOR
#define DOR 3
#endif

#ifndef FE
#define FE 4
#endif

#ifndef UDRE
#define UDRE 5
#endif

#ifndef RXC
#define RXC 7
#endif

#define FRAMING_ERROR (1<<FE)
#define PARITY_ERROR (1<<UPE)
#define DATA_OVERRUN (1<<DOR)
#define DATA_REGISTER_EMPTY (1<<UDRE)
#define RX_COMPLETE (1<<RXC)

// USART Receiver buffer
#define RX_BUFFER_SIZE 8
char rx_buffer[RX_BUFFER_SIZE];

#if RX_BUFFER_SIZE <= 256
unsigned char rx_wr_index,rx_rd_index,rx_counter;
#else
unsigned int rx_wr_index,rx_rd_index,rx_counter;
#endif

// This flag is set on USART Receiver buffer overflow
bit rx_buffer_overflow;

// USART Receiver interrupt service routine
interrupt [USART_RXC] void usart_rx_isr(void)
{
char status,data;
status=UCSRA;
data=UDR;
if ((status & (FRAMING_ERROR | PARITY_ERROR | DATA_OVERRUN))==0)
{
rx_buffer[rx_wr_index++]=data;
#if RX_BUFFER_SIZE == 256
// special case for receiver buffer size=256
if (++rx_counter == 0)
{
#else
if (rx_wr_index == RX_BUFFER_SIZE) rx_wr_index=0;
if (++rx_counter == RX_BUFFER_SIZE)
{
rx_counter=0;
#endif
rx_buffer_overflow=1;
}

```

```

    }
}

#endif

#ifndef _DEBUG_TERMINAL_IO_
// Get a character from the USART Receiver buffer
#define _ALTERNATE_GETCHAR_
#pragma used+
char getchar(void)
{
    char data;
    while (rx_counter==0);
    data=rx_buffer[rx_rd_index++];
    #if RX_BUFFER_SIZE != 256
    if (rx_rd_index == RX_BUFFER_SIZE) rx_rd_index=0;
    #endif
    #asm("cli")
    --rx_counter;
    #asm("sei")
    return data;
}
#pragma used-
#endif

// Standard Input/Output functions
#include <stdio.h>

/*-----
SUBROUTINA DE CONFIGUACION INICIAL DEL MICROCONTROLADOR
-----*/
void configuracion(void)
{
    // Declare your local variables here

    // Input/Output Ports initialization
    // Port B initialization
    // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
    // State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
    PORTB=0x00;
    DDRB=0x00;

    // Port C initialization
    // Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
    // State6=T State5=T State4=T State3=T State2=T State1=T State0=T
    PORTC=0x00;
    DDRC=0x00;

    // Port D initialization
    // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
    // State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
    PORTD=0x00;
    DDRD=0xC0;

    // Timer/Counter 0 initialization
    // Clock source: System Clock
    // Clock value: Timer 0 Stopped
    TCCR0=0x00;
    TCNT0=0x00;

    // Timer/Counter 1 initialization

```

```
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer2 Stopped
// Mode: Normal top=0xFF
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
MCUCR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x98;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x33;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// ADC initialization
// ADC disabled
```

```

ADCSRA=0x00;

// SPI initialization
// SPI disabled
SPCR=0x00;

// TWI initialization
// TWI disabled
TWCR=0x00;

// Alphanumeric LCD initialization
// Connections specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTB Bit 0
// RD - PORTB Bit 1
// EN - PORTB Bit 2
// D4 - PORTB Bit 4
// D5 - PORTB Bit 5
// D6 - PORTB Bit 6
// D7 - PORTB Bit 7
// Characters/line: 16
lcd_init(16);

