



*"Con el poder de tu mente,  
tu determinación, tu instinto  
y la experiencia, puedes  
volar muy alto"*

- Ayrton Senna-



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

Universidad de las Fuerzas Armadas ESPE-L

Departamento de la Ciencias de la Energía y Mecánica

Carrera de Ingeniería Automotriz

**“Diagnóstico avanzado en sistemas de inyección de gasolina mediante el uso de la red LPWAN - sigfox para determinar planes de mantenimiento y reparación”**

**Autores:**

Calvopiña Osorio, Fausto Andres

Gancino Gavilanez, John Renato

**Director:**

Ing. Erazo Laverde, Washington Germán

**Latacunga, agosto de 2022**



# ÍNDICE DE CONTENIDO

- Antecedentes**
- Planteamiento del problema**
- Descripción resumida del proyecto**
- Justificación e importancia**
- Objetivos del proyecto**
  - Objetivo General***
  - Objetivos Específicos***
- Metas**
- Hipótesis**



- Marco teórico**
  - Sistema de Inyección de gasolina**
  - Sensores del sistema**
  - Identificador de parámetros PIDs**
  - Protocolo de Comunicación**
  - Internet of the Things**
  - Arduino UNO**
  - Seed Shield Can Bus V2.0**
  - Red LPWAN**
  - Thinxtra developer XKit**
  - Ubidots**



- Análisis para la obtención y envío de datos.**
  - Caracterización del Vehículo**
  - Caracterización de los módulos utilizados**
  - Obtención de datos**
  - Envío de datos**
  - Disposición del hardware**
  - Programación de los módulos**
  - Registro y validación de Xkit SigFox**
  - Registro del Callback Ubidots**
  - Procesamiento de datos**
  - Planes de Mantenimiento y reparación**



- ❑ **Análisis de resultados**
  - ❑ **Prueba de ruta**
  - ❑ **Datos obtenidos**
  - ❑ **Comparación de datos**
  - ❑ **Pruebas en otros vehículos**
- ❑ **Conclusiones**
- ❑ **Recomendaciones**



# MARCO METODOLÓGICO





# ANTECEDENTES

Según (Guanoluisa & Paucar, 2020) establece el concepto del (IOT) internet de las cosas; como la evolución de la interacción de las personas con los componentes electrónicos que existen y que están por venir, con el objetivo de que exista comunicación entre los usuarios y la información que proporcionan los dispositivos. Al conectar los dispositivos a una red, permitirá avanzar el siguiente paso para el desarrollo de la automatización, precisión, domótica y entre otros aspectos que facilitarán la vida a los usuarios.

Los servicios de conectividad permiten la implementación de IoT, a través de la Red SIGFOX, con un bajo costo, bajo consumo de energía y además un largo alcance, transmitiendo datos sin la necesidad de mantener conexiones de red, administrando todo en la nube en lugar de usar dispositivos.

La generación actual de OBD es un sistema muy sofisticado y capaz de detectar problemas de emisiones. Sin embargo, es necesario que el técnico o agente regulador realice este proceso de forma manual. La idea es llevar OBD a un paso más allá, agregando transferencia de datos remota.





# PLANTEAMIENTO DEL PROBLEMA

Se debe mantener un monitoreo constante del sistema de inyección, de forma que se lleve a cabo el control periódico preventivo comúnmente basado en el odómetro (indica el kilometraje) o por tiempo, además que puede ayudar a evitar reparaciones costosas en el futuro, brinda seguridad a los ocupantes, buenas condiciones de trabajo y confort.

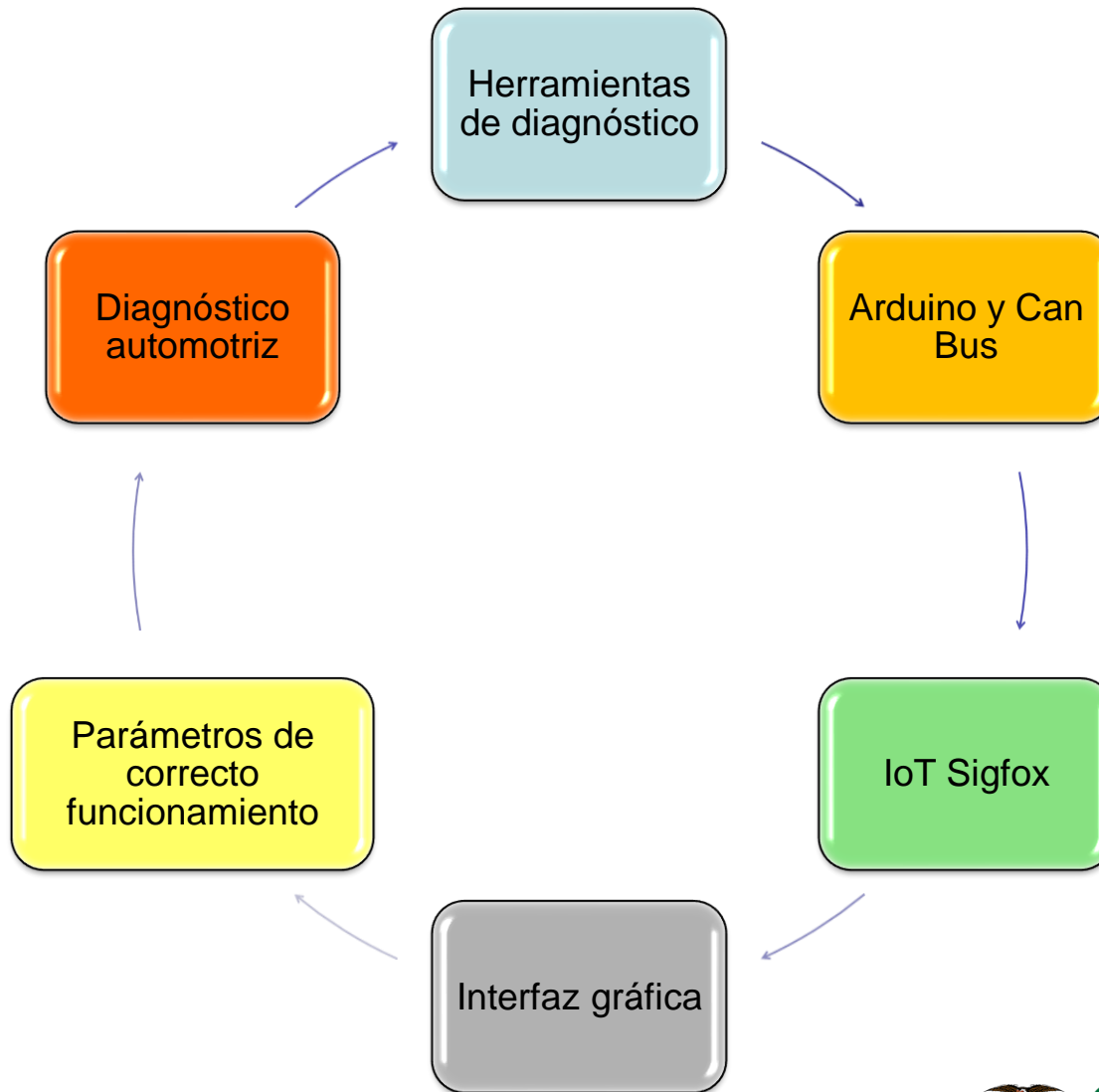
Con el desarrollo tecnológico de la red LPWAN - SIGFOX y la interoperabilidad del IoT (internet de las cosas) se procesan los datos que entregan los sensores en tiempo real, mediante la interfaz gráfica podrán ser compartidos con el usuario del vehículo, así como con el técnico de servicio para facilitar el diagnóstico, mantenimiento y/o reparación de este.



