



# **UNIVERSIDAD DEL AZUAY**

**Departamento Posgrados**

**Maestría en Sistemas Vehiculares**

**“Implementación de un modelo de protocolo CAN en un automóvil  
Volkswagen Jetta 2009 mediante Simulink”.**

**Trabajo de graduación previo a la obtención del título de Magister en  
Sistemas Vehiculares**

**Maestrante:**

**Héctor Fabián Garcia Calle.**

**Director de Tesis:**

**Dr. Efrén Esteban Fernández Palomeque.**

**Cuenca - Ecuador**

**Febrero 2020**

### **DEDICATORIA.**

Este trabajo de investigación se lo dedico a mis tíos, Elsa Calle y Ramiro Andrade, por haberme brindado su entera confianza durante el tiempo que ha tomado mi educación. Además, a mi hermano menor, Leonardo Garcia, por haber velado por la salud de un ser querido y como una muestra de que a pesar de las adversidades en las que nos encontremos, siempre es posible lograr nuestras metas con empeño y sacrificio.

**Héctor Garcia Calle.**

## **AGRADECIMIENTO.**

Al finalizar este trabajo de investigación quiero agradecer a mis padres, Ana Calle y Lauro Garcia, y demás familiares por haberme apoyado a lo largo de esta etapa de estudios. Además, al Instituto Superior Tecnológico Simón Bolívar, ya que sin su comprensión y apoyo durante el último año y medio no hubiese sido posible finalizar este tema de investigación. Por último, agradezco a María Balvoa, por haberme ayudado durante el transcurso de esta última etapa de mis estudios.

**Héctor Garcia Calle.**

### Resumen:

En décadas pasadas los vehículos con sistema de inyección electrónica incorporaban un módulo de control electrónico (ECU). Sin embargo, dado la complejidad de los mismos ha sido necesario la implementación de varias ECU's y sensores para los diferentes sistemas electrónicos. El presente documento pretende la implementación de un método de diagnosis mediante la verificación de elementos que varían en la trama de información del protocolo de comunicación CAN (Controller Area Network) de un vehículo Volkswagen Jetta 2009 para reducir el tiempo de diagnóstico. La información de la red de comunicación recolectada en bases de datos, mediante una tarjeta de adquisición National Instruments (NI), permitió analizar los data frames en tiempo real, mientras el módulo PCM (Power-Train Control Module) era manipulado. Los datos obtenidos en función de generar fallas intencionales se compararon con los estados en funcionamiento normal, lo que permitió reconocer los bytes que cambiaban en el momento en el cual el vehículo se encontraba en situación de falla. Además, se utilizó el software Matlab para desarrollar un modelo de red CAN mediante su plataforma Vehicle Network Toolbox (VNT) para verificar el comportamiento de la información en tiempo real y cuyos resultados se validaron mediante el prototipo de red CAN.

Abstract:

In past decades, vehicles with an electronic injection system incorporated an electronic control unit (ECU). However, given their complexity, it has been necessary to implement several ECUs and sensors for different electronic systems. This document studied the implementation of a diagnostic method by verifying elements that vary in the information frame of the CAN (Controller Area Network) communication protocol of a 2009 Volkswagen Jetta in order to reduce the diagnostic time. The information of the communication network was collected in databases by means of a National Instruments (NI) acquisition card, this allowed to analyze the data frames in real time while the PCM (Power-Train Control Module) module was manipulated. The data obtained in function of generating intentional failures were compared with the states in normal operation, this allowed to recognize the bytes that changed at the time when the vehicle was in a fault situation. In addition, Matlab software was used to develop a CAN network model through its Vehicle Network Toolbox (VNT) platform to verify the behavior of information in real time. The results were validated using the CAN network prototype.


Ing. Héctor Fabián García Calle.

Author

Dr. Efrén Fernández Palomeque.

