





**INSTITUTO SUPERIOR TECNOLÓGICO  
VIDA NUEVA**

**TECNOLOGÍA SUPERIOR EN MECÁNICA AUTOMOTRIZ**

**DISEÑO Y CONSTRUCCIÓN DEL BANCO DE PRUEBAS PARA  
COMPROBAR Y VERIFICAR LA COMPUTADORA  
AUTOMOTRIZ CON INYECCIÓN ELECTRÓNICA A  
GASOLINA DE UN VEHÍCULO AVEO FAMILY.**

**PRESENTADO POR:**

**ACURIO BARRIGA WILLIAM ANDRÉS**

**TUTOR:**

**ING. UGEÑO GUILCAPI DENIS MARCELO**

**ABRIL 2022**

**QUITO – ECUADOR**

## TECNOLOGÍA SUPERIOR EN MECÁNICA AUTOMOTRIZ

---

### CERTIFICACIÓN DEL TUTOR

---

En mi calidad de Tutor del Proyecto: “DISEÑO Y CONSTRUCCIÓN DEL BANCO DE PRUEBAS PARA COMPROBAR Y VERIFICAR LA COMPUTADORA AUTOMOTRIZ CON INYECCIÓN ELECTRÓNICA A GASOLINA DE UN VEHÍCULO AVEO FAMILY.” en la ciudad de Quito, presentado por el/la ciudadano/a ACURIO BARRIGA WILLIAM ANDRÉS, para optar por el título de Tecnólogo Superior en MECÁNICA AUTOMOTRIZ, certifico que dicho proyecto ha sido revisado en todas sus partes y considero que reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del tribunal examinador que se designe.

En la ciudad de Quito, mes de abril de 2022

---

**TUTOR: ING. UGEÑO GUILCAPI DENIS MARCELO**

**C.I.: 1721866778**

## TECNOLOGÍA SUPERIOR EN MECÁNICA AUTOMOTRIZ

---

### APROBACIÓN DEL TRIBUNAL

---

Los miembros del tribunal aprueban el informe de investigación, sobre el tema: “DISEÑO Y CONSTRUCCIÓN DEL BANCO DE PRUEBAS PARA COMPROBAR Y VERIFICAR LA COMPUTADORA AUTOMOTRIZ CON INYECCIÓN ELECTRÓNICA A GASOLINA DE UN VEHÍCULO AVEO FAMILY” en la ciudad de Quito, del/la estudiante: ACURIO BARRIGA WILLIAM ANDRÉS, de la Carrera en Tecnología Superior en MECÁNICA AUTOMOTRIZ.

Para constancia firman:

\_\_\_\_\_  
**ING.**

**DOCENTE ISTVN**

\_\_\_\_\_  
**ING.**

**DOCENTE ISTVN**

\_\_\_\_\_  
**ING.**

**DOCENTE ISTVN**

---

## **CESIÓN DE DERECHOS DE AUTOR**

---

Yo, **ACURIO BARRIGA WILLIAM ANDRÉS** portador/a de la cédula de ciudadanía **0504277070**, facultado/a de la carrera **TECNOLOGÍA SUPERIOR EN MECÁNICA AUTOMOTRIZ**, autor/a de esta obra certifico y proveo al Instituto Superior Tecnológico Vida Nueva, usar plenamente el contenido del informe con el tema **“DISEÑO Y CONSTRUCCIÓN DEL BANCO DE PRUEBAS PARA COMPROBAR Y VERIFICAR LA COMPUTADORA AUTOMOTRIZ CON INYECCIÓN ELECTRÓNICA A GASOLINA DE UN VEHÍCULO AVEO FAMILY”**, con el objeto de aportar y promover la lectura e investigación, autorizando la publicación de mi proyecto de titulación en la colección digital del repositorio institucional bajo la licencia de Creative Commons: Atribución-NoComercial-SinDerivadas.

En la ciudad de Quito, del mes de abril de 2022

---

**ACURIO BARRIGA WILLIAM ANDRÉS**

**C.I. 0504277070**

## **DEDICATORIA**

Gracias a todas las personas que he tenido  
la dicha de conocer, compartir y que me han  
brindado su apoyo abnegado, con sus consejos,  
ánimos y acciones que han contribuido en mi  
formación profesional y que se ve reflejado  
en este proyecto de aplicación práctica.

## **AGRADECIMIENTO**

El más sincero agradecimiento a mis padres y hermanos que siempre me apoyaron para llegar a cumplir con este sueño que es de ser un hombre de bien y con una profesión, y los compañeros q de una u otra manera siempre estaban en los momentos difíciles de mi vida estudiantil. Además, agradezco a los ingenieros que me brindaron sus conocimientos y sus consejos para ser un profesional de éxito, también en ámbito personal.

## ÍNDICE DE CONTENIDOS

RESUMEN.....	1
ABSTRACT .....	2
INTRODUCCIÓN .....	3
ANTECEDENTES .....	5
JUSTIFICACIÓN .....	6
OBJETIVOS.....	7
Objetivo General.....	7
Objetivos Específicos.....	7
MARCO TEÓRICO .....	8
TIPOS DE COMPUTADORAS AUTOMOTRICES ECU-ECM.....	9
ECM Electronic Control Module.....	9
Unidad de control electrónico ECU.....	10
PCM: Powertrain Control Module .....	11
BCM: Body Control Module.....	12
COMPONENTES.....	12
METODOLOGÍA DEL PROYECTO - DESARROLLO .....	14
ESCRIPCIÓN GENERAL DEL SISTEMA.....	14
DISEÑO DEL SISTEMA .....	14
Requisitos en base a especificaciones de potencia.....	15
Requisitos en base al desempeño solicitado.....	15
Diseño del hardware .....	15
Hardware de control y visualización .....	16
Microcontrolador principal .....	16



Detector de pulsos.....	18
Subsistema de respuesta .....	19
Interfaces de potencia electromecánicas .....	19
Inyectores .....	20
Interfaces de potencia de estado sólido.....	21
Subsistema de alimentación.....	22
Regulador de voltaje principal .....	23
Regulador de voltaje secundario .....	24
CONSTRUCCIÓN FÍSICA DEL SISTEMA.....	25
Construcción de la estructura.....	25
SIMULACIÓN EN PROTEUS.....	25
Simulación de funcionamiento.....	27
Montaje y conexión del dispositivo .....	29
Montaje del sistema electrónico .....	29
RESULTADOS - RESPUESTA.....	30
SIMULACIÓN Y FASE DE PRUEBA.....	30
COMPROBACIÓN DE FUNCIONAMIENTO DEL BANCO DE PRUEBAS DE COMPUTADORES .....	32
PRUEBAS DE FUNCIONAMIENTO .....	37
PRUEBA DE PULVERIZACIÓN.....	37
Procedimiento para emulación y prueba de inyectores con entrada superior .....	37
PRUEBA DE ESTANQUEIDAD .....	40
CHEVROLET AVEO FAMILY 2008 .....	42

CONCLUSIONES.....	48
RECOMENDACIONES .....	49
REFERENCIAS BIBLIOGRÁFICAS .....	50
ANEXOS .....	51

## ÍNDICE DE FIGURAS

Figura No. 1 Ejemplo de computador automotriz.....	8
Figura No. 2 Ejemplo de computador automotriz, sensores y actuadores .....	9
Figura No. 3 Ejemplo de ECM (Electronic Control Module), sensores y actuadores.....	10
Figura No. 4 Ejemplo de (Unidad de control electrónico ECU), sensores y actuadores.....	11
Figura No. 5 Ejemplo de (PCM: Powertrain Control Module).....	11
<i>Figura No. 6 Ejemplo de (BCM: Body Control Module).....</i>	<i>12</i>
Figura No. 7 Ejemplo de componentes computadora automotriz.....	13
Figura No. 8 Microcontrolador principal .....	17
Figura No. 9 Detector de pulsos.....	18
Figura No. 10 Pulsos eléctricos funcionamiento de inyector.....	18
Figura No. 11 Interfaces de potencia electromecánicas.....	20
Figura No. 12 Inyector módulo.....	20
Figura No. 13 Acople eléctrico salidas inyectores.....	22
Figura No. 14 Fuente de alimentación .....	23
Figura No. 15 Regulador de voltaje principal.....	24

Figura No. 16 Regulador de voltaje secundario.....	24
Figura No. 17 Estructura de plástico.....	25
Figura No. 18 Simulación sincronización módulo.....	26
Figura No. 19 Simulación sincronización módulo.....	27
Figura No. 20 Prueba en protoboard placas de potencia.....	28
Figura No. 21 <i>Ordenamiento del cableado articulado.</i> .....	28
Figura No. 22 <i>Instalación del cableado de la pantalla.</i> .....	29
Figura No. 23 <i>Pantalla LCD instalada.</i> .....	30
Figura No. 24 Testigo módulo funcional. ....	30
Figura No. 25 Testigo módulo apagado led en estado intermitente.....	31
Figura No. 26 Prueba de conexión apagada satisfactoria. ....	31
Figura No. 27 Testigo módulo encendido led en estado intermitente.....	32
Figura No. 28 Prueba de conexión encendido satisfactoria. ....	32
Figura No. 29 Prueba de pulverización.....	38
Figura No. 30 Botón de ignición e inicio de prueba. ....	38
Figura No. 31 Pantalla estado prueba. ....	39
Figura No. 32 Prueba de estanqueidad.....	40

Figura No. 33 Botón de ignición e inicio de prueba. ....	41
Figura No. 34 Pantalla estado prueba. ....	42

## ÍNDICE DE TABLAS

Tabla 1 <i>Conexión del microcontrolador.</i> .....	16
Tabla 2 Características del transistor IRF. ....	21
Tabla 3 Características de la fuente de alimentación .....	22
Tabla 4 <i>Parámetros de consideración de las pruebas.</i> .....	33
Tabla 5 Resultados pruebas de funcionamiento inyectores Chevrolet Aveo Family 2008.....	42
Tabla 6 Resultados pruebas de funcionamiento inyectores Chevrolet Aveo Family 2008.....	43
Tabla 7 Resultados pruebas de funcionamiento inyectores Chevrolet Aveo Family 2008.....	44
Tabla 8 Resultados pruebas de funcionamiento inyectores Chevrolet Aveo Family 2008.....	46

## **RESUMEN**

En este proyecto de investigación se realiza un banco de pruebas mediante un simulador de funcionamiento para la unidad de control electrónico ecu, el estudiante podrá mejorar el aprendizaje midiendo las señales características en cada uno de los sensores. Este banco de pruebas demostrara funcionamiento de los sensores y actuadores relacionadas a la ECU. Con el banco de pruebas se podrá verificar el funcionamiento de la unidad de control electrónico del motor a diferentes rangos de trabajo sin tener la necesidad de estar conectada al vehículo. En este contexto, el banco puede ser utilizado como una herramienta de diagnóstico para unidades de control electrónico, volviéndose un recurso útil en situaciones donde no se puede movilizar el vehículo y que se tiene la necesidad de analizar el funcionamiento de estos elementos. En este proyecto se propone un diseño e implementación de un banco de pruebas que funcione para simular el funcionamiento de una unidad de control electrónico del vehículo. El banco de pruebas tendrá la finalidad de ser usado, como una herramienta de apoyo, para el aprendizaje del funcionamiento del sistema sobre la gestión electrónica del encendido, inyección e incluso del sistema de inmovilización del vehículo.

### **PALABRAS CLAVE:**

ECU, BANCO DE PRUEBAS, COMPUTADORAS AUTOMOTRICES, PCM, MÓDULOS EDUCATIVOS

## **ABSTRACT**

In this research project a test bench is made by means of an operation simulator for the ecu electronic control unit, the student will be able to improve learning by measuring the characteristic signals in each of the sensors. This test bench will demonstrate the operation of the sensors and actuators related to the ECU. With the test bench it will be possible to verify the operation of the electronic engine control unit at different working ranges without having to be connected to the vehicle. In this context, the bench can be used as a diagnostic tool for electronic control units, becoming a useful resource in situations where the vehicle cannot be mobilized and there is a need to analyze the operation of these elements. This project proposes a design and implementation of a test bench that functions to simulate the operation of a vehicle electronic control unit. The test bench is intended to be used as a support tool for learning how the system works on the electronic management of ignition, injection and even the vehicle's immobilization system.

### **KEYWORDS:**

ECU, TEST BENCH, AUTOMOTIVE COMPUTERS, PCM, EDUCATIONAL MODULES.



## **INTRODUCCIÓN**

### **ELEMENTOS ELÈCTRICOS Y ELECTRÓNICOS UTILIZADOS EN COMPUTADORAS AUTOMOTRICES**

En las últimas décadas, con el avance en las tecnologías electrónicas incorporadas por los fabricantes para los diferentes sistemas del automóvil han surgido a la par múltiples formas para diagnosticar los problemas asociados a estos nuevos componentes. (Corti, 2014). Todas estas funciones se realizan a través de la computadora del automóvil, la cual recibe los parámetros de funcionamiento y cargas a las que está sometido el motor mediante sensores. La Unidad de Control Electrónico (ECU) es relativamente compleja debido a sus múltiples funciones como la de controlar la combustión a través de correcta sincronización al momento de generar pulsos de inyección de combustible y de ignición de chispa en el caso de motores MEP (Rueda Santander, 2005). Así también, se puede contar con herramientas didácticas, de fácil utilización para la detección oportuna de fallas. Esto ha introducido una gran carga y reto en los investigadores y mecánicos automotrices que ahora tienen el objetivo de resolver problemas, no solo del campo mecánico sino también en el electrónico del vehículo, sin necesariamente estar preparados para tareas que demandan un conocimiento combinado de teoría mecánica y electrónica.

En cuanto a los problemas que pueden existir, esto puede ser causa de todo tipo de problemas en el vehículo y en algunos casos provocar que no se pueda manejar. Normalmente una falla en la ECU producirá ciertos síntomas que pueden alertar al conductor de que algo anda mal entre estos tenemos algunos problemas de la ecu como tales como: la luz de Check Engine se enciende usualmente, cuando la computadora detecta un problema con los sensores o los circuitos, el motor se detiene este es otro indicio que algo no está bien con la computadora, es el comportamiento errático del motor, el auto no enciende se da cuando la ECU no funciona, el motor se quedará sin unidad de control y no prenderá. (Gajdarus, 2017)

En este contexto el nuevo reto que aparece frente a la necesidad de crear herramientas para realizar diagnósticos y reparaciones en los talleres mecánicos de una manera eficiente. Dentro de las diferentes necesidades identificadas por los investigadores en técnicas automotrices, específicamente en los parámetros de competitividad, en su entorno, se identifica ciertos problemas como los diseños e implementaciones de sistemas con microprocesador como unidad central de procesamiento además de circuitos electrónicos, así también estos puedan emular las señales que se generan de los sensores implicados en el sistema de inyección electrónico de un vehículo, el cual tiene una computadora automotriz (ECU, Electronic control unit, en español: Unidad de control electrónico). (Gajdarus, 2017) Uno de los problemas subyacentes, está en la búsqueda de sistemas interactivos a desarrollarse, desde la visión de la mecánica automotriz, como por ejemplo la ayuda para identificar, con cierto grado de certeza, dentro de un vehículo el error generado y registrado en la ECU a partir del código de programación junto con la detección temprana de fallas dentro de los sistemas que gobierna la placa.

## **ANTECEDENTES**

El presente trabajo ha sido realizado con la finalidad de ayudar al estudiante y profesionales de la carrera de tecnología superior en mecánica automotriz, tanto en la parte teórica y práctica interesados en el área de computadores automotrices.

Tomando en cuenta satisfacer las necesidades de una comprobación y verificación de computadores automotrices a inyección electrónica de combustible en un tiempo mínimo y con resultados precisos, se ha creado este banco de pruebas.

El mismo que dispondrá con todos los accesorios y elementos necesarios para simular los sensores y actuadores, encargados a la vez de poner en funcionamiento a la ECU.

Una de las cualidades más importantes que debe poseer un técnico superior en mecánica automotriz es su creatividad considerando los adelantos tecnológicos que surgen cada año, permitiendo optimizar comprobaciones y reparaciones de computadores automotrices de tecnología aplicada en los autos Aveo Family.

## JUSTIFICACIÓN

En el contexto del problema de investigación relacionado a los recursos tecnológicos del banco de pruebas y las fallas de las Ecu, se propone un nuevo método de diseño mediante el uso de una plataforma de desarrollo de microcontrolador (AVR) (Autosoporte, 2021), encargada del control de las entradas (sensores conectados al computador), y las salidas (actuadores como bobinas, etc.). En este contexto se busca desarrollar un banco de pruebas mediante una innovadora técnica de simulación de funcionamiento para la unidad de control electrónico (ECU) (Ribeiro, 2018); los entes que se beneficiaran del proyecto están en los estudiantes, docentes, técnicos, entre otros que tengan acceso a esta investigación, serán los favorecidos con técnicas que procuran mejorar el aprendizaje en procesos electrónicos automotrices, que van desde la medición de señales características en cada uno de los sensores ( Grupo Editorial CEAC, 2017), hasta la interpretación, valoración y prueba de la mismas en todo el sistema, junto con la comprobación en el funcionamiento de señales en las bobinas inyectores entre otros (Crouse, 2017). Este banco de pruebas tiene un gran beneficio a partir de la emulación del funcionamiento de señales (TPS, MAP, IAT, WTS, O2) y actuadores relacionados a la ECU (Guerrero, 2018). Mediante el banco de pruebas se podrá verificar los procesos relacionados con la unidad de control electrónico del motor a diferentes rangos de trabajo sin tener la necesidad estricta de estar conectada a el vehículo (Martí, 2018). En este contexto, el banco puede ser utilizado como una herramienta de diagnóstico para unidades de control electrónico, volviéndose un recurso útil en situaciones donde no se puede movilizar el vehículo y que se tiene la necesidad de analizar el funcionamiento de estos elementos. En este proyecto se propone técnicas secuenciales en los procesos de diseño e implementación que funcionen para simular el funcionamiento de una unidad de control electrónico del vehículo. El banco de pruebas tiene propósito de usarse, como una herramienta de apoyo

## **OBJETIVOS**

### **Objetivo General**

Diseñar y construir un banco de pruebas, mediante la simulación los sensores empleados en el sistema de inyección electrónica a gasolina de un vehículo Aveo Family, utilizando componentes eléctricos y electrónicos, para la comprobación del correcto funcionamiento de la ECU del automotor

### **Objetivos Específicos**

- Investigar el diseño de un banco de pruebas mediante etapas de comprobación además de verificación, para probar el funcionamiento de la computadora automotriz con un sistema de inyección electrónica del motor del aveo family.
- Diseñar e implementar un banco de pruebas, mediante una estructura electro automotriz, para los caos de uso en posibles fallas, de circuitos eléctricos y electrónicos además de la programación de los sistemas involucrados.
- Evaluar el banco de pruebas, con test de funcionamiento electro automotriz, para la comprobación y verificación de fallas en el computador además de la visualización de resultados.