# DESCRIPCIÓN RESUMIDA DEL PROYECTO



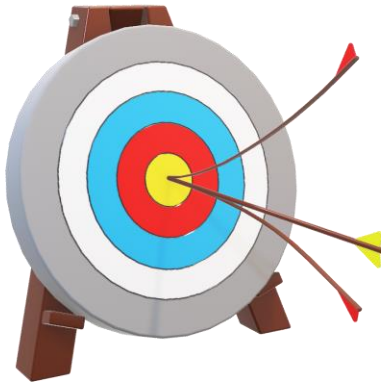
# JUSTIFICACIÓN E IMPORTANCIA





## OBJETIVO GENERAL

Desarrollar el diagnóstico avanzado en sistemas de inyección de gasolina mediante el uso de la red LPWAN - SIGFOX para determinar planes de mantenimiento y reparación.



# OBJETIVOS



Obtener información sobre la aplicación de los conceptos de IoT en conjunto con la tecnología LPWAN – SIGFOX y el módulo Shield Can Bus V2.0, para facilitar el diagnóstico, realizar planes de mantenimiento y reparación para el sistema de inyección electrónica a gasolina



Fundamentar teóricamente IoT, redes de comunicación, programación, módulo Shield Can Bus V2.0, placa SIGFOX RCZ4 DEV KIT, Arduino uno, sistemas de inyección de gasolina, OBD-II



Generar la base de datos de los rangos de funcionamiento permitidos en los sensores y actuadores del sistema de inyección según el tipo de vehículos a los que se vaya a monitorear



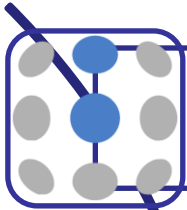
Realizar el código de programación para el funcionamiento del módulo Shield Can Bus V2.0



Procesar la información obtenida de los sensores y actuadores del sistema de inyección electrónica a gasolina



# OBJETIVOS



Configurar el Callback para observar los datos obtenidos de la ECU mediante la utilización de widgets en el dashboard.



Comparar la información recopilada del OBD-II mediante el uso del dashboard con los valores de la base de datos previamente investigada, que contienen los parámetros correctos de funcionamiento de cada componente



Analizar los valores obtenidos al de los componentes del sistema de inyección del vehículo con la base de datos para determinar el procedimiento que se deberá seguir al momento de reparar las averías.



Crear el plan de reparación para los sensores y actuadores cuando los valores obtenidos se encuentren fuera de su rango de funcionamiento óptimo.



Diseñar el plan de mantenimiento al sistema de inyección para los vehículos de la flota según su kilometraje y/o estado de los componentes.





Diagnosticar  
en tiempo real



Aplicar el  
código de  
programación



Realizar los  
planes de  
mantenimiento  
y reparación





## HIPÓTESIS

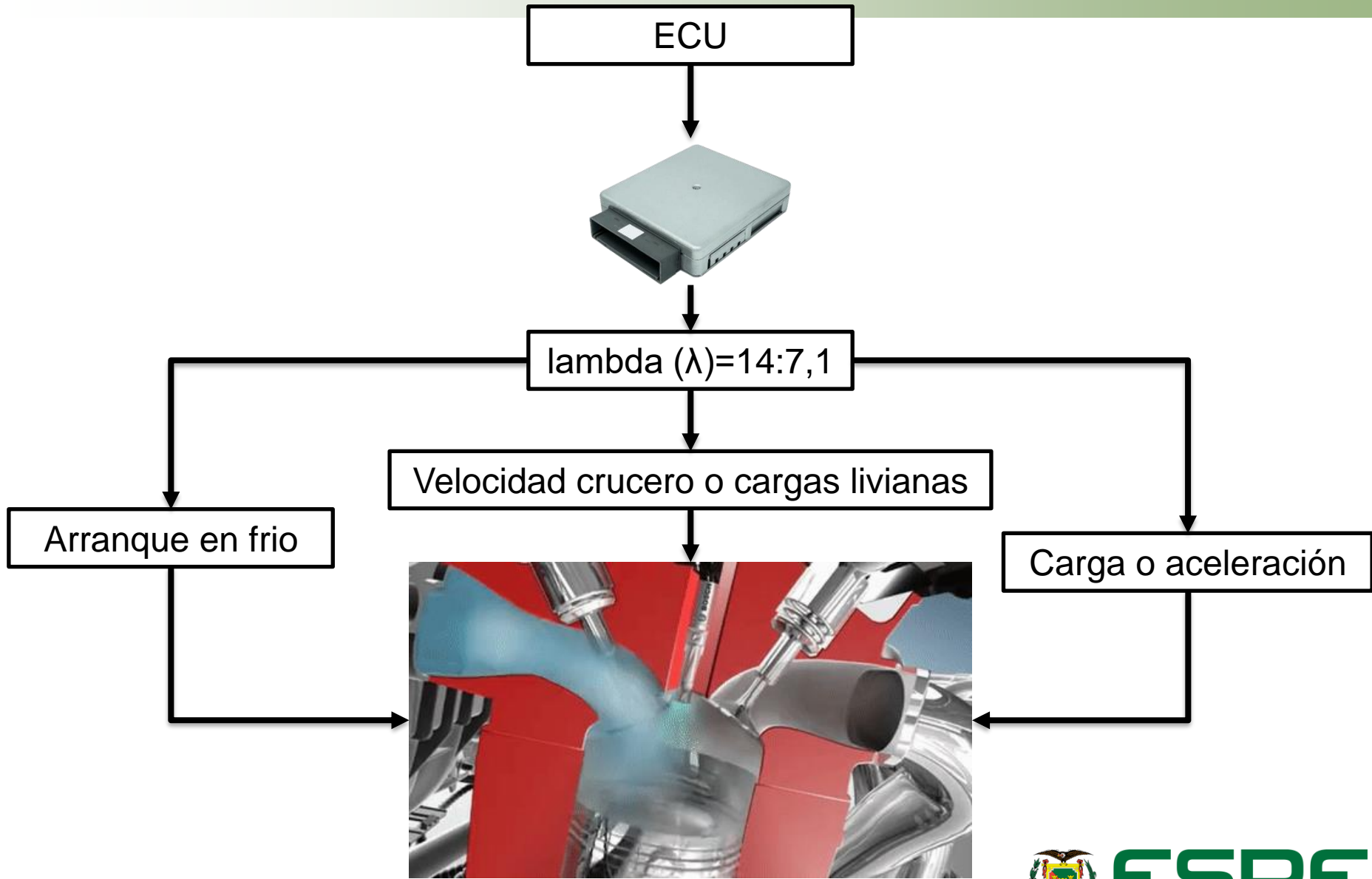
¿El diagnóstico avanzado del sistema de inyección mediante la red LPWAN - SIGFOX permitirá reducir los tiempos en el desarrollo de planes de mantenimiento y reparación?