Thesis Director



  
Translated by  
Ing. Paúl Arpi

# IMPLEMENTACIÓN DE UN MODELO DE PROTOCOLO CAN EN UN AUTOMÓVIL VOLKSWAGEN JETTA 2009 MEDIANTE SIMULINK.

Héctor García  
Universidad del Azuay  
Escuela de Mecánica Automotriz  
Cuenca, Ecuador  
hega@es.uazuay.edu.ec

Efrén Fernández  
Universidad del Azuay  
Escuela de Mecánica Automotriz  
Cuenca, Ecuador  
efernandez@uazuay.edu.ec

**Resumen**— En décadas pasadas los vehículos con sistema de inyección electrónica incorporaban un módulo de control electrónico (ECU). Sin embargo, dado la complejidad de los mismos ha sido necesario la implementación de varias ECU's y sensores para los diferentes sistemas electrónicos. El presente documento pretende la implementación de un método de diagnosis mediante la verificación de elementos que varían en la trama de información del protocolo de comunicación CAN (Controller Area Network) de un vehículo Volkswagen Jetta 2009 para reducir el tiempo de diagnóstico. La información de la red de comunicación recolectada en bases de datos, mediante una tarjeta de adquisición National Instruments (NI), permitió analizar los data frames en tiempo real, mientras el módulo PCM (Power-Train Control Module) era manipulado. Los datos obtenidos en función de generar fallas intencionales se compararon con los estados en funcionamiento normal, lo que permitió reconocer los bytes que cambiaban en el momento en el cual el vehículo se encontraba en situación de falla. Además, se utilizó el software Matlab para desarrollar un modelo de red CAN mediante su plataforma Vehicle Network Toolbox (VNT) para verificar el comportamiento de la información en tiempo real y cuyos resultados se validaron mediante el prototipo de red CAN.

**Palabras Clave**— CAN bus, Matlab, Simulación, Módulo de control electrónico.

## I. INTRODUCCIÓN

El avance de la tecnología automotriz ha permitido la implementación de módulos de control electrónicos en el automóvil, siendo necesario su análisis cuando el vehículo falla [1]. Los dispositivos de diagnóstico se centran en el diagnóstico de la ECU del motor, desarrollado para controlar las emisiones producidas por la combustión [2]. No obstante, la cantidad de ECU's se han incrementado dependiendo de los sistemas que el automotor posee (PCM, Airbag, etc.) [3]. Además, diversos sensores se han incorporado al vehículo para generar la información necesaria para regular el funcionamiento del mismo. Por tal razón, en la década de los 80's Robert Bosch creó el CAN bus [4], que hace referencia a la transmisión serial de información, ver Fig. 1.

CAN bus es un protocolo de comunicación en el cual una trama de datos digitales es enviada a cada ECU, los cuales analizan si la información es de utilidad o no [5], [6]; siendo capaces de detectar la falta de alguna señal del vehículo, reenviando de esta manera el módulo principal la información donde no se detectó la señal respectiva para ser evaluada nuevamente [7]. Los métodos de diagnóstico para CAN bus se basan en el uso de osciloscopios, escáneres automotrices,

cajas de pruebas, softwares especializados e interfaces computacionales con conexión al *Diagnostic Link Connector* (DLC) como CANoe o Kvaser [8] para monitorear la transmisión de datos en tiempo real. Sin embargo, estos métodos de análisis no permiten saber específicamente donde se genera una falla [1], lo cual se podría lograr con el análisis del bus de datos [9] del vehículo.[10]

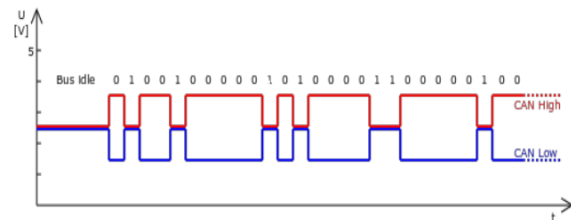


Fig. 1. Protocolo CAN.

Debido a que el protocolo de comunicación CAN es ampliamente utilizado en la industria a nivel mundial [11], se han desarrollado varios estudios respecto al mismo, entre los cuales resaltan aquellos relacionados al desarrollo de una investigación, 2009, en la cual se establecieron varios pasos para decodificar las señales presentes en el CAN bus [12], para llegar al reconocimiento de la información que se transmitía hacia los diferentes módulos del vehículo, dando como resultado una metodología que permite determinar tal información de forma adecuada. En 2011, se realizó el proceso de ingeniería inversa en sistemas mecatrónicos [2], dicho proyecto se centró en analizar y reproducir las señales que ingresaban al módulo original del vehículo para la programación de una ECU usando tales parámetros. En 2019, se llevó a cabo el diseño un panel de instrumentos virtual [13], el cual ocupaba las señales en tiempo real de la red de comunicación del panel físico, validando los resultados mediante el funcionamiento en simultáneo con el panel virtual. Los trabajos mencionados se centran en la lectura, identificación e interpretación de las señales en las tramas de datos de la red de comunicación del vehículo para distintos propósitos. No obstante, en ninguno de los artículos mencionados se ha tenido en consideración el estudio de los bytes de los *data frames* y su aplicación para diagnóstico en tiempo real.

