

SOFTWARE de DIAGNOSIS

(Sepa cómo funciona la Diagnósis en los Coches)

Información facilitada por AUTOXUGA; Fabricante de Software de Diagnósis

1.- AVANCES DE LA DIAGNOSIS EN LA AUTOMOCION

1.1.- LOS COMIENZOS DE LA DIAGNOSIS. A finales del siglo XIX se introdujo en Europa el automóvil como medio de transporte. Estos primeros vehículos llevaban un motor de combustión interna de cuatro tiempos bastante pesado y rudimentario. Más adelante, Daimler ideó una variante mucho más ligera que sería el precursor de todos los motores a explosión posteriores. Con los años, los automóviles fueron incorporando innovaciones que aumentaron su rendimiento y mejoraron sus prestaciones. Estas mejoras incluían el uso de diferencial, correas, baterías, etc., pero en su diseño, el motor de combustión interna no experimentó cambios substanciales y será en el siglo XX cuando se introduzca la electrónica.

1.2.- NECESIDAD DE LA DIAGNOSIS. Ya bien entrado el siglo XX las innovaciones mecánicas siguieron sin afectar al diseño básico de los motores, suponiendo tan solo la adición de elementos orientados a la optimización de los mismos y será a finales de los años 70 cuando se empieza a incorporar masivamente la electrónica a los automóviles y se añaden los primeros sensores a los motores para que éstos verificaran su correcto funcionamiento. Igualmente se adoptaron Centralitas o UCEs para que manejaran los sensores de los motores con un objetivo de que estos elementos electrónicos sirvieran para controlar las **emisiones de gases contaminantes** y facilitar la diagnóstico de averías de los coches.

1.3.- INCORPORACION ELECTRÓNICA EN EL AUTOMÓVIL. A partir de la década de los 80 la mayor parte de las innovaciones provienen principalmente de la incorporación de la electrónica en los vehículos y no de la incorporación de mejoras mecánicas. Se añadieron multitud de sensores y se fueron mejorando las Centralitas o UCEs del motor hasta que al día de hoy un automóvil incorpora más de 1.000 sensores y más de una UCE o Centralita, de manera que los coches actuales tienen UCEs de **Motor, Airbag, ESP, ABS, Cuadro instrumentos, Cambio automático, Electrónica central, Clima, Inmovilizador, Cierre centralizado, Luces, Asistente aparcamiento, Navegador, Dirección asistida, Volante multifunción, Regulación velocidad, Gateway, Techo, Neumáticos**, etc.

1.4.- SISTEMAS ELECTRÓNICOS ACTUALES. Actualmente los sensores y captadores se encargan medir **temperaturas, presiones puntuales y diferenciales, revoluciones, caudales, volúmenes aire, valores lambda, altitudes, resistencias, velocidades, sentido marcha coche, señales de tráfico**, etc. y la información que captan estos sensores son enviadas a las UCEs o Centralitas que las almacenan de manera que es muy fácil conocer el estado de cómo se encuentra el automóvil tanto relacionado con el Motor como con el resto de componentes.

En realidad, los sensores se limitan a detectar una serie de valores que envían continuamente a las UCEs o Centralitas y una vez allí se comparan con los valores de diseño que están almacenados en sus memorias y cuando un valor es incorrecto, la Centralita o UCE notifica un fallo que en la gran mayoría de los casos se detecta con un Equipo de Diagnóstico Profesional o Apps que, en este caso, es el tema que se va a exponer en esta divulgación tecnológica.

1.5.- CONOZCA EL STANDARD OBD-II PARA LA DIAGNOSIS. En Abril de 1985 un organismo estatal de California, el CARB (California Air Resources Board), aprobó una

regulación para un sistema de diagnóstico a bordo de los coches u **OBD** (On-Board Diagnostic) y esta diagnosis OBD se aplicaba a sistemas que se consideraban responsables de emisiones de gases de escape. Por ello se obligaba a cada fabricante a diseñar sus propios conectores para una diagnosis simple que también se podía realizar por códigos intermitentes.

Por todo lo anterior surge el **OBD-II** en los años 90 y se empezó a hacer obligatoria a partir de enero de 1996 por lo que la **SAE** (Sociedad de Ingenieros de Automoción) definió un **conector estándar OBD de 16 pines** así como un conjunto de códigos para homogeneizar todo lo relacionado con la diagnosis de los coches.

PROTOCOLOS OBD

2.- DEFINICION DE LOS PROTOCOLOS OBD

2.1.- **NORMATIVA DE LOS PROTOCOLOS OBD.** Los protocolos **OBD** son unos **estándares** definidos por el **ISO** y la **SAE** que permiten que un **ordenador** o un **teléfono móvil** se pueda comunicar con la Centralita electrónica o UCE del **motor** del coche y la finalidad de estos **protocolos OBD** es el poder controlar las **emisiones** de los automóviles con el objetivo de **reducir la contaminación** del aire y en caso de producirse un **funcionamiento inadecuado** del motor, el sistema hace que se encienda una **lámpara de Averías**, también llamada **MIL**, que avise al conductor de que algo relacionado con la contaminación no funciona correctamente.

2.2.- **EVOLUCION DE LOS PROTOCOLOS OBD.** En **OBD** solo hay **6 Protocolos** que a lo largo del tiempo se fueron implantado en los vehículos y que son los siguientes:

A).- El primero que se adoptó fue el **ISO 9141** que es un Protocolo OBD que se empieza a instalar en los coches a partir de **1989**. Los códigos de averías que devolvía este protocolo se basaban en la emisión de 2 o 4 destellos luminosos, según establecía cada fabricante de coches y que el mismo código, según la Marca y Modelo del coche, indicaba una avería distinta. Para obtener los códigos se utilizaba un Aparato LED Comercial o bien una lámpara de pruebas con un simple Led y una resistencia de 250 ohmios tal como se indica en la siguiente imagen que muestra un código de averías 1231 que para la marca BMW significa que la “**tensión de batería**” está mal.



Mostramos el significado de algunas averías de la Marca BMW que se extraían por códigos intermitentes.

Codigos Intermitentes para BMW:

Codigo Averia	Descripcion
0000	Fin extraccion averias
1000	Fin extraccion averias
2000	Fin extraccion averias
1211	UCE o Modulo de control defectuoso
1215	Medicion de caudal masa aire
1216	Sensor posicion mariposa
1218	Fases finales con elemento actuador defectuosas
1221	Sonda lambda
1222	Regulacion lambda
1223	Sonda de temperatura de agua
1224	Sonda de temperatura de aire
1231	Tension de bateria

B).- El segundo Protocolo OBD es el **ISO 9141-2** que se implanta en vehículos fabricados a partir del año **2000** hasta el **2004** y ya no utilizan los destellos luminosos sino que la extracción de los códigos de las Centralitas o UCEs de los coches se realiza con un **Software Informático** que se introduce en un **Ordenador**, necesitando un **circuito electrónico** que sirve de puente entre la Centralita o UCE del coche y el Ordenador. La comunicación entre Ordenador y Centralita se realiza a través del Puerto **RS232** del ordenador que se conecta al **circuito electrónico** el cual tiene que permitir la emisión y recepción de señales a una velocidad de **10400 baudios**. La forma de trabajar de este Protocolo para inicializar las comunicaciones tiene que ser **lento**, que quiere decir que la inicialización o despertar de la UCE se realiza a una velocidad 5 baudios ya que la UCE está en reposo y para activarla deberá enviarse un byte (33, 01, 02, 03, 7A, 82, etc.) que representa su dirección y, de esta manera, se activará la UCE que responderá con el envío de su Protocolo interno (0808, 018A, 6B8F, C446, etc.) de funcionamiento. Utilizan el Tipo de Inicialización **Lento** los Protocolos: ISO9141, ISO9141-2, VAG, MB1 y algunas veces el

ISO14230 que para su la inicialización puede utilizar la línea de comunicaciones K o bien la línea L para coches más antiguos.

C).- El tercer Protocolo OBD es el **ISO 14230-4 (KWP2000)** que se instaló en vehículos desde el **2003** hasta **2008** produciéndose la comunicación entre el Ordenador y la Centralita a través del puerto **RS232** a 10400 baudios.