# MARCO TEÓRICO

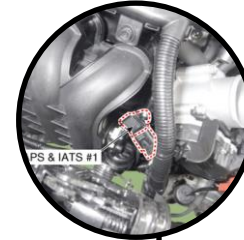
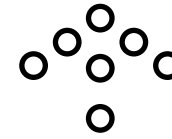


# Sistema de Inyección de gasolina



# Sensores del sistema

Denominación	Voltaje De Referencia	Voltaje De Señal
Sensor De Presión De Aire MAP (Maniflod Absolute Pressure Sensor)	5Vcc	0,5 a 4,5Vcc
Sensor De Posición Del Cigüeñal (Ckp)	5V	-5 a 5Vca
Sensor De Temperatura Del Refrigerante (Ect)	5V	0,5 a 4,5Vcc
Sensor De Posición De La Mariposa De Aceleración (Tps)	5V	0,5V ralentí 4,5V Abierto



SENSOR MAP



SENSOR CKP



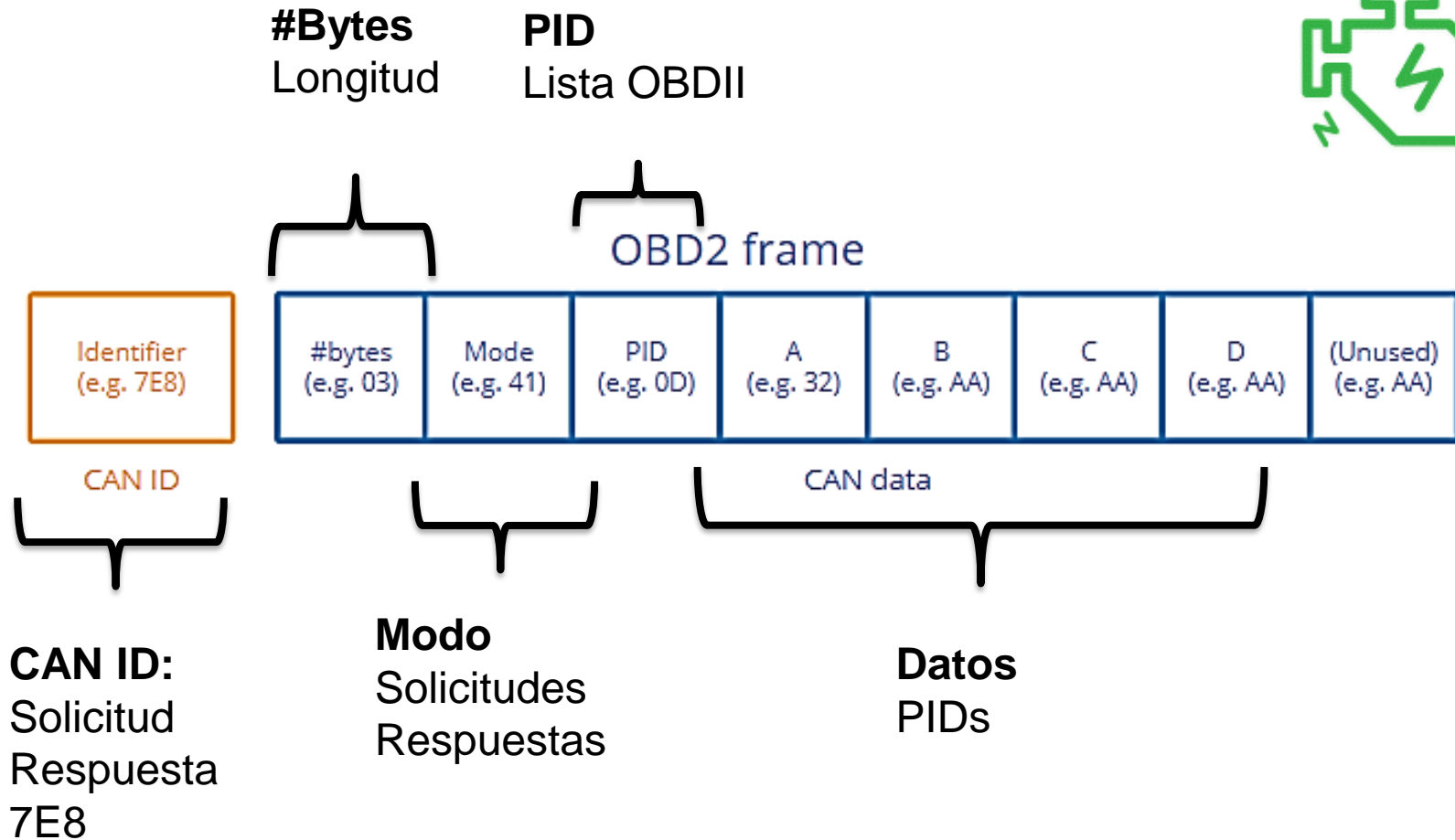
SENSOR ECT



SENSOR TPS



# Identificador de parámetros PIDs



# Protocolo de Comunicación

- Conjunto De Normas
- Estandarizaciones

○ SAE J1962

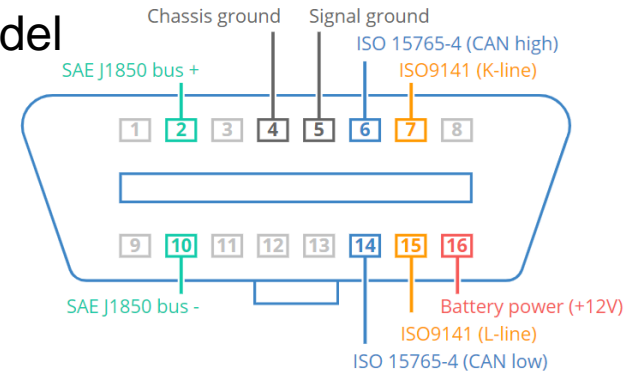
El estándar describe tanto el conector OBD2 del vehículo como el conector utilizado por el equipo de prueba externo