## MARCO TEÓRICO

### PARÁMETROS CARACTERÍSTICOS DE OPERACIÓN DE COMPUTADORES AUTOMOTRICES EN SISTEMAS DE INYECCIÓN ELECTRÓNICA

#### INTRODUCCIÓN AL SISTEMA DE INYECCIÓN A GASOLINA

#### COMPUTADORA AUTOMOTRIZ

La mayoría de las funciones de tu auto están controladas por la computadora del motor, también conocida como Unidad de Control del Motor (ECU), Módulo de Control del Motor (ECM) o Módulo de Control del Tren Motriz (PCM). (Bosch, 2015)

Al igual que la computadora que tienes en casa, estos equipos procesan mucha información y pueden ejecutar una serie de funciones. En un auto, la computadora automotriz monitorea o controla una gran cantidad de procesos que mantienen el motor funcionando a su nivel óptimo, mostrado en la Figura No. 1. Asimismo, es capaz de adaptarse a las condiciones ambientales y a tus propios patrones de conducción para garantizar que tu vehículo funcione de la manera más eficiente y segura posible. (Rueda Santander, 2005)



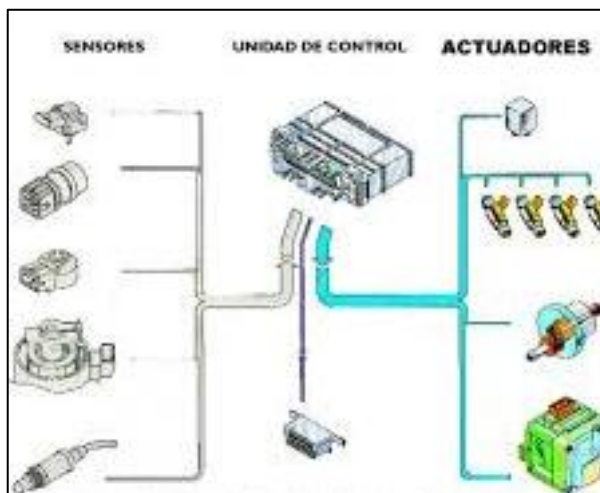
**Figura No. 1** Ejemplo de computador automotriz

**Elaborado por:** (Corti, 2014)

**Fuente:** Datos de la investigación

Sus funciones más elementales son el control del consumo de combustible y la emisión de gases, pero también se ocupan de asuntos más complejos como el sistema de inyección, el

sistema de frenado anti bloqueo (ABS) y el control de tracción, indicado en la Figura No. 2, cada vez más común en los autos modernos. (Yerera, 2017)



**Figura No. 2** Ejemplo de computador automotriz, sensores y actuadores

**Elaborado por:** (Corti, 2014)

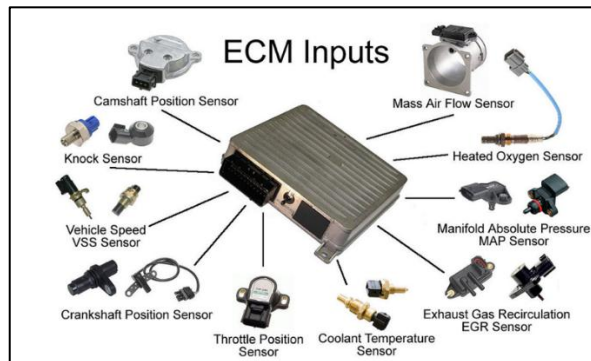
**Fuente:** Datos de la investigación

## **TIPOS DE COMPUTADORAS AUTOMOTRICES ECU-ECM**

Existen diversos tipos de computadoras automotrices cada uno con una función en específico. (Pardiñas, 2012) A continuación estaremos dando a conocer cada uno de ellos:

### **ECM Electronic Control Module**

Principalmente este tipo de modulo es el encargado de llevar el registro de todos los sensores, es decir que, si algún sensor arroja que alguna pieza del motor está fallando, el ECM será el encargado de recibir dicha información, mostrado en la Figura No. 3, para posteriormente guardarla en la memoria y establecer un código de falla. (Corti, 2014)

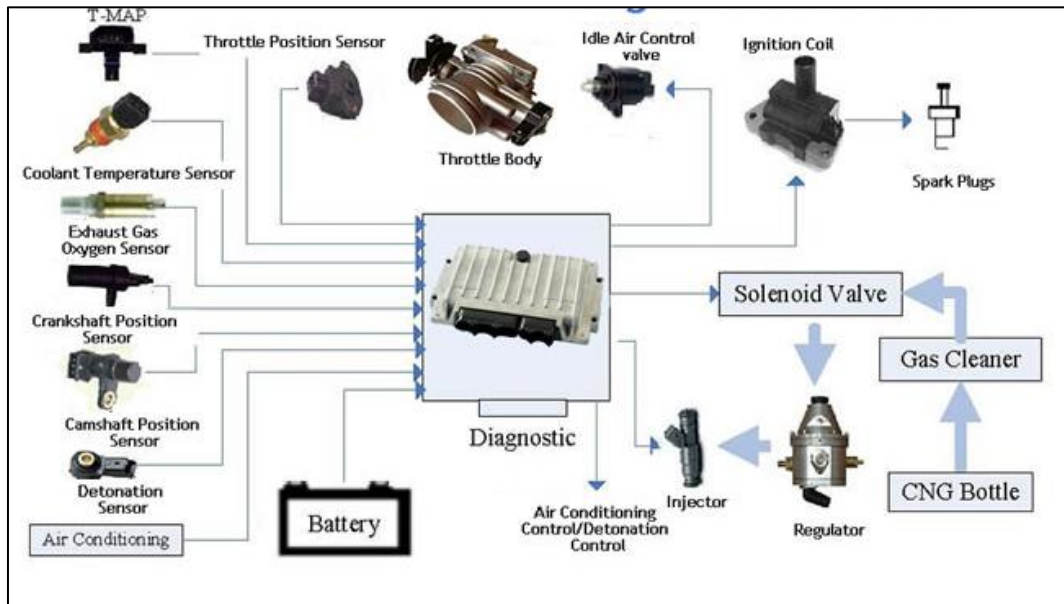


**Figura No. 3** Ejemplo de ECM (Electronic Control Module), sensores y actuadores  
**Elaborado por:** (Corti, 2014)  
**Fuente:** Datos de la investigación

### Unidad de control electrónico ECU

La función de la unidad de control electrónico es recibir los datos enviados por los sensores para procesar y calcular las señales de activación los actuadores. Se optimiza el control y regulación del vehículo ya que permite procesar fácilmente muchos factores que influyen en el funcionamiento. La ECU está formada por una placa electrónica en una caja de aluminio. La alimentación, las entradas de los sensores y las salidas de los conectores se encuentran unidos a la unidad de control por medio de un conector. El funcionamiento de la ECU debe ser óptimo pese a las condiciones del ambiente en que se encuentra, indicado en la Figura No. 4. Por lo que debe tener buena resistencia a la temperatura del entorno, a las sustancias químicas utilizadas en los motores, humedad, y resistencia mecánica a las vibraciones (Bauer, 2002) (Bosch, 2015).

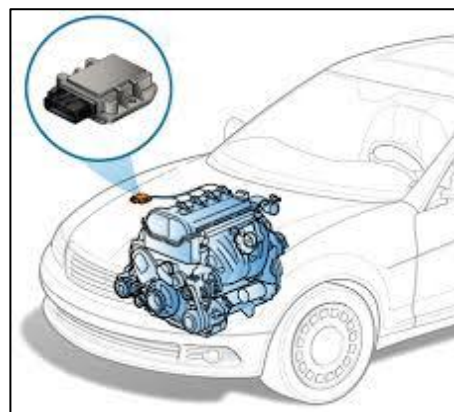




**Figura No. 4** Ejemplo de (Unidad de control electrónico ECU), sensores y actuadores  
**Elaborado por:** (Bosch, 2015)  
**Fuente:** Datos de la investigación

### PCM: Powertrain Control Module

Trabaja en conjunto con la ECM ya que esta es la encargada de recibir información igualmente proveniente de los sensores, la prepara, hace una comparación de los datos, y finalmente la devuelve a la ECM para que a su vez guarde dicha información, indicado en la Figura No. 5, misma que estará disponible para ser leída por el scanner. (TEXA., 2013)



**Figura No. 5** Ejemplo de (PCM: Powertrain Control Module).  
**Elaborado por:** (TEXA., 2013)  
**Fuente:** Datos de la investigación

## BCM: Body Control Module

Es la encargada de chequear, verificar, y medir partes del cuerpo del vehículo, como, por ejemplo, se ocupa de que cuando se encienda el auto con las llaves del mismo, este responda de la manera correcta, se encarga también del control de climatización del automóvil (A/C), del correcto funcionamiento de los espejos eléctricos, da datos detallados sobre las puertas, si están abiertas o cerradas, indicando además el lado, indicado en la Figura No. 6. Todos estos datos son enviados a la computadora principal es decir la ECM. (Yerera, 2017)

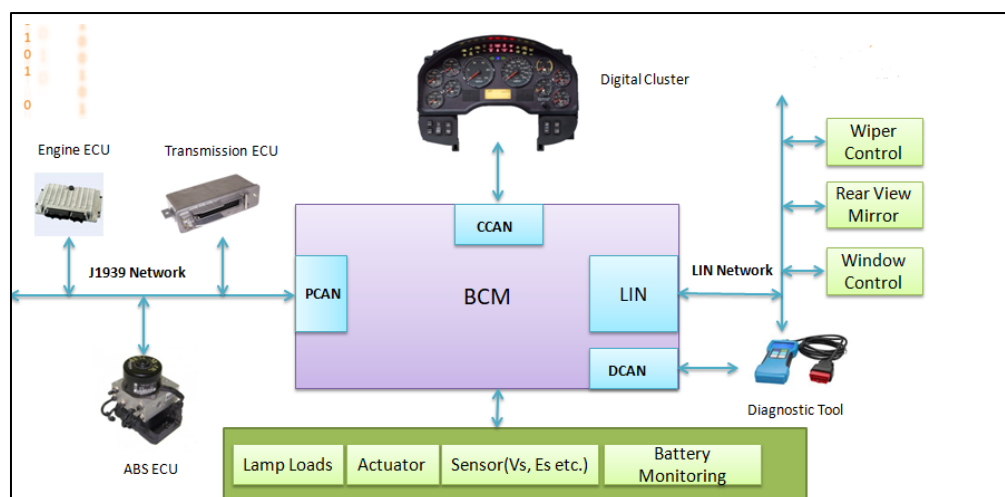


Figura No. 6 Ejemplo de (BCM: Body Control Module).

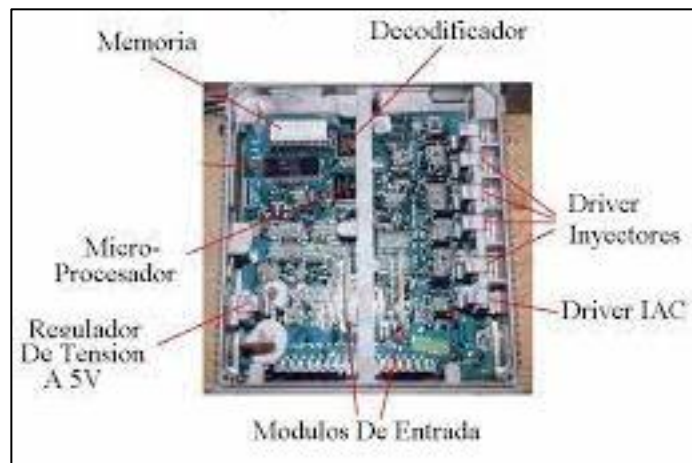
Elaborado por: (Yerera, 2017)

Fuente: Datos de la investigación

## COMPONENTES

La mayoría de los circuitos que se encuentran al interior de una unidad de control están destinados a operaciones donde se busca para la estabilidad en este proceso, indicado en la Figura No. 7, el siguiente es un resumen de las partes más básicas de un módulo de control Automotriz (TEXA., 2013) (Consejo Nacional de Educación para la Vida y el Trabajo, 2010):

1. Alimentación o Circuito Fuente
2. CPU o Circuito de procesamiento de datos
3. Drivers o Salidas
4. Circuito de Datos y Periferia



**Figura No. 7 Ejemplo de componentes computadora automotriz.**  
**Elaborado por:** (TEXA., 2013)  
**Fuente:** Datos de la investigación

## **METODOLOGÍA DEL PROYECTO - DESARROLLO**

### **DISEÑO Y CONSTRUCCIÓN DEL BANCO DE PRUEBAS PARA COMPROBACIÓN Y VERIFICACIÓN DE COMPUTADORAS AUTOMOTRICES CON INYECCIÓN ELECTRÓNICA A GASOLINA**

#### **ESCRIPCIÓN GENERAL DEL SISTEMA**

El dispositivo es un sistema digital de generación de señales con frecuencia, ciclo útil y combinación configurables, para controlar la conmutación de inyectores de motores gasolina de baja y alta impedancia.

Los mecanismos que constituyen el sistema permiten:

- Controlar en tiempo real la conmutación de inyectores, permitiendo ajustar la frecuencia y ciclo útil.
- Representar los distintos procesos de inyección, como son simultánea, secuencial y semisequencial.
- Controlar electrónicamente la presión de la bomba eléctrica de combustible a través del método de modulación de ancho de pulso.
- Utilizar un sistema de retroalimentación para determinar individualmente la conexión correcta de los inyectores al sistema de control.
- Realizar pruebas de trabajo para determinar el haz de pulverización, la estanqueidad y el caudal suministrado por los inyectores.

#### **DISEÑO DEL SISTEMA**

Los requisitos se detallan teniendo en cuenta las condiciones del ambiente de trabajo, las especificaciones de potencia y las funciones que realiza el dispositivo.

##### **Requisitos en base al ambiente de trabajo**

- Potencia de transmisión y sensibilidad de recepción adecuada para permitir una distancia de separación considerable entre los dos nodos de la red inalámbrica.
- Capacidad de coexistencia con otros dispositivos inalámbricos que trabajen en la

misma banda con la mínima interferencia posible.

- Inmunidad a la interferencia eléctrica generada por elementos inductivos conectados al sistema de control.
- Capacidad de trabajo con severas temperaturas, vibraciones y ambientes sucios y húmedos.

### **Requisitos en base a especificaciones de potencia**

- Manejo de potencia suficiente para conmutar las cargas resistivas e inductivas administradas por el sistema.
- Consumo eléctrico reducido en estado de espera.
- Entrega de potencia eficiente por parte de los reguladores de voltaje, para permitir el funcionamiento seguro de todos los dispositivos electrónicos que forman parte del dispositivo.
- Protección contra conexión invertida y picos de voltaje, para evitar daños prematuros y permanentes en el dispositivo.

### **Requisitos en base al desempeño solicitado**

- Realizar pruebas de trabajo para determinar el haz de pulverización, la estanqueidad y el caudal suministrado por los inyectores.
- Alta velocidad de adquisición, procesamiento y ejecución.
- Interfaz gráfica explícita y legible, para la configuración de parámetros y visualización de variables.
- Exactitud en la base de tiempo utilizada para la generación de señales cuadradas.
- Vida útil larga con mantenimiento mínimo.

### **Diseño del hardware**

El comprobador de inyectores gasolina con comunicación inalámbrica se puede utilizar con cualquier tipo de inyector gasolina sea de baja o alta impedancia.

## Hardware de control y visualización

### Microcontrolador principal

El ATMEGA328P se encarga del procesamiento de la información, al adquirir todos los datos de las variables externas, manipularlos y entregarlos de forma adecuada para la visualización y envío remoto.

La siguiente **¡Error! No se encuentra el origen de la referencia.**, resume el destino de conexión de cada uno de los pines del microcontrolador. Define además si son entradas o salidas, y en el caso de ser entradas, si éstas son análogas o digitales

**Tabla 1** *Conexión del microcontrolador.*

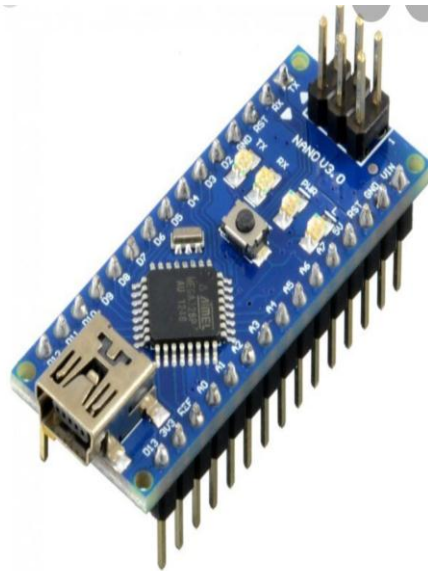
	NOMBRE	PIN	E/S	A/D	FUNCIÓN / OBSERVACIÓN
PORTA	RA0/AN0	2	E	D	SEÑAL BAJO NIVEL DE LÍQUIDO
	RA1/AN1	3	S	D	CONTROL INYECTOR 1
	RA2/AN2	4	S	D	CONTROL INYECTOR 2
	RA3/AN3	5	S	D	CONTROL INYECTOR 3
	RA4/T0CKI	6	S	D	CONTROL INYECTOR 4
	RA5/AN4	7	S	D	CKP
PORTB	RB0/INT0	33	-	-	-
	RB1/INT1	34	-	-	-
	RB2/INT2	35	-	-	-
	RB3/CCP2	36	S	D	VENT
	RB4	37	S	D	CMP1
	RB5/PGM	38	S	D	CMP2
	RB6/PGC	39	S	D	IGN
	RB7/PGD	40	S	D	START
PORTC	RC0/TI0SO	15	E	D	TPS
	RC1/TI0SI/CCP2	16	E	D	MAP
	RC2/CCP1	17	S	D	IAT
	RC3/SCK/SCL	18	S	D	WTS
	RC4/SDI/SDA	23	S	D	02
	RC5/SDO	24	-	-	VENT
	RC6/TX/CK	25	S	D	MAIN
	RC7/RX/DT	26	E	D	F. PUMP
PORTD	RD0/PSP0	19	S	D	CONTROL COIL 1
	RD1/PSP1	20	S	D	CONTROL COIL 2

	RD2/PSP2	21	S	D	CONTROL COIL 3
	RD3/PSP3	22	S	D	CONTROL COIL 4
	RD4/PSP4	27	S	D	CONTROL IAC 1
	RD5/PSP5	28	S	D	CONTROL IAC 2
	RD6/PSP6	29	S	D	CONTROL IAC 3
	RD7/PSP7	30	S	D	CONTROL IAC 4
PORTE	RE0/AN5	8	S	D	CONTROL LUZ DE PANEL
	RE1/AN6	9	S	D	ACTIVACIÓN BUZZER
	RE2/AN7	10	S	D	ACTIVACIÓN BUZZER

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

La Figura No. 8, se muestra el circuito principal de conexión del microcontrolador. Además del circuito de reseteo en el pin 1 y el circuito generador de reloj en los pines 13 y 14, existe el circuito para programación serial. Este se usa para actualizar el firmware sin necesidad de sacar el microcontrolador de la placa de control.

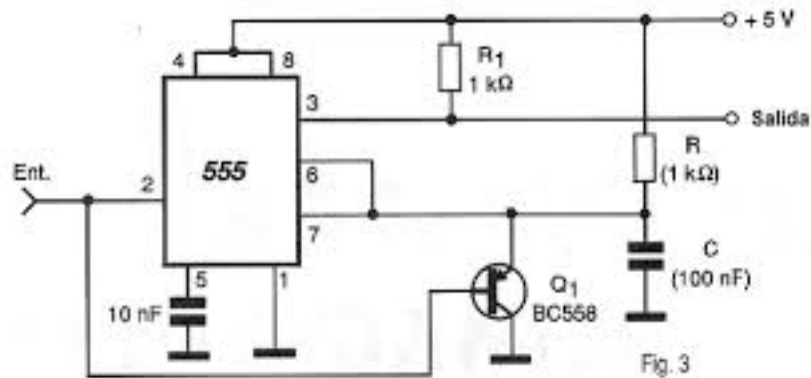


**Figura No. 8 Microcontrolador principal**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

## Detector de pulsos



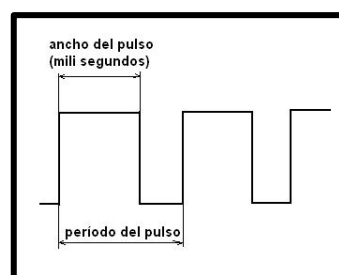
**Figura No. 9 Detector de pulsos**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación, INCB

El circuito de Figura No. 9, indica la configuración electrónica para detectar la correcta conexión de un inyector. Este mismo circuito se repite en función de la cantidad de inyectores existentes en el banco.

Como se puede observar en la siguiente imagen, cada vez que el inyector se apaga, genera un voltaje pico mayor al voltaje máximo proporcionado por la fuente de alimentación. Este voltaje se usa como recurso principal para el circuito de detección.



**Figura No. 10 Pulsos eléctricos funcionamiento de inyector**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación, INCB

El voltaje pico ingresa al amplificador operacional en configuración de comparador de



ventana, a través de la resistencia. El valor considerable de esta resistencia permite el acoplamiento de impedancias entre el circuito y el inyector, disminuyendo además el valor del provocando además una caída en el voltaje pico, como se puede apreciar en la Figura No. 10.

A través de un divisor de voltaje, se calibra el valor de umbral. Entonces, si el voltaje de entrada es mayor que el valor en la salida 1 se tiene 12V. Si el valor es menor o igual, se tiene 0V en la salida.

Un multivibrador monoestable, genera un pulso de 1706 mseg cada vez que existe un flanco descendente en su terminal de disparo. El ancho de pulso está definido por la siguiente ecuación:

$$T = 1,1 * R * C$$

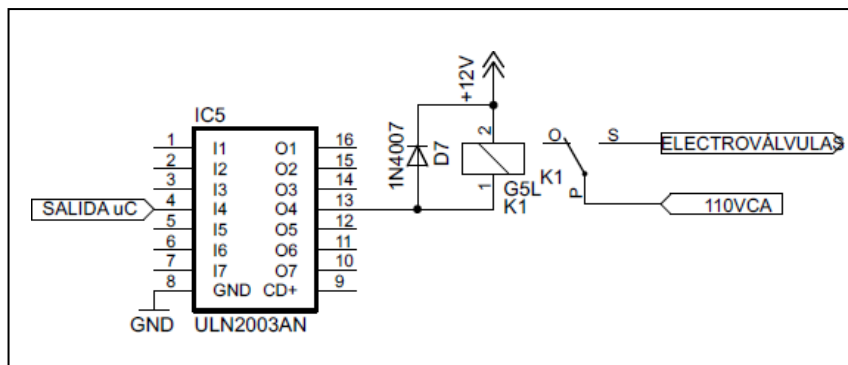
$$T = 1,1 * 47K\Omega * 33\mu F = 1706,1mseg$$

**Ecuación 1:** Ancho de pulso para un multivibrador monoestable

Cuando la salida del LM555 se pone en alto, el arreglo coloca 0V en su salida; en el estado contrario, el integrado desconecta su salida, permitiendo que la resistencia coloque 5V. La resistencia R y capacitor C forman un filtro pasa bajos que mantiene el estado de “1L” durante un tiempo más prolongado.

### Subsistema de respuesta

### Interfaces de potencia electromecánicas



**Figura No. 11 Interfaces de potencia electromecánicas**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación, Cabrera P. - Nato E.

El arreglo de semiconductores, permite al mecanismo tener la corriente necesaria para manejar los relés electromecánicos que conmutan los indicadores de las electroválvulas e indicadores del panel, mostrados en la Figura No. 11.

Cuando la salida del microcontrolador se pone en alto, el arreglo lo coloca 0V de hasta 500 mA en su salida conectada al relé. Esto provoca que el elemento electromecánico se active y pueda conmutar cargas de hasta 1000W en corriente alterna. El diodo D protege a IC de voltajes reversos inducidos en la bobina del relé cuando éste se apaga.

**Inyectores**

El sistema de potencia puede manejar hasta cuatro inyectores gasolina de baja o alta impedancia, controlados por cuatro canales distintos, indicado en la Figura No. 12. Esta configuración permite que el banco pueda emular los distintos tipos de inyección, así como también distintas frecuencias y ciclos útiles.



**Figura No. 12 Inyector módulo**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

## Interfaces de potencia de estado sólido

Se utiliza este tipo de interfaces basadas en transistores de potencia porque los inyectores y la bomba de combustible, son cargas que se manejan con señales cuadradas de alta frecuencia y moduladas en ancho de pulso.

El transistor de potencia seleccionado es el MOSFET canal N IRF560. Las ventajas que presenta ante los bipolares es su gran resistencia a altas temperaturas de operación y manejo de corriente mucho mayor, de la **¡Error! No se encuentra el origen de la referencia.** Además, cuando se excita tiene una resistencia fuente – drenaje muy pequeño, lo que provoca una caída de potencia menor en su interior. Puede trabajar a altas frecuencias y permite el paralelaje.

**Tabla 2 Características del transistor IRF.**

Tensión drenaje – fuente	200V
Corriente continua de drenaje	18A
Temperatura operativa máxima	150°C
Resistencia fuente – drenaje	0.180Ω

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

En lo referente a su circuito manejador, cuando la salida del microcontrolador se pone en alto, el transistor NPN se polariza y permite el flujo de corriente entre emisor y colector, identificado en la Figura No. 13. Entonces, se asegura un voltaje cercano a 12V en la compuerta, que lo hace conducir.



**Figura No. 13 Acople eléctrico salidas inyectoras**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

Cuando la salida del microcontrolador se pone en bajo, el transistor NPN no conduce y por lo tanto, la resistencia de 100 ohms se pone en circuito abierto. Así, la resistencia de 10x3 ohms asegura un voltaje igual a 0V, lo que provoca que el MOSFET no conduzca.

### **Subsistema de alimentación**

La fuente de alimentación es la encargada de suministrar la energía que tanto la lógica como el sistema de potencia necesitan para funcionar. Es del tipo conmutada (realiza la transformación de energía eléctrica mediante transistores en conmutación). Sus características técnicas son:

**Tabla 3 Características de la fuente de alimentación**

Tensión de entrada	110 – 220VAC
Frecuencia	50 / 60Hz
Tensión de salida	12VDC $\pm$ 5%
Corriente de salida constante	30A

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

Es importante resaltar que existen dos reguladores de voltaje. El primero es usado para la alimentación constante de toda la lógica, mostrado en la Figura No. 14. El segundo soporta las interfaces como scanner OBDII.



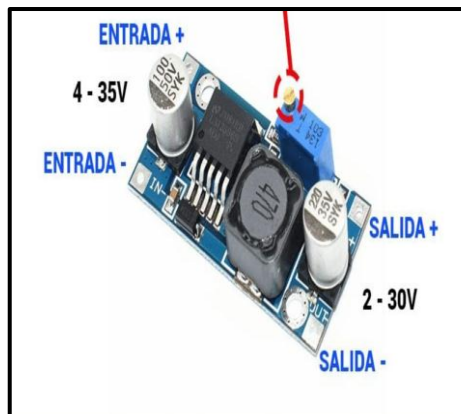
**Figura No. 14 Fuente de alimentación**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

### **Regulador de voltaje principal**

La tensión de alimentación de la fuente conmutada se filtra y se aplica regulada al resto del circuito. El regulador convierte el voltaje de 12V de entrada en una tensión constante de 5.0V. En la Figura No. 15, la fuente de alimentación consta de un rectificador, una protección por sobretensión, un filtro, un regulador lineal fijo de 9V y un regulador variable calibrado a 5V.

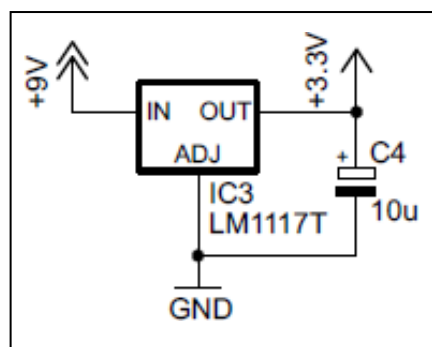


**Figura No. 15 Regulador de voltaje principal**  
**Elaborado por:** William Andrés Acurio Barriga  
**Fuente:** Datos de la investigación

El rectificador recorta picos negativos de voltaje y protege al circuito cuando por error, se invierte la polaridad de la fuente en el momento de la instalación. El zener, resguarda al circuito de sobrevoltajes instantáneos. La regulación se hace en etapas: primero se regula a 9V y luego a 5V. De esta manera, la caída de voltaje se disipa en algunos integrados, asegurando que los reguladores de voltaje se calienten menos.

### **Regulador de voltaje secundario**

Esta fuente de alimentación es similar a la principal. Su diferencia radica en que se basa en el regulador fijo a 3.3V. Se necesita este voltaje para soporta las interfaces como scanner OBDII, mostrado en la Figura No. 16.



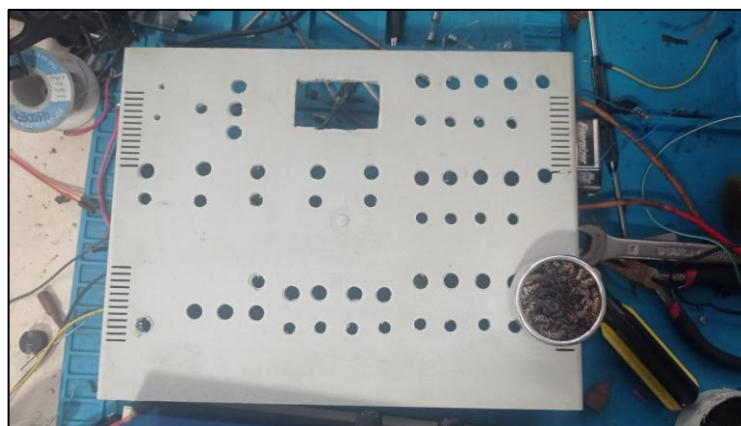
**Figura No. 16 Regulador de voltaje secundario**  
**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación, Cabrera P. - Nato E.

## **CONSTRUCCIÓN FÍSICA DEL SISTEMA**

### **Construcción de la estructura**

La estructura del banco de inyectores se realizó de material paneles pultrusionados de poliéster reforzado con fibra de vidrio E (PRFV) de sección hueca, esto permite optimizar la relación peso/inercia de 5mm de grosor; al cual su acabado es mediante un tratamiento de pintura mono capa, en la que se procederá a montar nuestras interfaces de potencia y placas de energía, comunicación entre módulo y ecu, todo esto indicado en la Figura No. 17.



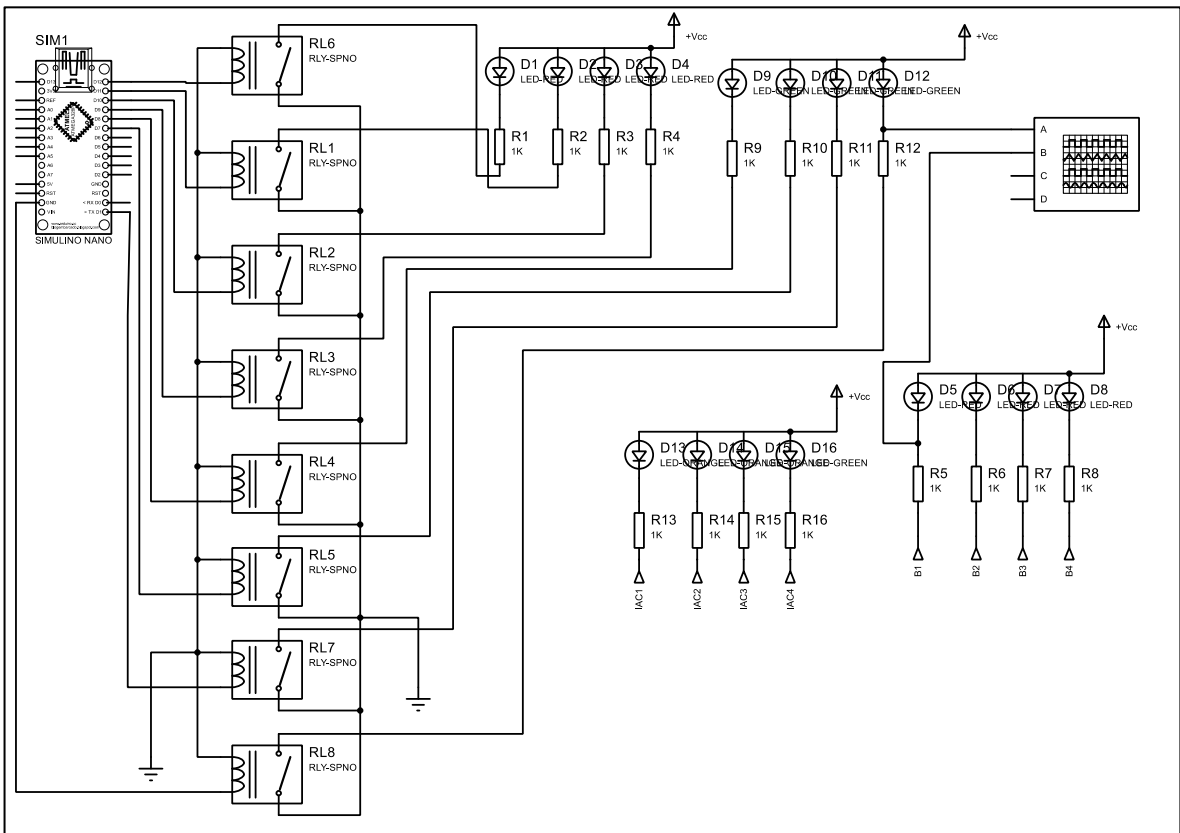
**Figura No. 17 Estructura de plástico.**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

## **SIMULACIÓN EN PROTEUS**

Una de las ventajas que ofrece el Atmega328p es de simular su funcionamiento por medio del software PROTEUS el cual es de gran ayuda para poder depurar errores antes del montaje del circuito electrónico en el banco de pruebas, de la **¡Error! No se encuentra el origen de la referencia..** Al momento de correr la simulación la adquisición de señales de entrada y salida reporta una operación satisfactoria y la pantalla LCD muestra información de la conexión entre la interfaz gráfica y el sistema de potencia están en modo de aviso del uso de energía del módulo.

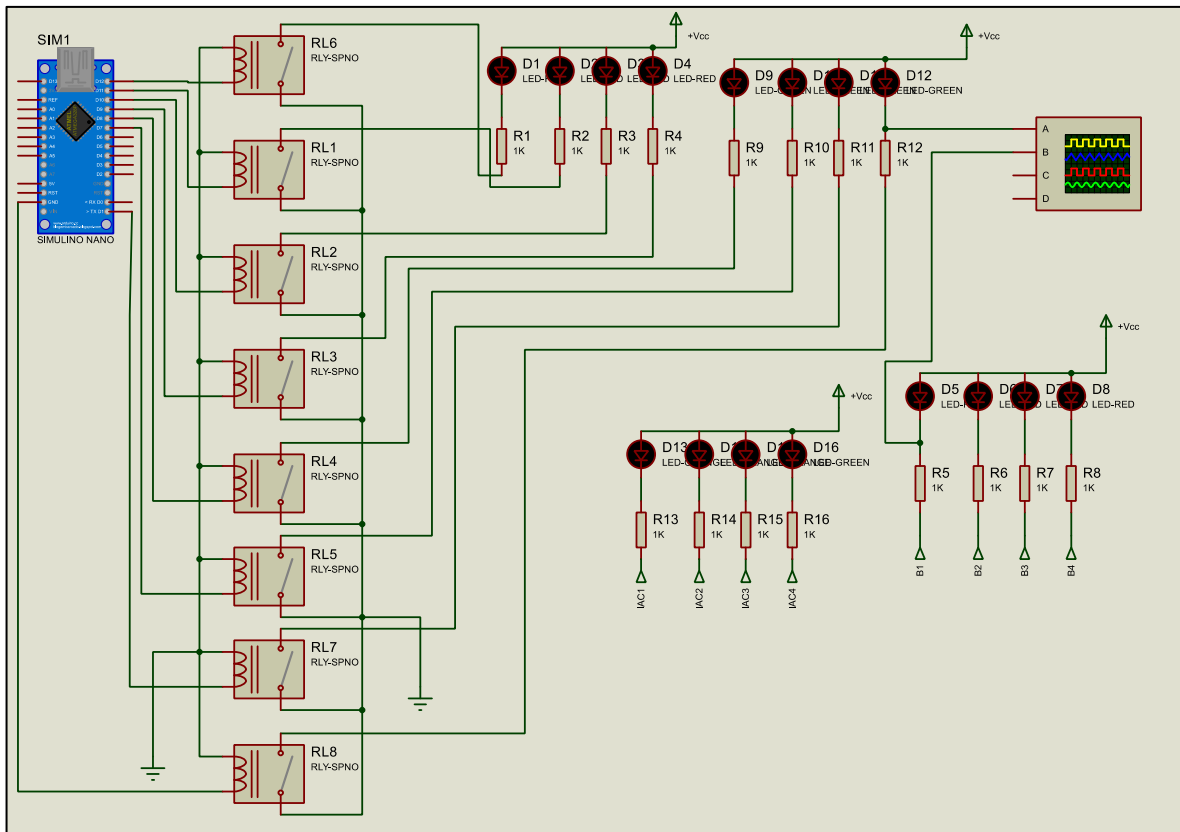


**Figura No. 18 Simulación** sincronización módulo  
**Elaborado por:** William Andrés Acurio Barriga  
**Fuente:** Datos de la investigación

Una vez generada la conexión la adquisición de señales de entrada y salida reporta que las interfaces están sincronizadas adecuadamente y la pantalla LCD, despliega la información de voltaje y corriente del sistema, el cual se encuentra en modo de transmisión de las órdenes de comando de cada una de las pruebas que va a realizar enviadas desde la ecu 1 microcontrolador y viceversa indicándonos aquí la etapa de control del banco de pruebas. Donde los leds del 1al 4 nos muestran el estado de los 4 inyectores, leds del 9 al 12 estado main, bomba(f. pump) además de ventiladores, los led del 13 a 16 indican señales de los 4 sensores IAC, leds del 5 al 8 muestran las 4 bobinas; además los relés del 1 al 8 manejan las señales del microcontrolador las mismas que dependen de las señales de la ECU, cabe recalcar que funciona con una lógica positiva la cual cierra el circuito con el negativo de todo el sistema incluyendo la interfaz de comunicación, todo esto mostrado en **¡Error! No**



se encuentra el origen de la referencia..



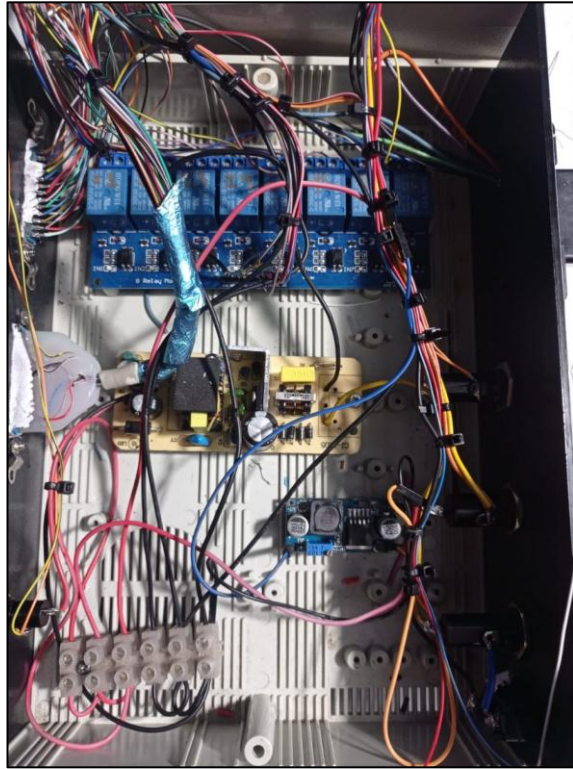
**Figura No. 19 Simulación sincronización módulo**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

### Simulación de funcionamiento

Previo a la simulación del circuito en el software PROTEUS probamos en el protoboard la comunicación y funcionamiento de las placas para poder depurar algún error de funcionamiento, indicado en las **¡Error! No se encuentra el origen de la referencia.** y **¡Error! No se encuentra el origen de la referencia..**



**Figura No. 20 Prueba en protoboard placas de potencia.**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

Interferencia es el principal efecto de caer la comunicación.



**Figura No. 21 Ordenamiento del cableado articulado.**

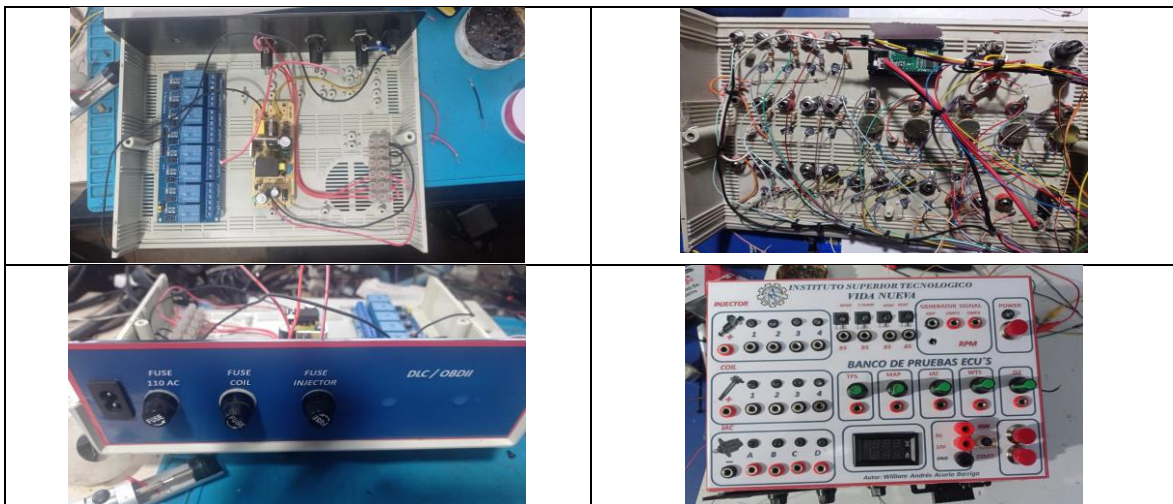
**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

## Montaje y conexión del dispositivo

### Montaje del sistema electrónico

El éxito de un correcto funcionamiento del sistema electrónico viene dado por la seguridad en la conexión de las placas ya que van conectadas las 3 en transmisión y recepción de datos por medio de 3 bus de datos. Además, se tomó en cuenta la amplitud y espacio para optimizar nuestro orden de ubicación del sistema electrónico, se puede ver en la Figura No. 22 *Instalación del cableado de la pantalla*



**Figura No. 22** *Instalación del cableado de la pantalla*

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

La pantalla LCD también se instala en el montaje del sistema electrónico ya que su bus de datos esta entrelazada con la ecu y el microcontrolador principal, mostrado en la **¡Error!**  
**No se encuentra el origen de la referencia..**



**Figura No. 23 Pantalla LCD instalada.**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación.

## RESULTADOS - RESPUESTA

### PROCEDIMIENTOS, FUNCIONAMIENTO Y PRUEBAS DEL EQUIPO

#### SIMULACIÓN Y FASE DE PRUEBA

Esta prueba se realiza antes de asegurar la placa plástica que soporta al módulo, la placa de potencia principal, la pantalla LCD y el voltímetro para tener acceso al sistema electrónico y depurar algún tipo de error o de interferencia al módulo, indicado en la Figura No. 24.



**Figura No. 24 Testigo módulo funcional.**

**Elaborado por:** William Andrés Acurio Barriga  
**Fuente:** Datos de la investigación.

La ubicación del conector hembra del módulo debe estar visible y libre de algún cableado porque puede generar interferencia perdiendo la comunicación con la interfaz gráfica.



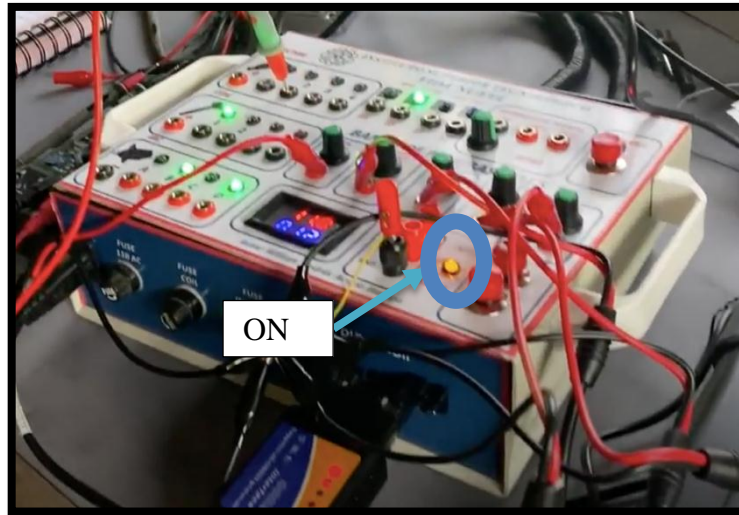
**Figura No. 25** Testigo módulo apagado led en estado intermitente.

**Elaborado por:** William Andrés Acurio Barriga  
**Fuente:** Datos de la investigación.



**Figura No. 26** Prueba de conexión apagada satisfactoria.

**Elaborado por:** William Andrés Acurio Barriga  
**Fuente:** Datos de la investigación.



**Figura No. 27 Testigo módulo encendido led en estado intermitente.**  
 Elaborado por: William Andrés Acurio Barriga  
 Fuente: Datos de la investigación.

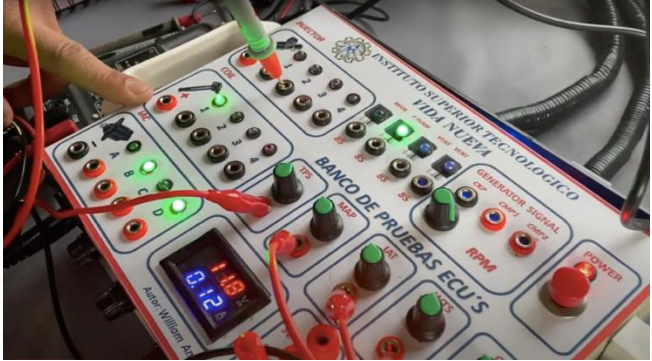

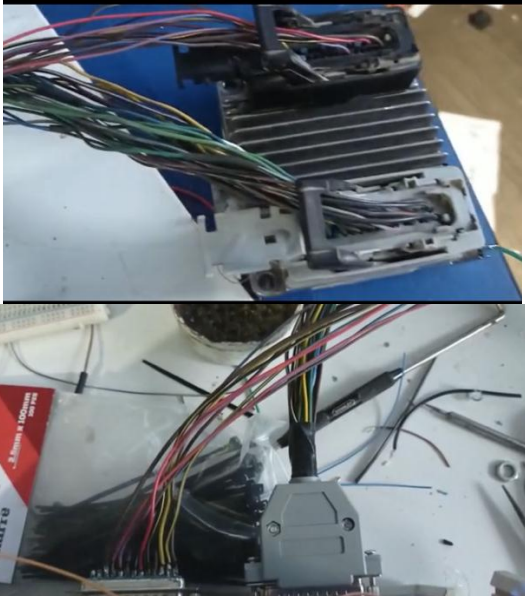


**Figura No. 28 Prueba de conexión encendido satisfactoria.**  
 Elaborado por: William Andrés Acurio Barriga  
 Fuente: Datos de la investigación.