El Protocolo **ISO14230** puede ser **Lento, Rápido o Nolnit** y si el Protocolo ISO14230 es **Lento** va a funcionar de manera similar al ISO9141-2, aunque lo normal es que en lugar de 0808 o 9494, puedan aparecer otros distintos bytes que igualmente indican el protocolo de la UCE.

Si el Protocolo ISO14230 es **Rápido**, se envía UN byte **00** a la UCE a **360 baudios** y antes de que transcurran **25 milisegundos** debe enviarse un paquete de datos cuya forma puede variar, pero que según la normativa debiera ser una cadena como esta **8110F18103** para el caso de Peugeot, en lo cual **10** es la dirección UCE, **F1** es la dirección TESTER, **81** es inicializar comunicación y **03** es el checksum del paquete.

Si el protocolo ISO14230 es **Nolnit** que indica que no hay que inicializar las comunicaciones, el procedimiento a seguir es exactamente igual que el Rápido, pero en este caso no se envía el byte **00** a la UCE y en concreto este Protocolo es el que se suele utilizar en casi todas las Marcas de Coches con ligeras variantes.

D).- Los restantes protocolos son el **SAE J1850 PWM** para **Ford** que se implanta en los vehículos americanos de Ford y Mazda hasta el año **2003** y la comunicación es a 41600 baudios. También el **SAE J1850 VPW** para **GM** que se implanta en vehículos americanos General Motors hasta el año **2003** y la comunicación es a 10400 baudios y quedan por mencionar el Protocolo **ISO 15765-4** para **CAN BUS** que se implanta desde el **2008** hasta

la fecha y que a su vez la comunicación entre el Ordenador y la UCE del Coche de este protocolo CAN BUS va a funcionar de 4 formas distintas:

- 1.- **ISO 15765-4 CAN** con mensajes de **11 bits** y velocidad de **500** Kilobaudios (**Kb**)
- 2.- **ISO 15765-4 CAN** con mensajes de **29 bits** y velocidad de **500** Kilobaudios (**Kb**)
- 3.- **ISO 15765-4 CAN** con mensajes de **11 bits** y velocidad de **250** Kilobaudios (**Kb**)
- 4.- **ISO 15765-4 CAN** con mensajes de **29 bits** y velocidad de **250** Kilobaudios (**Kb**)

2.3.- DIFERENCIA ENTRE PROTOCOLOS OBD Y SISTEMAS MARCAS. En OBD solamente hay **6 Protocolos** distintos, mientras que en las Marcas hay más de **60.000 Sistemas**. Los Protocolos **OBD** son unos **estándares** definidos por el **ISO** y la **SAE** para poder controlar las **emisiones** de los automóviles y **reducir la contaminación** y cuando se produce un **funcionamiento inadecuado** del motor se almacena el código de la avería y se enciende en el cuadro una **lámpara de averías** que también se conoce como **MIL**.

Las Marcas utilizan más de **60.000 Sistemas** porque cada Marca o Fabricante anda por libre utilizando sus propios sistemas de diagnosis que no siguen ninguna normativa estándar y, por tanto, los Sistemas de Marcas, además de chequear las emisiones contaminantes como los OBD, también chequean otros datos del motor y otras UCEs o Centralitas tales como: **Airbag, ESP, ABS, Cuadro instrumentos, Cambio automático, Electrónica central, Clima, Inmovilizador, Cierre centralizado, Luces, Asistente aparcamiento, Navegador, Dirección asistida, Volante multifunción, Regulación velocidad, Gateway, Techo, Neumáticos**, etc.

Como en **OBD** sólo hay **6 protocolos** y en las marcas existen **MILES de Sistemas** (más de **60.000**) que vienen a ser unos protocolos propios de cada Fabricante, es por ello, que el desarrollo de los Equipos de Diagnóstico o de Apps para las Marcas de Coches es complejo y complicado, debido a que no siguen ningún estándar ni hacen pública su información para diagnosticar los Sistemas que utilizan y, además, se necesitan unos **circuitos electrónicos** complejos para el caso de la Diagnóstico Profesional y si se quiere utilizar unos **adaptadores** que suplan a los circuitos de los equipos profesionales también hay que diseñar unos circuitos bastante complejos para que funcionen las Apps a través de los teléfonos móviles.

2.4.- INTERCAMBIO DATOS ENTRE UCEs COCHES Y ORDENADOR. Para que se pueda realizar un intercambio de Datos entre las UCEs o Centralitas de los coches y un Ordenador o Teléfono Móvil hay que establecer unas reglas concretas para que un dispositivo externo como es el caso de un Ordenador o Teléfono Móvil se pueda comunicar con las UCEs o Centralitas de los coches al objeto de que se puedan realizar las comunicaciones entre ambos elementos que necesitarán un nuevo elemento que va a ser el **Circuito Electrónico** al que llamaremos **Hardware** y que tiene que estar desarrollado de manera que entienda como funcionan los Protocolos de comunicación, los Sistemas de Marcas y los Ordenadores o Teléfonos Móviles.

El otro elemento imprescindible para que se produzca el intercambio de datos o mensajes entre las UCEs o Centralitas de los coches y el Ordenador o Teléfono móvil a través del Circuito Electrónico (**Hardware**) va a ser el **Programa Informático (Software)** que realmente es el que interpreta los mensajes enviados por el Ordenador así como los recibidos de las UCEs o Centralitas, mostrando a continuación los resultados finales comprensibles para las personas.

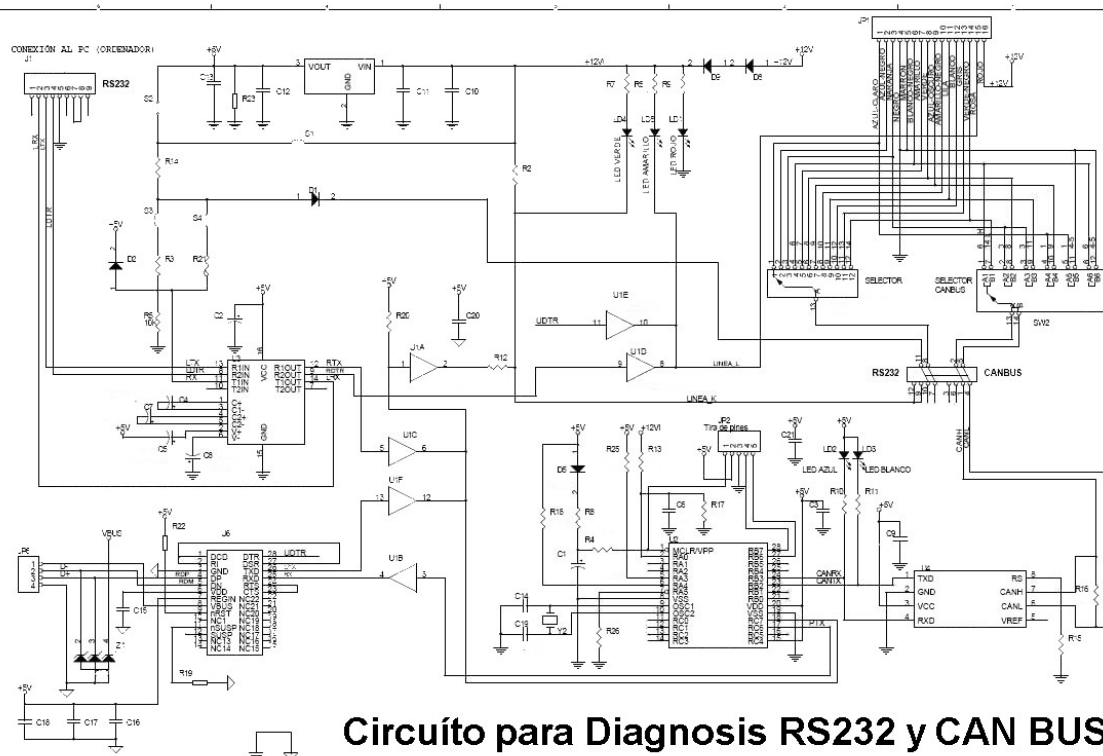
COMPONENTES

3.- EL HARDWARE o CIRCUITO ELECTRONICO

3.1.- REQUISITOS DEL CIRCUITO ELECTRONICO. Es la pasarela entre la UCE del coche y el Ordenador y que va a evitar que cuando se conecte el Ordenador con la UCE no se estropee ninguno de estos componentes y por tanto el circuito lo podemos desarrollar de forma básica ó SIMPLE cuya función consista solamente en regular voltajes y adaptar impedancias (valores de resistencias) entre la UCE del coche y el Ordenador para evitar que no se deterioren estos componentes, o bien podemos desarrollar un circuito COMPLEJO que, además de lo anterior, mantenga las comunicaciones entre la UCE y el Ordenador, y para este caso, hay que añadirle al circuito electrónico un Cuarzo (señal de reloj) y un Micro Controlador (PIC) al que se le introducirá un pequeño programa que se encargue de mantener las comunicaciones entre el Ordenador y la UCE del coche.