○ SAE J1979

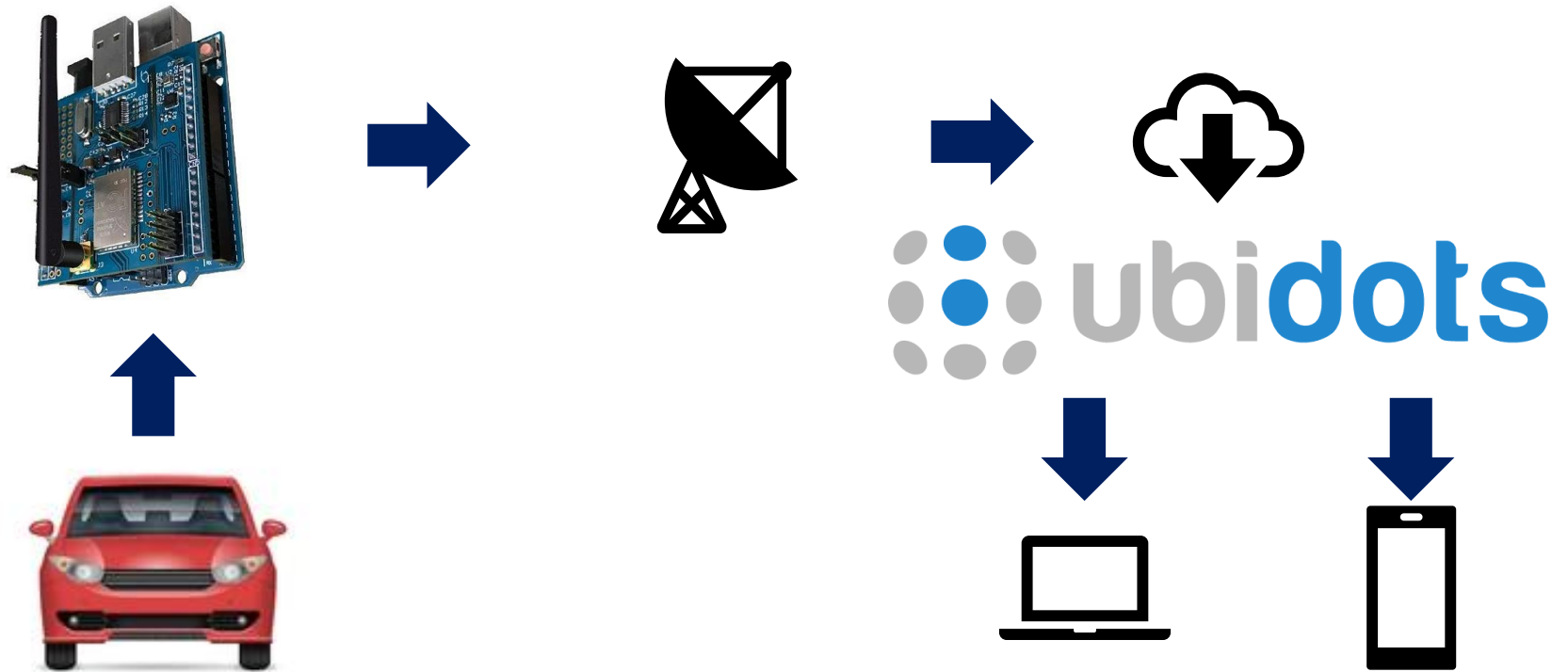
describe los métodos para solicitar información de diagnóstico a través del protocolo OBD2

○ ISO 15765-2

El estándar describe la 'Capa de transporte'



# Internet of the Things (IoT)





# ARDUINO UNO



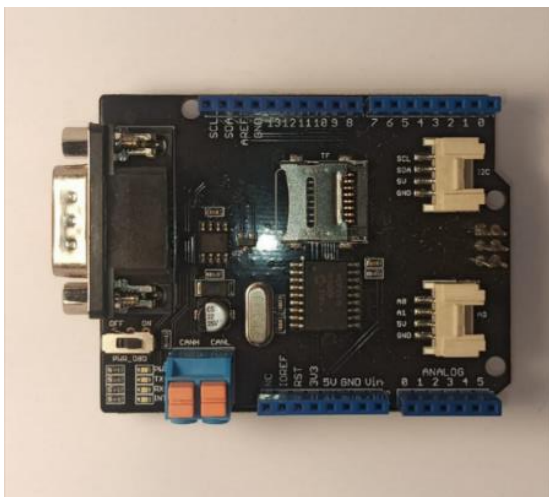
Código abierto – software libre  
Versatilidad



Placas más utilizadas  
Atmega 328T  
14 pines de entradas / salidas digitales

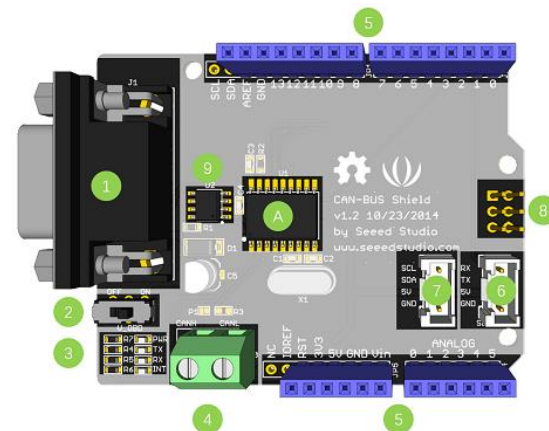


# Seed Shield Can Bus V2.0



Compatible con Arduino  
Controlador MCP2515 CAN-BUS

Interacción con la ECU – Puerto OBD II  
Protocolo CAN-BUS (ISO 15765-4)



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA



# Red LPWAN



- Bajo consumo y extensa área
- Transmisión de datos a larga distancia
- Innovación tecnológica



# Thinxtra developer XKit



- Utiliza la red LPWAN
- Comunicaciones de baja velocidad y bajo consumo de energía
- Radio configuración RCZ4 (parte de latino América)



# UBIDOTS



- Visualización de los datos extraídos del backend de Sigfox
- Tableros donde se muestra la información en tiempo real

- Aplicación móvil
- Visualización de los tableros desde el celular a una larga distancia



# Análisis para la obtención y envío de datos



# Caracterización del Vehículo



Kia rio sedan 2019

Ubicación del  
puerto OBD II



Protocolo Can-Bus  
(ISO 15765)

Pin 6 y 14

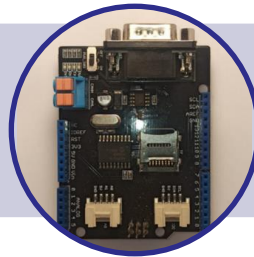


**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

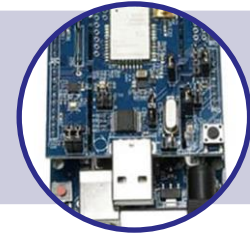


# Caracterización de los módulos utilizados

**Arduino UNO**



**Thixtra  
SIGFOX**



**Shield CAN-BUS.**



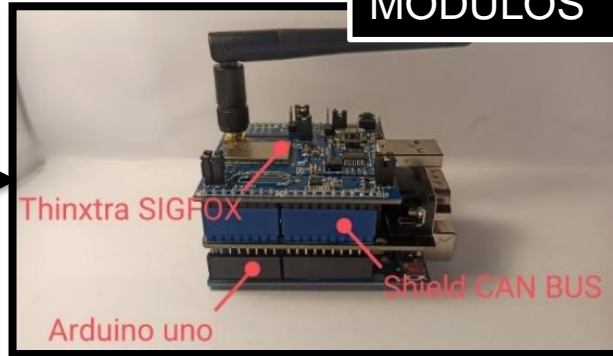
**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