El presente documento da a conocer como a través del análisis de las tramas de datos de las señales que ingresan al módulo electrónico principal, y mediante la distinción en la variación de su información se pueden detectar errores en la comunicación del vehículo. El documento se encuentra

organizado de la siguiente manera: en la sección II se abordan aspectos generales de los módulos de control electrónico analizados, pertenecientes al vehículo de prueba, así como el análisis del estado de las señales CAN y la adquisición de la información de las ECU's. En la sección III se detallan los tipos de sensores y señales que ingresan a la PCM, sus líneas de conexión, emulación y análisis de los *data frames* respectivos. En la sección IV se presenta el desarrollo de la base de datos y simulación de la red de comunicación del vehículo en Matlab, utilizando la plataforma VNT, teniendo en cuenta los parámetros de las señales y su acondicionamiento mediante un *cluster* generado en LabView para decodificar la información de las tramas de datos, obteniendo como respuesta el accionamiento de lámparas testigo para cada señal en el instante que se produce la falla en la comunicación del vehículo. Por último, en la sección V se sintetizan las conclusiones, validando la metodología llevada a cabo que permite detectar las situaciones de falla en el CAN bus para los sensores estudiados.

## II. UNIDADES DE CONTROL ELECTRÓNICO VOLKSWAGEN

### A. Módulos PCM, IPC y Airbag

Power-Train Control Module J220 e Instrument Panel Cluster J285 transmiten la información a una velocidad de 500 kbit/s por medio de un par de cables trenzados. Si bien estos dos módulos comparten información a través del bus de datos, la IPC es alimentada adicionalmente con información de otros sensores [10].

Los módulos electrónicos pertenecientes al vehículo Volkswagen Jetta 2009 cuentan con líneas de conexión numeradas y específicas para el suministro de energía, señal, etc., ver Tabla 1, que fueron ubicados mediante la lectura de sus diagramas eléctricos o a partir del análisis de sus circuitos electrónicos, esencial para llevar a cabo la adquisición de datos cuando la información sea transmitida a través de la red de comunicación.

TABLA 1. ECU'S VOLKSWAGEN .

LÍNEAS DE CONEXIÓN	PCM ME 17.5.6	AIRBAG 5WK44285	IPC 5C6920851B
ALIMENTACIÓN (+)	3,5	26	32
IGNICIÓN (+)	87,92	-	-
PANTALLA (+)	-	-	31
MASA (-)	1	28	16
CAN HIGH	68	49	28
CAN LOW	67	50	29

### B. Análisis de la señal CAN

Después de haber realizado las conexiones entre los módulos y la fuente de poder, se enlazó un osciloscopio al bus de datos para monitorear el estado de las señales CAN Low y CAN High, de esta manera se pudo descartar posibles interferencias y visualizar los patrones de onda correctos relacionados al protocolo de comunicación [14], ver Fig. 2.



Fig. 2. Señales CAN en módulos Volkswagen.

Una vez validada la transmisión de señales en la red de comunicación, se vinculó el conjunto de módulos con un escáner automotriz, verificando así que se podía acceder a la información de cada una de las ECU's presentes en el sistema como se observa en la Fig. 3.

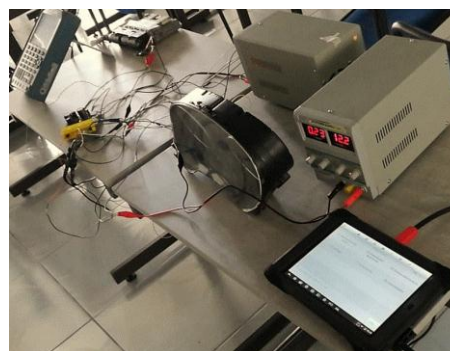


Fig. 3. Acceso al módulo PCM, ICP y Airbag.

### C. Adquisición y lectura de datos CAN

Para determinar los identificadores de las señales y los bytes que cambiaban en los *data frames* de la red de comunicación, se enlazó la tarjeta de adquisición NI para recolectar datos del conjunto de módulos a través de un probador-conector OBD2 - CAN Bus para analizar toda la información que viajaba a través del bus de información.

En la Fig. 4 se puede observar que el sistema cuenta con 20 identificadores, de los cuales algunos de ellos presentaban tramas cuyos valores permanecían estáticos, mientras que otros variaban sin la necesidad de aplicar ninguna señal de entrada a la PCM.