3.2.- CIRCUITO ELECTRONICO. Para ver la complejidad de los circuitos electrónicos de los Equipos de Diagnósis mostramos el Esquema Eléctrico del circuito de Diagnósis Profesional AUTOXUGA que va a actuar de interfaz RS232 para intercambio de datos binarios entre el Ordenador o Smartphone y las UCEs ó Centralitas de los Coches.

El esquema de este Circuito está realizado para que se pueda conectar a través de un puerto RS232 de 9 pines de un Ordenador a la toma de diagnóstico del Coche de 16 pines y, además, va a permitir la comunicación con otras UCEs provistas no solo de los **6 Protocolos** OBDII, sino que se puede comunicar con los **miles** de Sistemas de Marcas, tanto RS232 como CAN BUS, CAN ODX que se seleccionarán con un conmutador.



3.3.- EXPLICACION DEL CIRCUITO ELECTRONICO. En la parte superior izquierda del esquema se presenta el conector **RS232** de 9 pines para conectar al ordenador y en la parte superior derecha se muestra el conector OBD del coche con el conmutador que en la posición izquierda permite funcionar el circuito como RS232 y en la posición derecha va a funcionar este mismo circuito como CAN BUS.

En la parte central del lado izquierdo figura un integrado **MAX232** que es un circuito que convierte las señales de un puerto serie **RS-232** a señales compatibles con los niveles **TTL** de los circuitos lógicos utilizados por las UCEs o Centralitas de los coches, a los cuales les llega la tensión de Batería que para el caso de los vehículos de turismo es de 12V y que en la entrada de la UCE o Centralita se rebajan a una tensión media de alimentación ó **Vcc de 5V** (de 4,75V a 5,25V) con un **nivel lógico L ó Bajo** de 0,0V a 0,8V y **nivel lógico H ó Alto** de 2,2V a 5V. En el siguiente cuadro se muestra la correspondencia entre los niveles RS232 y TTL que convierte el MAX232.

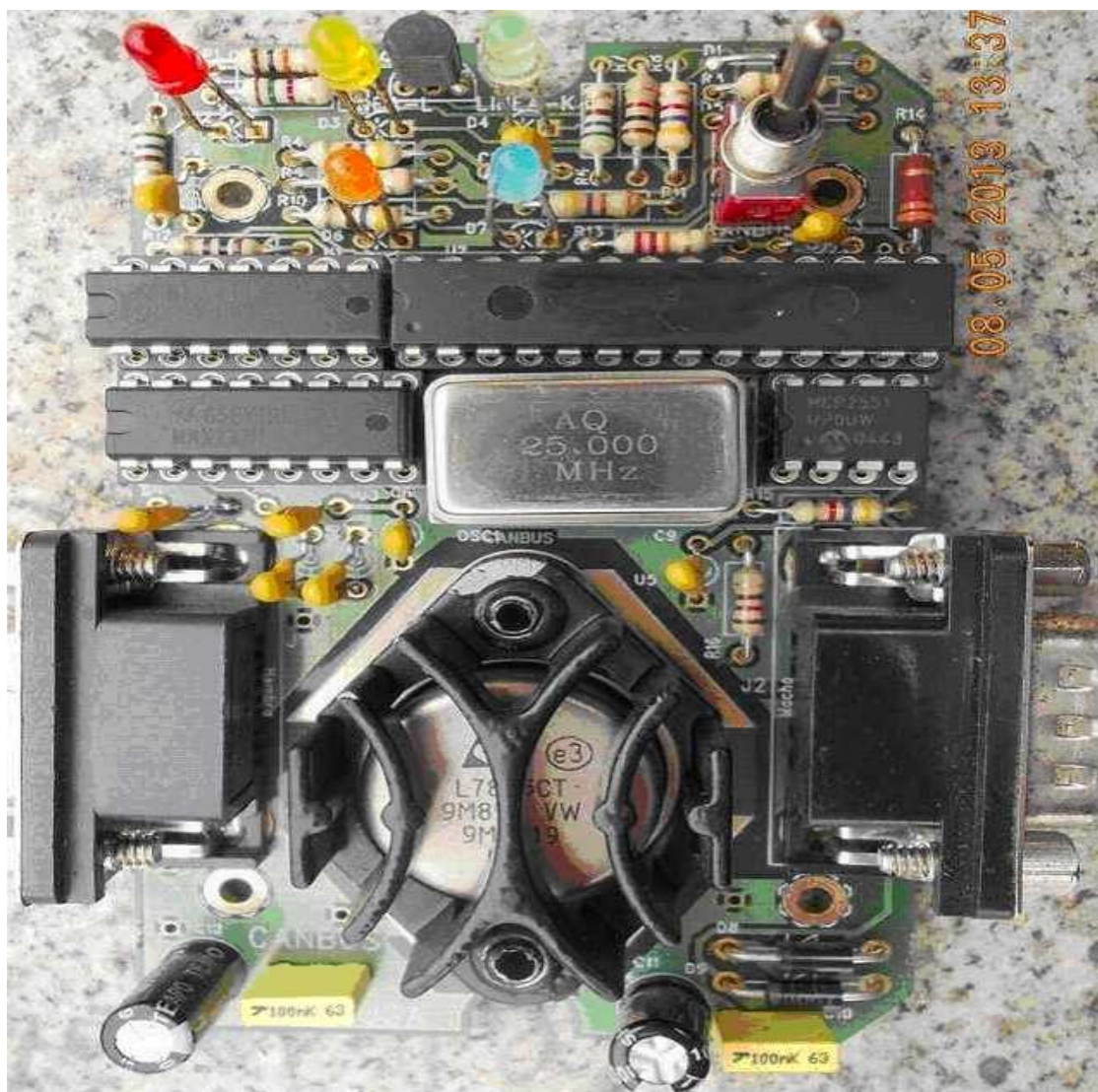
En el cuadro siguiente se muestra la conversión de datos de Transmisión (Tx) y Recepción (Rx) que transforman las Tensiones o Voltajes del Puerto RS232 de +3 a +15V en 0V o nivel lógico 0 (Bajo) y las Tensiones o Voltajes del Puerto RS232 de -3 a -15V en 5V o nivel lógico 1 (Alto).

Tipo línea RS232	Voltaje RS232	Tensión ó Voltaje TTL hacia el MAX232 o desde el MAX232
Transmisión/Recepción datos (Tx/Rx)	+3 V a +15 V	0 V = Nivel lógico 0
Transmisión/Recepción datos (Tx/Rx)	-3 V a -15 V	5 V = Nivel lógico 1

Siguiendo con el circuito anterior, en la parte inferior izquierda se muestra un integrado **CP2102** que es un conversor de USB a RS232, funcionando dicho integrado a velocidades desde los 300 bps hasta 1 Mbps.

El **PIC o Micro Controlador** de la parte inferior de dicho circuito que tiene 28 pines es el que se encarga de realizar las comunicaciones cuando el circuito funciona para CAN BUS, el cual convierte las señales RS232 procedentes del Ordenador en señales de CAN BUS que se envían a las UCEs o Centralitas del coche y también convierte las señales CAN BUS procedentes de las UCEs o Centralitas del coche en señales RS232 para enviarlas al Ordenador para que sean procesadas por el Software o Programa Informático del Ordenador y que viene a ser la verdadera herramienta de diagnóstico.

En la figura de abajo se muestra la realización práctica del esquema anterior, o lo que es lo mismo, un circuito electrónico o hardware de un Equipo de Diagnóstico Profesional para Talleres Mecánicos.



3.4.- ADAPTADOR PARA DIAGNOSIS CON TELEFONO MOVIL. En la imagen inferior presentamos un Circuito Electrónico ó Hardware conocido como **Adaptador** que también se conecta a la toma de Diagnosis OBD de los coches y es mucho más simple que el Circuito Electrónico de los Equipos de Diagnosis Profesionales antes expuesto y que es el que se utiliza para realizar la diagnosis con teléfonos móviles que tiene que funcionar en Bluetooth para que se pueda conectar con dichos Teléfonos.

Un **Adaptador de calidad** como el de la figura de más abajo si funciona con una **App o Software** instalado en un Teléfono Móvil con sistema operativo Android viene a ser como un Equipo de Diagnosis reducido (**Adaptador + Software o App**) y que por desconocimiento o error la mayoría de las personas llaman **Equipo de Diagnosis** al

Adaptador cuando realmente es un **intermediario o puente** entre el Teléfono Móvil o Smartphone y las UCEs o Centralitas del coche.



Dentro de estos Adaptadores los hay **SIMPLES** y **COMPLEJOS**, de manera, que un buen Adaptador debe ser capaz de **filtrar mensajes** y evitar que los **bits** que reciba sufran alteraciones que pueden ser por ejemplo el cambio de un **0 por un 1** en una cadena de bits tal como: **000110010000001000001100** en donde al cambiar un **0 por un 1** como se puede ver en la cadena siguiente **000111010000001000001100** que en la posición **6** se cambió un **0 por un 1** el mensaje que envía el Adaptador a la UCE va a ser totalmente distinto del que recibió y por tal motivo con toda seguridad este simple cambio de un **0 por un 1** puede deteriorar la UCE o Centralita Electrónica del coche.

Mientras que los datos o mensajes que envía el Ordenador, Teléfono Móvil o UCE son secuencias de bits de **ceros y unos** (0 y 1), en realidad en la programación se utilizan en su lugar notación **hexadecimal** que para el caso de la cadena anterior es **19020C** y que

realmente es una cadena para lectura de averías en protocolo **CAN ODX**, mientras que la segunda cadena en la que se varió un 0 por un 1 es **1D020C**, que no se parece en nada a la cadena anterior, porque la función que debe realizar la UCE está indicada por el primer byte que tenía que ser **19** y sin embargo se envió **1D**.

Para que se entienda la importancia de la fiabilidad de los **Circuitos Electrónicos** de los Equipos de Diagnóstico y de los **Adaptadores Bluetooth** para la Diagnóstico con Teléfonos Móviles, más adelante se explicará con detalle lo que significa cada byte como por ejemplo el **19** y otros más que se utilizan en la Diagnóstico.

4.- LA CALIDAD CIENTIFICA DEL SOFTWARE

4.1.- DESARROLLO DEL SOFTWARE PARA LA DIAGNOSIS. El **Software o Programa Informático** es el **elemento fundamental** del Equipo de Diagnóstico, dado que es el que ordena iniciar las comunicaciones con la UCE, finalizar dichas comunicaciones y descifrar los datos e instrucciones que se envían a la UCE para leer las averías, leer valores, activar actuadores, etc., además de interpretar las respuestas de la UCE e informar al operario para que realice las operaciones pertinentes.

El Software debe ser capaz de implementar los protocolos OBD y los sistemas de las Marcas para comunicarse con las UCEs o Centralitas de los coches. Su misión consiste en enviar los mensajes a las UCEs según el sistema que tengan, recibir las respuestas de las UCEs, interpretarlas y mostrar al usuario en la pantalla del ordenador ó teléfono móvil las respuestas de forma comprensible para las personas.

4.2.- CONVERSIONES CIENTIFICAS DEL SOFTWARE. Los datos que muestran los equipos de diagnóstico en pantalla **NO son los mismos** que devuelven las Centralitas o UCEs ya que éstas responden con una serie de datos o **CADENAS en Hexadecimal** y el

software del equipo de diagnosis lo que hace es transformar esos datos con fórmulas para que las personas entiendan los resultados que se ofrecen.

Los valores en Hexadecimal que se muestran en la imagen son los que devuelven las UCEs y el Software los tiene que **convertir** en valores que comprendan los mecánicos y para ello tiene que dividir, multiplicar, sumar, etc. esos valores con unas **fórmulas concretas** para obtener datos o **valores de medición** que entiendan las personas.

Variando el Contador = existe comunicacion: UCE<=>PC

ProtocoloUCE

Len Count Tipo Fin

Long Contador Bloque

Leer Estados y Parametros BMW

La UCE devuelve valores en hexadecimal y el scanner los transforma en decimal "UCE:Valor deci.", pero los valores finales o "Valor medicion" son el resultado de multiplicar, dividir, etc. los valores UCE en hex. Si se observa que algun valor no se corresponde con el "titulo asignado" y pertenece a otro debido a las infinitas variantes de UCEs, sugerimos nos envíe los datos para actualizar el scanner

MAGNITUDES DE LA UCE A EXAMINAR	VALORES MEDICION	Hexa	Deci:UCE
Tension bateria	14.12 V	1663	5731
Nº rpm	780 rpm	185F	6239
Presion real ComRail	3116.77 bar	27EF	10223
Presion teorica ComRail	3076.83 bar	276C	10092
Masa aire real	2756.5 mg	9588	38280
Masa aire teorico	2816.5 mg	9600	38400
Cantidad rectific. para cil. 1	-1.01 mg	7EB5	32437
Cantidad rectific. para cil. 2	1.84 mg	825D	33373
Cantidad rectific. para cil. 3	0.17 mg	8039	32825
Cantidad rectific. para cil. 4	-1.01 mg	7EB5	32437
Cantidad inyeccion principal	5.96 mg	87A3	34723
Valor real presion sobrealimentacion	1001.88 mbar	1F4F	8015
Valor teorico presion sobrealimentacion	991.88 mbar	1EFF	7935
Presion atmosferica	993.88 mbar	1F0F	7951
Temperatura refrigerante	58.62 °C	195C	6492
Temperatura aire aspiracion	47.19 °C	1680	5760

En este caso concreto, el valor que se muestra al mecánico en la pantalla del escáner es de **14.12 V** pero la Centralita devolvió el valor **1663 en Hexadecimal** que en decimal equivale a **5731**.

El valor de **780 rpm** que es lo que entiende el mecánico, realmente la Centralita ha devuelto **185F** en Hexadecimal y esto en decimal equivale a **6239** y que como se ve **no se parecen en nada** a los valores que muestran en pantalla los equipos de diagnosis

En el resto de valores pasa lo mismo y esto quiere decir que el software de los Equipos de Diagnosis esta desarrollado para que los valores que devuelvan las Centralitas en **Hexadecimal** los multiplique, divida, sume, reste, etc. a través de una **fórmula para cada valor** con el objetivo de que los mecánicos comprendan esos datos.

DIAGNOSIS: FUNDAMENTOS

5.- LA METODOLOGIA DE LA DIAGNOSIS

5.1.- EL PROCESO DE COMUNICACIÓN CON LAS UCEs. Primero debe mantenerse la comunicación entre Equipo de Diagnosis y UCE del coche y en algunos casos esto se consigue enviando y recibiendo paquetes de bits a 50 milisegundos, en otros casos a 100 milisegundos o 250 milisegundos, etc.

A continuación se envía el código ó dirección de la UCE a chequear (01, 15, 17, etc..), dependiendo del vehículo y del protocolo de comunicaciones que utiliza la UCE y una vez establecida la comunicación entre el Equipo de Diagnosis y la UCE se envía desde el Equipo de Diagnosis lo que se desea chequear: **Diagnostico de Averías, Leer bloque valores, etc..** enviándose cada orden dentro de la cadena correspondiente.

Una vez que el software se conecta con la UCE, en algunos protocolos se recibe directamente la identificación de la UCE, la clave de la pieza de recambios, datos técnicos básicos del motor, clave del fabricante, etc. y a partir de aquí se solicita a la UCE que nos

devuelva las averías registradas y, de esta manera, la UCE devolverá la información en Binario que por ejemplo puede ser la siguiente:

0000001000000110 lo que en Hexadecimal es **0206** y en Decimal **00518**

0000001000101011 lo que en Hexadecimal es **022B** y en Decimal **00555**

5.2.- CODIGOS ENVIADOS A LAS UCES SEGÚN PROTOCOLO ISO14230. En la siguiente tabla se muestra el significado de los códigos que se envían a las UCEs haciendo la petición de lo que se desee analizar, que puede ser: **Leer averías, borrar averías, leer parámetros, etc.**

Codigo (hex)	Significado
10	Iniciar Sesión Diagnósis
11	Reset UCE
12	Leer datos almacenados en UCE
13	Leer Averías
14	Borrar Averías
17	Leer Estado de Averías
18	Leer Averías por Estado
20	Parar Sesión Diagnósis
21	Leer Datos por Identificación Local
22	Leer Datos por Identificación habitual
3E	Tester presente
81	Iniciar Comunicación
82	Terminar Comunicación

5.3.- INTERPRETACION CADENAS COMUNICACIÓN CON LAS UCES. El Software del Equipo de Diagnósis o de la App envía varias cadenas para leer las averías que tiene

almacenadas la UCE y si la UCE comprende la cadena que ha recibido, la respuesta debe comenzar con la suma del **primer byte** de la cadena enviada y **40** (en hexadecimal).

Si la UCE o Centralita **NO comprende** la cadena que ha recibido, la respuesta va a comenzar por **7F**.

En este caso el Software envió las siguientes cadenas: **222000** que hace la petición de “Leer Datos por Identificación habitual”. También envió las cadenas **1802FFFF**, **1802FF00**, **1800FF00** y **18000000** que significan “Leer Averías por Estado” y para finalizar envía la cadena: **17FF00** que indica “Leer Estado de Averías”, y esto es lo que hace el Software cuando se trata de **leer averías**.

Cadenas enviadas a la UCE	Respuestas de Cadenas recibidas de la UCE	Ver Test
222000 =>	B8F11204622000001D	
1802FFFF =>	B8F1121158034521123FB1E23F30E16A	
1802FF00 =>	B8F112037F18122D	
1800FF00 =>	B8F112037F18122D	
18000000 =>	B8F112037F18122D	
17FF00 =>	B8F112037F173101	

La 2ª respuesta de la UCE (**B8F1121158034521123FB1E23F30E16A**) contiene **3 averías** cuya **interpretación** es:

B8 = Byte inicial que forma parte del protocolo para esta UCE

F1 = Identificador del Tester o elemento que realiza la diagnosis

12 = Identificador de la UCE, equivale dirección de la UCE, en este caso 12 es Motor

11 = Longitud o cantidad de bytes que contiene la respuesta

58 = Respuesta que indica que la UCE entiende la pregunta, igual a 18+40

03 = Cantidad de averías, en este caso tiene **3 averías**

452112 = Identifica la **1ª avería** cuyo código es 4521 y el tipo es **12 (avería real)**

3FB1E2 = Identifica la **2ª avería** cuyo código es 3FB1 y el tipo es **E2 (esporádica)**

5F30E1 = Identifica la **3ª avería** cuyo código es 5F30 y el tipo es **E1 (esporádica)**

6A = Dígito de **control ó checksum** que indica **final cadena** y permite comprobar que la cadena está bien recibida y no ha habido interferencias en la comunicación

5.4.- EXPLICACION DE LOS TIPOS DE AVERIAS. En la cadena con averías devuelta por la UCE, después de cada código de avería figura un byte que indica el tipo de avería, es decir, indica si es permanente o esporádica y la causa.

En la tabla de la página siguiente se muestran ejemplos de tipos de averías que utiliza un fabricante de coches. Estas averías pueden ser **Permanentes o Esporádicas** cuya diferencia consiste en que la **permanente** se refiere a que un **componente lo hay que reparar** y la **esporádica** son averías que cogen los coches modernos y que deben borrarse para que no lleguen a ser permanentes.

Si es que se borran cada mes las averías en los coches se va gastar mucho menos en reparaciones y si no se hacen estas comprobaciones cualquier avería que en principio apareció como esporádica se puede convertir en permanente, pudiendo ser la causa de una avería grave en el coche que en caso de hacer un mantenimiento adecuado al mismo se puede conseguir que éste dure muchos más años que un coche al que no se le hizo ningún tipo de mantenimiento.

5.5.- AVERIAS PERMANENTES Y ESPORADICAS. En esta tabla indicamos los bytes de las averías **permanentes** y de las **esporádicas**, de tal manera, que si un coche tiene una avería **esporádica** y se borra va a circular en mejores condiciones. Las averías

permanentes aunque se borren, vuelven a aparecer porque afectan a algun componente que esté total o parcialmente deteriorado. Un cable cortado. Una mala conexión, etc.

Permanente	Esporádica	
01	81	Señal en positivo
02	82	Señal en positivo
03	83	Sin señal
04	84	Avería mecánica
06	86	Señal muy alta
07	87	Señal muy baja
08	88	Limites de regulación sobrepasados
10	90	Señal fuera de la tolerancia
12	92	Valor límite superior
13	93	Valor límite inferior
14	94	Error en los ajustes básicos
1A	9ª	Salida abierta
1B	9B	Señal no plausible
1C	9C	Cortocircuito a positivo
1D	9D	Cortocircuito a masa
1E	9E	Abierto o cortocircuito a positivo
1F	9F	Abierto o cortocircuito a masa
20	A0	Resistencia superior
21	A1	Resistencia inferior
24	A4	Conexión abierta
27	A7	Circuito defectuoso
2C	AC	Cortocircuito

5.6- LECTURA DE AVERIAS. El programa informático o Software que es el VERDADERO **Equipo de Diagnosis** envía varias cadenas a la UCE (por ej: **17FF00**, **1802FFFF**, **18000000**..) según el protocolo que tenga la UCE o Centralita del coche que en el caso de las Marcas van a tener más de **60.000** sistemas distintos y sin embargo en OBD solo

existen **6 protocolos**, hay que seleccionar el sistema concreto que se pretende chequear y se le envían las cadenas para que según comiencen por: **13, 17, 18, 21, 22**, etc. se le pide a la UCE que devuelva la petición que se hizo.

En el cuadrado de abajo se le envía a la UCE la 2ª cadena **1802FFFF** que le pregunta **18** = “**Leer Averías por Estado**” y la UCE tiene que responder de manera que la cadena 2ª completa **1802FFFF** la hay que separar por bytes tales como: **18 02 FF FF**, en lo que cada **byte** significa:

18 (primer byte) = **Leer Averías por Estado**

02 (segundo byte) = **Identificador del Status** (lugar almacenamiento avería)

FF (Tercer byte) = Byte **High** (alto)

FF (Cuarto byte) = Byte **Low** (bajo)

Cadenas enviadas a la UCE	Respuestas de Cadenas recibidas de la UCE
17FF00	8EF110570404802713360713400713380791
1802FFFF	B8F1120858024B101145019295
18000000	85F11158011502A097
1800FF00	82F1105800DB
1802FF00	85F3755801977360B0
1802FF00	03F1117F18122E
Prueba con otras cadenas	Prueba con byte 7F = orden NO reconocida por la UCE n Reconocida = 1º byte despues protocolo enviado

La UCE ha devuelto **5 cadenas con averías** en donde vamos a explicar el contenido de la primera cadena que es: **8EF110570404802713360713400713380791**

Vamos a desglosar la 1ª cadena: **8EF110570404802713360713400713380791** en donde explicamos cada conjunto de bytes y lo que representan.

***8EF110** (primer conjunto de bytes) = depende del protocolo y pueden ser 3 bytes, 4 bytes y que en este caso son 3 bytes: **8E** = byte del protocolo que incluye la longitud total de la

cadena (longitud = 0E = 14 bytes) contados a partir del 57 al 07 ya que el 91 es otra cosa;
F1 = Identificación del Tester que viene a ser la Máquina de Diagnóstico o la App de Diagnóstico; **10** = Identificación de la UCE que a su vez representa la dirección de la misma que en este caso es una UCE de Motor

***57** (byte indicador de respuesta correcta para la pregunta **17** ya que $17 + 40 = 57$) y esto es por lo siguiente: la **respuesta** de la UCE sale de sumar en todos los casos para los Protocolos ISO14230 el número **40** al código que se envíe que en este caso fue el **17** ya que la **cadena enviada** fue la **17FF00** y en la recibida **8EF11057** y en donde como se explicado el primer conjunto de bytes representa el protocolo.

Para el código **18** la respuesta correcta será $18 + 40 = 58$, y se obtiene de la misma manera que para el código **17**.

***04** (byte indicador del nº de averías) = **4 averías**

***048027** que viene a ser la primera avería que se desglosa en: **0480** que es el **código** de la avería y **27** es su status o **tipo** de avería, siendo según códigos que en la normativa OBD es **P0480** y que indica: “**Ventilador refrigerante motor**” y el **27** quiere decir que el “**circuito está defectuoso**”, tal como se ha explicado anteriormente

***133607** que es la segunda avería, al igual que antes será: **1336** = “**Sensor árbol levas**” y **07** = “**rango de funcionamiento**”. ***134007** y ***133807** siguen los mismos criterios.

***91** (último byte) = checksum ó CRC = Control de Redundancia cíclica ó mecanismo de detección de errores en sistemas digitales

5.7.- BORRADO DE AVERIAS. Para borrar averías se envía la cadena **14FF00** que significa **14** = “**Borrar Averías**”, y para el código **14** la respuesta correcta debe ser la suma de: $14 + 40 = 54$.

Si la UCE **no reconoce** el código **14** se recibe **7F14xx** en donde **xx** puede ser: **10**, **11**, **12**, **21**, **22**, etc. en donde **7F1411** = “Modo no Soportado”; **7F1412** = “SubFunción no Soportada o formato no valido”, etc.

5.8.- RESPUESTAS RECIBIDAS UCE SEGÚN PROTOCOLO ISO14230. Cuando la UCE devuelve **7F** porque no ha reconocido la cadena enviada, la respuesta va a estar formada por **3 bytes**, **7F** que indica que no reconoce la cadena o no ha podido realizar la tarea indicada por la misma, el segundo byte devuelve el **código** o byte enviado y el **tercer byte** indica las causas por las que no se pudo realizar la función indicada por la cadena enviada. La tabla siguiente muestra algunas respuestas indicando el fallo:

Código (hex)	Significado
10	Denegación General
11	Modo no Soportado
12	SubFunción no Soportada o formato no válido
21	Ocupado - Repetir Solicitud
22	Condiciones No Correctas o error en secuencia petición
23	Rutina no completa o Servicio en progreso
31	Solicitud fuera de rango
77	Error de Checksum del Bloque de transferencia de datos
78	Mensaje bloque recibido correctamente-Pendiente Respuesta
7F	Respuesta Negativa
80	Fuera tiempo transmisión o No soportado en diagnostico activo
C1	respuesta inicio comunicaciones
C2	respuesta fin comunicaciones

5.9.- LEER DATOS DE LA UCE. El Software envía varias cadenas a la UCE, por ejemplo: 2101, 2102, 2103, etc. según su protocolo. Y si cogemos la 1ª cadena **2101** y la separamos en 2 bytes (**21 01**), cada byte significa: **21** (primer byte) = “**Leer Datos por Identificación Local**”, **01** (segundo byte) = “**grupo de valores a leer**”. Si el Equipo de Diagnóstico envía a una UCE la cadena anterior **2101**, la UCE puede responder con la siguiente cadena:

80F111466101000000000000000000001800420000A2EF6BA72F6400
003110001000000000000031000000000000008000B958A5594D34001D73620000321940
000000002D20000055FB00F0637D en donde hay que localizar el byte **61** que es el resultado de sumar **21 + 40 = 61** cuyo siguiente byte **01** es el enviado como pregunta y que fue el **2101**, que se corresponde con el **6101**.

La anterior cadena en hexadecimal (desglosada) que devolvió la UCE indica:

80F11146 (primer conjunto de bytes) = depende del protocolo de cada UCE.
6101 que es la respuesta correcta para la pregunta **2101** ya que **21 + 40 = 61**
000000000000000000001800420000A2EF6BA72F640003110001000000000000310000
00000000008000B958A5594D34001D73620000321940000000002D20000055FB0
0F063 que es la cadena que analizando cada byte: **18, 42**, etc., y su rango de variación, se puede averiguar si se trata de una **tensión, temperatura, rpm, velocidad, porcentaje, presión**, etc. que para su interpretación hay que **sumar, multiplicar y/o dividir** por unos números para obtener el resultado en unidades reconocidas por los mecánicos. Por ejemplo, el byte **31** en hexadecimal que se corresponde con el **49** en decimal que es la “**Tensión de la Batería**”, se divide por **4** para obtener el valor en Voltios = **12.25V**. El byte **2F** en hexadecimal que se corresponde con el **47** en decimal va a ser la “**Tensión del Sensor presión atmosférica**” que al dividirlo por **10** se obtiene el valor en Voltios = **4.7V**

El **7D** que es el ultimo byte = checksum ó CRC = es el mecanismo detección errores.

Si la respuesta de la UCE es **7F21xx** que indica que “**No se reconoce la pregunta**” porque devolvió **7F** es necesario probar con otras preguntas tales como: **2102, 2103, 2104, 2105**, etc. hasta **21FF** para obtener una cadena de datos válida de la UCE.

5.10.- ESTRUCTURA CODIGOS AVERIAS OBD II SEGÚN NORMATIVA. La normativa OBD II es muy sencilla de implementar describiendo una serie de códigos de avería que

devuelven las UCEs para conocer cuál es el componente del motor que está causando el fallo y se rige los códigos de avería OBD II con la siguiente nomenclatura:

El **PRIMER** dígito comienza por **P** seguido de 4 números y se refieren a averías de motor o transmisión. Si el primer dígito comienza por **B** seguido de 4 números se refiere a averías de Interior, ABS. Si el primer dígito comienza por **C** seguido de 4 números se refiere a averías de chasis, carrocería, airbag. Si el primer dígito comienza por **U** seguido de 4 números se refiere a averías de otros sistemas, bus de comunicaciones.

El **SEGUNDO** dígito si es **0** indica códigos definidos por la normativa y si es **1** se refiere a códigos específicos de los fabricantes

El **TERCER** dígito indica: **Px1xx** medición de aire y combustible, **Px2xx** medición de aire y combustible, **Px3xx** Sistema encendido, **Px4xx** Control emisiones adicional, **Px5xx** Velocidad y regulación ralentí, **Px6xx** UCE y señales de salida, **Px7xx** Transmisión, **Px8xx** Transmisión, **Px9xx** Módulos de control, señales de entrada y salida

El **CUARTO** y **QUINTO** dígito identifican componentes y sistemas individuales

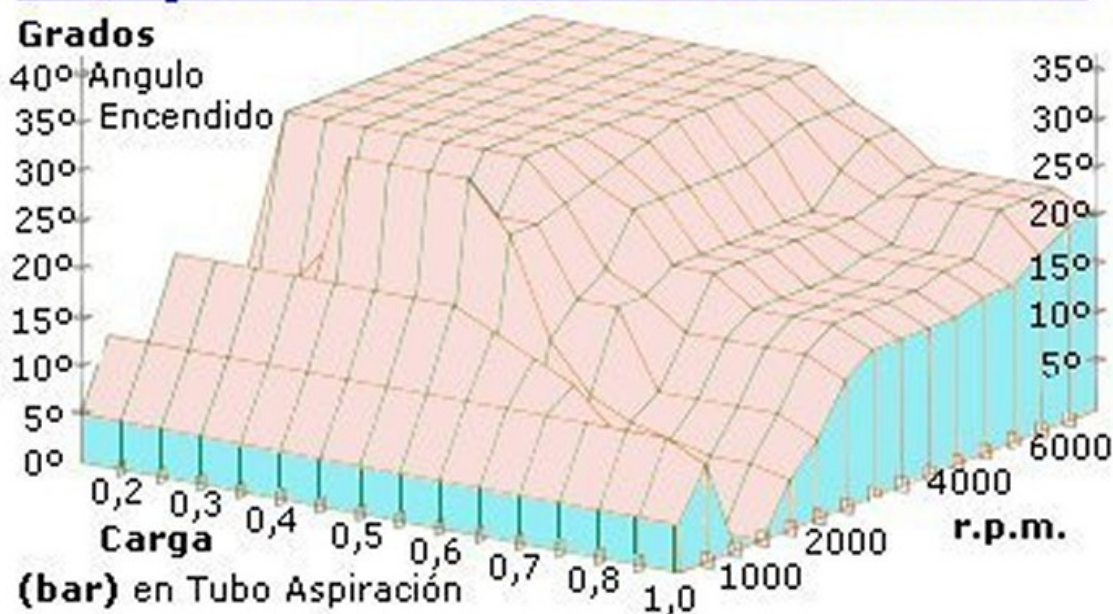
MANIPULACION DE UCEs

6.- VARIACION DEL SOFTWARE ORIGINAL

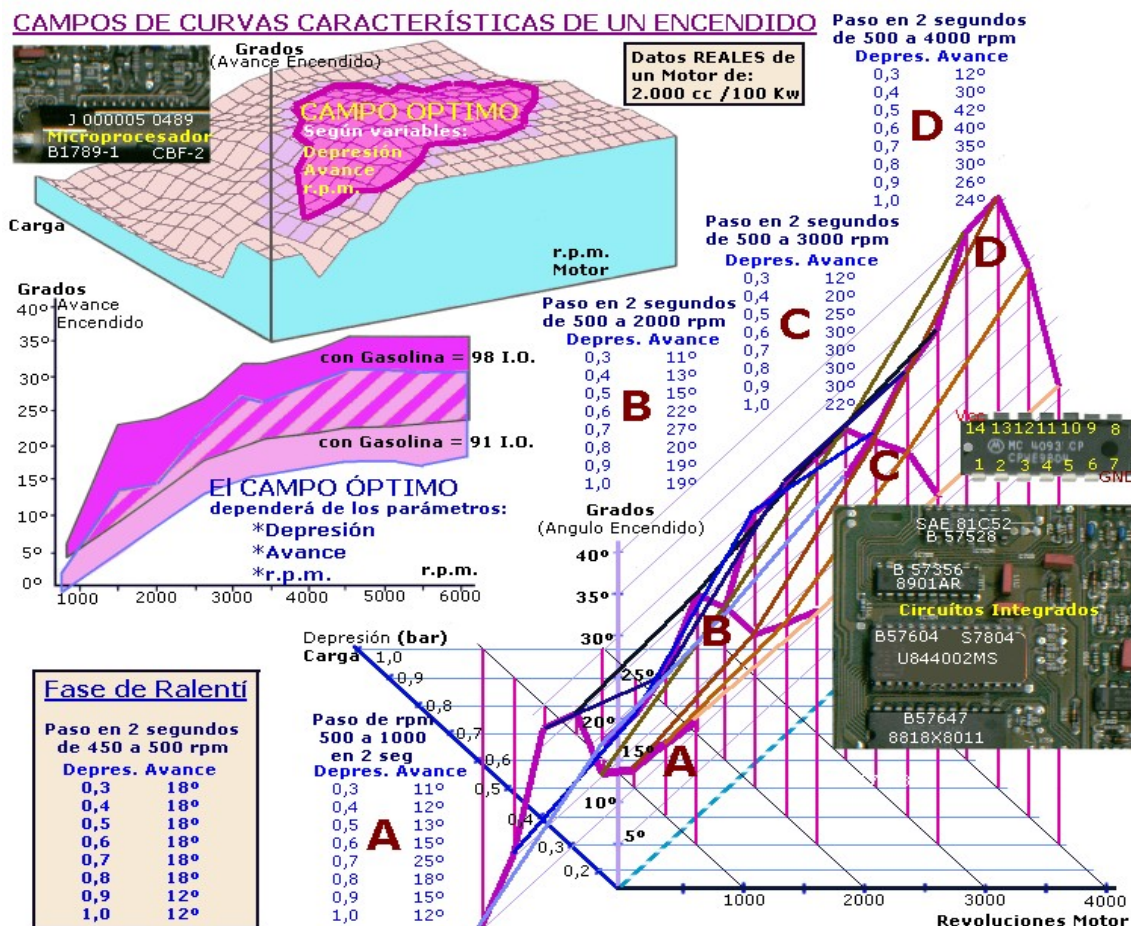
6.1.- VENTAJAS E INCONVENIENTES. Generalmente a cada coche el fabricante le asigna unos parámetros de funcionamiento para que el vehículo ofrezca el mayor rendimiento posible con el mínimo consumo y en muchas ocasiones los usuarios solicitan de intermediarios para variar los parámetros originales del coche introduciéndole nuevos valores lo cual en la mayoría de las ocasiones va a ser perjudicial para el vehículo ya que se puede reducir notablemente su vida útil.

6.2.- SOFTWARE UCE Y CURVAS CARACTERISTICAS. Las Curvas se obtuvieron de un Motor con 31.280 kms de uso y los VALORES de **Carga, r.p.m. y ángulo de encendido** se plasmaron en el GRÁFICO llevándose los datos extraídos sobre los ejes: **Carga; r.p.m.; Angulo.**

Campo de Curvas Características



6.3.- MANIPULACION DEL SOFTWARE DE LAS UCES. En este caso un motor con un Campo de Curvas Características original del fabricante se modificó para conseguir más potencia, lo cual fue necesario variar ciertos datos del Software que generalmente se desconocen pero, visto en una figura puede parecer muy bonita y sin embargo la realidad es totalmente distinta porque se va a solicitar una potencia excesiva que en su diseño no se ha tenido en cuenta y por tanto se va a reducir su vida útil.



La conclusión que podemos sacar es que el **máximo RENDIMIENTO** de un motor va a depender del Avance; Carga y r.p.m.; recordando que el **avance** depende de las r.p.m. y **depresión** o PRESIÓN de entrada de Aire a los Cilindros y el **CAMPO ÓPTIMO** de funcionamiento de un motor va a **coincidir** con el mínimo gasto de combustible y máximas prestaciones.

6.4.- LA REALIDAD DESCONOCIDA. Las figuras anteriores son muy llamativas y para las personas que desconocen el diseño de los motores les puede gustar mucho y aconsejar la manipulación del Software de las UCEs ya que consiguen prestaciones distintas, pero lo que desconocen estas personas es que la **manipulación del Software** no se ciñe a hacer

Dirección	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	ASCII
053300	00	00	00	00	64	00	00	00	00	00	00	00	64	00	00	00d.....d...
053310	00	00	00	00	64	00	00	00	00	00	00	00	64	00	00	00d.....d...
053320	00	00	00	00	64	00	00	00	00	00	00	00	64	00	00	00d.....d...
053330	C9	2F	C7	2C	27	3F	E8	3F	9B	76	ED	51	10	27	11	27	É/Ç,'?è?■vÍQ.'.'
053340	D1	33	7D	79	8D	A0	7E	79	8E	A0	7F	79	8F	A0	80	79	Ń3}y■ ~y■ ■y■ ■y
053350	90	A0	81	79	91	A0	E1	79	F1	A0	E2	79	F2	A0	E3	79	■ ■y' áyñ âyò äy
053360	F3	A0	E4	79	F4	A0	E5	79	F5	A0	30	34	35	39	30	36	ó äyô äyõ 045906
053370	30	31	39	47	20	20	31	2C	34	6C	20	52	33	20	45	44	019G 1,41 R3 ED
053380	43	20	20	53	47	20	20	31	32	38	31	20	32	38	53	41	C SG 1281 28SA
053390	34	38	38	32	20	30	32	38	31	30	31	30	32	32	30	20	4882 0281010220
0533A0	46	44	35	48	47	38	30	30	20	20	20	30	34	35	39	30	FD5HG800 04590
0533B0	36	30	31	39	47	20	20	30	39	2F	30	30	00	00	C8	00	6019G 09/00..È.
0533C0	14	00	78	05	E8	03	19	1A	4C	04	E8	03	85	86	78	05	..x.è...L.è.■x.
0533D0	E8	03	87	88	FB	FF	FF	FF	FF	FF	FB	FF	FF	FF	FF	F7	è.■üÿÿÿÿÿÿÿÿÿ÷
0533E0	FF	FF	FF	FF	01	C3	93	2D	81	42	01	C3	FF	FF	FF	FF	ÿÿÿÿ.ñ■-■B.ñÿÿÿÿ
0533F0	31	00	00	00	E0	01	00	00	00	10	00	00	73	00	33	FF	1...à.....s.3ÿ
053400	5E	04	33	FF	88	00	D4	FE	6A	FF	53	02	4C	FA	53	02	^.3ÿ■.ôþjÿS.Lús.
053410	88	00	D4	FE	31	00	00	00	E0	01	00	00	67	00	00	00	■.ôþ1...à...g...
053420	1A	00	00	00	FF	00	00	00	FA	00	00	00	0A	00	00	00ÿ...ú.....
053430	64	00	00	00	4E	00	00	00	6A	FF	53	02	4C	FA	53	02	d...N...jÿS.Lús.

INNOVACION EN DIAGNOSIS

7.- COCHES CONECTADOS Y DIAGNOSIS ACTUAL

7.1.- PUESTA EN PRÁCTICA DE ESTE PROYECTO. Partiendo de la innovación descrita en este estudio, hoy en día se puede observar la Diagnósis que realice a un coche una persona en **Alemania, México, USA, Italia, España, Brasil, Argentina, Rusia, Portugal, Chile, Polonia, Egipto**, etc. y orientar la reparación a distancia, de manera que supone este proyecto un avance cuantitativo y cualitativo sobre la actual tecnología que cada vez se impone más en los automóviles.



En la figura se representa como se producen las comunicaciones con un Teléfono Móvil sobre un coche. Pues se envía la información a Internet automáticamente y a su vez, se puede visionar el lugar exacto en donde se está realizando la diagnóstico, no solo en un

mapa con la ubicación del coche y para ello sólo es necesario conectar un **dispositivo-Adaptador bluetooth** en su toma de diagnosis.

En esta imagen se pueden ver distintos países en donde se realizó la diagnosis a unos coches concretos y también se ven las averías que cada coche tiene, el día y hora en que se realizó la diagnosis, el fallo que presenta el coche y pulsando en “Ver Talleres cercanos ubicación coche” se ve el lugar exacto en donde se está realizando dicha diagnosis cuya imagen se muestra a continuación de ésta.

6.- AVERIAS DEL COCHE:

Marca: **AUDI** UCE: **Motor diesel** Sistema: **OBDModernos**

Fecha: **5/14/2019** Hora: **9:59:37 AM**

P0471 Sensor de presión de los gases de escape - problema de funcionamiento/rango |

Ubicación: **Ruta Del Colesterol, Unnamed Road, 33939, Langreo, Asturias, España**

[Ver Talleres cercanos ubicación coche](#)

7.- AVERIAS DEL COCHE:

Marca: UCE: Sistema:

Fecha: **14/5/2019** Hora: **08:53:46**

P3500 Código específico fabricante

Ubicación: **Via Torre delle Palme, 10b, 90136 Palermo PA, Italia**

[Ver Talleres cercanos ubicación coche](#)

8.- AVERIAS DEL COCHE:

Marca: **RENAULT** UCE: **Motor OBD** Sistema: **OBDAntiguos**

Fecha: **14.05.2019** Hora: **07:46:41**

P0089 Regulador de presión de combustible 1 - problema de funcionamiento |

Ubicación: **Landauer Str. 4, 68766 Hockenheim, Niemcy**

[Ver Talleres cercanos ubicación coche](#)

9.- AVERIAS DEL COCHE:

Marca: UCE: Sistema:

Fecha: **13/5/2019** Hora: **21:40:54**

P0262 Inyector 1 - señal alta |

Ubicación: **Camila Quiroga 904, B1852GQS Burzaco, Buenos Aires, Argentina**

[Ver Talleres cercanos ubicación coche](#)

12.- AVERIAS DEL COCHE:

Marca: UCE: Sistema:

Fecha: **Monday, May 13, 2019** Hora: **22:49:57**

P3500 Código específico fabricante

Ubicación: **A-381, Boldumsaz, Туркменистан**

[Ver Talleres cercanos ubicación coche](#)

13.- AVERIAS DEL COCHE:

Marca: **MERCEDES** UCE: **Airbag** Sistema: **AirbagArmada86E**

Fecha: **13.5.2019** Hora: **19:39:10**

9123 Llave bloqueo marcha código equivocado

Ubicación: **Elisabeth-Kadow-Straße 75, 41470 Neuss, Deutschland**

La diagnosis de un coche con un Teléfono Movil y las **Apps** se realiza de manera muy sencilla y la ventaja que tienen las Apps es que además de mostrar al instante en el **teléfono móvil** el resultado de la verificación de la UCE o Centralita del coche, también se envía esta información a la nube o internet para que desde **cualquier lugar del mundo** tanto el **taller** como el **propietario** del coche conozcan las **averías** que tiene el vehículo. Además de la información anterior, igualmente se puede ver el **lugar exacto** en donde se

está **realizando la diagnosis** al coche viéndose a su vez una **fotografía** de la ubicación del vehículo tal como se puede apreciar en la página siguiente de la presente divulgación.



Como resumen, se puede decir que la realización de la diagnosis a un coche se puede efectuar con un Equipo de **Diagnosis** Profesional en un Taller o bien con un **Adaptador** que suele costar **50 a 60 euros** y que con unas **Apps** específicas para Marcas de coches que se venden como **BonosXuga** a partir de **10 euros** y un BonoXuga permite realizar un **Diagnostico completo** de todas las UCEs con soluciones técnicas, va a suponer un cambio radical en el mantenimiento y cuidado de los coches porque si un vehículo funciona en óptimas condiciones va a ahorrar de **900 a 1.300 euros/año** en Combustible y Averías, además de reducir notablemente la **contaminación** del Medio Ambiente.

Solo falta indicar que la comunicación entre el **Teléfono** Movil y el **Adaptador** se realiza por **Bluetooth**. Por ello, la primera vez en que se realice la diagnosis, hay que conectar el Adaptador en la toma de diagnosis del coche. Se debe **activar el bluetooth en el Tfno Móvil** y, para ello, se pulsa **buscar dispositivos** para **vincular** el Adaptador al Teléfono Móvil. El nombre del adaptador en el Teléfono Movil va a aparecer como **OBDII** y la contraseña o password puede ser **0000** ó bien **1234**

ADAPTADORES

8.- MULTIPLES ADAPTADORES DE MALA CALIDAD

8.1.- CONFUSION PARA LOS USUARIOS. En Internet se venden múltiples Adaptadores de deficiente calidad en donde algunos de ellos solo funcionan con los Protocolos OBDII y el resto ni siquiera llegan a funcionar con dichos protocolos, aunque los vendedores publicitan que son elementos de diagnosis, cuando esto es totalmente falso, ya que en caso de que funcione en OBD solamente van a mostrar datos de presiones, revoluciones, temperaturas.



CIRCUITOS ELECTRÓNICOS

9.- CIRCUITOS VARIOS DE DISTINTOS ADAPTADORES

9.1.- CIRCUITOS DE ADAPTADORES QUE SE VENDEN EN INTERNET. Muchos de estos circuitos solo funcionan con programas sencillos de OBD y se venden como equipos de diagnosis, cuando realmente estos circuitos son de baja calidad, lo cual no hay que confundir con el circuito electrónico de Autoxuga que esta desarrollado basándose en el circuito de diagnosis para equipos profesionales, cuyo esquema electrónico figura más arriba.

Circuitos Electrónicos Adaptadores