# Disposición del hardware

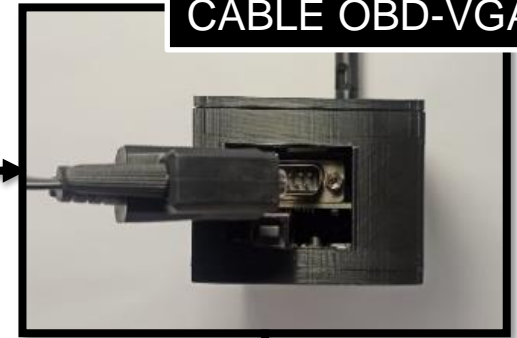
IDENTIFICAR  
CONECTOR OBD II



UBICAR LOS  
MODULOS



CONECTAR  
CABLE OBD-VGA



COLOCAR EN  
LUGAR IDEAL



# Programación de los módulos

```
1 #include <WISOL.h>
2 #include <Wire.h>
3 #include <math.h>
4 #include <SimpleTimer.h>
5 #include <avr/wdt.h>
6 #include <SPI.h>
7 #define CAN_2515
```

LIBRERIAS

ENVIO

```
51 void sendPid(unsigned char __pid)
52 {
53     unsigned char tmp[8] = {0x02, 0x01, __pid, 0, 0, 0, 0, 0};
54     CAN.sendMessage(CAN_ID_PID, 0, 8, tmp);
55 }
```

```
39 void set_mask_filt()
40 {
41     CAN.init_Mask(0, 0, 0x7FC);
42     CAN.init_Mask(1, 0, 0x7FC);
43     CAN.init_Filt(0, 0, 0x7E8);
44     CAN.init_Filt(1, 0, 0x7E8);
45     CAN.init_Filt(2, 0, 0x7E8);
46     CAN.init_Filt(3, 0, 0x7E8);
47     CAN.init_Filt(4, 0, 0x7E8);
48     CAN.init_Filt(5, 0, 0x7E8);
49 }
```

MASCARAS Y FILTROS



# Programación de los módulos

TEMPORIZADOR

```
79 }
80 //temporizador de envio para el Sigfox cada 3 minutos
81 unsigned long tiempo_envio = 3000;//30000;
82 timer.setInterval(tiempo_envio,enviar_OBDII);
83 Serial.println("");
84 delay(1000);
85 while (CAN_OK != CAN.begin(CAN_500KBPS))
86 {
87
88     delay(100);
89 }
```

```
93 void loop()
94 {
95     timer.run();
96
97     digitalWrite(13,!digitalRead(13));
98
99     delay(100);
100
101     //1 rpm
102     taskCanRecv();
103     pid=0xC;
104     sendPid(pid);
105     delay(100);
106
```

SOLICITUD DE PID



# Programación de los módulos

```
145 void taskCanRecv()  
146 {  
147     unsigned char len = 0;  
148     unsigned char buf[8];  
149     if (CAN_MSGAVAIL == CAN.checkReceive())  
150     {  
151         CAN.readMsgBuf(&len, buf);  
152         if(pid==0x0C) //rpm  
153         {  
154             numero=(buf[3]*256)+buf[4];  
155             rev=numero*0.25;  
156             word revl=rev;  
157             OBDII[0]=highByte(revl);  
158             OBDII[1]=lowByte(revl);  
159         }  
160     }  
161 }
```

RECEPCIÓN DE  
DATOS

COMUNICACIÓN  
CON SIGFOX

```
215 void envio_dato(uint8_t *sendData, const uint8_t len)  
216 {  
217     // No downlink message require  
218     recvMsg *RecvMsg;  
219     RecvMsg = (recvMsg *)malloc(sizeof(recvMsg));  
220     Isigfox->sendPayload(sendData, len, 0, RecvMsg);  
221     for(int i = 0; i < RecvMsg->len; i++)  
222     {  
223         Serial.print(RecvMsg->inData[i]);  
224     }  
225     Serial.println("");  
226     free(RecvMsg);  
227 }
```



# Programación de los módulos

TEMPORIZADOR

```
79  }
80  //temporizador de envio para el Sigfox cada 3 minutos
81  unsigned long tiempo_envio = 3000;//30000;
82  timer.setInterval(tiempo_envio,enviar_OBDII);
83  Serial.println("");
84  delay(1000);
85  while (CAN_OK != CAN.begin(CAN_500KBPS))
86  {
87
88      delay(100);
89  }
```

```
93 void loop()
94 {
95     timer.run();
96
97     digitalWrite(13,!digitalRead(13));
98
99     delay(100);
100
101     //1 rpm
102     taskCanRecv();
103     pid=0xC;
104     sendPid(pid);
105     delay(100);
106
```

SOLICITUD DE PID



# Registro y validación de Xkit SigFox

Si ya tiene una cuenta de identificación de Sigfox de otro servicio (Backend, Connect, Buy, Build), [validala](#).

Nombre \*  Apellido \*

Correo electrónico \*  Contraseña \*

País \*  Número de teléfono \*

Selección...

Nombre de la empresa \*  Tipo de empresa \*

Selección...

Posición

No soy un robot

Me gustaría recibir boletines de Sigfox

Acepto que mi operador local de Sigfox pueda acceder a mi información de contacto (no se utiliza para boletines)

DISPOSITIVO TIPO DE DISPOSITIVO USUARIO GRUPO

### Dispositivo - Nuevo

Información del dispositivo

Identificador (imei)

Nombre

PAC

certificado de producto final

¿Dónde puedo encontrar el certificado de producto final?  
El proveedor del dispositivo debe proporcionar el número de certificado de producto final. Si no es así, utilice la barra de búsqueda a continuación.  
Buscar el nombre del producto final

De lo contrario, puede ponerse en contacto con la mesa de servicio de su distribuidor Sigfox.  
Si el dispositivo aún no ha obtenido su certificado de producto final, puede registrarse como prototipo.  
 Registrarse como prototipo (solo de prototipos que se pueden registrar en su grupo: 1000)

Escribe  (Fichas disponibles: 0)

Latitud (°N° a +90°)

Longitud (°E° a +180°)

Mapa

Renovación automática de suscripción

Activable

Copyright © Sigfox - 11.8.18/PT/1a/2022/14.05/01/19 - 1/14 - Términos y condiciones / Política de cookies.

Registro en el Backen de Sigfox



Etiqueta del dispositivo



Registro y validación del dispositivo



# Registro del Callback Ubidots

API Key

Utilice su API key para crear Tokens temporales a través del siguiente [Auth endpoint](#). Este API key no se puede usar para autenticar peticiones a la API, su único propósito es generar Tokens.

API Key 1

Tokens

Utilice los Tokens para autenticar las peticiones a la API. Puede crear Tokens desde esta interfaz o desde el siguiente link a través de nuestra API.