ID	Time Stamp	Length	Data	Type	Dir	Rate [Hz]	No of Frame
0x280	161.877	8	93 00 00 00 00 FF 07 00	S	Rx	98.6	15866
0x288	161.868	8	49 FF 03 00 00 78 5A 4C	S	Rx	50.3	7933
0x320	161.868	8	12 01 FF FF FF C9 FF 9E	S	Rx	50.0	7923
0x380	161.877	8	00 5B FF 00 00 30 FF	S	Rx	98.6	15866
0x420	161.769	8	83 FF FF 00 08 7F FF 84	S	Rx	5.0	792
0x480	161.869	8	53 2C 00 00 E1 01 10 8F	S	Rx	50.3	7933
0x488	161.877	8	37 00 00 5E A6 FF 00 30	S	Rx	98.6	15865
0x48A	161.869	8	32 00 00 02 00 00 00 30	S	Rx	50.3	7933
0x520	161.869	8	55 C0 00 00 80 68 41 01	S	Rx	5.0	793
0x52A	161.769	8	00 00 00 00 00 00 00 00	S	Rx	1.0	159
0x580	161.379	8	87 00 00 0F 70 0C 43	S	Rx	1.0	159
0x588	161.869	8	18 00 5E 00 25 10 00 96	S	Rx	49.7	7933
0x58C	161.797	8	80 00 00 00 FF 00 00 00	S	Rx	10.0	1589
0x58F	161.879	8	00 F0 FF 7F 00 00 1A	S	Rx	2.0	318
0x5D2	161.770	8	01 47 57 36 41 4A 33 44	S	Rx	5.0	792
0x5F3	161.770	8	08 00 FF 0F 00 00 00 00	S	Rx	2.0	317
0x60E	161.770	2	08 00	S	Rx	1.0	159
0x621	161.870	8	00 00 80 FF 90 7F 91 00	S	Rx	10.1	1585
0x62E	161.771	7	68 41 21 75 15 9A 19	S	Rx	1.0	159
0x727	161.739	7	04 01 01 00 02 00 00	S	Rx	5.0	792

Fig. 4. Identificadores y tramas de datos CAN de los módulos Volkswagen.

## III. IDENTIFICACIÓN Y EMULACIÓN DE SEÑALES

Debido a la cantidad de identificadores que presentaba el sistema, fue necesario filtrar aquellos que pertenecían a la PCM para analizarlos cuando el módulo estuviese excitado con señales de entrada. Por consiguiente, para identificar las señales que correspondían al módulo principal se optó por desconectar en orden el módulo AIRBAG e IPC, los cuales suspendieron la emisión de ciertos datos a la red de comunicación cuando ya no se encontraban enlazados al sistema.



En la Fig. 5 se visualiza que las tramas de datos en color pertenecen a la ECU principal, con lo cual se generó una base de datos que se utilizó de referencia para detectar la variación de los bytes al momento que se excitaba el módulo con señales de los sensores respectivos.

ID	Time Stamp	Length	Data	Type	Dir	Rate [Hz]	No of Frame
0x280	62.744	8	93 00 00 00 00 FF 0C 00	S	Rx	101.8	6276
0x288	62.736	8	11 FF 03 00 00 78 7D 68	S	Rx	49.9	3138
0x320	21.183	8	00 00 00 01 00 09 00 80	S	Rx	49.9	737
0x380	62.745	8	00 5B FF 00 00 00 30 FF	S	Rx	101.8	6276
0x420	21.024	8	83 00 FF 00 08 32 FF 00	S	Rx	1.3	81
0x480	62.737	8	1A 2C 00 00 EL 00 10 C7	S	Rx	49.9	3138
0x488	62.745	8	28 00 00 5E A6 FF FF 00	S	Rx	101.8	6276
0x48A	62.737	8	22 00 00 02 00 00 00 20	S	Rx	49.8	3138
0x520	21.123	8	A6 87 00 00 80 68 41 01	S	Rx	1.0	73
0x52A	21.025	8	00 00 00 00 00 00 00 00	S	Rx	0.6	17
0x580	61.928	8	81 00 00 00 00 10 28 S	Rx	1.0	62	
0x588	62.737	8	18 00 5E 00 25 10 00 96	S	Rx	49.8	3138
0x58C	62.715	8	80 00 00 00 FF 00 00 00	S	Rx	10.0	628
0x58F	62.427	8	00 F0 FF 7F 00 00 00 08	S	Rx	2.0	125
0x5D2	21.025	8	00 00 00 00 00 33 56 57	S	Rx	1.3	81
0x5F3	21.025	8	00 00 00 00 00 00 00 00	S	Rx	0.9	33
0x60E	21.025	2	08 00	S	Rx	0.6	17
0x621	21.124	8	00 00 80 00 90 32 02 00	S	Rx	10.2	154
0x62E	21.026	7	00 00 20 75 05 00 12	S	Rx	0.6	17
0x727	21.013	7	01 01 04 00 06 00 00	S	Rx	0.6	41

Fig. 5. Identificadores de los módulos PCM e IPC.

### A. Señales del módulo PCM

Para emular las señales que alimentaban al módulo, ver Fig. 6, se identificaron inicialmente las líneas de conexión a través de las cuales la información ingresaba utilizando el diagrama eléctrico del vehículo.

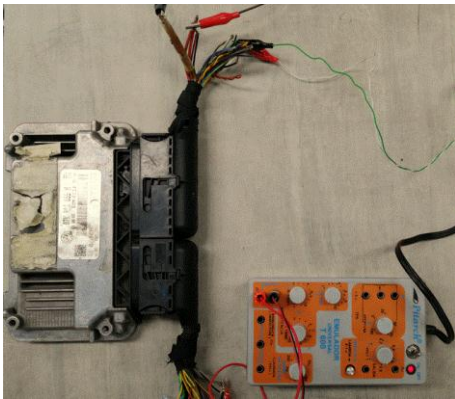


Fig. 6. Emulación de las señales de revoluciones del motor y temperatura del refrigerante.

La Tabla. 2 presenta en resumen las líneas de conexión, entradas y salidas, de las señales de los sensores que fueron parte del estudio y excitaban a la ECU principal, las cuales fueron emuladas utilizando un generador de frecuencias universal en los procesos que se detallaran más adelante.

Tabla 2. ENTRADA DE SEÑALES AL MÓDULO PCM.

ECU	PCM		
	ENGINE SPEED	COOLANT TEMPERATURE	AIR PRESSURE
AMARILLO/CELESTE	51	-	-
AMARILLO/NEGRO	59	-	-
AMARILLO (ENTRELAZADO)	-	14	-
CELESTE (ENTRELAZADO)	-	57	-
NEGRO	-	-	13
GRIS (CABLE DELGADO)	-	-	42
MARRÓN/BLANCO	-	-	56

### B. Emulación de señales

Las señales que se emularon pertenecen al sensor de revoluciones del motor y de temperatura del refrigerante del motor, ver Fig. 7, que de acuerdo al diagrama del vehículo correspondían a una resistencia variable y un sensor de efecto hall.

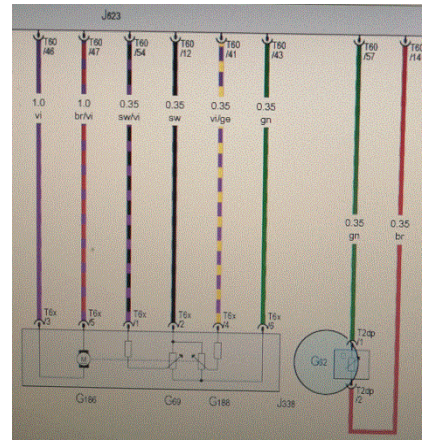


Fig. 7. Diagrama eléctrico del sensor de temperatura del refrigerante.

Mediante el uso de un emulador universal se introdujeron señales a la PCM, cuyo comportamiento se analizó directamente utilizando un escáner enlazado al sistema y empleando la función datos en vivo para visualizar las curvas de revoluciones y temperatura del motor que se generaban, ver Fig. 8, asegurando así que las líneas de conexión del módulo fuesen las correctas.

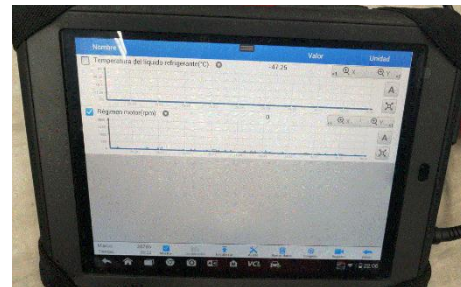


Fig. 8. Gráfica de las señales de revoluciones del motor y temperatura del refrigerante.

### C. Analisis de la trama de datos

En la Tabla. 3 se sintetiza la información perteneciente a los identificadores de las dos señales que estimularon al módulo PCM, observándose para cada uno de los datos diferentes posiciones y valores dependiendo de su estado y presentados en sistema hexadecimal.

Tabla 3. ESTADO DE SEÑALES DEL MÓDULO PCM.

PCM		SENSOR/ESTADO	TRAMA DE DATOS							
ID	SEÑAL		1	2	3	4	5	6	7	8
0x280	ENGINE SPEED				x	x				
		SIN SEÑAL/REPOSO			00	00				
		CONTACTO/MIN			1A	00				
0	COOLANT							x	x	



	SIN SEÑAL/ REPOSO						7D	67
								7D
	CONTACTO/ MIN						68	57

Las señales con las cuales se estimuló a la PCM fueron registradas mediante la tarjeta de adquisición de datos, que mediante el análisis de la información de las ECU's en tiempo real permitió consolidar en primer lugar los identificadores de cada una de las señales que alimentaban al módulo principal a más de cuáles eran los bytes y la posición en la que los datos variaban y se actualizaban.

#### IV. DESARROLLO DE SIMULACION EMPLEANDO EL SOFTWARE MATLAB

Después de haber determinado los valores iniciales que el módulo PCM enviaba a la red de comunicación cuando los sensores no alimentaban a la ECU principal, se procedió a configurar la interfaz de simulación VNT, plataforma de Simulink dedicado a redes de comunicación CAN, ver Fig. 9, utilizando la base de datos generada en los procesos anteriores.

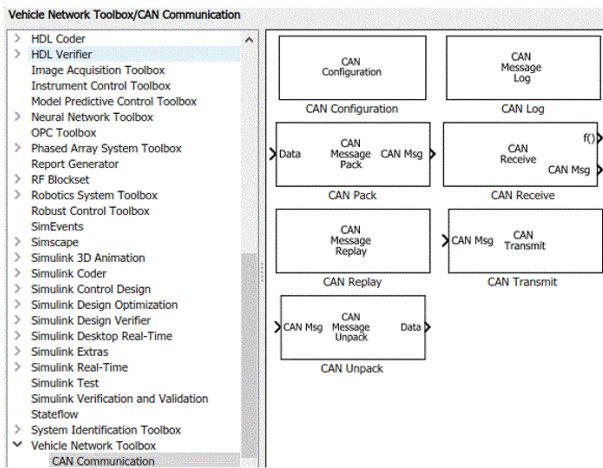


Fig. 9. Comunicación CAN Matlab - Simulink.

#### A. Desarrollo de base de datos .mat

La interfaz VNT cuenta con varios bloques CAN, los cuales junto con la tarjeta de adquisición NI se utilizaron para generar las bases de datos. En cada una de ellas se especificó la variable guardada para la simulación correspondiente.

En la Fig. 10 se observan las configuraciones usadas para la adquisición de datos, simulación y lectura de información en tiempo real del vehículo; además de los archivos con la información de las variables almacenadas con extensión .mat, que es propia del software Matlab.

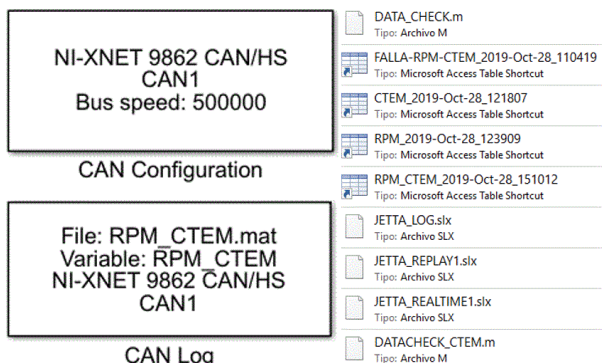


Fig. 10. Base de datos .mat

#### B. Desarrollo de base de datos .dbc

A pesar de contar con las bases de datos de los sensores emulados, para poder decodificar la información recolectada durante la simulación y en tiempo real, fue necesario crear un archivo .dbc con características específicas de la trama de datos del bus de comunicación, para lo cual se utilizó el software LabView, creando así las señales que serían interpretadas en la simulación. Sin embargo, aunque era necesario proveer el archivo con información de factor de escala y offset, ver Fig. 11, este inconveniente se resolvió utilizando algoritmos, ver Ecuación (1) y Ecuación (2), para las señales de revoluciones del motor y la temperatura del refrigerante.

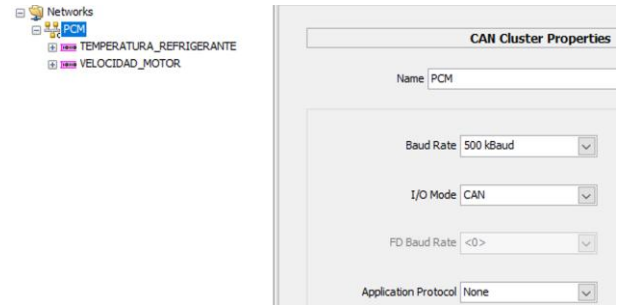


Fig. 11. Archivo .dbc

#### C. Ajuste de las señales de salida

Las señales que se obtuvieron a la salida del bloque *Sub-System*, ver Fig. 12, al no haber sido manipuladas, se hallaban fuera de rango. Debido a esto se recurrió a un algoritmo [15] para cada una de las señales, las cuales permitieron en el caso de las revoluciones del motor y la temperatura del refrigerante, acércalas a los valores reales.

$$RPM = \frac{(256xA) + B}{4} \quad (1)$$

$$CTEM = A - 40 \quad (2)$$

Donde A y B representan los valores de los bytes que cambian dentro del conjunto de datos de cada una de las señales respectivamente.

Al finalizar los procesos en los bloques de fórmula se graficaron las magnitudes de las dos señales para visualizar su comportamiento durante la simulación y en tiempo real.

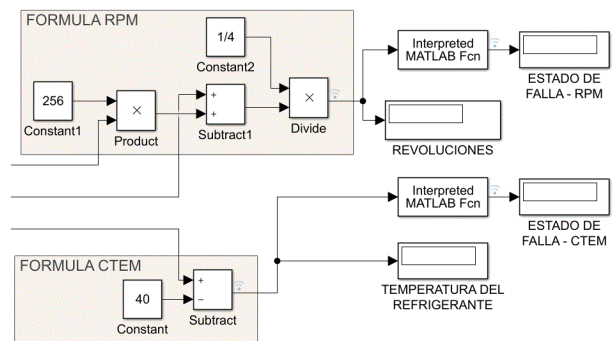


Fig. 12. Modelo para simulación de protocolo CAN.

#### D. Reconocimiento de datos

En la Fig. 12 del modelo de simulación se aprecian bloques Matlab - FCN que requerían de *scripts* para analizar las señales de los sensores y activar los testigos en el momento que se produjera la falla en los dispositivos.

Al programar los *scripts*, ver Fig. 13, se tomaron en cuenta los valores obtenidos al momento en el cual el módulo PCM no registraba ninguna señal de entrada que lo estuviese estimulando, asumiendo dicha información como los estados de falla de los sensores estudiados.

```
function y = temperatura(u)
for i=1:1000
    j(i)=u;
end

k=length(j);
l=find(j==64);
m=find(j==63);
n=length(l);
o=length(m);

if n == k || o==k
    y = 0;
else y = 1;
end
end
```

Fig. 13. Identificador de fallas.

#### E. Desarrollo del modelo de simulación

La simulación se realizó en dos partes, debido a que una de ellas utilizaba la base de datos de las señales emuladas y la otra la información que era enviada en tiempo real desde la red de comunicación del vehículo mediante el uso de la interfaz NI que interpreta el protocolo CAN.

El objetivo de la simulación se centró en la visualización de un testigo, ver Fig. 14, en el instante en el cual la PCM enviaba las señales iniciales de reposo de los sensores, dando a entender que se había producido una falla en el vehículo. Además, se añadió un gráfico en tiempo real para observar la evolución de los parámetros de acuerdo con las señales registradas.

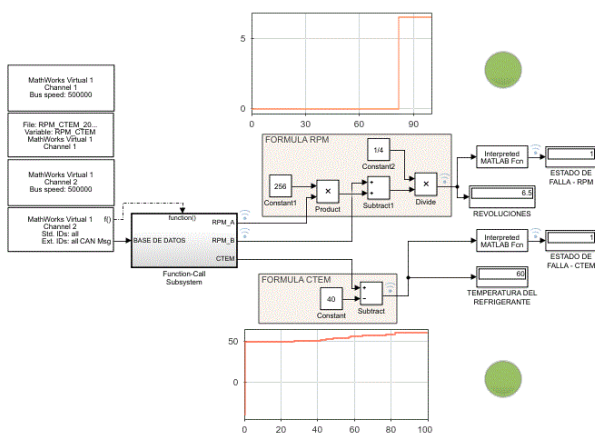


Fig. 14. Simulación de revoluciones del motor y temperatura del refrigerante en Simulink utilizando bloques VNT.

#### V. CONCLUSIONES

De acuerdo a la información obtenida durante los procesos del proyecto se concluyó que:

Las fallas de la red de comunicación se presentaron con un valor estático para las señales de revoluciones del motor, mientras que para la señal de temperatura del refrigerante los valores cambiaban en un rango específico, esto debido al estado de funcionamiento del módulo y el dispositivo emulador de señales. Se recomienda trabajar en un ambiente más controlado para determinar valores constantes para la base de datos correspondiente.

Como resultado del análisis de las tramas de datos pertenecientes a los identificadores de las señales de revoluciones del motor y temperatura del refrigerante, se pudo determinar cuando la red de comunicación sufría un desperfecto en tiempo real sin la necesidad de utilizar dispositivos adicionales. El proceso de análisis para los identificadores se puede replicar para la información no tratada en el bus de datos, tomando en consideración las líneas de conexión de las señales.

Durante el proceso de análisis y experimentación fue posible visualizar identificadores esporádicos, por lo que se dedujo que para conexiones erróneas en la PCM existían tramas de datos específicos que no se visualizaban en condiciones normales de funcionamiento.

El modelo de simulación desarrollado para el vehículo Volkswagen Jetta 2009 puede ser aplicable para otros automotores, permitiendo determinar fallas en el bus de datos del vehículo siempre y cuando cuenten con módulos idénticos a los estudiados.

#### REFERENCIAS

- [1] P. Christopher, "Manual De Procedimientos Para Interactuar Entre Protocolos De Comunicación Automotriz," Universidad Del Azuay, 2013.
- [2] S. Freiberger, M. Albrecht, And J. Käufel, "Reverse Engineering Technologies For Remanufacturing Of Automotive Systems Communicating Via Can Bus.," J. Remanufacturing, Vol. 1, No. 1, P. 6, 2011.
- [3] S. M. Caro, "Interfaz Gráfica De Usuario Para El Monitoreo De Tramas Can Por Medio De Un Soc Cyclone V.," Pist. Educ., Vol. 39, No. 04, Pp. 354–372, 2017.
- [4] Volkswagen Ag, "Data Exchange On The Can Bus I." Pp. 1–32, 2001.
- [5] P. Teodoro, "Estudio Para Simular Una Red Can Con Aplicación En Comunicación De Dispositivos Electrónicos En El Automóvil.," Universidad Del Azuay, 2013.
- [6] "Thirst For Knowledge – Sae J1979 Protocol." [Online]. Available: <https://happilyembedded.wordpress.com/2015/07/23/thirst-for-knowledge-sae-j1979-protocol/>.
- [7] F. Y. Hiodo, J. L. Porsani, And V. R. Elis, "Introducción A Can Bus: Descripción, Ejemplos Y Aplicaciones De Tiempo Real.," Rev. Bras. Geofis, Vol. 20, No. 2, Pp. 129–134, 2002.
- [8] "555b82d10f23d78d9d2d82a33591fcb56fae0c7e @Www.Kvaser.Com." [Online]. Available: <https://www.kvaser.com/about-can/the-can-protocol/>.
- [9] L. I. J. G. E. Hernandez, "Software Analizador De Can Para Avt-852," Centro De Tecnología Avanzada, 2017.
- [10] I. Volkswagen Of America, "The New Jetta Electrical System Design And Function," No. 873403. Pp. 1–58, 2004.
- [11] S. Corrigan, "Introduction To The Controller Area Network ( Can )," 2016.
- [12] T. Hermans, J. Denil, P. De Meulenaere, And J. Anthonis, "Decoding Of Data On A Can Powertrain Network," No. 1, 2009.
- [13] M. Kubis, M. Gutten, D. Korenciak, And M. Danko, "Communication Of Cluster Using Can Bus," Orig. Res. Artic., Vol. 21, Pp. 69–74, 2019.
- [14] M. Esro, A. A. Basari, S. K. S. A. M. I. Sadiqin, And Z. Syariff, "Controller Area Network ( Can ) Application In Security System," Pp. 299–302, 2009.
- [15] J. Echeverry, "Tecnologías De Control Electrónico."