// Global enable interrupts
#asm("sei")
}

/*-----
SUBROUTINA QUE REALIZA EL PARPADEO CUANDO LA PRESION ESTA FUERA DE 35 O
36 PSI
-----*/
void muestra(unsigned char x, unsigned char y)
{
  lcd_gotoxy(x,y);          //UBICA AL CURSOR DEL LCD EN LA POSICION X Y Y.
  itoa(dato,mensaje);      //TRANSFORMA EL DATO DE ENTERO A ASQUI.
  lcd_puts(mensaje);       //PONE EN EL LCD EL VALOR DEL DATO.
  lcd_putsf("PSI");        //AGREGA EN EL LCD LAS LETRAS PSI.
  delay_ms(tiempo);        //ESPERA UN TIEMPO.
  lcd_gotoxy(x,y);         //PONE AL CURSOR DEL LCD EN LA POSICION X Y Y.
  lcd_putsf("   ");        //PONE EN EL LCD TODOS LOS ESPACIOS EN BLANCO PARA
  TITILAR.
  delay_ms(tiempo);        //ESPERA UN TIEMPO.
  lcd_gotoxy(x,y);         //UBICA NUEVAMENTE AL CURSOR DEL LCD EN X Y Y.
  lcd_puts(mensaje);       //PONE EN EL LCD EL VALOR DE DATO.
  lcd_putsf("PSI");        //AGREGA EN EL LCD LAS LETRAS PSI.
}

/*-----
SUBROUTINA QUE MUESTRA LOS VALORES DE 35 O 36 PSI FIJOS
-----*/
void muestra1(unsigned char x, unsigned char y)
{
  lcd_gotoxy(x,y);          //UBICA AL CURSOR DEL LCD EN LA POSICIÓN X Y Y.
  itoa(dato,mensaje);      //TRANSFORMA EL DATO DE ENTERO A ASQUI.
  lcd_puts(mensaje);       //PONE EN EL LCD EL VALOR DEL DATO.
  lcd_putsf("PSI");        //AGREGA EN EL LCD LAS LETRAS PSI.
}

/*-----
PROGRAMA PRINCIPAL DEL MICROCONTROLADOR

```

```

-----*/
void main(void)
{
configuracion();           //LLAMA A SUBROUTINA DE LA CONFIGURACION INICIAL.
while (1)
{
putchar(1);               //ENVIA ATRAVES DEL PUERTO SERIAL EL NUMERO 1.
dato=getchar();           //DATO RECIBE EL VALOR TRANSMITIDO POR EL SENSOR DE
CADA RUEDA.
if (dato>36 || dato<35) //PREGUNTA SI LA VARIABLE DATO ES MAYOR 36 O MENOR A 35
{
muestra(0,0);
}
else                       //CASO CONTRARIO SI NO ES MAYOR A 36 O MENOR A 35
{
muestra1(0,0);
}
putchar(2);               //ENVIO ATRAVES DEL PUERTO SERIAL EL NUMERO 2
dato=getchar();           //DATO RECIBE EL VALOR TRANSMITIDO POR EL SENSOR DE
CADA RUEDA
if (dato>36 || dato<35) //PREGUNTA SI LA VARIABLE DATO ES MAYOR 36 O MENOR A 35
{
muestra(11,0);
}
else                       //CASO CONTRARIO SI NO ES MAYOR A 36 O MENOR A 35
{
muestra1(11,0);
}
putchar(3);               //ENVIO ATRAVES DEL PUERTO SERIAL EL NUMERO 3
dato=getchar();           //DATO RECIBE EL VALOR TRANSMITIDO POR EL SENSOR DE
CADA RUEDA
if (dato>36 || dato<35) //PREGUNTA SI LA VARIABLE DATO ES MAYOR 36 O MENOR A 35
{
muestra(0,1);
}
else                       //CASO CONTRARIO SI NO ES MAYOR A 36 O MENOR A 35
{
muestra1(0,1);
}
putchar(4);               //ENVIO ATRAVES DEL PUERTO SERIAL EL NUMERO 4
dato=getchar();           //DATO RECIBE EL VALOR TRANSMITIDO POR EL SENSOR DE
CADA RUEDA
if (dato>36 || dato<35) //PREGUNTA SI LA VARIABLE DATO ES MAYOR 36 O MENOR A 35
{
muestra(11,1);
}
else                       //CASO CONTRARIO SI NO ES MAYOR A 36 O MENOR A 35
{
muestra1(11,1);
}
}
}
}

```

Anexo 2: Código fuente para la rueda delantera izquierda

```
/******
```

```
This program was produced by the
CodeWizardAVR V2.05.0 Professional
Automatic Program Generator
© Copyright 1998-2010 Pavel Haiduc, HP InfoTech s.r.l.
http://www.hpinfotech.com
```

```
Project : CIRCUITO CON EL SENSOR EN CADA RUEDA
Version : 1.0
Date : 22/04/2013
Author : ANDRES MORA
Company : UDA
Comments: TRANSFORMA EL VALOR DEL SENSOR Y ENVIA AL CIRCUITO
VISUALIZADOR
```

```
Chip type : ATmega8
Program type : Application
AVR Core Clock frequency: 8,000000 MHz
Memory model : Small
External RAM size : 0
Data Stack size : 256
*****/
```

```
/*-----
```

SECCION DE LIBRERIAS

```
-----*/
```

```
#include <mega8.h>
#include <stdlib.h>
#include <delay.h>
#include <stdio.h>
#include <alcd.h>
```

```
/*-----
```

SECCION DE VARIABLES

```
-----*/
```

```
unsigned int dato; //VARIABLE PARA RECOGER DATOS DEL PUERTO SERIE.
unsigned int sensor1; //VARIABLE PARA ENVIAR DATOS EN EL PUERTO SERIE.
```

```
/*-----
```

SUBROUTINA PARA CONVERTIR DATOS ANALOGICOS A DIGITALES.

```
-----*/
```

```
#define ADC_VREF_TYPE 0x60
```

```
// Read the 8 most significant bits
// of the AD conversion result
unsigned char read_adc(unsigned char adc_input)
{
    ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
    // Delay needed for the stabilization of the ADC input voltage
    delay_us(10);
    // Start the AD conversion
    ADCSRA|=0x40;
    // Wait for the AD conversion to complete
    while ((ADCSRA & 0x10)==0);
    ADCSRA|=0x10;
    return ADCH;
}
```

```

/*-----
SUBROUTINA DE CONFIGURACION INICIAL DEL MICROCONTROLADOR
-----*/
void configuracion(void)
{
// Declare your local variables here

// Input/Output Ports initialization
// Port B initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTB=0x00;
DDRB=0x00;

// Port C initialization
// Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTC=0x00;
DDRC=0x00;

// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTD=0x00;
DDRD=0x00;

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
TCCR0=0x00;
TCNT0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer2 Stopped
// Mode: Normal top=0xFF
// OC2 output: Disconnected

```

```
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
MCUCR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x33;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// ADC initialization
// ADC Clock frequency: 250,000 kHz
// ADC Voltage Reference: AVCC pin
// Only the 8 most significant bits of
// the AD conversion result are used
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x85;

// SPI initialization
// SPI disabled
SPCR=0x00;

// TWI initialization
// TWI disabled
TWCR=0x00;

// Alphanumeric LCD initialization
// Connections specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTB Bit 0
// RD - PORTB Bit 1
// EN - PORTB Bit 2
// D4 - PORTB Bit 4
// D5 - PORTB Bit 5
// D6 - PORTB Bit 6
// D7 - PORTB Bit 7
// Characters/line: 16
lcd_init(16);
}
```

```

/*-----
PROGRAMA PRINCIPAL DEL MICROCONTROLADOR
-----*/
void main (void)
{
configuracion(); //LLAMA A CONFIGURACION INICIAL DEL
MICROCONTROLADOR.
while (1)
{
dato=getchar(); //DATO ES IGUAL AL DATO ADQUIRIDO EN EL PUERTO.
SERIE
if (dato==1) //PREGUNTA SI DATO ES IGUAL A 1.
{
sensor1=read_adc(0)*60/255; //ADQUIERE EL DATO DEL PUERTO ANALOGICO 0.
putchar(sensor1); //ENVIA EL DATO ATRAVES DEL PUERTO SERIE.
};
}
}

```

Este código sirve para las demás ruedas del sistema. Se debe cambiar el número en dato==2, dato==3, dato==4 y de que puerto adquiere el dato.

Anexo 3: Código fuente para la simulación de los 4 sensores

```

/*****

```

```

This program was produced by the
CodeWizardAVR V2.05.0 Professional
Automatic Program Generator
© Copyright 1998-2010 Pavel Haiduc, HP InfoTech s.r.l.
http://www.hpinfotech.com

```

```

Project : Sensor TPMS
Version : 1.0
Date   : 22/04/2013
Author  : Andres Mora
Company : UDA
Comments: Simila los sensores TPMS.

```

```

Chip type      : ATmega8
Program type   : Application
AVR Core Clock frequency: 8,000000 MHz
Memory model   : Small
External RAM size : 0
Data Stack size : 256

```

```

*****/

```

```

/*-----

```

SECCION DE LIBRERIAS

```

-----*/

```

```

#include <mega8.h>      //LIBRERIA DEL ATMEGA 8.
#include <stdlib.h>     //LIBRERIA PARA PASAR DE ENTERO A ASQUI.
#include <delay.h>      //LIBRERIA PARA TIEMPOS.

```

```

/*-----

```

SECCION DE VARIABLES

```

-----*/

```

```

unsigned int dato;
unsigned char mensaje[5];
unsigned char tiempo=100;

```

```

// Alphanumeric LCD Module functions
#include <alcd.h>

```

```

#define ADC_VREF_TYPE 0x60

```

```

/*-----

```

SUBROUTINA PARA CONVERTIR DATOS DE ANALOGICO A DIGITAL.

```

-----*/

```

```

// Read the 8 most significant bits
// of the AD conversion result
unsigned char read_adc(unsigned char adc_input)
{
  ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
  // Delay needed for the stabilization of the ADC input voltage
  delay_us(10);
  // Start the AD conversion

```

```

ADCSRA|=0x40;
// Wait for the AD conversion to complete
while ((ADCSRA & 0x10)==0);
ADCSRA|=0x10;
return ADCH;
}

/*-----
SUBROUTINA QUE INICIALIZA AL MICROCONTROLADOR.
-----*/
void configuracion(void)
{
// Declare your local variables here

// Input/Output Ports initialization
// Port B initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTB=0x00;
DDRB=0x00;

// Port C initialization
// Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTC=0x00;
DDRC=0x00;

// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTD=0x00;
DDRD=0x00;

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
TCCR0=0x00;
TCNT0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;

```

```
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer2 Stopped
// Mode: Normal top=0xFF
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
MCUCR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;

// USART initialization
// USART disabled
UCSRB=0x00;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// ADC initialization
// ADC Clock frequency: 250,000 kHz
// ADC Voltage Reference: AVCC pin
// Only the 8 most significant bits of
// the AD conversion result are used
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x85;

// SPI initialization
// SPI disabled
SPCR=0x00;

// TWI initialization
// TWI disabled
TWCR=0x00;
```

```

// Alphanumeric LCD initialization
// Connections specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTB Bit 0
// RD - PORTB Bit 1
// EN - PORTB Bit 2
// D4 - PORTB Bit 4
// D5 - PORTB Bit 5
// D6 - PORTB Bit 6
// D7 - PORTB Bit 7
// Characters/line: 16
lcd_init(16);
}

/*-----
SUBROUTINA QUE REALIZA EL PARPADEO CUANDO LA PRESION ESTA FUERA DE 35 O
36 PSI.
-----*/
void muestra(unsigned char x, unsigned char y)
{
lcd_gotoxy(x,y);           //UBICA AL CURSOR DEL LCD EN LA POSICION X Y Y.
itoa(dato,mensaje);       //TRANSFORMA EL DATO DE ENTERO A ASQUI.
lcd_puts(mensaje);        //PONE EN EL LCD EL VALOR DE DATO.
lcd_putsf("PSI");         //AGREGA EN EL LCD LAS LETRAS PSI.
delay_ms(tiempo);         //ESPERA UN TIEMPO.
lcd_gotoxy(x,y);          //PONE AL CURSOR DEL LCD EN LA POSICION X Y Y.
lcd_putsf(" ");           //PONE EN EL LCD TODOS LOS ESPACIOS EN BLANCO PARA
TITILAR.
delay_ms(tiempo);         //ESPERA UN TIEMPO.
lcd_gotoxy(x,y);          //UBICA NUEVAMENTE AL CURSOR DEL LCD EN X Y Y.
lcd_puts(mensaje);        //PONE EN EL LCD EL VALOR DE DATO.
lcd_putsf("PSI");         //AGREGA EN EL LCD LAS LETRAS PSI.
}

/*-----
SUBROUTINA QUE MUESTRA LOS VALORES DE 35 O 36 PSI FIJOS
-----*/
void muestra1(unsigned char x, unsigned char y)
{
lcd_gotoxy(x,y);           //UBICA AL CURSOR DEL LCD EN LA POSICION X Y Y.
itoa(dato,mensaje);       //TRANSFORMA DATO DE ENTERO A ASQUI.
lcd_puts(mensaje);        //PONE EN EL LCD EL VALOR DE DATO.
lcd_putsf("PSI");         //AGREGA EN EL LCD LAS LETRAS PSI.
}

/*-----
PROGRAMA PRINCIPAL
-----*/
void main(void)
{
configuracion();           //LLAMA A SUBROUTINA CONFIGURACION PARA
INICIALIZAR AL MICROCONTROLADOR.
while (1)

```

```

{
dato=read_adc(0)*60/255; //EN LA VARIABLE DATO SE GUARDA EL VALOR DIGITAL DEL
PUERTO ANALOGICO 0.
if (dato>36 || dato<35) //PREGUNTA SI DATO ES MENOR A 35 O MAYOR A 36.
{
muestra(0,0); //SI ES MENOR O MAYOR LLAMA A SUBROUTINA MUESTRA
PARA TITILAR UBICACION LCD 0,0.
}
else
{
muestra1(0,0); //SI ES IGUAL A 35 O 36 LLAMA A SUBROUTINA MUESTRA1
PARA MOSTRAR EL VALOR UBICACION LCD 0,0.
}

dato=read_adc(1)*60/255; //EN LA VARIABLE DATO SE GUARDA EL VALOR DIGITAL
DEL PUERTO ANALOGICO 1.
if (dato>36 || dato<35) //PREGUNTA SI DATO ES MENOR A 35 O MAYOR A 36.
{
muestra(11,0); //SI ES MENOR O MAYOR LLAMA A SUBROUTINA MUESTRA
PARA TITILAR UBICACION LCD 11,0.
}
else
{
muestra1(11,0); //SI ES IGUAL A 35 O 36 LLAMA A SUBROUTINA MUESTRA 1 PARA
MOSTRAR EL VALOR UBICACION LCD 11,0.
}

dato=read_adc(2)*60/255; //EN LA VARIABLE DATO SE GUARDA EL VALOR DIGITAL DEL
PUERTO ANALOGICO 2.
if (dato>36 || dato<35) //PREGUNTA SI DATO ES MENOR A 35 O MAYOR A 36.
{
muestra(0,1); //SI ES MENOR O MAYOR LLAMA A SUBROUTINA MUESTRA
PARA TITILAR UBICACION LCD 0,1.
}
else
{
muestra1(0,1); //SI ES IGUAL A 35 O 36 LLAMA A SUBROUTINA MUESTRA1 PARA
MOSTRAR EL VALOR UBICACION LCD 0,1.
}

dato=read_adc(3)*60/255; //EN LA VARIABLE DATO SE GUARDA EL VALOR DIGITAL DEL
PUERTO ANALOGICO 3.
if (dato>36 || dato<35) //PREGUNTA SI DATO ES MENOR A 35 O MAYOR A 36.
{
muestra(11,1); //SI ES MENOR O MAYOR LLAMA A SUBROUTINA MUESTRA PARA
TITILAR UBICACION LCD 11,1.
}
else
{
muestra1(11,1); //SI ES IGUAL A 35 O 36 LLAMA A SUBROUTINA MUESTRA1 PARA
MOSTRAR EL VALOR UBICACION LCD 11,1.
}
}

```