NOMBRE	TOKENS	PARA LIMITE	ACTIONS
Ubidots App	.....	1 req/ts	
Monitoreo	8BFF-389hIhas...	1 req/ts	

Device type Thinxtra\_DevKit\_1 - Callback edition

Callbacks

Type: DATA | UPLINK

Channel: URL

Custom payload config: rpm: uint 16 vel 2 uint 8 lat 3 uint 8 lps 4 uint 8 ect 5 uint 8 map 6 uint 8

URL: `http://host:port/deviceName@time@key1@ver1@key2@ver2...`

Available variables: device, time, data, reqNumber, deviceTypeid

Custom variables: customData@rpm, customData@vel, customData@lat, customData@lps, customData@ect, customData@map

URL pattern: `https://industrial.api.ubidots.com/api/v1.6/devices/{device}/`

Use HTTP Method: POST

Send SW:  (Server Name Indication) for SSL/TLS connections

Headers: x-auth-token: BBFF-31VM4hvmc2XggJ1uR0h811Y4LgDky

Content type: application/json

```
Body
{
  "rpm": {customData@rpm},
  "vel": {customData@vel},
  "lat": {customData@lat},
  "lps": {customData@lps},
  "ect": {customData@ect},
  "map": {customData@map}
}
```

Credenciales  
API

Callbacks en  
Sigfox

Configuración  
Callback

Device type Thinxtra\_DevKit\_1 - New Callback

Create callbacks to connect Sigfox cloud to your server/platform.

A callback is a custom http request containing your device(s) data, along with other variables, sent to a given server/platform when the aforesaid device(s) message is received by Sigfox cloud.

**Custom callback**

Creates a new callback from Sigfox cloud to your own server. This is the "default" callback type. You can create a full custom request (http method, content type, headers, etc).

**AWS IoT**

AWS IoT is a managed cloud platform that lets connected devices easily and securely interact with cloud applications and other devices. AWS IoT can support billions of devices and trillions of messages, and can process and route those messages to AWS endpoints and to other devices reliably and securely.

**AWS Kinesis**

Amazon Kinesis is a platform for streaming data on AWS, offering powerful services to make it easy to load and analyze streaming data, and also providing the ability for you to build custom streaming data applications for specialized needs.

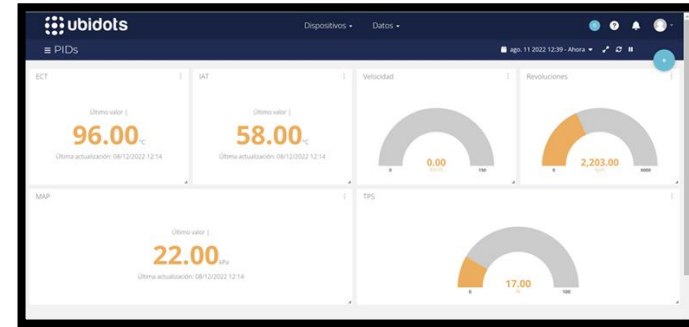
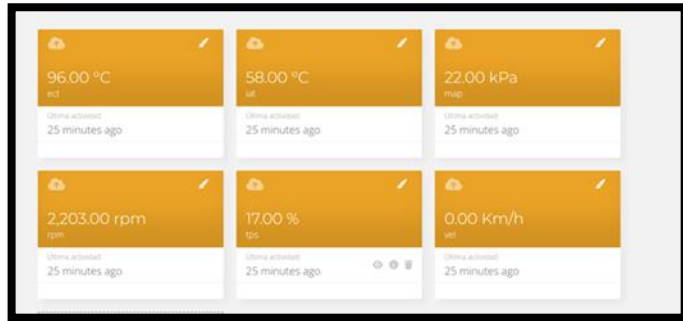
**Microsoft Azure Event hub**

Event Hubs is an event processing service that provides event and telemetry ingress to the cloud at massive scale, with low latency and high reliability. This service is especially useful for: application instrumentation, user experience or workflow processing, Internet of Things (IoT) scenarios.





# Procesamiento de datos



Datos recibidos

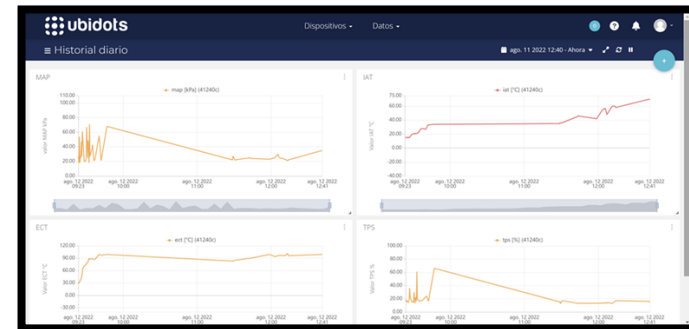


Creación de tableros de información



Tablero de información

A screenshot of the 'Crear un nuevo tablero' form. It has two tabs: 'CONFIGURACIONES' and 'APARIENCIA'. The 'CONFIGURACIONES' tab is active, showing fields for 'Nombre' (New Dashboard), 'Rango de tiempo por defecto' (Últimas 24 horas), 'Tablero Dinámico' (Estático), 'Resolución de pantalla' (Auto), and 'Formato de fecha' (08/12/2022 12:39). At the bottom, there are 'CANCELAR' and 'GUARDAR' buttons.



# Procesamiento de datos

The screenshot shows a configuration window for a data condition. The title bar includes 'ots', 'Dispositivos', and 'Datos'. The main area is divided into two sections: 'Si condición(es)' and 'Entonces acciones'. Under 'Si condición(es)', there is a field with the value '41240c:ect', followed by a dropdown menu set to 'Valor', the text 'es', a dropdown menu set to 'Mayor que', a text input field with '120', the text 'por', a text input field with '1', and the unit 'minuto'. Below these fields are two buttons: '+ Y' and '+ O'.

Mensajes de advertencia



Estructura del mensaje de alerta

The screenshot shows a configuration window for an alert message structure. The title bar includes 'ots', 'Dispositivos', and 'Datos'. The main area is divided into two sections: 'Si condición(es)' and 'Entonces acciones'. Under 'Si condición(es)', there is a dropdown menu for 'Dirección de correo electrónico' with the value 'facalvopina@espe.edu.ec' and a note 'Agregue correos electrónicos separados por comas'. Under 'Entonces acciones', there are two tabs: 'EN LA ACTIVACIÓN' and 'VUELTA A LA NORMALIDAD'. The 'EN LA ACTIVACIÓN' tab is active. It contains a dropdown menu for 'Asunto' with the value 'Fallo Nombre del dispositivo: alerta!', a dropdown menu for 'Mensaje' with the value 'Hola, la temperatura del motor está sobre los parámetros de correcto funcionamiento, el valor de activación fue Valor de activación a las Marca de tiempo de activación', and a section for 'Repetir acción cada' with a value of '5' and 'Minutos(s)', and 'hasta' with a value of '3' and 'veces'. There is also a note 'mientras la condición este activa'.