## COMPROBACIÓN DE FUNCIONAMIENTO DEL BANCO DE PRUEBAS DE COMPUTADORES

La fase de prueba se realizó de manera satisfactoria colocamos la tapa plástica y aseguramos con los pernos de fijación a la estructura para finalmente usar al 100% libre de errores el banco de pruebas, mostrado en la **¡Error! No se encuentra el origen de la referencia.** .

Tabla 4 *Parámetros de consideración de las pruebas.*

<p>Escogemos los inyectores a comprobar con su respectivo acople y montamos al comprobador de inyectores.</p>	
<p>Ajustamos los parámetros de los inyectores los cuales indican su funcionamiento con el led de color rojo</p>	
<p>Revisar la conexión de la ECU y el banco de pruebas para una correcta operación del mismo.</p>	

Encendemos el banco de para realiza la conexión con el lector.



Buscamos nuestra red wi-fi llamada banco de inyectores en las redes inalámbricas disponibles.





Escogemos la red “OBDII” y conectamos.



Abrimos nuestra interfaz gráfica Car Scanner del lector.



Verificar que exista comunicación entre el banco y el programa del lector.



## **PRUEBAS DE FUNCIONAMIENTO**

### **PRUEBA DE PULVERIZACIÓN**

Para emular la entrada de combustible en los inyectores, se ubica en 2 diferentes señales de las partes del inyector, estas son la entrada superior y entrada lateral.

#### **Procedimiento para emulación y prueba de inyectores con entrada superior**

- a) Para emular el funcionamiento, se debe estar consciente del tipo de los inyectores, es importante recordar la verificación los adaptadores derechos adecuados esto se puede ver en el panel de inyectores del distribuidor de combustible en el banco de pruebas.
- b) Al montar los inyectores se debería lubricar los rings en los inyectores en este caso los leds nos indican esta acción.
- c) Al estar conectados internamente los cables de pulso de la ECU. Se puede ingresar a la interfaz gráfica del lector.

Ajuste los parámetros correctos a libre opción del usuario, indicado en la Figura No. 29.



**Figura No. 29 Prueba de pulverización.**  
 Elaborado por: William Andrés Acurio Barriga  
 Fuente: Datos de la investigación.

d) Presione el pulsador para la ignición y el inicio de prueba.



**Figura No. 30 Botón de ignición e inicio de prueba.**

**Elaborado por:** William Andrés Acurio Barriga  
**Fuente:** Datos de la investigación.

Observar el menú de estado de la interfaz gráfica ya que en esta prueba realiza el feedback a los inyectores, esto se puede observar en la Figura No. 30 y Figura No. 31.



**Figura No. 31 Pantalla estado prueba.**  
**Elaborado por:** William Andrés Acurio Barriga  
**Fuente:** Datos de la investigación.

El sistema se detendrá dependiendo de los parámetros enviados por la ECU al terminar la prueba. Además, tiene una alarma de aviso.

Se debe revisar de forma simultánea en la pantalla LCD el control de todos los parámetros de energía. En este contexto todos los parámetros en la prueba de pulverización están funcionando correctamente, así mismo se puede verificar el ángulo de pulverización de cada inyector el cual queda a consideración del usuario que realice el diagnostico de cada inyector.

## PRUEBA DE ESTANQUEIDAD

Este proceso es similar a la prueba de pulverización. Para seleccionar la función de prueba de estanqueidad, escoja los parámetros de tiempo de la prueba y porcentaje de presión de la bomba mediante la variación de los parámetros de los sensores, luego presione el botón "ignición e iniciar prueba", entonces el sistema iniciara. Juzgue la fuga observando si existe la alarma de goteo. Tomar en cuenta, que el goteo por minuto deba ser de menos de una gota, según los estándares del fabricante.

- a) El tiempo defecto designado por el sistema es de 1 minuto, y la presión debe ajustarse a 10% más que la especificación del fabricante.
- b) El sistema se detendrá dependiendo de los parámetros enviados por la ECU al terminar la prueba. Además, tiene una alarma de aviso, como se muestra en la Figura No. 32.



**Figura No. 32 Prueba de estanqueidad.**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación.

- c) Presione el pulsador para la ignición y el inicio de prueba, indicado en la Figura No. 33.

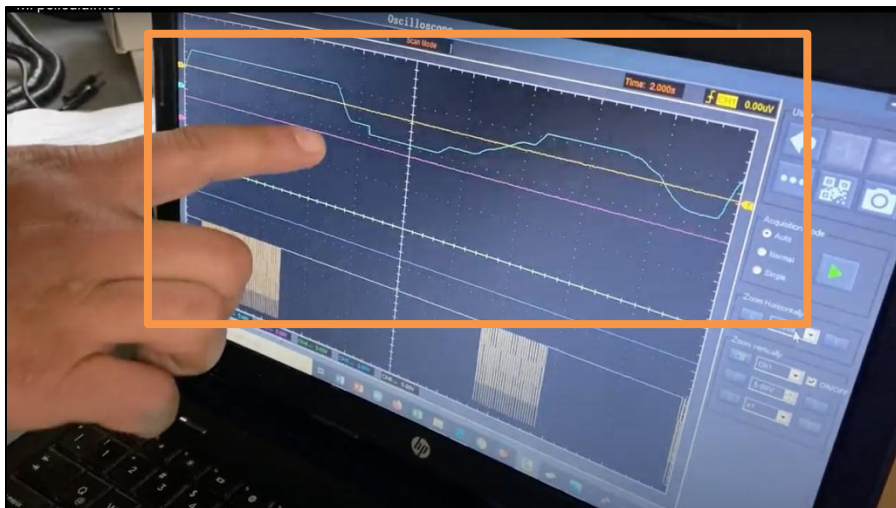


**Figura No. 33 Botón de ignición e inicio de prueba.**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación.

Observar el menú de estado de la interfaz gráfica, en esta se muestra el estado en tiempo real de la prueba, mostrado en la Figura No. 34.




**Figura No. 34 Pantalla estado prueba.**  
**Elaborado por:** William Andrés Acurio Barriga  
**Fuente:** Datos de la investigación.

Se debe revisar de forma simultanea en la pantalla LCD el control de todos los parámetros de energía. De esta forma los parámetros en la prueba de estanqueidad tienen un funcionamiento correcto analizando que no exista goteo de inyector el cual queda a consideración del usuario que realice el diagnostico de cada inyector.

**CHEVROLET AVEO FAMILY 2008**

**Tabla 5 Resultados pruebas de funcionamiento inyectores Chevrolet Aveo Family 2008.**

<b>CHEVROLET AVEO FAMILY 2008- PRUEBA 1</b>	
	
Presión de Comprobación:	(35 PSI)
Resistencia del Bobinado:	Inyector 1: 2.8 Ω Inyector 2: 2.7 Ω Inyector 3: 2.7 Ω Inyector 4: 2.8 Ω
Prueba Estanqueidad:	Inyector 1: ok Inyector 2: ok Inyector 3: goteo Inyector 4: ok




Prueba Pulverización :	Inyector 1: ok Inyector 2: ok Inyector 3: chorro incompleto Inyector 4: ok
Prueba Feedback:	Coil 1: ok Coil 2: ok Coil 3: ok Coil 4: ok
<b>ANALISIS RESULTADOS</b>	En esta prueba los inyectores se encuentran dentro del rango permitido en la variación de parámetros, para este caso de uso se recomienda el cambio de los micros filtros y realizar una limpieza.

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

**Tabla 6 Resultados pruebas de funcionamiento inyectores Chevrolet Aveo Family 2008.**

<b>CHEVROLET AVEO FAMILY 2008- PRUEBA 2</b>	
	
Presión de Comprobación:	(35 PSI)
Resistencia del Bobinado:	Inyector 1: 2.2 $\Omega$ Inyector 2: 2.3 $\Omega$ Inyector 3: 2.3 $\Omega$ Inyector 4: 2.4 $\Omega$

Prueba Estanqueidad :	Inyector 1: ok Inyector 2: ok Inyector 3: ok Inyector 4: ok
Prueba Pulverización:	Inyector 1: ok Inyector 2: ok Inyector 3: ok Inyector 4: ok
Prueba Feedback:	Coil 1: ok Coil 2: ok Coil 3: ok Coil 4: ok
ANALISIS RESULTADOS	En esta prueba los inyectores se encuentran dentro del rango permitido en la variación de parámetros, para este caso de uso se recomienda el cambio de los micros filtros y realizar una limpieza en la tina ultrasónica.

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

**Tabla 7 Resultados pruebas de funcionamiento inyectores Chevrolet Aveo Family 2008.**


<b>CHEVROLET AVEO FAMILY 2008- PRUEBA 3</b>	
	
Presión de Comprobación:	(35 PSI)

Resistencia del Bobinado:	Inyector 1: 12,2 $\Omega$ Inyector 2: 12,1 $\Omega$ Inyector 3: 12,3 $\Omega$ Inyector 4: 12,2 $\Omega$
Prueba Estanqueidad :	Inyector 1: ok Inyector 2: ok Inyector 3: ok Inyector 4: ok
Prueba Pulverización:	Inyector 1: ok Inyector 2: ok Inyector 3: : no hay inyección Inyector 4: ok
Prueba Feedback:	Coil 1: ok Coil 2: ok Coil 3: : no hay inyección Coil 4: ok
ANALISIS RESULTADOS	En esta prueba los inyectores 1, 2, 4, se encuentran dentro del rango permitido de variación de parámetros la cual se recomendaría el cambio de filtros y realizar una limpieza. Así también en el inyector 3, no completó ninguna de las pruebas realizadas recordar que se usa otra ECU, además el reporte de feedback, indica un estado de desconexión razón por la cual la ECU presenta daños en el inyector 3. Se recomienda el recambio del integrado que maneja ese inyector para su posterior prueba en el banco.

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

Tabla 8 Resultados pruebas de funcionamiento inyectores Chevrolet Aveo Family 2008.

<b>CHEVROLET AVEO FAMILY 2008- PRUEBA 4</b>	
	
Presión de Comprobación:	(35 PSI)
Resistencia del Bobinado:	Inyector 1: 14,2 Ω Inyector 2: 14,2 Ω Inyector 3: 14,3Ω Inyector 4: 14,2Ω
Prueba Estanqueidad :	Inyector 1: ok Inyector 2: ok Inyector 3: ok Inyector 4: ok
Prueba Pulverización:	Inyector 1: ok Inyector 2: ok Inyector 3: ok Inyector 4: ok
Prueba Feedback:	Coil 1: ok Coil 2: ok Coil 3: ok Coil 4: ok

ANÁLISIS RESULTADOS	En esta prueba los inyectores se encuentran dentro del rango permitido en la variación de parámetros, para este caso de uso se recomienda el cambio de los micros filtros y realizar una limpieza en la tina ultrasónica.
---------------------	---

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación

## CONCLUSIONES

Una vez finalizando este trabajo de investigación se puede considerar las siguientes conclusiones y recomendaciones, a fin de que sean consideradas por quien lo utilice como fuente de información:

- Se diseñó un banco de inyectores gasolina programado y activado por una AVR, mediante comunicación con una ECU en tiempo real.
- Se desarrolló una interfaz de hardware amigable con el usuario, para activar y modificar parámetros para pruebas de caudal, estanqueidad y pulverización con cada sensor correspondiente.
- Se implementó un hardware basado en electrónica automotriz para el monitoreo de señales emuladas desde la ECU a los inyectores y mostrarse tanto en la interfaz de indicadores de salida así como en la pantalla de estado.
- Se cumplieron pruebas de funcionamiento de inyectores, los cuales mostraron las fallas perceptibles de las salidas de la ECU al ser evaluadas, con estos resultados se llega a un diagnóstico así como la detección de la causa.

## RECOMENDACIONES

- Se recomienda elaborar etapas de protección de voltaje e intensidad para la etapa del microcontrolador, esto para pruebas de la programación en un protoboard.
- Se debe analizar los datos mínimos y máximos de voltaje e intensidad para los módulos de comunicación con la ECU, para usar acopladores, los cuales eviten daños internos al momento de comprobar la transmisión.
- Es recomendable no exponer a temperaturas altas o radiaciones intensa a los elementos electrónicos sensibles, además de puntos de soldadura.
- Se debe mantener la señal de nivel de líquido de prueba o combustible sobre el mínimo para protección de los parámetros de inyección, por problemas que puedan surgir asociados a la refrigeración.
- Se deberá verificar que los voltajes de entrada del sistema sean valores convencionales de la red eléctrica, no se conectara el banco de pruebas a tomas de energía eléctrica de 220 V; tampoco, adicionar o realizar conexiones de cosumo de corriente superior al establecido por el la carga del banco.

## REFERENCIAS BIBLIOGRÁFICAS

- Consejo Nacional de Educación para la Vida y el Trabajo. (2010). *Sensor de Flujo de Masa de Aire*. Obtenido de <https://goo.gl/Hnkkuj>: <https://goo.gl/Hnkkuj>
- Corti, E. C. (2014). Transient Spark Advance Calibration Approach. Energy Procedia. En E. C. Corti, *Transient Spark Advance Calibration Approach. Energy Procedia* (Vol. 45, págs. 967-976).
- Bauer, H. (2002). *Regulación electrónica Diesel*. Stuttgart: ROBERT BOSCH GMBH.
- Bosch. (2015). *Inyección Electrónica*,. Obtenido de <https://goo.gl/RX3CRA>: <https://goo.gl/RX3CRA>
- Pardiñas, J. (2012). *Sistemas auxiliares del motor*. MADRID: EDITEX.
- Rueda Santander, J. a. (2005). Manual técnico de Fuel Injection. *Rueda Santander, J. and Paredes Guevara, R. .*
- TEXA. (2013). *Electricidad y sensores en el automovil*. Milán: TEXAEDU.
- Yerera, S. (2017). *Sistemas de Inyección Electrónica*. Buenos Aires: Universidad Nacional de la Plata .



## ANEXOS

### Programación Comunicación de AVR y ECU

```
#include <math.h>

#ifdef __AVR__
#include <avr/pgmspace.h>
#else
#define PROGMEM
#endif

#include <BCAN.h>

#include "OBD.h"

const char PID_NAME_0x00[] PROGMEM = "PIDs supported [01 - 20]";
const char PID_NAME_0x01[] PROGMEM = "Monitor status since DTCs cleared";
const char PID_NAME_0x02[] PROGMEM = "Freeze DTC";
const char PID_NAME_0x03[] PROGMEM = "Fuel system status";
const char PID_NAME_0x04[] PROGMEM = "Calculated engine load";
const char PID_NAME_0x05[] PROGMEM = "Engine coolant temperature";
const char PID_NAME_0x06[] PROGMEM = "Short term fuel trim — Bank 1";
const char PID_NAME_0x07[] PROGMEM = "Long term fuel trim — Bank 1";
const char PID_NAME_0x08[] PROGMEM = "Short term fuel trim — Bank 2";
const char PID_NAME_0x09[] PROGMEM = "Long term fuel trim — Bank 2";
const char PID_NAME_0x0a[] PROGMEM = "Fuel pressure";
const char PID_NAME_0x0b[] PROGMEM = "Intake manifold absolute pressure";
const char PID_NAME_0x0c[] PROGMEM = "Engine RPM";
const char PID_NAME_0x0d[] PROGMEM = "Vehicle speed";
const char PID_NAME_0x0e[] PROGMEM = "Timing advance";
const char PID_NAME_0x0f[] PROGMEM = "Intake air temperature";
const char PID_NAME_0x10[] PROGMEM = "MAF air flow rate";
const char PID_NAME_0x11[] PROGMEM = "Throttle position";
const char PID_NAME_0x12[] PROGMEM = "Commanded secondary air status";
const char PID_NAME_0x13[] PROGMEM = "Oxygen sensors present (in 2 banks)";
const char PID_NAME_0x14[] PROGMEM = "Oxygen Sensor 1 - Short term fuel trim";
const char PID_NAME_0x15[] PROGMEM = "Oxygen Sensor 2 - Short term fuel trim";
const char PID_NAME_0x16[] PROGMEM = "Oxygen Sensor 3 - Short term fuel trim";
const char PID_NAME_0x17[] PROGMEM = "Oxygen Sensor 4 - Short term fuel trim";
const char PID_NAME_0x18[] PROGMEM = "Oxygen Sensor 5 - Short term fuel trim";
const char PID_NAME_0x19[] PROGMEM = "Oxygen Sensor 6 - Short term fuel trim";
const char PID_NAME_0x1a[] PROGMEM = "Oxygen Sensor 7 - Short term fuel trim";
const char PID_NAME_0x1b[] PROGMEM = "Oxygen Sensor 8 - Short term fuel trim";
const char PID_NAME_0x1c[] PROGMEM = "OBD standards this vehicle conforms to";
const char PID_NAME_0x1d[] PROGMEM = "Oxygen sensors present (in 4 banks)";
const char PID_NAME_0x1e[] PROGMEM = "Auxiliary input status";
const char PID_NAME_0x1f[] PROGMEM = "Run time since engine start";
const char PID_NAME_0x20[] PROGMEM = "PIDs supported [21 - 40]";
const char PID_NAME_0x21[] PROGMEM = "Distance traveled with malfunction indicator lamp (MIL) on";
const char PID_NAME_0x22[] PROGMEM = "Fuel Rail Pressure (relative to manifold vacuum)";
const char PID_NAME_0x23[] PROGMEM = "Fuel Rail Gauge Pressure (diesel, or gasoline direct injection)";
const char PID_NAME_0x24[] PROGMEM = "Oxygen Sensor 1 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x25[] PROGMEM = "Oxygen Sensor 2 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x26[] PROGMEM = "Oxygen Sensor 3 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x27[] PROGMEM = "Oxygen Sensor 4 - Fuel-Air Equivalence Ratio";
```

```

const char PID_NAME_0x28[] PROGMEM = "Oxygen Sensor 5 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x29[] PROGMEM = "Oxygen Sensor 6 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x2a[] PROGMEM = "Oxygen Sensor 7 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x2b[] PROGMEM = "Oxygen Sensor 8 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x2c[] PROGMEM = "Commanded EGR";
const char PID_NAME_0x2d[] PROGMEM = "EGR Error";
const char PID_NAME_0x2e[] PROGMEM = "Commanded evaporative purge";
const char PID_NAME_0x2f[] PROGMEM = "Fuel Tank Level Input";
const char PID_NAME_0x30[] PROGMEM = "Warm-ups since codes cleared";
const char PID_NAME_0x31[] PROGMEM = "Distance traveled since codes cleared";
const char PID_NAME_0x32[] PROGMEM = "Evap. System Vapor Pressure";
const char PID_NAME_0x33[] PROGMEM = "Absolute Barometric Pressure";
const char PID_NAME_0x34[] PROGMEM = "Oxygen Sensor 1 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x35[] PROGMEM = "Oxygen Sensor 2 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x36[] PROGMEM = "Oxygen Sensor 3 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x37[] PROGMEM = "Oxygen Sensor 4 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x38[] PROGMEM = "Oxygen Sensor 5 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x39[] PROGMEM = "Oxygen Sensor 6 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x3a[] PROGMEM = "Oxygen Sensor 7 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x3b[] PROGMEM = "Oxygen Sensor 8 - Fuel-Air Equivalence Ratio";
const char PID_NAME_0x3c[] PROGMEM = "Catalyst Temperature: Bank 1, Sensor 1";
const char PID_NAME_0x3d[] PROGMEM = "Catalyst Temperature: Bank 2, Sensor 1";
const char PID_NAME_0x3e[] PROGMEM = "Catalyst Temperature: Bank 1, Sensor 2";
const char PID_NAME_0x3f[] PROGMEM = "Catalyst Temperature: Bank 2, Sensor 2";
const char PID_NAME_0x40[] PROGMEM = "PIDs supported [41 - 60]";
const char PID_NAME_0x41[] PROGMEM = "Monitor status this drive cycle";
const char PID_NAME_0x42[] PROGMEM = "Control module voltage";
const char PID_NAME_0x43[] PROGMEM = "Absolute load value";
const char PID_NAME_0x44[] PROGMEM = "Fuel-Air commanded equivalence ratio";
const char PID_NAME_0x45[] PROGMEM = "Relative throttle position";
const char PID_NAME_0x46[] PROGMEM = "Ambient air temperature";
const char PID_NAME_0x47[] PROGMEM = "Absolute throttle position B";
const char PID_NAME_0x48[] PROGMEM = "Absolute throttle position C";
const char PID_NAME_0x49[] PROGMEM = "Absolute throttle position D";
const char PID_NAME_0x4a[] PROGMEM = "Absolute throttle position E";
const char PID_NAME_0x4b[] PROGMEM = "Absolute throttle position F";
const char PID_NAME_0x4c[] PROGMEM = "Commanded throttle actuator";
const char PID_NAME_0x4d[] PROGMEM = "Time run with MIL on";
const char PID_NAME_0x4e[] PROGMEM = "Time since trouble codes cleared";
const char PID_NAME_0x4f[] PROGMEM = "Maximum value for Fuel-Air equivalence ratio, oxygen sensor voltage,
oxygen sensor current, and intake manifold absolute pressure";
const char PID_NAME_0x50[] PROGMEM = "Maximum value for air flow rate from mass air flow sensor";
const char PID_NAME_0x51[] PROGMEM = "Fuel Type";
const char PID_NAME_0x52[] PROGMEM = "Ethanol fuel percentage";
const char PID_NAME_0x53[] PROGMEM = "Absolute Evap system Vapor Pressure";
const char PID_NAME_0x54[] PROGMEM = "Evap system vapor pressure";
const char PID_NAME_0x55[] PROGMEM = "Short term secondary oxygen sensor trim";
const char PID_NAME_0x56[] PROGMEM = "Long term secondary oxygen sensor trim";
const char PID_NAME_0x57[] PROGMEM = "Short term secondary oxygen sensor trim";
const char PID_NAME_0x58[] PROGMEM = "Long term secondary oxygen sensor trim";
const char PID_NAME_0x59[] PROGMEM = "Fuel rail absolute pressure";
const char PID_NAME_0x5a[] PROGMEM = "Relative accelerator pedal position";
const char PID_NAME_0x5b[] PROGMEM = "Hybrid battery pack remaining life";
const char PID_NAME_0x5c[] PROGMEM = "Engine oil temperature";
const char PID_NAME_0x5d[] PROGMEM = "Fuel injection timing";
const char PID_NAME_0x5e[] PROGMEM = "Engine fuel rate";
const char PID_NAME_0x5f[] PROGMEM = "Emission requirements to which vehicle is designed";

```

```
const char* const PID_NAME_MAPPER[] PROGMEM = {
  PID_NAME_0x00,
  PID_NAME_0x01,
  PID_NAME_0x02,
  PID_NAME_0x03,
  PID_NAME_0x04,
  PID_NAME_0x05,
  PID_NAME_0x06,
  PID_NAME_0x07,
  PID_NAME_0x08,
  PID_NAME_0x09,
  PID_NAME_0x0a,
  PID_NAME_0x0b,
  PID_NAME_0x0c,
  PID_NAME_0x0d,
  PID_NAME_0x0e,
  PID_NAME_0x0f,
  PID_NAME_0x10,
  PID_NAME_0x11,
  PID_NAME_0x12,
  PID_NAME_0x13,
  PID_NAME_0x14,
  PID_NAME_0x15,
  PID_NAME_0x16,
  PID_NAME_0x17,
  PID_NAME_0x18,
  PID_NAME_0x19,
  PID_NAME_0x1a,
  PID_NAME_0x1b,
  PID_NAME_0x1c,
  PID_NAME_0x1d,
  PID_NAME_0x1e,
  PID_NAME_0x1f,
  PID_NAME_0x20,
  PID_NAME_0x21,
  PID_NAME_0x22,
  PID_NAME_0x23,
  PID_NAME_0x24,
  PID_NAME_0x25,
  PID_NAME_0x26,
  PID_NAME_0x27,
  PID_NAME_0x28,
  PID_NAME_0x29,
  PID_NAME_0x2a,
  PID_NAME_0x2b,
  PID_NAME_0x2c,
  PID_NAME_0x2d,
  PID_NAME_0x2e,
  PID_NAME_0x2f,
  PID_NAME_0x30,
  PID_NAME_0x31,
  PID_NAME_0x32,
  PID_NAME_0x33,
  PID_NAME_0x34,
  PID_NAME_0x35,
  PID_NAME_0x36,
```

```
PID_NAME_0x37,  
PID_NAME_0x38,  
PID_NAME_0x39,  
PID_NAME_0x3a,  
PID_NAME_0x3b,  
PID_NAME_0x3c,  
PID_NAME_0x3d,  
PID_NAME_0x3e,  
PID_NAME_0x3f,  
PID_NAME_0x40,  
PID_NAME_0x41,  
PID_NAME_0x42,  
PID_NAME_0x43,  
PID_NAME_0x44,  
PID_NAME_0x45,  
PID_NAME_0x46,  
PID_NAME_0x47,  
PID_NAME_0x48,  
PID_NAME_0x49,  
PID_NAME_0x4a,  
PID_NAME_0x4b,  
PID_NAME_0x4c,  
PID_NAME_0x4d,  
PID_NAME_0x4e,  
PID_NAME_0x4f,  
PID_NAME_0x50,  
PID_NAME_0x51,  
PID_NAME_0x52,  
PID_NAME_0x53,  
PID_NAME_0x54,  
PID_NAME_0x55,  
PID_NAME_0x56,  
PID_NAME_0x57,  
PID_NAME_0x58,  
PID_NAME_0x59,  
PID_NAME_0x5a,  
PID_NAME_0x5b,  
PID_NAME_0x5c,  
PID_NAME_0x5d,  
PID_NAME_0x5e,  
PID_NAME_0x5f,  
};
```

```
const char PERCENTAGE[] PROGMEM = "%";  
const char KPA[] PROGMEM = "kPa";  
const char PA[] PROGMEM = "Pa";  
const char RPM[] PROGMEM = "rpm";  
const char KPH[] PROGMEM = "km/h";  
const char DEGREES_BEFORE_TDC[] PROGMEM = "° before TDC";  
const char GRAMS_PER_SECOND[] PROGMEM = "grams/sec";  
const char SECONDS[] PROGMEM = "seconds";  
const char RATIO[] PROGMEM = "ratio";  
const char COUNT[] PROGMEM = "count";  
const char KM[] PROGMEM = "km";  
const char VOLTS[] PROGMEM = "V";  
const char MINUTES[] PROGMEM = "minutes";  
const char GPS[] PROGMEM = "g/s";
```

```

const char DEGREES[] PROGMEM = "°";
const char DEGREES_CELCIUS[] PROGMEM = "°C";
const char LPH[] PROGMEM = "L/h";

const char* const PID_UNIT_MAPPER[] PROGMEM = {
  NULL,
  NULL,
  NULL,
  NULL,
  PERCENTAGE,
  DEGREES_CELCIUS,
  PERCENTAGE,
  PERCENTAGE,
  PERCENTAGE,
  PERCENTAGE,
  KPA,
  KPA,
  RPM,
  KPH,
  DEGREES_BEFORE_TDC,
  DEGREES_CELCIUS,
  GRAMS_PER_SECOND,
  PERCENTAGE,
  NULL,
  NULL,
  PERCENTAGE,
  PERCENTAGE,
  PERCENTAGE,
  PERCENTAGE,
  PERCENTAGE,
  PERCENTAGE,
  PERCENTAGE,
  PERCENTAGE,
  NULL,
  NULL,
  NULL,
  SECONDS,
  NULL,
  KM,
  KPA,
  KPA,
  RATIO,
  RATIO,
  RATIO,
  RATIO,
  RATIO,
  RATIO,
  RATIO,
  RATIO,
  PERCENTAGE,
  PERCENTAGE,
  PERCENTAGE,
  PERCENTAGE,
  COUNT,
  KM,
  PA,
  KPA,

```

```

RATIO,
RATIO,
RATIO,
RATIO,
RATIO,
RATIO,
RATIO,
RATIO,
DEGREES_CELCIUS,
DEGREES_CELCIUS,
DEGREES_CELCIUS,
DEGREES_CELCIUS,
NULL,
NULL,
VOLTS,
PERCENTAGE,
RATIO,
PERCENTAGE,
DEGREES_CELCIUS,
PERCENTAGE,
PERCENTAGE,
PERCENTAGE,
PERCENTAGE,
PERCENTAGE,
PERCENTAGE,
MINUTES,
MINUTES,
NULL,
GPS,
NULL,
PERCENTAGE,
KPA,
PA,
PERCENTAGE,
PERCENTAGE,
PERCENTAGE,
PERCENTAGE,
KPA,
PERCENTAGE,
PERCENTAGE,
DEGREES_CELCIUS,
DEGREES,
LPH,
NULL,
};

OBDClass::OBDClass() :
    _responseTimeout(OBD_DEFAULT_TIMEOUT),
    _lastPidResponseMillis(0)
{
    memset(_supportedPids, 0x00, sizeof(_supportedPids));
}

OBDClass::~OBDClass()
{
}

```

```

int OBDClass::begin()
{
    if (!BCAN.begin(500E3)) {
        return 0;
    }

    memset(_supportedPids, 0x00, sizeof(_supportedPids));

    // first try standard addressing
    _useExtendedAddressing = false;
    BCAN.filter(0x7e8);
    if (!supportedPidsRead()) {
        // next try extended addressing
        _useExtendedAddressing = true;
        BCAN.filterExtended(0x18daf110);

        if (!supportedPidsRead()) {
            return 0;
        }
    }

    return 1;
}

void OBDClass::end()
{
    BCAN.end();
}

bool OBDClass::pidSupported(uint8_t pid)
{
    if (pid == 0) {
        return true;
    }

    return (_supportedPids[(pid - 1) / 0x20] & ((1UL << (31 - ((pid - 1) % 32)))) ? true : false;
}

bool OBDClass::pidValueRaw(uint8_t pid)
{
    switch(pid) {
        case PIDS_SUPPORT_01_20: // raw
        case MONITOR_STATUS_SINCE_DTCS_CLEARED: // raw
        case FREEZE_DTC: // raw
        case PIDS_SUPPORT_21_40: // raw
        case PIDS_SUPPORT_41_60: // raw
        case MONITOR_STATUS_THIS_DRIVE_CYCLE: // raw
        case FUEL_SYSTEM_STATUS: // raw
        case COMMANDED_SECONDARY_AIR_STATUS: // raw
        case OBD_STANDARDS_THIS_VEHICLE_CONFORMS_TO: // raw
        case OXYGEN_SENSORS_PRESENT_IN_2_BANKS: // raw
        case OXYGEN_SENSORS_PRESENT_IN_4_BANKS: // raw
        case AUXILIARY_INPUT_STATUS: // raw
        case FUEL_TYPE: // raw
        case EMISSION_REQUIREMENT_TO_WHICH_VEHICLE_IS_DESIGNED: // raw
            return true;
    }
}

```

```

    default:
        return (pid > 0x5f);
    }
}

String OBDClass::pidName(uint8_t pid)
{
    if (pid > 0x5f) {
        return "Unknown";
    }

#ifdef __AVR__
    const char* pgmName = pgm_read_ptr(&PID_NAME_MAPPER[pid]);
    String name;

    if (pgmName != NULL) {
        while (char c = pgm_read_byte(pgmName++)) {
            name += c;
        }
    }

    return name;
#else
    return PID_NAME_MAPPER[pid];
#endif
}

String OBDClass::pidUnits(uint8_t pid)
{
    if (pid > 0x5f) {
        return "";
    }

#ifdef __AVR__
    const char* pgmUnits = pgm_read_ptr(&PID_UNIT_MAPPER[pid]);
    String units;

    if (pgmUnits != NULL) {
        while (char c = pgm_read_byte(pgmUnits++)) {
            units += c;
        }
    }

    return units;
#else
    return PID_UNIT_MAPPER[pid];
#endif
}

float OBDClass::pidRead(uint8_t pid)
{
    if (!pidSupported(pid)) {
        return NAN;
    }

#define A value[0]

```



```

#define B value[1]
#define C value[2]
#define D value[3]
uint8_t value[4];

if (!pidRead(0x01, pid, &value, sizeof(value))) {
    return NAN;
}

switch (pid) {
    default:
    case PIDS_SUPPORT_01_20: // raw
    case MONITOR_STATUS_SINCE_DTCS_CLEARED: // raw
    case FREEZE_DTC: // raw
    case PIDS_SUPPORT_21_40: // raw
    case PIDS_SUPPORT_41_60: // raw
    case MONITOR_STATUS_THIS_DRIVE_CYCLE: // raw
        // NOTE: return value can lose precision!
        return ((uint32_t)A << 24 | (uint32_t)B << 16 | (uint32_t)C << 8 | (uint32_t)D);

    case FUEL_SYSTEM_STATUS: // raw
    case RUN_TIME_SINCE_ENGINE_START:
    case DISTANCE_TRAVELED_WITH_MIL_ON:
    case DISTANCE_TRAVELED_SINCE_CODES_CLEARED:
    case TIME_RUN_WITH_MIL_ON:
    case TIME_SINCE_TROUBLE_CODES_CLEARED:
        return (A * 256.0 + B);

    case CALCULATED_ENGINE_LOAD:
    case THROTTLE_POSITION:
    case COMMANDED_EGR:
    case COMMANDED_EVAPORATIVE_PURGE:
    case FUEL_TANK_LEVEL_INPUT:
    case RELATIVE_THROTTLE_POSITION:
    case ABSOLUTE_THROTTLE_POSITION_B:
    case ABSOLUTE_THROTTLE_POSITION_C:
    case ABSOLUTE_THROTTLE_POSITION_D:
    case ABSOLUTE_THROTTLE_POSITION_E:
    case ABSOLUTE_THROTTLE_POSITION_F:
    case COMMANDED_THROTTLE_ACTUATOR:
    case ETHANOL_FUEL_PERCENTAGE:
    case RELATIVE_ACCELERATOR_PEDAL_POSITION:
    case HYBRID_BATTERY_PACK_REMAINING_LIFE:
        return (A / 2.55);

    case COMMANDED_SECONDARY_AIR_STATUS: // raw
    case OBD_STANDARDS_THIS_VEHICLE_CONFORMS_TO: // raw
    case OXYGEN_SENSORS_PRESENT_IN_2_BANKS: // raw
    case OXYGEN_SENSORS_PRESENT_IN_4_BANKS: // raw
    case AUXILIARY_INPUT_STATUS: // raw
    case FUEL_TYPE: // raw
    case EMISSION_REQUIREMENT_TO_WHICH_VEHICLE_IS_DESIGNED: // raw
        return (A);

    case OXYGEN_SENSOR_1_SHORT_TERM_FUEL_TRIM:
    case OXYGEN_SENSOR_2_SHORT_TERM_FUEL_TRIM:
    case OXYGEN_SENSOR_3_SHORT_TERM_FUEL_TRIM:

```

```

case OXYGEN_SENSOR_4_SHORT_TERM_FUEL_TRIM:
case OXYGEN_SENSOR_5_SHORT_TERM_FUEL_TRIM:
case OXYGEN_SENSOR_6_SHORT_TERM_FUEL_TRIM:
case OXYGEN_SENSOR_7_SHORT_TERM_FUEL_TRIM:
case OXYGEN_SENSOR_8_SHORT_TERM_FUEL_TRIM:
    return ((B / 1.28) - 100.0);
    break;

case ENGINE_COOLANT_TEMPERATURE:
case AIR_INTAKE_TEMPERATURE:
case AMBIENT_AIR_TEMPERATURE:
case ENGINE_OIL_TEMPERATURE:
    return (A - 40.0);

case SHORT_TERM_FUEL_TRIM_BANK_1:
case LONG_TERM_FUEL_TRIM_BANK_1:
case SHORT_TERM_FUEL_TRIM_BANK_2:
case LONG_TERM_FUEL_TRIM_BANK_2:
case EGR_ERROR:
    return ((A / 1.28) - 100.0);

case FUEL_PRESSURE:
    return (A * 3.0);

case INTAKE_MANIFOLD_ABSOLUTE_PRESSURE:
case VEHICLE_SPEED:
case WARM_UPS_SINCE_CODES_CLEARED:
case ABSOLULTE_BAROMETRIC_PRESSURE:
    return (A);

case ENGINE_RPM:
    return ((A * 256.0 + B) / 4.0);

case TIMING_ADVANCE:
    return ((A / 2.0) - 64.0);

case MAF_AIR_FLOW_RATE:
    return ((A * 256.0 + B) / 100.0);

case FUEL_RAIL_PRESSURE:
    return ((A * 256.0 + B) * 0.079);

case FUEL_RAIL_GAUGE_PRESSURE:
case FUEL_RAIL_ABSOLUTE_PRESSURE:
    return ((A * 256.0 + B) * 10.0);

case OXYGEN_SENSOR_1_FUEL_AIR_EQUIVALENCE_RATIO:
case OXYGEN_SENSOR_2_FUEL_AIR_EQUIVALENCE_RATIO:
case OXYGEN_SENSOR_3_FUEL_AIR_EQUIVALENCE_RATIO:
case OXYGEN_SENSOR_4_FUEL_AIR_EQUIVALENCE_RATIO:
case OXYGEN_SENSOR_5_FUEL_AIR_EQUIVALENCE_RATIO:
case OXYGEN_SENSOR_6_FUEL_AIR_EQUIVALENCE_RATIO:
case OXYGEN_SENSOR_7_FUEL_AIR_EQUIVALENCE_RATIO:
case OXYGEN_SENSOR_8_FUEL_AIR_EQUIVALENCE_RATIO:
case 0x34:
case 0x35:
case 0x36:

```

```

case 0x37:
case 0x38:
case 0x39:
case 0x3a:
case 0x3b:
    return (((A * 256.0 + B) * 2.0) / 65536.0);

case EVAP_SYSTEM_VAPOR_PRESSURE:
    return (((int16_t)(A * 256.0 + B)) / 4.0);

case CATALYST_TEMPERATURE_BANK_1_SENSOR_1:
case CATALYST_TEMPERATURE_BANK_2_SENSOR_1:
case CATALYST_TEMPERATURE_BANK_1_SENSOR_2:
case CATALYST_TEMPERATURE_BANK_2_SENSOR_2:
    return (((A * 256.0 + B) / 10.0) - 40.0);

case CONTROL_MODULE_VOLTAGE:
    return ((A * 256.0 + B) / 1000.0);

case ABSOLUTE_LOAD_VALUE:
    return ((A * 256.0 + B) / 2.55);

case FUEL_AIR_COMMANDED_EQUIVALENCE_RATE:
    return (2.0 * (A * 256.0 + B) / 65536.0);

case ABSOLUTE_EVAP_SYSTEM_VAPOR_PRESSURE:
    return ((A * 256.0 + B) / 200.0);

case 0x54:
    return ((A * 256.0 + B) - 32767.0);

case FUEL_INJECTION_TIMING:
    return (((A * 256.0 + B) / 128.0) - 210.0);

case ENGINE_FUEL_RATE:
    return ((A * 256.0 + B) / 20.0);
}
}

String OBDClass::vinRead()
{
    char vin[18];

    memset(vin, 0x00, sizeof(vin));

    if (!pidRead(0x09, 0x02, vin, 17)) {
        // failed
        return "";
    }

    return vin;
}

uint32_t OBDClass::pidReadRaw(uint8_t pid)
{
    if (!pidSupported(pid)) {
        return 0;
    }
}

```

```

}

#define A value[0]
#define B value[1]
#define C value[2]
#define D value[3]
uint8_t value[4];

if (!pidRead(0x01, pid, &value, sizeof(value))) {
    return 0;
}

switch (pid) {
    case COMMANDED_SECONDARY_AIR_STATUS:
    case OBD_STANDARDS_THIS_VEHICLE_CONFORMS_TO:
    case OXYGEN_SENSORS_PRESENT_IN_2_BANKS:
    case OXYGEN_SENSORS_PRESENT_IN_4_BANKS:
    case AUXILIARY_INPUT_STATUS:
    case FUEL_TYPE:
    case EMISSION_REQUIREMENT_TO_WHICH_VEHICLE_IS_DESIGNED:
        return (A);

    case FUEL_SYSTEM_STATUS:
        return ((uint32_t)A << 8 | (uint32_t)B);

    default:
        return ((uint32_t)A << 24 | (uint32_t)B << 16 | (uint32_t)C << 8 | (uint32_t)D);
}

String OBDClass::ecuNameRead()
{
    char ecuName[21];

    memset(ecuName, 0x00, sizeof(ecuName));

    if (!pidRead(0x09, 0x0a, ecuName, 20)) {
        // failed
        return "";
    }

    return ecuName;
}

void OBDClass::setTimeout(unsigned long timeout)
{
    _responseTimeout = timeout;
}

int OBDClass::supportedPidsRead()
{
    for (int pid = 0x00; pid < 0xe0; pid += 0x20) {
        uint8_t value[4];

        if (pidRead(0x01, pid, value, sizeof(value)) != 4) {
            return 0;
        }
    }
}

```

```

#define A value[0]
#define B value[1]
#define C value[2]
#define D value[3]
_supportedPids[pid / 0x20] = ((uint32_t)A << 24 | (uint32_t)B << 16 | (uint32_t)C << 8 | (uint32_t)D << 0);

if ((_supportedPids[pid / 0x20] & 0x00000001) == 0x00000000) {
    // no more
    break;
}
}

return 1;
}

int OBDClass::clearAllStoredDTC()
{
    //Function clears stored Diagnostic Trouble Codes (DTC)

    // make sure at least 60 ms have passed since the last response
    unsigned long lastResponseDelta = millis() - _lastPidResponseMillis;
    if (lastResponseDelta < 60) {
        delay(60 - lastResponseDelta);
    }

    for (int retries = 10; retries > 0; retries--) {
        if (_useExtendedAddressing) {
            BCAN.beginExtendedPacket(0x18db33f1, 8);
        } else {
            BCAN.beginPacket(0x7df, 8);
        }
        BCAN.write(0x00); // number of additional bytes
        BCAN.write(0x04); // Mode / Service 4, for clearing DTC
        if (BCAN.endPacket()) {
            // send success
            break;
        } else if (retries <= 1) {
            return 0;
        }
    }

    return 1;
}

int OBDClass::pidRead(uint8_t mode, uint8_t pid, void* data, int length)
{
    // make sure at least 60 ms have passed since the last response
    unsigned long lastResponseDelta = millis() - _lastPidResponseMillis;
    if (lastResponseDelta < 60) {
        delay(60 - lastResponseDelta);
    }

    for (int retries = 10; retries > 0; retries--) {
        if (_useExtendedAddressing) {
            BCAN.beginExtendedPacket(0x18db33f1, 8);
        } else {

```

```

    BCAN.beginPacket(0x7df, 8);
}
BCAN.write(0x02); // number of additional bytes
BCAN.write(mode);
BCAN.write(pid);
if (BCAN.endPacket()) {
    // send success
    break;
} else if (retries <= 1) {
    return 0;
}
}
}
bool splitResponse = (length > 5);
for (unsigned long start = millis(); (millis() - start) < _responseTimeout;) {
    if (BCAN.parsePacket() != 0 &&
        (splitResponse ? (BCAN.read() == 0x10 && BCAN.read()) : BCAN.read()) &&
        (BCAN.read() == (mode | 0x40) && BCAN.read() == pid)) {
        _lastPidResponseMillis = millis();
        // got a response
        if (!splitResponse) {
            return BCAN.readBytes((uint8_t*)data, length);
        }
        int read = BCAN.readBytes((uint8_t*)data, 3);
        for (int i = 0; read < length; i++) {
            delay(60);
            // send the request for the next chunk
            if (_useExtendedAddressing) {
                BCAN.beginExtendedPacket(0x18db33f1, 8);
            } else {
                BCAN.beginPacket(0x7df, 8);
            }
            BCAN.write(0x30);
            BCAN.endPacket();

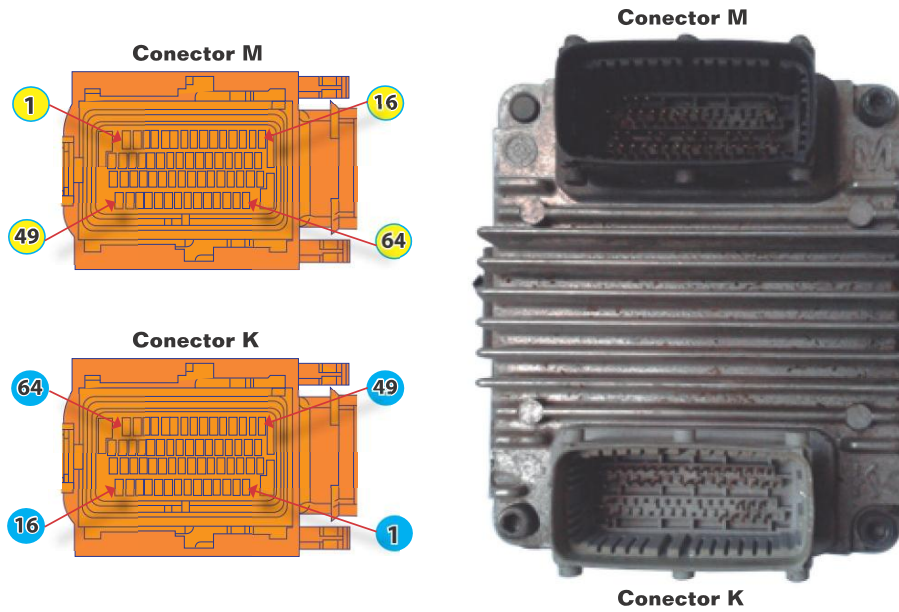
            // wait for response
            while (BCAN.parsePacket() == 0 ||
                BCAN.read() != (0x21 + i)); // correct sequence number

            while (BCAN.available()) {
                ((uint8_t*)data)[read++] = BCAN.read();
            }
        }
        _lastPidResponseMillis = millis();
        return read;
    }
}

return 0;
}
}
OBDClass OBD;
Datasheet ECM AVEO 2008

```

✓ **ECM (Modulo de Control Electrónico) de 128 terminales utilizada en vehículos de la línea de Pontiac (Chevrolet) en vehículos AVEO y PONTIAC G3.**



**Tabla de energías y tierras del ECM**

Terminal	Conector	Código de color	Función Terminal/circuito.
1	K	Naranja	Alimentación de voltaje de batería.
17	K	Naranja	Alimentación de voltaje de batería.
18	K	Rosa	Alimentación de voltaje de ignición.
37	M	Negro/blanco	Alimentación de tierra.
38	M	Negro/blanco	Alimentación de tierra.
39	M	Negro/blanco	Alimentación de tierra.
40	M	Negro/blanco	Alimentación de tierra.
41	M	Negro/blanco	Alimentación de tierra.

**Nota.** Datos extraídos en un vehículo PONTIAC G3 2008 1.6 Lts

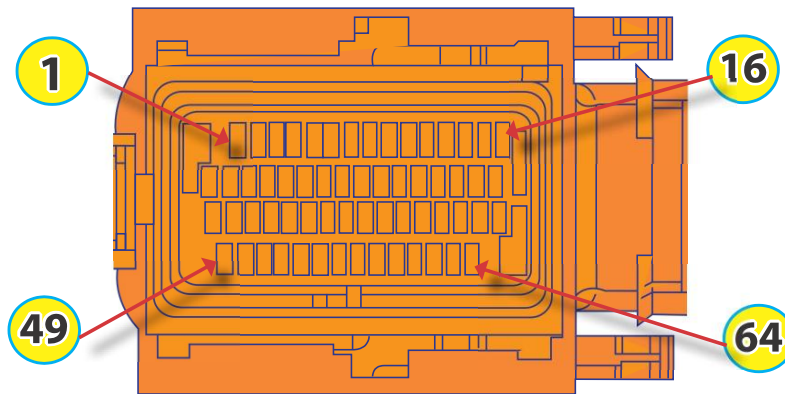
	<p><b>1</b> Sensor de posición de la mariposa de aceleración TPS</p> 	<p><b>2</b> Válvula de control de aire en marcha mínima IAC</p> 	<p><b>3</b> Sensor de presión absoluta del múltiple de admisión MAP</p> 	<p><b>4</b> Inyectores de combustible</p> 	
	<p><b>5</b> Sensor de temperatura del refrigerante ECT</p> 	<p><b>6</b> Sensor de temperatura del aire ACT.</p> 	<p><b>7</b> Válvula recirculadora de los gases de escape EGR</p> 		
	<p><b>8</b> Caja de relevadores compartimiento del motor</p> 	<p><b>9</b> Modulo de Control Electrónico ECM</p> 	<p><b>10</b> Bobinas de encendido</p> 		
	<p><b>11</b> Sensor de posición del árbol de levas CMP</p> 	<p><b>12</b> Sensor de posición del Angulo de giro del cigüeñal CKP</p> 	<p><b>13</b> Sensor de oxígeno HO2S</p> 		



**CAPÍTULO**  
**3**

**Pin outs**

**Conector M**



Terminal	Código de color	Función del circuito
1	Verde oscuro/blanco	Control de la bobina de encendido para los cilindros 1 y 4
2	Café	Control de calefactor del sensor de oxígeno 2.
3	Café	tensión de alimentación de la bomba de combustible
4	Blanco	Control del solenoide de la purga del canister (EVAP).
5	Amarillo/negro	Señal del sensor de posición de cigüeñal CKP.
6	Blanco	Señal del sensor de posición de la mariposa de aceleración TPS.
7	Azul/claro	Señal del sensor de temperatura del aire IAT.
8	Azul/claro	Señal del sensor de presión absoluta del múltiple de admisión.

Terminal	Código de color	Función del circuito
9	Verde oscuro	Sensor de posición de la válvula recirculadora de los gases de escape (EGR).
10		
11	Amarillo	Control del inyector 4.
12	Blanco	Tierra hacia el calefactor del sensor de oxígeno 1.
13	Verde claro	Control alto de la válvula de marcha mínima IAC.
14	Verde oscuro	Control bajo de la válvula de marcha mínima IAC.
15	Blanco/azul oscuro	
Blanco	Control bajo de la válvula de marcha mínima IAC.	

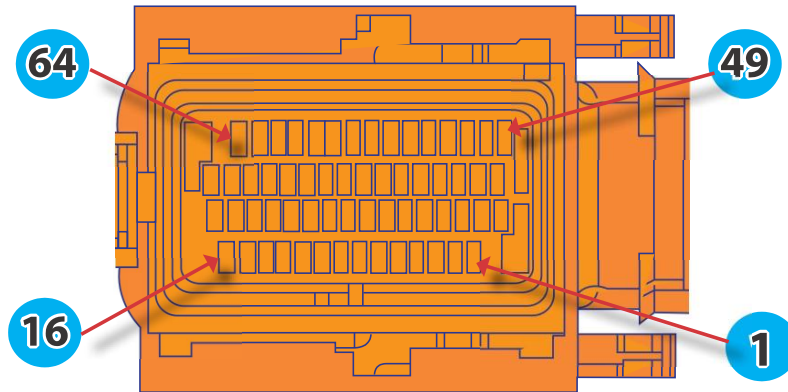
14-6 Aveo y Pontiac 2008 1.6 Lts ECM de 128 terminales

Terminal	Código de color	Función del circuito
16	Azul claro	Alimentación de 5VCD al sensor de presión absoluta (MAP) y válvula recirculadora de los gases de escape (EGR).
17		
18	Amarillo/negro	Señal del sensor de detonación (KS).
19	Azul oscuro	Control del solenoide de la válvula recirculadora de los gases de escape EGR.
20		
21	Azul oscuro/blanco	Señal del sensor de posición del cigüeñal (CKP).
22	Verde claro	Control del inyector 2.
23		
24	Café/blanco	Control del inyector 3.
25	Café/negro	
Azul claro	Control del inyector 1.	
26	Azul claro/negro	
Azul/claro	Señal del sensor de posición del árbol de levas (CKP).	
27		
28	Verde claro	Señal del sensor de temperatura del refrigerante del motor (ECT).
29	Azul oscuro/negro Azul oscuro	Señal del sensor de oxígeno 1.
30	Verde claro	Control alto de la válvula de marcha mínima (IAC).
31		
32	Morado	5vcd al sensor de posición de la mariposa de aceleración (TPS).
33	Naranja	Control de bobina de encendido para los cilindros 1 y 4.
34		

Terminal	Código de color	Función del circuito
35	Azul claro	Control de bobina de encendido 2 y 3.
36		
37	Negro/blanco	Alimentación de tierra a la computadora (ECM).
38	Negro/blanco	Alimentación de tierra a la computadora (ECM).
39	Negro/blanco	Alimentación de tierra a la computadora (ECM).
40	Negro/blanco	Alimentación de tierra a la computadora (ECM).
41	Negro/blanco	Alimentación de tierra a la computadora (ECM).
42		
43		
44		
45		
46		
47		
48	Café/naranja Café	Baja referencia.
49		
50		
51	Azul claro	Control de bobina de encendido cil.2 y 3.
52	Azul oscuro	Válvula de recirculación de gases de escape EGR.
53		
54		
55		
56		
57		
58		
59		
60		
61		
62		
63		
64	Azul oscuro/blanco	Baja referencia.



## Conector K



Terminal	Código de color	Función del circuito
1	Naranja	Alimentación de voltaje de batería a la computadora (ECM).
2		
3		
4		
5	Gris	Tierra del (switch de octanaje)
6	Verde oscuro	Señal del sensor de velocidad del vehículo STD.
7	verde	señal de carretera en mal estado.
8		
9		
11		
12	Gris	Control del relevador del sistema de enfriamiento en alta velocidad.

Terminal	Código de color	Función del circuito
13	Amarillo	Tierra del control de la válvula del distribuidor de admisión.
14	Morado	Serial datos altos CAN.
15	Amarillo	Serial datos bajos CAN.
16	Café	Control del calefactor del sensor de oxígeno 1.
17	Naranja	Alimentación de voltaje de batería a la computadora (ECM).
18	Rosa	Alimentación de voltaje de ignición a la computadora (ECM).
19		
20		
21	Verde oscuro	señal del sensor de carretera (w/o ABS)

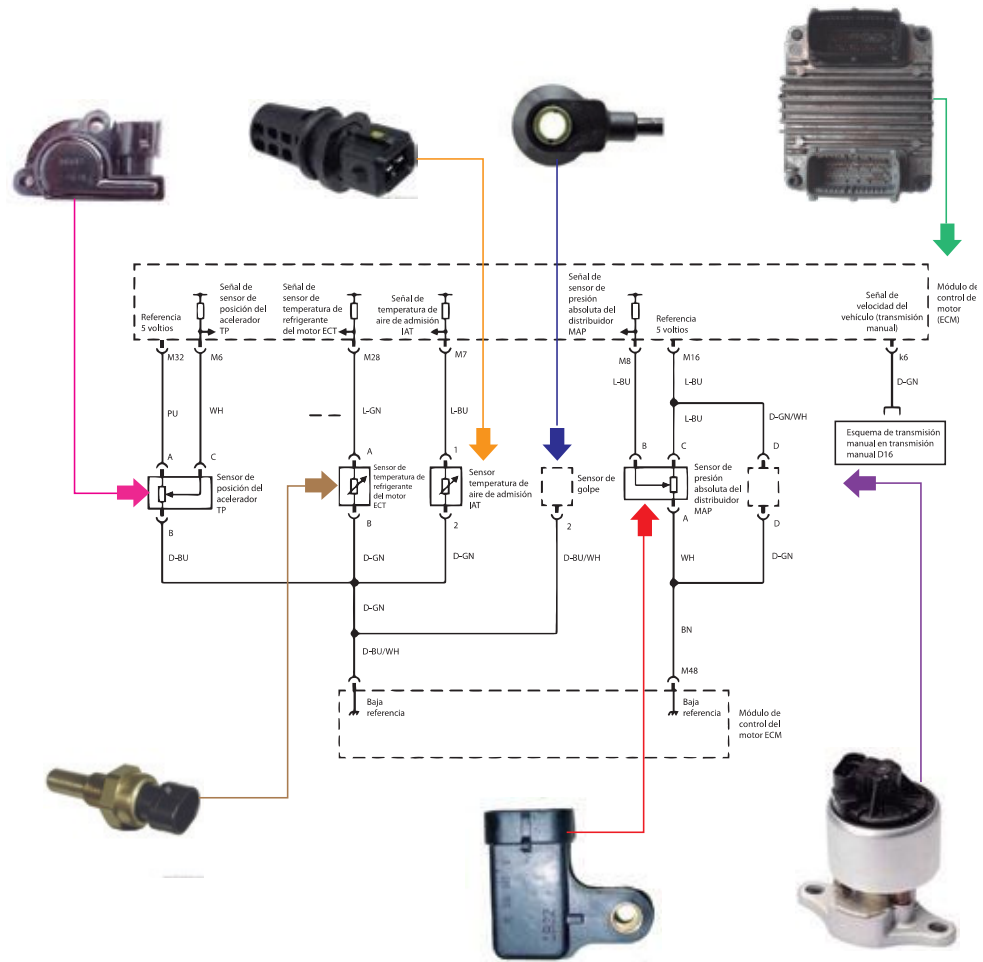
Terminal	Código de color	Función del circuito
22	Azul claro	Señal del sensor (ACP) de la presión del refrigerante del A/C.
23		
24	Blanco	Control del indicador de temperatura del refrigerante del motor.
25	Gris	Señal de velocidad del vehículo.
26		
27		
28	Gris	Control del relevador de moto ventilador en baja velocidad.
29	Verde oscuro	Control del relevador del A/C.
30	Blanco	Señal del sensor de nivel de combustible.
31		
32	Negro/blanco	Serial DATA.
33		
34	Azul claro	Baja referencia sensor de nivel de combustible y presión del refrigerante del A/C.
35	Verde oscuro/blanco	Señal de solicitud del A/C.
36	Verde claro/naranja Verde claro	Señal del sensor de oxígeno 2.
37		
38		
39		
40		
41		
42		
43		
44		
45		
46		
47		
48	Amarillo	Control del solenoide de la válvula de emisiones evaporativas (EVAP)

49		
50	Blanco/rosa	Svcd a los sensores de datos en controles del motor y al sensor de camino irregular sin (ABS).
51	Gris	Señal del sensor del nivel del tanque de combustible.
52	Azul claro	Señal del sensor de presión del tanque.
53	Verde/negro	Baja referencia del sensor de oxígeno 2.
54	Amarillo/azul oscuro	Control del relevador de la bomba de combustible.
55		
56		
57		
58		
59		
60		
61		
62		
63		
64	Café	Control del indicador de luz MIL.

# CAPÍTULO 4

## Diagramas del sistema

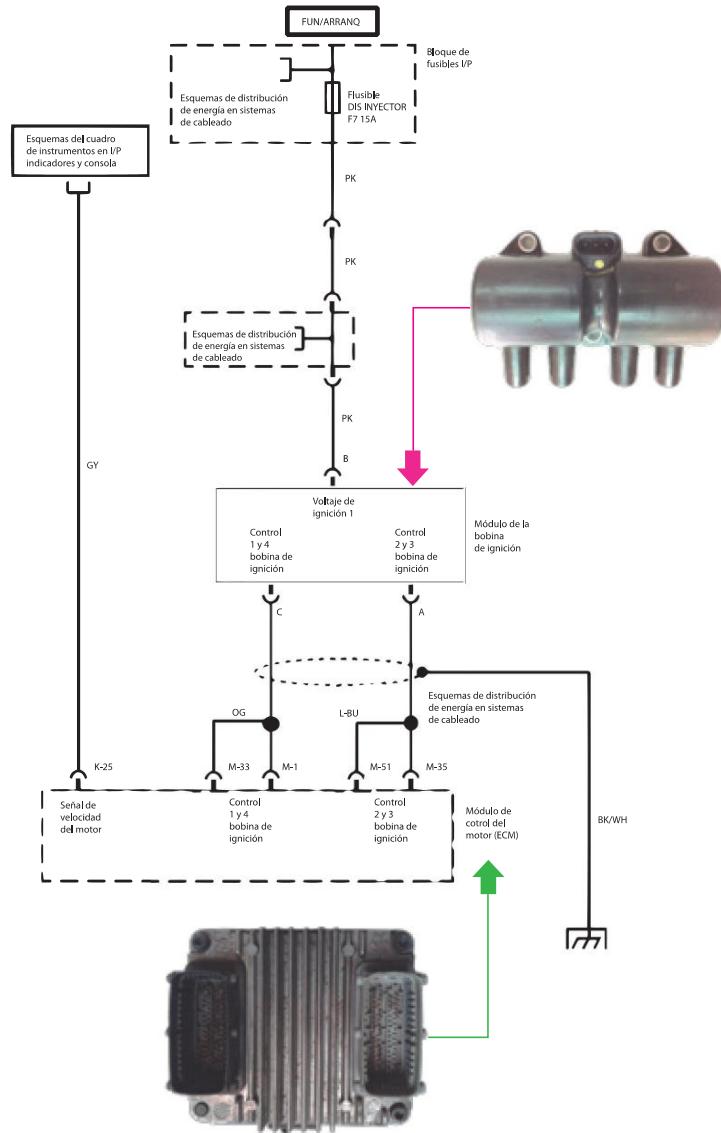
### ✓ Diagrama eléctrico Chevrolet AVEO y PONTIAC G3 2005 -2008



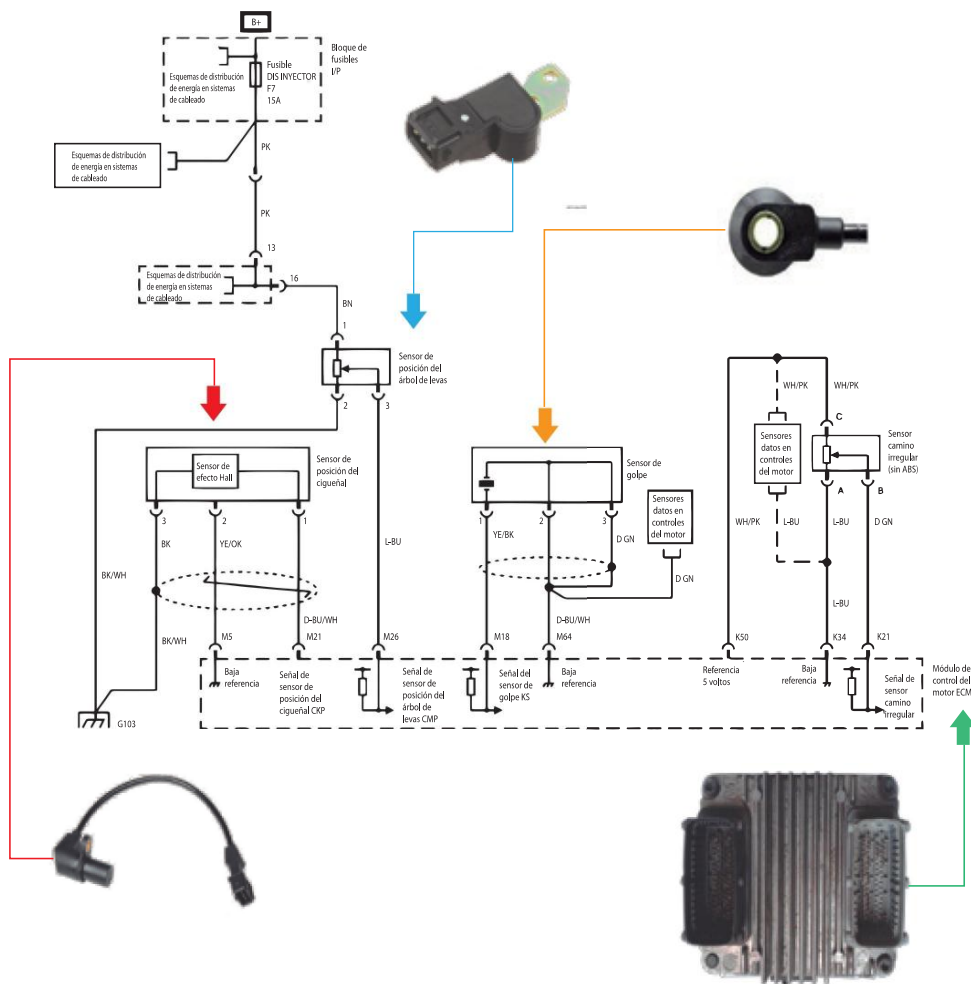
14-10 Aveo y Pontiac 2008 1.6 Lts ECM de 128 terminales



## Diagrama eléctrico Chevrolet AVEO y PONTIAC G3 2005 -2008



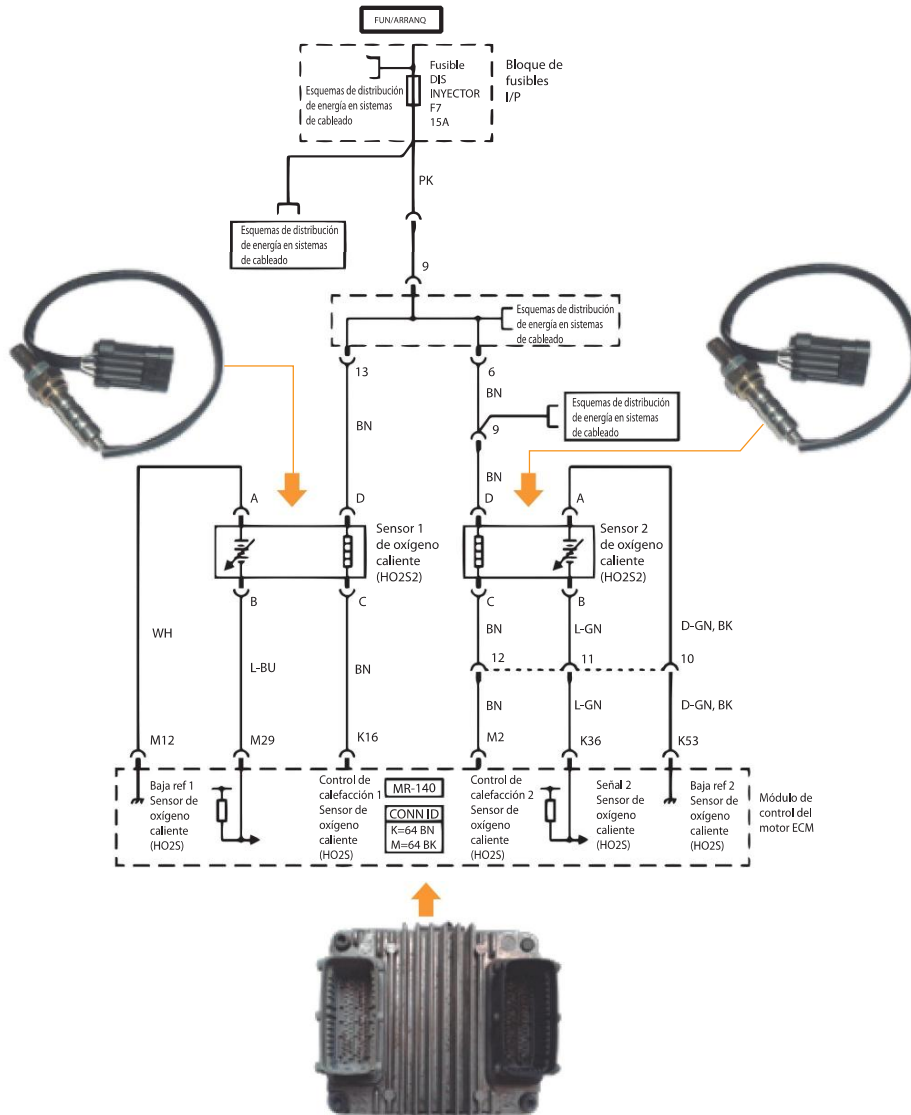
**Diagrama eléctrico Chevrolet AVEO y PONTIAC G3 2005 -2008**



14-12 Aveo y Pontiac 2008 1.6 Lts ECM de 128 terminales



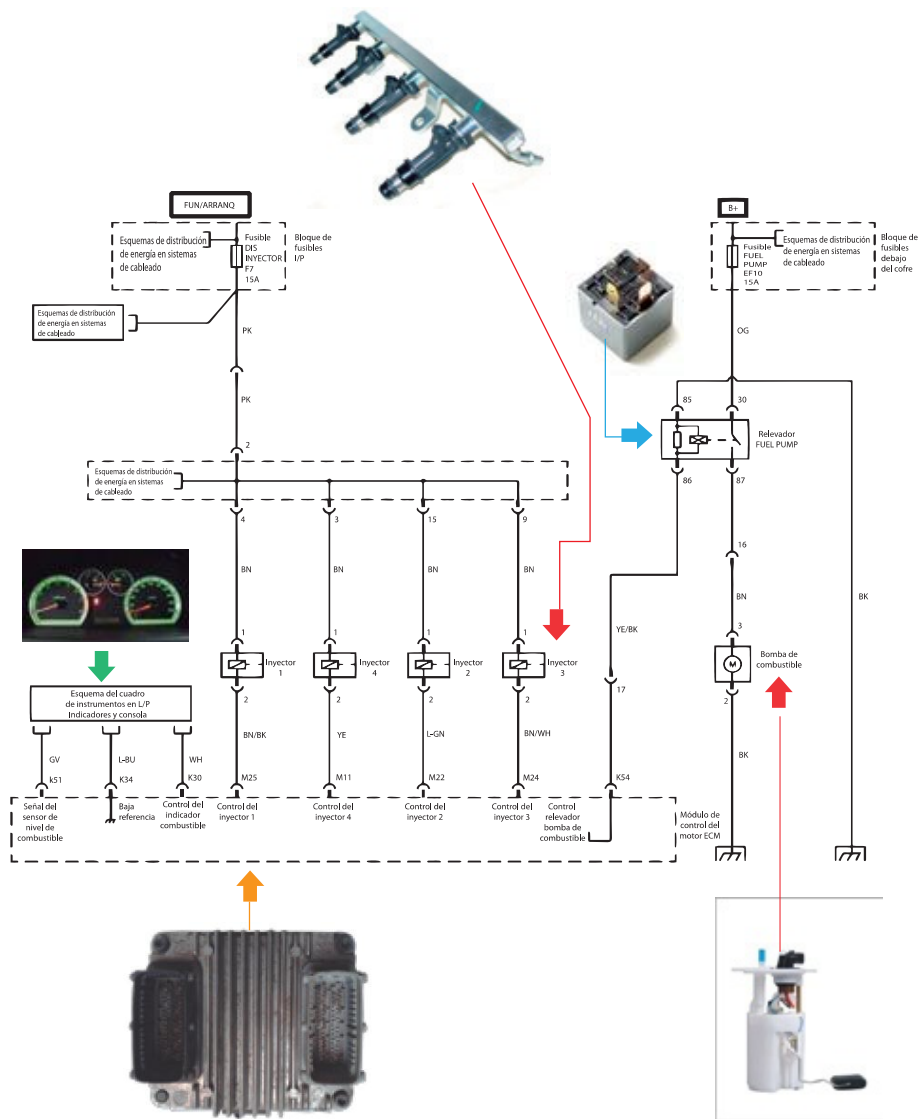
**Diagrama eléctrico Chevrolet AVEO y PONTIAC G3 2005 -2008**



14-14 Aveo y Pontiac 2008 1.6 Lts ECM de 128 terminales

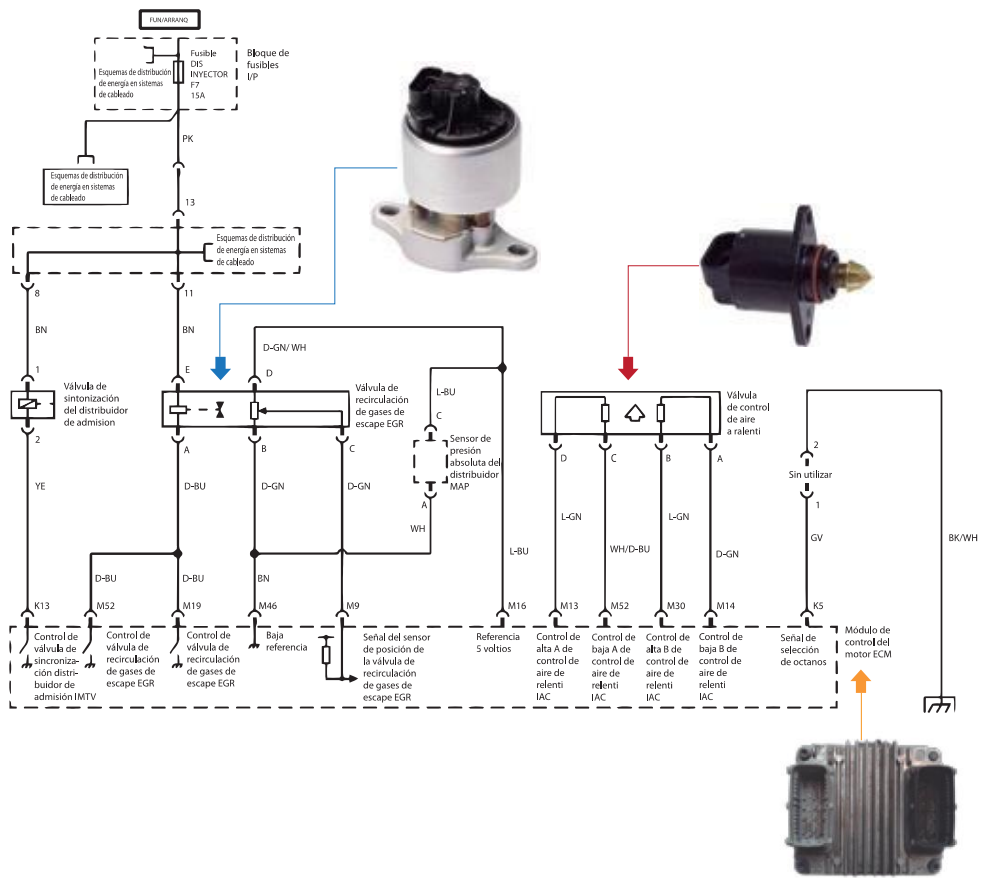


## Diagrama eléctrico Chevrolet AVEO y PONTIAC G3 2005 -2008

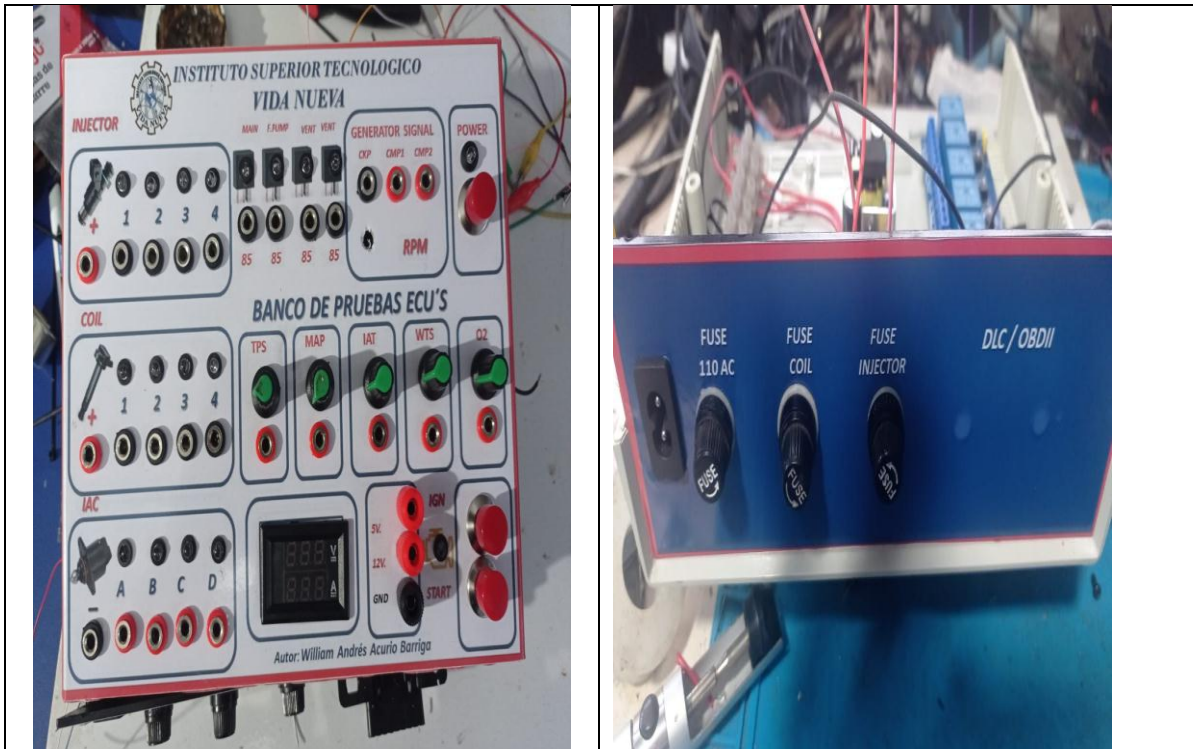


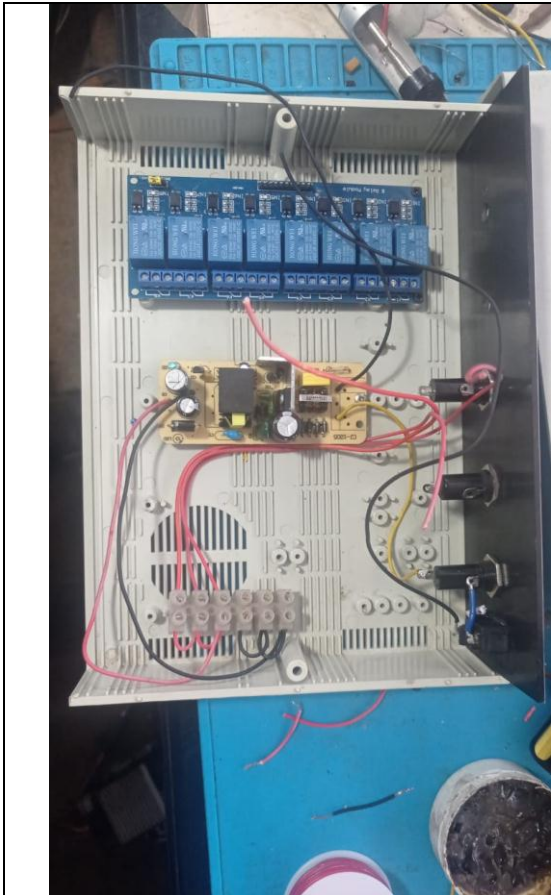


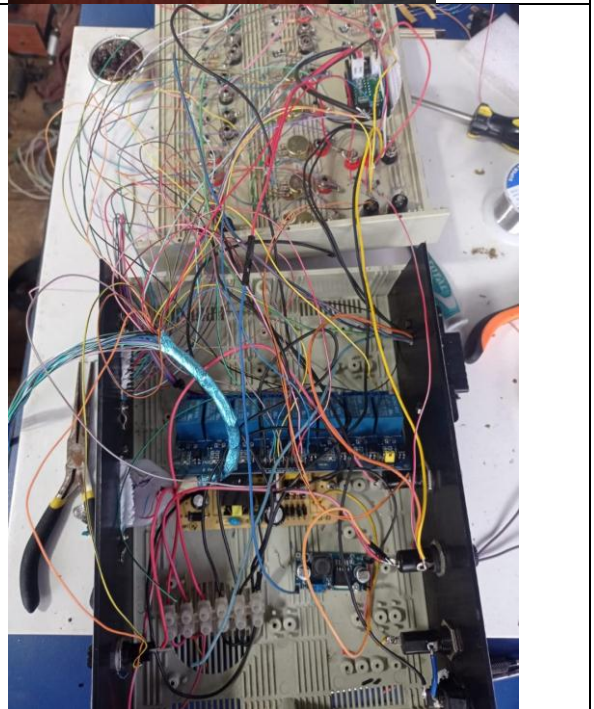
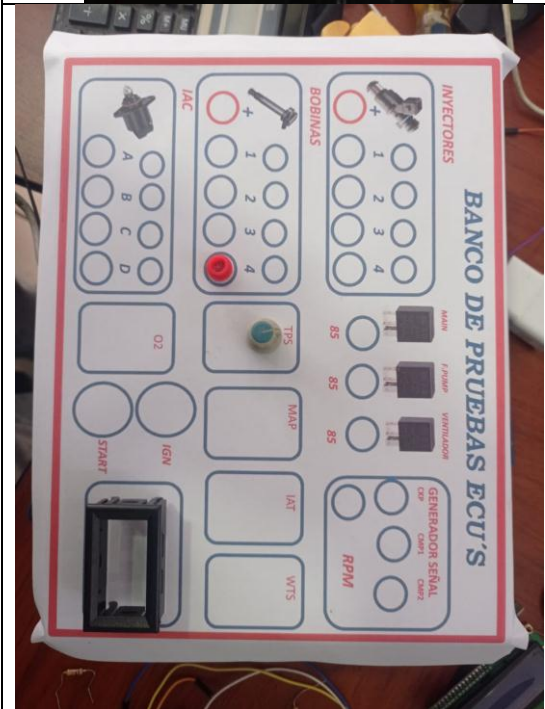
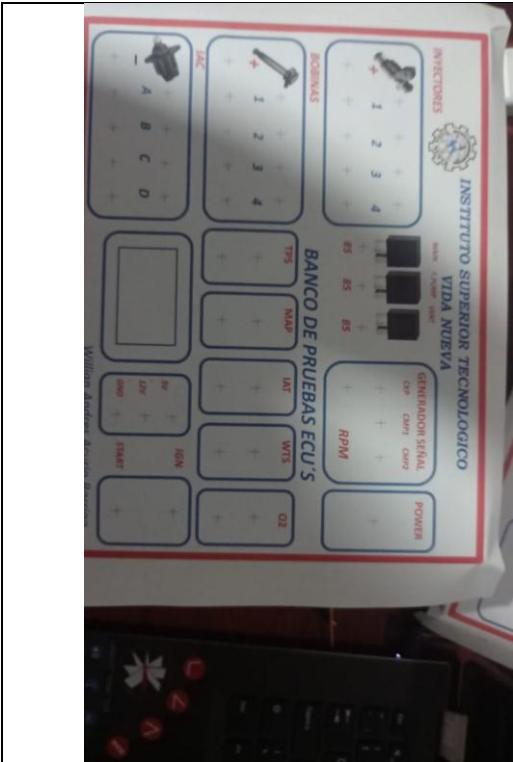
## Diagrama eléctrico Chevrolet AVEO y PONTIAC G3 2005 -2008

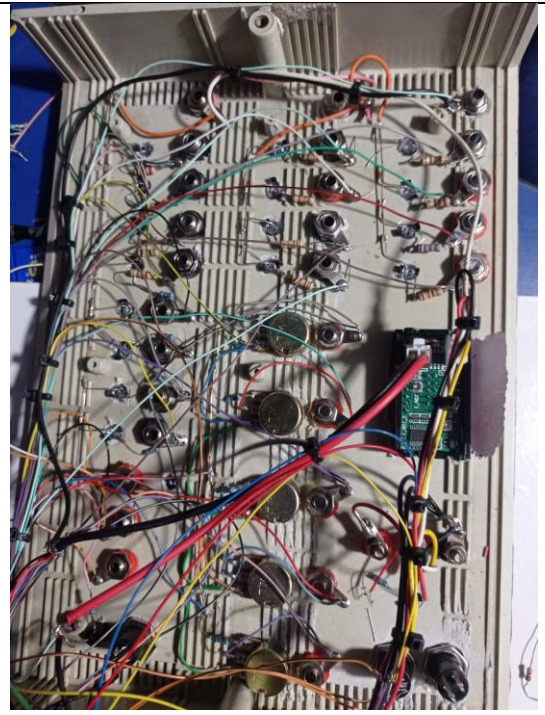
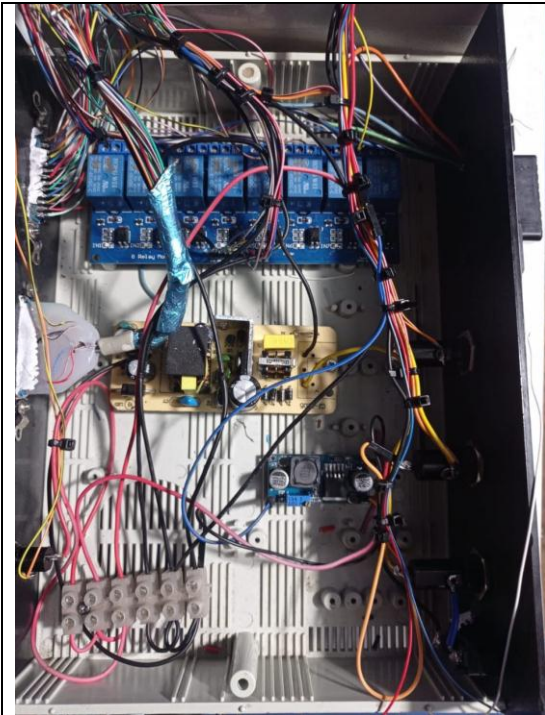


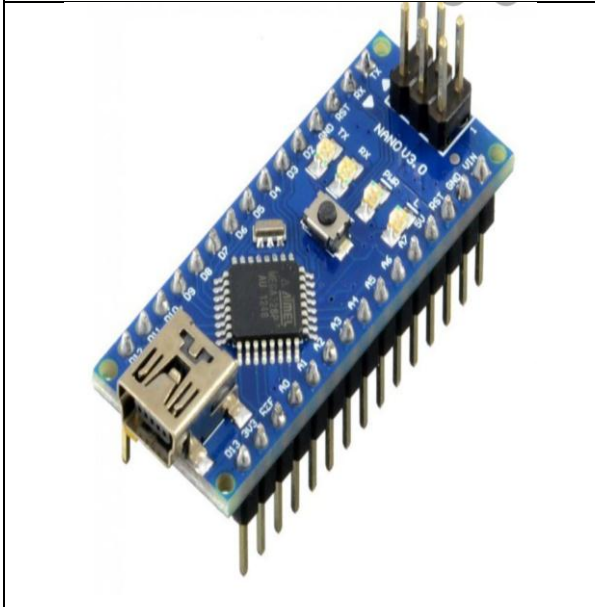
## Figuras de diseño, ensamblaje y pruebas



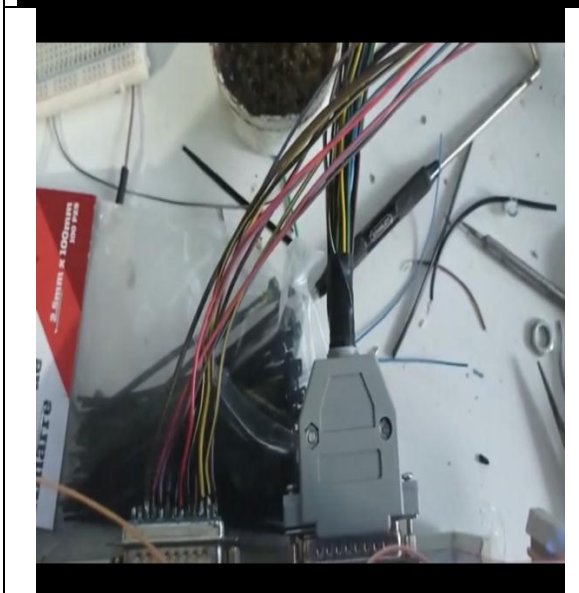
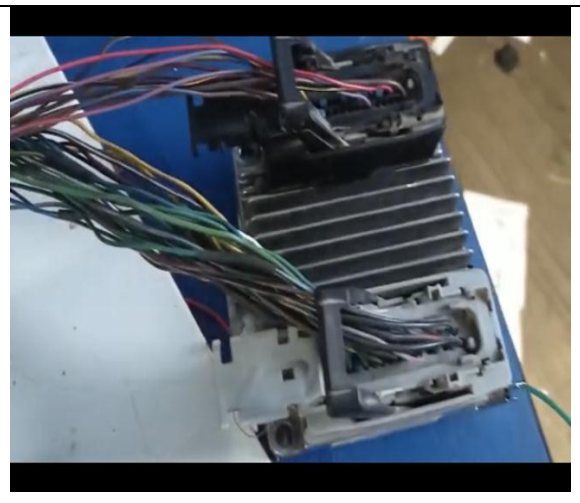
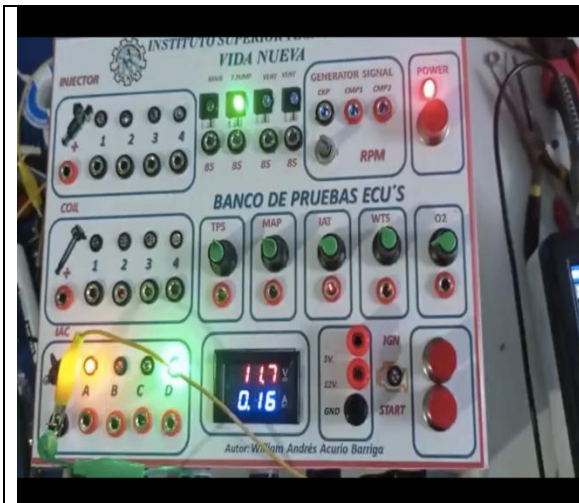




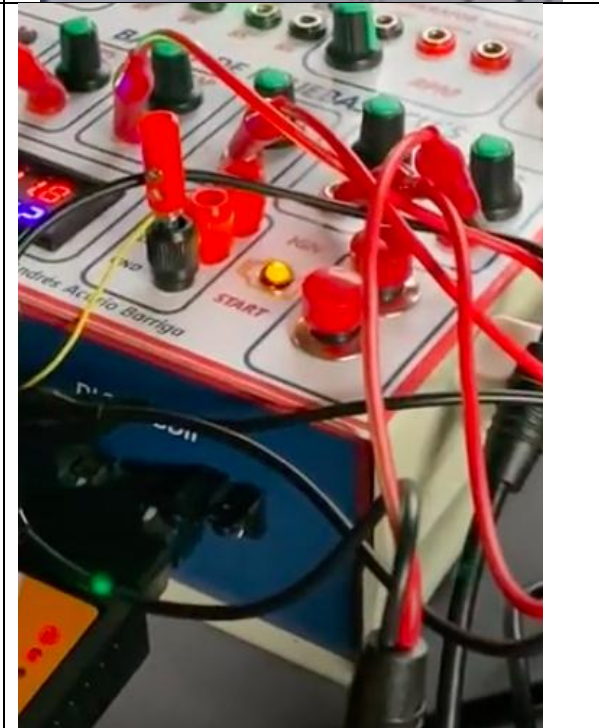
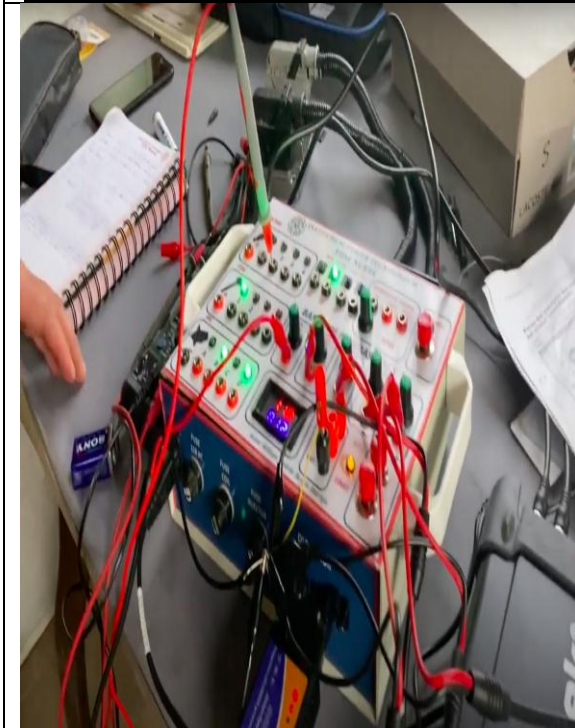
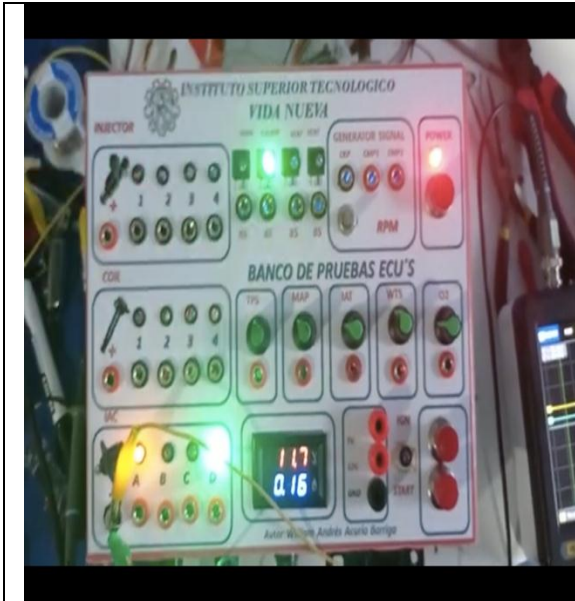


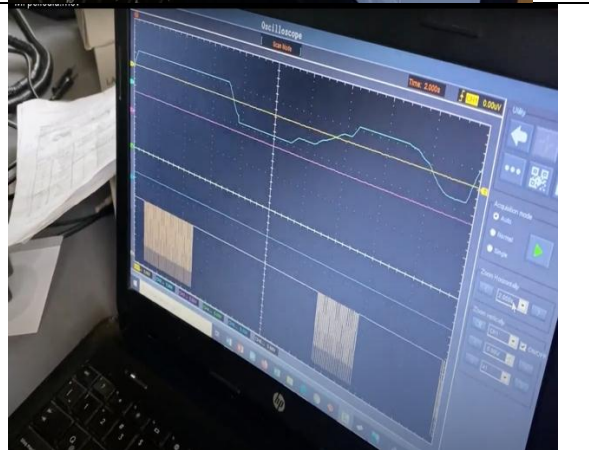
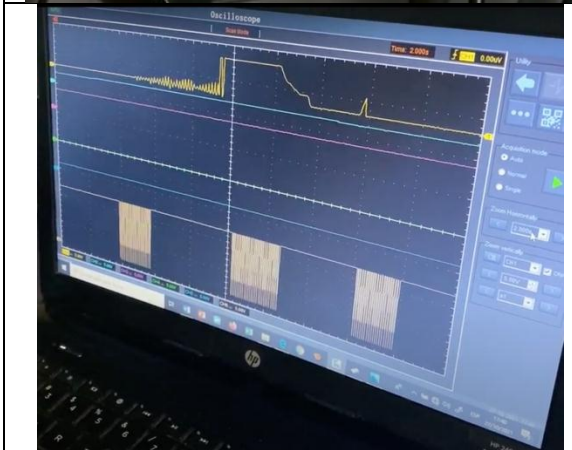
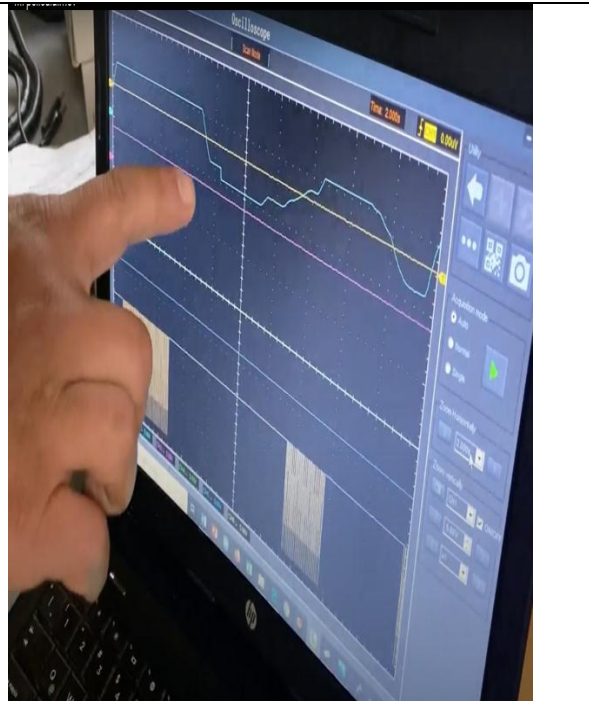


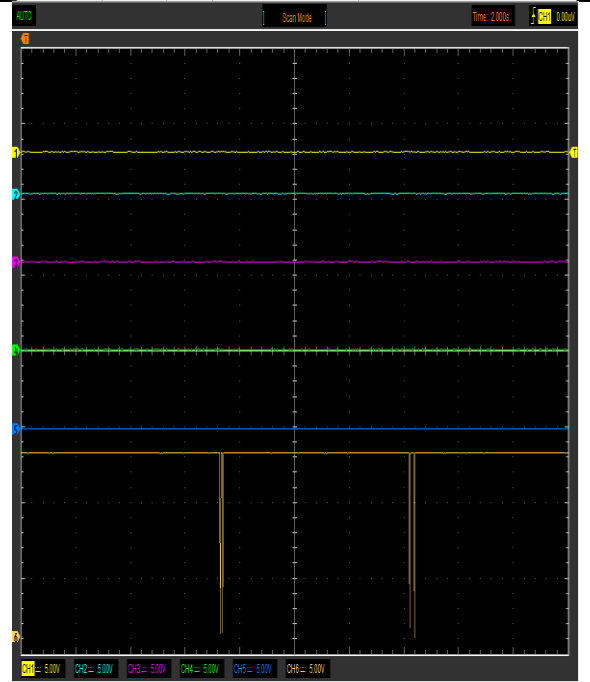
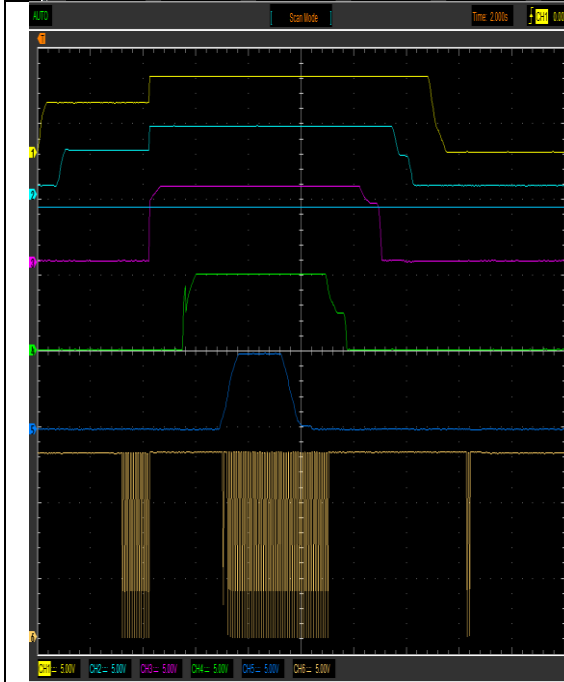
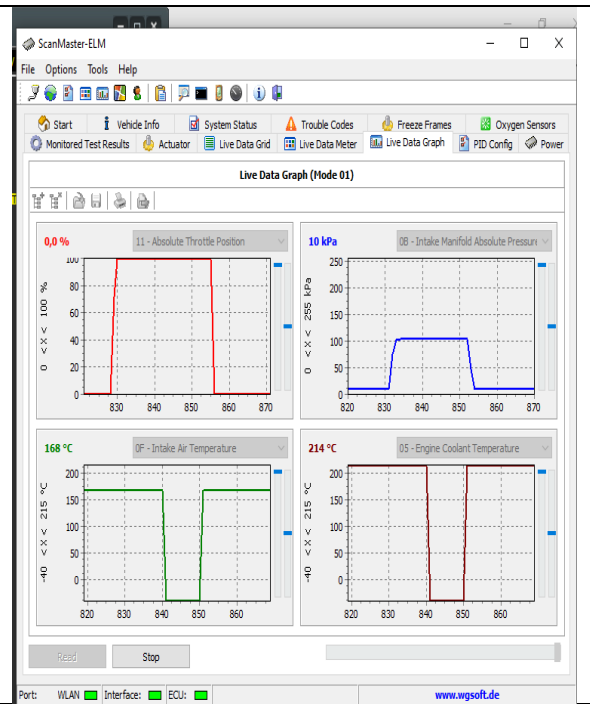
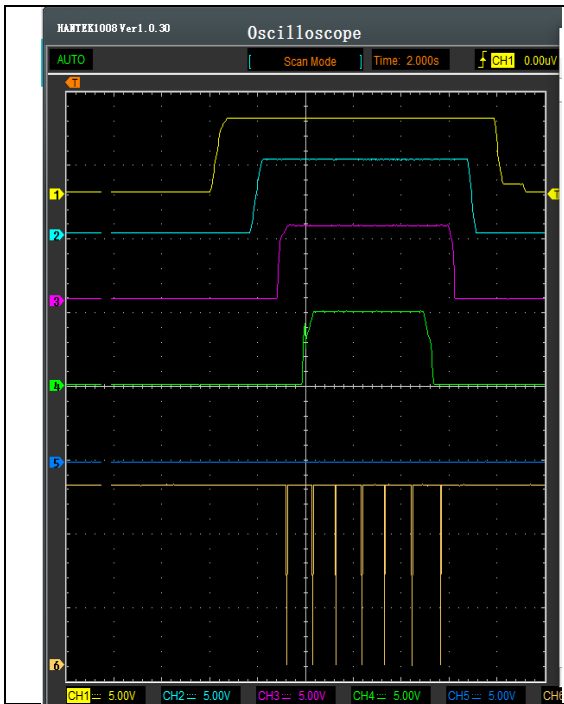


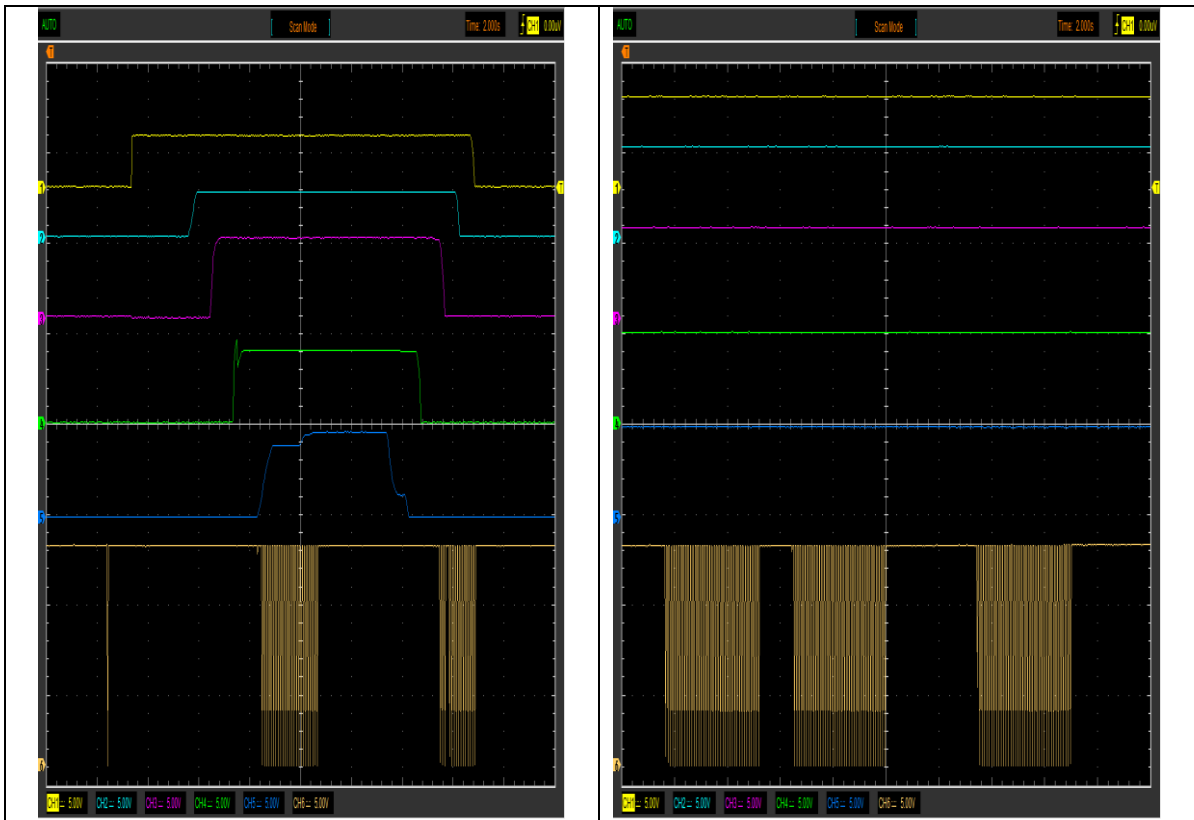


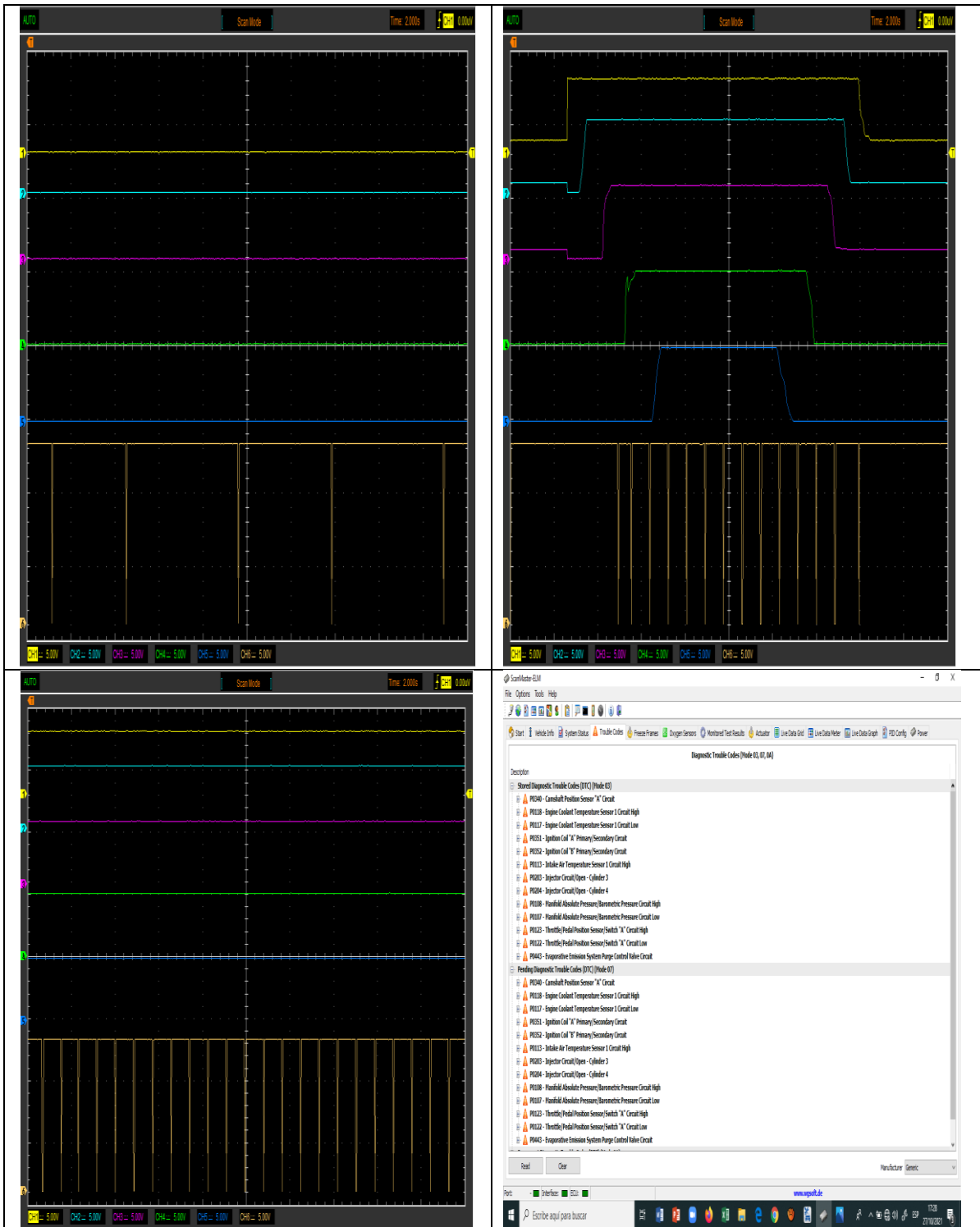












## Presupuesto

<b>ITEM</b>	<b>RUBRO</b>	<b>CANTIDAD</b>	<b>V/U</b>	<b>V/T</b>
1	Banco de pruebas inyectores gasolina (medio uso)	1	\$230	\$230
2	ECU	1	\$200	\$200
3	Kit electrónica y electricidad	1	\$ 75	\$ 75
4	Capacitación electrónica avanzada	1	\$ 100	\$ 100
12	10% de imprevistos	1	\$120	\$120

**INVERSIÓN \$725**

**TOTAL      \$725**

**Elaborado por:** William Andrés Acurio Barriga

**Fuente:** Datos de la investigación