

Recibido: 08 de Agosto 2020 / Aceptado: 10 de Noviembre 2020 / Publicado: 01 de Enero 2021

Sección: Ciencias de la Ingeniería
Artículo de Investigación Original
<https://doi.org/10.55204/trc.v1i1.6>

Implementación de un prototipo electrónico para registros telemáticos y detección de fallos en motores de automóviles mediante sistema OBD II

Implementation of an electronic prototype for telematic records and fault detection in automobile engines using OBDII system

Andrea Gabriela Montesdeoca Vivanco¹[0000-0002-8667-9979]

¹Facultad de Informática y Electrónica, Escuela Superior Politécnica de Chimborazo. Riobamba, Ecuador
andrea.montesdeoca1994@esPOCH.edu.ec

Resumen. Los dispositivos que componen su construcción fueron adquiridos mediante un análisis, estudio y requerimientos que permitan realizar las diferentes funciones propuestas por dicho prototipo, para ello se consideró el costo, facilidad de adquisición y compatibilidad de software. Se utilizó un sistema OBDII encargado del diagnóstico del vehículo, permitiendo un monitoreo constante y almacenamiento de datos para detección de fallas que puedan presentar y afectar el funcionamiento de los sensores del motor; esto mediante un módulo micro SD. Se fabricó una placa de control a la cual se le integró una tarjeta de desarrollo Arduino Mega encargada de controlar las funciones mediante algoritmos de programación, librerías para cada módulo y de manera directa con el sistema OBDII. El prototipo permite visualizar los datos obtenidos mediante dos plataformas virtuales denominadas Datalogger y Thingspeak. El datalogger almacena los datos obtenidos y procesados por la unidad de control electrónica de motor (ECU) de cada sensor y la muestra mediante un blog de notas. La plataforma Thingspeak propia de Matlab muestra las señales en un entorno analítico del internet de las cosas (IoT) que permite visualizar, agregar y analizar las señales de los sensores del motor directo en una nube de internet debido al módulo GSM 1800L. Las pruebas de funcionamiento, se lograron extraer datos de los principales sensores del vehículo y se logró evidenciar mediante análisis de error absoluto y relativo de los datos obtenidos un resultado entre 1% a 2% de error considerando que mediante este análisis los datos extraídos son aceptables.

Palabras Clave: Sistema OBDII, Unidad Electronica De Control (ECU), Internet De Las Cosas (IoT), Datalogger, Thingspeak.

Abstract. The devices that make up its construction were acquired through an analysis, study and requirements that allow the different functions proposed by said prototype to be carried out, for this, the cost, ease of acquisition and software compatibility were considered. An OBDII system in charge of vehicle diagnosis was used, allowing constant monitoring and data storage to detect faults that may present and affect the operation of the engine sensors; this through a micro SD module. A control board was manufactured to which an Arduino Mega development board was integrated in charge of controlling the functions through programming algorithms, libraries for each module and directly with the OBDII system. The prototype allows to visualize the data obtained through two virtual platforms called Datalogger and Thingspeak. The datalogger stores the data obtained and processed by the electronic engine control unit (ECU) of each sensor and displays it through a blog of notes. Matlab's own Thingspeak platform shows the signals in an analytical environment of the Internet of Things (IoT) that allows to visualize, add and analyze the signals of the engine sensors directly in an internet cloud due to the GSM 1800L module. The performance tests, it was possible to extract data from the main sensors of the vehicle and it was possible to demonstrate by means of absolute and relative error analysis of the data obtained a result between 1% to 2% of error considering that through this analysis the extracted data is acceptable.

Keywords: OBDII System, Electronic Control Unit (ECU), Internet Of Things (IoT), Datalogger, Thingspeak.

Como Citar (APA): Montesdeoca Vivanco, A. G. (2021). Implementación de un prototipo electrónico para registros telemáticos y detección de fallos en motores de automóviles mediante sistema OBD II. *Tesla Revista Científica*, 1(1), 68–87. <https://doi.org/10.55204/trc.v1i1.6>



Los contenidos de este artículo están bajo una licencia de Creative Commons Atribución 4.0 Internacional (CC BY 4.0)
Los autores conservan los derechos morales y patrimoniales de sus obras.

INTRODUCCIÓN

Con el desarrollo tecnológico a nivel mundial, específicamente en el campo de la electrónica como la base fundamental en el mejoramiento y aporte a los demás campos de la ingeniería entre las más comunes: industrial, automotriz y mecánica. Donde se han dado soluciones a un sin número de problemas presentes diariamente a nivel mundial; con respecto al campo automotriz y la evolución de sus sistemas manuales a la emigración de sistemas electrónicos por medio del uso e implementación de módulos y sensores, han ofrecido facilidades al momento de análisis de fallas y errores en un vehículo automotriz.

La creciente sofisticación de los componentes electrónicos en los vehículos modernos ha hecho que la conducción sea más agradable, cómoda y, en muchos casos, más segura. La interconectividad de los sensores y actuadores electrónicos, y su capacidad de configuración, ayuda a afinar la experiencia de conducción, lo que a su vez aumenta la prevalencia y la sofisticación de estos componentes. (Sharma, et al 2020)

La computación y las redes automotrices surgieron por necesidad cuando los controles mecánicos del motor utilizados durante la década de 1970 no pudieron cumplir con las nuevas y estrictas regulaciones de emisiones (Pelkmans, et al 2003). Los primeros módulos se utilizaron para el control de vehículos locales, aunque los investigadores proféticos imaginaron que las redes de controladores permitían la resolución colaborativa de problemas (Lesser, & Corkill, 1983).

En su investigación (Ambrosio Lázaro & Sánchez Gaspariano, 2017) afirma que el desarrollo automotriz se ha centrado en la implementación de materiales ligeros, miniaturización, inteligencia, movilidad y energía, para ello su función con sistemas electrónicos mediante módulos y algoritmos de programación, es por ello que la electrónica dentro del área automotriz toma una gran importancia en la creación de tecnologías en los vehículos; esta evolución tecnológica cada vez sustituyen los sistemas mecánicos a electrónicos, un claro ejemplo es la interacción de una memoria o ECU (unidad de control electrónica de motor) la misma que establece un control completo a lo sensores instalados en el vehículo.

El puerto de diagnóstico a bordo (OBD) es un puerto de conexión que cualquiera puede usar para recopilar información sobre las emisiones, el kilometraje, la velocidad y los datos de los componentes de un vehículo. Hay dos estándares OBD, a saber, OBD-I y OBD-II. El OBD-I

se introdujo en 1987 pero tenía muchas fallas, por lo que fue reemplazado por OBD-II introducido en 1996 (Kalmeshwar y Prasad, 2017).

El puerto OBD-II debe encontrarse en casi cualquier vehículo moderno, y los CAV no son una excepción. Los puertos OBD modernos pueden proporcionar datos en tiempo real (Lin et al., 2005). OBD también proporciona una vía para adquirir datos de las unidades de control electrónico de CAV y posiblemente modificar el software integrado en esas unidades de control. Muchos fabricantes también utilizan puertos OBD para realizar actualizaciones de firmware (Checkoway et al., 2011).

Además de un dispositivo PassThru , también se puede usar un dispositivo ECOM para interactuar con el puerto OBD-II y leer y escribir en el bus CAN, aunque es posible que se requiera un adaptador para la compatibilidad del conector. (Sharma, et al 2020)

El puerto de diagnóstico a bordo (OBD-II) lo utilizan los técnicos cuando realizan el mantenimiento de un vehículo y, por esta razón, tiene acceso a todos los buses CAN dentro de un vehículo. Todos los vehículos en los EE. UU. Deben ser compatibles con el estándar PassThru (Checkoway et al., 2011)., que es una API basada en Windows que proporciona una interfaz de software para comunicarse con las redes internas de un vehículo y, por lo general, se implementa mediante un dispositivo PassThru que se conecta directamente al puerto OBD II del vehículo.

Las unidades de control electrónico (ECU) son sistemas electrónicos integrados que controlan otros subsistemas en un vehículo. Todos los vehículos modernos utilizan ECU para controlar las funciones de los vehículos adquiriendo señales electrónicas de otros componentes, así como procesando y enviando señales de control. Algunas ECU importantes son el módulo de control de frenos, el módulo de control del motor, los sistemas de monitoreo de presión de los neumáticos y las unidades de medición inercial. Sus funcionalidades son las siguientes.

El módulo de control de frenos recopila datos de los sensores de velocidad de las ruedas y el sistema de frenos, y también procesa los datos para determinar si se libera o no la presión de frenado en tiempo real (Kassakian, 1996). El módulo de control del motor controla el combustible, el aire y la chispa, además de recopilar datos de muchos sensores alrededor del vehículo para garantizar que todos los componentes estén dentro de un rango de funcionamiento normal (Kassakian, 1996).

Los sistemas de monitoreo de presión de llantas recopilan datos de los sensores dentro de las llantas y determinan si la presión de las llantas está en niveles ideales. Los Estados Unidos han exigido legalmente que todos los vehículos estén equipados con sistemas de control de la presión de los neumáticos desde 2007 (Singh, Kingsley, & Chen, 2009), y la Unión Europea emitió el mismo reglamento en 2012 (European Parliament, Council of the European Union, 2009).

Las unidades de medición inercial recopilan datos de acelerómetros, magnetómetros y giroscopios y calculan la velocidad, la aceleración, la tasa angular y la orientación del vehículo. Estos cálculos son fundamentales para los CAV porque sirven como entradas para el funcionamiento de un sistema de conducción automatizado seguro (Jitpakdee, & Maneewarn, 2008). Por ejemplo, un cambio en la pendiente de la carretera cambiaría la velocidad angular y la orientación de un CAV, y el sistema de conducción automatizado puede emitir un ajuste en la velocidad de un vehículo para mantener operaciones seguras.

Los CAV involucran una mayor cantidad de ECU que un vehículo no automatizado (nivel 2 de SAE e inferior) porque poseen muchos más sensores y requieren muchos más cálculos para tomar decisiones autónomas al conducir. Los lectores pueden pensar en las ECU de los CAV como miniordenadores, cada uno desempeña un papel específico y colabora con otros para realizar la conducción autónoma. Es habitual ver colaboraciones complejas entre ECU (Koscher, et al, 2010).

Los modelos de ataque y las estrategias de defensa en las ECU se analizan en la sección III-B. Las comunicaciones entre las ECU se producen en las redes de área del controlador, que se analizarán a continuación. Red de área del controlador (CAN). Las ECU generalmente se conectan a través de una CAN. En un vehículo, la CAN es una red central para conectar las ECU juntas para que puedan comunicarse entre sí. Un bus CAN está estructurado típicamente como un sistema de red de dos hilos y semidúplex que puede admitir comunicaciones de alta velocidad (HPL, 2002).

Los mayores beneficios de las CAN son la baja cantidad de cableado y la ingeniosa prevención de la pérdida de mensajes y la colisión de mensajes (HPL, 2002). En los CAV, los paquetes de red se transmiten a todos los nodos de la red CAV y los paquetes no contienen un campo de autenticación o un campo de identificación de fuente (Thing, & Wu, 2016). Por lo tanto, un nodo comprometido puede recopilar todos los datos que se transfieren a través de la

red y transmitir datos maliciosos a otros nodos, lo que hace que todo el CAN sea vulnerable a los ciberataques. Los modelos de ataque y las estrategias de defensa de las CAN se analizan en la sección III-C.

Según una investigación de la revista Motor (SERVICES, 2017) no se puede negar que la electrónica está suponiendo un avance tecnológico sin precedentes, posiblemente en los futuros libros de historia se le dé la misma importancia, si no más que, por ejemplo, a la máquina de vapor y su revolución industrial, dada la intensa influencia que está teniendo en nuestro modo de vida, los sistemas electrónicos sustituyen complejos sistemas mecánicos por elementos más precisos, más pequeños y generalmente más económicos. Incluso gracias a la electrónica es posible crear sistemas capaces de realizar trabajos de tal complejidad que ni siquiera podrían existir en versión “analógica”.

Por lo cual nos surgió nuestra problemática:

¿Existen dispositivos capaces de efectuar un análisis y diagnóstico de los sensores del motor de un automóvil para la detección de fallos mediante sistema OBD II?

Mediante este proyecto de grado se pretende desarrollar un prototipo electrónico conectado mediante un sistema OBD II para analizar y diagnosticar la unidad electrónica de control del motor, es decir, el funcionamiento y fallo de los sensores cuyos datos censados serán mostrados en un datalogger.

El estudiante Cando de la Escuela Superior Politécnica desarrollo un simulador para el diagnóstico de la unidad de control electrónico de motor (ECU), mediante las diferentes tecnologías existentes en el país, dicho dispositivo consta de 4 etapas las cuales mediante un analizador examinan las fallas existentes en los sensores del motor, las muestras y señales de cada sensor se interpretan y visualizan mediante una pantalla HMI y un osciloscopio integrado al sistema (ROBERTO, 2017).

Según un estudio realizado con respecto a los fallos presentes en los sistemas electrónicos de los vehículos automotrices en general, se consideró que los establecimientos, concesionarios y centro especializados en su mayoría no constan con un sistema capaz de detectar fallos mediante registro telemático.

En la actualidad los analizadores de sistemas electrónicos en un vehículo empleados al diagnóstico de los diferentes sensores integrados al motor y cuyos datos enviados a la unidad de control electrónica no constan de un análisis con registro telemático y en muchos de los

Implementación de un prototipo electrónico para registros telemáticos y detección de fallos en motores de automóviles mediante sistema OBD II

casos la adquisición de datos no es reflejada y conservada en entornos de visualización tales como base de datos y datalogger. Es por ello, que se pretende desarrollar e instalar un prototipo que pueda realizar un análisis constante del funcionamiento de los sensores automotrices y lleve un registro telemático para el usuario del vehículo o de los centros especializados mediante sistema OBD II, en caso de presentar alguna falla en las mediciones muestre una alerta para realizar la respectiva revisión. Los sensores para evaluar en este trabajo de investigación son: SP, VELOCIDAD, IMPULSOS, OXÍGENO, CMP, CKP, TPS, IAT, MAP, MAF.

Se planteo con ello los siguientes objetivos: Conocer los fundamentos básicos de funcionamiento de los sensores del motor a ser analizados para detectar fallos en el vehículo. Especificar los requerimientos que debe cumplir el prototipo para la adquisición telemétrica de datos mediante el sistema OBD II. Determinar el software, hardware y diseño electrónico adecuados para el diseño propuesto del prototipo. Realizar las evaluaciones del prototipo electrónico implementado para verificar el cumplimiento de los requerimientos planteados.

MATERIALES Y MÉTODOS

El proyecto se llevó a cabo siguiendo una metodología conformada por 5 fases como lo muestra la Figura 1.

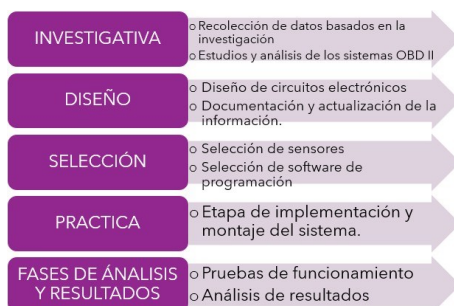


Figura 1 Metodología planteada para el desarrollo del proyecto.

Y a su vez, cada fase contiene procedimientos internos que posibilitan el desarrollo del proyecto paso a paso.

2.1 Etapas del prototipo

Las etapas consisten en la selección de las herramientas adecuadas que provean el mejor funcionamiento y eficiencia al prototipo. Para el desarrollo del prototipo se establecen diferentes etapas que cumplen con los requerimientos necesarios para la implementación del proyecto, tal como se muestra en la siguiente figura 2.

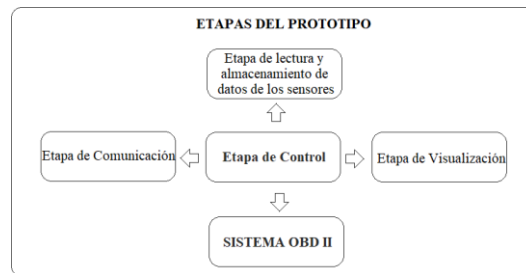


Figura 2 Etapas del Prototipo

La implementación del prototipo consta de 5 etapas de desarrollo las cuales son:

- **Etapa de control:** Es la encargada de comandar el funcionamiento del prototipo mediante un microcontrolador y algoritmos de programación establecidos para su funcionamiento
- **Etapa de visualización:** Se encarga de mostrar los datos obtenidos de los sensores instalados en el motor y fallas.
- **Etapa de comunicación:** Mediante un módulo GSM permite comunicar los resultados del sistema a una plataforma de sitio web.
- **Etapa de Lectura y Almacenamiento de datos de los sensores:** Se encarga de almacenar los datos receptados por los sensores.
- **Sistema OBD II:** Es el sistema principal dentro del funcionamiento del prototipo, su función es realizar una conexión y comunicación con los sensores internos del motor.

2.2 Diseño del prototipo

Basado en los objetivos planteados al inicio del proyecto, en la figura 2 se muestra un diagrama en el cual se detalla las etapas del diseño del prototipo.

Para el diseño de la placa o tarjeta electrónica que permitirá la puesta en marcha del prototipo se realiza una estructura, en la cual se muestra todos los elementos y módulos electrónicos que permitan su diseño, la figura 3 muestra a continuación la estructura de la tarjeta electrónica del prototipo.

Implementación de un prototipo electrónico para registros telemáticos y detección de fallos en motores de automóviles mediante sistema OBD II

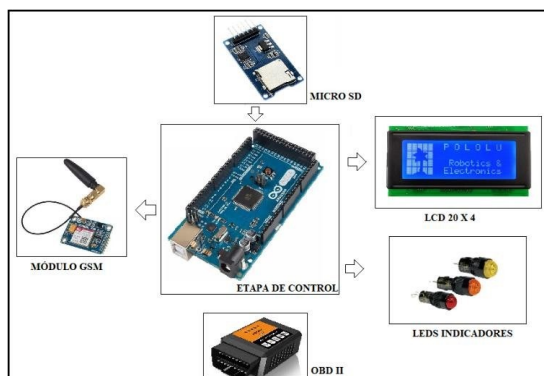


Figura 3 Estructura de la tarjeta electrónica del prototipo

Como se muestra en la figura 3 la estructura del diseño de la placa electrónica muestra los módulos más importantes para su implementación.

Parámetros de diseño

Circuito esquemático general de la tarjeta electrónica

Culminada todos los diseños de las diferentes etapas mediante software proteus se unifican todos los diseños para mostrar un resultado general del circuito esquemático, dentro de ellos se establece los siguientes:

- Arduino – Modulo micro SD.
- Arduino – Modulo GSM.
- Arduino – Pantalla LCD.
- Arduino – Sistema OBDII.

El entorno ISIS de Proteus muestra un diseño general que será utilizado como muestra para un entorno ARES que determina el diseño PCB, la siguiente figura 4 muestra el circuito esquemático general de la tarjeta electrónica.

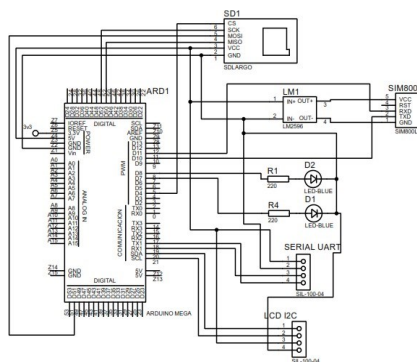


Figura 4 Circuito esquemático de la tarjeta electrónica

2.3 Diseño PCB del circuito electrónico

Culminado el proceso de diseño esquemático, se realiza el PCB propio del software Proteus, a la cual se denomina ARES cuya funcionalidad es crear un PCB del diseño esquemático realizado en ISIS, luego el sistema permite implementar un circuito o placa electrónica.

Diseño de pistas

Se realiza el cálculo de pistas tomando en cuenta lo siguiente:

Ecuación 1: Ecuación para el cálculo del grosor de las pistas Ec.1.

$$lonza/pie^2 = 35 \text{ micras de espesor (1)}$$

Para el cálculo de la pista se tomó en cuenta la siguiente fórmula establecida a continuación:

En donde:

I= Corriente Máxima

ΔT = Diferencia de temperatura

K1, K2, K3= Constantes para el calculo

Los valores K1, K2, K3 vienen establecidos de manera que existen diseños de pistas PCB a dos caras en la baquelita, de esta manera se realizan los siguientes cálculos tomando en cuenta que 1A es la corriente máxima que debe soportar las pistas.

Datos:

$$I = 1 \text{ AMP}$$

$$\Delta T = 25^\circ \text{C}$$

$$\text{GROSOR} = 1 \text{ oz /ft}^2$$

$$K1 = 0.0647$$

$$K2 = 0.4281$$

$$K3 = 0.6732$$

Ecuaciones: Ecuación del área Ec.2. y ancho de la pista Ec.3.

$$AREA = \left(\frac{I}{K1 * \Delta T^{K2}} \right)^{\frac{1}{K3}}$$

$$AREA = \left(\frac{1}{0.0647 * 25^{0.4281}} \right)^{\frac{1}{0.6732}}$$

$$AREA = 7.539$$

$$ANCHO = \frac{AREA}{GROSOR * 1.378}$$
$$ANCHO = \frac{7.539}{1 * 1.378}$$
$$ANCHO = 5.471th$$

Una vez calculado el ancho de las pistas, se toma en cuenta que (th) es la simbología establecida por el software Proteus y que en este caso es equivalente al ancho; la siguiente figura 5 muestra el diseño PCB del circuito electrónico.

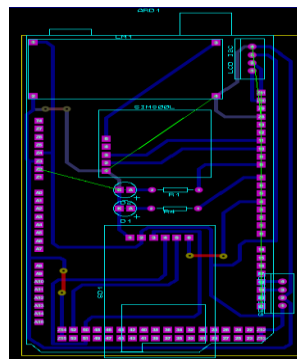


Figura 5. Diseño PCB del circuito electrónico

Visualización 3D

Se realiza una visualización 3D de la placa del circuito esquemático y PCB para apreciar la correcta instalación de los elementos y módulos electrónicos. A continuación, en la Figura 6 se visualiza en 3D del circuito electrónico.

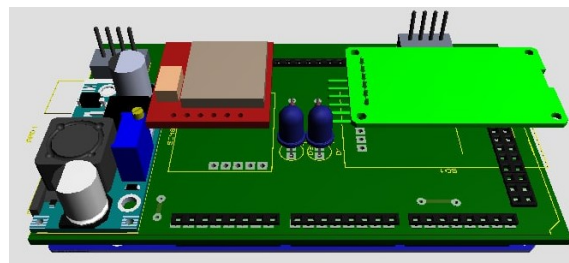


Figura 6. Visualización 3D del circuito electrónico

RESULTADOS Y DISCUSIÓN

3.1 Implementación del prototipo

En esta etapa se muestra el proceso de construcción física del prototipo, los diseños mediante software Proteus sirven como base principal para la construcción mediante la utilización de módulos y dispositivos electrónicos.

Construcción de la placa PCB

Para la construcción de la placa PCB se utiliza una máquina CNC router adecuada para la fabricación de circuitos electrónicos, este sistema de construcción es el más utilizado y



actualizado, la figura 7 muestra el proceso de corte de la placa PCB.

Figura 7. Proceso de corte de la placa PCB.

El resultado final del corte muestra todo el diseño realizado para la colocación de cada módulo y elementos electrónicos, la siguiente figura 8 muestra el diseño final de la placa PCB:

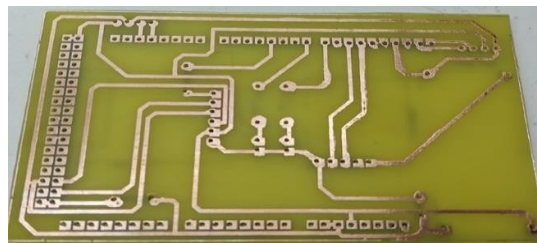


Figura 8. Diseño final de la placa PCB

Colocación de elementos y módulos electrónicos

Al resultado final de la placa PCB se añade los diferentes módulos y elementos electrónicos adquiridos tales como:

- Arduino Mega.
- Módulo GSM.
- Modulo SD.
- LCD 20 X 7.
- Leds indicadores.
- Bornes para puerto OBDII.

Las siguientes Figuras 9 y 10 muestran la implementación de los elementos electrónicos en la placa PCB.

Implementación de un prototipo electrónico para registros telemáticos y detección de fallos en motores de automóviles mediante sistema OBD II

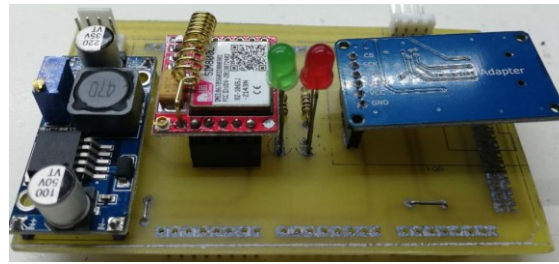


Figura 9 Implementación de elementos electrónicos

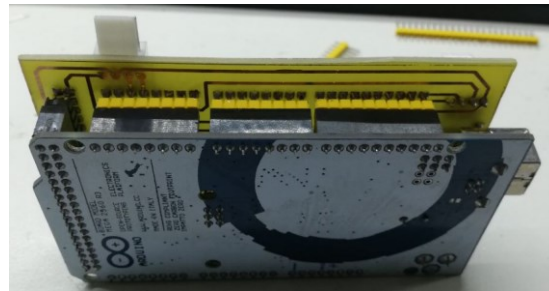


Figura 10 Arduino en la placa PCB

Armado total del prototipo

Se ubica la tarjeta de control electrónica y el módulo de visualización LCD en la caja de protección contra factores externos, con este proceso se da más estética al resultado final y los módulos que constituyen al dispositivo están totalmente protegidos, la figura 11 muestra la ubicación de los módulos.



Figura 11 Ubicación de los módulos

Prototipo final

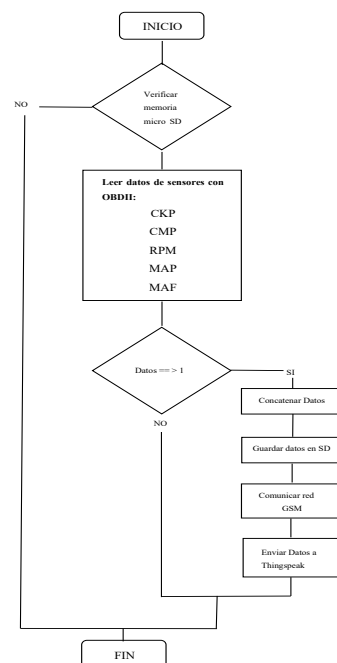
Luego de diferentes procesos y etapas, se muestra en la figura 12 el resultado del prototipo final electrónico para registros telemáticos y detección de fallos del motor mediante sistemas OBDII.



Figura 12 Prototipo Final

3.2 Diseño del algoritmo de programación

Para el funcionamiento del prototipo se diseña un algoritmo de programación que permita ejecutar las diferentes funciones establecidas por el dispositivo, para ello se muestra en la figura



13 el diagrama de flujo referente al algoritmo de programación del prototipo.

Figura 13 Diagrama de flujo referente al algoritmo de programación del prototipo

3.3 Funcionamiento del prototipo

Para empezar con las pruebas de funcionamiento vamos a entender la constitución externa del prototipo de la siguiente manera:

Pantalla principal

La pantalla principal muestra los datos requeridos y emitidos por cada sensor del vehículo, se puede apreciar los datos de:

Implementación de un prototipo electrónico para registros telemáticos y detección de fallos en motores de automóviles mediante sistema OBD II

- Tipo de sensor
- Valor del sensor

La figura 14 a continuación muestra la pantalla principal del prototipo.



Figura 14 Pantalla principal del prototipo

Leds indicadores

En el dispositivo se visualiza 2 tipos de led indicadores, los cuales cumplen con las siguientes funciones:

- Led verde: Refleja el encendido del prototipo, comprobando su correcto funcionamiento.
- Led azul: Indica la correcta lectura de datos.

Puerto USB de programación Arduino

El prototipo está integrado por un puerto de comunicación USB, que sirve para establecer comunicación entre ordenado y placa electrónica para su programación, en la cual se puede realizar varios cambios o ingresar un nuevo algoritmo de programación según requerimientos del usuario.

Puerto de Alimentación

El prototipo consta de un puerto de alimentación común, su alimentación de entrada para el funcionamiento es de 12 VDC, este puerto es utilizado cuando se manipula el prototipo de manera externa,

Ranura para micro SD

Se integró una ranura para micro SD, la cual es la encargada de realizar el almacenamiento de los sensores en el vehículo, la tarjeta micro SD es de fácil adquisición y puede ser colocada en un adaptador para que su información sea extraída a un ordenador.

Puerto de conexión para sistema OBD II

Dependiendo del vehículo la ubicación de la conexión del puerto puede cambiar de lugar, el puerto OBD II suele estar ubicado en la zona de los pies del conductor, ya sea debajo del volante o en la caja de fusibles, otros fabricantes sitúan la conexión OBD II en la parte del cenicero o

incluso en el asiento del copiloto; mediante la tarjeta electrónica y el Arduino se adecuó un puerto de salida conectado hacia el sistema OBD II.

La implementación y verificación de funcionamiento de un prototipo electrónico para registros telemáticos y detección de fallos del motor mediante sistema OBD II, se procedieron a realizar las diferentes pruebas de obtención de datos del prototipo.

3.4 Análisis de funcionamiento

Se realizó un análisis para determinar el correcto funcionamiento y efectividad del prototipo; el dispositivo se instaló en un vehículo Híbrido Toyota modelo Camry modelo 2017, el módulo de conexión vehículo y sistema OBDII son de fácil de conexión. Se encuentra situado en la parte baja del volante del conductor, la figura 15 muestra la posición de conexión al puerto OBDII del vehículo.



Figura 15 Posición de conexión a la unidad de control electrónica de motor

El vehículo Toyota Camry en el cual se efectuaron la prueba, arrojó como resultado la captura de 10 datos importantes los cuales son:

- Engine RPM (rpm).
- Engine coolant temperature (°C).
- Engine oil temperature (°C).
- Intake temperature (°C).
- flow pressure (grams/s).
- Barometric pressure (kPa).
- Vehicle speed (km/h).
- Vehicle running distance (km).
- Throttle position (%).
- Ambient temperature (°C).

3.5 Funcionamiento del prototipo

Implementación de un prototipo electrónico para registros telemáticos y detección de fallos en motores de automóviles mediante sistema OBD II

Para realizar la adquisición de datos el vehículo debe estar en modo encendido con el motor en ON, cada uno de los principales sensores funcionan cuando el vehículo se encuentra en movimiento, la toma de datos se la realiza en tiempo real. En la figura 16 se muestra el funcionamiento del prototipo.



Figura 16 Funcionamiento del prototipo.

Visualización de resultados en Datalogger

Los datos enviados por la unidad de control electrónica de motor (ECU) de los sensores automotrices son almacenados en una tarjeta de memoria microSD, la ventaja de la tarjeta es su extracción y conexión en un computador; los datos adquiridos se encuentran almacenados en texto plano con sus variables y nombres separados por comas, para visualizar los datos en un computador se lo hace mediante blog de notas o Excel de Windows.

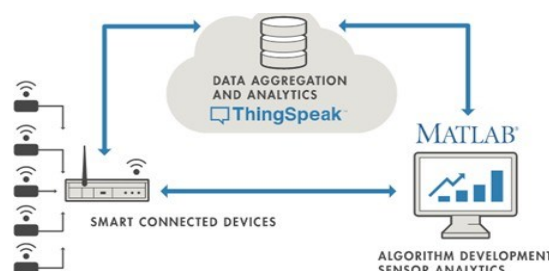
Visualización de datos en Excel

Los datos de los sensores almacenados en texto plano se los procesa en el software Excel con la finalidad de tomar gráficas de funcionamiento de cada sensor. Al cargar el archivo a Excel, en la pantalla se debe delimitar los caracteres a separar.

El resultado de los procesos anteriores muestra el datalogger con los nombres y las variables organizadas en columnas,

Visualización de resultados en Thingspeak

La plataforma Thingspeak almacena los datos emitidos por los sensores y se las puede visualizar mediante la red internet en cualquier lugar del mundo, para usar la plataforma es necesario iniciar sesión en MatWorks. La figura 17 muestra el proceso de los datos por la red hasta la plataforma Thingspeak.



Implementación de un prototipo electrónico para registros telemáticos y detección de fallos en motores de automóviles mediante sistema OBD II

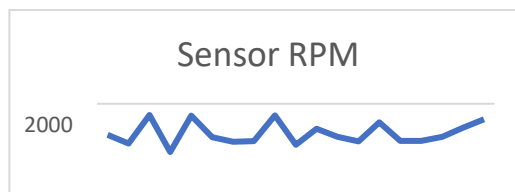
Figura 17 Proceso de los datos por la red hasta la plataforma Thingspeak

Adquisición de datos del sensor RPM

La figura 25 visualiza los datos extraídos con respecto al sensor RPM, el dato nos permite comprobar si las revoluciones actuales a las que giran el árbol de levas y el cigüeñal muestran el rendimiento de funcionalidad del motor.

Datos del sensor RPM almacenados en el Datalogger

El sensor muestra un rango de funcionamiento óptimo. Según (Lauch, 2018)[4] lo normal leído por un scanner es de 600 a 1800 RPM, afirmando su correcta funcionalidad, la figura 18



muestra el comportamiento del sensor RPM según sus datos obtenidos.

Figura 18 Etiqueta en eje Y rangos de 600 a 1800 RPM

- Cálculo del promedio, error absoluto y relativo de los datos del sensor RPM

El promedio del número de RPM censados es de 1044,84., El margen de error absoluto es de 305,484 y relativo del 29,23%.

- Datos del sensor RPM visualizados en la plataforma Thingspeak.

La plataforma permite una visualización de datos interpretados en gráficas, mediante comunicación GSM los datos son enviados a una nube denominada Thingspeak propia de Matlab. En la figura 19 se muestra los datos de RPM en la plataforma Thingspeak.



Figura 19 Sensor RPM en Thingspeak

CONCLUSIONES Y DISCUSION

Se implementó un prototipo para registros telemáticos y detección de fallos del motor mediante sistema OBD II para vehículos de marca Toyota Camry modelo 2017 para detectar fallos en el motor, mediante el uso del hardware y software adecuado.

Implementación de un prototipo electrónico para registros telemáticos y detección de fallos en motores de automóviles mediante sistema OBD II

Mediante el análisis e investigación se conoció el funcionamiento de los sensores automotrices incorporados en el motor de un vehículo, el control electrónico (ECU) que permiten medir las variables de estudio de esta investigación y determinar posibles fallas.

Para el almacenamiento de los datos obtenidos como respuesta del procesamiento de la ECU al conectarse el sistema OBDII con la tarjeta Arduino, permite que estos se almacenen en la nube (web) para tener un registro del funcionamiento del motor del vehículo cuando la cobertura de la red le permite.

De las pruebas realizadas del funcionamiento del prototipo al momento de adquirir los datos de los sensores se determinó que el sensor RPM operan en un rango normal de 600 a 1800 RPM (mediante el datalogger), del sensor de temperatura del refrigerante, aceite, admisión y ambiente, muestran un error relativo del 1% y 2% que se debe a la temperatura ambiente y la producida por el funcionamiento del motor.

Los resultados obtenidos del sensor de distancia y velocidad, tiene un error relativo del 10,55% y 27.13% que se encuentra dentro de los rangos permitidos, tomando en cuenta que estos sensores toman datos de manera variable. El consumo de corriente del prototipo es de 0.187A (amperios) que no afecta a la carga de la batería del vehículo.

Del análisis de costo en la implementación del prototipo para registros telemáticos y detección de fallos del motor, el sistema OBDII, es el que tiene mayor costo dentro de la implementación, por lo que puede ser adquirido por los talleres automotriz.

REFERENCIAS

- Ambrosio Lázaro , R., & Sánchez Gaspariano, L. A. (04 de 06 de 2017). *saberesyciencias.com.mx*. <https://saberesyciencias.com.mx/2017/06/04/la-importancia-de-la-electronica-en-el-desarrollo-del-automovil/>
- Checkoway, S., McCoy, D., Kantor, B., Anderson, D., Shacham, H., Savage, S., ... & Kohno, T. (2011, August). Comprehensive experimental analyses of automotive attack surfaces. In *USENIX Security Symposium* (Vol. 4, No. 447-462, p. 2020).
- European Parliament, Council of the European Union, (2009). "Regulation (EC) No 661/2009 of the European Parliament and of the Council of 13 July 2009 concerning type-approval requirements for the general safety of motor vehicles, their trailers and systems,

- components and separate technical units intended therefor.” Official Journal of the European Union.
- HPL, S. C. (2002). Introduction to the controller area network (CAN). *Application Report SLOA101*, 1-17.
- Jitpakdee, R., & Maneewarn, T. (2008). Neural networks terrain classification using inertial measurement unit for an autonomous vehicle. In *2008 SICE Annual Conference* (pp. 554-558). IEEE.
- Kalmeshwar, M., & Prasad, K. N. (2017, December). Development of On-Board Diagnostics for Car and it's Integration with Android Mobile. In *2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)* (pp. 1-6). IEEE.
- Kassakian, J. G., Wolf, H. C., Miller, J. M., & Hurton, C. J. (1996). Automotive electrical systems circa 2005. *IEEE spectrum*, 33(8), 22-27.
- Koscher, K., Czeskis, A., Roesner, F., Patel, S., Kohno, T., Checkoway, S., ... & Savage, S. (2010). Experimental security analysis of a modern automobile. In *2010 IEEE symposium on security and privacy* (pp. 447-462). IEEE.
- Lesser, V. R., & Corkill, D. G. (1983). The distributed vehicle monitoring testbed: A tool for investigating distributed problem solving networks. *AI magazine*, 4(3), 15-15.
- Lin, C. E., Li, C. C., Yang, S. H., Lin, S. H., & Lin, C. Y. (2005, February). Development of on-line diagnostics and real time early warning system for vehicles. In *2005 Sensors for Industry Conference* (pp. 45-51). IEEE.
- Pelkmans, L., Hultén, S., Cowan, R., Azkarate, G., & Christidis, A. (2003). Trends in vehicle and fuel technologies: review of past trends. *Inst. for Prospective Technologies Studies, JRC Report EUR 20746 EN*.
- Roberto, C. C. (2017). <http://dSPACE.espace.edu.ec/>. Obtenido de <http://dSPACE.espace.edu.ec/bitstream/123456789/8958/1/108T0222.pdf>
- Services, W. B. (12 de 07 de 2017). *motorpasion*. Obtenido de <https://www.motorpasion.com/n/hasta-que-punto-la-electronica-es-la-nueva-mecanica-del-motor>

Implementación de un prototipo electrónico para registros telemáticos y detección de fallos en motores de automóviles mediante sistema OBD II

Sharma, C., Moylan, S., Vasserman, E. Y., & Amariuca, G. T. (2020). Review of the Security of Backward-Compatible Automotive Inter-ECU Communication. *IEEE Access*, 9, 114854-114869.

Singh, S., Kingsley, K., & Chen, C. L. (2009). *Tire pressure maintenance—a statistical investigation* (No. HS-811 086).

Thing, V. L., & Wu, J. (2016). Autonomous vehicle security: A taxonomy of attacks and defences. In *2016 IEEE International Conference on Internet of Things (IThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)* (pp. 164-170). IEEE.

FINANCIACIÓN

Los autores no recibieron financiación para el desarrollo de la presente investigación.

CONFLICTO DE INTERESES

Los Autores declaran que no existe conflicto de intereses

CONTRIBUCIÓN DE AUTORÍA

Autor	Montesdeoca Vivanco, A. G.
Participar activamente en:	
Planificación y diseño	X
Adquisición de fondos	X
Administración del proyecto	X
Redacción –borrador original	X
Redacción –revisión y edición	X
Interpretación y validación de resultados	X
La discusión de los resultados	X
Revisión y aprobación de la versión final del trabajo.	X

RECONOCIMIENTO A REVISORES:

La revista reconoce el tiempo y esfuerzo del editor Juan Carlos Santillán L., y de revisores anónimos que dedicaron su tiempo y esfuerzo en la evaluación y mejoramiento del presente artículo.