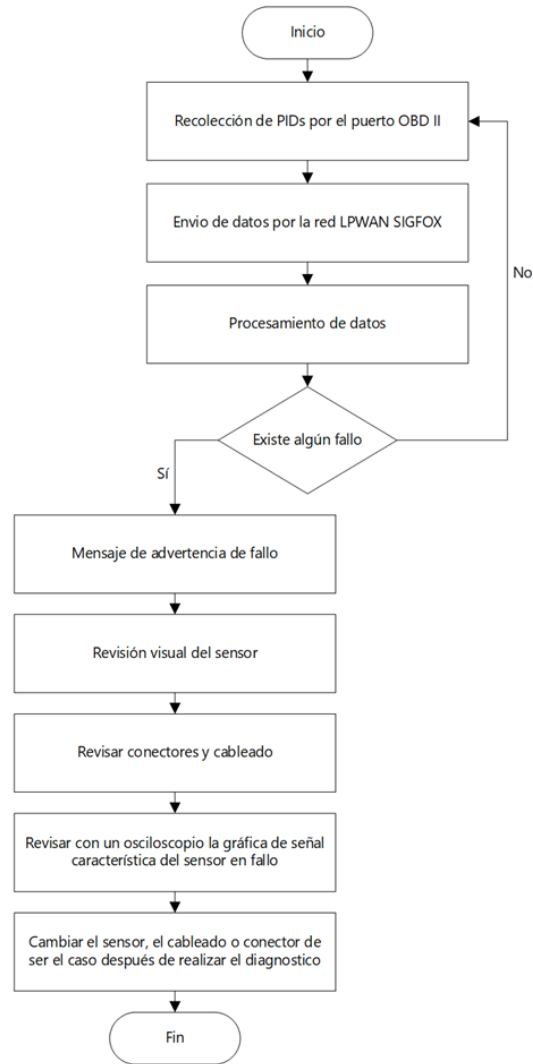


# Plan de Mantenimiento

Elemento	Inspección	Reemplazo
Aceite de motor	-	10.000 Km
Filtro de aceite	-	10.000 Km
Aceite de transmisión	-	60.000 Km
Bujías	25.000 Km	75.000 Km
Filtro de aire	-	30.000 Km
Refrigerante de motor	10.000 Km	30.000 Km
Pastillas de freno	20.000 Km	60.000 Km
Forros de freno	20.000 Km	80.000 Km
Líquido de freno y embrague		30.000 Km
Neumáticos	5.000 Km	Según el uso
Batería	4 meses	4 años



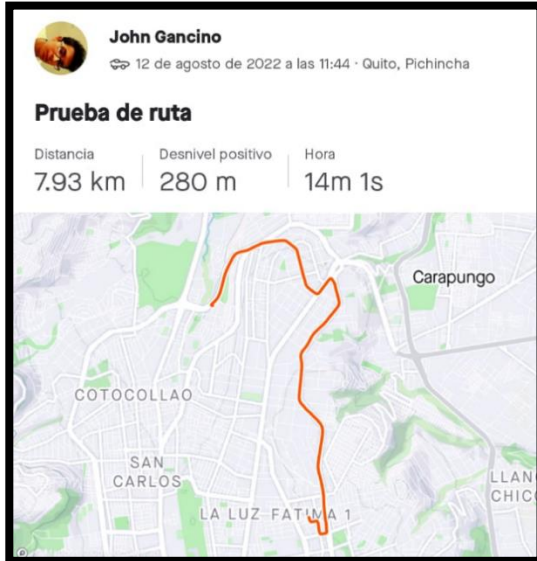
# Plan de reparación



# PRUEBAS Y ANÁLISIS DE RESULTADOS

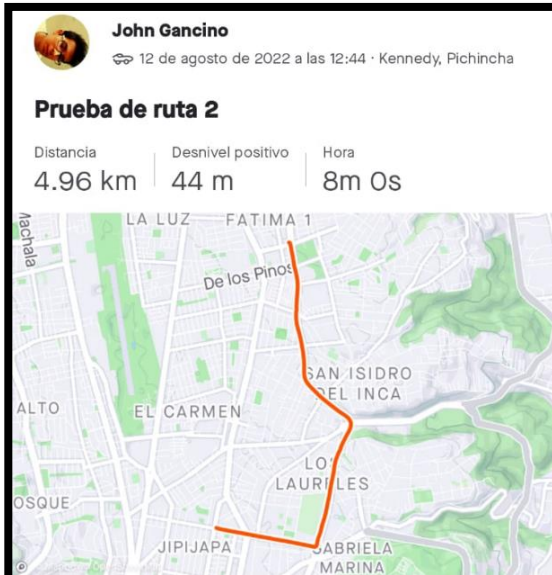


# PRUEBA DE RUTA



## PRUEBA DE RUTA

- Inicio 11:44 am
- Tiempo 14min 01s
- Colinas del norte – Hacienda Victoria



## PRUEBA DE RUTA 2

- Inicio 12:44 am
- Tiempo 8 min 01s
- Hacienda Victoria – Avenida Jipijapa



# Datos obtenidos

## Valores Prueba de ruta 1

	RPM	VEL (km/h)	IAT (°C)	TPS (%)	ECT(°C)	MAP (kPa)
11:43	599	0	45	13	99	23
11:43	600	0	46	13	90	24
11:58	890	15	42	13	90	25



# Datos obtenidos

## Valores Prueba de ruta 2

	RPM	VEL (km/h)	IAT(°C)	TPS (%)	ECT(°C)	MAP (kPa)
12:44	2668	32	65	8	96	28
12:45	594	0	66	5	95	25
12:46	1032	18	64	6	96	24
12:46	2339	55	63	7	96	24
12:47	878	22	63	5	92	24
12:47	598	0	65	5	93	25
12:50	987	12	56	6	87	37
12:53	3123	55	48	8	87	28
13:00	619	0	56	5	95	26





# COMPARACIÓN DE DATOS

Velocidad del motor (rpm)	Rango de medición
599	0 – 16.384 rpm
600	
890	
2.668	
594	
1.032	
2.339	
878	
598	
987	
3.123	
619	

Velocidad (km/h)	Rango de medición
0	0 – 255 km/h
0	
15	
32	
0	
18	
55	
22	
0	
12	
55	
0	



# COMPARACIÓN DE DATOS

IAT (°C)	Rango de medición
45	-40 a 215 °C
46	
42	
65	
66	
64	
63	
63	
65	
56	
48	
56	

TPS	Rango de medición
5	0-100%
5	
5	
8	
5	
6	
7	
5	
5	
6	
8	
5	



# COMPARACIÓN DE DATOS

ECT (°C)	Rango de medición
99	-40 a 255 (°C)
90	
90	
96	
95	
96	
96	
92	
93	
87	
87	
95	

MAP (kPa)	Rango de medición
23	0 – 255 (kPa)
24	
25	
28	
25	
24	
24	
24	
25	
37	
28	
26	



# PRUEBAS EN OTROS VEHÍCULOS



```
COM4  
  
21:40:21.009 -> CAN inCAN init ok!  
21:41:25.001 -> SEND PID: 0xC  
21:41:29.755 -> SEND PID: 0xD  
21:41:33.347 -> SEND PID: 0x5  
21:41:37.614 -> SEND PID: 0xA6
```

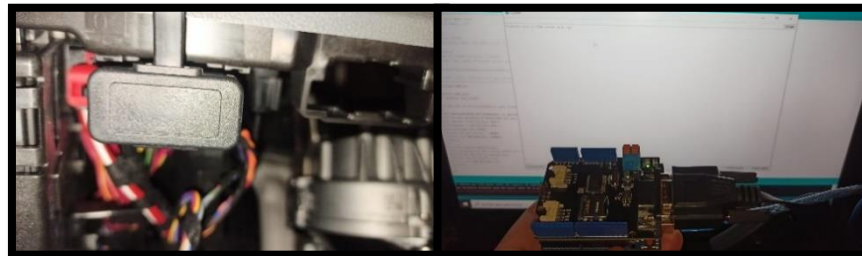
Vehículo  
de prueba



Conexión



Monitor  
serial



- Se desarrolló el diagnóstico avanzado en sistemas de inyección de gasolina mediante el uso de la red LPWAN - SIGFOX para determinar planes de mantenimiento y reparación en tiempo real.
- Se obtuvo información sobre la aplicación de los conceptos de IoT en conjunto con la tecnología LPWAN – SIGFOX y el módulo Shield Can Bus V2.0, que facilitó el diagnóstico, la realización de planes de mantenimiento y reparación para el sistema de inyección electrónica a gasolina utilizando temas relacionados con el proyecto como: libros, artículos científicos, investigaciones científicas, páginas de internet de contenido confiable.
- Con la ayuda de fuentes bibliográficas y fichas técnicas de los fabricantes se fundamentó teóricamente los conceptos a utilizar en la investigación como el IoT, redes de comunicación, programación, módulo Shield Can Bus V2.0, placa SIGFOX RCZ4 DEV KIT, Arduino uno, sistemas de inyección de gasolina, OBD-II, planes de mantenimiento y reparación.



- Para la obtención de los datos y envío a la nube de estos, se realizó el código de programación para el funcionamiento del módulo Shield Can Bus V2.0, juntamente con la placa de Arduino UNO y el módulo SIGFOX RCZ4 DEV KIT.
- Una vez conocidos los sensores de los que se va a obtener los datos se generó la base de datos de los rangos de funcionamiento permitidos en los sensores del sistema de inyección según el tipo de vehículos a los que se vaya a monitorear.
- Se procesó la información obtenida de los sensores del sistema de inyección electrónica a gasolina extraída con el módulo Shield Can Bus V2.0 y enviada backend de Sigfox.
- Con los datos en el backend de Sigfox se configuró el Callback de forma que se puedan observar los datos obtenidos de la ECU mediante la utilización de widgets en el dashboard.



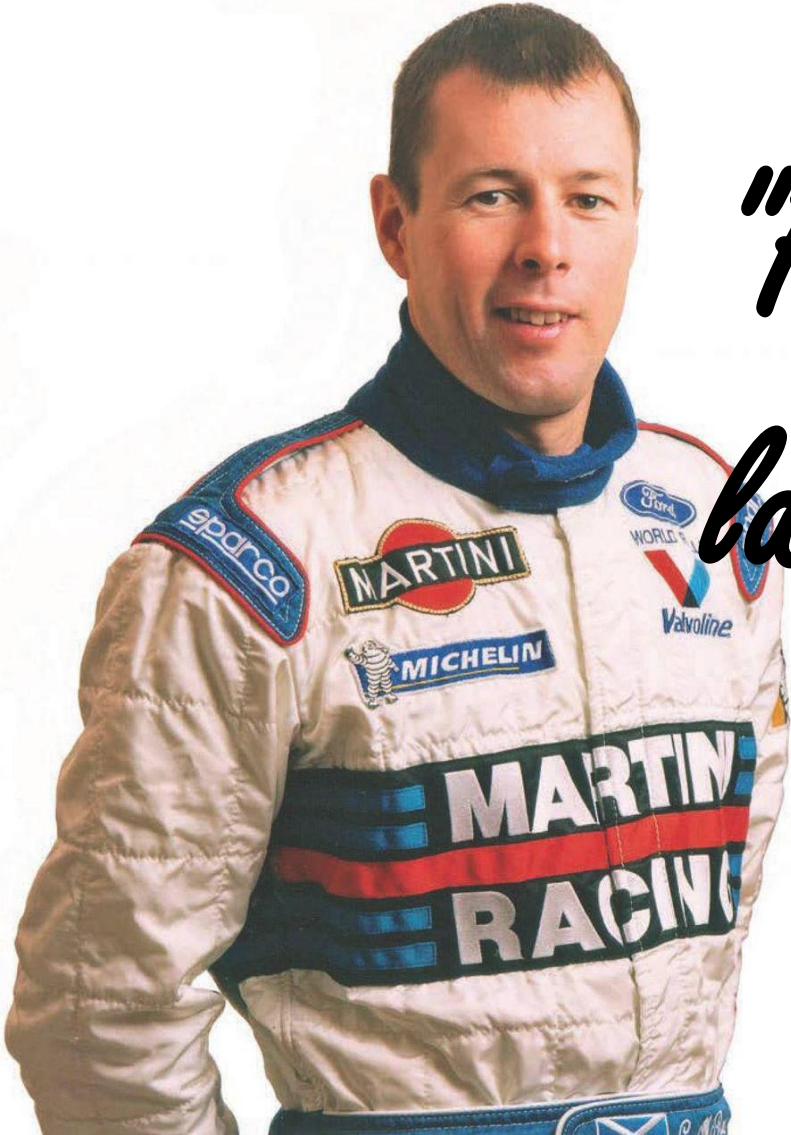
- Se reviso que los valores obtenidos se encuentren dentro de los parámetros correctos de funcionamiento, para esto se comparó la información recopilada del OBD-II mediante el uso del dashboard con los valores de la base de datos previamente investigada.
- Se analizo los valores obtenidos de los componentes del sistema de inyección del vehículo con la base de datos para determinar el procedimiento que se deberá seguir al momento de reparar las averías.
- El plan de reparación se creo para el caso de que los sensores del sistema de inyección funcionen fuera del rango correcto, se deben seguir los pasos especificados en este para poder arreglar la avería.
- Para que un vehículo tenga una vida útil larga se debe tener en cuenta un plan de mantenimiento del sistema de inyección para los vehículos que se diseñó en función del kilometraje y/o estado de los componentes, este plan de mantenimiento no se puede utilizar dentro de la aplicación debido a que el PID del odómetro no se pudo obtener.



- Utilizar la información sobre IoT de las diferentes ramas y clasificarlas de forma que sean útiles en el área automotriz.
- Para las pruebas hay que revisar que el vehículo tenga protocolo de comunicación SAE 15765-4, se observa que en el conector OBD II se tengan los pines 6 y 14.
- En la recolección de datos revisar que la conexión del módulo con el puerto OBD II se encuentre bien de forma que se puedan obtener los valores de los sensores que se está analizando.
- Cuando existan fallas en el sistema de inyección y el módulo envíe el mensaje de advertencia, se deberá utilizar otros equipos para el diagnóstico como son el osciloscopio y el scanner.
- Ubicar el módulo en un lugar donde no tenga interferencia en la antena, para lo cual se sitúa sobre el tablero de instrumentos, cerca del parabrisas.







*"Pie a chapa, y, ante  
la duda, gas a fondo"*

*- Colin McRae*



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA