

ESCUELA POLITÉCNICA DEL EJÉRCITO

EXTENSIÓN LATACUNGA



CARRERA DE INGENIERÍA AUTOMOTRIZ

**“DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE
DIAGNÓSTICO PREDICTIVO PARA AUTOMOTORES BAJO
NORMA OBD II, CON CONEXIÓN INALÁMBRICA WI-FI Y
CONTROLADORES TÁCTILES DE LA MARCA APPLE”**

Tesis presentada como requisito previo a la obtención del grado de

INGENIERO AUTOMOTRIZ

JOHN JAIRO CEBALLOS MARCILLO

Latacunga-Ecuador

SEPTIEMBRE 2012

ESCUELA POLITÉCNICA DEL EJÉRCITO
CARRERA DE INGENIERÍA AUTOMOTRIZ
DECLARACIÓN DE RESPONSABILIDAD

Yo: John Jairo Ceballos Marcillo

DECLARO QUE:

El proyecto de grado titulado **“DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE DIAGNÓSTICO PREDICTIVO PARA AUTOMOTORES BAJO NORMA OBD II, CON CONEXIÓN INALÁMBRICA WI-FI Y CONTROLADORES TÁCTILES DE LA MARCA APPLE”**, fue desarrollado con la debida investigación científica, sabiendo respetar todos los derechos intelectuales de terceros, acorde con las citas que se hace al pie de página correspondiente, las fuentes se añaden a la bibliografía.

Por lo que se puede afirmar que este trabajo es de mi exclusiva autoría.

En honestidad de esta declaración, me responsabilizo de lo comprendido, la veracidad y el alcance científico que tiene este proyecto de grado realizado.

Latacunga, Septiembre del 2012.

John Jairo Ceballos Marcillo

CI: 180428119-2

ESCUELA POLITÉCNICA DEL EJÉRCITO
CARRERA DE INGENIERÍA AUTOMOTRIZ
CERTIFICADO

Ing. Germán Erazo (DIRECTOR)
Ing. Alexandra Corral (CODIRECTORA)

CERTIFICAN:

Que el trabajo denominado **“DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE DIAGNÓSTICO PREDICTIVO PARA AUTOMOTORES BAJO NORMA OBD II, CON CONEXIÓN INALÁMBRICA WI-FI Y CONTROLADORES TÁCTILES DE LA MARCA APPLE”**, ha sido guiado y revisado periódicamente y cumple con normas y estatutos establecidos, en el Reglamento de Estudiantes de la Escuela Politécnica del Ejército.

Siendo este un proyecto de excelente calidad y contenido científico que servirá para la enseñanza/aprendizaje y a la aplicación de conocimientos y al desarrollo profesional por lo que si recomendamos su publicación.

Latacunga, Septiembre del 2012

Ing. Germán Erazo.

DIRECTOR

Ing. Alexandra Corral.

CODIRECTOR

ESCUELA POLITÉCNICA DEL EJÉRCITO
CARRERA DE INGENIERÍA AUTOMOTRIZ
AUTORIZACIÓN

Yo: John Jairo Ceballos Marcillo,

Autorizamos a la Escuela Politécnica del Ejército para que publique en la biblioteca virtual de la institución el trabajo denominado **“DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE DIAGNÓSTICO PREDICTIVO PARA AUTOMOTORES BAJO NORMA OBD II, CON CONEXIÓN INALÁMBRICA WI-FI Y CONTROLADORES TÁCTILES DE LA MARCA APPLE”**, en el que se encuentra contenido, ideas y criterios que hemos desarrollado bajo nuestra exclusiva autoría.

Latacunga, Septiembre del 2012.

John Jairo Ceballos Marcillo

CI: 180428119-2

DEDICATORIA

El presente trabajo de grado lo dedico eternamente en primera instancia a los mejores padres que la vida me pudo haber dado, por su comprensión, apoyo incondicional, sus fuerzas a todo momento y porque sin ustedes y su fe en mi esfuerzo, mi vida estudiantil no sería tan exitosa como lo es. Agradezco igualmente a mi familia entera, Jessica, Jazmín, Santiago, Sebastián y Martín, por ser mi compañía y mi respaldo moral en todo momento. De todo corazón este trabajo es PARA USTEDES.

John Jairo Ceballos Marcillo

AGRADECIMIENTO

A Dios hermoso que me dio la vida para crecer y ser una mejor persona cada día, a mis padres Jairo Ceballos y Rosa Marcillo, mis dos ejes de dirección, sin ellos no se donde estaría en este momento.

A toda mi familia por darme siempre su apoyo incondicional, especialmente a mis hermanas que con su sabiduría siguieron el camino junto a mi en los momentos difíciles.

A todos mis maestros a lo largo de mi vida estudiantil, al igual que a todos mis compañeros y amigos que encontré en este arduo camino.

John Jairo Ceballos Marcillo.

RESUMEN

La introducción de sistemas táctiles de la marca Apple, ha contribuido con la mejora de los sistemas inalámbricos, con aplicaciones desarrolladas para actividades específicas de industrias especiales. En este caso, en la industria automotriz, permite mantener un sistema de información organizada y universal, así como nos permite mantener una base de datos determinada para cada unidad y marca.

El presente proyecto facilitará el estudio, análisis e interpretación de los diferentes sistemas eléctricos, electrónicos y mecánicos de los diferentes tipos y marcas de automóviles, que mantengan un sistema de comunicación OBD II. Los profesionales y estudiantes tendrían una herramienta táctil, con una interfaz gráfica muy amigable y fácil de manejar; con comunicación universal con los tipos específicos de conexión normados internacionalmente.

El esquema básico de conexión, está compuesto por una interfaz de lectura de códigos de tipo OBD II con generador de señal de emisión de datos Wi-Fi, un software de aplicación desarrollado en una interfaz única de usuario (NUI), un dispositivo táctil de la marca Apple (Iphone, Ipad), y una red establecida de registro de información periódica e individual para cada modelo y tipo de automotor analizado.

Para los docentes y técnicos especializados en los laboratorios de Mecánica de Patio y Prototipos, además de los especialistas en el área de Autotrónica, este sistema les permitiría, además de la comunicación directa con el automóvil, la interpretación de códigos DTC's, con su respectivo despliegue y borrado. La adición de un módulo de seguridad integral del dispositivo táctil, al mismo tiempo permitiría una investigación más exhaustiva en el área de seguridad automotriz.

ABSTRACT

The introduction of touch screen systems from Apple Inc., has contributed with the reach of a higher performance of wireless systems, with applications made for specific activities inside special industries. In this case, in the automotive industry, it let us keep an organized and universal information system, as it let us also keep a unique database for each unit and car brand.

This project help us in the study, analysis and interpretation of the different electric, electronic and mechanic systems of all the kinds and brands of automobiles, which were build with the OBD II communication system. Professionals and students would have a touch screen tool, with a friendly graphic interface easy to handle, including universal compatibility with most of the protocols standardized globally.

The basic connection scheme is composed of one OBD II code lecture interface with a built in WI-FI sign generator, an application developed into the native unity interface, a touch screen device from Apple Inc. (IPhone, Ipad), and a established net with periodic registration of information and that should be made individually for each model and kind of automobile analyzed.

To teachers and students specialized into the Mechanic and Prototype Labs, as well as the Autotronics Specialist, this entire system would help them in the direct communication with the automobile, the interpretation of the DTCs codes, with the respective analysis and cleaning. In addition, an integrated security module inside the Tablet would introduce them into a more deep investigation of the automotive security area.

ÍNDICE DE CONTENIDOS

CARÁTULA	I
DECLARACIÓN DE RESPONSABILIDAD	II
CERTIFICADO	III
AUTORIZACIÓN	IV
DEDICATORIA	V
AGRADECIMIENTO	VI
RESUMEN	VII
ABSTRACT	VIII
ÍNDICE DE CONTENIDOS	IX
ÍNDICE DE TABLAS	XIV
ÍNDICE DE FIGURAS	XVI
PRESENTACIÓN	XXIV
CAPÍTULO 1	1
EL PROBLEMA	1
1.1. PLANTEAMIENTO DEL PROBLEMA	1
1.2. FORMULACIÓN DEL PROBLEMA	2
1.3. OBJETIVO GENERAL	3
1.4. OBJETIVOS ESPECÍFICOS DEL PROYECTO	3
1.5. JUSTIFICACIÓN E IMPORTANCIA	4
CAPÍTULO 2	5
MARCO TEÓRICO	5
2.1. INTRODUCCIÓN A LOS SISTEMAS DE COMUNICACIÓN BASADOS EN REDES INALÁMBRICAS WI FI	5
2.1.1 SISTEMAS DE COMUNICACIÓN.....	5
a. VISIÓN TECNOLÓGICA.....	7
b. INTRODUCCIÓN A LOS SISTEMAS DE COMUNICACIÓN.....	8
i. DEFINICIONES.....	9
ii. NORMAS Y ESTÁNDARES.....	9
iii. LÍNEAS DE COMUNICACIÓN.....	10
iv. TIPOS DE TRANSMISIÓN.....	12
v. ELEMENTOS DE UN SISTEMA DE COMUNICACIÓN.....	15
c. ARQUITECTURAS Y MECANISMOS DISPONIBLES.....	16
i. COMUNICACIÓN DE INTERFACE AÉREA DE MEDIANO Y LARGO ALCANCE.....	18
ii. SISTEMA COOPERATIVO DE INFRAESTRUCTURA VEHICULAR.....	19
iii. INFRAESTRUCTURA DE INTEGRACIÓN VEHICULAR VANET.....	20

d.	TECNOLOGÍAS DE COMUNICACIÓN.....	22
i.	REDES WLAN.....	23
ii.	COMUNICACIONES DEDICADAS DE POCO ALCANCE..	27
iii.	REDES EN MALLA.....	28
iv.	GSM.....	28
v.	WIMAX.....	31
vi.	WI FI	32
e.	SEGURIDAD Y PRIVACIDAD.....	37
i.	RIESGOS DE LAS REDES INALÁMBRICAS.....	37
ii.	TIPOS DE ATAQUES.....	38
iii.	MECANISMOS DE SEGURIDAD.....	40
iv.	MEDIDAS DE SEGURIDAD RECOMENDADAS.....	41
2.1.2	DESARROLLO Y APLICACIONES.....	43
2.2	SISTEMAS TÁCTILES APPLE Y SOFTWARE APLICADO CON INTERFAZ ÚNICA DE USUARIO (NUI).....	44
2.2.1	SOFTWARE APLICADO NUI.....	44
a.	INTRODUCCIÓN Y DESARROLLO.....	45
b.	CONFIGURACIÓN.....	46
c.	IMPLEMENTACIÓN.....	48
2.2.2	PERSONALIZACIÓN DE DISPOSITIVOS.....	50
a.	DISPOSITIVOS TÁCTILES.....	50
b.	DISPOSITIVOS MULTITÁCTILES.....	54
c.	TABLETS.....	56
d.	IPENS.....	57
2.2.3	IMPLEMENTACION Y UTILIZACIÓN.....	58
a.	GIROSCOPIOS.....	58
b.	ACELERÓMETROS.....	59
2.2.4	APLICACIONES DEDICADAS DE INTERFAZ ÚNICAS DE USUARIO (IPA).....	61
2.3	DIAGNÓSTICO PREDICTIVO Y PREVENTIVO. FUNDAMENTOS MECÁNICOS, ELÉCTRICO Y ELECTRÓNICOS.....	63
2.3.1	INTRODUCCIÓN.....	63
a.	FUNDAMENTOS.....	64
b.	MECANISMOS DE ANÁLISIS PREDICTIVO.....	65
c.	PROCEDIMIENTO Y ANÁLISIS.....	68
2.3.2	ELÉCTRICA Y ELECTRÓNICA.....	70
a.	FUNDAMENTOS.....	70
b.	PROCEDIMIENTO Y ANÁLISIS.....	71
c.	RELACIÓN MECÁNICA – ELECTRÓNICA.....	75
2.3.3	NUEVAS TECNOLOGÍAS APLICABLES.....	77
2.3.4	APLICACIÓN EN MECÁNICA DE COMPETICIÓN.....	78
2.4	SISTEMA DE DIAGNÓSTICO A BORDO OBD II.....	81
2.4.1	INTRODUCCIÓN.....	81
2.4.2	FUNCIONES DEL OBD II.....	82
2.4.3	DETECCIÓN DE ERRORES CON OBD II.....	83
2.4.4	ACCESO A LA INFORMACIÓN DEL SISTEMA OBD II.....	84
a.	EL CONECTOR.....	84
b.	CÓDIGOS DE ERROR.....	85
c.	PROTOCOLOS DE COMUNICACIÓN.....	86
d.	LECTORES DE CÓDIGOS.....	87
2.4.5	INTERFACES DE CONEXIÓN.....	88
a.	ISO 9141-2.....	89
b.	SAE J1850 PWM.....	89
c.	SAE J1850 VPW.....	89
d.	ISO 9141.....	90
e.	ISO 14230.....	90

f. ISO 15765 (CAN).....	90
2.4.6 CÓDIGOS OBD II.....	91
CAPÍTULO 3.....	92
PLANTEAMIENTO DE HIPÓTESIS.....	92
3.1 HIPÓTESIS	92
3.1.1 HIPÓTESIS GENERAL	92
3.1.2 HIPÓTESIS ESPECÍFICAS	92
3.2 VARIABLES DE INVESTIGACIÓN.....	93
3.2.1 VARIABLE INDEPENDIENTE.....	93
3.2.2 VARIABLE DEPENDIENTE.....	93
3.2.3 OPERACIONALIZACIÓN DE VARIABLES.....	93
a. VARIABLE INDEPENDIENTE.....	93
b. VARIABLE DEPENDIENTE.....	94
CAPÍTULO 4.....	95
DISEÑO DEL SISTEMA INALÁMBRICO DE DIAGNÓSTICO	
PREDICTIVO TÁCTIL.....	95
4.1 DISEÑO DE REDES DE COMUNICACIÓN.....	95
4.1.1 CONECTIVIDAD WI-FI.....	96
4.1.2 RED PUNTO A PUNTO INTERFÁZ – MÓDULOS DE CONTROL.....	97
4.1.3 MANEJABILIDAD DE PROTOCOLOS DE COMUNICACIÓN	
BAJO NORMA OBD II.....	99
a. VALORES TIPO.....	100
i. OBDNET_PID_VALUE.....	100
ii. OBDNET_PID_INFO.....	101
b. CALLBACKS O PETICIONES DE RETORNO.....	102
i. OBDNET_CALLBACK_PID_LIST_ITEM.....	102
ii. OBDNET_CALLBACK_SELECTED_PID_ITEM.....	103
iii. OBDNET_CALLBACK_STREAMING_PID_VALUE.....	103
c. FUNCIONES.....	104
i. INITIALIZATION FUNCTIONS.....	104
ii. CREATION FUNCTIONS.....	106
iii. STATUS FUNCTIONS.....	107
iv. DEBUGGING FUNCTIONS.....	108
v. CONNECTION FUNCTIONS.....	109
vi. MISC. FUNCTIONS.....	111
vii. PARAMETER FUNCTIONS.....	111
viii. LOGGING FUNCTIONS.....	114
4.2 DIAGRAMACIÓN DE CONEXIONES.....	116
4.2.1 SISTEMA DE ADQUISICIÓN DE DATOS.....	116
4.2.2 SISTEMA DE INDICADORES GRÁFICOS.....	122
4.3 PERSONALIZACIÓN DE LOS SISTEMAS DE INTERFAZ GRÁFICA	
DE USUARIO.....	154
4.3.1 PERSONALIZACIÓN DEL MENU DE PRESENTACIÓN.....	157
4.3.2 PERSONALIZACIÓN DE LAS HERRAMIENTAS DE	
DIAGNÓSTICO.....	164
a. MEDIDORES DE COMPRESIÓN Y VACÍO.....	165
b. MEDIDORES DE PODER Y TORQUE DEL MOTOR.....	170
c. MEDIDORES DE TPS.....	179
d. MEDIDORES DE APP.....	183

e. MEDIDOR INTERACTIVO PREDICTIVO.....	183
f. MEDIDORES DE MAF, MAP Y CARGA COMPRESIÓN..	206
4.3.3 PERSONALIZACIÓN DE LOS ELEMENTOS DE MEDICIÓN Y TELEMETRÍA.....	210
4.3.4 PERSONALIZACIÓN DE LOS INDICADORES TÁCTILES GLCD	230
CAPÍTULO 5.....	236
CONSTRUCCIÓN Y PRUEBAS DE FUNCIONAMIENTO DEL SISTEMA DE DIAGNÓSTICO PREDICTIVO.....	236
5.1 CONSTRUCCION DEL SISTEMA DE RED.....	236
5.1.1 PARÁMETROS DE SEGURIDAD DE EQUIPOS INFORMÁTICOS	238
5.1.2 INTERCONEXIÓN DE DISPOSITIVOS DE RED.....	239
5.1.3 INSTALACIÓN Y CONEXIÓN DE DISPOSITIVOS GLCDS.....	240
5.1.4 MONTAJE DE DISPOSITIVO TÁCTIL APPLE.....	241
5.2 PRUEBAS DE VERIFICACIÓN DE CONECTORES Y CABLEADO.....	242
5.2.1 MÉTODOS DE VERIFICACIÓN.....	242
5.2.2 ANÁLISIS COMPROBACIÓN DE CIRCUITOS DE AUTOMÓVILES.	243
5.3 PRUEBAS DE FUNCIONAMIENTO.....	245
5.3.1 PRUEBAS DE COMUNICACIÓN Y CONECTIVIDAD.....	245
5.3.2 EXPLORACIÓN DE CÓDIGOS DE AVERÍA Y ESTADO DE VARIABLES PIDS.....	246
5.3.3 MEDICIÓN DE MAGNITUDES EN FUNCIONAMIENTO MECÁNICO Y ELECTRÓNICO NORMAL DE AUTOMÓVILES.....	248
5.4 VERIFICACIÓN Y TABULACIÓN DE DATOS.....	253
5.5 ANÁLISIS DE RESULTADOS.....	263
CAPÍTULO 6.....	268
MARCO ADMINISTRATIVO.....	268
6.1 RECURSOS	268
6.1.1 RECURSOS HUMANOS.....	268
6.1.2 RECURSOS TECNOLÓGICOS	269
6.1.3 RECURSOS MATERIALES	269
6.2 PRESUPUESTO.....	269
6.3 FINANCIAMIENTO.....	269
6.4 CRONOGRAMA.....	270
CONCLUSIONES.....	271
RECOMENDACIONES.....	272

BIBLIOGRAFÍA.....	273
ANEXO A: CODIGOS OBD II	276
ANEXO B: ELEMENTO PATH GLCD	302
ANEXO C: CRONOGRAMA.....	316
ANEXO D: ARTÍCULO DEL PROYECTO.....	318

ÍNDICE DE TABLAS

TABLA 3.1. Operacionalización de variable independiente.....	93
TABLA 3.2. Operacionalización de variables dependientes.....	94
TABLA 4.1. OBDNET_PID_VALUE.....	100
TABLA 4.2. OBDNET_PID_INFO.....	101
TABLA 4.3. OBDNET_CALLBACK_PID_LIST_ITEM.....	102
TABLA 4.4. OBDNET_CALLBACK_SELECTED_PID_ITEM.....	103
TABLA 4.5. OBDNET_CALLBACK_STRAMING_PID_VALUE.....	103
TABLA 4.6. OBDNetInitialize.....	105
TABLA 4.7. OBDNetUninitialize.....	106
TABLA 4.8. OBDNetCreate.....	106
TABLA 4.9. OBDNetDestroy.....	107
TABLA 4.10. OBDNetStatusGetErrorCodeString.....	108
TABLA 4.11. OBDNetOpenConnection.....	109
TABLA 4.12. OBDNetCloseConnection.....	110
TABLA 4.13. OBDNetIsConnected.....	110
TABLA 4.14. MISC. FUNTIONS.....	111
TABLA 4.15. OBDNetParameterGetCount.....	112
TABLA 4.16. OBDNetParameterGetUniqueld.....	112
TABLA 4.17. OBDNetParameterGetPidList.....	113
TABLA 4.18. OBDNetParameterGetInfo.....	113
TABLA 4.19. OBDNetParameterGetCurrentValue.....	114
TABLA 4.20. Logging Functions.....	115
TABLA 4.21. Tabla prioridad PID Airflow (Flujo de aire).....	117
TABLA 4.22. Tabla prioridad Distance (Distancia).....	117
TABLA 4.23. Tabla prioridad PID Emissions (Emisiones).....	118
TABLA 4.24. Tabla prioridad PID Environment (Ambiente).....	118

TABLA 4.25. Tabla prioridad PID Fuel (Combustible).....	118
TABLA 4.26. Tabla prioridad PID General (General).....	119
TABLA 4.27. Tabla prioridad PID Performance (Desempeño).....	119
TABLA 4.28. Tabla prioridad PID Speed (Velocidad).....	119
TABLA 4.29. Tabla prioridad PID System (Sistema).....	120
TABLA 4.30. Tabla prioridad PID Time (Tiempo).....	122
TABLA 4.31. Tabla prioridad PID Transmission (Transmisión).....	122
TABLA 5.1. MEDICIONES DE ESTADO PIDs.....	252
TABLA 5.2. TABLA DE MEDICIÓN DE VELOCIDAD.....	254
TABLA 5.3. TABLA DE MEDICIÓN DE SENSOR APP.....	254
TABLA 5.4. TABLA DE MEDICIÓN DE SENSOR TPS.....	255
TABLA 5.5. TABLA DE MEDICIÓN DE COMPRESIÓN Y VACÍO.....	256
TABLA 5.6. TABLA DE MEDICIÓN DE ACELERACIÓN.....	256
TABLA 5.7. TABLA DE MEDICIÓN DE PODER.....	257
TABLA 5.8. TABLA DE MEDICIÓN DE TORQUE.....	257
TABLA 5.9. TABLA DE MEDICIÓN DE ACELERACIÓN EN Gs.....	258
TABLA 5.10. CONSUMO INSTANTÁNEO DE COMBUSTIBLE.....	258
TABLA 5.11. CONSUMO PROMEDIO DE COMBUSTIBLE.....	259
TABLA 5.12. MEDICIÓN DE FLUJO DE COMBUSTIBLE.....	259
TABLA 5.13. TABLA DE MEDICIÓN DE SENSOR ECT.....	259
TABLA 5.14. TABLA DE MEDICIÓN DE SENSOR IAT.....	260
TABLA 5.15. TABLA DE MEDICIÓN DE VOLTAJE DE BATERÍA.....	260
TABLA 5.16. TABLA DE MEDICIÓN DE PODER EN HPs.....	261
TABLA 5.17. TABLA DE MEDICIÓN DE MÁXIMO PODER.....	261
TABLA 5.18. TABLA DE MEDICIÓN DE TORQUE DE SALIDA.....	262
TABLA 5.19. TABLA DE MEDICIÓN DE MAF.....	262
TABLA 5.20. TABLA DE MEDICIÓN DE MAP.....	263
TABLA 6.1 Presupuesto.....	270

ÍNDICE DE FIGURAS

Fig. 2.1.	Sistemas de comunicación automotrices inteligentes.....	7
Fig. 2.2.	Esquema de una red Punto a Punto unida a dos redes de difusión.....	10
Fig. 2.3.	Red de líneas multipunto.....	11
Fig. 2.4.	Red con utilización de líneas privadas.....	11
Fig. 2.5.	Red de tipo pública.....	11
Fig. 2.6.	Red de línea dedicada.....	12
Fig. 2.7.	Esquema de transmisión asíncrona.....	12
Fig. 2.8.	Red con configuración sincronizada.....	13
Fig. 2.9.	Esquema de transducción.....	15
Fig. 2.10.	Esquema de funcionamiento modulador – codificador.	16
Fig. 2.11.	Comunicación por transferencia de información.....	17
Fig. 2.12.	Esquema de comunicación de mediano y largo alcance.....	18
Fig. 2.13.	Comunicación IrDA de largo alcance por bits y caracteres.....	19
Fig. 2.14.	Interconexión de Unidades Roadside.....	20
Fig. 2.15.	Unidades a bordo interconectados.....	20
Fig. 2.16.	Frecuencias de utilización en redes WLAN.....	24
Fig. 2.17.	Enlace dedicado típico.....	27
Fig. 2.18.	Red en malla con infraestructura por nodos.....	28
Fig. 2.19.	Composición Estación Móvil.....	29
Fig. 2.20.	Arquitectura de intercomunicación HLR – VLR.....	30
Fig. 2.21.	Esquema de componentes GSM.....	30
Fig. 2.22.	Red de Comunicaciones WiMax.....	31
Fig. 2.23.	Configuración WiMax punto-punto.....	31
Fig. 2.24.	Red Mallada WiMax.....	32
Fig. 2.25.	Configuración típica WI-FI.....	33
Fig. 2.26.	Red WI-FI Punto a Punto (Ad Hoc).....	33
Fig. 2.27.	Red WI-FI de Infraestructura.....	34

Fig. 2.28.	Tipos de Ataques a redes inalámbricas.....	38
Fig. 2.29.	Diagrama de flujo de seguridad WEP.....	40
Fig. 2.30.	Diseños de seguridad.....	42
Fig. 2.31.	Firewalls antes y después de la VPN.....	42
Fig. 2.32.	Configuración Interfaz de Usuario.....	46
Fig. 2.33.	Interfaz Gráfica de Usuario.....	47
Fig. 2.34.	Pantalla Táctil.....	51
Fig. 2.35.	Componentes Pantalla Táctil Resistiva.....	52
Fig. 2.36.	Pantalla Táctil Capacitiva.....	53
Fig. 2.37.	Configuración Dispositivo Multitáctil.....	55
Fig. 2.38.	Tablet PC.....	56
Fig. 2.39.	IPEN.....	57
Fig. 2.40.	Giroscopio de última tecnología.....	59
Fig. 2.41.	Acelerómetro Digital.....	60
Fig. 2.42.	Grafica de tendencia de un valor de amplitud de vibración de un cojinete.	66
Fig. 2.43.	Curva de bañera. Probabilidad de fallo vs. Tiempo.....	67
Fig. 2.44.	Conector OBD II y especificaciones.....	85
Fig. 2.45.	Códigos de Error OBD II.....	85
Fig. 2.46.	Lectores de Códigos OBD II Simplificados.....	87
Fig. 2.47.	Medidores digitales, registro y ubicación en la Ipa de Ipad.....	88
Fig. 4.1.	Diagramas de bloques SDK.....	116
Fig. 4.2.	Tabla de prioridad Lógica SDK.....	116
Fig. 4.3.	Diagrama SDK CALC.ACCELERATION.....	123
Fig. 4.4.	Diagrama SDK CALC.ACCELERATION_G.....	123
Fig. 4.5.	Diagrama SDK CALC.AFR_ACTUAL.....	123
Fig. 4.6.	Diagrama SDK CALC.AFR_COMMANDED.....	124
Fig. 4.7.	Diagrama SDK CALC.AIR_DENSITY.....	124
Fig. 4.8.	Diagrama SDK CALC.BOOST_PRESSURE.....	124
Fig. 4.9.	Diagrama SDK CALC.DISTANCE.....	124
Fig. 4.10.	Diagrama SDK CALC.DTE.....	125

Fig. 4.11.	Diagrama SDK CALC.ENGINE_POWER.....	125
Fig. 4.12.	Diagrama SDK CALC.ENGINE_TORQUE.....	127
Fig. 4.13.	Diagrama SDK CALC.FC.AFC.....	127
Fig. 4.14.	Diagrama SDK CALC.FC.AFC.{01..17}.....	127
Fig. 4.15.	Diagrama SDK CALC.FC.AVERAGE.....	128
Fig. 4.16.	Diagrama SDK CALC.FC.FUEL_VOLUME.....	128
Fig. 4.17.	Diagrama SDK CALC.FC.IFC.....	128
Fig. 4.18.	Diagrama SDK CALC.FC.IFC_AVG.....	129
Fig. 4.19.	Diagrama SDK CALC.FILLUP.....	129
Fig. 4.20.	Diagrama SDK CALC.FILLUP.CORRECTION.....	129
Fig. 4.21.	Diagrama SDK CALC.FILLUP.FUEL_PRICE.....	129
Fig. 4.22.	Diagrama SDK CALC.FILLUP.FUEL_USED.....	129
Fig. 4.23.	Diagrama SDK CALC.FRP.....	130
Fig. 4.24.	Diagrama SDK CALC.FUEL_CONSUMPTION.....	130
Fig. 4.25.	Diagrama SDK CALC.FUEL_FLOW.....	131
Fig. 4.26.	Diagrama SDK CALC.ENGINE_POWER.....	132
Fig. 4.27.	Diagrama SDK CALC.FUEL_LEVEL.....	131
Fig. 4.28.	Diagrama SDK CALC.GEAR.....	132
Fig. 4.29.	Diagrama SDK CALC.GEAR.CURRENT.....	132
Fig. 4.30.	Diagrama SDK CALC.GEAR.NUM_GEAR.....	132
Fig. 4.31.	Diagrama SDK CALC.SHIFT_INDICATOR.....	132
Fig. 4.32.	Diagrama SDK CALC.GEAR.SUGGESTED.....	133
Fig. 4.33.	Diagrama SDK CALC.IS_NEW_DAY.....	133
Fig. 4.34.	Diagrama SDK CALC.LAMBDA_ACTUAL.....	133
Fig. 4.35.	Diagrama SDK CALC.LAMBDA_COMMANDED.....	133
Fig. 4.36.	Diagrama SDK CALC.MAF.....	133
Fig. 4.37.	Diagrama SDK CALC.MAF_A.....	134
Fig. 4.38.	Diagrama SDK CALC.MAF_B.....	133
Fig. 4.39.	Diagrama SDK CALC.MAP_A.....	135
Fig. 4.40.	Diagrama SDK CALC.MAP_B.....	135
Fig. 4.41.	Diagrama SDK CALC.MAP.....	136
Fig. 4.42.	Diagrama SDK CALC.TRIP.....	137
Fig. 4.43.	Diagrama SDK CALC.TRIP.ACCEL.....	137

Fig. 4.44.	Diagrama SDK CALC.TRIP.ACCEL.BRAKING. MAX{ABTPF}.....	137
Fig. 4.45.	Diagrama SDK CALC.TRIP.ACCEL.FORWARD. MAX.{ABTPF}.....	138
Fig. 4.46.	Diagrama SDK CALC.TRIP.AFC.....	138
Fig. 4.47.	Diagrama SDK CALC.TRIP.AFC.{ABTPF}.....	138
Fig. 4.48.	Diagrama SDK CALC.TRIP.BOOST.....	138
Fig. 4.49.	Diagrama SDK CALC.TRIP.BOOST.AVG.{ABTPF}.....	138
Fig. 4.50.	Diagrama SDK CALC.TRIP.BOOST MAX_BOOST.{ABTPF}.....	139
Fig. 4.51.	Diagrama SDK CALC.TRIP.CO2.....	139
Fig. 4.52.	Diagrama SDK CALC.TRIP.CO2.RATE.{ABTPF}.....	139
Fig. 4.53.	Diagrama SDK CALC.TRIP.CO2.TOTAL.{ABTPF}.....	139
Fig. 4.54.	Diagrama SDK CALC.TRIP.DISTANCE.....	139
Fig. 4.55.	Diagrama SDK CALC.TRIP.DISTANCE.{ABTPF}.....	140
Fig. 4.56.	Diagrama SDK CALC.TRIP.FILLUPS.....	140
Fig. 4.57.	Diagrama SDK CALC.TRIP.FILLUPS.{ABTPF}.....	140
Fig. 4.58.	Diagrama SDK CALC.TRIP.FUEL_COST.....	140
Fig. 4.59.	Diagrama SDK CALC.TRIP.FUEL_COST.{ABTPF}.....	141
Fig. 4.60.	Diagrama SDK CALC.TRIP.FUEL_FLOW.....	141
Fig. 4.61.	Diagrama SDK CALC.TRIP.FUEL_FLOW AVG.{ABTPF}.....	141
Fig. 4.62.	Diagrama SDK CALC.TRIP.FUEL_FLOW MAX.{ABTPF}.....	141
Fig. 4.63.	Diagrama SDK CALC.TRIP.FUEL_USED.....	142
Fig. 4.64.	Diagrama SDK CALC.TRIP.FUEL_USED.{ABTPF}.....	142
Fig. 4.65.	Diagrama SDK CALC.TRIP.GEAR.....	142
Fig. 4.66.	Diagrama SDK CALC.TRIP.GEAR.{1..6}. DISTANCE_PCT.{ABTPF}.....	143
Fig. 4.67.	Diagrama SDK CALC.TRIP.GEAR.{1..6}. TIME_PCT.{ABTPF}.....	143

Fig. 4.68.	Diagrama SDK CALC.TRIP.GEAR.N. DISTANCE_PCT.{ABTPF}.....	143
Fig. 4.69.	Diagrama SDK CALC.TRIP.GEAR.N TIME_PCT.{ABTPF}.....	143
Fig. 4.70.	Diagrama SDK CALC.TRIP.GEAR.N. TIME_PCT.{ABTPF}.....	143
Fig. 4.71.	Diagrama SDK CALC.TRIP.GEAR.WG. TIME_PCT.{ABTPF}.....	143
Fig. 4.72.	Diagrama SDK CALC.TRIP.POWER.....	144
Fig. 4.73.	Diagrama SDK CALC.TRIP.POWER.MAX.{ABTPF}...	144
Fig. 4.74.	Diagrama SDK CALC.TRIP.RPM.....	144
Fig. 4.75.	Diagrama SDK CALC.RPM.MAX.{ABTPF}.....	144
Fig. 4.76.	Diagrama SDK CALC.TRIP.SPEED.....	144
Fig. 4.77.	Diagrama SDK CALC.TRIP.SPEED. AVG_NI.{ABTPF}.....	145
Fig. 4.78.	Diagrama SDK CALC.TRIP.SPEED. AVG_WI.{ABTPF}.....	145
Fig. 4.79.	Diagrama SDK CALC.TRIP.SPEED. MAX.{ABTPF}.....	145
Fig. 4.80.	Diagrama SDK CALC.TRIP.STOPS.....	145
Fig. 4.81.	Diagrama SDK CALC.TRIP.STOPS.{ABTPF}.....	145
Fig. 4.82.	Diagrama SDK CALC.TRIP.TIME.....	146
Fig. 4.83.	Diagrama SDK CALC.TRIP.TIME.DRIVE.{ABTPF}.....	146
Fig. 4.84.	Diagrama SDK CALC.TRIP.TIME. ELAPSED.{ABTPF}.....	146
Fig. 4.85.	Diagrama SDK CALC.TRIP.TIME.IDLE.{ABTPF}.....	146
Fig. 4.86.	Diagrama SDK CALC.TRIP.TIME.RUN.{ABTPF}.....	146
Fig. 4.87.	Diagrama SDK CALC.TRIP.TIME.START.{ABTPF}.....	147
Fig. 4.88.	Diagrama SDK CALC.TRIP.TORQUE.....	147
Fig. 4.89.	Diagrama SDK CALC.TRIP.TORQUE.MAX.{ABTPF}..	147
Fig. 4.90.	Diagrama SDK CALC.TTE.....	147
Fig. 4.91.	Diagrama SDK CALC.VACUUM.....	147
Fig. 4.92.	Diagrama SDK CALC.VSS_C.....	148

Fig. 4.93. Diagrama SDK.CALC.VSS_D.....	148
Fig. 4.94. Diagrama CONST.ADDITIONAL_WEIGHT.....	148
Fig. 4.95. Diagrama CONST.CURB_WEIGHT.....	148
Fig. 4.96. Diagrama CONST.DRAG_COEFFICIENT.....	149
Fig. 4.97. Diagrama CONST.ENGINE_DISPLACEMENT.....	149
Fig. 4.98. Diagrama CONST.FINAL_DRIVE_RATIOS.....	149
Fig. 4.99. Diagrama CONST.FRONTAL_AREA.....	149
Fig. 4.100. Diagrama CONST.FRP_PID.....	149
Fig. 4.101. Diagrama CONST.FUEL_CO2_CONTENT.....	150
Fig. 4.102. Diagrama CONST.FUEL_DENSITY.....	150
Fig. 4.103. Diagrama CONST.FUEL_TANK_CAPACITY.....	150
Fig. 4.104. Diagrama CONST.LAMBDA_ACTUAL_PID.....	150
Fig. 4.105. Diagrama CONST.LAMBDA_PID_COMMANDED...	150
Fig. 4.106. Diagrama CONST.MAF_PID.....	151
Fig. 4.107. Diagrama CONST.MAP_PID.....	151
Fig. 4.108. Diagrama CONST.MAX_RPM.....	151
Fig. 4.109. Diagrama CONST.MIN_RPM.....	151
Fig. 4.110. Diagrama CONST.SHIFT_RPM.....	152
Fig. 4.111. Diagrama CONST.SPEED_CORRECTION.....	152
Fig. 4.112. Diagrama CONST.SPEED_ZONES.....	152
Fig. 4.113. Diagrama CONST.SPEED_ZONES {1..6}.MAX_SPEED.....	152
Fig. 4.114. Diagrama CONST.SPEED_ZONES. {1..6}.MIN_SPEED.....	153
Fig. 4.115. Diagrama CONST.SPEED_ZONES. {1..6}.SHIFT_SPEED.....	153
Fig. 4.116. Diagrama CONST.STOICHIOMETRIC_ AIR_FUEL_RATIO.....	153
Fig. 4.117. Diagrama CONST.TIRE_RESISTANCE.....	153
Fig. 4.118. Diagrama CONST.VOLUMETRIC_EFFICIENCY.....	154
Fig. 4.119. Diagrama CONST.WHEEL_CIRCUMFERENCE.....	154
Fig. 4.120. General Settings SkinXL Editor.....	155
Fig. 4.121. Dashboard Screen Settings DashXL Editor.....	155

Fig. 4.122. DashXL Editor.....	156
Fig. 4.123. Plantilla de Partida.....	159
Fig. 4.124. Diseño de Pestaña Menu Principal.....	160
Fig. 4.125. Diseño de Identificador de Tesis.....	163
Fig. 4.126. Diseño de Medidor Compresión/Vacío.....	165
Fig. 4.127. Diseño de Medidor de Poder y Torque.....	171
Fig. 4.128. Diseño de Medidor de TPS.....	180
Fig. 4.129. Configuración Path TPS.....	181
Fig. 4.130. Diseño de Medidor de APP.....	183
Fig. 4.131. Configuración Path APP.....	185
Fig. 4.132. Plantilla dinámica diagnóstico.....	188
Fig. 4.133. Diseño de Diagnóstico Flujo de Aire.....	207
Fig. 4.134. Diseño de Dashboard Medición y Telemetría.....	212
Fig. 4.135. Diseño de Dashboard GLCD FUERZAS G.....	231
Fig. 4.136. Diseño de Dashboard GLCD VELOCIDAD y RPM.....	233
Fig. 4.137. Diseño elemento Path GLCD Velocidad.....	234
Fig. 5.1. Ubicación de puerto OBD-II.....	236
Fig. 5.2. Conexión WI-FI con el generador de señal.....	237
Fig. 5.3. Configuración de Dirección IP y Máscara de Subnet.....	237
Fig. 5.4. Configuración del panel Connection.....	239
Fig. 5.5. Íconos de conexión e ingreso a Dashboards.....	240
Fig. 5.6. Conexión SSH con el Directorio DashCommand.app.....	240
Fig. 5.7. Ubicación archivo .dxls.....	241
Fig. 5.8. Diseño Soporte Tablet Parabrisas.....	241
Fig. 5.9. Comprobación de conexión por Data Logging.....	244
Fig. 5.10. Consola de Depuración de la Compilación.....	245
Fig. 5.11. Engine codes y Readiness Monitors.....	246
Fig. 5.12. DTCs de tipo almacenados, pendientes y permanentes.....	247
Fig. 5.13. Monitoreos activos de sistemas.....	247
Fig. 5.14. Chevrolet Optra 1.8L T/M Design 2007 Sedán.....	249
Fig. 5.15. Mazda 3 1.6L T/M 2009.....	249
Fig. 5.16. Peugeot 206 1.4L T/M 2006.....	249

Fig. 5.17. Hoja de información del Vehículo.....	250
Fig. 5.18. Request de la apertura de comunicación OBD-II.....	251
Fig. 5.19. Lector Principal de funcionamiento.....	252
Fig. 5.20. Lectores de Telemetría creando base de datos.....	253
Fig. 5.21. Medición de velocidad en Prueba de Manejo.....	254
Fig. 5.22. Medición de Sensor APP Prueba Estacionado.....	255
Fig. 5.23. Medición de Sensor TPS Prueba de Manejo.....	255
Fig. 5.24. Medición de Compresión/Vacío Prueba de Manejo.....	256
Fig. 5.25. Medición de Aceleración en Prueba de Manejo.....	257
Fig. 5.26. Medición de Aceleraciones en Gs.....	258
Fig. 5.27. Medición de Sensor ECT	259
Fig. 5.28. Medición de sensor IAT.....	260
Fig. 5.29. Medición de Voltaje (Referencial).....	261
Fig. 5.30. Medición de MAF.....	262
Fig. 5.31. Medición de MAP en prueba de manejo.....	263

PRESENTACIÓN

En el proyecto, **“DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE DIAGNÓSTICO PREDICTIVO PARA AUTOMOTORES BAJO NORMA OBD II, CON CONEXIÓN INALÁMBRICA WI-FI Y CONTROLADORES TÁCTILES DE LA MARCA APPLE”** se relacionan diversos conocimientos adquiridos en el transcurso de nuestra carrera.

La Escuela Politécnica del Ejército impulsa la investigación científica y la experimentación, además de la innovación tecnológica dentro de cada carrera, es por eso que se vió la necesidad de diseñar y construir un proyecto en base a una herramienta tecnológica que permita al profesional automotriz personalizar dispositivos electrónicos, mediante la utilización de lenguajes de programación nativos de Apple y software de conjugación de interfaz nativa de usuario (NUI) con interfaz gráfica de usuario (GUI) elaborada en Visual Basic.

El presente proyecto fue elaborado, en base a desarrollos tecnológicos de punta existentes en la actualidad, con el uso de dispositivos táctiles Tablets que permitan una conexión inalámbrica con el automóvil, aparte de su aspecto físico elegante y sus características de conexión y fiabilidad de manejo de información inmejorables.

El capítulo 1 presenta el análisis metodológico del problema a resolver.

El capítulo 2 trata sobre el marco teórico empleado en el desarrollo de la presente investigación.

El capítulo 3, se refiere al planteamiento de las hipótesis y la operacionalización de las variables.

El capítulo 4 versa sobre el diseño de todo el sistema de conexión, utilizando informática Objective-C aplicada al campo automotriz.

El capítulo 5 plantea el proceso de construcción y conexión del dispositivo con el diseño incluido en su directorio informático, con la interfaz OBD-II del automóvil.

El capítulo 6 establece el marco administrativo del presente trabajo, tomando en cuenta los recursos, tanto humanos como tecnológicos, la financiación y el cronograma utilizado.

Para finalizar se incluyen conclusiones y recomendaciones, obtenidas durante todo el proceso, tanto de diseño como de construcción, por el investigador, Director y Codirector de tesis.

CAPÍTULO 1

EL PROBLEMA

1.1. PLANTEAMIENTO DEL PROBLEMA

La continua renovación de la tecnología instalada en los vehículos, crea la necesidad de contar con equipos y herramientas de diagnóstico predictivo, que sirvan de soporte a los profesionales especializados dentro de un proceso de verificación, diagnóstico y obtención de parámetros de óptimo funcionamiento de los sistemas de control de gestión integral en los motores de combustión interna, bajo normas de control OBDII.

Es un hecho visible que los talleres enfocados al servicio de reparación y mantenimiento, los diseñadores de sistemas automotrices, estudiantes y aficionados a las áreas del automóvil; tienen la necesidad de contar con sistemas que incorporen las ventajas de aplicación de las nuevas tecnologías de información y comunicación, que tengan un enlace inmediato universal entre varios grupos de monitoreo, así como que provean de información rápida y en tiempo real, sin la necesidad de una intervención directa en los mecanismo a ser analizados.

En el mercado no existen sistemas de diagnóstico que puedan integrar todas las funciones de un escáner OBDII, con los métodos adecuados para identificar, por ejemplo, el estado mecánico de motor a través de las pruebas de compresión y vacío, o conocer como está calibrado el radio de la mezcla aire combustible; así como, de igual manera, saber el comportamiento de algunos sistemas adicionales, como son los turbos y los intercoolers.

Como es de conocimiento, las pruebas de diagnóstico predictivo son realizadas independientemente, unas de otras, como operaciones mecánicas de intervención directa humana. Esto ocasiona el incremento de los tiempos de parada de los automotores en los talleres de servicio, con lo que la productividad de los mismos se ve limitada a la rapidez con la que, los operarios realicen estas inspecciones en cada estación de trabajo.

Con la introducción de sistemas inteligentes, se puede realizar un análisis integral de las condiciones mecánicas, sin la intervención de operarios manuales, puesto que los datos del estado de los diferentes sistemas se pueden obtener a través de la comunicación entre vehículo y herramienta de diagnóstico, tanto en condiciones de funcionamiento como cronológicamente en un record del vehículo en diferentes análisis realizados, mejorando la productividad del proceso en todas sus estaciones.

1.2. FORMULACIÓN DEL PROBLEMA

La ESPE Extensión Latacunga oferta el Programa de III Nivel de Ingeniería Automotriz el que tiene como propósito formar profesionales capacitados para analizar sistemas electrónicos aplicados en los vehículos, por lo que para la correcta formulación del problema, se toman en cuenta las siguientes interrogantes:

¿Qué requisitos teóricos son necesarios introducir en el proyecto para diseñar un sistema de diagnóstico predictivo?

¿Es necesario que la herramienta táctil sea especializada para un solo tipo de automotor?

¿Qué factor informáticos, tanto teóricos como prácticos, son necesarios para diseñar un interfaz gráfica de usuario, basada en un lenguaje de programación nativo de la marca Apple, como lo es el Objective-C?

¿Cuál es la relación de conjunción entre los factores mecánicos y electrónicos del automóvil, que intervendrían en nuestro análisis para un excelente diagnóstico predictivo - preventivo?

¿Cómo se puede personalizar una IPA, para que el usuario tenga a su alcance el análisis de todas las PIDs disponibles, de acuerdo a la lista de PIDs soportada por cada ECU?

¿Cuál es el nivel de seguridad, fiabilidad y confianza tanto en el envío como en la recepción de datos en un ambiente inalámbrico?

1.3. OBJETIVO GENERAL

Diseñar y construir un sistema de diagnóstico predictivo para automotores bajo norma OBD II, con conexión inalámbrica Wi-Fi y controladores táctiles de la marca Apple.

1.4. OBJETIVOS ESPECÍFICOS DEL PROYECTO

- Diseñar un sistema de diagnóstico predictivo mecánico electrónico integral.
- Desarrollar la conectividad entre dispositivos generadores de señal WI-FI y dispositivos táctiles de la marca Apple.
- Personalizar los dispositivos táctiles para la interpretación de datos y análisis telemétrico para el uso en Mecánica de Competición.
- Construir un dispositivo universal de lectura y borrado de DTCs con módulos de sensores especiales.

1.5. JUSTIFICACIÓN E IMPORTANCIA

Las comunicaciones inalámbricas han sido identificadas como tecnologías clave para incrementar la seguridad en la carretera y la eficiencia del transporte, de la misma manera que el acceso móvil a Internet, para asegurar la conectividad inalámbrica. El potencial de esta tecnología se ha asentado con el establecimiento de ambiciosos programas de investigación en todo el mundo, tales como la iniciativa Europea eSafety, los programas estadounidenses derivados de la *Intelligent Vehicle Initiative* y los programas japoneses *InternetITS* y *Advanced Highway Systems (AHS)*.

El ámbito de los sistemas de comunicaciones inalámbricos en transporte cubre áreas de interés como comunicaciones móviles, sistemas de transporte o electrónica del automóvil. A partir de estos componentes, se han desarrollado una serie de aplicaciones y sistemas como son las comunicaciones Vehículo-Vehículo (V2V), comunicaciones Vehículo-Infraestructura (V2I) y Vehículo-Persona (V2P), incluyendo implicaciones en eficiencia del transporte y seguridad, implicaciones en electrónica de automoción, aspectos de responsabilidad civil o esfuerzos en la realización de estándares y asignación de espectros electromagnéticos.

La implementación de un sistema de diagnóstico predictivo, universal e integrado, solo será posible si se tiene conocimiento amplio y una descripción detallada de la tecnología integrada en los automóviles para las distintas necesidades de control y en forma específica para la red que se desee diagnosticar, por esta razón se van a estudiar los protocolos de comunicación, configuraciones de red, sistemas integrados, diagramación, cableado, accesibilidad, diagnóstico de componentes, el análisis de resultados luego de la verificación de transmisión de datos y señales mediante los dispositivos entrelazados a través de la señal WI FI.

CAPÍTULO 2

MARCO TEÓRICO

2.1. INTRODUCCIÓN A LOS SISTEMAS DE COMUNICACIÓN BASADOS EN REDES INALÁMBRICAS WI FI

2.1.1 SISTEMAS DE COMUNICACIÓN.

El incremento en el número de componentes de vehículo controlados electrónicamente mediante ECU es continuo. Si bien, la cooperación entre esas ECU y el acceso a la información que circula por ellas, o entre ellas y el conjunto de sensores y actuadores, sigue quedando restringido a los fabricantes de vehículos y las grandes empresas fabricantes de componentes, es necesario la creación de nuevas interfaces intuitivas y de rápido acceso de comunicación con estos componentes, para la utilización de ingenieros y aficionados en el campo automotriz, en los campos del diagnóstico y el confort.

En este concepto, la utilización de sistemas de comunicación con redes de tipo inalámbricas, nos sería el tipo de enlace apto para el intercambio de información, tomando en cuenta los tiempos de respuesta, que en este caso son muy altos, y los medios en los cuales van a ser instalados óptimos.

Está claro que todo este conjunto de servicios asociados al transporte requieren un tipo de sistemas de comunicaciones específico, primando conceptos como el tiempo real, la privacidad, la conectividad entre vehículos e infraestructuras, etc.

Sin embargo, dado el poco espacio de tiempo transcurrido desde la detección de la necesidad de sistemas de comunicaciones inalámbricos en

transporte, en la actualidad, la mayoría de estos sistemas específicos se encuentran en fase de desarrollo y en algunos casos de investigación. Sin embargo, la necesidad de implantación en el mercado de los citados servicios y aplicaciones hace que, de momento, se use de las tecnologías disponibles en la actualidad, aunque en algunos casos, no se ajusten totalmente a todos los requisitos de las diferentes aplicaciones.

Es por ello que, aunque algunas de estas tecnologías inalámbricas funcionan perfectamente cuando las embarcamos en un vehículo, no han sido diseñadas específicamente para ello y en algunos casos presentan problemas e incompatibilidades que deben ser mejorados. Se deben estudiar todas las vías posibles para la integración de los nuevos dispositivos embarcados de comunicación Vehicle to Vehicle y Vehicle to Interface dentro de los propios sistemas de vehículos.

Por lo tanto, las líneas de estudio pasan por el planteamiento de estándares de conexión y comunicación entre los nuevos dispositivos de equipamiento del vehículo, facilitan la conexión entre ambos, evitando el desarrollo de dispositivos específicos o únicos; y la posibilidad de que el equipamiento embarcado pueda acceder a ciertas capas de información de los sistemas del vehículo, garantizando la privacidad de aquellas otras capas que corresponden a uso propietario. Para ello será necesario el desarrollo de hardware y software de cifrado que garanticen fielmente las comunicaciones seguras entre ambas partes.

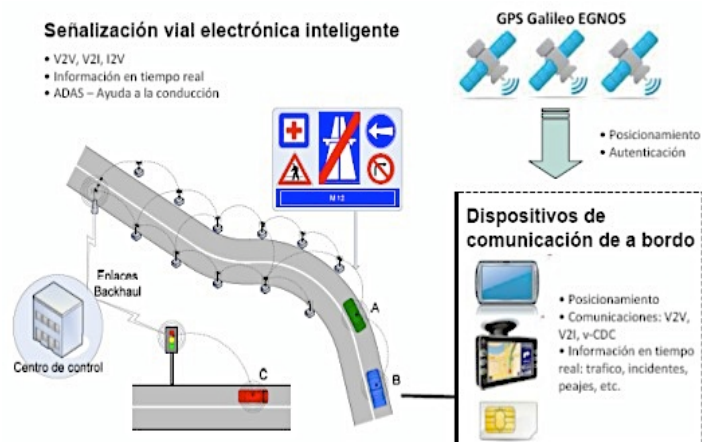
El presente proyecto facilitará el estudio, análisis e interpretación de los diferentes sistemas eléctricos, electrónicos y mecánicos de los diferentes tipos y marcas de automóviles, que mantengan un sistema de comunicación OBD II. Los profesionales y estudiantes tendrían una herramienta táctil, con una interfaz gráfica muy amigable y fácil de manejar; con comunicación universal con los tipos específicos de conexión normados internacionalmente.

El esquema básico de conexión, está compuesto por una interfaz de lectura de códigos de tipo OBD II con generador de señal de emisión de datos Wi-Fi, un software de aplicación desarrollado en una interfaz única de usuario (NUI), un dispositivo táctil de la marca Apple (Iphone, Ipad), y una red establecida de registro de información periódica e individual para cada modelo y tipo de automotor analizado, que nos permitirá, de manera intuitiva, acceder a toda la red y poder manejar esta información para un diagnóstico predictivo adecuado.

a. VISIÓN TECNOLÓGICA

Las redes vehiculares son un novedoso escenario de comunicación. Estas redes permiten el diálogo entre vehículos, y de estos con la infraestructura. Gracias a estas redes se puede proporcionar más información y nuevos servicios a conductores y pasajeros. Uno de esos nuevos servicios es la creación de evidencias sobre el comportamiento de un vehículo.

Utilizando las redes vehiculares se puede obtener la descripción de ese comportamiento a través de los vehículos del entorno. Con ello se impide que el propio vehículo describa su actuación de una forma modificada acorde con sus intereses. Para abordar este nuevo servicio es necesario garantizar la seguridad de la información intercambiada.



Fuente: Sáenz, V. (1999). Globalización, nuevas tecnologías y comunicación.

Fig 2.1 Sistemas de comunicación automotrices inteligentes

El transporte terrestre mediante vehículos es, una de las actividades humanas imprescindibles. Por este motivo, las inversiones para la mejora de todos los elementos implicados son cada vez más notables. Los avances en esta dirección están orientados a facilitar la tarea de conducción y proporcionar un mayor bienestar, seguridad a los conductores y pasajeros.

Constantemente se introducen avances tecnológicos en los productos, acompañando las necesidades de los empresarios y los anhelos de los usuarios, siempre en áreas de un mayor desempeño, durabilidad, facilidad de mantenimiento, seguridad y confort. La vanguardia tecnológica ha ido al encuentro de la satisfacción de ese binomio empresario de transportes-usuario. Así se puede apreciar cada vez más en el mercado una competencia de todas las empresas de introducción innovadora en ofrecer un producto que se ajuste a las necesidades reales y con un alto grado de confiabilidad y calidad en sus unidades.

Para satisfacer estas necesidades, las tecnologías de la información y las comunicaciones están experimentando un fuerte desarrollo en el ámbito vehicular. En los últimos tiempos, las redes ad-hoc vehiculares (Vehicular Ad-hoc NETwork, VANET) se han sumado a las posibilidades de conectividad actuales (Wi-Fi, GPRS, UMTS). Este nuevo tipo de redes están especialmente concebidas para interconectar vehículos entre sí y con las infraestructuras de comunicación que se despliegan en las carreteras.

b. INTRODUCCIÓN A LOS SISTEMAS DE COMUNICACIÓN

Para familiarizarse eficazmente con la aplicación de los Sistemas de Comunicación es necesario conocer sus generalidades. A continuación se muestra una perspectiva general sobre los puntos clave para poder tomarlos en cuenta en el desarrollo de tecnologías aplicativas.

i. DEFINICIONES

Información: Parte útil de un mensaje en un proceso de transmisión.

Transmisión: Proceso por el cual se envía una señal entre dos puntos.

Comunicación: Proceso por el cual se transmite información dentro de una señal durante una transmisión.

Nodo: Punto, momento o espacio en donde todos los elementos de una red que comparten las mismas características se vinculan e interactúan.

Emisor: Objeto que codifica un mensaje y lo transmite a través de un cierto canal hacia el receptor.

Receptor: Equipo o persona cuya principal función es la de recibir.

Infraestructura: Conjunto de elementos o servicios que se consideran necesarios para el funcionamiento de una organización o para el desarrollo de una actividad o comunicación.

Enlace: Conexión entre equipos o nodos.

Tramas: Unidad de envío de datos.

Transductores: Dispositivo capaz de transformar o convertir un determinado tipo de energía de entrada, en otra de diferente a la salida.

ii. NORMAS Y ESTANDARES

Dadas las posibilidades que podemos encontrarnos en un proceso de comunicación, se debe establecer los aspectos necesarios para que esa comunicación exista. Para ellos es necesario el uso de estándares, que son acuerdos al respecto de cada aspecto de dicho proceso.

Existen dos tipos de estándares:

- *Estándares de facto o de hecho:* Son aquellos que se imponen por su uso generalizado.

- *Estándares de iure o de derecho*: Son aquellos estándares creados por una organización de estándares, fabricantes o similares.

Existen distintas organizaciones para la fijación de estándares como son IEEE (Institute of Electrical and Electronics engineers), ANSI (American National Standards Institute), W3C, ISO (International Organization for Standardization), ADSL Forum, etc.

iii. LÍNEAS DE COMUNICACIÓN

Las líneas de comunicación son las vías a través de las cuales los circuitos de datos pueden intercambiar información. Cuando se interconectan dos o más equipos de comunicación a través de líneas de comunicaciones se construye una red de comunicación.

Podemos clasificar las líneas de comunicaciones según su topología y según su propietario.

a. *Según la topología de la conexión* tenemos:

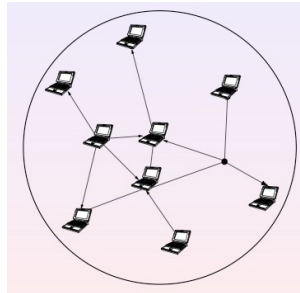
- **Líneas punto a punto**: Son líneas que unen dos equipos sin posibilidad de que otros equipos soliciten información a través de dichas líneas.



Fuente: Grupo de investigación.

Fig 2.2 Esquema de una red Punto a Punto unida a dos redes de difusión

- **Líneas multipunto:** Las líneas multipunto están formadas por una red troncal constituida por un bus de comunicaciones común a todos los equipos que se conectan a la red.

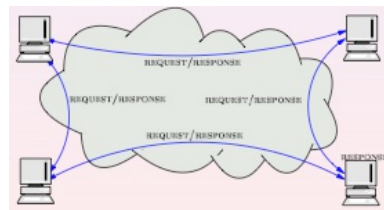


Fuente: Grupo de investigación.

Fig. 2.3 Red de líneas multipunto

b. Según el propietario de las líneas:

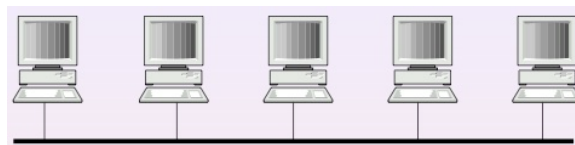
- **Líneas privadas:** Una línea es privada cuando su propietario no es público. Una red de área local utiliza líneas privadas.



Fuente: Grupo de investigación.

Fig. 2.4 Red con utilización de líneas privadas

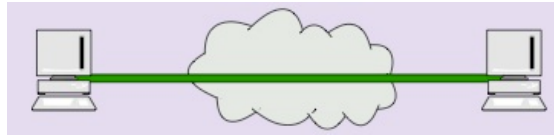
- **Líneas públicas:** Las líneas son de titularidad pública (normalmente de las compañías telefónicas). El usuario solamente contrata su uso en régimen de alquiler. El uso de la red completa es compartido por todos los usuarios que contratan sus servicios.



Fuente: Grupo de investigación.

Fig. 2.5 Red de tipo pública

- **Líneas dedicadas:** La línea es de titularidad pública, pero sólo se usa por los usuarios que la han contratado, normalmente los dos extremos de la línea.



Fuente: Grupo de investigación.

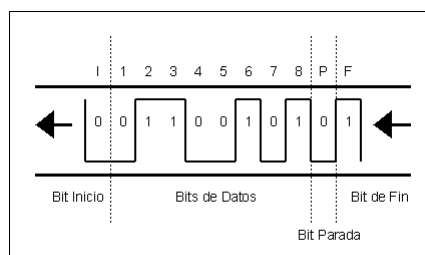
Fig. 2.6 Red de línea dedicada

iv. TIPOS DE TRANSMISIÓN

a. Clasificación según la transmisión

No todos los equipos entregan los datos a la línea de la misma manera, por lo que distinguiremos los siguientes tipos de transmisiones:

- **Transmisión asíncrona:** Emisor y receptor se ponen de acuerdo con cada carácter de datos que envíen. El emisor envía una señal para indicar que se va a iniciar un nuevo carácter de datos, lo que pone en marcha un reloj interno en el receptor que indica los tiempos que debe tener en cuenta para dicha recepción.

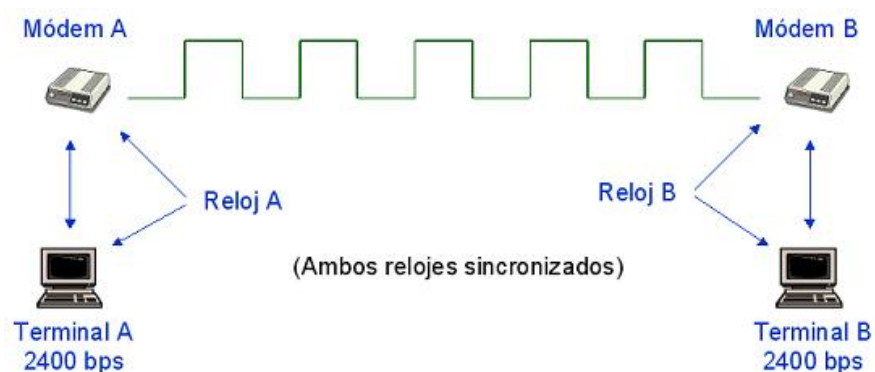


Fuente: Grupo de investigación.

Fig. 2.7 Esquema de transmisión asíncrona

Este tipo de transmisión es poco sensible a fallos de sincronismo, pero requiere del envío de más datos que la transmisión síncrona.

- **Transmisión síncrona:** En este tipo de transmisión, emisor y receptor envían y reciben los bits con una cadencia constante. Para sincronizar el comienzo y envío de los datos, se envía un carácter de sincronismo. Dicho carácter debe ser irrepetible aunque se desplacen los bits de datos, de forma que sea diferenciable en cualquier caso del resto de datos. Este tipo de transmisión es más sensible a los fallos de sincronismo que el tipo asíncrono, pero es más eficiente porque necesita del envío de menos datos de sincronismo en relación con los datos enviados.



Fuente: Grupo de investigación.

Fig. 2.8 Red con configuración sincronizada

Tipos de sincronismo

Tenemos distintos métodos de sincronizar emisor y receptor, por lo que distinguiremos los siguientes tipos de sincronismo:

- **Sincronismo de bit:** En este tipo de sincronismo, se envía un bit para indicar cuando se inicia la transmisión. Dicho bit pone en marcha el reloj interno del receptor. En la transmisión síncrona, es la propia señal de reloj, transmitida por al línea junto con los datos, la que se encarga de efectuar el sincronismo de bit.
- **Sincronismo de carácter:** Este tipo de sincronismo consiste en indica cuando comienza y cuando termina un carácter. En las transmisiones asíncronas, esto se indica con los bits de inicio y

parada, y en las síncronas con el carácter de sincronía.

- **Sincronismo de bloque:** Es más avanzado que los dos anteriores. Se define un conjunto de caracteres especiales que sirven para dividir el mensaje en bloques. Si los caracteres de inicio y final de bloque no se detectan cuando se debe, ha habido un fallo de sincronía.

b. Clasificación según la señal transmitida

- **Señal analógica o digital:** Una señal digital es aquella que puede tomar un conjunto de valores finito, mientras que una señal analógica puede tomar infinitos valores entre un valor máximo y otro mínimo. Una línea telefónica es analógica, mientras que un bus de datos de un ordenador es digital.
- **Transmisión en banda base y en banda ancha:** La transmisión necesita a veces de la transformación de la señal para su transmisión con el objeto de adecuar la señal a los elementos receptores de la misma. Este proceso se llama modulación. Si la señal se envía sin ningún tipo de modulación, se dice que la transmisión se efectúa en banda base. Si por el contrario necesita de modulación, se dice que la transmisión se produce en banda ancha.

c. Clasificación según el medio de transmisión

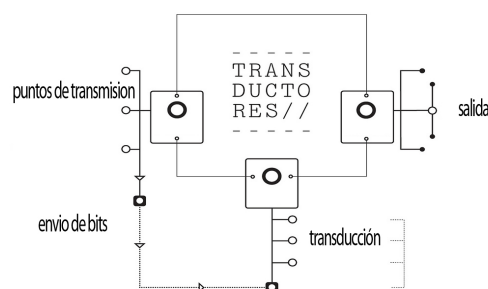
- **Transmisión serie:** Todos los datos se envían por la misma línea se forma secuencial.
- **Transmisión paralelo:** Los datos se envían en grupos de bits, uno por cada línea del mismo canal.

v. ELEMENTOS DE UN SISTEMA DE COMUNICACIONES

Vamos a contar con los elementos básicos de un proceso de comunicación, que son emisor y receptor, el canal y el mensaje. Pero vamos a añadir otros elementos que son imprescindibles cuando nos referimos a las comunicaciones desde un punto de vista informático.

Estos elementos son:

- **Los transductores:** Un transductor es un dispositivo que se encarga de transformar la señal para adecuarla a la naturaleza del canal. Por ejemplo: Un transductor en una comunicación telefónica sería el micrófono del terminal telefónico, que se encargaría de transformar la voz humana (vibración del aire) a una señal telefónica (pulso eléctrico que se transmite por la línea telefónica).

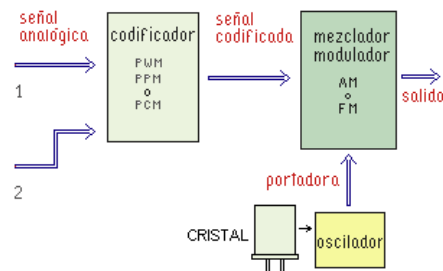


Fuente: Grupo de investigación.

Fig. 2.9 Esquema de transducción

- **Moduladores y codificadores:** Aunque la naturaleza de la señal sea la adecuada, en ocasiones se necesita de otro tipo de transformación para que el aprovechamiento del canal sea óptimo. De estas transformaciones se encargan los moduladores y los codificadores. El modulador se encarga de realizar las transformaciones necesarias para que la señal, además de adecuarse al canal, sea eficiente en cuanto al envío de información en esa señal (volvemos a distinguir entre señal e información). En el caso del codificador, la señal puede enviar información, pero esta información necesita de una transformación a un

formato entendible por emisor y receptor, así como ciertos mecanismos contra posibles errores, el codificador de este tipo de transformación.



Fuente: Grupo de investigación.

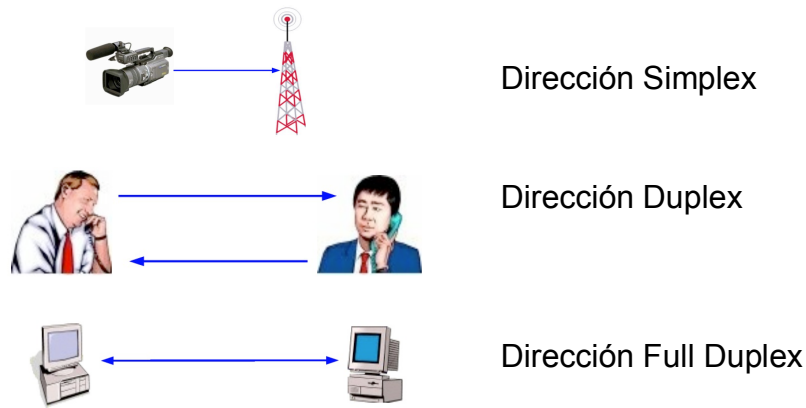
Fig. 2.10 Esquema de funcionamiento modulador - codificador

- **Elementos hardware que intervienen en el proceso de comunicación:** Nombraremos algunos elementos que aunque no son propios del proceso de comunicación, en el caso de la informática son requeridos para que dicha comunicación sea posible. Este es el caso de los amplificadores, repetidores de señal, concentradores, distribuidores, conmutadores, antenas, etc.

c. ARQUITECTURAS Y MECANISMOS DISPONIBLES

A continuación, vamos a establecer las características generales de las arquitecturas y mecanismos de red actuales:

- **Direccionamiento:** Proporciona direcciones de interfaces. Cuanto más subimos en el protocolo, más subirán las funcionalidades y menos tendrán que ver con la comunicación física en sí.
- **Transferencia de información:** Debemos fijar reglas de transferencia de información (en una dirección - simplex-, ambas -half-duplex-, ambas simultáneamente -full-duplex-). Generalmente, la transmisión será siempre full-duplex salvo en capas inferiores y casos puntuales en donde el medio físico no aguante ese sistema.



Fuente: Grupo de investigación.

Fig. 2.11 Comunicación por transferencia de información

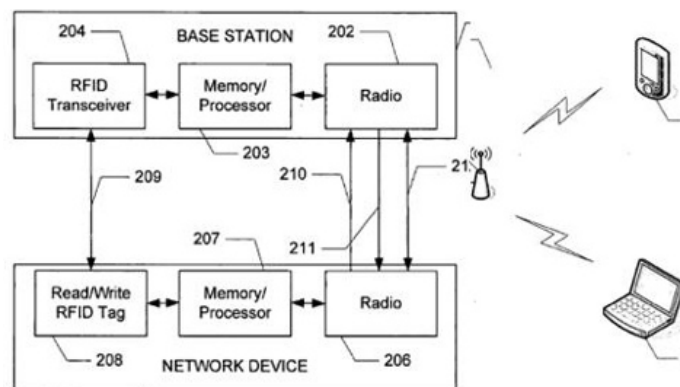
- **Detección y corrección de errores:** Serán útiles códigos redundantes para detectar errores u obligará una respuesta, para conseguir los llamados códigos autocorregidos.
- **Numeración de paquetes:** Es necesario para la reconstrucción de la transmisión por el receptor de forma consistente.
- **Mecanismo de control de flujo:** Sirven para adecuarlas diferentes velocidades de los interfaces de red. Veremos varios de estos mecanismos.
- **Mecanismos de control de congestión:** Sirven para evitar la saturación de los nodos intermedios.
- **Mecanismo de segmentación y concatenación:** En ocasiones las diferentes capas de la arquitectura no soportan tamaños iguales de paquetes unificados de información (PDU), de forma que será necesario proveer a nuestra red con herramientas de división de la PDU en varias paquetes simplificados unificados (SDU) y a la inversa para reconstruir esa segmentación en la entidad par. Ya más raro resulta encontrar la necesidad de concatenar PDUs pequeñas en una grande pero es de gran utilidad para reducir la información de control y aprovechar al

máximo los recursos del canal.

- **Multiplexación y División:** Dividir los recursos que tiene un canal entre diferentes comunicaciones (Multiplexación) o separar diferentes comunicaciones para que vayan en diferentes canales (División). La multiplexación normalmente exige un servicio tanto en la capa usuaria como en la capa proveedora. No confundir Multiplexación con segmentación. Evidentemente, habrá que poder identificar correctamente las diferentes comunicaciones.

i. COMUNICACIÓN DE INTERFACE AÉREA DE MEDIANO Y LARGO ALCANCE.

Las comunicaciones inalámbricas de largo alcance utilizan radio frecuencias (RF) y por lo general usan partes del espectro de radio frecuencia que están bajo licencias, esto es, se aplican cargos por su uso. Por otra parte las comunicaciones de corto y mediano alcance pueden utilizar rayos infrarrojos o bien partes del espectro de RF, los cuales por lo general no necesitan licenciamiento y su uso es gratuito.



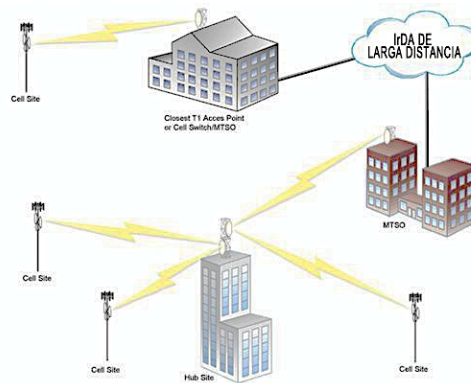
Fuente: Domínguez, C. (2002). Redes inalámbricas.

Fig. 2.12 Esquema de comunicación de mediano y largo alcance

Existen diversos estándares de comunicación inalámbrica de corto alcance, pero los tres principales son Infrarrojo de la "Infrared Data Association"

(IrDA), tecnología inalámbrica Bluetooth y redes inalámbricas de área local (WLAN).

El IrDA crea un sistema de comunicación inalámbrico utilizando luz infrarroja. Mientras que las comunicaciones por RF pueden penetrar muchos objetos, el IrDA está limitado a una línea de visibilidad.



Fuente: Domínguez, C. (2002). Redes inalámbricas.

Fig. 2.13 Comunicación IrDA de largo alcance por bits y caracteres

ii. SISTEMA COOPERATIVO DE INFRAESTRUCTURA VEHICULAR

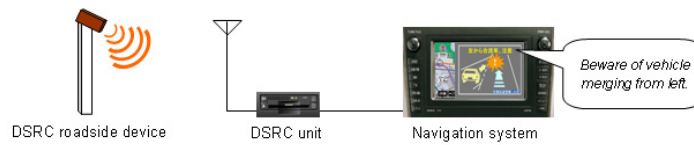
El Sistema Cooperativo de infraestructura Vehicular (SCIV) corresponde al compendio integrado de redes de alta tecnología, en las cuales existe un intercambio de información, tanto del automóvil que recibe esta información y provee una interpretación más exacta del comportamiento de la unidad.

El SCIV de última generación comprende las siguientes unidades:

Roadside Units

Las RSU son nodos fijos de la red ad-hoc que tienen la función básica de enviar, recibir y retransmitir paquetes para aumentar el rango de cobertura de la red, algo muy importante en las aplicaciones de seguridad vial. También pueden ofrecer acceso a internet al encontrarse conectadas a la red fija de algún operador. De esta manera, las RSU permiten a los vehículos estar conectados a la infraestructura fija, de modo que cualquier

unidad de aplicación puede conectarse.

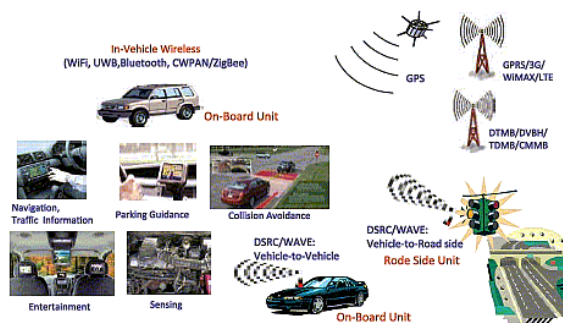


Fuente: Gomi, A. (2010). Diseño sistema de comunicaciones V2R

Fig. 2.14 Interconexión de Unidades Roadside

On Board Unit

Las OBU's son dispositivos electrónicos que se instalan en los vehículos para poder comunicarse con otros vehículos o con la infraestructura. Un ejemplo claro son los dispositivos con interfaz natural de usuario (NUI) con hardware externo.



Fuente: Gomi, A. (2010). Diseño sistema de comunicaciones V2R

Fig. 2.15 Unidades a bordo interconectados

iii. INFRAESTRUCTURA DE INTEGRACIÓN VEHICULAR VANET

Una de las áreas de investigación más activas en tecnología automotriz de los últimos años es la de VANET (vehicular ad hoc network), cuyo objetivo es desarrollar plataformas de comunicación entre vehículos en movimiento, la infraestructura vial y la de interpretación de datos.

Para ello se aprovecha las capacidades de cómputo y comunicaciones inalámbricas con que cuentan actualmente los automóviles y las correspondientes capacidades que pueden obtenerse a través de la infraestructura vial.

Las principales áreas de aplicación de las VANET son:

- **Seguridad activa:** La seguridad es uno de los temas primordiales en el desarrollo de tecnología automotriz, razón por la que este grupo de aplicaciones de las VANET es el de mayor interés. Su objetivo es hacer más segura la conducción de vehículos mediante la comunicación oportuna de señales de advertencia sobre una posible colisión, una velocidad excesiva de arribo a una curva, fallas en las condiciones del vehículo (frenos, luces, tren motriz, etc.). Estas aplicaciones pueden emplearse, incluso, para permitir que el vehículo intente de forma automática evitar el accidente o para que reaccione de la mejor manera, en caso de que éste sea inevitable.
- **Servicios públicos:** Ejemplos sobresalientes de este tipo de aplicaciones es el apoyo que pueden obtener los vehículos de emergencia (ambulancias, policía y cuerpos de rescate) mediante “sirenas virtuales”, anunciadas con anticipación a los demás vehículos.

El conductor puede, así mismo, ser advertido sobre una potencial infracción al reglamento de tránsito, por ejemplo, al entrar a una zona con límite de velocidad inferior a la actual o al anunciar vuelta en una zona no autorizada, etc.

- **Mejoras de conducción:** Este tipo de aplicaciones pueden emplearse para que los vehículos avisen un cambio en sus condiciones de movimiento (reducción o incremento de velocidad, cambio de carril, etc.) o su localización a los otros vehículos cercanos; también es posible recabar información de la infraestructura vial sobre la velocidad óptima de llegada a un semáforo para coincidir con la fase verde, zonas de embotellamiento o la disponibilidad de lugares de estacionamiento o permitir el diagnóstico remoto del vehículo, etc.

- **Negocios y entretenimiento móvil:** En este grupo se encuentran aplicaciones para facilitar la realización de transacciones durante el viaje, como pueden ser el pago de cuotas de peaje, descarga de contenidos multimedia, reservaciones, etc.

Las VANET, se basan en la restricción de que no puede asumirse la disponibilidad de un coordinador de la comunicación, aún cuando en algunos casos la infraestructura vial puede proporcionar puntos de acceso a una red fija y, por lo tanto a un sistema centralizado de comunicación. Un problema importante a resolver, al no contar con un servidor central, es cómo desarrollar un mecanismo eficaz de auto organización entre los pares en movimiento. Este movimiento, además de otras características como la baja altura de las antenas y la atenuación/reflexión de los cuerpos metálicos en movimiento crean condiciones adversas para la propagación de las señales de radio.

d. TECNOLOGÍAS DE COMUNICACIÓN

Las tecnologías de la información y la comunicación (TIC) pueden y deben desempeñar un papel clave como motor de la eficiencia de los sistemas mecánicos actuales en todos los campos de acción, como son la autotrónica, el diseño y el estudio-análisis de componentes y su interacción, promoviendo cambios en el comportamiento de los usuarios y los fabricantes de automotores, mejorando la eficiencia en el uso de los recursos innovadores y, paralelamente, reduciendo las deficiencias y problemas en su interpretación.

En el ámbito de la telemática vehicular, las redes de próxima generación (NGN) será la infraestructura sobre la que se desplegarán servicios como aviso de accidentes, notificaciones de tráfico o información de interés en ruta. El modelo de arquitectura NGN, caracterizado por planos independientes e interfaces abiertas para favorecer la convergencia de las

redes.

Éstas constituyen un claro caso de uso, ya que combinan diversas tecnologías de comunicación V2X y la provisión de múltiples servicios en un contexto exigente de movilidad.

La aplicación de las tecnologías de comunicación en la automoción, nos conlleva al análisis de redes WLAN, en malla, GSM, WiMax, WI FI y comunicaciones dedicadas de poco alcance.

i. REDES WLAN

Las redes inalámbricas de área local WLAN por sus siglas en inglés Wireless Local Area Network son redes que comúnmente cubren distancias de los 10 a los 100 metros.

Debido a que las LAN's a menudo son utilizadas para comunicaciones de una relativa alta capacidad de datos, normalmente tienen índices de datos más altos. Por ejemplo 802.11, una tecnología WLAN, tiene un ámbito nominal de 100 metros e índices de transmisión de datos de hasta 11 Mbps. Los dispositivos que normalmente utilizan WLAN's son los que tienen una plataforma más robusta y abastecimiento de potencia como son las computadoras personales en particular.

Como toda LAN, requiere de un medio físico para transmitir. Las WLANs utilizan las bandas de frecuencia de 2.4 GHz y 5-GHz de tipo:

- **Luz infrarroja (IR)**

- *Ventajas:*

- Emisores y receptores muy simples y baratos
- No interfiere con otros dispositivos de Radio Frecuencia.

- *Desventajas:*

- Poco Ancho de Banda

- Necesidad de comunicación “visual” Esta es una desventaja importante. Por ejemplo, no se podría comunicar un pc en una sala con una impresora que esté en otra sala. Esto limita mucho las posibilidades de comunicación entre dispositivos y da un aspecto de comunicación “de juguete”.
- Habitualmente comunicaciones sólo entre 2 interlocutores

- **Radio frecuencias (RFs)**

- *Ventajas:*

- Mayor área de cobertura
 - No necesita comunicación “visual” entre dispositivos
 - Mayor Ancho de Banda

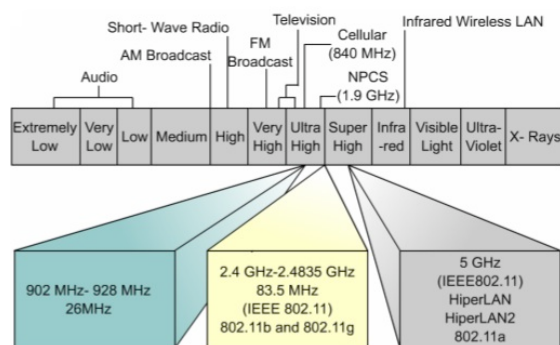
- *Desventajas:*

- Difícil de apantallar -> Interferencias

No solo interferencias entre diferentes dispositivos conectados a una red, sino también entre otro tipo de dispositivos independientes que generen campos electromagnéticos, por ejemplo, microondas.

- Rango de frecuencias limitado

Hoy día, el espectro radioeléctrico está ocupado casi al 100% así que se buscan huecos, pero como la gestión del espacio radioeléctrico es distinta en cada país, nos encontramos ante dificultades en la estandarización del espacio radioeléctrico a utilizar en una determinada tecnología.



Fuente: Domínguez, C. (2002). Redes inalámbricas.

Fig. 2.16 Frecuencias de utilización en redes WLAN

Arquitectura de WLAN IEEE 802.11.

La capa física proporciona una serie de servicios a la capa de acceso al medio (MAC). La capa física de servicios consiste en dos protocolos:

La comunicación entre MACs de diferentes estaciones se realizará a través de la capa física mediante una serie de puntos de acceso al servicio, donde la capa MAC invocará las primitivas de servicio.

Nivel de Acceso al Medio (Media Access Control MAC).

Para entender, MAC se define como subnivel inferior, provee el acceso compartido de las tarjetas de red al medio físico, es decir, define la forma en que se va a acceder al medio físico empleado en la red para el intercambio de datos.

Además, la capa de administración MAC controlará aspectos como sincronización y los algoritmos del sistema de distribución, que se define como el conjunto de servicios que precisa o propone el modo infraestructura.

VENTAJAS DE LA RED WLAN

- **Flexibilidad**

Dentro de la zona de cobertura de la red inalámbrica los nodos se podrán comunicar y no estarán atados a un cable para poder estar comunicados por el mundo.

- **Poca planificación**

Antes de cablear un edificio o unas oficinas se debe pensar mucho sobre la distribución física de las máquinas, mientras que con una red inalámbrica sólo nos tenemos que preocupar de que el edificio o las oficinas queden dentro del ámbito de cobertura de la red.

- **Diseño**

Los receptores son bastante pequeños y pueden integrarse dentro de un dispositivo y llevarlo en un bolsillo, etc.

- **Robustez**

Ante eventos inesperados que pueden ir desde un usuario que se tropieza con un cable o lo desenchufa, hasta un pequeño terremoto o algo similar. Una red cableada podría llegar a quedar completamente inutilizada, mientras que una red inalámbrica puede aguantar bastante mejor este tipo de percances inesperados.

INCONVENIENTES DE LA RED WLAN

- **Calidad de Servicio**

Las redes inalámbricas ofrecen una peor calidad de servicio que las redes cableadas. Estamos hablando de velocidades que no superan habitualmente los 10 Mbps, frente a los 100 que puede alcanzar una red normal y corriente. Por otra parte hay que tener en cuenta también la tasa de error debida a las interferencias.

- **Coste**

Aunque cada vez se está abaratando bastante aún sale bastante más caro. Peor aún no merece la pena debido a la poca calidad de servicio, falta de estandarización y coste.

- **Seguridad**

En dos vertientes:

- Por una parte seguridad e integridad de la información que se transmite. Este campo está bastante criticado en casi todos los estándares actuales, que, según dicen no se deben utilizar en entornos críticos en los cuales un “robo” de datos pueda ser

peligroso.

- Por otra parte este tipo de comunicación podría interferir con otras redes de comunicación (policía, bomberos, hospitales, etc.) y esto hay que tenerlo en cuenta en el diseño.

- **Soluciones Propietarias**

Como la estandarización está siendo bastante lenta, ciertos fabricantes han sacado al mercado algunas soluciones propietarias que sólo funcionan en un entorno homogéneo y por lo tanto estando atado a ese fabricante.

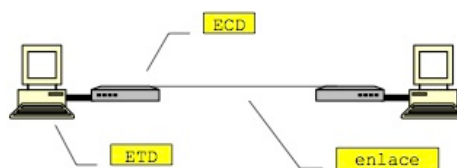
- **Restricciones**

Estas redes operan en un trozo del espectro radioeléctrico. Éste está muy saturado hoy en día y las redes deben amoldarse a las reglas que existan dentro de cada país.

ii. COMUNICACIONES DEDICADAS DE POCO ALCANCE

Es un camino de comunicación –trayecto- estáticamente definido entre dos sistemas que se comunican ya sea por un enlace físico determinado o bien por una traza lógicamente definida por completo dentro de un sistema de comunicaciones. Este sistema puede, de todos modos, ser multiplexado y/o conmutado.

Este circuito no existe físicamente como punto a punto, sino que es una definición lógica y se establece virtualmente sobre la red, predefiniendo un camino para la transmisión. Por esta razón, se lo conoce como Circuito Virtual Permanente.



Fuente: Universidad Tecnológica Nacional. Conceptos básicos de redes.

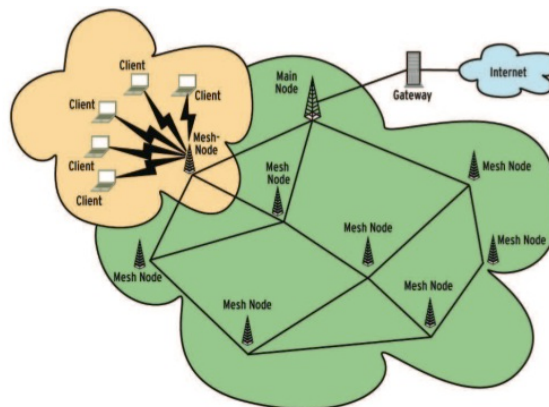
Fig. 2.17 Enlace dedicado típico

iii. REDES EN MALLA

En una red en malla, cada nodo está en contacto con sus vecinos directos. Los nodos pueden ser ordenadores, portátiles o dispositivos tales como teléfonos móviles con soporte de protocolo de interconexión.

En las mallas, los datos se sincronizan tan a menudo como se pueda, normalmente cada pocos segundos. Esta solución mantiene la dinámica de la red y permite reaccionar a las condiciones cambiantes debidas al movimiento de los nodos, los obstáculos o a las conexiones directas.

La malla evalúa cada ruta activa en base a criterios específicos y elige la mejor ruta. Por ello, el enrutamiento es una de las tareas más críticas y más costosas de las redes de malla.



Fuente: Universidad Tecnológica Nacional. Conceptos básicos de redes

Fig. 2.18 Red en malla con infraestructura por nodos.

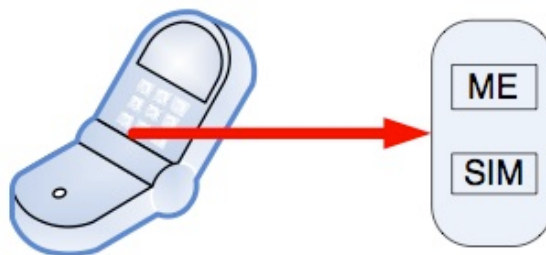
iv. GSM

La red de tipo GSM se diseñó para incluir una amplia variedad de servicios que incluyen transmisiones de voz, datos y servicios de manejo de mensajes codificados entre unidades móviles o cualquier otra unidad portátil.

Componentes de GSM.

Los componentes principales GSM son:

- **El centro de conmutación móvil (MSC, Mobile Switching Center)**, es el corazón de todo sistema GSM y se encarga de establecer, gestionar y despejar conexiones, así como de enrutar los enlaces a la célula correcta. El MSC proporciona la interfaz con el sistema y presta servicios de determinación de cargos y contabilidad.
- **La célula**, cuyo tamaño es de aproximadamente 35 km.
- **La unidad móvil (MS, Mobile Station)**. La MS (*Mobile Station*, Estación Móvil) es un dispositivo que recibe y transmite las señales de radio con la BSS (*Base Station Subsystem*, Subsistema de Estación Base).



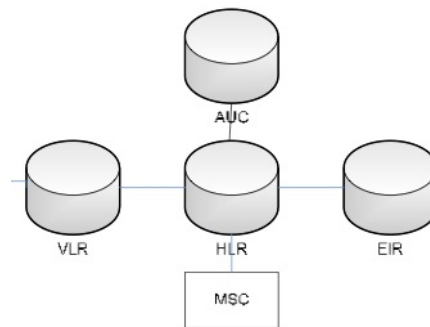
MS: Estación Móvil

Fuente: Bellido, F. (2004). Comunicación inalámbrica GSM.

Fig. 2.19 Composición Estación Móvil

- **La estación de transmisión-recepción base (BTS, Base Transceiver Station)**. Establece la interfaz a la unidad móvil. Contiene los transmisores/receptores que sirven a una celda.
- **La HLR (Home Location Register)** es una base de datos que proporciona información sobre el usuario, su base de suscripción de origen y los servicios suplementarios que se le proveen.

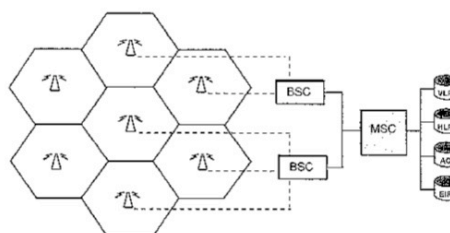
- **El VLR (Visitor Location Register)** es también una base de datos que contiene información sobre la situación de encendido/apagado de las estaciones móviles y si se han activado o desactivado cualesquiera de los servicios suplementarios.



Fuente: Bellido, F. (2004). Comunicación inalámbrica GSM.

Fig. 2.20 Arquitectura de intercomunicación HLR - VLR

- **El centro de validación (AC o AUC, Authentication Center)** que sirve para proteger a cada suscriptor contra un acceso no autorizado o contra el uso de un número de suscripción por personas no autorizadas; opera en relación estrecha con el HLR.
- **El registro de identidad del equipo (EIR, Equipment Identity Register)** que sirve para registrar el tipo de equipo que existe en la estación móvil y también puede desempeñar funciones de seguridad como bloqueo de señales que se ha determinado que emanan de estaciones móviles robadas, así como evitar que ciertas estaciones que no han sido aprobadas por el proveedor de la red que use ésta.



Fuente: Bellido, F. (2004). Comunicación inalámbrica GSM.

Fig. 2.21 Esquema de componentes GSM

v. WIMAX

La tecnología estandarizada por el IEEE bajo el apelativo 802.16, por lo común conocida como WiMAX, es considerada el hermano mayor del Wi-Fi. Eso responde al hecho que el WiMAX promete más alcance, más anchura de lado y más potencia que el Wi-Fi, acompañadas de más funcionalidad en términos, especialmente, de calidad de servicio y seguridad.

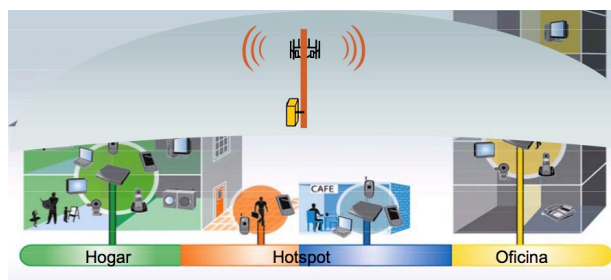
La primera diferencia importante entre el Wi-Fi y el WiMAX radica en los diferentes ámbitos de aplicación para los cuales fueron diseñadas. El Wi-Fi surgió como una tecnología para cubrir los últimos metros del acceso y permitir al usuario librarse de la tiranía de los hilos en entornos de oficina o al hogar.



Fuente: Manual de tecnologías Wi-Fi y WiMax

Fig. 2.22 Red de Comunicaciones WiMax

El WiMAX, por contra, está destinado a ser la evolución e implementación de radioenlaces punto a punto.



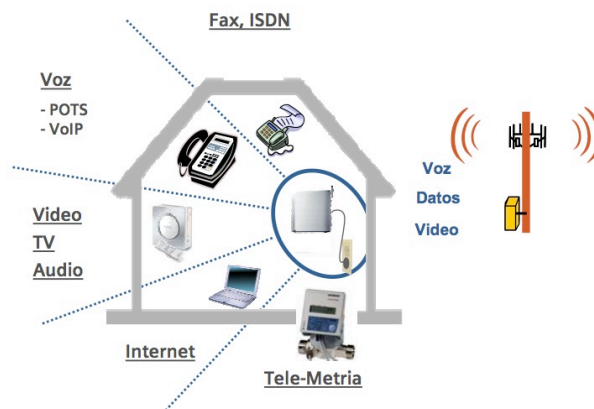
Fuente: Manual de tecnologías Wi-Fi y WiMax

Fig. 2.23 Configuración WiMax punto-punto

Topologías de redes WiMAX

Las topologías propias de una red WiMAX pueden ser:

- **De distribución:** análogo a las redes de infraestructura Wi-Fi. La estación base ocupa el rol del punto de acceso y centraliza el acceso de los usuarios distribuidos por la celda en la red fija y a la salida hacia Internet. Además, se ocupa de gestionar el canal y distribuir los recursos en función de las necesidades de calidad de cada usuario.
- **Malladas:** análogo a las redes ad hoc, permiten la comunicación directamente entre estaciones móviles sin necesidad de pasar por una estación base. La gestión de los recursos es entonces distribuida. Un inconveniente de esta última característica es que, por razones de compatibilidad en los sistemas de gestión, a menudo todos los equipos han de ser del mismo fabricante.

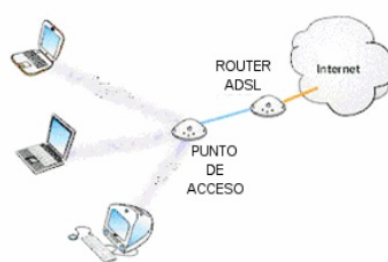


Fuente: Manual de tecnologías Wi-Fi y WiMax

Fig 2.24 Red Mallada WiMax

vi. WI FI

Wi-Fi (Wireless Fidelity) es una de las tecnologías de comunicación inalámbrica (sin cables - wireless) más extendidas. También se conoce como IEEE 802.11.



Fuente: Manual de tecnologías Wi-Fi y WiMax

Fig. 2.25 Configuración típica WI-FI

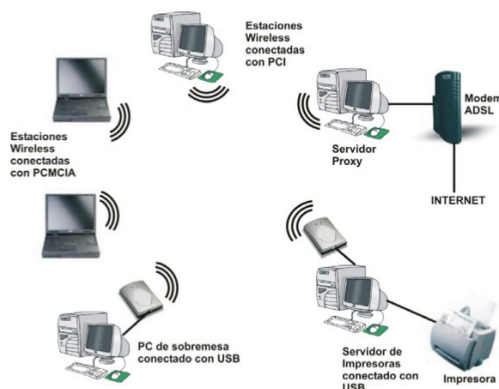
La denominación Wi-Fi (Wireless-Fidelity) aplicada al protocolo inalámbrico IEEE 802.11b significa que, vía radio, mantiene con fidelidad las características de un enlace Ethernet cableado.

Topología de Red WI-FI

Como en la mayoría de redes LAN, en las redes WI-FI podemos encontrar dos tipos de topología, Red Ad-Hoc y Red Modo Infraestructura.

Una red "Ad Hoc" consiste en un grupo de ordenadores que se comunican cada uno directamente con los otros a través de las señales de radio sin usar un punto de acceso.

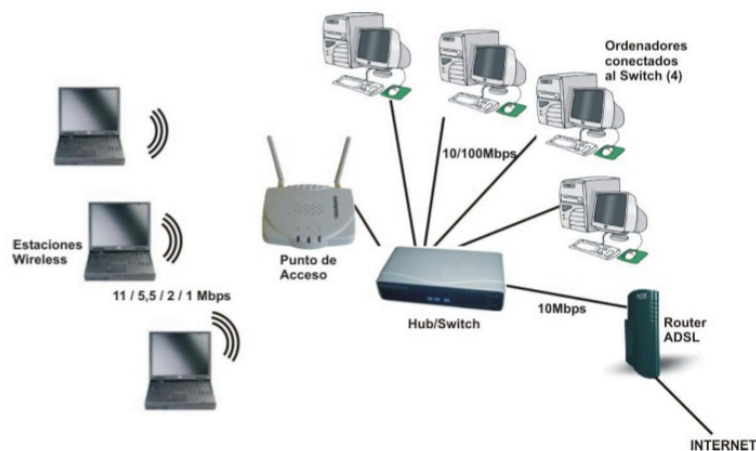
Los ordenadores de la red inalámbrica que quieren comunicarse entre ellos necesitan usar el mismo canal radio y configurar un identificador específico de WiFi (denominado ESSID) en "Modo Ad Hoc".



Fuente: Manual de tecnologías Wi-Fi y WiMax

Fig. 2.26 Red WI-FI Punto a Punto (Ad Hoc)

Se conoce como configuración "Modo Infraestructura" a la forma típica de trabajar cuando se utilizan Puntos de Acceso (AP). Si queremos conectar nuestra tarjeta WI-FI a uno de ellos, debemos configurarla para trabajar en este modo de trabajo. Es mas eficaz que la red ad-hoc, en la que los paquetes "se lanzan al aire, con la esperanza de que lleguen al destino..", mientras que el modo Infraestructura gestiona y se encarga de llevar cada paquete a su sitio mejorando, además, la velocidad.



Fuente: Manual de tecnologías Wi-Fi y WiMax

Fig. 2.27 Red WI-FI de Infraestructura

COMPONENTES BÁSICOS

Los elementos que forman una red WI-FI son los siguientes:

- **Punto de acceso (AP):** Es el dispositivo que gestiona la información transmitida y la hace llegar a destino. Asimismo, proporciona la unión entre la red WI-FI y la red fija.
- **Antena:** Las antenas son los elementos que envían al aire señales en forma de ondas electromagnéticas que contienen la información dirigida en el dispositivo de destino; y a la vez, captan del aire las señales de las cuales se extraerá la información que llega de otro dispositivo.

- **Dispositivo externo WI-FI:** La tarjeta WI-FI es una tarjeta de red de área local (CHAL) que cumple la certificación WI-FI y permite por lo tanto la conexión de un terminal de usuario en una red 802.11.
- **Antena de usuario y conector pigtail:** La antena de usuario proporciona la cobertura necesaria a un usuario para el acceso a la red WI-FI. El conector pigtail es un tipo de cable que conecta y adapta la tarjeta WI-FI y la antena del usuario.

Limitaciones tecnológicas de la familia 802.11 WI-FI

Independientemente de la banda de frecuencia en que trabajan, todos los estándares de la subfamilia 802.11 comparten algunas limitaciones que es conveniente conocer antes de tomar una decisión sobre coberturas, alcance o velocidades que se pueden alcanzar.

Estas limitaciones son cinco:

- **Alcance:** Aunque comercialmente se habla típicamente de un alcance de hasta 100 metros, este dato depende, en primer lugar, de la ubicación y de la presencia de obstáculos en el camino entre el punto de acceso y el terminal, y en segundo lugar, de las condiciones meteorológicas y de las interferencias.

Así, en espacio abierto, con buenas condiciones meteorológicas y antenas exteriores de los terminales, este alcance puede ser bastante superior. Sin embargo, en el interior de un edificio, donde las paredes representan un obstáculo muy importante, la distancia será notablemente inferior. Asimismo, si hay otras redes Wi-Fi próximas, o sencillamente otras fuentes de interferencias, es también mucho probable que las distancias disminuyan.

- **Anchura de banda:** Nominalmente, los diferentes estándares pueden alcanzar, físicamente (es decir, en el canal aéreo, descontando cualquier ineficiencia que puedan introducir los protocolos superiores), las velocidades mencionadas en la mesa/tabla presentada a el apartado anterior. Ahora bien, a causa del efecto de los protocolos necesarios para transportar la información de usuario sobre el canal aéreo, la velocidad útil es mucho menor. Además, en función de las condiciones del entorno y, por lo tanto, de la calidad de cada comunicación entre un terminal y el punto de acceso, la anchura de lado de esta comunicación se adapta, con el fin de utilizar codificaciones más robustas ante interferencias y/o errores. Es por eso que a veces nos encontremos con una conexión con el punto de acceso de 11 Mbps, otros en 5 Mbps, en 2 Mbps o, incluso, en 1 Mbps.
- **Calidad de servicio:** No todo el tráfico tiene la misma importancia desde el punto de vista de cada usuario. Así, se puede considerar que una llamada de VoIP tendría que tener prioridad sobre una transferencia de ficheros.
- **Seguridad:** En un principio, las redes Wi-Fi no presentaban mecanismos de seguridad muy sofisticados, ya que el énfasis se puso en cómo transmitir datos sobre el aire, que era un desafío tecnológico más urgente. Con el éxito de esta tecnología, sin embargo, y la publicación de las debilidades de los mecanismos de seguridad originales, se hizo necesario introducir mejoras en este aspecto con seguridad de tipo WEP y WPA.
- **Movilidad:** Popularmente, se considera que las redes Wi-Fi son móviles, ya que no hay que conectarse desde una ubicación fija para acceder a los servicios que nos ofrece, y además se puede ir caminando y navegando por Internet o leyendo el correo electrónico al mismo tiempo. Estrictamente hablando, eso se considera itinerancia, y no movilidad. De hecho, no es posible utilizar una red Wi-Fi desde un vehículo en movimiento a velocidad normal, por razones físicas asociadas a la

velocidad. Además, incluso cuando nos movemos a baja velocidad (caminando), a causa del escaso alcance de cobertura de un punto de acceso, rápidamente tenemos que establecer conexión con otro punto de acceso, la cual cosa implica "saltar" del uno al otro.

e. SEGURIDAD Y PRIVACIDAD

La irrupción de la nueva tecnología de comunicación basada en redes inalámbricas ha proporcionado nuevas expectativas de futuros para el desarrollo de sistemas de comunicación, así como nuevos riesgos.

Pero como todas la nuevas tecnologías en evolución, presenta unos riesgos debidos al optimismo inicial y en la adopción de la nueva tecnología sin observar los riesgos inherentes a la utilización de un medio de transmisión tan 'observable' como son las ondas de radio.

i. RIESGOS DE LAS REDES INALAMBRICAS

La utilización del aire como medio de transmisión de datos mediante la propagación de ondas de radio ha proporcionado altos riesgos de seguridad para las redes inalámbricas. La salida de estas ondas de radio fuera del lugar donde está ubicada la red permite la exposición de los datos a posibles intrusos que podrían obtener información sensible a la empresa o red de comunicación y a la seguridad informática de la misma.

Varios son los riesgos derivables de este factor. Por ejemplo, se podría perpetrar un ataque por inserción, bien de un usuario no autorizado o por la ubicación de un punto de acceso ilegal más potente que capte las estaciones cliente en vez del punto de acceso legítimo, interceptando la red inalámbrica.

También sería posible crear interferencias y una más que posible denegación de servicio con solo introducir un dispositivo que emita ondas de radio a una frecuencia de 2.4GHz (frecuencia utilizada por las redes

inalámbricas).

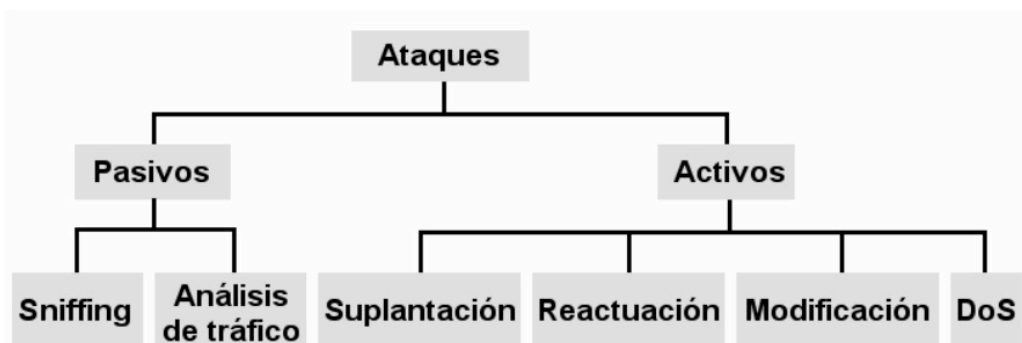
La posibilidad de comunicarnos entre estaciones cliente directamente, sin pasar por el punto de acceso permitiría atacar directamente a una estación cliente, generando problemas si esta estación cliente ofrece servicios TCP/IP o comparte ficheros. Existe también la posibilidad de duplicar las direcciones IP o MAC de estaciones cliente legítimas.

Los puntos de acceso están expuestos a un ataque de Fuerza bruta para averiguar los passwords, por lo que una configuración incorrecta de los mismos facilitaría la irrupción en una red inalámbrica por parte de intrusos.

A pesar de los riesgos anteriormente expuestos, existen soluciones y mecanismos de seguridad para impedir que cualquiera con los materiales suficientes pueda introducirse en una red.

ii. TIPOS DE ATAQUES

Para poder definir nuestros mecanismos de defensa, en primera instancia debemos analizar los tipos de ataques a los cuales nos vamos a enfrentar, ya sea de manera pasiva (indetectable inmediatamente) o de manera activa (detección inmediata).



Fuente: Grupo de investigación.

Fig. 2.28 Tipos de Ataques a redes inalámbricas

ATAQUES PASIVOS

- **Sniffing**

- El tráfico de redes inalámbricas puede espiarse con mucha más facilidad que en una LAN.
- Basta con disponer de un portátil con una tarjeta inalámbrica.
- El tráfico que no haya sido cifrado, será accesible para el atacante y el cifrado con WEP también

- **Análisis de tráfico**

El atacante obtiene información por el mero hecho de examinar el tráfico y sus patrones: a qué hora se encienden ciertos equipos, cuánto tráfico envían, durante cuánto tiempo, etc.

ATAQUES ACTIVOS

- **Suplantación**

- Mediante un sniffer para hacerse con varias direcciones MAC válidas.
- El análisis de tráfico le ayudará a saber a qué horas debe conectarse suplantando a un usuario u otro.
- Instalar puntos de acceso ilegítimos (*rogue*) para engañar a usuarios legítimos para conectar a éste, en lugar del autorizado.

- **Modificación**

El atacante borra, manipula, añade o reordena los mensajes transmitidos.

- **Reactuación**

Inyectar en la red paquetes interceptados utilizando un sniffer para repetir operaciones que habían sido realizadas por el usuario legítimo.

- **Denegación de servicio**

El atacante puede generar interferencias hasta que se produzcan

tantos errores en la transmisión que la velocidad caiga a extremos inaceptables o la red deje de operar en absoluto.

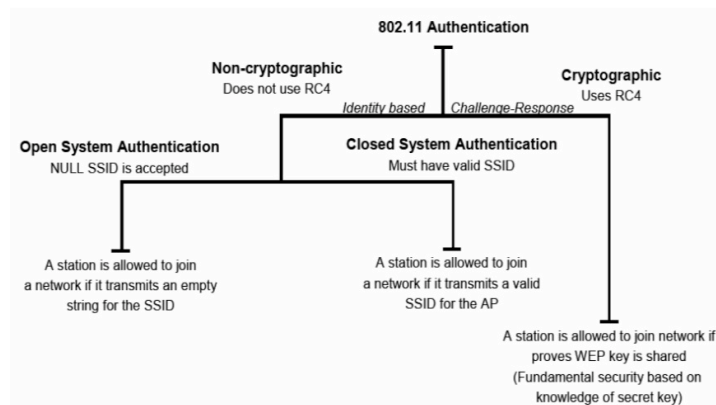
- **Otros ataques:** inundar con solicitudes de autenticación, solicitudes de de-autenticación de usuarios legítimos, ramas RTS/CTS para silenciar la red, etc.

iii. MECANISMOS DE SEGURIDAD

WEP (Wired Equivalent Protocol)

WEP comprime y cifra los datos que se envían a través de las ondas de radio. Con WEP, la tarjeta de red encripta el cuerpo y la señal de cada trama 802.11 antes de la transmisión utilizando el algoritmo de encriptación RC4 proporcionado por RSA Security. La estación receptora, sea un punto de acceso o una estación cliente es la encargada de descifrar la trama.

La vulnerabilidad de WEP reside en la insuficiente longitud del Vector de Inicialización (IV) y lo estáticas que permanecen las llaves de cifrado, pudiendo no cambiar en mucho tiempo. Si utilizamos solamente 24 bits, WEP utilizará el mismo IV para paquetes diferentes, pudiéndose repetir a partir de un cierto tiempo de transmisión continua. Es a partir de entonces cuando un intruso puede, una vez recogido suficientes tramas, determinar incluso la llave compartida.



Fuente: Grupo de investigación.

Fig. 2.29 Diagrama de flujo de seguridad WEP

A pesar de todo, WEP proporciona un mínimo de seguridad para pequeños negocios o instituciones educativas, si no está deshabilitada, como se encuentra por defecto en los distintos componentes inalámbricos.

OSA (Open System Authentication)

Es otro mecanismo de autenticación definido por el estándar 802.11 para autenticar todas las peticiones que recibe. El principal problema que tiene es que no realiza ninguna comprobación de la estación cliente, además las tramas de gestión son enviadas sin encriptar, aún activando WEP, por lo tanto es un mecanismo poco fiable.

ACL (Access Control List)

Este mecanismo de seguridad es soportado por la mayoría de los productos comerciales. Utiliza, como mecanismo de autenticación, la dirección MAC de cada estación cliente, permitiendo el acceso a aquellas MAC que consten en la Lista de Control de Acceso.

CNAC (Closed Network Access Control)

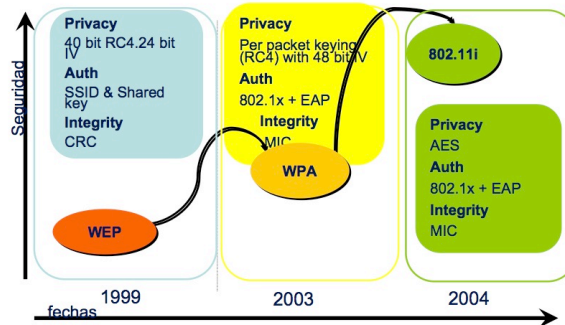
Este mecanismo pretende controlar el acceso a la red inalámbrica y permitirlo solamente a aquellas estaciones cliente que conozcan el nombre de la red (SSID) actuando este como contraseña.

iv. MEDIDAS DE SEGURIDAD RECOMENDADAS

Como primera medida, se debe separar la red de la organización en un dominio público y otro privado. Los usuarios que proceden del dominio público (los usuarios de la red inalámbrica) pueden ser tratados como cualquier usuario de Internet (externo a la organización).

Así mismo, instalar cortafuegos y mecanismos de autenticación entre la red inalámbrica y la red clásica, situando los puntos de acceso delante del cortafuegos.

Como contradicción, es recomendable no utilizar excesivas normas de seguridad por que podría reducir la rapidez y la utilidad de la red inalámbrica.

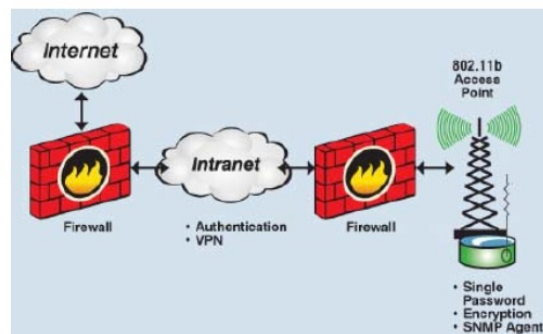


Fuente: Grupo de investigación.

Fig. 2.30 Diseños de seguridad

Algunas de estas recomendaciones podrían ser, aún a riesgo de resultar extremadas:

- Utilizar cobertura metálica en las paredes exteriores, en el caso del entorno vehicular en los componentes de la carrocería.
- Vidrio aislante térmico (atenúa las señales de radiofrecuencia).
- Poner dispositivos WLAN lejos de las paredes exteriores.
- Utilizar pintura metálica.
- Limitar el poder de una señal cambiando la atenuación del transmisor.



Fuente: Grupo de investigación.

Fig. 2.31 Firewalls antes y después de la VPN

2.1.2 DESARROLLO Y APLICACIONES

Una de las aplicaciones más interesantes de la tecnología sin hilos es la posibilidad de cambiar de red sin necesidad de modificar la configuración del sistema, es decir, ofrecer Itinerancia (roaming) entre los diferentes puntos de acceso.

Esta característica permite que el usuario pueda moverse por el territorio sin tener que hacer ninguna modificación en su dispositivo electrónico de la misma manera que en el entorno a la red de comunicación de datos, es decir que no se pierde la cobertura y que se permite la movilidad entre diferentes áreas inalámbricas.

A fin de que la red permita itinerancia se ha desarrollado una aplicación que permita configurar los puntos de acceso para que trabajen en diferentes canales de frecuencia, de manera que no se produzcan problemas de funcionamiento en aquellas zonas en que se crucen las coberturas de diferentes puntos de acceso, y el usuario "salte" de manera transparente de uno al otro.

Los resultados más inmediatos, desde el punto de vista del usuario, serán la posibilidad de proveer de calidad de servicio a ciertos tipos de tráfico, especialmente la comunicación de voz y datos (VoIP).

Más a largo plazo se encuentran los trabajos, que acaban de empezar, para alcanzar la movilidad en vehículos (802.11p), mecanismos más rápidos (y estandarizados) de itinerancia entre puntos de acceso (802.11r) y para simplificar la interoperación de redes Wi-Fi con otras tecnologías (802.11u).

La implantación de los sistemas inalámbricos inteligentes contribuye significativamente a resolver problemas en los sistemas de transporte. Sin embargo, no hay que perder de vista los procesos desarrollados en algunos países, donde la experiencia muestra que la explotación de las tecnologías

inalámbrica sólo tiene un impacto significativo si los gobiernos tienen voluntad de colaborar en el desarrollo de las arquitecturas necesarias.

Los sistemas embarcados en los vehículos tienen la principal función de interactuar con el conductor con el objetivo de proporcionarle ayuda en la tarea de conducción, mantenimiento y preparación.

Dependiendo del nivel de ayuda que le aporten, estos sistemas en desarrollo se dividen en:

- **Sistemas avanzados de ayuda a la conducción**, ADAS – Advanced Driver Assistance Systems, que proveen información a nivel táctico y operacional de la conducción (ej.: control de velocidad).
- **Sistemas de información en el vehículo**, IVIS – In Vehicle Information Systems, que aportan información de tareas secundarias (ej.: sistema de navegación, diagnóstico).

Estos equipos no sólo se utilizan en el transporte privado, también se incorporan en el transporte público ya que son numerosas las empresas que deciden probar y poner en práctica los sistemas en sus servicios públicos.

2.2. SISTEMAS TÁCTILES APPLE Y SOFTWARE APLICADO CON INTERFAZ ÚNICA DE USUARIO (NUI)

2.2.1. SOFTWARE APLICADO NUI

La forma en la que el usuario interactúa con su sistema operativo depende mucho de la interfaz de usuario, pero a pesar de que esta suele ser agradable, puede tornarse aburrida si no se la modifica de vez en cuando.

Más allá de su finalidad esencial de interacción entre hombre y máquina, las interfaces de usuario conforman también la cara visible o escaparate de las aplicaciones tal y como las percibe el cliente final que las usa.

Como consecuencia directa, las interfaces de usuario entregadas en productos finales están cada vez más y más cuidadas para ofrecer al usuario sensaciones de seguridad, fiabilidad, ergonomía, sencillez de uso y precisión en la aplicación suministrada. Todas estas cualidades se transmiten subliminalmente a través de la interfaz de usuario.

a. INTRODUCCIÓN Y DESARROLLO

La creciente popularidad de los ordenadores y su uso cotidiano han cambiado la relación de los usuarios frente a la máquina y, por ende, lo que los usuarios esperan de ella. Nuevas formas de ordenadores como asistentes para datos personales (PDA), teléfonos móviles, sistemas empotrados o nuevas generaciones de electrodomésticos comienzan a popularizarse, obligándonos a interactuar con sus interfaces que no siempre son todo lo naturales ni intuitivas que uno hubiera deseado.

El desarrollo de interfaces de usuario es una tarea que consume grandes cantidades de recursos. Por otro lado, las técnicas de modelado conceptual han demostrado ser muy valiosas para incrementar el nivel de abstracción a la hora de construir sistemas, sin embargo, históricamente dejaban de lado el modelado de la interfaz de usuario.

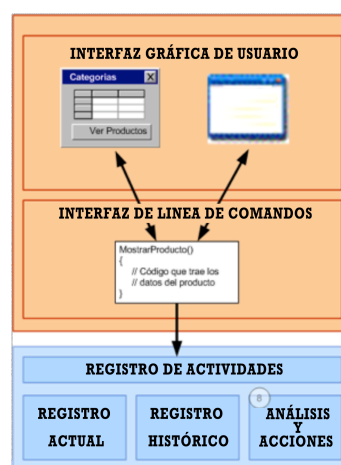
Los usuarios exigen cualidades al software destinadas a facilitar su trabajo, ahorrar tiempo –de uso y de aprendizaje–, evitar y corregir los errores. Los procesos de desarrollo de aplicaciones dedican cada vez mayores esfuerzos a diseñar y mantener las interfaces de usuario de sus aplicaciones. Gran parte de este trabajo se realiza a mano por diseñadores y programadores que crean y dotan de funcionalidad a las interfaces construidas. Este proceso consume muchos recursos: personal, herramientas, tiempo y dinero.

Si tales requisitos de interfaz de usuario logran ser capturados de una manera precisa y se establece una semántica clara para su aplicación, se abre la puerta al ámbito de la implementación de las interfaces de usuario. Es decir, la producción de modo automático de interfaces de usuario que cumplen con los requisitos expresados.

b. CONFIGURACIÓN

El usuario de una NUI dirige el funcionamiento de este mediante instrucciones denominadas genéricamente entradas. Las entradas se introducen mediante diversos dispositivos, por ejemplo un teclado, y se convierten en señales electrónicas que pueden ser procesadas por una computadora. Estas señales se transmiten a través de circuitos conocidos como buses, que son coordinadas y controladas por la UCP o CPU, el microprocesador que realiza las funciones aritméticas y lógicas) y por un soporte lógico conocido como sistema operativo.

Una vez que la CPU ha ejecutado las instrucciones indicadas por el usuario, puede comunicar los resultados enviando señales electrónicas, o salidas, que se transmiten de vuelta por el bus a uno o más dispositivos de salida, como por ejemplo una impresora o un monitor.



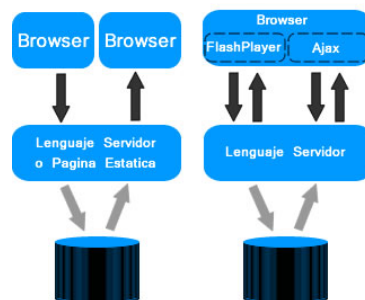
Fuente: Chile TSD Datacom.

Fig. 2.32 Configuración Interfaz de Usuario

Además de la velocidad del ordenador, también son consideraciones importantes la eficacia y facilidad de uso del soporte lógico y el diseño ergonómico de los componentes físicos. La ergonomía trata de optimizar las acciones de las personas en relación con su entorno. En el caso de los ordenadores, consiste en hacer que los dispositivos de entrada y salida puedan usarse de forma fácil, cómoda y eficiente.

El configuración de la NUI por parte del usuario y el ordenador suele efectuarse a través de una interfaz de línea de comandos o de una interfaz gráfica de usuario (GUI, siglas en inglés). Las interfaces de línea de instrucciones exigen que el usuario introduzca instrucciones breves mediante un teclado para dirigir las acciones de la computadora. Las GUI emplean ventanas para organizar archivos y aplicaciones representadas por iconos (pequeñas imágenes) y menús que presentan una lista de instrucciones. El usuario manipula directamente estos objetos visuales en el monitor señalándolos, seleccionándolos y arrastrándolos o moviéndolos con un mouse o una bola apuntadora.

El uso de las GUI es más fácil de aprender que el de las interfaces de línea de instrucciones, puesto que las instrucciones deben ser memorizadas y suelen variar según los distintos sistemas informáticos. Sin embargo, la introducción de instrucciones con una GUI es más lenta, por lo que las GUI suelen tener la opción de emplear un sistema equivalente al de línea de instrucciones como alternativa rápida para los usuarios más expertos.



Fuente: Chile TSD Datacom.

Fig. 2.33 Interfaz gráfica de usuario

c. IMPLEMENTACIÓN

Existen tres puntos de vista distintos en la implementación de la NUI: el del usuario, el del programador y el del diseñador (analogía de la construcción de una casa). Cada uno tiene un modelo mental propio de la interfaz, que contiene los conceptos y expectativas acerca de la misma, desarrollados a través de su experiencia.

El modelo permite explicar o predecir comportamientos del sistema y tomar las decisiones adecuadas para modificar el mismo. Los modelos subyacen en la interacción con las computadoras, de ahí su importancia.

Modelo del usuario: El usuario tiene su visión personal del sistema, y espera que éste se comporte de una cierta forma. Se puede conocer el modelo del usuario estudiándolo, ya sea realizando tests de usabilidad, entrevistas, o a través de una realimentación. Una interfaz debe facilitar el proceso de crear un modelo mental efectivo.

Para ello son de gran utilidad las metáforas, que asocian un dominio nuevo a uno ya conocido por el usuario. Un ejemplo típico es la metáfora del escritorio, común a la mayoría de las interfaces gráficas actuales.

Modelo del diseñador: El diseñador mezcla las necesidades, ideas, deseos del usuario y los materiales de que dispone el programador para diseñar un producto de software. Es un intermediario entre ambos.

El modelo del diseñador describe los objetos que utiliza el usuario, su presentación al mismo y las técnicas de interacción para su manipulación. Consta de tres partes: presentación, interacción y relaciones entre los objetos.

La presentación es lo que primero capta la atención del usuario, pero más tarde pasa a un segundo plano, y adquiere más importancia la interacción con el producto para poder satisfacer sus expectativas. La presentación no

es lo más relevante y un abuso en la misma (por ejemplo, en el color) puede ser contraproducente, distrayendo al usuario.

La segunda parte del modelo define las técnicas de interacción del usuario, a través de diversos dispositivos.

La tercera es la más importante, y es donde el diseñador determina la metáfora adecuada que encaja con el modelo mental del usuario. El modelo debe comenzar por esta parte e ir hacia arriba. Una vez definida la metáfora y los objetos del interfaz, los aspectos visuales saldrán de una manera lógica y fácil.

Estos modelos deben estar claros para los participantes en el desarrollo de un producto, de forma que se consiga una interfaz atractiva y a la vez efectiva para el trabajo con el programa.

Una interfaz no es simplemente una cara bonita; esto puede impresionar a primera vista pero decepcionar a la larga. Lo importante es que el programa se adapte bien al modelo del usuario, cosa que se puede comprobar utilizando el programa más allá de la primera impresión.

Modelo del programador: Es el más fácil de visualizar, al poderse especificar formalmente. Está constituido por los objetos que manipula el programador, distintos de los que trata el usuario (ejemplo: el programador llama base de datos a lo que el usuario podría llamar agenda). Estos objetos deben esconderse del usuario.

Los conocimientos del programador incluyen la plataforma de desarrollo, el sistema operativo, las herramientas de desarrollo y especificaciones. Sin embargo, esto no significa necesariamente que tenga la habilidad de proporcionar al usuario los modelos y metáforas más adecuadas. Muchos no consideran el modelo del usuario del programa, y sí sus propias expectativas acerca de cómo trabajar con la computadora.

2.2.2 PERSONALIZACIÓN DE DISPOSITIVOS

La gran mayoría de las tecnologías de pantalla táctil más significativas fueron patentadas durante las décadas de los 1970 y 1980 actualmente han expirado. Este hecho ha permitido que desde entonces los diseños que utilizan dichas tecnologías no estén sujetos a royalties, lo que ha permitido que los dispositivos táctiles se hayan extendido más fácilmente.

Con la creciente aceptación de multitud de productos con una pantalla táctil integrada, el coste marginal de esta tecnología ha sido rutinariamente absorbido en los productos que las incorporan haciendo que prácticamente desaparezca.

Como ocurre habitualmente con cualquier tecnología, el hardware y el software asociado a las pantallas táctiles ha alcanzado un punto de madurez suficiente después de más de tres décadas de desarrollo, lo que le ha permitido que actualmente tengan grado muy alto de fiabilidad.

Como tal, las pantallas táctiles pueden hallarse en la actualidad en aviones, automóviles, consolas, sistemas de control de maquinaria y dispositivos de mano de cualquier tipo.

a. DISPOSITIVOS TÁCTILES

Lo primero que hay que aclarar es en qué consiste eso de Táctil. Una interfaz táctil, en cuanto a hardware se refiere, no es más que una interfaz de ratón (que, por cierto, hace años que existe, no es nada nuevo, ya que datan de 1971), es decir, consiste en una película más o menos fina sobre la pantalla que hace las veces de ratón, o más bien del Touchpad de un portátil, ya que su funcionamiento es en realidad más parecido a este último.

Una pantalla táctil es una pantalla que mediante un toque directo sobre su superficie permite la entrada de datos y órdenes al dispositivo. A su vez,

actúa como periférico de salida, mostrando los resultados introducidos previamente. Este contacto también se puede realizar con lápiz u otras herramientas similares.



Fuente: Grupo de investigación.

Fig 2.34 Pantalla Táctil

Actualmente hay pantallas táctiles que pueden instalarse sobre una pantalla normal. Así pues, la pantalla táctil puede actuar como periférico de entrada y periférico de salida de datos, así como emulador de datos interinos erróneos al no tocarse efectivamente

Tecnologías

Hay diferentes tecnologías de implementación de las pantallas táctiles:

Resistiva

Una pantalla táctil resistiva está formada por varias capas. Las más importantes son dos finas capas de material conductor entre las cuales hay una pequeña separación. Cuando algún objeto toca la superficie de la capa exterior, las dos capas conductoras entran en contacto en un punto concreto. De esta forma se produce un cambio en la corriente eléctrica que permite a un controlador calcular la posición del punto en el que se ha tocado la pantalla midiendo la resistencia. Algunas pantallas pueden medir, aparte de las coordenadas del contacto, la presión que se ha ejercido sobre la misma.



Fuente: Grupo de investigación.

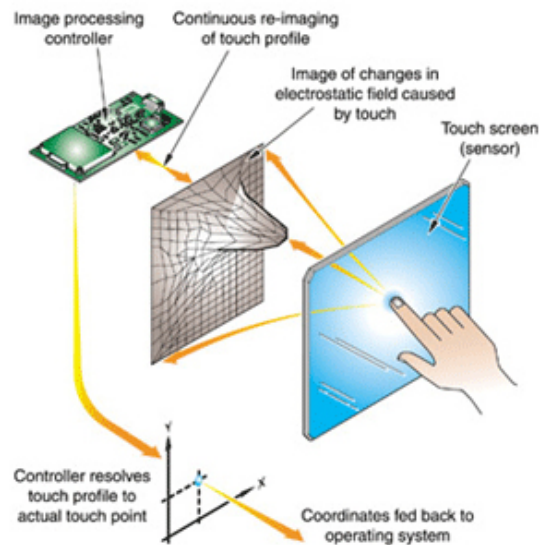
Fig. 2.35 Componentes Pantalla Táctil Resistiva

Onda Acústica Superficial

La tecnología de onda acústica superficial (denotada a menudo por las siglas SAW, del inglés Surface Acoustic Wave) utiliza ondas de ultrasonidos que se transmiten sobre la pantalla táctil. Cuando la pantalla es tocada, una parte de la onda es absorbida. Este cambio en las ondas de ultrasonidos permite registrar la posición en la que se ha tocado la pantalla y enviarla al controlador para que pueda procesarla.

Capacitivas

Una pantalla táctil capacitiva está cubierta con un material, habitualmente óxido de indio y estaño que conduce una corriente eléctrica continua a través del sensor. El sensor por tanto muestra un campo de electrones controlado con precisión tanto en el eje vertical como en el horizontal, es decir, adquiere capacitancia. Cuando el campo de capacitancia normal del sensor (su estado de referencia) es alterado por otro campo de capacitancia, como puede ser el dedo de una persona, los circuitos electrónicos situados en cada esquina de la pantalla miden la 'distorsión' resultante en la onda senoidal característica del campo de referencia y envía la información acerca de este evento al controlador para su procesamiento matemático.



Fuente: Grupo de investigación.

Fig. 2.36 Pantalla Táctil Capacitiva

Infrarrojos

Las pantallas táctiles por infrarrojos consisten en una matriz de sensores y emisores infrarrojos horizontales y verticales. En cada eje los receptores están en el lado opuesto a los emisores de forma que al tocar con un objeto la pantalla se interrumpe en un haz infrarrojo vertical y otro horizontal, permitiendo de esta forma localizar la posición exacta en que se realizó el contacto.

Galga Extensiométrica

Cuando se utilizan galgas extensiométricas la pantalla tiene una estructura elástica de forma que se pueden utilizar galgas extensiométricas para determinar la posición en que ha sido tocada a partir de las deformaciones producidas en la misma. Esta tecnología también puede medir el eje Z o la presión ejercida sobre la pantalla.

Imagen Óptica

Es un desarrollo relativamente moderno en la tecnología de pantallas táctiles, dos o más sensores son situados alrededor de la pantalla, habitualmente en las esquinas. Emisores de infrarrojos son situados en el

campo de vista de la cámara en los otros lados de la pantalla. Un toque en la pantalla muestra una sombra de forma que cada par de cámaras puede triangularla para localizar el punto de contacto. Esta tecnología está ganando popularidad debido a su escalabilidad, versatilidad y asequibilidad, especialmente para pantallas de gran tamaño.

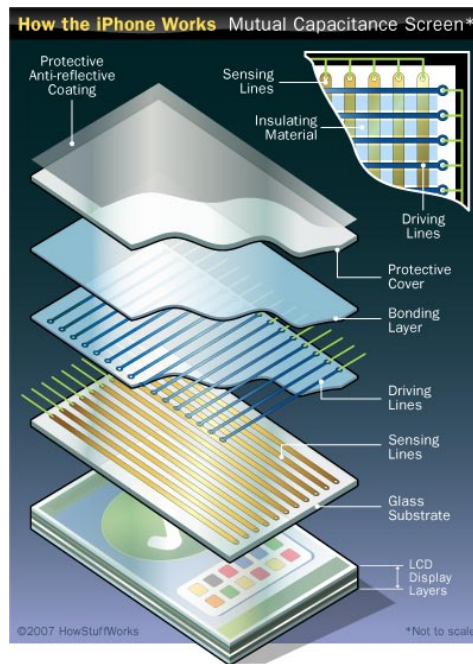
Tecnología de Señal Dispersiva

Introducida en el año 2002, este sistema utiliza sensores para detectar la energía mecánica producida en el cristal debido a un toque. Unos algoritmos complejos se encargan de interpretar esta información para obtener el punto exacto del contacto.

b. DISPOSITIVOS MULTITÁCTILES

Dispositivos Multitáctiles es el nombre con el que se conoce a una técnica de interacción persona-computador y al hardware que la implementa. La tecnología multitáctil consiste en una pantalla táctil o touchpad que reconoce simultáneamente múltiples puntos de contacto, así como el software asociado a esta que permite interpretar dichas interacciones simultáneas.

Las pantallas multitáctiles pueden ser de origen capacitivo (como con las que se equipan muchos teléfonos móviles modernos y que se accionan solamente con los dedos). También pueden estar formadas por una matriz de sensores ópticos. La característica más importante es que permiten obtener la posición de varios puntos de contacto sobre ella de manera simultánea. A menudo también permite calcular la presión o el ángulo de cada uno de los puntos de contacto de forma independiente, lo que permite hacer gestos e interactuar con varios dedos o manos de manera simultánea (de manera similar a los acordes de mouse) y proveer así de una interacción más rica a través de gestos mucho más intuitivos.



Fuente: Grupo de investigación.

Fig. 2.37 Configuración Dispositivo Multitáctil

Dependiendo del tamaño del dispositivo, algunos de éstos son lo suficientemente grandes para que varios usuarios puedan utilizarlos de manera simultánea. Otro aspecto importante de esta técnica es que permite realizar zooms en la aplicación que se esté manejando con sólo usar dos dedos de una manera más directa que con otros dispositivos como el ratón o el estilo.

El dispositivo de interacción permite a los usuarios manipular de forma directa los objetos virtuales representados sobre la superficie del dispositivo. Como resultado de esta experiencia, la interacción entre las personas y el ordenador se lleva a cabo de manera natural e intuitiva sin la necesidad de «controles» periféricos.

Este dispositivo tiene un gran potencial en relación a aplicaciones colaborativas tales como editores colaborativos (gráfico, audio, video, etc.), educación, etc. El funcionamiento del dispositivo se basa en dos sistemas independientes, pero sincronizados entre sí. Por un lado se encuentra el sistema de visualización, que muestra al usuario los objetos virtuales a

manipular. Por otro lado se encuentra el sistema de control, que permite al usuario interactuar con dichos objetos.

c. TABLETS

Una tablet PC o tableta (ordenador personal en tableta) es una computadora portátil con la que se puede interactuar a través de una pantalla táctil o multitáctil. Para trabajar con la computadora, el usuario puede utilizar una pluma stylus o los dedos, sin necesidad de teclado físico ni mouse.

El Tablet PC es sin duda el tipo de ordenador más versátil y cómodo. Más ligero que un portátil y algo menos que una PDA, el Tablet PC es ideal para trabajo de campo o para, sencillamente, trabajar con un ordenador como si de una pizarra se tratara. De hecho, el Tablet PC es especialmente práctico para poder tomar notas mientras hablas por teléfono o mientras estás en una reunión, ya que con un Tablet PC tomas tus anotaciones como si las escribieras sobre papel. Ya no tendrás que pasar tus notas al ordenador; se irán guardando en tu Tablet PC mientras las escribes sobre su pantalla.

Todo ello gracias a que se interactúa con el Tablet PC mediante el uso de un lápiz óptico y una pantalla táctil. Así, las notas que tomes las escribes sobre la pantalla sensible y el Tablet PC las guarda. Con un software especial que suele venir ya preinstalado en el sistema operativo del Tablet PC, lo que se haya escrito será reconocido por el Tablet PC y convertido a texto normal si se quiere, o bien guardarlo como imagen.



Fuente: Grupo de investigación.

Fig 2.38 Tablet PC

El tablet PC no dispone de un teclado o ratón. Se ingresan los datos de entrada con un bolígrafo, aunque usted puede agregar un teclado y el ratón si desea que a través de wireless o USB.

Dicho esto, los Tablet PC son una gran inversión para las personas que necesitan un dispositivo de entrada con más funcionalidad que un ordenador portátil estándar. Los Tablet PC son una gran idea para muchas personas, porque tienen algunas ventajas excelentes a través de ordenadores portátiles estándar y mini ordenadores portátiles.

d. IPENS

Un bolígrafo digital (IPEN) lee el patrón de una llamada punto-patrón de papel. Ya que el patrón es muy detallado, de la pluma “sabe” que su punta se encuentra en el papel. Esto se puede convertir en información útil. Por ejemplo, si utiliza el lápiz para rellenar las casillas en un formulario, la pluma puede “recordar” dónde ha estado y que las cajas se ha marcado.

La información del formulario se pueden transferir a un sistema informático, ya sea por “acoplamiento” de la pluma con un ordenador, o mediante el uso de Bluetooth a través de un teléfono celular en el sistema informático.



Fuente: Grupo de investigación.

Fig. 2.39 IPEN

De lo contrario, un lápiz digital es una construcción simple. Aparte de la posibilidad de navegar por un patrón de puntos, un lápiz digital es un bolígrafo normal. Por tanto, es ideal para procesos basados en papel digital.

2.2.3 IMPLEMENTACION Y UTILIZACIÓN

Los dispositivos interactivos de nueva generación, tanto táctiles como multitáctiles, están implementados con herramientas tecnológicas avanzadas electrónicamente como son los giroscopios y acelerómetros en su última evolución integral.

La utilización de estos implementos, nos permite crear aplicaciones, tanto por medio de NUI como de GUI, en un amplio campo de acción, como lo es en este caso el diagnóstico predictivo, interpretación de datos y análisis en tiempo real de los parámetros mecánico-electrónicos del automóvil.

a. GIROSCOPIOS

Un giroscopio o giróscopo es un objeto esférico, o en forma de disco, montado en un soporte cardánico, de forma que puedan girar libremente en cualquier dirección. Es utilizado para medir la orientación o para mantenerla, al estar basado su funcionamiento en el principio de conservación del momento angular. Cuenta con numerosas aplicaciones, como la de disminuir el balanceo de navíos, para estabilizar plataformas de tiro, la suspensión de los helicópteros o como brújula, ya que siempre se orienta hacia el norte geográfico, y no el magnético como las brújulas convencionales. Otra aplicación más lúdica de los giroscopios es el trompo o peonza.

Cuando se somete el giróscopo a un momento de fuerza que tiende a cambiar la orientación del eje de rotación su comportamiento es aparentemente paradójico ya que el eje de rotación, en lugar de cambiar de

dirección como lo haría un cuerpo que no girase, cambia de orientación en una dirección perpendicular a la dirección "intuitiva".



Fuente: Grupo de investigación.

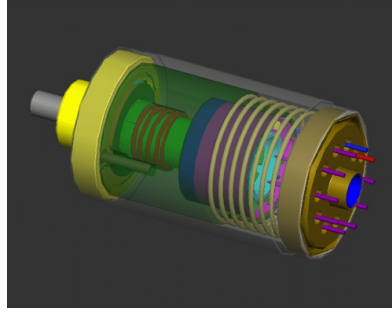
Fig. 2.40 Giroscopio de última tecnología

La inercia o efecto giroscópico es utilizado habitualmente, como el caso del giro que se le proporciona a la pelota de rugby al lanzarla, para que así mantenga una trayectoria más recta que si no girara. También se observa en los patinadores, que giran más deprisa al recoger sus brazos y frenan al desplegarlos. A su vez, los cuerpos en rotación poseen movimiento de precesión, que es el giro que soporta, con una cierta inclinación respecto al eje perpendicular al plano del suelo, ante cualquier fuerza que tienda a cambiar el plano de rotación.

b. ACELERÓMETROS

Un acelerómetro como se intuye por su nombre es un instrumento para medir la aceleración de un objeto al que va unido, lo hace midiendo respecto de una masa inercial interna.

Existen varios tipos de tecnologías y diseños que aunque todos tienen el mismo fin (medir la aceleración) pueden ser muy distintos unos de otros según la aplicación a la cual van destinados y las condiciones en las que han de trabajar.



Fuente: Grupo de investigación.

Fig. 2.41 Acelerómetro Digital

Hay dos parámetros principales a la hora de escoger el medidor adecuado, los rangos de funcionamiento de temperatura y frecuencia. Otros parámetros importantes pueden ser el tamaño y la gravedad, si tienen más funciones, la resistencia a golpes y por supuesto el precio

Los acelerómetros han pasado de estar dedicados a un uso industrial (medir vibraciones y oscilaciones) y de investigación a estar presentes en muchos aparatos cotidianos, veremos algunos ejemplos de ellos (NINTENDO Wii, Footpod, portátiles, IPAD, etc...)

Se puede considerar al acelerómetro piezoeléctrico como el transductor estándar para medición de vibración en máquinas. Se produce en varias configuraciones, pero la ilustración del tipo a compresión sirve para describir el principio de la operación. La masa sísmica está sujeta a la base con un perno axial, que se apoya en un resorte circular. El elemento piezoeléctrico está ajustado entre la base y la masa. Cuando una materia está sujeta a una fuerza, se genera una carga eléctrica entre sus superficies.

Cuando se mueve el acelerómetro en la dirección arriba abajo, la fuerza que se requiere para mover la masa sísmica esta soportada por el elemento activo. Según la segunda ley de Newton, esa fuerza es proporcional a la aceleración de la masa. La fuerza sobre el cristal produce la señal de salida, que por consiguiente es proporcional a la aceleración del transductor.

Los acelerómetros son lineales en el sentido de la amplitud, lo que quiere decir que tienen un rango dinámico muy largo. Los niveles más bajos de aceleración que puede detectar son determinados únicamente por el ruido electrónico del sistema electrónico, y el límite de los niveles más altos es la destrucción del mismo elemento piezoeléctrico. Este rango de niveles de aceleración puede abarcar un rango de amplitudes de alrededor de 10 , lo que es igual a 160 dB. Ningún otro transductor puede igualar esto.

La mayoría de los acelerómetros que hoy en día se usan en la industria son del tipo "PCI", lo que quiere decir que tienen un preamplificador interno de circuito integrado. Este preamplificador recibe su energía de la polarización de la corriente directa por el alambre de la misma señal, así que no se necesita alambrado suplementario. El aparato con que está conectado el aparato debe tener su fuerza de corriente directa disponible para este tipo de transductor. El acelerómetro PCI tendrá un límite de baja frecuencia, debido al mismo amplificador y este se sitúa generalmente a 1 Hz para la mayoría de las unidades disponibles comercialmente. Algunas unidades fueron diseñadas especialmente para ir hasta 0, 1 Hz si se necesitan datos de muy baja frecuencia.

2.2.4. APLICACIONES DEDICADAS DE INTERFAZ ÚNICAS DE USUARIO (IPA).

Una aplicación es un tipo de programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de trabajo. Esto lo diferencia principalmente de otros tipos de programas como los sistemas operativos (que hacen funcionar al ordenador), las utilidades (que realizan tareas de mantenimiento o de uso general), y los lenguajes de programación (con el cual se crean los programas informáticos).

Suele resultar una solución informática para la automatización de ciertas tareas complicadas como pueden ser la contabilidad, la redacción de documentos, o la gestión de un almacén. Algunos ejemplos de programas

de aplicación son los procesadores de textos, hojas de cálculo, y base de datos.

Ciertas aplicaciones desarrolladas «a medida» suelen ofrecer una gran potencia ya que están exclusivamente diseñadas para resolver un problema específico. Otros, llamados paquetes integrados de software, ofrecen menos potencia pero a cambio incluyen varias aplicaciones, como un programa procesador de textos, de hoja de cálculo y de base de datos.

Posee ciertas características que le diferencia de un sistema operativo (que hace funcionar al ordenador), de una utilidad (que realiza tareas de mantenimiento o de uso general) y de un lenguaje (con el cual se crean los programas informáticos). Suele resultar una solución informática para la automatización de ciertas tareas complicadas como puede ser la contabilidad o la gestión de un almacén. Ciertas aplicaciones desarrolladas 'a medida' suelen ofrecer una gran potencia ya que están exclusivamente diseñadas para resolver un problema específico.

En general, una aplicación es un programa compilado, escrito en cualquier lenguaje de programación. Las aplicaciones pueden tener distintas licencias de distribución como ser freeware, shareware, trialware, etc. Las aplicaciones tienen algún tipo de interfaz, que puede ser una interfaz de texto o una interfaz gráfica (o ambas). También hay que destacar que la distinción entre aplicaciones y sistemas operativos muchas veces no es clara. De hecho, en algunos sistemas integrados no existe una clara distinción para el usuario entre el sistema y sus aplicaciones.

La diferencia entre los programas de aplicación y los de sistema estriba en que los de sistema suponen ayuda al usuario para relacionarse con el computador y hacer un uso más cómodo del mismo, mientras los de aplicación son programas que cooperan con el usuario para la realización de las actividades mencionadas.

Dentro de los programas de aplicación, puede ser útil una distinción entre

aplicaciones verticales, de finalidad específica para un tipo muy delimitado de usuarios (médicos, campo automotriz), y aplicaciones horizontales, de utilidad para una amplísima gama de usuarios de cualquier tipo.

Los adelantos tecnológicos logrados por la humanidad en materia de telemática y telecomunicaciones revolucionaron totalmente la concepción del comercio tangible a través del soporte de papel asegurado por sellos y firmas, surgiendo una nueva concepción de la expresión gráfica y escrita a través de la utilización de medios no tangibles. Nos encontramos ante un proceso novísimo de desmaterialización, en el sentido de sustituir el soporte tradicional del "papel", más no de prescindir de lo material que algunos suelen confundir, pues sabemos que el mundo electrónico o digital no es uno totalmente intangible.

En la ciencia el uso de la informática como herramienta de ayuda a la medicina es una realidad en auge. Desde tiempos antiguos muchas de las actividades humanas se han basado en la repetición, a veces muy gravosa, de actos o cálculos y del mismo modo que se inventaron operaciones matemáticas básicas para simplificarlas, surgió la necesidad de mejorar las limitadas prestaciones que ofrece la mente del hombre para calcular, a medida que las diversas ciencias se hicieron más complejas

2.3 DIAGNÓSTICO PREDICTIVO Y PREVENTIVO. FUNDAMENTOS MECÁNICOS, ELÉCTRICO Y ELECTRÓNICOS

2.3.1 INTRODUCCIÓN

Uno de los conceptos fundamentales del diagnóstico es que no todos los componentes (mecánicos, eléctricos, electrónicos, etc.) se comportan de acuerdo con el patrón de fallo de vida útil. No todos los componentes siguen esta ley, o sea que no siempre la probabilidad de fallo aumenta cuando el elemento envejece. Esto sólo se cumple sí y cuándo todas las unidades iguales a ese componente presentan una baja y uniforme

probabilidad de fallo durante un cierto lapso de tiempo conocido, a partir del cual la probabilidad de fallo crece rápidamente.

En tales casos la posibilidad de restaurar o de reemplazar el componente justo antes de cumplirse esa vida útil conocida, es técnicamente factible. Si además merece la pena ser realizado, estaríamos ante un caso típico y tradicional de mantenimiento preventivo. Su característica por definición es que restauraremos ó reemplazaremos ese componente en plazos programados, independientemente de la condición en la cuál ese componente se encuentre en el momento de realizar la acción.

Habremos prevenido que el fallo ocurra, mediante una acción cíclica programada a intervalos fijos independientes de la condición del componente.

a. FUNDAMENTOS

Casi todos los fallos funcionales dan “algún aviso” de que están ocurriendo o por ocurrir: “el rodamiento hace ruido audible antes de fallar”, “el neumático del automóvil muestra visible desgaste cuando comienza a perder su funcionalidad”, “la temperatura del radiador aumenta cuando pierde agua”. Esto nos permite evitar que la falla ocurra si encontramos la forma de “chequear” si esas “condiciones” están presentes, efectuando la reparación antes de que el fallo ocurra si la condición chequeada se presenta. No hacer nada, si la condición no se presenta.

Es decir: existe un fallo potencial que advierte que un fallo funcional ha comenzado a ocurrir. si conozco ese fallo potencial, si es técnicamente factible chequearlo y si merece la pena ser efectuado, encontré una tarea predictiva ó “a condición” que me permite realizar el mantenimiento nuevamente antes de que la falla funcional y sus consecuencias ocurran.

Como su nombre lo indica este tipo de mantenimiento se diseño con la idea de prever y anticiparse a los fallos de las maquinas y equipos, utilizando

para ello una serie de datos sobre los distintos sistemas y sub-sistemas e inclusive partes.

Bajo esa premisa se diseña el programa con frecuencias calendario o uso del equipo, para realizar cambios de sub-ensambles, cambio de partes, reparaciones, ajustes, cambios de aceite y lubricantes, etc., a maquinaria, equipos e instalaciones y que se considera importante realizar para evitar fallos.

Es importante trazar la estructura del diseño incluyendo en ello las componentes de Conservación, Confiabilidad, Mantenibilidad, y un plan que fortalezca la capacidad de gestión de cada uno de los diversos estratos organizativos y empleados sin importar su localización geográfica, ubicando las responsabilidades para asegurar el cumplimiento.

b. MECANISMOS DE ANALISIS PREDICTIVO

Cuando se monitoriza una variable física relacionada con el estado de la máquina, en este caso el automóvil, se buscan analizar los siguientes cuatro objetivos: vigilancia, protección, diagnóstico y pronóstico

Vigilancia. Cuando se mide una variable física con este objetivo se busca que la técnica predictiva empleada indique la existencia de un problema. Debe distinguir entre condición buena o mala para funcionar, e incluso, si es mala, indicar cuán mala es.

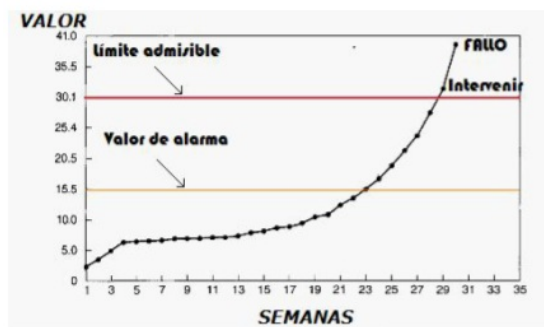
Protección. Su objetivo es evitar fallas catastróficas. Una máquina está protegida, si cuando los valores que indican su condición llegan a valores considerados peligrosos, la máquina se detiene automáticamente.

Diagnóstico de fallos. Su objetivo es identificar cuál es el problema específico que presenta el equipo, no sólo si existe un problema o no.

Pronóstico. El objetivo es estimar cuánto tiempo más podría funcionar la máquina sin riesgo de un fallo catastrófico.

La información más importante que arroja este tipo de seguimiento de los equipos es la tendencia de los valores, ya que es la que permitirá calcular o prever, con cierto margen de error, cuando un equipo fallará; por ese el motivo se denominan técnicas predictivas.

En la figura 2.42 se indica la gráfica de un valor de vibración correspondiente a un cojinete, y que presenta un tendencia alcista. Cuando se alcanza un determinado valor es conveniente reemplazar el cojinete. Si no se realiza, el cojinete terminará fallando.



Fuente: Grupo de investigación.

Fig. 2.42 Grafica de tendencia de un valor de amplitud de vibración de un cojinete.

La mayor ventaja, indudablemente, es que en la mayoría de las ocasiones no es necesario realizar grandes desmontajes, y en muchos casos ni siquiera pararla. Si tras la inspección se aprecia algo irregular se propone o se programa una intervención. Además de prever el fallo catastrófico de una pieza, y por tanto, pudiendo anticiparse a éste, las técnicas de mantenimiento predictivo ofrecen una ventaja adicional: la compra de repuestos se realiza cuando se necesita, eliminando pues stocks (capital inmovilizado).

La razón fundamental por la que el mantenimiento predictivo ha tenido un notable desarrollo en los últimos tiempos hay que buscarla en un error cometido tradicionalmente por los ingenieros de mantenimiento para estimar la realización de tareas de mantenimiento de carácter preventivo: las <<curvas de bañera>>, detalladas en la figura 2.43, que representan la probabilidad de fallo frente al tiempo de uso de la máquina, y que se suponían ciertas y lógicas, han resultado no corresponder con la mayoría de los elementos que componen un equipo.

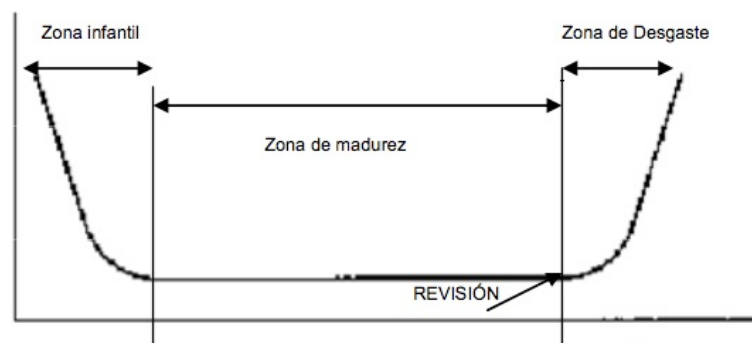


Fig. 2 Curva de bañera. Probabilidad de fallo vs. Tiempo

Fuente: Grupo de investigación.

Fig. 2.43 Curva de bañera. Probabilidad de fallo vs. Tiempo

En estas curvas se reconocían tres zonas:

- Zona inicial, de baja fiabilidad, por averías infantiles
- Zona de fiabilidad estable, o zona de madurez del equipo
- Zona final, nuevamente de baja fiabilidad, o zona de envejecimiento.

Como se daba por cierta esta curva para cualquier equipo, se suponía que transcurrido un tiempo (la vida útil del equipo), éste alcanzaría su etapa de envejecimiento, en el que la fiabilidad disminuiría mucho, y por tanto, la probabilidad de fallo aumentaría en igual proporción. De esta manera, para alargar la vida útil del equipo y mantener controlada su probabilidad de fallo era conveniente realizar una serie de tareas en la zona de envejecimiento, algo parecido a un 'lifting', para que la fiabilidad aumentara.

c. PROCEDIMIENTO Y ANÁLISIS

Para un correcto análisis de los síntomas que presenta el automóvil, antes de dar un diagnóstico predictivo exacto y de calidad, se debe tomar en cuenta los siguientes factores:

Importancia y criticidad de los equipos dentro del proceso. Este es un factor determinante para realizar un tipo de análisis u otro (o no hacerlo), e incluso para determinar el número de puntos a medir, y la cantidad de veces a hacerlo.

Tipos de averías que se intentan evitar. A la hora de analizar un automotor, ayuda a simplificar el hecho de que haya un tipo de avería que deba evitarse a toda costa, y otros tipos de fallo que no sean tan importantes; eso nos permitirá enfocar la técnica en una dirección que verdaderamente pueda ofrecer resultados.

Riesgos a eliminar vs. riesgos a evitar. Ligado al punto anterior, se puede crear un listado con los riesgos asumibles por parte del usuario del automotor, y los que se desean eliminar por completo.

Y por supuesto siempre debe tener la información del fabricante sobre el equipo:

- Información técnica.
- Información histórica, si tuviera un seguimiento periódico.
- Y durante las mediciones, información sobre modificaciones realizadas en el equipo, averías, preventivos o cualquier tipo de anomalía que haya sido detectada en la máquina
- Tener la mejor herramienta disponible en el mercado, así como herramienta especializada para cada marca.

Tomando en cuenta estos factores, y habiendo realizado un análisis profundo en cada caso que se pueda presentar, el procedimiento a seguir es el siguiente:

Determine las metas y objetivos.

El primer paso para desarrollar un programa de mantenimiento preventivo es determinar exactamente qué es lo que se quiere obtener del programa. Usualmente el mejor inicio es trabajar sobre una base limitada y expandirse después de obtener algunos resultados positivos.

Establecer los requerimientos para el mantenimiento preventivo.

Decidir que tan extenso pueda ser su programa de mantenimiento preventivo. Qué debe de incluir y dónde debe de iniciar.

Maquinaria y Equipo a incluir.

La mejor forma de iniciar esta actividad es determinar cual es la maquinaria y equipo más crítico en el vehículo, de acuerdo al caso; algunas veces esto es muy fácil y otras veces no, esto depende de lo que manufacture la compañía, así como la tecnología que incluye el vehículo.

Áreas de operación a incluir.

Puede ser mejor, seleccionar un sección del sistema a ser estudiado para facilitar el inicio; ésta aproximación permite que concentre sus esfuerzos y más fácilmente realice mediciones del progreso. Es mucho mejor el expandir el programa una vez que probó que se obtienen resultados.

Decida si se van a incluir disciplinas adicionales al programa de mantenimiento preventivo.

Se debe determinar si implementará rutas de lubricación, realizar inspecciones y hacer ajustes y/o calibraciones, o cambiar partes en base a frecuencia o uso. (Mantenimiento preventivo tradicional.)

Reunir y organizar los datos.

El programa de mantenimiento preventivo deberá incluir procedimientos detallados que deben ser completados en cada inspección o ciclo. Existen

varias formas para realizar estos procedimientos en las órdenes de trabajo de mantenimiento preventivo.

Los procedimientos permiten insertar detalles de liberación de componentes o equipo, trabajo por hacer, diagramas a utilizar, planos de cada sistema, ruta de lubricación, ajustes, calibración, arranque y prueba, reporte de condiciones, carta de condiciones, manual del fabricante, recomendaciones del fabricante, observaciones, etc.

Relacionar los procedimientos a la orden de trabajo y los reportes maestros individuales de mantenimiento preventivo. De ser posible utilizar o diseñar procedimientos para la ordene de trabajo correctivo, o rutinario.

2.3.2 ELÉCTRICA Y ELECTRÓNICA

El campo de acción de la electricidad y la electrónica en el plano del mantenimiento predictivo y preventivo juega un papel muy importante, aplicada exclusivamente al control de los sistemas del automóvil. La complejidad y diversidad de los modelos de motores, sistemas eléctricos y electrónicos en los automóviles hizo necesaria la preparación de métodos de análisis del comportamiento mecánico y electrónico de cada sistema.

Por ejemplo en estos vehículos con mayor tecnología, se reemplaza el carburador por un sistema electrónico de inyección de gasolina: E.F.I. (electronic fuel injection) . Su introducción se debió a un aumento en las exigencias de los organismos de control del medio ambiente para disminuir las emisiones de los motores y a que, por la utilización de sensores y actuadores, se establezca una compuerta de lectura de datos para su correcto mantenimiento, ya sea por medios analógicos, como digitales.

a. FUNDAMENTOS

En el análisis de un diagnóstico predictivo que incluye un estudio del sistema eléctrico y electrónico, primero se debe establecer el

comportamiento de los principales componentes activos del sistema. Dado esta afirmación, se debe estudiar tanto el funcionamiento de sensores y actuadores, como estos con manejados por una unidad inteligente computarizada denominada Unidad Central Computarizada (ECU).

b. PROCEDIMIENTO Y ANÁLISIS

En el diagnóstico predictivo en el campo de la electricidad y la electrónica automotriz, en un principio se debe comprender la utilización de herramientas básicas como los son el multímetro, el osciloscopio y el escáner automotriz, con la finalidad de eficazmente aplicar cada uno a determinados casos de estudio en los cuales van a tener sus ventajas y desventajas unos de otros.

MULTÍMETRO AUTOMOTRÍZ

Los multímetros son una herramienta de prueba y de diagnóstico invaluable para los técnicos electricistas, técnicos en mantenimiento, aire acondicionado y refrigeración así como otros profesionales que desean usar este instrumento en sus respectivas áreas (como es el caso de la electricidad automotriz) y expertos en múltiples disciplinas.

En un automóvil se efectúan muchos procesos de trabajo mediante máquinas eléctricas, estos pueden ser generadores o alternadores. Es por ello que será necesario conocer a fondo tanto en la estructura como de su funcionamiento para hacer reparaciones.

El multímetro en el automóvil nos permite hacer las siguientes mediciones, y comprobaciones:

- Medición de Voltajes (ac – dc)
- Medición de Corrientes (Amp)
- Medición de Resistencias (Ohms)
- Medición de Frecuencias (Hz)
- Medición de RPM

- Comprobación de Continuidad
- Comprobación de Diodos
- Comprobación de caídas de tensión
- Comprobación de un cortocircuito
- Comprobación de intensidad

Siendo los campos de inspección preliminares:

- Batería.
- Sistema de Carga
- Sistema de Arranque
- Terminales y conductores eléctricos

OSCILOSCOPIO AUTOMOTRIZ

Los circuitos electrónicos se caracterizan por la presencia de señales en diversos puntos de los mismos, es decir, tensiones o corrientes que evolucionan en el tiempo. En la mayoría de los casos la velocidad de esta evolución torna imposible su seguimiento con los instrumentos de deflexión o digitales de uso corriente (multímetros o testers).

Dada la importancia de la información que la evolución temporal de estas tensiones y corrientes brinda acerca del funcionamiento del circuito bajo ensayo, se desarrolló un instrumento especial para facilitar su observación y efectuar mediciones de tensión y tiempo.

El Osciloscopio es el instrumento capaz de registrar los cambios de tensión producidos en circuitos eléctricos/electrónicos y mostrarlos en forma gráfica en la pantalla de un tubo de rayos catódicos. Este instrumento genera en su interior un haz de electrones que se aceleran e impactan sobre la pantalla del mismo produciendo un punto luminoso que puede ser desplazado en forma vertical y horizontal proporcionalmente a la diferencia de potencial aplicada sobre unos electrodos. Si la tensión que produce la desviación vertical es la que se desea observar y provocamos mediante un

generador interno un desplazamiento horizontal del punto a velocidad constante, obtendremos sobre la pantalla una representación de la evolución temporal de la señal observada.

Cuando las señales a observar son periódicas es posible representarlas en forma estática en una pantalla mediante el recurso de sobreimprimir los ciclos sucesivos, obteniéndose una imagen de la evolución temporal de la magnitud a lo largo de uno o más ciclos, o sea de la forma de onda. El circuito de sincronización (conocido como circuito de disparo o gatillado) es el encargado de hacer coincidir entre sí los sucesivos ciclos de la onda sobre la pantalla para obtener una imagen estable.

Esta herramienta de recepción y medición de señales nos permite obtener la siguiente información:

- Corrientes básica AC-DC pulsantes.
- Comprobación de Sensores.
- Sensores de giro, reluctancia variable, Hall y opto electrónicos.
- Sensores de temperatura, sensor de detonación, sonda lambda.
- Sensores de rueda, sensores ABS.
- Comprobación de actuadores.
- Bobinas, inyectores, corrientes de excitación.
- Análisis de módulos de encendido.

EL ESCANER AUTOMOTRIZ

El scanner automotriz es una herramienta de lectura de la información que proporciona la ECU o Unidad de control Electrónico, es decir, este dispositivo nos permite recolectar, leer, y en ciertas ocasiones guardar información del control total de las operaciones electrónicas que se produzcan en nuestro automóvil.

A continuación se enlista varias de las habilidades que nos provee un escáner automotriz, tanto especial de cada marca, como universal.

- **Leer y Borrar Códigos de Falla PIDs.** El escáner automáticamente borrará el código de falla de la computadora de abordo sin necesidad de desconectar el fusible PCM (modulo de control del tren motriz) o desconectar la batería. Esta capacidad es limitada a los modelos que permitan realizar esta acción.
- **Autoconfiguración.** Permite al usuario desplegar en pantallas los datos seleccionados que pertenecen al área de diagnostico en la cual está interesado.
- **Línea de datos.** Permite al usuario monitorear datos entrantes y salientes de sensores, solenoides, y transmisión de la computadora de abordo. Los datos son actualizados en cuanto el motor se enciende para poder ver en acción los circuitos monitoreados por el sistema.
- **Conversión sistema métrico / ingles.** Permite al usuario monitorear la línea de datos en sistema de medición estándar o métrico. Los manuales de reparaciones pueden proveer valores para hallar el diagnostico en el sistema métrico o en sistema inglés, por lo que la conversión es de gran utilidad.
- **Compatibilidad con PC, Mac y Tablets.** Los datos de las computadoras de abordo pueden ser transferidos de un escáner a una laptop o a una PC.
- **Captura digital.** El escáner tiene la capacidad de grabar y reproducir los datos; esta habilidad es útil para diagnosticar fallas de tipo intermitentes.
- **Códigos de falla.** Los códigos de falla se presentan cuando una falla en un circuito ha sido detectada por la computadora del vehículo. Los códigos duros se refieren a fallas que están presentes al momento del diagnostico. Los códigos suaves son aquellos que aparecieron en el pasado, pero no están presentes al momento del diagnostico.

- **Ajuste de combustible.** El integrador surte combustible a un inyector predeterminado y ajusta la distribución a base cronometraje si el PROM falla (Memoria Programable de solo Lectura).
- **Prueba de Actuadores.** Una prueba bi-direccional utilizada para operar transmisiones y solenoides controlados por computadora. Es también útil para revisar voltajes y señales salientes.
- **ABS.** Esta modalidad es utilizada para obtener códigos de fallas en el sistema de frenos anti-bloqueo.
- **Datos Freeze Frame.** Datos capturados por la computadora de abordo en el momento en que la falla se presenta. Los datos presentan las condiciones de operación del vehículo cuando el problema ocurre. Estos datos son utilizados para re-crear el problema antes o después de la reparación. El escáner puede desplegar de una a diez freeze frames.
- **Pruebas de sensores de oxígeno.** Monitorea los sensores de oxígeno
- Otras pruebas complementarias de acuerdo a cada marca y modelos.

c. RELACIÓN MECÁNICA – ELECTRÓNICA

Este complemento implícito entre ambos campos se basa, fundamentalmente, en las actividades de la industria automovilística que, desde hace algunos años, ha introducido una serie de transformaciones en el que respecta a la instalación eléctrica de control de los vehículos.

Se han creado nuevos dispositivos de servicio, reemplazando, cada vez más, los controles mecánicos por otros completamente eléctricos o electrónicos, y utilizándose tecnologías innovadoras, como son la microelectrónica, la telemática, la informática.

Los objetivos principales son aumentar la seguridad, el confort, y reducir la contaminación ambiental. Para darse cuenta de la importancia de tal innovación, basta mencionar algunas de las aplicaciones de la electrónica en un vehículo de uso civil:

- El sistema de mando del motor está compuesto de tres grupos que incluyen los sensores (y las señales emitidas por el sensor), la ECU (electronic control unit) del motor y los actuadores.. Las funciones de la ECU del motor se dividen en control EFI, control ESA, control ISC, función de diagnóstico, funciones de respaldo y a prueba de fallos y otras funciones.

Además del control del motor, también la E.C.U. o a veces otra computadora a bordo inclusive integrada al sistema, mediante el empleo del sistema de bus de datos CAN (Controlador de área de red) en el vehículo se interconectan componentes electrónicos como unidades de control o sensores inteligentes.

Gracias a la utilización del sistema de bus de datos CAN, se obtienen las siguientes ventajas en el sistema general del vehículo. Al adoptar el sistema MPX se logran las siguientes ventajas:

- Reduce el número de mazos de cables.
- El compartir las información permite la reducción en el número de interruptores, sensores y actuadores.
- Dado que la ECU próxima a los interruptores y sensores lee la información de las señales y la transmite a las demás ECU, se puede reducir la longitud del cableado.

Otros sistemas también se controlan electrónicamente, tales como:

- La estimación del octanaje del combustible
- El sistema de control de los cambios ECT (transmisión controlada electrónicamente).

- El sistema de control de corte EGR .
- El sistema de inducción variable de aire.
- El sistema SCV (Válvula de control de torbellino)

2.3.3. NUEVAS TECNOLOGÍAS APLICABLES

La continua renovación de la tecnología instalada en los vehículos, crea la necesidad de contar con equipos y herramientas de diagnóstico predictivo, que sirvan de soporte a los profesionales especializados dentro de un proceso de verificación, diagnóstico y obtención de parámetros de óptimo funcionamiento de los sistemas de control de gestión integral en los motores de combustión interna, bajo normas de control OBDII.

La introducción de sistemas táctiles de la marca Apple, ha contribuido con la mejora de los sistemas inalámbricos, con aplicaciones desarrolladas para actividades específicas de industrias especiales. En este caso, en la industria automotriz, permite mantener un sistema de información organizada y universal, así como nos permite mantener una base de datos determinada para cada unidad y marca.

Nuevos procesos de comunicación desarrolladas especialmente para dispositivos táctiles de última generación, mediante interconexión inalámbrica de tipo WIFI, en la actualidad nos permite transformar estos gadgets en aparatos electrónicos de avanzada, dentro del campo automotriz, simplificando la mayoría de procesos dentro de una interfaz intuitiva y amigable al usuario.

Es así que, este proyecto tecnológico, nos abre una compuerta nueva a la obtención de datos del automotor en tiempo real, de manera más fácil. Datos como el performance (velocidad del motor, compresión, vacío, aceleración, torque), medidores de motor (avance al encendido, temperatura refrigerante, MAP, MAF, carga que maneja, presión de combustible, relación aire – combustible), lectura y borrado de PIDs, Fuerzas G's de aceleración y frenado, diagnóstico OBDII, etc, ya son de

análisis de programadores en lenguaje Objective-C, para la personalización de gauges y GLCDS, de acuerdo a las necesidades que se requieran cubrir, como en el caso de esta Tesis, en el diagnóstico Predictivo y Preventivo.

2.3.4. APLICACIÓN EN MECÁNICA DE COMPETICIÓN

El desarrollo de este proyecto tecnológico puede ser introducido en el área de la competición, tanto en los campos de telemetría vehicular, calibración de valores de manera instantánea, medidores de fuerzas G tanto de frenado como aceleración, niveles de soporte del automóvil sobre la superficie y medidores de tipo GLCDs personalizables en tiempo real.

TELEMETRÍA Y CALIBRACIÓN INALÁMBRICA

Telemetría es una forma de tecnología muy propagada en diversas áreas de la ciencia, como la medicina o la biología. Consiste en la medición de datos a analizar y consecuentemente enviar esos datos a una estación base para su posterior análisis.

Tales sistemas difieren entre sí tanto por la tecnología empleada como por la arquitectura adecuada a cada necesidad. Normalmente se implementa en sitios de difícil acceso posibilitando el monitoreo constante desde otro punto de acuerdo a las distancias y las bandas de transmisión.

Considerando que el lugar de instalación es un automóvil y que circula en un ambiente urbano, fue escogida la interfaz de hardware y la técnica de comunicación adecuada para facilitar el acceso a un acumulador de datos remoto: Acceso a interfaz de comunicación vía la red WIFI nativa del dispositivo táctil de la marca Apple, con un alcance nominal de 30 metros, extensible a 200 metros con calibradores de señal especiales.

La posibilidad de monitoreo remoto de automóviles puede ser aplicado a diversos fines competitivos:

Monitoreo en tiempo real de vehículos de competencia, disminuyendo el número de veces que un vehículo debe ser retirado de circulación para mantenimiento.

Para el análisis estadístico de los equipos de desempeño del automóviles, que utilicen el sistema en sus vehículos, obteniendo así un registro continuo y actualizado del trabajo y correspondiente desgaste de las piezas y sistemas del automotor.

Verificar el nivel de emisión de gases contaminantes, posibilitando así evitar una investigación de legislaciones que estén vigentes dentro de la competición, mejorando la calidad del aire y disminuyendo el consumo de combustibles fósiles. Pues, a pesar de existir una legislación que establece límites para la emisión de gases (Resolución 0910 de 2008), en la práctica, es nula la disminución pues no existe una forma fácil para la medición de este parámetro a gran escala y en el momento y lugar exactos que se produce la combustión.

MEDICIÓN DE FUERZAS G

Las fuerzas G son una comparación entre la fuerza provocada por una aceleración en un objeto y su peso. Es lo que en ingeniería se llama “número adimensional”, o lo que es lo mismo, que no tiene unidad de medida. Normalmente se compara con el valor de la gravedad -de ahí proviene el nombre de fuerza G- para dar una idea intuitiva del valor del número.

Para su análisis debemos comprender cómo afectan estas “fuerzas” G tanto en el comportamiento del Automóvil, cómo en su efectos hacia el conductor.

Tomamos como ejemplo un monoplaza, ya que en estos prototipos de carrera es donde más claro de ve el desarrollo de las fuerzas G en los automóviles, determinando su efecto en los siguientes casos:

- El monoplace acelera:

Las aceleraciones de un monoplace de la categoría reina no suelen sobrepasar el valor de 1 G. A pesar de esto, no debemos pensar que se trata de una aceleración pequeña, puesto que esa es la desaceleración que recibe un coche de calle al realizar una frenada de emergencia.

- El monoplace gira:

Estas aceleraciones son difíciles de soportar puesto que, o cambian rápidamente de dirección o son realmente intensas, con valores cercanos a las 4 G. Los pilotos deben ejercer una fuerza igual a la lateral para poder mantener la visión de la curva, cosa que implica un entrenamiento intenso del cuello en el gimnasio, con máquinas específicamente desarrolladas para lograr dicho objetivo. Aún así, hay circuitos que entrañan un gran reto para los cuellos de los pilotos por varias razones:

- El sentido de giro del circuito es opuesto al habitual (en el caso de la F1, sentido anti horario), cosa que implica un mayor tiempo de giro en una dirección no habitual para el cuello.
- Hay curvas rápidas con giros no peraltados, como sería el caso de curvas como la 130R (Suzuka), Estoril (Magny Cours), Blanchimont (Spa), la curva nº 11 (Estambul Park) o Subida do Boxes (Interlagos) para dar ciertos ejemplos.

- El monoplace experimenta cambios de nivel en el terreno:

Aunque son casos atípicos, resultan interesantes, puesto que las curvas con cambios de nivel suelen estar entre las más conocidas del campeonato, en este caso con mayor frecuencia en los campeonatos de Rally y en circuitos urbanos. En este tipo de curvas, a las aceleraciones laterales y frontales se les suman aceleraciones verticales que presionan al coche hacia el terreno, o lo que es más peligroso, disminuyen la presión del vehículo al suelo.

- El monoplaza frena:

La frenada es sin duda alguna la situación en la cual mas aceleraciones recibe el piloto. Los materiales utilizados para los dispositivos de frenado son radicalmente diferentes a los materiales utilizados en los coches de calle, éstos se asemejan más a los utilizados en aviones comerciales o aviones de combate, ya que se logran frenadas de hasta 5,5 G.

MEDIDORES DE TIEMPO REAL

La personalización de medidores de tiempo real GLCDs son de gran utilidad, tanto para el equipo de mantenimiento, como para el piloto y copiloto del automotor, ya que les ayuda a reducir espacio, instalaciones, peso total del automóvil, aumentar visibilidad del piloto y copiloto, y mantener una comunicación directa con el equipo sobre el funcionamiento correcto de cada dispositivo.

Esta utilidad aumenta en gran parte el desempeño de la unidad dentro de la competición, siendo un plus a su utilización continua dentro del plan total de la carrera.

2.4 SISTEMA DE DIAGNÓSTICO A BORDO OBD II

2.4.1 INTRODUCCIÓN

Durante los años 70 y principios de los 80 algunos fabricantes empezaron a usar componentes electrónicos de control y diagnóstico de errores en sus automóviles. Al principio fue solo para conocer y controlar las emisiones del vehículo y adaptarlas a los estándares exigidos, pero con el paso del tiempo estos sistemas fueron volviéndose cada vez mas sofisticados, hasta los años 90, donde surgió el estándar OBD II.

El OBD II, como su nombre indica "On Board Diagnostic Second Generation", es un sistema que permite diagnosticar los errores que se

producen en el vehículo sin necesidad de desmontar partes para descubrir la procedencia de dicho error.

Este sistema de codificación única se encuentra actualmente implantado en todos los turismos y vehículos industriales ligeros que se producen y a diferencia de otros sistemas desarrollados antes de 1996, este se caracteriza por ser un sistema estandarizado, que permite, de manera fácil, ver que errores se han producido en un vehículo cualquiera utilizando una única codificación y claro esta, un conector estandarizado.

2.4.2 FUNCIONES DEL OBD II

Los vehículos disponen de una o varias ECU "Electronic Control Unit", en general son muchas, que se encargan de gestionar ciertos parámetros de nuestro vehículo para asegurar su correcto funcionamiento. Las relaciones entre estos parámetros deben mantenerse acotadas, dependiendo de las condiciones externas varían ciertos rangos, o sino, es que se está produciendo algún mal funcionamiento en nuestro vehículo.

Los parámetros principales que dictan como debe estar funcionando nuestro motor, y si verifican si todo funcionando correctamente son:

- Velocidad
- Carga
- Temperatura del motor
- Consumo de combustible
- Temperatura ambiente
- Caudal de aire
- Emisiones

Para conocerlos, los automóviles actuales, incorporan una gran cantidad de sensores, que permiten conocer a las ECU, cuales son las condiciones externas, y decidir como actuar sobre el motor. En caso de que alguno de

los parámetros se salga de los rangos marcados, el sistema OBD II, es el encargado de almacenar esta información, y avisar al conductor de que algo sufre un mal funcionamiento, señalizando con un indicador luminoso que es recomendable ir al taller a revisar que error se a producido.

Una vez el vehículo llega al taller, el equipo de mecánicos, puede acceder a la información almacenada por el OBD II, ver que error era el que se había producido, y arreglarlo en caso de necesidad sin tener que hacer múltiples pruebas para descubrir la procedencia del error.

Por supuesto, el sistema no es perfecto, y habitualmente, cuando se produce un error en nuestro vehículo, se desencadena una serie de errores inducidos, que también son almacenados. Aquí es donde entra la experiencia y el saber del mecánico para descubrir donde esta la raíz del problema.

2.4.3 DETECCIÓN DE ERRORES CON OBD II

El sistema OBDII es el encargado de almacenar los códigos de los errores que se generan en nuestro vehículo, pero no solo almacena el error, sino que lo almacena de tal manera, que su lectura, nos indique donde se ha producido el error y nos facilite el trabajo de reparación.

Las decenas de unidades de control que hay repartidas en nuestro vehículo, son las encargadas de interpretar todas las informaciones que generan los sensores, velocidades, caudales, presiones, etc. y gracias a estas informaciones, actúan sobre los componentes del vehículo, que regularan el funcionamiento de este.

La cantidad de información con la que se trabaja, y la situación de dichas ECU's en el vehículo, hace imposible, escanear una por una las diferentes ECU's para buscar un mal funcionamiento. Para ello se usa el OBDII, cuando alguna unidad de control, recibe una señal anómala, informa al

resto del sistema, y el OBDII es el encargado de avisar al conductor de que se ha producido un error y almacenarlo.

Por tanto, se ha de dejar claro que no es el sistema OBDII el encargado de detectar los errores, sino, simplemente es el encargado de informar y almacenar los errores detectados por el resto de unidades de control y facilitarnos el acceso a estos.

2.4.4 ACCESO A LA INFORMACIÓN DEL SISTEMA OBD II

Cuando el sistema almacena alguna información de error, nos indica, generalmente con una señal luminosa, que algo está funcionando incorrectamente y por tanto es aconsejable que acudamos a un taller para que revisen el automóvil.

Una vez en el taller, el equipo de mecánicos, conectará nuestro automóvil un escáner o lector del sistema OBDII que le facilitara la información almacenada. A principios de los 80, cuando se extendió, el uso de este sistema de diagnóstico, cada fabricante era libre de incorporar su propio conector y utilizar los códigos de error que quisiera. Esto dificultaba mucho la utilización de este sistema para las reparaciones, ya que la inversión que requería en los talleres mecánicos era altísima y poco práctica (debían disponer de muchos lectores y de muchas tablas de códigos).

Para que el uso de este sistema fuera práctico y viable, en 1996, se llegó a un consenso entre los fabricantes y se estandarizaron los códigos y el conector. Así con un único lector de códigos y una tabla de errores, se puede diagnosticar un error en cualquier coche, independientemente del fabricante.

a. EL CONECTOR

El conector del sistema OBDII tiene que cumplir las siguientes especificaciones según la normativa, ISO 15031-3:2004

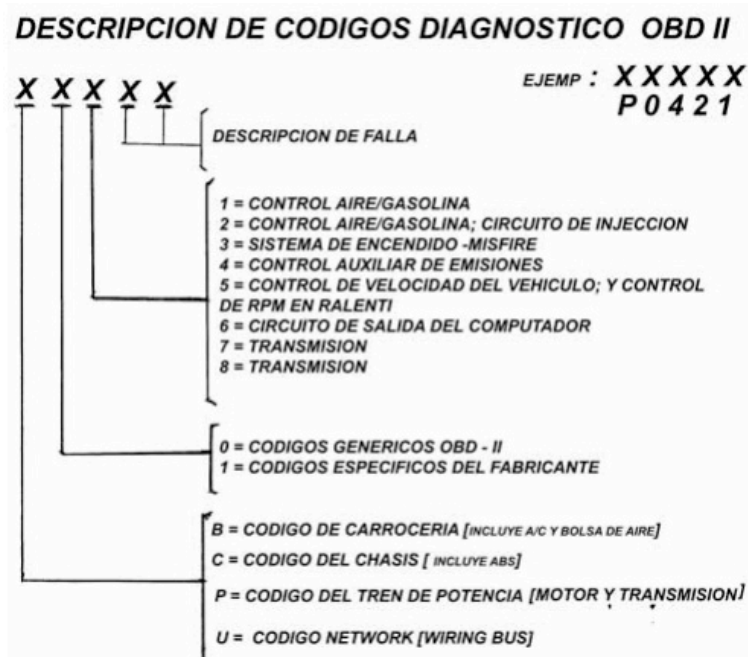


Fuente: On Board Diagnosis OBDII.

Fig. 2.44 Conector OBD II y especificaciones.

b. CÓDIGOS DE ERROR

Los códigos también están estandarizados y deben seguir el esquema siguiente.



Fuente: On Board Diagnosis OBDII.

FIG. 2.45 Códigos de Error OBD II

c. PROTOCOLOS DE COMUNICACIÓN

Existen 3 protocolos de comunicación del sistema OBDII con los lectores. Los fabricantes han escogido que protocolo utilizar y todos los vehículos que salen de su fábrica salen con el mismo protocolo, por tanto es fácil saber que tipo de protocolo funcionan las comunicaciones de nuestro coche. La evolución del OBD II permitió el desarrollo de otros 3 protocolos de comunicación, cuya principal meta fue aumentar la velocidad y reducir los tiempos de multiplexado, permitir un mayor envío de datos con mayor seguridad.

Estos 6 protocolos de comunicación son:

- ISO 9141-2 en vehículos Europeos, Asiáticos y Chrysler con variantes(Key Word Protocol=Palabra Clave)
- SAE J1850 VPW que significa Ancho de Pulso Variable (Variable Pulse Width) y lo utiliza GM USA
- SAE J1850 PWM que indica Modulación Ancho de Pulso (Pulse Width Modulation) utilizado por Ford USA.
- KWP 1281 y KWP 2000 utilizado por el grupo VAG.
- ISO 14230 que lo utiliza Renault, etc.
- ISO 15765 (CAN)

Como es fácil deducir, cada uno de estos protocolos, requiere de un tratamiento de la información, antes de conectar el OBDII con el PC, diferente. Y por tanto, se requieren interfaces de conexión diferentes. Esto no es del todo exacto, ya que existe la posibilidad de fabricar un interfaz de conexión del OBDII con el PC, capaz de utilizar todos los protocolos e incluso seleccionar automáticamente cual es el protocolo utilizado por el vehículo a conectar.

d. LECTORES DE CÓDIGOS

Para poder extraer los datos del OBDII de un vehículo, se necesita un interfaz de conexiones, que recodifique la información que obtiene del vehículo, para que esta pueda ser entendida por el software del pc.

Todos los sistemas que se han visto hasta el momento, están formados por:

- Cable de conexión de del conector OBDII al Interfaz
- Interfaz de conexiones
- Cable del interfaz al puerto COM del PC.
- Software para lectura de los códigos

Aunque existen otras posibilidades, algo más simplificadas, y que pueden ser adquiridas fácilmente. Se trata de instrumentos de lectura de códigos, que disponen de capacidad de lectura del OBDII sin necesidad de ningún PC. Estos sistemas realizan el tratamiento de la información del OBDII del vehículo y muestran en su pantalla los códigos de error.



Fuente: Grupo de investigación.

FIG. 2.46 Lectores de Códigos OBD II Simplificados

Actualmente, las nuevas tecnologías como lo son los dispositivos multitáctiles, nos permiten crear una compuerta de comunicación con los protocolos anteriormente descritos. Esta conectividad se la puede realizar mediante Bluetooth o mediante WI-FI, dependiendo las necesidades y los alcances que se quieran obtener.

En esta tesis, se busca elaborar dicha compuerta de comunicación con una interface universal única, con conectividad Wi-Fi, con las siguientes características:

- Monitoreo activo.
- Registro de datos en tiempo real.
- Velocidad del vehículo (incluyendo tiempo de 0 – 60)
- RPM
- Consumo de combustible
- Temperatura del refrigerante del motor
- Presión del combustible
- Comprobación del Check Engine y PIDs, etc.

Esta comunicación se produce directamente con la ECU del vehículo a través de un hardware OBDII, el mismo que, electrónicamente, convierte la señal digital en señal inalámbrica para su posterior interpretación. Para la comunicación entre nuestro dispositivo multitáctil, en este caso tanto Ipad como Ipod de la marca Apple, se necesita una programación de tipo Objective C, para la creación de la ipa especifica para el enlace.



Fuente: Apple Store. Automotive Apps.

Fig. 2.47 Medidores digitales, registro y ubicación en la Ipa de Ipad

2.4.5 INTERFACES DE CONEXIÓN

Como ya se expuso anteriormente, para que exista una comunicación entre el software y la ECU del automóvil se estableció una normativa para todos los fabricantes, con un conector en común, pero cada grupo encargado de este objetivo dio paso a diferentes interfaces para la conexión con cada

protocolo, los mismos que serán explicados a continuación, con sus respectivas características y planos eléctricos de interpretación de datos.

a. ISO 9141-2

La normativa ISO 9141-2 es de uso generalizado en vehículos Europeos, Asiáticos y de la marca Chrysler con variantes muy poco significativas, este protocolo tiene un tasa de intercambio de datos de 10.4 kb y en su estructura es similar a la normativa RS-232. En la estructura del conector se diferencia por cuanto el pin 7 se denomina línea K, el pin 15 línea L el mismo que es opcional y en ocasiones referido como línea K dual. La señalización es de tipo UART, con una señal de tipo alta en donde el puerto de entrada de voltaje es Vbatt. La principal característica es que el largo de la comunicación esta restringida a 12 bytes, incluido el CRC.

b. SAE J1850 PWM

Esta norma de comunicación es estándar de Ford Motor Company, con una tasa de intercambio de datos a 41.6 kb, cuyos pines de enlace pin2 se denomina Bus-, pin10 Bus+, con un voltaje referencial de +5V, y una longitud de mensaje restringida a 12 bytes. Su principal ventaja es el uso de un esquema multi-master arbitrario llamado “Acceso de manejo sensible múltiple con no-destruccion arbitraria (CSMA/NDMA)”.

c. SAE J1850 VPW

La denominación VPW significa de Ancho de Pulso Variables, cuyo desarrollo fue un gran avance y ayuda a la multiplexación y a la comunicación de varias ECUs en un solo automóvil, así como al manejo de sus componentes.

La tasa de enlace se establece entre los 10.4 y 41.6 kb, siendo estándar de manejo de datos de General Motors. En el conector el pin2 se llama BUS+ el cual maneja una conexión negativa, con un voltaje referencial de +7v y

un punto llamado de “decisión” a +3.5V. El largo de mensaje se ubicó en 12 bytes.

d. ISO 9141

Predecesor de la norma ISO 9141-2, su uso fue momentáneo por cuanto se necesitaba un enlace para la formación de prototipos, sus principales usuarios se actualizaron a la norma consecuente, por cuanto su estructura se estandarizo en las compañías de producción.

e. ISO 14230

También llamado Keyword Protocol 2000, usado por la mayoría de constructores europeos y asiáticos como Alfa Romeo, Audi, BMW, Citroen, Fiat, Honda, etc. En su conector el pin 7 es llamado Línea K, el pin 15 Línea L o también referido como línea doble K. Físicamente idéntico a la ISO 9141-2, el intercambio de datos tiene una tasa de 1.2 a 10.4 kb, donde un mensaje podría contener hasta 255 bytes en el campo de datos estructural.

f. ISO 15765 (CAN)

Este protocolo de comunicación se establecería como el estándar en la industria automotriz. Para el año 2008, todos los vehículos vendidos en Canadá y US tendrían que estar equipados con la red CAN, consecuentemente eliminando la incompatibilidad de los cinco existentes protocolos firmados digitalmente.

Esta red es simplemente un par de cables con un resistor de 120 Ohms en ambas terminaciones de la red en la mayoría de los casos. Los únicos componentes conectados a la red CAN son unidades de control electrónico (nodos). Otros componentes, como sensores, motores, luces, switches, etc, son cableados directamente a los nodos existentes específicos según el caso.

En los automóviles podrían existir más de un sistema de red CAN, y en casos de automóviles de última tecnología podría incluir, adicional a esta red, las redes MOST, LIN, Bluetooth y FlexRay.

En el caso de que la red Can sea usada para el diagnósticos a bordo, solo puede responder a una petición desde un probador o escáner compatible con este tipo de tecnología, caso de vehículos desde el año 2008, que en algunos casos fueron diseñados bajo la norma ISO 15765, también conocida como Diagnóstico en CAN.

La tasa de comunicación es muy alta en comparación con el resto de protocolos, varía entre los 250kbit/seg y los 500kbit/seg. Físicamente puede ser identificados ya que en el conector sus pines de comunicación son pin 6 como CAN de tipo alto, y pin 14 como CAN de tipo bajo.

g. CÓDIGOS OBD II

La lista completa de codificación OBDII se expresará en su totalidad en el Anexo A.

CAPÍTULO 3

PLANTEAMIENTO DE HIPÓTESIS

3.1. HIPÓTESIS

3.1.1. HIPÓTESIS GENERAL

Un sistema universal de diagnóstico predictivo integral con dispositivos táctiles y tecnología Wi-Fi de última generación para automotores, bajo norma OBDII, puede ser usado como herramienta para el estudio, análisis y puesta a punto de los sistemas mecánicos y de gestión electrónica de vehículos estándar y de competición.

3.1.2. HIPÓTESIS ESPECÍFICAS

- Los profesionales como estudiantes deben comprender nueva información referente a lenguajes informáticos de mayor utilización en cuanto a tecnología.
- El lenguaje de programación Objective-C permite especializarse en campos de la informática automotriz y provee una compuerta de introducción en el diseño de sistemas avanzados automotrices computarizados.
- El número PIDs a ser analizados dependen de la marca y procedencia del vehículo, así como del nivel de avance tecnológico empleado en la construcción de la ECU.

3.2. VARIABLES DE INVESTIGACIÓN

3.2.1. VARIABLE INDEPENDIENTE

Sistema de comunicación Wi-Fi implementado para la transmisión de datos de módulos electrónicos y parámetros mecánicos de un automotor comandado por un software dedicado NUI.

3.2.2. VARIABLE DEPENDIENTE

Procedimiento de diagnóstico predictivo de condiciones eléctricas, electrónicas y mecánicas, a corta y a la larga distancia, con la utilización eficiente de dispositivos táctiles de la marca Apple.

3.2.3. OPERACIONALIZACIÓN DE VARIABLES

i. Variable independiente

Tabla 3.1. Operacionalización de variable independiente.

CONCEPTO	CATEGORÍA	INDICADOR	PREGUNTAS
Sistema de comunicación Wi-Fi implementado para la transmisión de datos de módulos electrónicos y parámetros mecánicos de automotores, comandado por un software dedicado NUI.	Sistema de comunicación Wi-Fi.	Número de componentes a la red inalámbrica Wi-Fi	¿Cómo puedo realizar un diseño para la implementación de una red inalámbrica Wi-Fi?
	Diagnóstico Predictivo	Número de parámetros mecánicos y electrónicos	¿Qué parámetros mecánicos y electrónicos son necesarios determinar en un diagnóstico predictivo?
	Software dedicado NUI	Personalización gráfica de la interfaz	¿Cómo puedo personalizar la interfaz gráfica de acuerdo a las necesidades de estudio?

ii. Variable dependiente

Tabla 3.2. Operacionalización de variables dependientes.

CONCEPTO	CATEGORÍA	INDICADOR	PREGUNTAS
Procedimiento de diagnóstico predictivo de condiciones eléctricas, electrónicas y mecánicas, a corta y a la larga distancia, con la utilización eficiente de dispositivos táctiles de la marca Apple.	Sistema de comunicación Wi Fi.	Número de componentes emisores y receptores, en una red inalámbrica Wi-Fi	¿Qué componentes forman una red inalámbrica Wi-Fi y cuáles son sus características?
	Diagnóstico Predictivo	Número de procedimientos de análisis mecánico y electrónico	¿Qué procedimientos de análisis mecánicos y electrónicos son necesarios para determinar en un diagnóstico predictivo?
	Software dedicado NUI	Cantidad de datos en comunicación y recepción.	¿Qué pasos debo seguir para establecer la recepción de datos entre el emisor Wi-Fi y el Software dedicado NUI personalizado?

Fuente: Grupo de investigación.

CAPÍTULO 4

DISEÑO DEL SISTEMA INALÁMBRICO DE DIAGNÓSTICO PREDICTIVO TÁCTIL.

4.1. DISEÑO DE REDES DE COMUNICACIÓN.

El primer paso para diseñar nuestro proyecto, es seleccionar el ambiente de conexión inalámbrica adecuado, tomando en cuenta los siguientes parámetros:

- Mayor alcance nominal.
- Fiabilidad y estabilidad de la conexión.
- Ancho de Banda y nivel de la señal.
- Obstáculos potenciales que interrumpas el enlace.
- Interferencias.
- Seguridad en la transmisión.
- Programación a ser utilizada.
- Protocolos de comunicación.

Nuestra red de enlace, entre dispositivo táctil e interfaz de generación de señal, se ve optimizada, tanto en comunicación de envío como de recepción, con la utilización de la interconexión de tipo WI-FI ya que, en comparación de el resto de tipos de tecnologías de comunicación, nos da las siguientes características:

- **Alcance** hasta 100 mts en lugares abiertos, dependiendo de estado meteorológicos y otros factores de recepción, como alcance del generador de señal, alcance del receptor, etc.

- **Fiabilidad** alta, con la utilización de una máscara de subnet y una dirección IP estática propia, y un puerto de comunicación dedicado para el envío y recepción de bits de información.
- **Ancho de Banda** de 3 Mbps nominal, más que suficiente para cadenas de datos que van desde 120kb hasta 1Mb como tope de información.
- **Obstáculos e interferencias** que ésta red puede sobrepasar, los mismos que son mínimos, ya que el automóvil no está diseñado como un espacio totalmente cerrado para la comunicación de radiofrecuencias.
- **Seguridad de transmisión** mediana para ser una medio inalámbrico.
- **Programación** compatible con la empleada en el dispositivo, que es de tipo Objective-C, nativa de aplicaciones UNIX para dispositivos táctiles.
- **Protocolos de comunicación** IEEE 802.11b, óptimos con el receptor y el generador de señal, que utilizan protocolos de tipo estándar 802.11a/b/g.

4.1.1. CONECTIVIDAD WI-FI

Para un eficiente envío de datos, así como la recepción de los mismos, la tecnología WI-FI tiene varios parámetros (Dirección IP, Máscara de Subnet y dirección de router), los mismos que establecen una identidad virtual que ubica a cada dispositivo como receptor o generador dentro de la comunicación.

En nuestro proyecto utilizaremos los siguientes identificadores:

Dirección IP: 192.168.0.11

Máscara de Subnet: 255.255.255.0

Se utiliza un dirección IP con unidad 11 para evitar interferencias y que los envíos no se multipliquen en el ambiente, ya que por lo general, las redes de tipo infraestructura usan unidades que varían desde el 1 hasta el 10.

No se utiliza un identificador de tipo dirección de router porque, nuestra red diseñada va a ser de tipo Ad-Hoc, donde no se usa un hotspot para la unión a la red wireless.

En algunas ocasiones, también se establecerá un puerto de conexión dedicado, el mismo que, de acuerdo a nuestro diseño, será el puerto 35000. Su determinación y utilidad se determina de acuerdo a los niveles de seguridad que se requieran, y el alcance que se desee lograr.

4.1.2 RED PUNTO A PUNTO INTERFAZ – MÓDULOS DE CONTROL

Establecido el mecanismo de direccionamiento WI-FI, se procede a conformar la red de tipo punto a punto, entre nuestra interfaz generadora de señal, y el módulo de control y recepción de datos.

Al utilizar aplicaciones dedicadas de interfaz única de usuario (IPA), se debe usar una programación compatible con OBDNet, nativa y estandarizada de OBDII en la mayoría de modelos y marcas presentes, por lo que Objective-C es la mejor opción en cuestión de compilación y manejo de variables para este proyecto.

El lenguaje Objective-C nos provee las siguientes cualidades positivas con respecto a el resto de lenguajes informáticos existentes:

- Nativo en la creación de IPAs.
- Optimizado en el uso de recursos de hardware.
- Velocidad mayor de muestreo.
- Mejor gestión de memoria.

- Perfecta integración entre editores, compiladores, debuggers y simuladores.
- Conectividad de alta velocidad.
- Potencia, eficacia, sintaxis limpia, y fácil aplicación.

Como Objective-C es un lenguaje de programación nativo de desarrollo informático sdk, que se utiliza para la creación de IPAs, en esta parte del diseño se van a explicar solamente las funciones que se utilizarán posteriormente en el software de personalización; y ,ya que nosotros no vamos a compilar ni realizar debuggs, se va a exponer de igual manera, qué operación realiza cada una de las funciones, qué datos se obtienen y su respectiva procedencia, así como la forma básica de empleo de la información recibida y enviada, sin adentrarnos mucho en su estructura original de conformación.

Para la conexión entre interfaz y módulo de control, primero se debe inicializar el OBDNet, como lo expresamos anteriormente. Para realizar este proceso en Objective-C se llama a la función OBDNetInitialize, la misma que analiza y establece que tipo de lenguaje de programación informática se esta utilizando, para enviar información que sea de lectura para nuestro respectivo protocolo.

Después de haber inicializado OBDNet, se necesita crear una nueva función con el nombre OBDNET_HANDLE para identificar nuestra conexión, en este caso inalámbrica WI-FI. Con ésta nosotros podemos extender nuestra red punto a punto a una red en malla, para no solamente tener un dispositivo táctil al alcance, sino varios tablets recibiendo y enviando información al mismo tiempo.

Para finalizar el proceso, nuestra IPA llama las siguientes funciones complementarias:

- **OBDNetCreate** para crear una red inicial con valores en tiempo real fundamental para que, en el manejo de varios vehículos de estudio, no permita la conjunción de datos entre unidades analizadas.
- **OBDNetWorker** para designar parte de la programación a cada campo de análisis que nuestra interfaz requiera, por ejemplo el estudio de PIDs, GLCDs, medidores de tiempo real, y parte de diagnóstico predictivo.

4.1.3 MANEJABILIDAD DE PROTOCOLOS DE COMUNICACIÓN BAJO NORMA OBD II

Ya inicializado nuestro enlace, la programación se debe conectar al respectivo servidor de envío de información, creado entre nuestro generador de señal y la salida del puerto OBDII del automóvil. Este servidor de envío es diferente de acuerdo a la complejidad de la ECU, y a la cantidad de dispositivos electrónicos que esta maneja.

La función `OBDNetConnection` nos permite entrar en este servidor, cuyo único requisito es que tengamos `OBDNET_HANDLE` activo como parte de la red.

Como paso inicial de la obtención de datos, la ECU siempre envía primero una lista completa de PIDs activos, si este fuera el caso, como primer parámetro de análisis. La función `OBDNetGetPidList` realiza este estudio, con información procedente de datos enviados de cada PID por la función `OBDNET_CALLBACK_PID_LIST_ITEM`.

Los procesos consiguientes en todo el esquema de comunicación del protocolo se detalla a continuación en tablas personalizadas, determinando tanto su uso, su sintaxis, y los miembros que conforman cada uno de estos, para una fácil comprensión del lenguaje Objective-C dentro de nuestra IPA.

a. VALORES TIPO

Como parte básica de nuestro programa, se definen los valores tipo que se van a utilizar. En el caso del OBDNet estos valores tipo son OBDNET_PID_VALUE y OBDNET_PID_INFO.

i. OBDNET_PID_VALUE

Tabla 4.1. OBDNET_PID_VALUE

USO
Establece el valor real de lectura informática en bits del parámetro PID que se requiera. Se puede obtener mediante la utilización de la función OBDNetParameterGetCurrentValue o mediante el callback de definición #define OBDNET_CALLBACK_STREAMING_PID_VALUE.
SINTAXIS
<pre>typedef struct { unsigned long PidValueType; unsigned long TimeStamp; double EnglishValue; double MetricValue; const char * EnglishValueString; const char * MetricValueString; const char * UniqueId; } OBDNET_PID_VALUE;</pre>
PARÁMETROS
<ul style="list-style-type: none">• PidValueType es el tipo de dato que la estructura contiene. Puede ser OBDNET_PID_VALUETYPE_NUMBER, en caso de valores numéricos únicos, POBDII_PID_VALUETYPE_STRING en caso de una cadena de valores específicos OBDII. Además indica si este valor es valido para EnglishValue - EnglishValueString en el sistema inglés de medición, o MetricValue - MetricValueString en el sistema métrico.• TimeStamp establece el número de milisegundos desde que la petición de información.• EnglishValue, el valor en el Sistema Inglés, si el PidValueType es POBDII_PID_VALUETYPE_NUMBER.• MetricValue, el valor en el Sistema Métrico, si el PidValueType es POBDII_PID_VALUETYPE_NUMBER.• EnglishValueString, una cadena de valores en el sistema inglés, si el PidValueType es POBDII_PID_VALUETYPE_STRING.

- MetricValueString, una cadena de valores en el sistema métrico, si el PidValueType es POBDII_PID_VALUETYPE_STRING.
- UniqueId identifica el PID específico. Por ejemplo “SAE.RPM”, etc.

Fuente: OBDNet SDK.

ii. *OBDNET_PID_INFO*

Tabla 4.2. OBDNET_PID_INFO

USO
Establece la información del PID requerido, usando esta estructura cuando se la solicita a la función determinada OBDNetGetParameterGetInfo.
SINTAXIS
<pre>typedef struct { unsigned long Supported; unsigned long Priority; const char * UniqueId; const char * Name; const char * Category; const char * EnglishUnits; const char * MetricUnits; } OBDNET_PID_INFO;</pre>
PARÁMETROS
<ul style="list-style-type: none"> • Supported indica si el PID es permitido o está soportado dentro de la información receptada, si nos devuelve el valor Zero, quiere decir que no esta soportada, mientras que si recibimos Non-Zero, el PID está disponible . • Priority es la prioridad del PID dentro de una serie de datos, si existiera más de un PID que señalar. Además nos señala cuan a menudo el PID se actualiza dentro del enlace presente. Sus valores pueden ser 1, 2 o 3. Por defecto, la prioridad de valor 1 son actualizados cada fracción o frame; de valor 2 son actualizados cada 10 prioridades de un frame, y finalmente de valor 3 se actualiza cada 5 prioridades de 2 frames, entendiéndose como frame al ciclo de recolección de datos provenientes de la ECU. • UniqueId identifica el PID específico. Por ejemplo “SAE.RPM”, etc. • Name, el nombre respectivo del PID, por ejemplo “Engine RPM”, etc. • Category, la categoría del PID, por ejemplo “Misfire”. • EnglishUnits, las unidades inglesas del PID, por ejemplo “MPH” o “RPM”. • MetricUnits, las unidades métricas del PID, por ejemplo “KPH” o “RPM”.

Fuente: OBDNet SDK.

b. CALLBACKS O PETICIONES DE RETORNO.

Luego de haber establecido nuestros valores tipo, cada una de las funciones incluidas en su sintaxis, realiza los llamados CALLBACKS o Peticiones de retorno de información. Los CALLBACKS por defecto para OBDNet son:

- `#define OBDNET_CALLBACK_PID_LIST_ITEM`
- `#define OBDNET_CALLBACK_SELECTED_PID_ITEM`
- `#define OBDNET_CALLBACK_PID_VALUE.`

i. *OBDNET_CALLBACK_PID_LIST_ITEM*

Tabla 4.3. OBDNET_CALLBACK_PID_LIST_ITEM

USO
Este callback va a ser requerido una vez por cada PID dentro de la lista de PIDs cuando se llame a la función OBDNetParameterGetPidList.
SINTAXIS
<pre>typedef void (OBDNET_CALLBACK * OBDNET_CALLBACK_PID_LIST_ITEM_PROTOTYPE) (OBDNET_HANDLE OBDNetHandle, const char * lpszPidUniqueId)</pre>
PARÁMETROS
<ul style="list-style-type: none">• OBDNetHandle es el direccionamiento hacia la conexión OBDNet de la cual es obtenida la lista de PIDs.• lpszPidUniqueId es la única identificación ID de un PID en la lista.
VALOR DE RETORNO
Esta función no nos regresa un valor.

Fuente: OBDNet SDK.

ii. *OBDNET_CALLBACK_SELECTED_PID_ITEM*

Tabla 4.4. *OBDNET_CALLBACK_SELECTED_PID_ITEM*

USO
Este callback va a ser requerido una vez por cada PID dentro de la lista de PIDs cuando se llame la función <code>OBDNetLoggingGetSelectedPidList</code> .
SINTAXIS
<pre>typedef void (OBDNET_CALLBACK * OBDNET_CALLBACK_SELECTED_PID_ITEM_PROTOTYPE) (OBDNET_HANDLE OBDNetHandle, const char * lpszPidUniqueId)</pre>
PARÁMETROS
<ul style="list-style-type: none">• <code>OBDNetHandle</code> es el direccionamiento hacia la conexión <code>OBDNet</code> de la cual es obtenida la lista de PIDs.• <code>lpszPidUniqueId</code> es la única identificación ID de un PID en la lista.
VALOR DE RETORNO
Esta función no nos regresa un valor.

Fuente: `OBDNet SDK`.

iii. *OBDNET_CALLBACK_STRAMING_PID_VALUE*

Tabla 4.5. *OBDNET_CALLBACK_STRAMING_PID_VALUE*

USO
Este callback va a ser utilizado cada vez que el valor transmitido del PID es actualizado.
SINTAXIS
<pre>typedef void (OBDNET_CALLBACK * OBDNET_CALLBACK_STRAMING_PID_VALUE_PROTOTYPE) (OBDNET_HANDLE OBDNetHandle, const OBDNET_PID_VALUE * pPidValue)</pre>
PARÁMETROS
<ul style="list-style-type: none">• <code>OBDNetHandle</code> es el direccionamiento hacia la conexión <code>OBDNet</code> de la cual es

obtenida la lista de PIDs.

- `pPidValue` es el puntero hacia la estructura `OBDNET_PID_VALUE` que contiene los datos del PID.

VALOR DE RETORNO

Esta función no nos regresa un valor.

Fuente: OBDNet SDK.

c. FUNCIONES

Adentrándonos en la parte central del sistema, nuestro protocolo de información requiere el uso de varias funciones, tanto para receptor, leer, analizar, corregir y mostrar todos los datos dentro de la interfaz gráfica de usuario.

Estas funciones, por el trabajo que desempeñan dentro del proceso de nuestro diseño, se reúnen en ocho grandes grupos tipo que son:

- Initialization Functions (Funciones de inicialización)
- Creation Functions (Funciones de creación)
- Status Functions (Funciones de estado)
- Debugging Functions (Funciones de corrección)
- Connection Functions (Funciones de conexión)
- Misc. Functions (Funciones Complementarias)
- Parameter Functions (Funciones de Parámetros)
- Logging Functions (Funciones de enlace)

i. INITIALIZATION FUNCTIONS (FUNCIONES DE INICIALIZACIÓN)

Representan el punto de partida de todo el sistema, ya que permiten activar o desactivar sistemas necesarios para que nuestra API trabaje de manera óptima. Estas funciones de inicialización son:

OBDNetInitialize

Tabla 4.6. OBDNetInitialize

USO				
<p>Esta es la primera función en ser llamada dentro del proceso. Inicializa la OBDNet, además de establecer su naturaleza API de acuerdo al tipo de lenguaje utilizado, en este caso, Objective-C.</p>				
SINTAXIS				
<pre>OBDNET_STATUS OBDNetInitialize(unsigned long ApiType, void * lpReserved);</pre>				
PARÁMETROS				
<ul style="list-style-type: none">• <code>ApiType</code> es el tipo de API que se quiere utilizar. Este parámetro puede ser uno de los siguientes: <table border="1"><tbody><tr><td><code>OBDNET_APITYPE_</code> <code>C_PLUSPLUS</code></td><td>C/C++. Indica que se esta codificando en lenguaje C o C++. OBDNet no tiene preferencia sobre alguno de ellos</td></tr><tr><td><code>OBDNET_APITYPE_</code> <code>VISUALBASIC</code></td><td>Visual Basic. Indica que se esta codificando en lenguaje Visual Basic. En este caso el OBDNet si requiere un manejo especial para parámetros en cadena y valores de retorno, por lo que deben estar perfectamente contruidos para relacionarse con la interfaz VisualBasic</td></tr></tbody></table>	<code>OBDNET_APITYPE_</code> <code>C_PLUSPLUS</code>	C/C++. Indica que se esta codificando en lenguaje C o C++. OBDNet no tiene preferencia sobre alguno de ellos	<code>OBDNET_APITYPE_</code> <code>VISUALBASIC</code>	Visual Basic. Indica que se esta codificando en lenguaje Visual Basic. En este caso el OBDNet si requiere un manejo especial para parámetros en cadena y valores de retorno, por lo que deben estar perfectamente contruidos para relacionarse con la interfaz VisualBasic
<code>OBDNET_APITYPE_</code> <code>C_PLUSPLUS</code>	C/C++. Indica que se esta codificando en lenguaje C o C++. OBDNet no tiene preferencia sobre alguno de ellos			
<code>OBDNET_APITYPE_</code> <code>VISUALBASIC</code>	Visual Basic. Indica que se esta codificando en lenguaje Visual Basic. En este caso el OBDNet si requiere un manejo especial para parámetros en cadena y valores de retorno, por lo que deben estar perfectamente contruidos para relacionarse con la interfaz VisualBasic			
<ul style="list-style-type: none">• <code>lpReserved</code> es Reservado, mostrando Null para este parámetro.				
VALOR DE RETORNO				
<p>Si la función logra su cometido nos da un valor de retorno <code>OBDNET_STATUS_SUCCESS</code>. Si falla nos da cualquier otro valor, pasando este valor a <code>OBDNetStatusGetErrorCodeString</code> para obtener una descripción textual del error mostrado.</p>				

Fuente: OBDNet SDK.

OBDNetUninitialize

Tabla 4.7. OBDNetUninitialize

USO
Su utilización se produce cuando hemos terminado nuestro trabajo con la OBDNet y deseamos cerrarla correctamente. Es la última función en ser llamada.
SINTAXIS
<pre>OBDNET_STATUS OBDNET_API OBDNetUninitialize (void);</pre>
VALOR DE RETORNO
Si la función logra su cometido nos da un valor de retorno OBDNET_STATUS_SUCCESS. Si falla nos da cualquier otro valor, pasando este valor a OBDNetStatusGetErrorCodeString para obtener una descripción textual del error mostrado.

Fuente: OBDNet SDK.

ii. CREATION FUNCTIONS (FUNCIONES DE CREACIÓN)

Como requisito fundamental, es necesario crear ciertos elementos adicionales para que puedan ser configurados o personalizados por las funciones, por lo que para su automatización dentro del diseño se implementaron funciones de creación. Entre estas tenemos:

OBDNetCreate

Tabla 4.8. OBDNetCreate

USO
Crea un OBDNET_HANDLE. Su uso es indispensable para, como lo explicamos en un principio, nos permita entrar en el servidor de enlace con la ECU. Su uso es aplicado para casi todas las otras funciones OBDNet.

SINTAXIS

```
OBDNET_STATUS OBDNET_API OBDNetCreate (  
    void  
);
```

VALOR DE RETORNO

Obtenemos un **OBDNET_HANDLE** activo.

Fuente: OBDNet SDK.

OBDNetDestroy

Tabla 4.9. OBDNetDestroy

USO

Elimina un **OBDNET_HANDLE** cuando ya terminaste con su utilización. Esta operación es requerida para en cada caso utilizar un único y diferente **HANDLE** activo.

SINTAXIS

```
OBDNET_STATUS OBDNET_API OBDNetDestroy (  
    OBDNET_HANDLE OBDNetHandle  
);
```

VALOR DE RETORNO

Obtenemos un **OBDNET_STATUS_SUCCESS** que indica que la operación de eliminación del archivo fue exitosa. Si se obtiene otro valor distinto se llama a **OBDNetStatusGetErrorCodeString** para obtener una descripción textual del error.

Fuente: OBDNet SDK.

iii. STATUS FUNCTIONS (FUNCIONES DE ESTADO)

Nos indica si la operación fue completada, o si en el proceso hubo un error indeterminado, el mismo que pasa a ser descrito textualmente llamando a la función de estado predeterminada. Para nuestro diseño, ésta función se denomina **OBDNetStatusGetErrorCodeString**.

OBDNetStatusGetErrorCodeString

Tabla 4.10. OBDNetStatusGetErrorCodeString

USO
Esta función nos crea una descripción textual detallada para un código de error cuyo valor de retorno haya sido dado por otra función de tipo OBDNet.
SINTAXIS
<pre>const char * OBDNET_API OBDNetStatusGetErrorCodeString (OBDNET_STATUS ErrorCode);</pre>
PARÁMETROS
<ul style="list-style-type: none">• ErrorCode es la codificación que se obtiene para identificar la descripción del error.
VALOR DE RETORNO
Una cadena de caracteres mantenido dentro de la descripción del error. Especificado por el parámetro ErrorCode.

Fuente: OBDNet SDK.

iv. DEBUGGING FUNCTIONS (FUNCIONES DE CORRECCIÓN)

Su análisis dentro de nuestro proyecto es solamente descriptivo. La introducción en su estudio detenido podría desviar la atención del tema principal, y la confusión general, ya que presenta conocimientos de lenguaje Objective-C de alto rendimiento. Su utilización en el diseño, fue por requisitos de compilación en la personalización.

Son funciones predeterminadas por la API, por la interfaz, por el lenguaje y por la programación para, en caso de que existan errores dentro del esquema de recepción, análisis y muestreo, exista un sistema de corrección automático de estos valores o de la parte de la programación que lo contiene.

v. CONNECTION FUNCTIONS (FUNCIONES DE CONEXIÓN)

Se podría decir que, estas funciones son las venas y arterias de nuestro diseño, por cuanto determinan el ancho de banda que se va a manejar, las vías de direccionamiento, la tasa de muestreo, etc.

En conjunción a una buena señal, se debe usar funciones de optimización, las mismas que son:

OBDNetOpenConnection

Tabla 4.11. OBDNetOpenConnection

USO
Abre o inicia la conexión con el servidor OBDNet. Su puerto definido para esta causa es el 1301, con una tasa de muestreo o timeout de entre 30-60 segundos.
SINTAXIS
<pre>POBDII_STATUS POBDII_API OBDNetOpenConnection (OBDNET_HANDLE OBDNetHandle, const char * lpszServer, unsigned long Port, unsigned long TimeoutInMilliseconds,);</pre>
PARÁMETROS
<ul style="list-style-type: none">• OBDNetHandle es la HANDLE activa creada por OBDNetCreate.• lpszServer es una línea de comando que contiene la dirección IP del servidor OBDNet que se intenta conectar.• Port es el número del puerto que se el servidor ha permitido.• TimeoutInMilliseconds es la cantidad de tiempo, en milisegundos, antes que el servidor declare por fallida la conexión.
VALOR DE RETORNO
Obtenemos un OBDNET_STATUS_SUCCESS que indica que la operación de eliminación del archivo fue exitosa. Si se obtiene otro valor distinto se llama a OBDNetStatusGetErrorString para obtener una descripción textual del error.

Fuente: OBDNet SDK.

OBDNetCloseConnection

Tabla 4.12. OBDNetCloseConnection

USO
Cierra la conexión con el servidor OBDNet.
SINTAXIS
<code>OBDNET_STATUS OBDNET_API OBDNetCloseConnection (OBDNET_HANDLE OBDNetHandle,);</code>
PARÁMETROS
<ul style="list-style-type: none">• OBDNetHandle es la HANDLE activa creada por OBDNetCreate.
VALOR DE RETORNO
Obtenemos un OBDNET_STATUS_SUCCESS que indica que la operación de eliminación del archivo fue exitosa. Si se obtiene otro valor distinto se llama a OBDNetStatusGetErrorCodeString para obtener una descripción textual del error.

Fuente: OBDNet SDK.

OBDNetIsConnected

Tabla 4.13. OBDNetIsConnected

USO
Su uso específico es de verificación de conexión entre el dispositivo y la interfaz. Es posible que la conexión se pierda entre ambos dispositivos sin la necesidad de llamar a OBDNetCloseConnection.
SINTAXIS
<code>BOOL OBDNET_API OBDNetIsConnected (OBDNET_HANDLE OBDNetHandle,);</code>
PARÁMETROS
<ul style="list-style-type: none">• OBDNetHandle es la HANDLE activa creada por OBDNetCreate.
VALOR DE RETORNO
Obtenemos un OBDNET_STATUS_SUCCESS que indica que la operación de eliminación del archivo fue exitosa. Si se obtiene otro valor distinto se llama a OBDNetStatusGetErrorCodeString para obtener una descripción textual del error.

Fuente: OBDNet SDK.

vi. MISC. FUNCTIONS (FUNCIONES COMPLEMENTARIAS)

Su trabajo dentro del proceso es secundario, pero no por eso menos importante. Su inclusión en el diseño se da con el fin de mantener estabilidad en el sistema, y robustez ante fallos de conexión y de seguridad. A continuación se explicarán a breves rasgos su práctica

Tabla 4.14. MISC. FUNCTIONS

FUNCION	USO
OBDNetWorker	Regularmente realiza trabajos procesales como mantener la conexión, el logueo continuo con el servidor, etc. Es recomendable llamarlo desde el loop de conexión principal, además de aplicarse continuamente aún cuando no está conectado.
OBDNetSetCallback	Establece el callback necesario de acuerdo a su necesidad, como se explicó en el apartado CallBacks de este capítulo.
OBDNetSetWorkerTimeout	Establece automáticamente el tiempo de espera, diferente de un modelo de automotor a otro, antes de que dé por fallida la conexión entre servidor y dispositivo.
OBDNetSetFunctionCallTimeout	Tiene una práctica parecida a OBDNetSetWorkerTimeout, solo que en esta ocasión determina el tiempo de espere antes de dar por exitosa o fallida una función dentro del programa.

Fuente: OBDNet SDK.

vii. PARAMETER FUNCTIONS (FUNCIONES DE PARÁMETROS)

Si las funciones de conexión representaban las arterias y venas de nuestro diseño, las funciones de parámetros son el corazón principal del esquema,

ya que son la parte de mayor práctica dentro de lo que a análisis de datos y de comprensión del estado real del automotor se refiere.

De estas operaciones, nosotros obtendríamos todos los valores a ser estudiados, tanto de PIDs, como de mediciones de tiempo real, telemetría y diagnóstico preventivo del automotor. A continuación enlistaremos todas sus características, que usaremos posteriormente, para la personalización de nuestras herramientas de lectura.

OBDNetParameterGetCount

Tabla 4.15. OBDNetParameterGetCount

USO
Solicita el número total de PIDs disponibles
SINTAXIS
<code>unsigned long OBDNET_API OBDNetParameterGetCount (OBDNET_HANDLE OBDNetHandle,);</code>
PARÁMETROS
<ul style="list-style-type: none"> • OBDNetHandle es el HANDLE activo hacia cual está direccionado.
VALOR DE RETORNO
El conteo total de PIDs especificados para OBDNET_HANDLE

Fuente: OBDNet SDK.

OBDNetParameterGetUniqueId

Tabla 4.16. OBDNetParameterGetUniqueId

USO
Solicita el ID único por cada PID disponible en ese momento. Para solicitar la lista entera, es mucho más eficiente llamar a la función OBDNetParameterGetPidList.
SINTAXIS
<code>const char * OBDNET_API OBDNetParameterGetUniqueId (OBDNET_HANDLE OBDNetHandle, unsigned long PidIndex,);</code>
PARÁMETROS
<ul style="list-style-type: none"> • OBDNetHandle es el HANDLE activo hacia cual está direccionado. • PidIndex es la raíz del PID hacia cual va direccionada la petición. Este valor debe encontrarse entre 0 y un valor obtenido por OBDNetParameterGetCount
VALOR DE RETORNO
El ID único para el PID especificado.

Fuente: OBDNet SDK.

OBDNetParameterGetPidList

Tabla 4.17. OBDNetParameterGetPidList

USO
Enumera todos los PIDs disponibles, haciendo una petición de retorno o callback de tipo OBDNET_CALLBACK_PID_LIST_ITEM para cada uno
SINTAXIS
OBDNET_STATUS OBDNET_API OBDNetParameterGetPidList (OBDNET_HANDLE OBDNetHandle,);
PARÁMETROS
<ul style="list-style-type: none">• OBDNetHandle es el HANDLE activo hacia cual está direccionado.
VALOR DE RETORNO
Obtenemos un OBDNET_STATUS_SUCCESS que indica que la operación de eliminación del archivo fue exitosa. Si se obtiene otro valor distinto se llama a OBDNetStatusGetErrorCodeString para obtener una descripción textual del error.

Fuente: OBDNet SDK.

OBDNetParameterGetInfo

Tabla 4.18. OBDNetParameterGetInfo

USO
Solicita información acerca del PID, como por ejemplo las unidades.
SINTAXIS
OBDNET_STATUS OBDNET_API OBDNetParameterGetInfo (OBDNET_HANDLE OBDNetHandle, const char * lpszUniqueId , OBDNET_PID_INFO * lpPidInfo ,);
PARÁMETROS
<ul style="list-style-type: none">• OBDNetHandle es el HANDLE activo hacia cual está direccionado.• lpszUniqueId es la identificación única del PID deseado.• lpPidInfo es un señalador hacia una estructura OBDNET_PID_INFO la que recibirá la información del PID.
VALOR DE RETORNO
Obtenemos un OBDNET_STATUS_SUCCESS que indica que la operación de eliminación del archivo fue exitosa. Si se obtiene otro valor distinto se llama a OBDNetStatusGetErrorCodeString para obtener una descripción textual del error.

Fuente: OBDNet SDK.

OBDNetParameterGetCurrentValue

Tabla 4.19. OBDNetParameterGetCurrentValue

USO
Después de conectar hacia un servidor OBDNet el mismo que esta enviando y recibiendo información, se puede usar esta función para obtener los datos resultantes del PID.
SINTAXIS
<pre>OBDNET_STATUS OBDNET_API OBDNetParameterGetCurrentValue (OBDNET_HANDLE OBDNetHandle, const char * lpszUniqueId, OBDNET_PID_INFO * lpPidInfo,);</pre>
PARÁMETROS
<ul style="list-style-type: none">• OBDNetHandle es el HANDLE activo hacia cual está direccionado.• lpszUniqueId es la identificación única del PID deseado.• lpPidInfo es un señalador hacia una estructura OBDNET_PID_INFO la que recibirá la información del PID.
VALOR DE RETORNO
Obtenemos un OBDNET_STATUS_SUCCESS que indica que la operación de eliminación del archivo fue exitosa. Si se obtiene otro valor distinto se llama a OBDNetStatusGetErrorCodeString para obtener una descripción textual del error.

Fuente: OBDNet SDK.

viii. LOGGING FUNCTIONS (FUNCIONES DE ENLACE)

Como parte final de todo el proceso de manejo del protocolo, se usan operaciones que dan la sustentabilidad a la red inalámbrica, garantizando una lectura fiable y eficaz, característica fundamental para poder realizar una medición y diagnóstico confiable.

Las funciones de enlace, como su nombre lo indica, son el puente tecnológico diseñado para, dentro de un ambiente alámbrico o inalámbrico, encriptar la información de medición, direccionar su ruteo y proveer la seguridad de envío y llegada que el sistema requiere.

En la siguiente tabla se definirán las funciones más apropiadas, para garantizar un enlace de calidad, tomando en cuenta, como lo explicamos anteriormente, los obstáculos, el tipo de red y los medios meteorológicos en que se va a desenvolver la señal.

Tabla 4.20. Logging Functions

FUNCION	USO
OBDNetLoggingGetSelectedPidList	Para obtener la lista de PID seleccionada. Esta lista es regresada mediante la petición de retorno OBDNET_CALLBACK_SELECTED_PID_ITEM
OBDNetLoggingClearSelectedPidList	Limpia la lista PID seleccionada
OBDNetLoggingAddSelectedPid	Adiciona un PID a la lista de PIDs ya seleccionada. Se lo utilizada cada vez que se requiera datos de un PID diferente.
OBDNetLoggingRemoveSelectedPid	Quita un PID a la lista de PIDs ya seleccionada. Se lo utilizada cada vez que ya no se requiera datos de un PID diferente.
OBDNetLoggingGetMode	Determina el estado de enlace del servidor y su interfaz.
OBDNetLoggingSetMode	Establece el tipo de enlace, tomando en cuenta la compatibilidad con el dispositivo.
OBDNetLoggingStartStreamingPid	Permite inicializar el análisis de un PID, cuya información es recibida mediante la petición de retorno OBDNET_CALLBACK_STREAMING_PID_VALUE
OBDNetLoggingStopStreamingPid	Detiene el análisis de un PID, incluido información de envío y recepción.
OBDNetLoggingStopStreamingAllPids	Detiene en su totalidad el enlace de comunicación de PIDs

Fuente: OBDNet SDK.

4.2. DIAGRAMACIÓN DE CONEXIONES

Las funciones de lenguaje Objective-C utilizadas anteriormente permiten a la interfaz comunicarse con los servidores OBDNet, con la finalidad de la adquisición de datos de la lista de PIDs disponible de acuerdo a la tecnología de composición de la ECU.

La interpretación de los bits que circulan en la red OBDII, tanto ceros y unos hablando informáticamente, al ser propios de varios sistemas del automotor, se conectan entre si, demostrando perspectiva de dependencia e independencia unos de otros. Su diagramación teórica viene formada mediante diagramas de bloques SDK y tablas de prioridad lógica SDK.

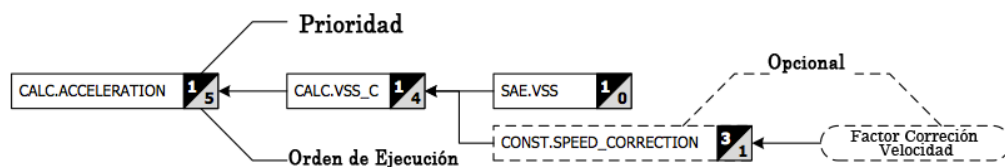


Fig. 4.1. Diagramas de bloques SDK

ID Parámetro (PID)	Descripción PID	Unidades Inglesas	Unidades Métricas	Categoría	Prioridad
CALC.ACCELERATION	Current acceleration based on vehicle speed	E: ft/s ²	M: m/s ²	Performance	P1

Fig. 4.2. Tabla de prioridad Lógica SDK

4.2.1 SISTEMA DE ADQUISICIÓN DE DATOS

OBDNet, lógicamente hablando, subdivide las prioridades dentro del funcionamiento de manejo de la ECU en segmentos de estudio independientes de acuerdo a su concatenación dentro de los diagramas de bloques SDK requeridos.

La segmentación de estudio dentro de la interfaz NUI es:

- Airflow (Flujo de aire).

- Distance (Distancia).
- Emissions (Emisiones).
- Environment (Medio ambiente)
- Fuel (Combustible)
- Performance (Desempeño)
- Velocity (Velocidad)
- System (Sistema)
- Time (Tiempos)
- Transmissions (Transmisión)

Definida internamente la segmentación dentro de la diagramación de bloques, la OBDNet, mediante llamado de funciones y callbacks, agrupa en cada segmento las constantes y variables que el sistema electromecánico de control del automóvil provee, en señal, a la ECU para finalmente otorgar una prioridad dentro del proceso, detallado así:

Tabla. 4.21. Tabla prioridad PID Airflow (Flujo de aire)

ID PARÁMETRO	DESCRIPCIÓN PID	UNIDADES INGLESAS	UNIDADES MÉTRICAS	PRIORIDAD
CALC.BOOST_PRESURE	Estimado de Presión de Compresión	psi	kPa, bar	P1
CALC.MAF	Masa flujo de aire – mejor calc. Método	lb/min	g/s	P1
CALC.MAF_A	Masa flujo aire – A (LOAD_ABS+RPM)	lb/min	g/s	P1
CALC.MAF_B	Masa flujo aire – B (MAP+RPM+IAT)	lb/min	g/s	P1
CALC.MAP	Presión absoluta – mejor calc. Método	inHg	kPa	P1
CALC.MAP_A	Presión Absoluta – A (MAF+RPM+IAT)	inHg	kPa	P1
CALC.MAP_B	Presión Absoluta – B (LOAD_ABS+RPM+IAT)	inHg	kPa	P1
CALC.TRIP.BOOST.AVG.{ABTPF}	Promedio presión compresión para viajes A,B,T,P y F	Psi	kPa, bar	P3
CALC.TRIP.BOOST.MAX_BOOST.{ABTPF}	Máxima presión de compresión para viajes A,B,T,P y F	Psi	kPa, bar	P3
CALC.VACUUM	Cálculo de toma de vacío	inHg	kPa	P1

Fuente: Calc & Cons PID reference.

Tabla. 4.22. Tabla prioridad Distance (Distancia)

ID PARÁMETRO	DESCRIPCIÓN PID	UNIDADES INGLESAS	UNIDADES MÉTRICAS	PRIORIDAD
CALC.DISTANCE	Distancia viajada desde el último reseteo de consumo de combustible dashboard	miles	km	P1
CALC.TRIP.DISTANCE.{ABTPF}	Distancia viajada para viaje A, B, T, P y F	miles	km	P3

Fuente: Calc & Cons PID reference.

Tabla. 4.23. Tabla prioridad PID Emissions (Emisiones)

ID PARÁMETRO	DESCRIPCIÓN PID	UNIDADES INGLESAS	UNIDADES MÉTRICAS	PRIORIDAD
CALC.TRIP.CO2.RATE.{AB TPF}	Promedio de cantidad de emisión de dióxido de carbono viaje A, B, P, T y F	oz/mi	g/km	P3
CALC.TRIP.CO2.TOTAL.{A BTPF}	Emisión total dióxido de carbono para viajes A, B, T, P y F	lb	kg	P3

Fuente: Calc & Cons PID reference.

Tabla. 4.24. Tabla prioridad PID Environment (Ambiente)

ID PARÁMETRO	DESCRIPCIÓN PID	UNIDADES INGLESAS	UNIDADES MÉTRICAS	PRIORIDAD
CALC.AIR_DENSITY	Densidad de aire del ambiente	lb/ft3	Kg/m3	P2

Fuente: Calc & Cons PID reference.

Tabla. 4.25. Tabla prioridad PID Fuel (Combustible)

ID PARÁMETRO	DESCRIPCIÓN PID	UNIDADES INGLESAS	UNIDADES MÉTRICAS	PRIORIDAD
CALC.AFR_ACTUAL	Relación aire/combustible calculada desde lambda actual	---	---	P1
CALC.AFR_COMMANDDED	Relación aire/combustible calculada desde lambda comandada	---	---	P1
CALC.DTE	Distancia para vaciar depósito	miles	km	P1
CALC.FC.AFC.{01..17}	Promedio de economía/consumo de combustible para periodos 1-17	Mpg(US) Mpg(UK)	l/100km	P1
CALC.FC.AVERAGE	Promedio economía combustible desde último reseteo de dashboard	Mpg(US) Mpg(UK)	L/100km	P1
CALC.FC.FUEL_VOLUME	Volumen de combustible consumido desde último reseteo de dashboard	gal(US) gal(UK)	L	P1
CALC.FC.IFC	Economía/consumo instantáneo de combustible	Mpg(US) Mpg(UK)	L/100km	P1
CALC.FC.IFC_AVG	Promedio instantáneo de economía/consumo de combustible	Mpg(US) Mpg(UK)	L/100km	P1
CALC.FILLUP.CORRECTION	Factor de corrección de consumo de combustible obtenido desde dashboard	%	coeficiente	P1
CALC.FILLUP.FUEL_PRICE	Precio ingresado de combustible	cur(input)	Cur(input)	P1
CALC.FILLUP.FUEL_USED	Volumen de combustible usado desde último llenado	gal(US) gal(UK)	L	P1
CALC.FRP	Presión en riel de combustible	Psi	kPa	P1
CALC.FUEL_FLOW	Cantidad de flujo de combustible derivado desde el MAF	gal(US)/h gal(UK)/h	L/h	P1
CALC.FUEL_FLOW_AVG	Promedio de cantidad de flujo de combustible derivado de MAF	gal(US)/h gal(UK)/h	L/h	P1
CALC.FUEL_LEVEL	Volumen de combustible remanente en el tanque de combustible	Mpg(US) Mpg(UK)	L	P1
CALC.LAMBDA_ACTUAL	Lectura actual de lambda desde el sensor de oxígeno	---	---	P1

CALC.TRIP.AFC.{ABTPF}	Consumo promedio combustible para viajes A, B, T, P y F	Mpg(US) Mpg(UK)	L/100km	P3
CALC.TRIP.FILLUPS. {ABTPF}	Número de llenados de tanque para viajes A, B, T, P y F	---	---	P3
CALC.TRIP.FUEL_COST. {ABTPF}	Costo del combustible consumido para viajes A, B, T, P y F	Moneda	Moneda	P3
CALC.TRIP.FUEL_FLOW. AVG.{ABTPF}	Cantidad Promedio Flujo de combustible viajes A, B, T, P y F	gal(US)/h gal(UK)/h	L/h	P3
CALC.TRIP.FUEL_FLOW. MAX.{ABTPF}	Máximo cantidad flujo combustible para viaje A, B, T, P, y F	gal(US)/h gal(UK)/h	L/h	P3
CALC.TRIP.FUEL_USED.{AB TPF}	Consumo de combustible en viaje A, B, T, P, y F	gal(US) gal(UK)	L	P3
CALC.TTE	Tiempo para vaciar el tanque	hh:mm	hh:mm	P1

Fuente: Calc & Cons PID reference.

Tabla. 4.26. Tabla prioridad PID General (General)

ID PARÁMETRO	DESCRIPCIÓN PID	UNIDADES INGLESAS	UNIDADES MÉTRICAS	PRIORIDAD
CALC.TRIP.STOPS.{ ABTPF}	Número de detenciones para viajes A, B, T, P y F	---	---	P3

Fuente: Calc & Cons PID reference.

Tabla. 4.27. Tabla prioridad PID Performance (Desempeño)

ID PARÁMETRO	DESCRIPCIÓN PID	UNIDADES INGLESAS	UNIDADES MÉTRICAS	PRIORIDAD
CALC.ACCELERATION	Aceleración actual basada en la velocidad del vehículo	ft/s2	m/s2	P1
CALC.ACCELERATION_G	Aceleración actual expresada en unidades de gravedad	G	G	P1
CALC.ENGINE_POWER	Poder del motor calculado hacia las ruedas	Hp	kW, ps	P1
CALC.ENGINE_TORQUE	Torque del motor calculado hacia las ruedas	Lb.ft	N.m, kgf.m	P1
CALC.TRIP.ACCEL.BRAKING .MAX.{ABTPF}	Máxima aceleración de frenado para viajes A, B, T, P y F.	G	G	P3
CALC.TRIP.ACCEL. FORDWARD.MAX.{ABTPF}	Máxima aceleración de avance para viajes A, B, T, P y F	G	G	P3
CALC.TRIP.POWER.MAX.{A BTPF}	Máximo poder del motor para viajes A, B, T, P, y F	hp	Kw, ps	P3
CALC.TRIP.TORQUE.MAX.{A BTPF}	Máximo torque para viajes A, B, T, P y F	Lb.ft	N.m, kgf.m	P3

Fuente: Calc & Cons PID reference.

Tabla. 4.28. Tabla prioridad PID Speed (Velocidad)

ID PARÁMETRO	DESCRIPCIÓN PID	UNIDADES INGLESAS	UNIDADES MÉTRICAS	PRIORIDAD
CALC.TRIP.RPM.AVG.{ABTP F}	Velocidad promedio del motor para viajes A, B, T, P, y F.	rpm	r/min	P3

CALC.TRIP.RPM.MAX.{ABTF}	Velocidad máxima del motor para viajes A, B, T, P, y F.	rpm	r/min	P3
CALC.TRIP.SPEED.AVG_NI.{ABTPF}	Velocidad promedio del vehículo para viajes A, B, T, P, y F.	mph	Km/h	P3
CALC.TRIP.SPEED.AVG_WI{ABTPF}	Velocidad promedio del vehículo para viajes A, B, T, P, y F.	mph	Km/h	P3
CALC.TRIP.SPEED.MAX.{ABTPF}	Velocidad máxima del vehículo para viajes A, B, T, P, y F.	mph	Km/h	P3
CALC.VSS_C	Velocidad corregida del vehículo.	mph	Km/h	P1
CALC.VSS_D	Velocidad del vehículo derivada de RPM	mph	Km/h	P1

Fuente: Calc & Cons PID reference.

Tabla. 4.29. Tabla prioridad PID System (Sistema)

ID PARÁMETRO	DESCRIPCIÓN PID	UNIDADES INGLÉSAS	UNIDADES MÉTRICAS	PRIORIDAD
CALC.FC.AFC	Promedio economía/consumo combustible para periodos pasados.	Mpg(US) Mpg(UK)	L/100km	P1
CALC.FILLUP	Maneja el valor señalado en el dashboard	---	---	P1
CALC.FUEL_CONSUMPTION	Colección de CALC PIDs de economía/consumo de combustible.	---	Km/h	P3
CALC.GEAR	Cálculos relacionados a la marcha	---	Km/h	P1
CALC.LAMBDA_COMMANDED	Lambda comandada directamente por la ECU	---	Km/h	P1
CALC.TRIP	Colección de todos los CALC PIDs de viaje computarizados	---	Km/h	P3
CALC.TRIP.ACCEL	Estadísticas de aceleración para viaje computarizado	---	---	P1
CALC.TRIP.AFC	Estadísticas promedio de consumo de combustible para viaje computarizado	---	---	P1
CALC.TRIP.BOOST	Estadísticas de presión de compresión para viaje computarizado	---	---	P1
CALC.TRIP.CO2	Estadísticas de emisiones de CO2 para viaje computarizado	---	---	P1
CALC.TRIP.DISTANCE	Estadísticas de distancia para viaje computarizado	---	---	P1
CALC.TRIP.FILLUPS	Estadísticas de llenado de tanque para viaje computarizado	---	---	P1
CALC.TRIP.FUEL_COST	Estadísticas de costo de combustible para viaje computarizado	---	---	P1
CALC.TRIP.FUEL_FLOW	Estadísticas de flujo de combustible para viaje computarizado	---	---	P1
CALC.TRIP.FUEL_USED	Estadísticas de consumo de combustible para viaje computarizado	---	---	P1
CALC.TRIP.GEAR	Estadísticas de marcha para viaje computarizado	---	---	P1
CALC.TRIP.POWER	Estadísticas de poder del motor para viaje computarizado	---	---	P1
CALC.TRIP.RPM	Estadísticas de velocidad del motor para viaje computarizado	---	---	P1
CALC.TRIP.SPEED	Estadísticas de velocidad del vehículo para viaje computarizado	---	---	P1
CALC.TRIP.STOPS	Estadísticas de número de detenciones para viaje computarizado	---	---	P1
CALC.TRIP.TIME	Estadísticas basadas en el tiempo para	---	---	P1

	viaje computarizado			
CALC.TRIP.TORQUE	Estadísticas de torque del motor para viaje computarizado	---	---	P1
CONST.ADDITIONAL_WEIGHT	Configuración valida de peso adicional	lb	Kg	P3
CONST.CURB_WEIGHT	Configuración valida de peso curb	lb	Kg	P3
CONST.DRAG_COEFFICIENT	Configuración valida de coeficiente drag	---	---	P3
CONST.ENGINE_DISPLACEMENT	Configuración valida de desplazamiento de motor	In3	L	P3
CONST.FINAL_DRIVE_RATIOS	Configuración valida de relación de manejo final	---	---	P3
CONST.FRONTAL_AREA	Configuración valida de área frontal	ft2	m2	P3
CONST.FRP_PID	PID de presión de riel de combustible usada por el sistema	---	---	P3
CONST.FUEL_CO2_CONTENT	Contenido de dióxido de carbono para el tipo de combustible actual	Lb/gal(US) Lb/gal(UK)	Kg/l	P3
CONST.FUEL_DENSITY	Densidad de combustible del tipo de combustible actual.	Lb/gal(US) Lb/gal(UK)	g/l	P3
CONST.FUEL_TANK_CAPACITY	Configuración valida de capacidad del tanque de combustible	Gal(US) Gal(UK)	L	P3
CONST.LAMBDA_ACTUAL_PID	PID de Lambda usado por el sistema	---	---	P3
CONST.LAMBDA_COMMANDED_PID	PID de Lambda comandado usado por el sistema	---	---	P3
CONST.MAF_PID	PID del MAF usado por el sistema	---	---	P3
CONST.MAP_PID	PID del MAP usado por el sistema	---	---	P3
CONST.MAX_RPM	Configuración valida de Máxima velocidad del motor.	Rpm	Rpm	P3
CONST.MIN_RPM	Configuración valida de Mínima velocidad del motor	Rpm	Rpm	P3
CONST.SHIFT_RPM	Configuración valida de punto de cambio de marcha	Rpm	Rpm	P3
CONST.SPEED_CORRECTION	Configuración valida de factor de corrección de velocidad.	%	Coefficiente	P3
CONST.SPEED_ZONES	Zonas de velocidad para cambio de dashboard	---	---	P3
CONST.SPEED_ZONES.{1..6}.MAX_SPEED	Máxima velocidad para cada cambio de marcha	Mph	Km/h	P3
CONST.SPEED_ZONES.{1..6}.MIN_SPEED	Mínima velocidad para cada cambio de marcha	Mph	Km/h	P3
CONST.SPEED_ZONES.{1..6}.SHIFT_SPEED	Velocidad ideal para cada de cambio de marcha	mph	Km/h	P3
CONST.STOICHIOMETRIC_AIR_FUEL_RATIO	Relación estequiométrica aire/combustible para actual combustible	---	---	P3
CONST.TIRE_RESISTANCE	Configuración valida de coeficiente de resistencia al rodaje del neumático	---	---	P3
CONST.VOLUMETRIC EFFICIENCY	Configuración valida de eficiencia volumétrica	%	Coefficiente	P3
CONST.WHEEL_CIRCUMFERENCE	Circunferencia de la rueda usada por el sistema	r/mile	r/km	P3

Fuente: Calc & Cons PID reference.

Tabla. 4.30. Tabla prioridad PID Time (Tiempo)

ID PARÁMETRO	DESCRIPCIÓN PID	UNIDADES INGLESAS	UNIDADES MÉTRICAS	PRIORIDAD
CALC.IS_NEW_DAY	Regresa verdadero cuando un nuevo día comienza	---	---	P3
CALC.TRIP.TIME.DRIVE. {ABTPF}	Tiempo de manejo para viajes A, B, T, P y F	hh:mm	ms	P3
CALC.TRIP.TIME.ELAPSED. {ABTPF}	Tiempo transcurrido para viajes A, B, T, P y F	hh:mm	ms	P3
CALC.TRIP.TIME.IDLE. {ABTPF}	IDLE transcurrido para viajes A, B, T, P y F	hh:mm	ms	P3
CALC.TRIP.TIME.RUN. {ABTPF}	Tiempo activo para viajes A, B, T, P y F	hh:mm	ms	P3
CALC.TRIP.TIME.START.{ABTPF}	Tiempo de inicio para viajes A, B, T, P y F	hh:mm	ms	P3

Fuente: Calc & Cons PID reference.

Tabla. 4.31. Tabla prioridad PID Transmission (Transmisión)

ID PARÁMETRO	DESCRIPCIÓN PID	UNIDADES INGLESAS	UNIDADES MÉTRICAS	PRIORIDAD
CALC.GEAR.CURRENT	Marcha actualmente conectada	---	---	P1
CALC.GEAR.NUM_GEAR	Número de marchas	---	---	P3
CALC.GEAR.SHIFT_INDICATOR	Indica que un cambio de marcha es sugerido y hacia donde.	---	---	P1
CALC.GEAR.SUGGESTED	Mejor marcha sugerida para la velocidad actual	---	---	P1
CALC.TRIP.GEAR.{1..6}. DISTANCE_PCT.{ABTPF}	Porcentaje de distancia viajada en cada marcha para A, B, T, P y F	%	%	P3
CALC.TRIP.GEAR.{1..6}. TIME_PCT.{ABTPF}	Porcentaje de tiempo viajado en cada marcha para A, B, T, P y F	%	%	P3
CALC.TRIP.GEAR.N. DISTANCE_PCT.{ABTPF}	Porcentaje de distancia viajada en neutral para A, B, T, P y F	%	%	P3
CALC.TRIP.GEAR.N. TIME_PCT.{ABTPF}	Porcentaje de tiempo viajado en neutral para A, B, T, P y F	%	%	P3
CALC.TRIP.GEAR.WG. DISTANCE_PCT.{ABTPF}	Porcentaje de distancia viajada en marcha equivocada para A, B, T, P y F	%	%	P3
CALC.TRIP.GEAR.WG. TIME_PCT.{ABTPF}	Porcentaje de tiempo viajado en marcha equivocada para A, B, T, P y F	%	%	P3

Fuente: Calc & Cons PID reference.

4.2.2 SISTEMA DE INDICADORES GRÁFICOS

La preparación del funcionamiento de nuestros indicadores gráficos, previo a la personalización de su interfaz gráfica de usuario, se la concibe mediante la utilización de la información de nuestras tablas de prioridad SDK por segmentos de acuerdo a los sistemas que correspondan, y

mediante operaciones lógicas de procesamiento de datos provenientes de la ECU, provistas por los diagramas de bloques SDK para cada CALC. operacional dentro de su función.

Como se procede con el manejo de variables dependientes e independientes, los resultados de las operaciones CALC. son dependientes de otras funciones relevantes a su naturaleza informática.

Su relación con las tablas de prioridad es implícito, y se lo explica detenidamente a continuación:

- CALC.ACCELERATION

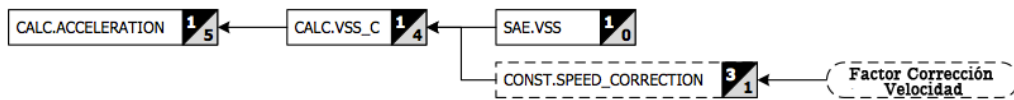


Fig. 4.3. Diagrama SDK CALC.ACCELERATION

Donde Aceleración, A is $A = \Delta d / \Delta t$, donde $\Delta d = (V_0 + V_1) / 2 * \Delta t$, $\Delta t = t_1 - t_0$, v es velocidad, y t es tiempo.

- CALC.ACCELERATION_G

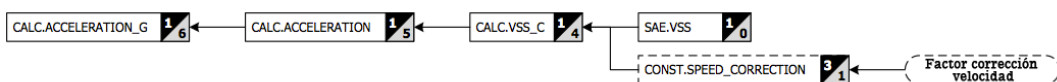


Fig. 4.4. Diagrama SDK CALC.ACCELERATION_G

- CALC.AFR_ACTUAL

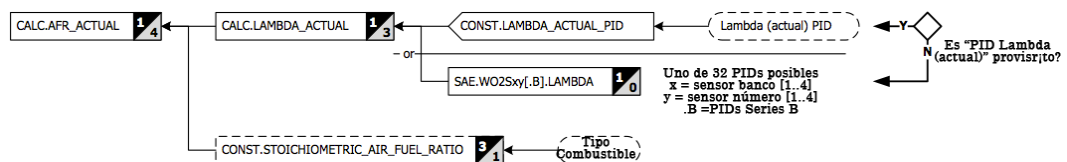


Fig. 4.5. Diagrama SDK CALC.AFR_ACTUAL

$\text{Lambda} = \text{AFR} / \text{AFR}_{\text{stoich}}$, entonces $\text{AFR} = \text{Lambda} * \text{AFR}_{\text{stoich}}$

- CALC.AFR_COMMANDED

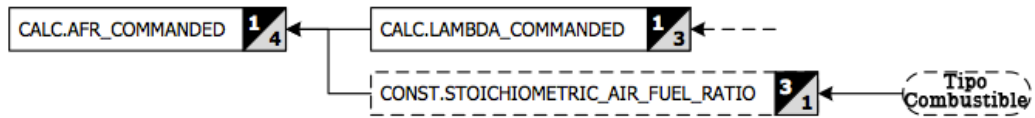


Fig. 4.6. Diagrama SDK CALC.AFR_COMMANDED

- CALC.AIR_DENSITY

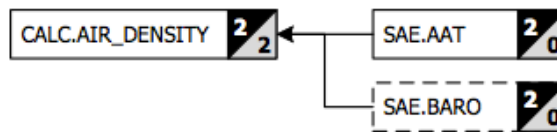


Fig. 4.7. Diagrama SDK CALC.AIR_DENSITY

Presión Barométrica es = 101.325 kPa cuando el PID no está disponible.

Promedio de humedad relativa = 70%

- CALC.BOOST_PRESSURE

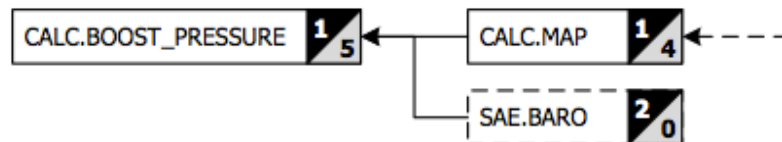


Fig. 4.8. Diagrama SDK CALC.BOOST_PRESSURE

Presión de Compresión = $MAP - P_{baro}$, valores negativos corresponden a vacío, valores positivos corresponden a compresión.

- CALC.DISTANCE

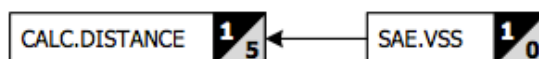


Fig. 4.9. Diagrama SDK CALC.DISTANCE

- CALC.DTE

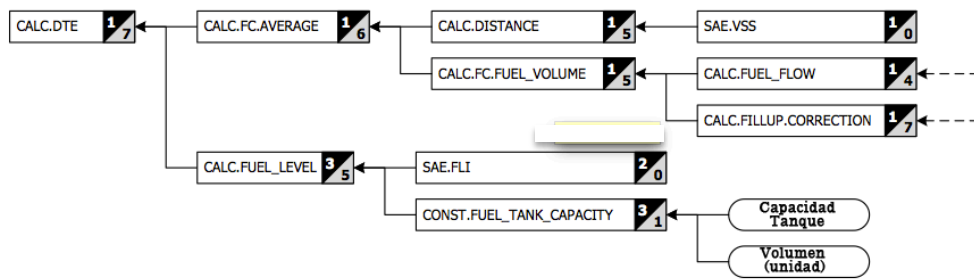


Fig. 4.10. Diagrama SDK CALC.DTE

DTE = Promedio economía combustible [mpg] * Combustible remanente en tanque [gal].

- CALC.ENGINE_POWER

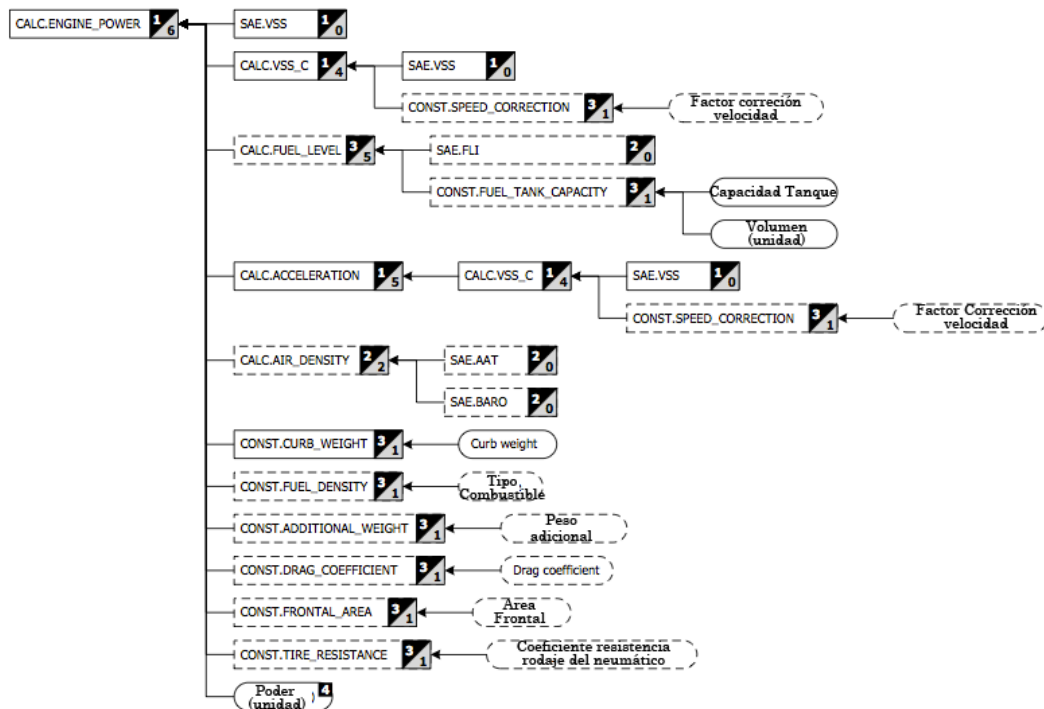


Fig. 4.11. Diagrama SDK CALC.ENGINE_POWER

La salida de poder del motor a las ruedas es calculada de la aceleración y la masa del vehículo. Este cálculo puede también compensar la pérdida de poder debido a la resistencia del aire y la fricción de las llantas contra el material del que se encuentre hecho la calle.

Cuando el PID SAE. FLI no está disponible, el peso del combustible en el tanque puede ser adicionado a la configuración de peso adicional.

$P_e = m \cdot a \cdot v$ donde m es masa, a es aceleración y v es velocidad.

Adicionalmente al ser incluido el **coeficiente drag** y el área frontal del automóvil, se asegura la compensación por la pérdida de poder debido a la resistencia del aire, que a su vez depende del valor PID CALC.AIR_DENSITY que si fuera el caso de no estar disponible dentro de la lista de PIDs, su valor es CALC.AIR_DENSITY 1.2242 [kg/m³].

Así, adicionando estos valores a la ecuación estándar de Poder = $F \cdot v$, y la ecuación de **drag forcé** $F_d = \frac{1}{2} \cdot P \cdot u^2 \cdot C_d \cdot A$, P_d quedaría así:

$$P_d = \frac{1}{2} \cdot P \cdot u^3 \cdot C_d \cdot A$$

donde,

P es la densidad del aire, u es la velocidad del vehículo, C_d es el coeficiente de drag, A es el área frontal.

Conjugando de igual manera el coeficiente de resistencia de los neumáticos a su deslizamiento por fricción, la fuerza de resistencia al rodaje es $F_{rr} = C_{rr} \cdot N_f$, donde N_f es la fuerza normal = $m \cdot (g + a)$, donde m es la masa del vehículo, g es la constante gravitacional, y a es la aceleración arriba o abajo la que asumimos como cero. La pérdida de poder debido al rodaje de los neumáticos, $P_{rr} = C_{rr} \cdot m \cdot g \cdot v$, donde C_{rr} es el coeficiente de resistencia al rodamiento del neumático, m es la masa total del vehículo, g es la constante gravitacional y v es la velocidad del vehículo.

Finalmente el cálculo del poder total es

$$P = P_e - P_d - P_{rr}$$

- CALC.ENGINE_TORQUE

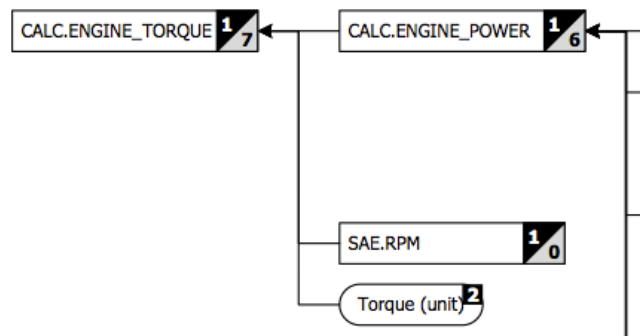


Fig. 4.12. Diagrama SDK CALC.ENGINE_POWER

Como poder $P = T \cdot \omega$ donde T es el torque y ω es la velocidad angular = $2\pi \cdot$ velocidad rotacional, $P = T \cdot 2\pi \cdot$ velocidad rotacional, entonces torque $T = P / (2\pi \cdot$ velocidad rotacional). π es el valor de pi.

- CALC.FC.AFC

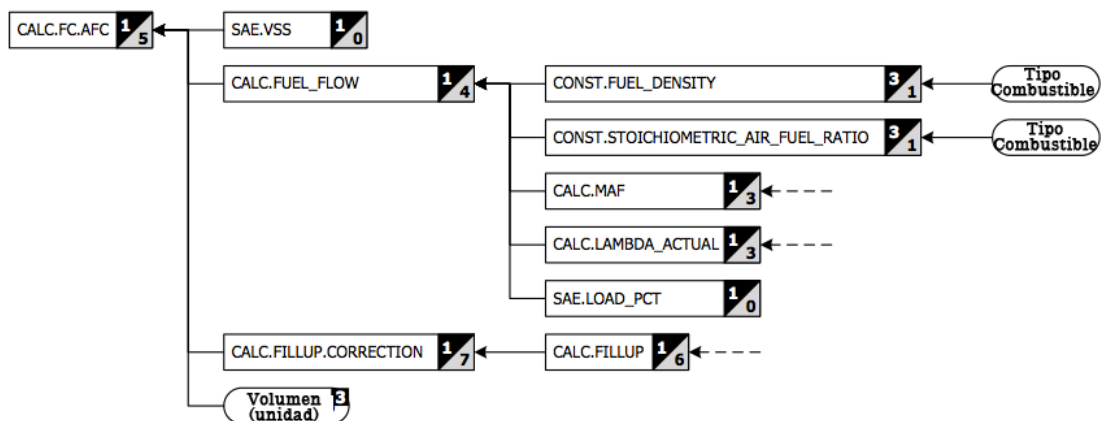


Fig. 4.13. Diagrama SDK CALC.FC.AFC

$AFC = \text{fuel}_t / d_t$, donde fuel_t es el total de combustible consumido durante un período de tiempo, y d_t es la distancia total viajada durante ese período de tiempo.

- CALC.FC.AFC.{01..17}

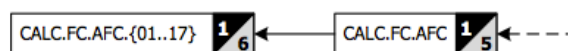


Fig. 4.14. Diagrama SDK CALC.FC.AFC.{01..17}

- CALC.FC.AVERAGE

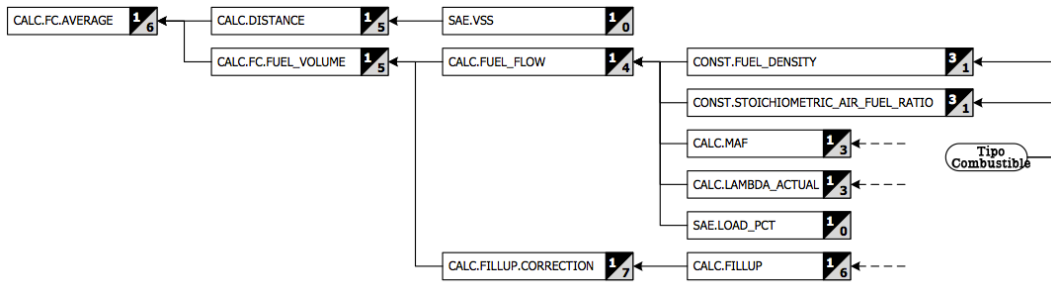


Fig. 4.15. Diagrama SDK CALC.FC.AVERAGE

- CALC.FC.FUEL_VOLUME

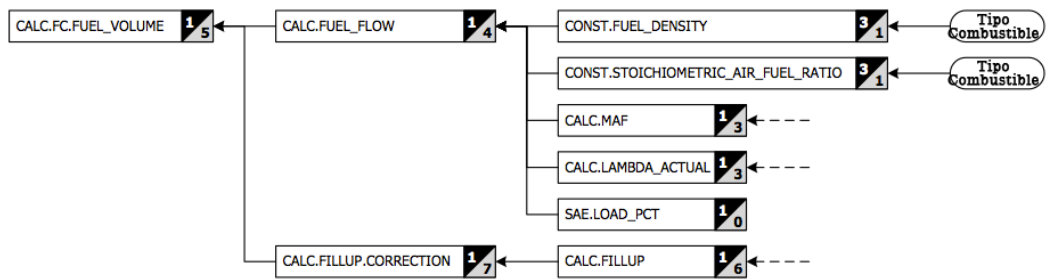


Fig. 4.16. Diagrama SDK CALC.FC.FUEL_VOLUME

- CALC.FC.IFC

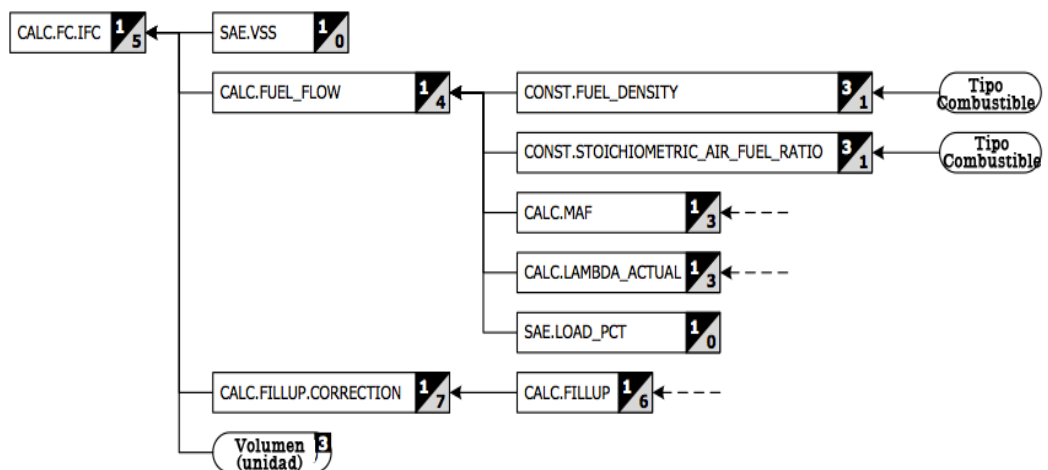


Fig. 4.17. Diagrama SDK CALC.FC.IFC

Consumo de combustible instantáneo, $IFC = fuel_flow / vehicle_speed$

- CALC.FC.IFC_AVG

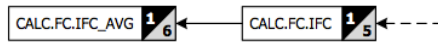


Fig. 4.18. Diagrama SDK CALC.FC.IFC_AVG

- CALC.FILLUP

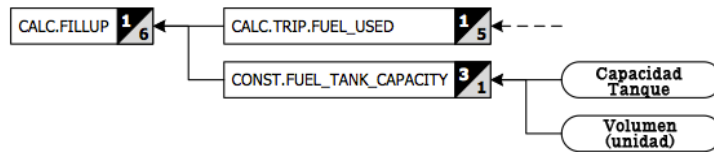


Fig. 4.19. Diagrama SDK CALC.FILLUP

Este PID maneja el proceso de calibración del consumo de combustible, el mismo que es ejecutado después de cada llenado del tanque de combustible.

- CALC.FILLUP.CORRECTION

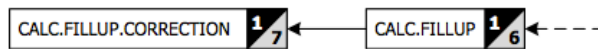


Fig. 4.20. Diagrama SDK CALC.FILLUP.CORRECTION

- CALC.FILLUP.FUEL_PRICE

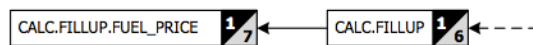


Fig. 4.21. Diagrama SDK CALC.FILLUP.FUEL_PRICE

- CALC.FILLUP.FUEL_USED

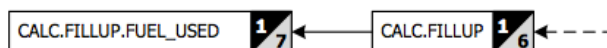


Fig. 4.22. Diagrama SDK CALC.FILLUP

- CALC.FRP

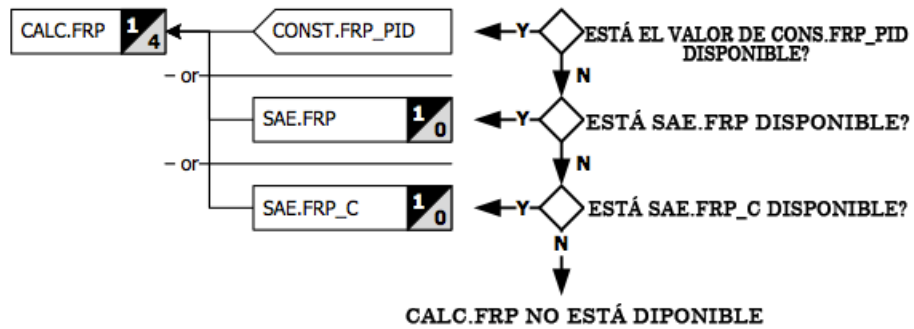


Fig. 4.23. Diagrama SDK CALC.FRP

Existen 2 PIDs FRP principales, uno correspondiente a la presión de combustible normal y otro para alta presión de combustible usada en aplicaciones de inyección de combustible directo. El PID mostrado corresponde al disponible en el vehículo.

- CALC.FUEL_CONSUMPTION

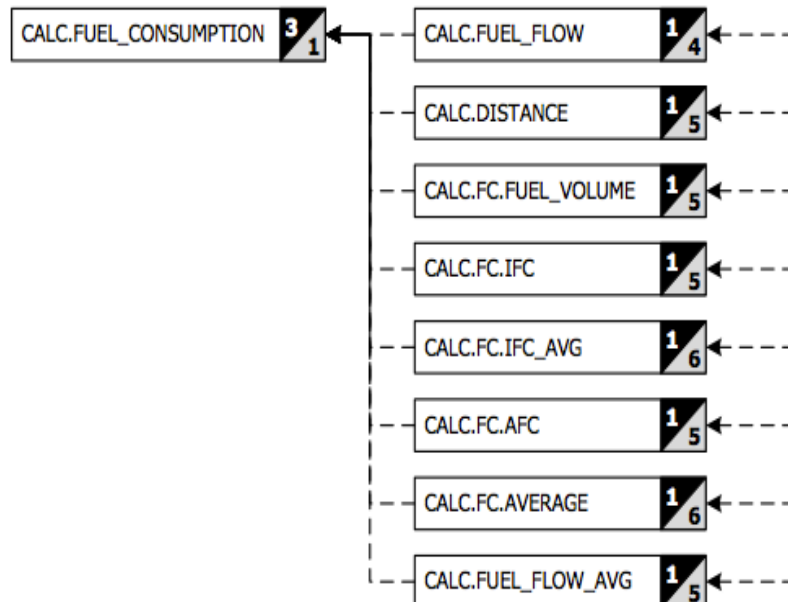


Fig. 4.24. Diagrama SDK CALC.FUEL_CONSUMPTION

- CALC.FUEL_FLOW

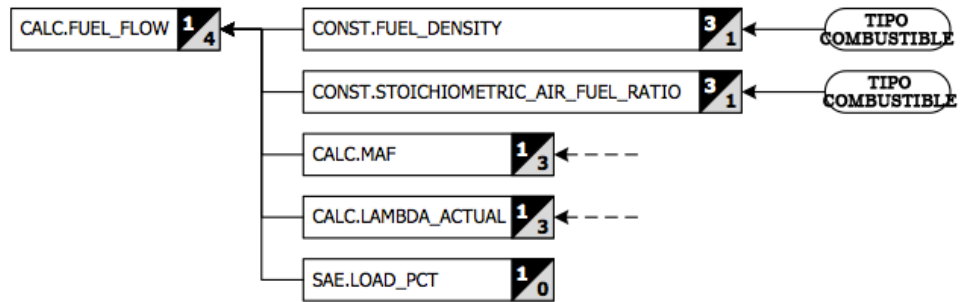


Fig. 4.25. Diagrama SDK CALC.FUEL_FLOW

Fuel_flow = mass_air_flow / (AFR_{actual} * fuel_density) donde AFR_{actual} = lambda * AFR_{stoich}. El valor SAE.LOAD_PCT es usado solamente cuando el tipo de combustible es Diesel para pre-ajustar el valor del MAF.

- CALC.FUEL_FLOW_AVG

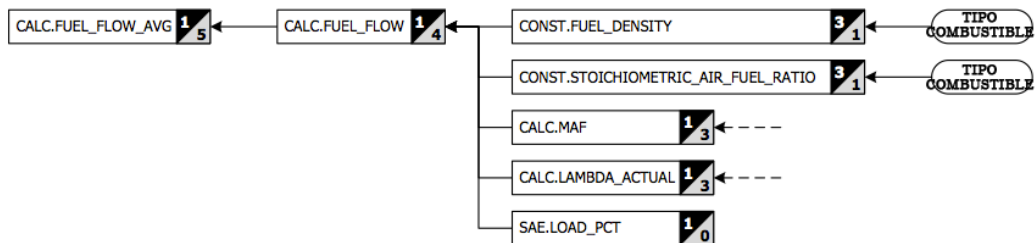


Fig. 4.26. Diagrama SDK CALC.ENGINE_POWER

- CALC.FUEL_LEVEL

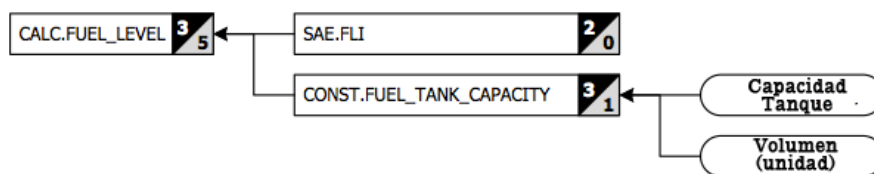


Fig. 4.27. Diagrama SDK CALC.FUEL_LEVEL

Fuel_level = fuel_tank_capacity * FLI / 100

- CALC.GEAR

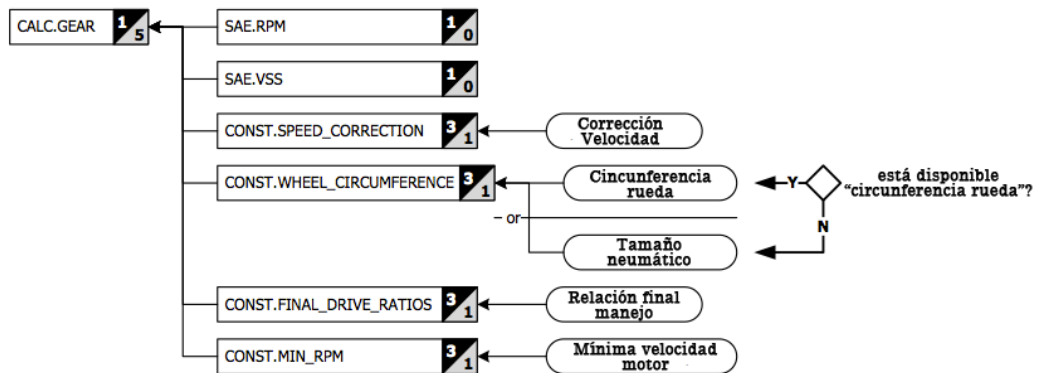


Fig. 4.28. Diagrama SDK CALC.GEAR

- CALC.GEAR_CURRENT

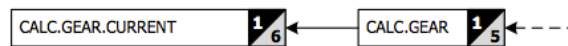


Fig. 4.29. Diagrama SDK CALC.GEAR.CURRENT

- CALC.GEAR.NUM_GEAR



Fig. 4.30. Diagrama SDK CALC.GEAR.NUM_GEAR

- CALC.GEAR.SHIFT_INDICATOR



Fig. 4.31. Diagrama SDK CALC.SHIFT_INDICATOR

Valores negativos indican que un cambio de marcha hacia inferior es sugerido, mientras que valores positivos indican una sugerencia a cambio de marcha superior.

- CALC.GEAR.SUGGESTED



Fig. 4.32. Diagrama SDK CALC.GEAR.SUGGESTED

- CALC.IS_NEW_DAY

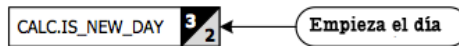


Fig. 4.33. Diagrama SDK CALC.IS_NEW_DAY

- CALC.LAMBDA_ACTUAL

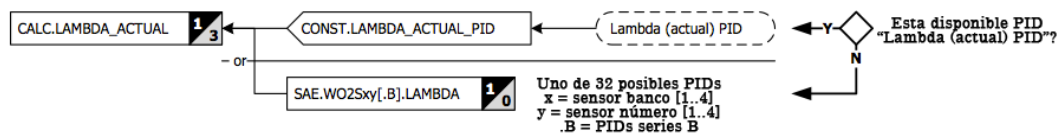


Fig. 4.34. Diagrama SDK CALC.LAMBDA_ACTUAL

- CALC.LAMBDA_COMMANDED



Fig. 4.35. Diagrama SDK CALC.LAMBDA_COMMANDED

- CALC.MAF

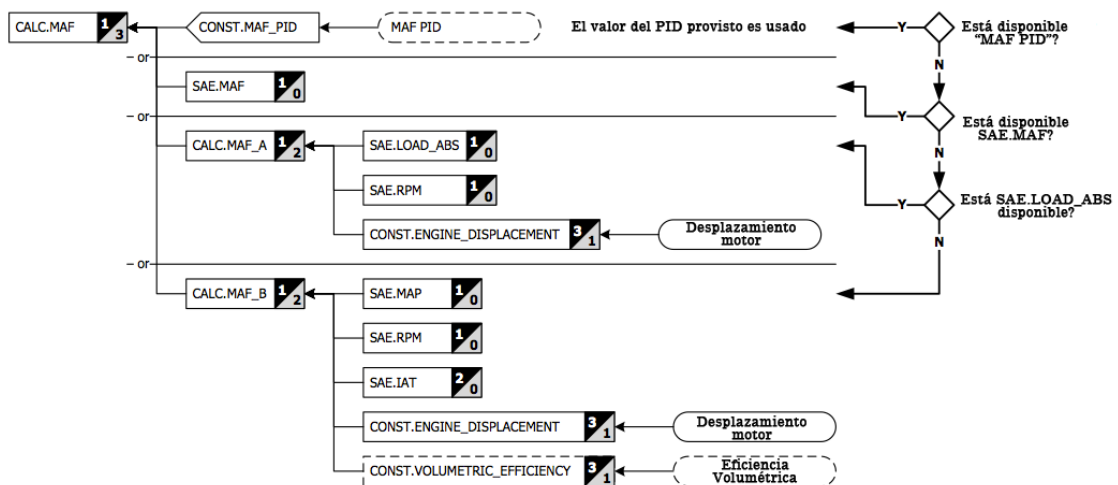


Fig. 4.36. Diagrama SDK CALC.MAF

- CALC.MAF_A

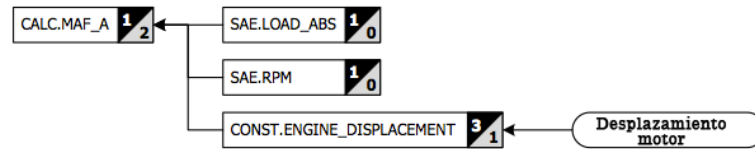


Fig. 4.37. Diagrama SDK CALC.MAF_A

OBD especifica que

$$\text{LOAD_ABS} = [\text{masa aire (g / admisión)}] / [1.184 \text{ (g / litro)} * \text{desplazamiento del cilindro (litros / admisión)}]$$

Por lo tanto, MAF puede ser calculado así:

$$\text{MAF [g/s]} = 1.184 \text{ [g/l]} * \text{desplazamiento [l/admisión]} * \text{load_abs} / 100 * \text{engine_speed [r/min]} / 2 \text{ [r/admisión]} / 60 \text{ [seg/min]}$$

- CALC.MAF_B

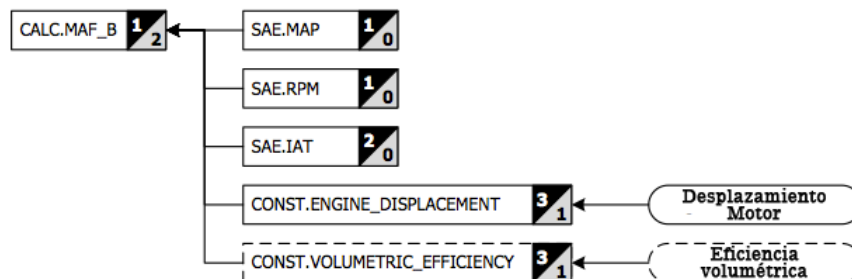


Fig. 4.38. Diagrama SDK CALC.MAF_B

Este método de cálculo del MAF está basado en la ley del Gas Ideal.

$$\text{MAF [g/s]} = (\text{MAP/IAT}) * (\text{M/R}) * (\text{RPM}/60) * (\text{ED}/2) * \text{VE, donde}$$

MAP [kPa] es el valor del MAP.

IAT [K] es la temperatura del aire en la admisión.

M [g/mol] es la masa molecular del aire.

R [J/(K*mol)] es la constante del gas para el aire.

RPM [r/min] es la velocidad del motor.

ED [l] es el desplazamiento en el motor.

VE es el valor de la eficiencia volumétrica, que cuando no es provista por la ECU, es usado el valor de 75% por defecto.

- CALC.MAP_A

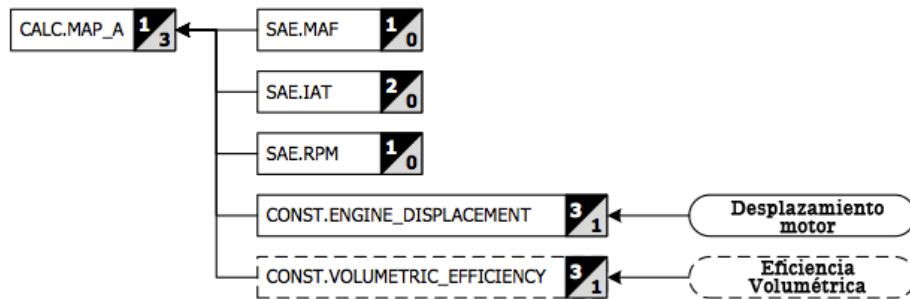


Fig. 4.39. Diagrama SDK CALC.MAP_A

Este método de cálculo del MAP está basado en la ley del Gas Ideal.

$$\text{MAP [kPa]} = (\text{MAF} \cdot \text{IAT}) / ((M/R) \cdot (\text{RPM}/60) \cdot (\text{ED}/2) \cdot \text{VE}), \text{ donde}$$

MAF [g/s] es el valor del MAF.

IAT [K] es la temperatura del aire en la admisión.

M [g/mol] es la masa molecular del aire.

R [J/(K*mol)] es la constante del gas para el aire.

RPM [r/min] es la velocidad del motor.

ED [l] es el desplazamiento en el motor.

VE es el valor de la eficiencia volumétrica, que cuando no es provista por la ECU, es usado el valor de 75% por defecto.

- CALC.MAP_B

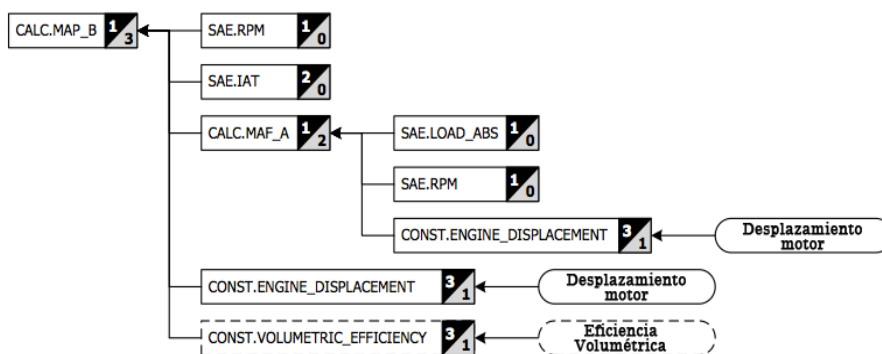


Fig. 4.40. Diagrama SDK CALC.MAP_B

Este método de cálculo del MAP está basado en la ley del Gas Ideal.

$$\text{MAP [kPa]} = (\text{MAF} * \text{IAT}) / ((\text{M/R}) * (\text{RPM}/60) * (\text{ED}/2) * \text{VE}), \text{ donde}$$

MAF [g/s] es el valor del MAF.

IAT [K] es la temperatura del aire en la admisión.

M [g/mol] es la masa molecular del aire.

R [J/(K*mol)] es la constante del gas para el aire.

RPM [r/min] es la velocidad del motor.

ED [l] es el desplazamiento en el motor.

VE es el valor de la eficiencia volumétrica, que cuando no es provista por la ECU, es usado el valor de 75% por defecto.

- CALC.MAP

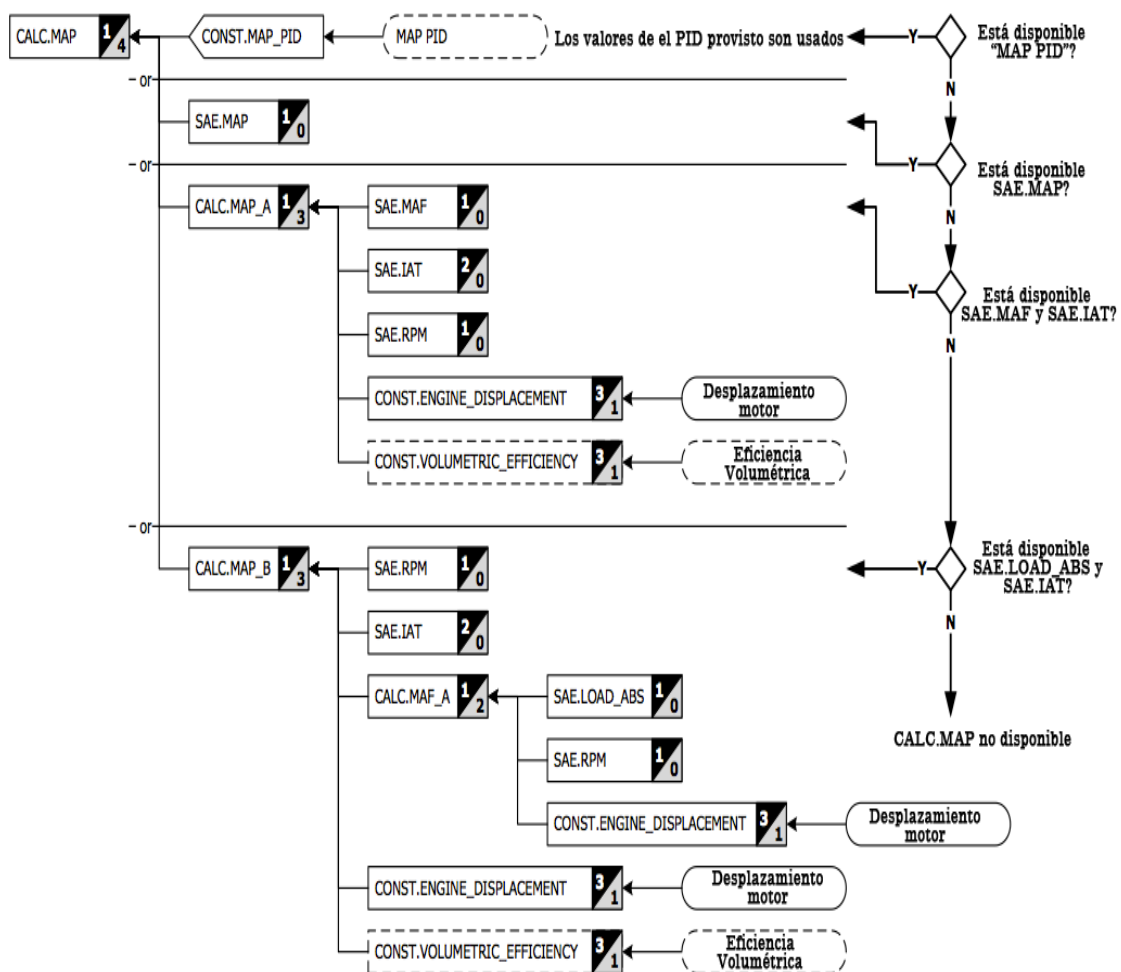


Fig. 4.41. Diagrama SDK CALC.MAP

- CALC.TRIP

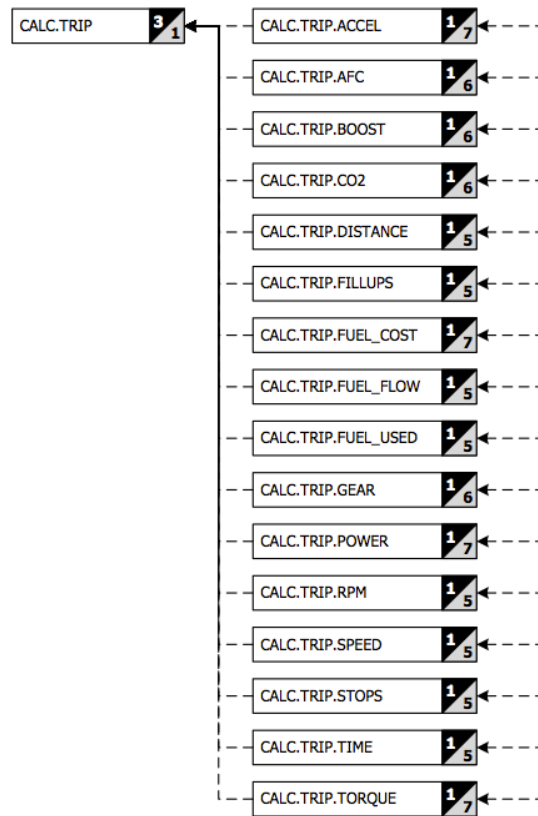


Fig. 4.42. Diagrama SDK CALC.TRIP

- CALC.TRIP.ACCEL

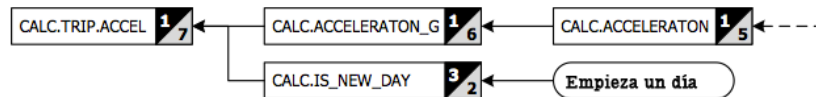


Fig. 4.43. Diagrama SDK CALC.TRIP.ACCEL

- CALC.TRIP.ACCEL.BRAKING.MAX.{ABTPF}

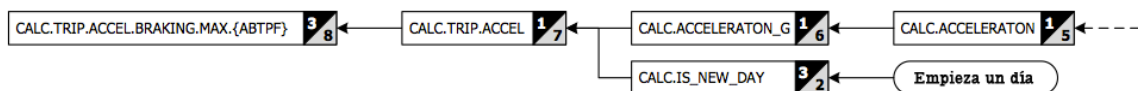


Fig. 4.44. Diagrama SDK CALC.TRIP.ACCEL.BRAKING.MAX{ABTPF}

- CALC.TRIP.ACCEL.FORWARD.MAX.{ABTPF}

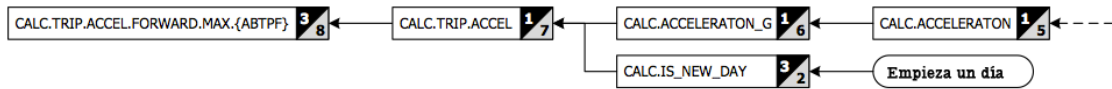


Fig. 4.45. Diagrama SDK CALC.TRIP.ACCEL.FORWARD.MAX.{ABTPF}

- CALC.TRIP.AFC

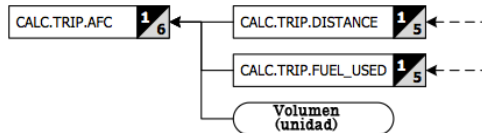


Fig. 4.46. Diagrama SDK CALC.TRIP.AFC

- CALC.TRIP.AFC.{ABTPF}

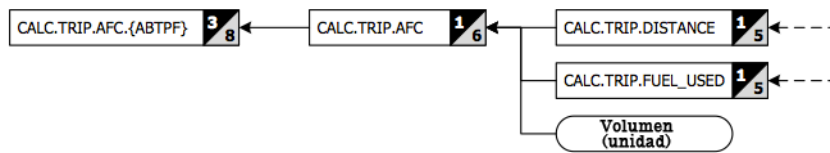


Fig. 4.47. Diagrama SDK CALC.TRIP.AFC.{ABTPF}

- CALC.TRIP.BOOST

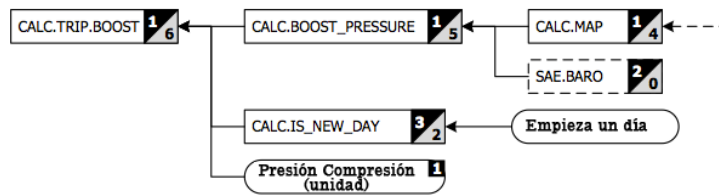


Fig. 4.48. Diagrama SDK CALC.TRIP.BOOST

- CALC.TRIP.BOOST.AVG.{ABTPF}

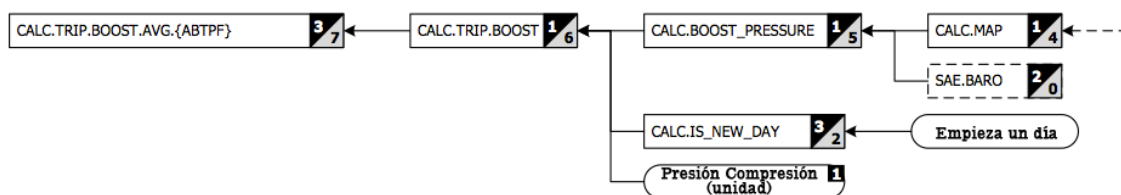


Fig. 4.49. Diagrama SDK CALC.TRIP.BOOST.AVG.{ABTPF}

- CALC.TRIP.BOOST.MAX_BOOST.{ABTPF}

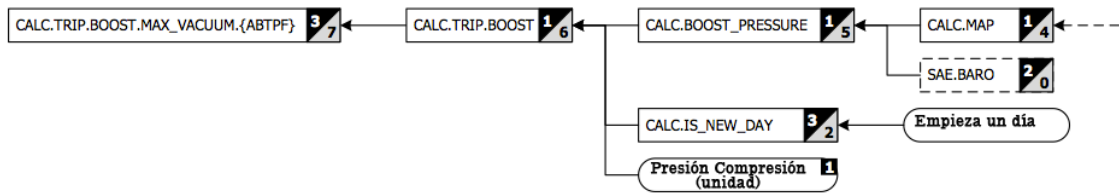


Fig. 4.50. Diagrama SDK CALC.TRIP.BOOST.MAX_BOOST.{ABTPF}

- CALC.TRIP.CO2

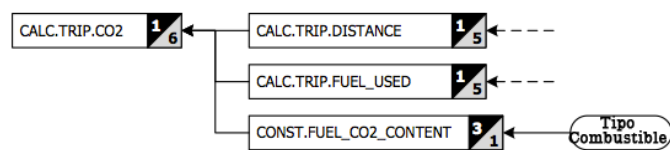


Fig. 4.51. Diagrama SDK CALC.TRIP.CO2

- CALC.TRIP.CO2.RATE.{ABTPF}

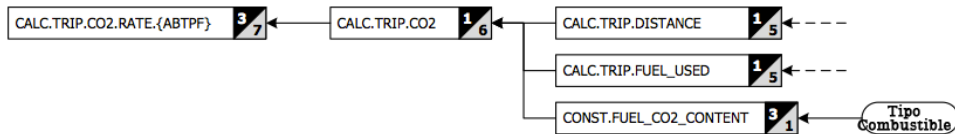


Fig. 4.52. Diagrama SDK CALC.TRIP.CO2.RATE.{ABTPF}

- CALC.TRIP.CO2.TOTAL.{ABTPF}

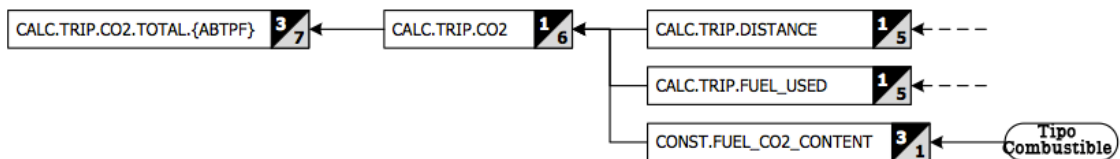


Fig. 4.53. Diagrama SDK CALC.TRIP.CO2.TOTAL.{ABTPF}

- CALC.TRIP.DISTANCE

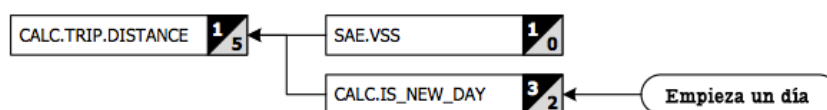


Fig. 4.54. Diagrama SDK CALC.TRIP.DISTANCE

- CALC.TRIP.DISTANCE.{ABTPF}



Fig. 4.55. Diagrama SDK CALC.TRIP.DISTANCE.{ABTPF}

- CALC.TRIP.FILLUPS

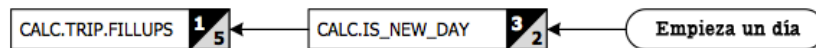


Fig. 4.56. Diagrama SDK CALC.TRIP.FILLUPS

- CALC.TRIP.FILLUPS.{ABTPF}

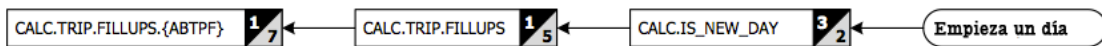


Fig. 4.57. Diagrama SDK CALC.TRIP.FILLUPS.{ABTPF}

- CALC.TRIP.FUEL_COST

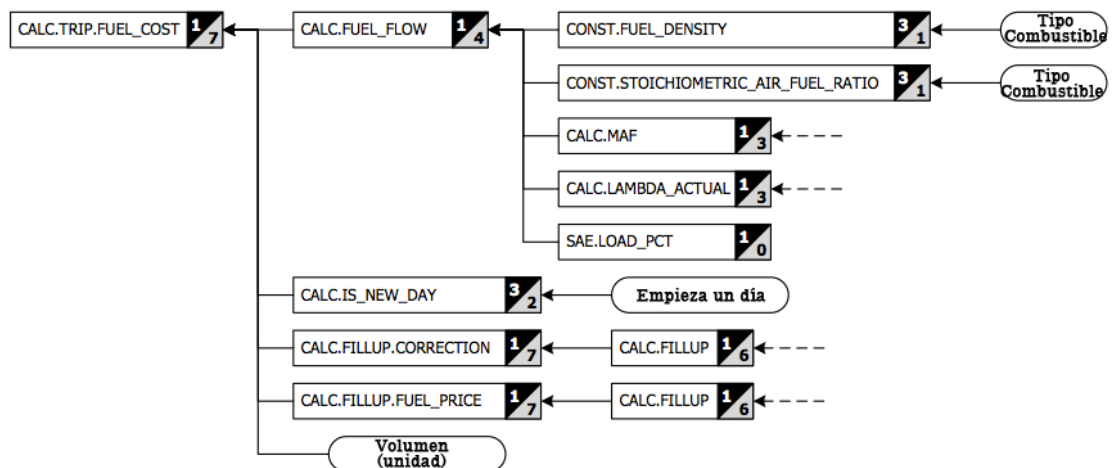


Fig. 4.58. Diagrama SDK CALC.TRIP.FUEL_COST

- CALC.TRIP.FUEL_COST.{ABTPF}

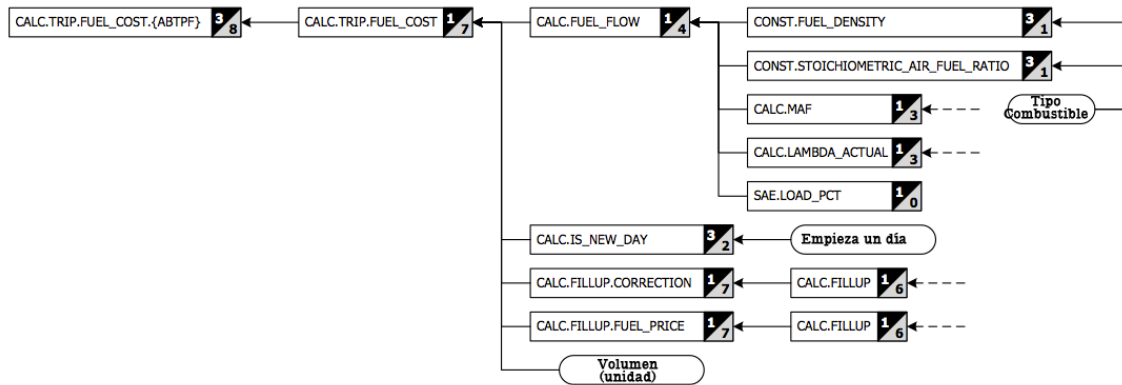


Fig. 4.59. Diagrama SDK CALC.TRIP.FUEL_COST.{ABTPF}

- CALC.TRIP.FUEL_FLOW

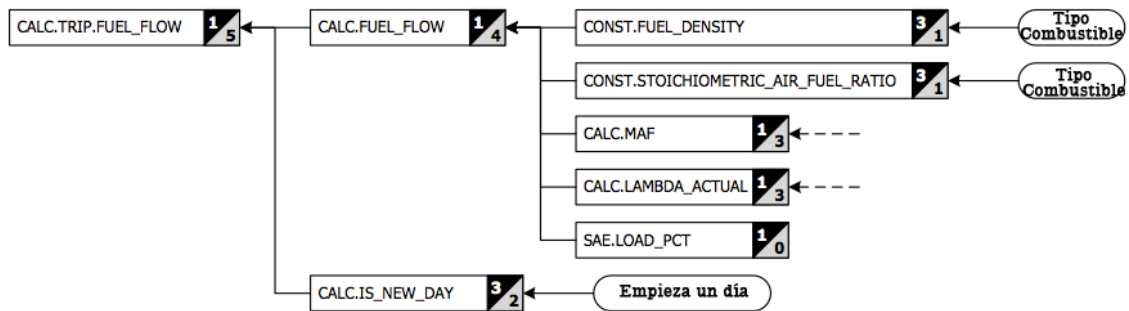


Fig. 4.60. Diagrama SDK CALC.TRIP.FUEL_FLOW

- CALC.TRIP.FUEL_FLOW.AVG.{ABTPF}

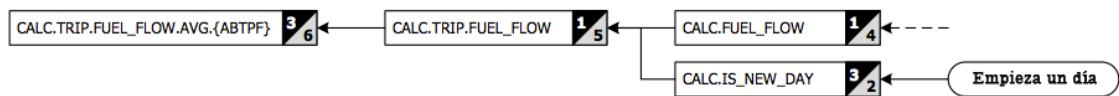


Fig. 4.61. Diagrama SDK CALC.TRIP.FUEL_FLOW.AVG.{ABTPF}

- CALC.TRIP.FUEL_FLOW.MAX.{ABTPF}

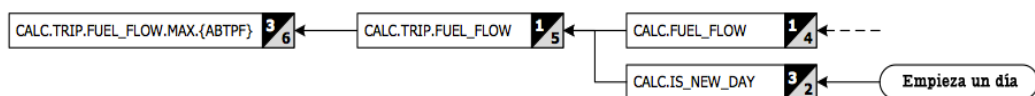


Fig. 4.62. Diagrama SDK CALC.TRIP.FUEL_FLOW.MAX.{ABTPF}

- CALC.TRIP.FUEL_USED

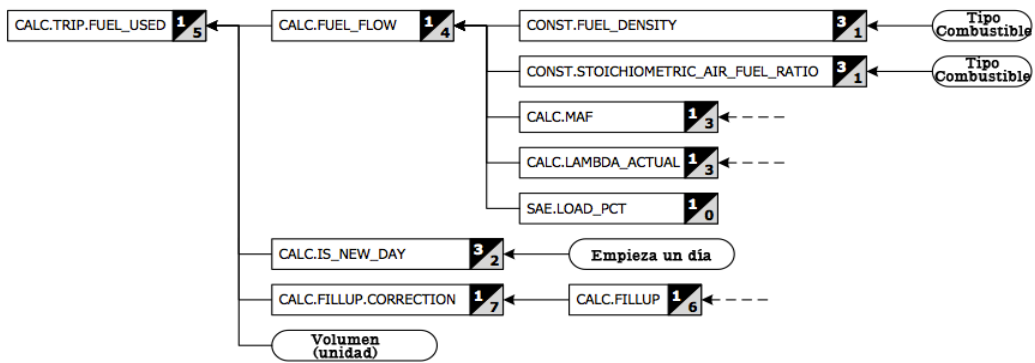


Fig. 4.63. Diagrama SDK CALC.TRIP.FUEL_USED

- CALC.TRIP.FUEL_USED.{ABTPF}

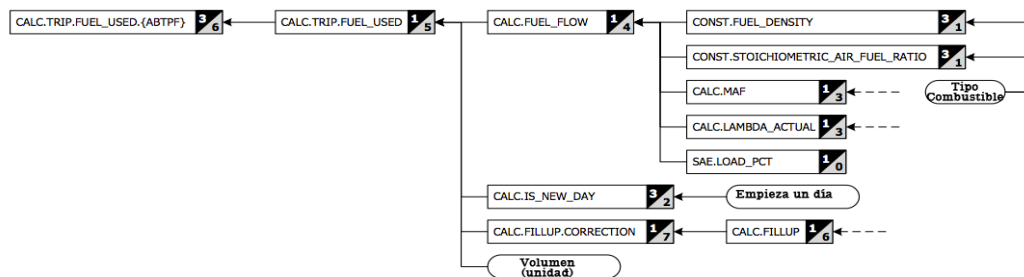


Fig. 4.64. Diagrama SDK CALC.TRIP.FUEL_USED.{ABTPF}

- CALC.TRIP.GEAR

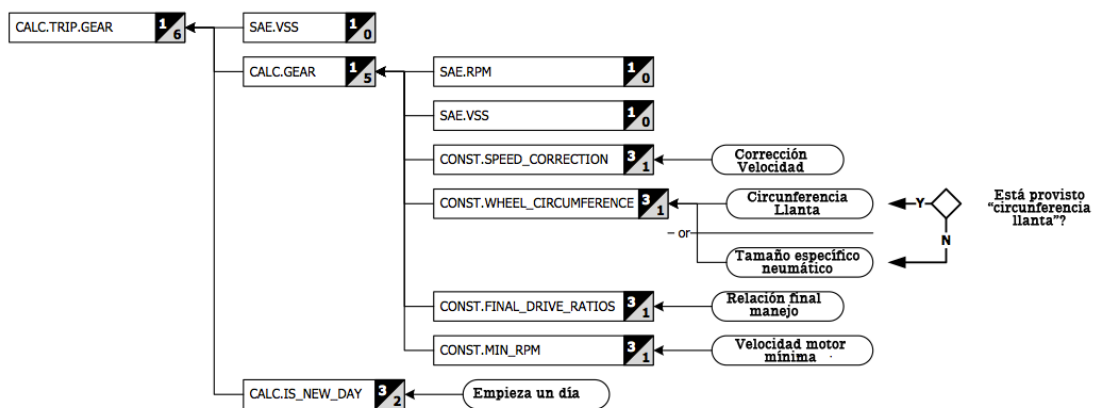


Fig. 4.65. Diagrama SDK CALC.TRIP.GEAR

- CALC.TRIP.GEAR.{1..6}.DISTANCE_PCT.{ABTPF}



Fig. 4.66. Diagrama SDK

CALC.TRIP.GEAR.{1..6}.DISTANCE_PCT.{ABTPF}

- CALC.TRIP.GEAR.{1..6}.TIME_PCT.{ABTPF}



Fig. 4.67. Diagrama SDK CALC.TRIP.GEAR.{1..6}.TIME_PCT.{ABTPF}

- CALC.TRIP.GEAR.N.DISTANCE_PCT.{ABTPF}



Fig. 4.68. Diagrama SDK CALC.TRIP.GEAR.N.DISTANCE_PCT.{ABTPF}

- CALC.TRIP.GEAR.N.TIME_PCT.{ABTPF}



Fig. 4.69. Diagrama SDK CALC.TRIP.GEAR.N.TIME_PCT.{ABTPF}

- CALC.TRIP.GEAR.WG.DISTANCE_PCT.{ABTPF}



Fig. 4.70. Diagrama SDK CALC.TRIP.GEAR.N.TIME_PCT.{ABTPF}

- CALC.TRIP.GEAR.WG.TIME_PCT.{ABTPF}



Fig. 4.71. Diagrama SDK CALC.TRIP.GEAR.WG.TIME_PCT.{ABTPF}

- CALC.TRIP.POWER

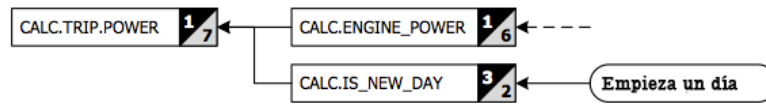


Fig. 4.72. Diagrama SDK CALC.TRIP.POWER

- CALC.TRIP.POWER.MAX.{ABTPF}

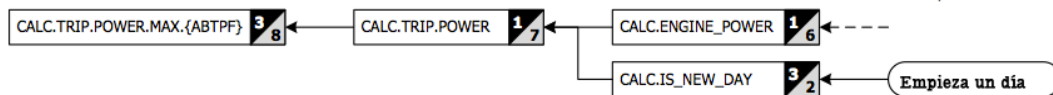


Fig. 4.73. Diagrama SDK CALC.TRIP.POWER.MAX.{ABTPF}

- CALC.TRIP.RPM

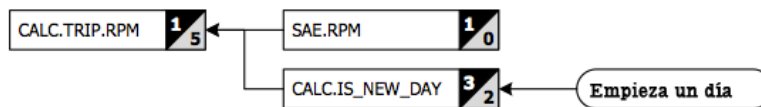


Fig. 4.74. Diagrama SDK CALC.TRIP.RPM

- CALC.TRIP.RPM.MAX.{ABTPF}

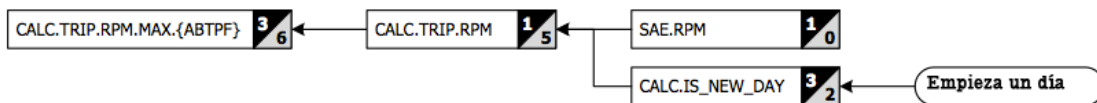


Fig. 4.75. Diagrama SDK CALC.RPM.MAX.{ABTPF}

- CALC.TRIP.SPEED

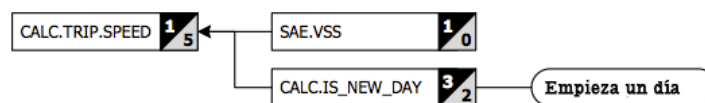


Fig. 4.76. Diagrama SDK CALC.TRIP.SPEED

- CALC.TRIP.SPEED.AVG_NI.{ABTPF}

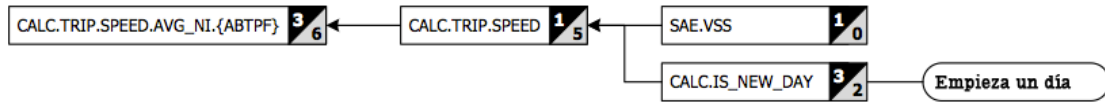


Fig. 4.77. Diagrama SDK CALC.TRIP.SPEED.AVG_NI.{ABTPF}

- CALC.TRIP.SPEED.AVG_WI.{ABTPF}

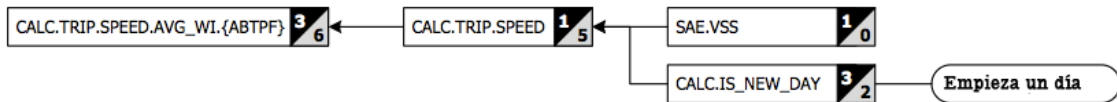


Fig. 4.78. Diagrama SDK CALC.TRIP.SPEED.AVG_WI.{ABTPF}

- CALC.TRIP.SPEED.MAX.{ABTPF}

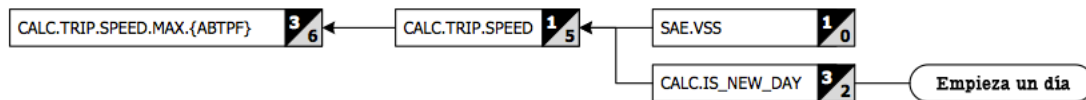


Fig. 4.79. Diagrama SDK CALC.TRIP.SPEED.MAX.{ABTPF}

- CALC.TRIP.STOPS

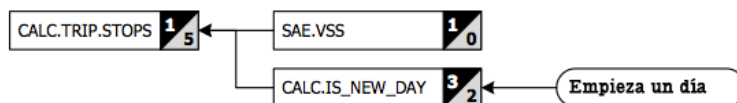


Fig. 4.80. Diagrama SDK CALC.TRIP.STOPS

- CALC.TRIP.STOPS.{ABTPF}

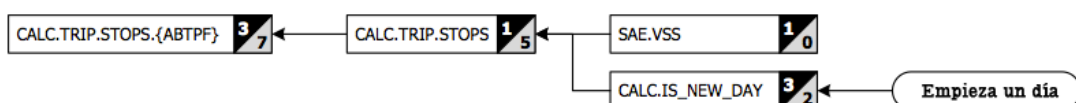


Fig. 4.81. Diagrama SDK CALC.TRIP.STOPS.{ABTPF}

- CALC.TRIP.TIME

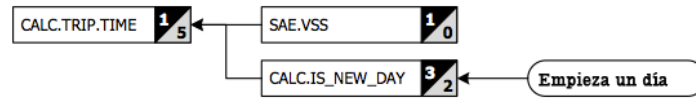


Fig. 4.82. Diagrama SDK CALC.TRIP.TIME

- CALC.TRIP.TIME.DRIVE.{ABTPF}

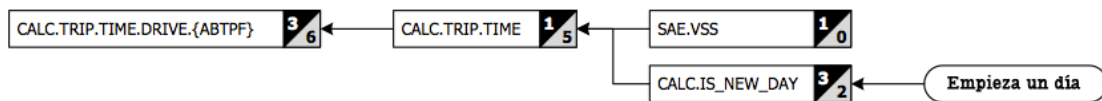


Fig. 4.83. Diagrama SDK CALC.TRIP.TIME.DRIVE.{ABTPF}

- CALC.TRIP.TIME.ELAPSED.{ABTPF}

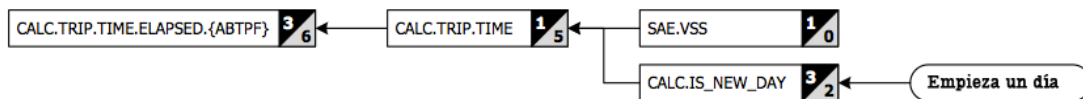


Fig. 4.84. Diagrama SDK CALC.TRIP.TIME.ELAPSED.{ABTPF}

- CALC.TRIP.TIME.IDLE.{ABTPF}

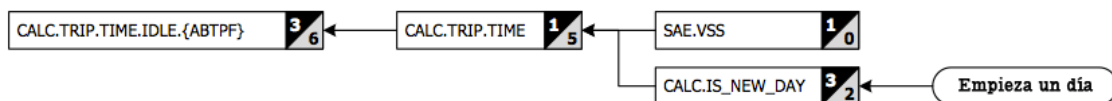


Fig. 4.85. Diagrama SDK CALC.TRIP.TIME.IDLE.{ABTPF}

- CALC.TRIP.TIME.RUN.{ABTPF}

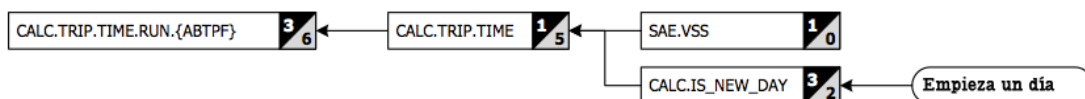


Fig. 4.86. Diagrama SDK CALC.TRIP.TIME.RUN.{ABTPF}

- CALC.TRIP.TIME.START.{ABTPF}

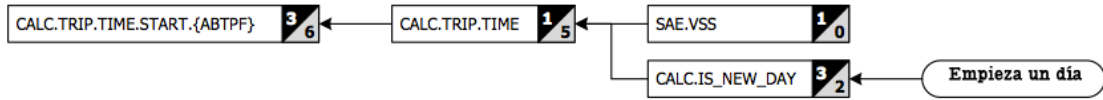


Fig. 4.87. Diagrama SDK CALC.TRIP.TIME.START.{ABTPF}

- CALC.TRIP.TORQUE

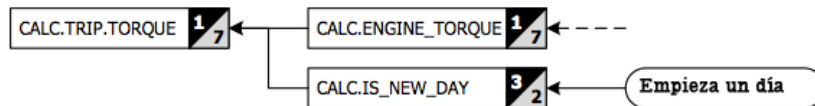


Fig. 4.88. Diagrama SDK CALC.TRIP.TORQUE

- CALC.TRIP.TORQUE.MAX.{ABTPF}

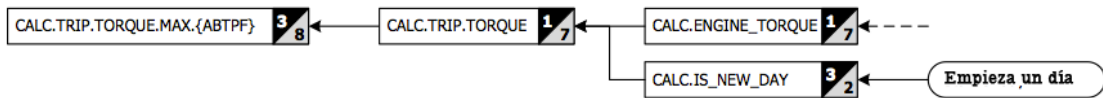


Fig. 4.89. Diagrama SDK CALC.TRIP.TORQUE.MAX.{ABTPF}

- CALC.TTE

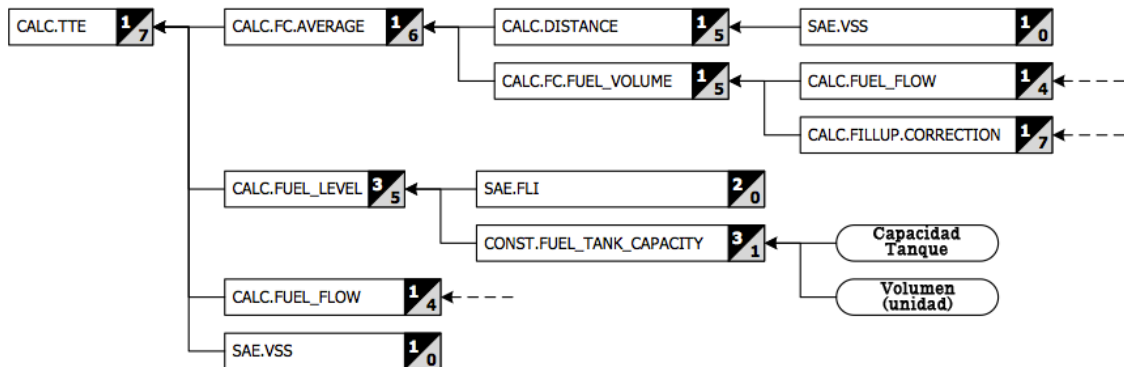


Fig. 4.90. Diagrama SDK CALC.TTE

- CALC.VACUUM

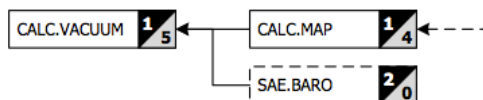


Fig. 4.91. Diagrama SDK CALC.VACUUM

- CALC.VSS_C

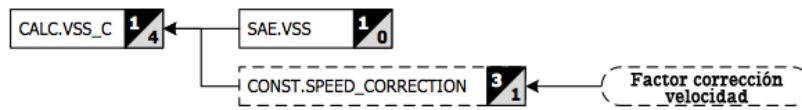


Fig. 4.92. Diagrama SDK CALC.VSS_C

- CALC.VSS_D

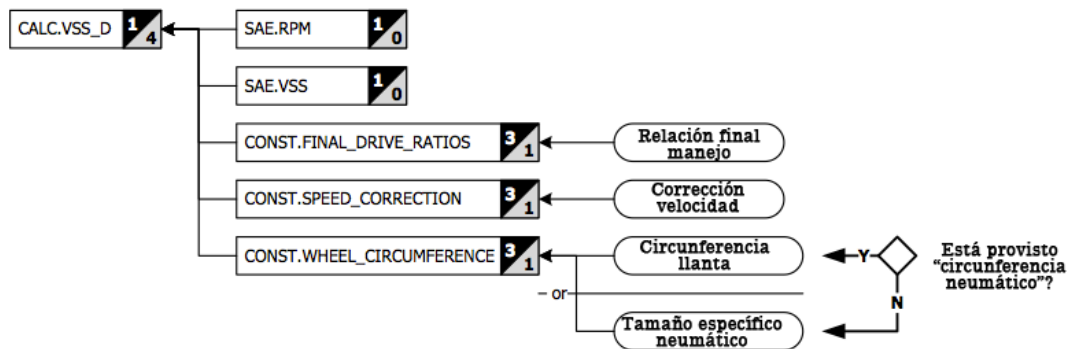


Fig. 4.93. Diagrama SDK CALC.VSS_D

Adjunto al manejo de las operaciones lógicas CALC. la OBDNet, en su estructura informática, maneja valores constantes provenientes de la información recibida de la ECU, constando principalmente:

- CONST.ADDITIONAL_WEIGHT

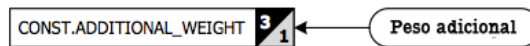


Fig. 4.94. Diagrama CONST.ADDITIONAL_WEIGHT

- CONST.CURB_WEIGHT



Fig. 4.95. Diagrama CONST.CURB_WEIGHT

- CONST.DRAG_COEFFICIENT

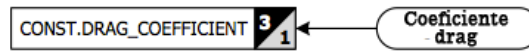


Fig. 4.96. Diagrama CONST.DRAG_COEFFICIENT

- CONST.ENGINE_DISPLACEMENT



Fig. 4.97. Diagrama CONST.ENGINE_DISPLACEMENT

- CONST.FINAL_DRIVE_RATIOS

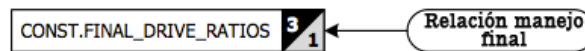


Fig. 4.98. Diagrama CONST.FINAL_DRIVE_RATIOS

- CONST.FRONTAL_AREA



Fig. 4.99. Diagrama CONST.FRONTAL_AREA

- CONST.FRP_PID

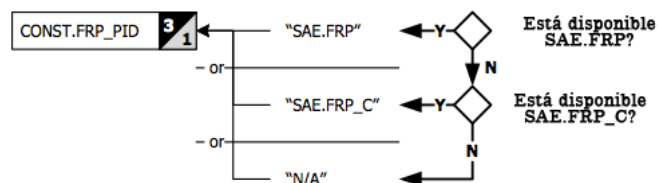


Fig. 4.100. Diagrama CONST.FRP_PID

- CONST.FUEL_CO2_CONTENT



Fig. 4.101. Diagrama CONST.FUEL_CO2_CONTENT

- CONST.FUEL_DENSITY



Fig. 4.102. Diagrama CONST.FUEL_DENSITY

- CONST.FUEL_TANK_CAPACITY

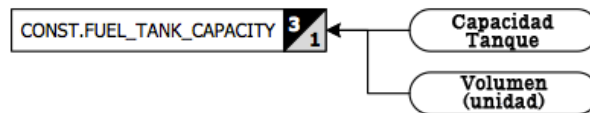


Fig. 4.103. Diagrama CONST.FUEL_TANK_CAPACITY

- CONST.LAMBDA_ACTUAL_PID

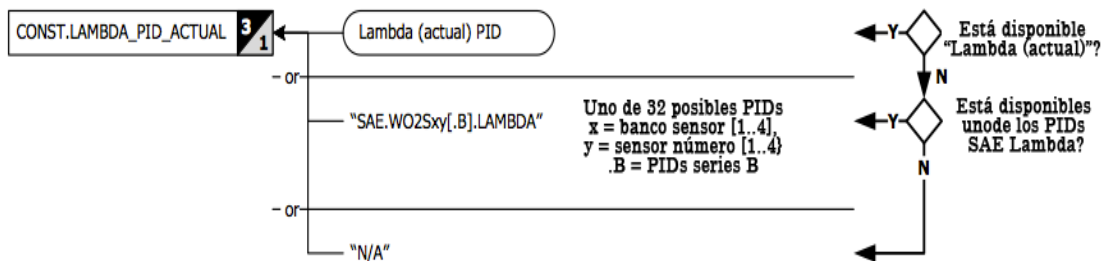


Fig. 4.104. Diagrama CONST.LAMBDA_ACTUAL_PID

- CONST.LAMBDA_CMDANDED_PID

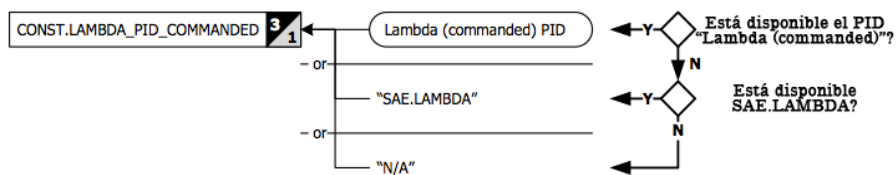


Fig. 4.105. Diagrama CONST.LAMBDA_PID_CMDANDED

- CONST.MAF_PID

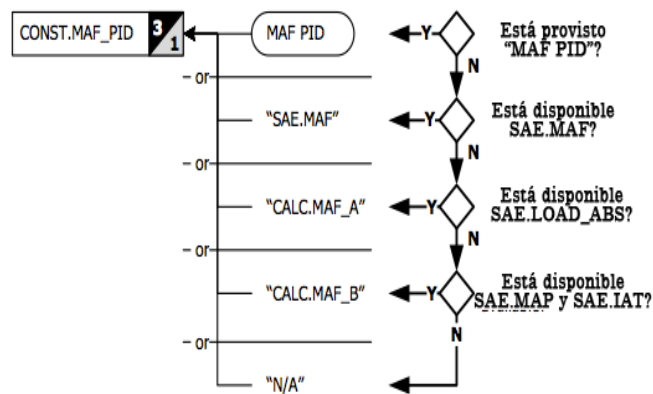


Fig. 4.106. Diagrama CONST.MAF_PID

- CONST.MAP_PID

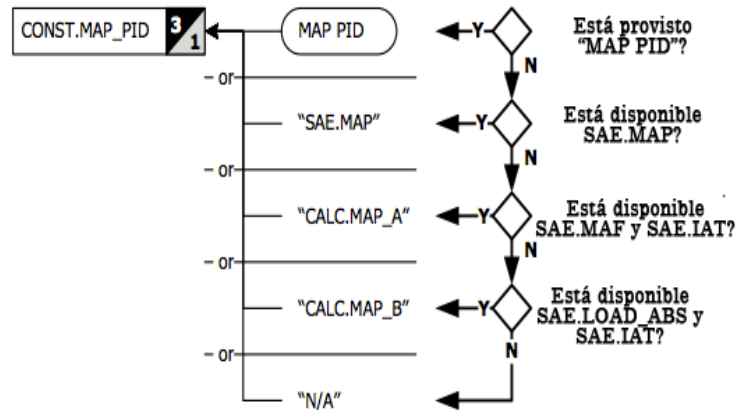


Fig. 4.107. Diagrama CONST.MAP_PID

- CONST.MAX_RPM



Fig. 4.108. Diagrama CONST.MAX_RPM

- CONST.MIN_RPM



Fig. 4.109. Diagrama CONST.MIN_RPM

- CONST.SHIFT_RPM



Fig. 4.110. Diagrama CONST.SHIFT_RPM

- CONST.SPEED_CORRECTION



Fig. 4.111. Diagrama CONST.SPEED_CORRECTION

- CONST.SPEED_ZONES

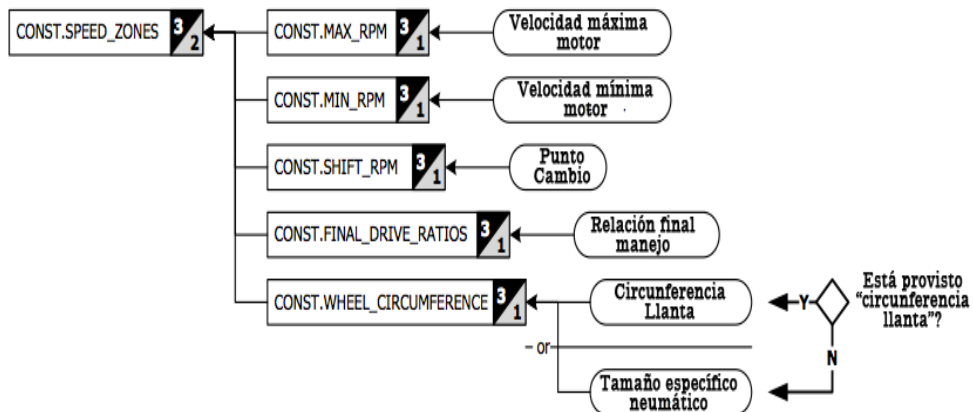


Fig. 4.112. Diagrama CONST.SPEED_ZONES

- CONST.SPEED_ZONES.{1..6}.MAX_SPEED

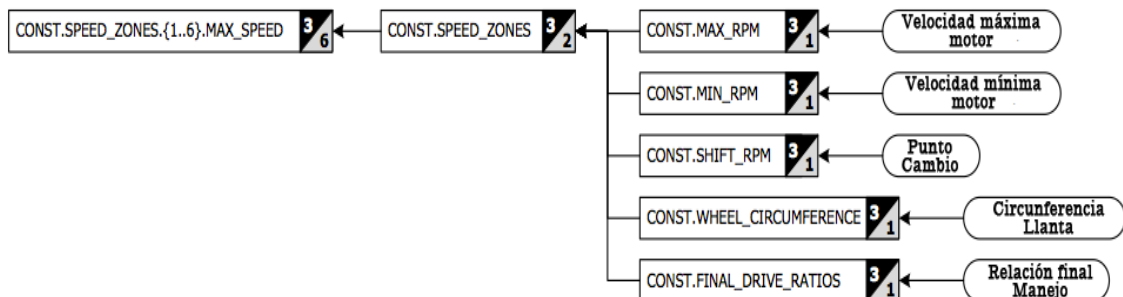


Fig. 4.113. Diagrama CONST.SPEED_ZONES.{1..6}.MAX_SPEED

- CONST.SPEED_ZONES.{1..6}.MIN_SPEED

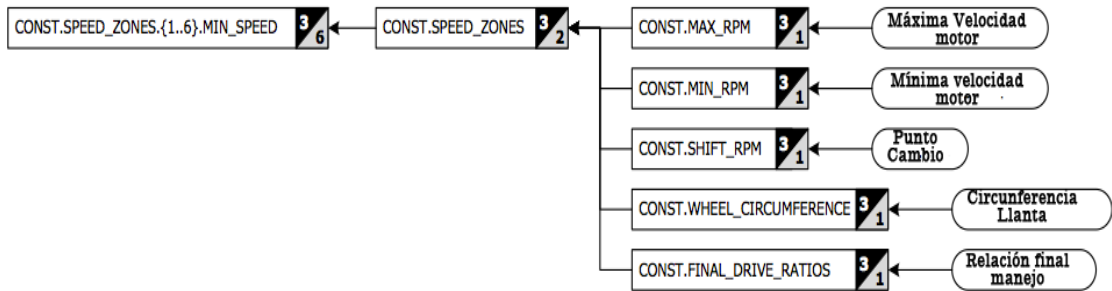


Fig. 4.114. Diagrama CONST.SPEED_ZONES.{1..6}.MIN_SPEED

- CONST.SPEED_ZONES.{1..6}.SHIFT_SPEED

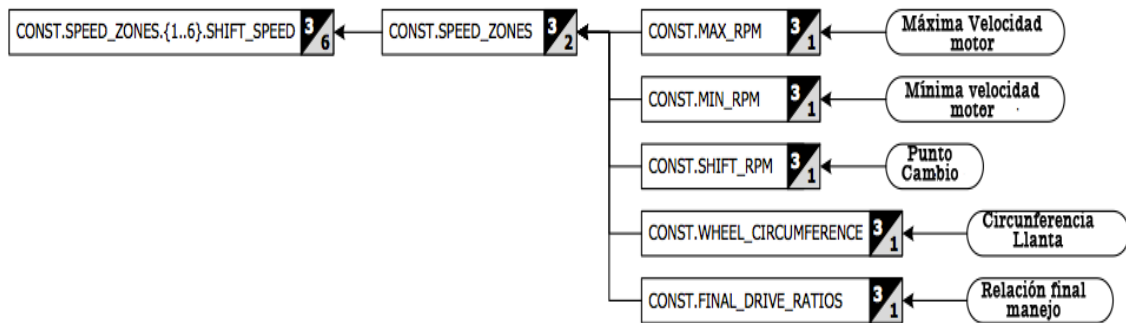


Fig. 4.115. Diagrama CONST.SPEED_ZONES.{1..6}.SHIFT_SPEED

- CONST.STOICHIOMETRIC_AIR_FUEL_RATIO

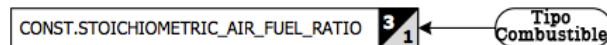


Fig. 4.116. Diagrama CONST.STOICHIOMETRIC_AIR_FUEL_RATIO

- CONST.TIRE_RESISTANCE



Fig. 4.117. Diagrama CONST.TIRE_RESISTANCE

- CONST.VOLUMETRIC_EFFICIENCY

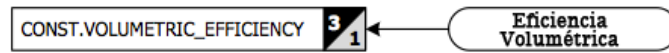


Fig. 4.118. Diagrama CONST.VOLUMETRIC_EFFICIENCY

- CONST.WHEEL_CIRCUMFERENCE

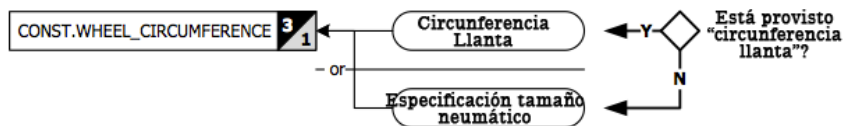


Fig. 4.119. Diagrama CONST.WHEEL_CIRCUMFERENCE

4.3 PERSONALIZACIÓN DE LOS SISTEMAS DE INTERFAZ GRÁFICA DE USUARIO.

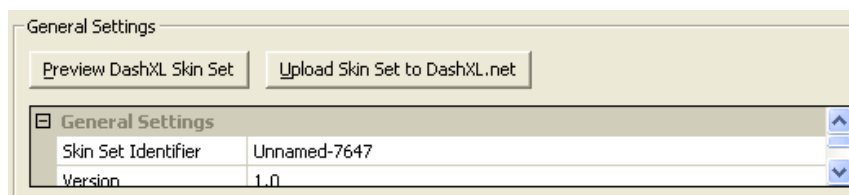
Con la finalidad de proveer al potencial usuario una interfaz amigable, en este apartado procedemos a combinar conocimientos de lenguaje Visual Basic con los estudiados en el anterior apartado correspondientes a lenguaje Objective-C.

La programación de objetos animados en nuestra plantilla, se la hace mediante software de combinación animada, en este caso por la especialización en Objective-C escogimos la Suite de SkinXL Editor, la misma que esta dividida en su estructura de trabajo en 3 secciones:

- General Settings
- Dashboard Screen Settings
- Dashboard Object Settings.

General Settings

Esta sección nos ayuda a dar la información básica de contenidos de nuestro proyecto, como es el caso de el nombre del proyecto, la versión del proyecto en caso de ir mejorándolo periódicamente, descripción textual de los componentes, y el aspecto que deseemos que tenga nuestra interfaz gráfica, opción relevante en el caso de utilizar diferentes iDevices (Ipad, Iphone), ya que tienen distinta resolución entre ellos.

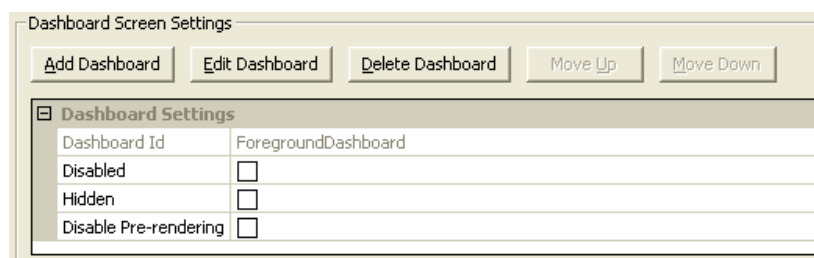


Fuente: SkinXL Editor.

Fig. 4.120. General Settings SkinXL Editor

Dashboard Screen Settings

Esta apartado nos permite seleccionar las funciones que se van a automatizar en la presentación de cada dashboard, su actividad, adición de más plantillas de trabajo, se eliminación, orden de las pantallas, así como cualidades de presentación activa pasiva, hidden o escondida a la vista del usuario, y la desactivación del pre-rendering, opción muy útil para disminuir el uso de memoria y de núcleos de proceso.

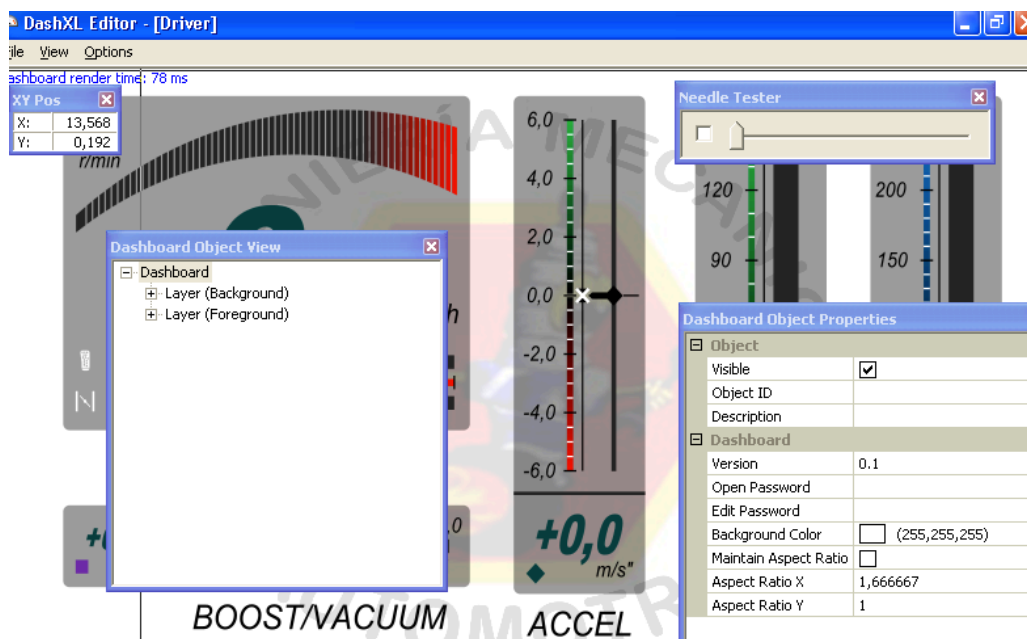


Fuente: SkinXL Editor

Fig. 4.121. Dashboard Screen Settings DashXL Editor

La parte primordial de los screen settings corresponde al editor de Dashboards (Edit Dashboard), el mismo que nos abre un complemento de software específico para edición de objetos y acción – reacción, llamado DashXL Editor.

DashXL Editor es un programador – compilador que simplifica y ayuda al usuario en la diagramación del proyecto en Objective-C ya que ,como lo hace Visual Basic, ahorra al usuario la creación del lenguaje puro en sí, centrándose únicamente en la utilización de las herramientas dispuestas.



Fuente: SkinXL Editor

Fig. 4.122. DashXL Editor

Sus componentes de manera parecida a un editor de coordenadas para software dedicado, se encuentran divididos en cuatro conjuntos con sus respectivas funciones así:

- **XY POS** nos permite ubicar el objeto en cierto punto exacto del plano, mediante el uso de coordenadas de localización en los ejes X y Y.

- **Dashboard Object View** despliega todos los componentes de cada medidor que creemos, así como nos permite utilizar nuevos objetos y agruparlos en medidores gauges digitales, GLCDs, y valores numéricos.
- **Dashboard Object Properties** contiene todas las características disponibles, de acuerdo a la versión de software, que podemos aplicar a nuestros objetos. Para nuestra personalización nos permite editar objetos, formas, líneas y bordes, colores, de acuerdo a los objetos que vayamos introduciendo en la plantilla.
- **Needle Tester** como comprobador de funcionamiento de los medidores, y los conceptos de diseño de objetos que se hayan aplicado.

Dashboard Object Settings

Los settings de objetos predispuestos en esta parte del software, se conjugan con las propiedades que el diagramador da a cada objeto en el Dashboard Object View del DashXL Editor. Una opción muy utilizada es la del enlazador de acciones, de gran ayuda en la conformación de botones activos dobles o triples de direccionamiento de skins unificados.

4.3.1 PERSONALIZACIÓN DEL MENÚ DE PRESENTACIÓN.

La interfaz gráfica de usuario deseada, debe organizar nuestro proyecto en varias plantillas de optimización con un fin principal, el orden y la presentación visual atractiva que se quiera dar.

En razón de esta premisa, el diseño de nuestro menú debe contener los siguientes apartados:

- Lector Principal (Lecturas de ayuda al conductor, copiloto y equipo de diagnóstico)
- Gauges GLCDs (Diagrámicas electrónicas de medición)
- Medición y Telemetría (Registro en tiempo real y tiempo parcial, total del análisis)
- Lectores Digitales (Lectores digitales)
- Fuerzas G (Medidor de fuerzas G)

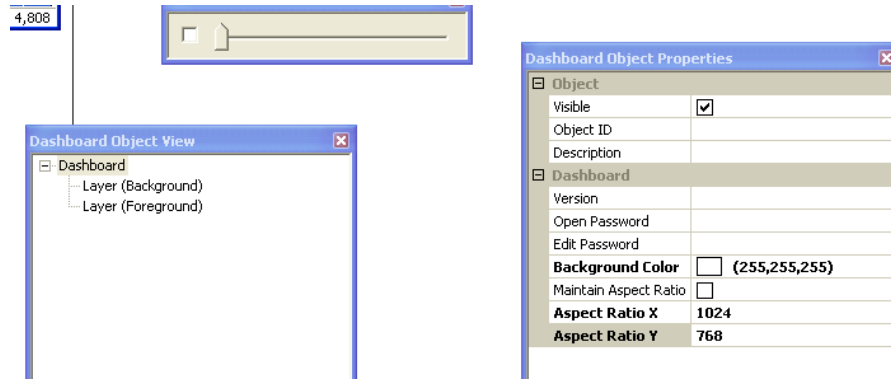
El proceso de personalización del diseño requerido para nuestros requerimientos se explica detalladamente:

PLANTILLA DE PARTIDA

- Se crea un nuevo archivo de DashXL Skin Editor.
(File -> New, o Ctrl + N).
- Adicionamos un nuevo Dashboard.
(Dashboard Screen Settings -> Add Dashboard).
- Seleccionamos el Dashboard creado y procedemos a su personalización.
(Dashboard Screen Settings -> Edit Dashboard)
- Se abre el complemento de software DashXL Editor, con dos grupos de objetos Layer (Background) [Objetos de Acción de fondo] y Layer (Foreground) [Objetos de Acción Principales] en el Dashboard Object View y un color predeterminado Gris (128,128,128) de fondo de plantilla en las propiedades.
- Cambiamos el color de fondo de plantilla a color Blanco (255, 255, 255)
(Dashboard Object Properties -> Background Color = (255, 255, 255)).

- Como se va a utilizar un iDevice Ipad, cambiamos la resolución de la pantalla a 1024x768.

(Dashboard Object Properties -> Aspect Ratio X = 1024, Aspect Ratio Y = 768).



Fuente: SkinXL Editor

Fig. 4.123. Plantilla de Partida

MENÚ DE SELECCIÓN

- Adicionamos un nuevo Dashboard, y lo llamamos Main Menu.
(Dashboard Screen Settings -> Add Dashboard -> Dashboard Settings -> Dashboard Id = MainMenu).
- A partir de las características de una plantilla de partida, adicionamos 6 objetos (5 botones) que vamos a utilizar para ingresar a cada sección de nuestro proyecto, y para regresar a nuestro menú de inicio (home), con los siguientes valores.

(Dashboard Object Settings -> Add Object)

VALORES PARA BUTTON1, BUTTON2, BUTTON3, BUTTON4, BUTTON5

ShowMenu ->

Object Id	BUTTON1	BUTTON2	BUTTON3	BUTTON4	BUTTON5
Object Type	Button	Button	Button	Button	Button
Object Action	SwitchScreen	SwitchScreen	SwitchScreen	SwitchScreen	SwitchScreen
Action Target	Lector Principal	Gauges GLCDs	Medicion Telemetry	Economia Poder Flujo de Aire	Fuerzas G
Transsition effect	Crossfade	Crossfade	Crossfade	Crossfade	Crossfade

VALORES PARA ShowMenu

ShowMenu ->

<i>Object Id</i>	<i>ShowMenu</i>
<i>Object Type</i>	None
<i>Object Action</i>	ShowObject
<i>Action Target</i>	BackgroundDashboard
<i>Action Value</i>	ShowMenuBack

- Creamos las pestañas de cada menú conforme a nuestro diseño presentado en la siguiente imagen.



Fuente: SkinXL Editor

Fig. 4.124. Diseño de Pestaña Menu Principal.

Para la obtención de esta pestaña, el grupo de objetos Background esta conformado asi:

- 2 rounded rectangles (cuadrados redondeados)

VALORES PARA Botón Lector Principal, Botón Gauges GLCDs, Botón Medición Telemetría, Botón Economía Poder Flujo Aire, Botón Fuerzas G Dashboard Object Properties ->

<i>Rounded Rectangle</i>	
<i>Radius</i>	10
<i>Uniform Radius</i>	yes
<i>Pen</i>	
<i>Width</i>	0,5
<i>Color</i>	(0,0,0)
<i>Opacity</i>	255
<i>Brush</i>	
<i>Color</i>	(255,255,255)
<i>Opacity</i>	255
<i>Style</i>	Gradient
<i>Gradient Style</i>	Linear
<i>Gradient Color Stops</i>	ShowMenuBack

- 1 container (circle button) (botón circular), conformado de 3 elipses, y un container de detalles (clips de redondeo).

- **Dashboard Object Properties ->**

Ellipse	
Horizontal Radius	8,5
Vertical Radius	8,5
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Brush	
Color	(0,128,0)
Opacity	255
Style	Gradient
Gradient Style	Linear
Hatch Style	Diagonal Cross

- Text (texto)

VALORES PARA Lector Principal

- **Dashboard Object Properties ->**

Text	
Position	10;10
Horizontal Center	8,5
Vertical Center	8,5
Format String	Lector Principal
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA Gauges GLCDs

- **Dashboard Object Properties ->**

Text	
Position	10;10
Horizontal Center	8,5
Vertical Center	8,5
Format String	Gauges GLCDs
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA Medición Telemetría

- **Dashboard Object Properties ->**

Text	
Position	10;10
Horizontal Center	8,5
Vertical Center	8,5
Format String	Medicion Telemetría
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA Economía Poder Flujo Aire

- **Dashboard Object Properties ->**

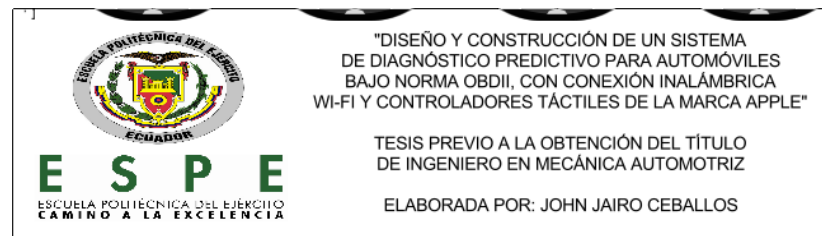
Text	
Position	10;10
Horizontal Center	8,5
Vertical Center	8,5
Format String	Economia Poder Flujo Aire
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA Button Fuerzas G

- **Dashboard Object Properties ->**

Text	
Position	10;10
Horizontal Center	8,5
Vertical Center	8,5
Format String	Fuerzas G
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

- Picture, donde seleccionamos una imagen prediseñada en otro software de edición de imágenes, de preferencia en formato png con fondo transparente, para que no haya interferencia de colores con el fondo del botón.
 - 1 container (Figura Casco), con un direccionador, una automatización de acción que hace una animación de movimiento el momento de dar un click sobre el botón dentro de la GUI.
- Terminado todas las pestañas, con su respectivo lugar y comprobación de funcionamiento con la needle tester, en nuestro diseño adicionamos el logotipo de la Universidad ESPEL, y el respectivo identificador de nuestro proyecto. En caso de otro usuario este espacio puede ser usado con cualquier otro fin de denominación o identificación del proyecto.



Fuente: SkinXL Editor

Fig. 4.125. Diseño de Identificador de Tesis.

- Para finalizar con el diseño de nuestros objetos dentro del plano del menú de selección, el momento de incluirlos en el Dashboard Object Settings, automáticamente se crean en el DashXL Editor una nueva sección, diferente de layer background y layer foreground, llamada layer (Buttons), en la que se encuentran los valores de control de los botones creados. La edición de valor en esta parte se divide en 2 reacciones que se obtienen de la activación de un botón, depressed y not depressed (presionado y suelto), conformado por:

- 5 containers de edición para cada botón

VALORES PARA Button 1, Button 2, Button 3, Button 4, Button 5

- **Dashboard Object Properties ->**

	<i>NotDepressed</i>	<i>Depressed</i>
Ellipse		
<i>Horizontal Radius</i>	6,5	6,5
<i>Vertical Radius</i>	6,5	6,5
Pen		
<i>Width</i>	0,5	0,5
<i>Color</i>	(0,0,0)	(0,0,0)
<i>Opacity</i>	255	255
<i>Style</i>	Transparent	Transparent
Brush		
<i>Color</i>	(255,255,255)	(0,64,0)
<i>Opacity</i>	255	255
<i>Style</i>	Transparent	Solid
<i>Gradient Style</i>	Radial	Radial
<i>Hatch Style</i>	Diagonal Cross	Diagonal Cross

4.3.2 PERSONALIZACIÓN DE LAS HERRAMIENTAS DE DIAGNÓSTICO.

La selección de la herramientas de diagnóstico que serán de nuestra utilidad, de entre muchas opciones que nos da la IPA y Objective-C, se realizó sin antes aplicar un filtro de nivel de beneficio, tomando en cuenta en primer momento el espacio y la visibilidad que se desea lograr, la visibilidad y aspecto pulcro dentro de la GUI, la organización interna de las plantillas, la cantidad de memoria que requeriría cada una para su aplicación y, principalmente, que el usuario debe tener una herramienta práctica y compacta de uso, amigable y cuyo aspecto no demuestre el nivel de complejidad incluido en toda la programación.

Es así que en nuestro proyecto se seleccionaron las siguientes herramientas de diagnóstico:

- COMPRESIÓN Y VACÍO
- PODER
- TORQUE
- TPS (SENSOR POSICIÓN MARIPOSA)

- APP (SENSOR POSICIÓN PEDAL ACELERADOR)
- MEDIDOR INTERACTIVO PREDICTIVO (RPM, Gs, FLUJO DE COMBUSTIBLE, ECT, IAT, VOLTAJE, HPs, etc).
- MAF
- MAP
- CARGA COMPRESIÓN

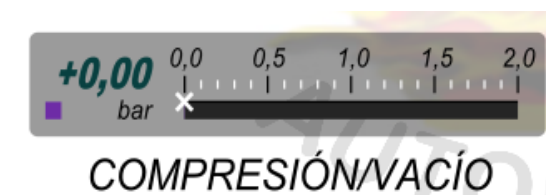
Estos elementos de diagnóstico, incursionan en los campos de principal afección en cuanto a diagnóstico predictivo como es el motor, su compresión/vacío, poder y torque que demuestre el motor en pruebas de carretera, sensores involucrados en la aceleración y aumento de velocidad, la temperatura, el flujo de combustible que ingrese al motor, los sensores MAF y MAP, la carga de compresión, y problemas referentes al adelanto al encendido.

a. MEDIDORES DE COMPRESIÓN Y VACÍO

Esta herramienta de diagnóstico es de inmensa utilidad para conocer en parte el estado del motor, predecir la presencia de fugas, roturas, empaques en mal estado, mal ajuste de pernos de sujeción, entre otros.

Para su diseño óptimo, se debe crear una escala de trabajo, con valores mínimo y máximo, rango con color de valores permisibles, medidor numérico y señalador de unidades de trabajo. Además se debe seguir una imagen de acción de fácil recepción y lectura, como en casos de medidores de competición.

Estéticamente, nuestra herramienta de medición de compresión y vacío tiene el siguiente aspecto:



Fuente: SkinXL Editor

Fig. 4.126. Diseño de Medidor Compresión/Vacío

DISEÑO DE MEDIDOR COMPRESIÓN Y VACÍO.

- Adicionamos un nuevo Dashboard, y lo llamamos Lector Principal.
(*Dashboard Screen Settings -> Add Dashboard -> Dashboard Settings -> Dashboard Id = Lector Principal*).
- A partir de las características de una plantilla de partida, adicionamos 3 objetos (2 medidores, 1 actuador de ciclo) que vamos a utilizar para el funcionamiento de la needle (aguja), tanto del visualizador numérico digital, como el de rango dinámico con los siguientes valores.

(*Dashboard Object Settings -> Add Object*)

VALORES PARA BoostGaugeB, BoostGaugeF, CycleBoostGauge

Lector Principal ->

Object Id	BoostGaugeB	BoostGaugeF	CycleBoostGauge
Object Type	Button	Button	None
Object Action	ExecuteMultipleObjectActions	ExecuteMultipleObjectActions	CycleState
Action Value	CycleBoostGauge	CycleBoostGauge	BoostState

- Partiendo del diseño gráfico deseado para el medidor, visualizado en la figura Fig. 4.126, debemos seleccionar que objetos incluir en su construcción. Es así que para la obtención de este aspecto, el grupo de objetos Background esta conformado por:

- 2 rounded rectangles (cuadrados redondeados)

VALORES PARA BoostGaugeB

Dashboard Object Properties ->

	Rounded Rectangle 1	Rounded Rectangle 2
Rectangle		
Center point	19,5; 87,5	19,5; 84
Size	69; 23	65; 12
Rounded Rectangle		
Radius	3	1
Uniform Radius	Yes	yes
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(37,37,37)	(0,0,0)

Opacity	255	100
Style	Solid	Solid
Gradient Style	Radial	Radial

- Text (texto), tanto para Compresión y Vacío, como para la unidad (bar o psi)

VALORES PARA BoostGaugeB

Dashboard Object Properties ->

Text	<i>Text (COMPRESION/VACIO)</i>	<i>Text (bar psi)</i>
Position	19,5; 94,5	2,5; 87
Horizontal Center	Center	Right
Vertical Center	Center	Center
Format String	COMPRESIÓN/VACÍO	bar
OBD-II Value		
PID	SAE.RPM	SAE.RPM
Value Type	Value (Number)	Value (Number)
Unit System	Metric	Metric
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

- Un rectángulo (rectangle), que servirá de guía para el transcurso de la aguja de medición por rango.

VALORES PARA BoostGaugeB

Dashboard Object Properties ->

	<i>Rectangle</i>
Rectangle	
Center point	27,5; 87
Size	42; 2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(37,37,37)
Opacity	255
Style	Solid
Gradient Style	Radial

- Una línea, que servirá de direccionador de la aguja de medición.

	<i>Line</i>
Line	
Start point	6,5; 86
End point	48,5; 86
Width	0,25
Pen	
Width	0,1

Color	(255,255,255)
Opacity	255
Style	Transparent
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial

- Dos contenedor especiales llamados Tick Marks, que son nuestro rango de trabajo para la lectura, el mismo que contiene líneas y texto así:

	Tick Marks 1	Tick Marks 2
Tick Marks		
Type	Straight	Straight
Tick Count	5	21
Skip Tick List		0,5,10,15,20
Label Start Range	0	0
Label End Range	2	2
Straight Ticks		
Direction	First Side	First Side
Start point	6,5; 85	6,5; 84,5
End point	48,5; 85	48,5; 84,5
Length	2	1
Label Position	4	4

- Terminada la edición física del medidor, en la sección de layer foreground, donde se encuentra posicionado nuestro objeto BoostGaugeF, se localizan todas las opciones de automatización de la herramienta, conformadas por:
 - Tres ítems de estado a partir de BoostState (ShowCurBoost, ShowMaxBoost, ShowMinBoost), para señalar la compresión actual, máxima y mínima respectivamente. Sus características son:

VALORES PARA ShowCurBoost (COMPRESIÓN ACTUAL)

Dashboard Object Properties ->

	Condition (es negativa)
Condition 1	
Operator	Less than
Condition Value	0
Condition 2	
Logic Operator	None
Operator	Equal
Condition Value	0

OBD-II Value 1	
PID	CALC.BOOST_PRESSURE
Value Type	Value (Number)
Unit System	Metric
OBD-II Value 1	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

Text	<i>Text (COMPRESIÓN)</i>
Position	2,5; 82,5
Horizontal Center	Right
Vertical Center	Center
Format String	%+04.2f
Use OBD-II Value	Yes
OBD-II Value	
PID	CALC.BOOST_PRESSURE
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,64,64)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

- o La automatización del rango dinámico mediante un contenedor lógico, con un rectángulo y una condición de funcionamiento (es negativa?):

<i>Rectangle</i>	
Rectangle	
Center point	27,5; 87
Size	42; 2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(37,37,37)
Opacity	255
Style	Solid
Gradient Style	Radial

<i>Condition (es negativa)</i>	
Condition 1	
Operator	Less than
Condition Value	0
Condition 2	
Logic Operator	None
Operator	Equal
Condition Value	0
OBD-II Value 1	
PID	CALC.BOOST_PRESSURE
Value Type	Value (Number)
Unit System	Metric

OBD-II Value 1	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

- Dos needles de funcionamiento del señalador de rango, para opciones de funcionamiento de Máxima compresión, y de mínima compresión (máximo vacío):

VALORES PARA Needle (MAX COMPRESIÓN)

Dashboard Object Properties ->

<i>Needle (MAX COMPRESIÓN)</i>	
Needle	
Type	Linear
Range Start	0
Range End	2
Linear Needle	
Type	Offset
Start Point	6,5; 86
End Point	48,5; 86
Clip Point	50; 40
OBD-II Value	
PID	CALC.BOOST_PRESSURE
Value Type	Value (Number)
Unit System	Metric

VALORES PARA Needle (MIN COMPRESIÓN {MAX VACÍO})

Dashboard Object Properties ->

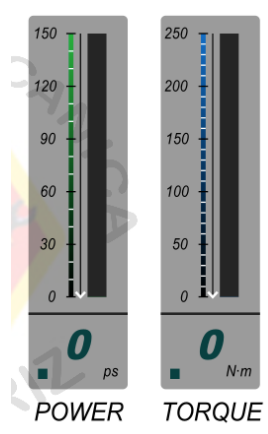
<i>Needle (MIN COMPRESIÓN)</i>	
Needle	
Type	Linear
Range Start	-2
Range End	0
Linear Needle	
Type	Offset
Start Point	48,5; 86
End Point	6,5; 86
Clip Point	50; 40
OBD-II Value	
PID	CALC.BOOST_PRESSURE
Value Type	Value Min (Number)
Unit System	Metric

b. MEDIDORES DE PODER Y TORQUE DEL MOTOR

Esta herramienta de diagnóstico nos ayuda al análisis instantáneo total o parcial de poder logrado por el motor. En conjunción con los medidores de compresión/vacío y de análisis de chispa, se puede diagnosticar pérdidas

de poder, tanto en estado estático del automóvil (ralentí), y en pruebas de manejo con el automóvil en movimiento.

Estéticamente, nuestras herramientas de medición de Poder y Torque tiene el siguiente aspecto:



Fuente: SkinXL Editor

Fig. 4.127. Diseño de Medidor de Poder y Torque (POWER y TORQUE)

DISEÑO DE MEDIDOR PODER (POWER).

- En la plantilla Lector Principal creada anteriormente, a partir de las características de una plantilla de partida, adicionamos 3 objetos (2 medidores y 1 actuador de ciclo) que vamos a utilizar para el funcionamiento de la needle (aguja), tanto del visualizador numérico digital, como el de rango dinámico con los siguientes valores.

(Dashboard Object Settings -> Add Object)

VALORES PARA PowerGaugeB, PowerGaugeF, CyclePowerGauge

Lector Principal ->

Object Id	PowerGaugeB	PowerGaugeF	CyclePowerGauge
Object Type	Button	Button	None
Object Action	ExecuteMultipleObjectActions	ExecuteMultipleObjectActions	CycleState
Action Value	CyclePowerGauge	CyclePowerGauge	PowerState

- Partiendo del diseño gráfico deseado para el medidor, visualizado en la figura Fig. 4.127, debemos seleccionar que objetos incluir en su construcción. Es así que para la obtención de este aspecto, el grupo de objetos Background esta conformado por:

- 2 rounded rectangles (cuadrados redondeados)

VALORES PARA PowerGaugeB

Dashboard Object Properties ->

	<i>Rounded Rectangle 1</i>	<i>Rounded Rectangle 2</i>
Rectangle		
Center point	93,5; 57	93,5; 53,5
Size	25; 84	21; 73
Rounded Rectangle		
Radius	3	1
Uniform Radius	Yes	yes
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(37,37,37)	(0,0,0)
Opacity	255	100
Style	Solid	Solid
Gradient Style	Radial	Radial

- Text (texto), tanto para Power, como para la unidad (kW o ps)

VALORES PARA PowerGaugeB

Dashboard Object Properties ->

Text	<i>Text (POWER)</i>	<i>Text (ps)</i>
Position	93,5; 94,5	102; 86,5
Horizontal Center	Center	Right
Vertical Center	Center	Center
Format String	POWER	ps
OBD-II Value		
PID	SAE.RPM	SAE.RPM
Value Type	Value (Number)	Value (Number)
Unit System	Metric	Metric
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

- Tres rectángulo (rectangle), que servirán de base, medidor y de guía para el transcurso de la aguja de medición por rango respectivamente.

VALORES PARA PowerGaugeB

Dashboard Object Properties ->

	Rectangle 1	Rectangle 2	Rectangle 3
Rectangle			
Center point	93,5; 75,25	97,5; 46,25	92; 46,25
Size	21; 0,5	4; 51,5	1; 51,5
Pen			
Width	0,5	0,5	0,5
Color	(0,0,0)	(0,0,0)	(0,0,0)
Opacity	255	255	255
Style	Transparent	Transparent	Transparent
Brush			
Color	(37,37,37)	(37,37,37)	(37,37,37)
Opacity	255	255	255
Style	Solid	Solid	Solid
Gradient Style	Radial	Radial	Radial

- Una línea, que servirá de direccionador de la aguja de medición.

	Line
Line	
Start point	94; 72
End point	94; 20,5
Width	0,25
Pen	
Width	0,1
Color	(255,255,255)
Opacity	255
Style	Transparent
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial

- Dos contenedor especiales llamados Tick Marks, que son nuestro rango de trabajo para la lectura, el mismo que contiene líneas y texto así:

	Tick Marks 1	Tick Marks 2
Tick Marks		
Type	Straight	Straight
Tick Count	6	16
Skip Tick List		0,3,6,9,12,15
Label Start Range	0	0
Label End Range	150	150
Straight Ticks		
Direction	First Side	First Side
Start point	93; 72	92,5; 72
End point	93; 20,5	92,5; 20,5
Length	2	1
Label Position	4,5	6

- Terminada la edición física del medidor, en la sección de layer foreground, donde se encuentra posicionado nuestro objeto PowerGaugeF, se localizan todas las opciones de automatización de la herramienta, conformadas por:
 - Dos ítems de estado a partir de PowerState (ShowCurPower, ShowMaxPower), para señalar estado del poder del motor actual y máxima respectivamente. Sus características son:

VALORES PARA ShowCurPower

Dashboard Object Properties ->

Text	<i>Text (POWER)</i>
Position	93,5; 81
Horizontal Center	Center
Vertical Center	Center
Format String	%.0f
Use OBD-II Value	Yes
OBD-II Value	
PID	CALC.ENGINE_POWER
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,64,64)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

- La automatización del rango dinámico mediante un contenedor lógico, con un rectángulo y una needle de funcionamiento:

	<i>Rectangle</i>
Rectangle	
Center point	97,5; 46,25
Size	4; 51,5
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(37,37,37)
Opacity	255
Style	Solid
Gradient Style	Radial

<i>Needle</i>	
Needle	
Type	Linear
Range Start	0
Range End	150
Linear Needle	
Type	Offset
Start Point	95,5; 72
End Point	95,5; 20,5
Clip Point	50; 40
OBD-II Value	
PID	CALC.ENGINE_POWER
Value Type	Value (Number)
Unit System	Metric

- Una needle de funcionamiento del señalador de rango, para opciones de funcionamiento de Poder actual y alcance de Poder Máximo.

VALORES PARA Needle (RANGO DINÁMICO PODER)

Dashboard Object Properties ->

<i>Needle (RANGO DINÁMICO PODER)</i>	
Needle	
Type	Linear
Range Start	0
Range End	150
Linear Needle	
Type	Offset
Start Point	94; 72
End Point	94; 20,5
Clip Point	50; 40
OBD-II Value	
PID	CALC.ENGINE_POWER
Value Type	Value (Number)
Unit System	Metric

DISEÑO DE MEDIDOR TORQUE (TORQUE).

- En la plantilla Lector Principal creada anteriormente, a partir de las características de una plantilla de partida, adicionamos 3 objetos (2 medidores y 1 actuador de ciclo) que vamos a utilizar para el funcionamiento de la needle (aguja), tanto del visualizador numérico digital, como el de rango dinámico con los siguientes valores.

(Dashboard Object Settings -> Add Object)

VALORES PARA TorqueGaugeB, TorqueGaugeF, CycleTorqueGauge

Lector Principal ->

Object Id	TorqueGaugeB	TorqueGaugeF	CycleTorqueGauge
Object Type	Button	Button	None
Object Action	ExecuteMultipleObjectActions	ExecuteMultipleObjectActions	CycleState
Action Value	CycleTorqueGauge	CycleTorqueGauge	TorqueState

- Partiendo del diseño gráfico deseado para el medidor, visualizado en la figura Fig. 4.127, debemos seleccionar que objetos incluir en su construcción. Es así que para la obtención de este aspecto, el grupo de objetos Background esta conformado por:

- 2 rounded rectangles (cuadrados redondeados)

VALORES PARA TorqueGaugeB

Dashboard Object Properties ->

	Rounded Rectangle 1	Rounded Rectangle 2
Rectangle		
Center point	119,5; 57	119,5; 53,5
Size	25; 84	21; 73
Rounded Rectangle		
Radius	3	1
Uniform Radius	Yes	yes
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(37,37,37)	(0,0,0)
Opacity	255	100
Style	Solid	Solid
Gradient Style	Radial	Radial

- Text (texto), tanto para Torque, como para la unidad (N.m)

VALORES PARA TorqueGaugeB

Dashboard Object Properties ->

Text	Text (TORQUE)	Text (N.m)
Position	119,5; 94,5	128; 86,5
Horizontal Center	Center	Right
Vertical Center	Center	Center
Format String	TORQUE	N.m
OBD-II Value		
PID	SAE.RPM	SAE.RPM
Value Type	Value (Number)	Value (Number)
Unit System	Metric	Metric
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

- Tres rectángulo (rectangle), que servirán de base, medidor y de guía para el transcurso de la aguja de medición por rango respectivamente.

VALORES PARA PowerGaugeB

Dashboard Object Properties ->

	Rectangle 1	Rectangle 2	Rectangle 3
Rectangle			
Center point	119,5; 75,25	123,5; 46,25	118; 46,25
Size	21; 0,5	4; 51,5	1; 51,5
Pen			
Width	0,5	0,5	0,5
Color	(0,0,0)	(0,0,0)	(0,0,0)
Opacity	255	255	255
Style	Transparent	Transparent	Transparent
Brush			
Color	(37,37,37)	(37,37,37)	(37,37,37)
Opacity	255	255	255
Style	Solid	Solid	Solid
Gradient Style	Radial	Radial	Radial

- Una línea, que servirá de direccionador de la aguja de medición.

	Line
Line	
Start point	120; 72
End point	120; 20,5
Width	0,25
Pen	
Width	0,1
Color	(255,255,255)
Opacity	255
Style	Transparent
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial

- Dos contenedor especiales llamados Tick Marks, que son nuestro rango de trabajo para la lectura, el mismo que contiene líneas y texto así:

	Tick Marks 1	Tick Marks 2
Tick Marks		
Type	Straight	Straight
Tick Count	6	26
Skip Tick List		0,5,10,15,20,25,30,35
Label Start Range	0	0
Label End Range	250	250
Straight Ticks		
Direction	First Side	First Side

<i>Start point</i>	119; 72	118,5; 72
<i>End point</i>	119; 20,5	118,5; 20,5
<i>Length</i>	2	1
<i>Label Position</i>	4,5	6

- Terminada la edición física del medidor, en la sección de layer foreground, donde se encuentra posicionado nuestro objeto TorqueGaugeF, se localizan todas las opciones de automatización de la herramienta, conformadas por:

- Dos ítems de estado a partir de TorqueState (ShowCurTorque, ShowMaxTorque), para señalar estado del poder del motor actual y máxima respectivamente. Sus características son:

VALORES PARA ShowCurTorque

Dashboard Object Properties ->

Text	<i>Text (POWER)</i>
<i>Position</i>	119,5; 81
<i>Horizontal Center</i>	Center
<i>Vertical Center</i>	Center
<i>Format String</i>	%.0f
<i>Use OBD-II Value</i>	Yes
OBD-II Value	
<i>PID</i>	CALC.ENGINE_TORQUE
<i>Value Type</i>	Value (Number)
<i>Unit System</i>	Metric
Brush	
<i>Color</i>	(0,64,64)
<i>Opacity</i>	255
<i>Style</i>	Solid
<i>Gradient Style</i>	Radial
<i>Hatch Style</i>	Diagonal Cross

- La automatización del rango dinámico mediante un contenedor lógico, con un rectángulo y una needle de funcionamiento:

	<i>Rectangle</i>
Rectangle	
<i>Center point</i>	123,5; 46,25
<i>Size</i>	4; 51,5
Pen	
<i>Width</i>	0,5
<i>Color</i>	(0,0,0)
<i>Opacity</i>	255
<i>Style</i>	Transparent
Brush	
<i>Color</i>	(37,37,37)

Opacity	255
Style	Solid
Gradient Style	Radial

Needle	
Needle	
Type	Linear
Range Start	0
Range End	250
Linear Needle	
Type	Offset
Start Point	122; 72
End Point	122; 20,5
Clip Point	50; 40
OBD-II Value	
PID	CALC.ENGINE_TORQUE
Value Type	Value (Number)
Unit System	Metric

- Una needle de funcionamiento del señalador de rango, para opciones de funcionamiento de Poder actual y alcance de Poder Máximo.

VALORES PARA Needle (RANGO DINÁMICO PODER)

Dashboard Object Properties ->

Needle (RANGO DINÁMICO PODER)	
Needle	
Type	Linear
Range Start	0
Range End	250
Linear Needle	
Type	Offset
Start Point	120; 72
End Point	120; 20,5
Clip Point	50; 40
OBD-II Value	
PID	CALC.ENGINE_TORQUE
Value Type	Value (Number)
Unit System	Metric

c. MEDIDORES DE TPS (SENSOR DE POSICIÓN DE MARIPOSA)

Esta herramienta de diagnóstico nos ayuda en la comprobación del correcto funcionamiento de nuestro sistema de aceleración. Un TPS en mal estado influye enormemente en el desempeño de nuestro automóvil, por cuanto se reduce la relación aire/combustible del nivel óptimo de trabajo.

Estéticamente, nuestra herramienta de medición de TPS tiene el siguiente aspecto:



Fuente: SkinXL Editor

Fig. 4.128. Diseño de Medidor de TPS (Sensor de Posición de la mariposa)

DISEÑO DE TPS (Sensor de Posición de Mariposa).

- En la plantilla Lector Principal creada, creamos un contenedor con ID “Gauge Velocidad RPM”, dentro del que introducimos los siguientes componentes:
 - 1 rectangle (rectángulo) que servirá de base, medidor y de guía, como en casos anteriores, para la aguja (needle).

VALORES PARA TPS

Dashboard Object Properties ->

<i>Rectangle</i>	
Rectangle	
Center point	28,25; 62
Size	42,5; 2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(37,37,37)
Opacity	255
Style	Solid
Gradient Style	Radial

- Text (texto), para el valor numérico porcentual de su apertura/cierre (%).

VALORES PARA TPS

Dashboard Object Properties ->

Text	<i>Text (%)</i>
Position	2,5; 62
Horizontal Center	Left
Vertical Center	Center
Format String	%
OBD-II Value	
PID	SAE.RPM

<i>Value Type</i>	<i>Value (Number)</i>
<i>Unit System</i>	<i>Metric</i>
Brush	
<i>Color</i>	<i>(0,0,0)</i>
<i>Opacity</i>	<i>255</i>
<i>Style</i>	<i>Solid</i>
<i>Gradient Style</i>	<i>Radial</i>
<i>Hatch Style</i>	<i>Diagonal Cross</i>

- Un direccionador de imagen PATH, para la creación de la imagen de la mariposa, en estilo de pixelaje monocromático b/n. La configuración de conformación de figura se la creó con los siguientes valores:

The screenshot shows the 'Arc To Settings' section of the SkinXL Editor. It contains the following fields and controls:

- Three sets of coordinate input fields: X1: 0, Y1: 0; X2: 0, Y2: 0; X3: 0, Y3: 0. Each set has a 'Pick' button to the right.
- 'Arc To Settings' section with:
 - rx: 0, ry: 83, x-axis-rotation: 0
 - Checkboxes: large-arc-flag, sweep-flag

Fuente: SkinXL Editor

Fig. 4.129. Configuración Path TPS

Además se ingresaron los siguientes valores de gráficos internos:

```

move_to(-8,335, 63,25);
line_to(-8,002, 63,25);
line_to(-8,002, 60,252);
line_to(-8,335, 60,252);
line_to(-8,335, 63,25);
close_path();
move_to(-9,276, 61,859);
arc_to(0,25, 0,25, -180,, 0, 0, -9,609, 61,526);
line_to(-10,208, 60,927);
arc_to(0,083, 0,083, -180,, 0, 0, -10,325, 60,927);
arc_to(0,083, 0,083, -180,, 0, 0, -10,325, 61,045);
line_to(-9,727, 61,643);
arc_to(0,25, 0,25, -180,, 0, 0, -9,393, 61,977);
line_to(-8,795, 62,575);
arc_to(0,083, 0,083, -180,, 0, 0, -8,677, 62,575);
arc_to(0,083, 0,083, -180,, 0, 0, -8,677, 62,458);
line_to(-9,276, 61,859);
close_path();
move_to(-11,, 63,25);
line_to(-10,667, 63,25);
line_to(-10,667, 60,252);
line_to(-11,, 60,252);
line_to(-11,, 63,25);

```

close_path());

- Dos contenedor especiales llamados Tick Marks, que son nuestro rango de trabajo para la lectura, el mismo que contiene líneas y texto así:

	<i>Tick Marks 1</i>	<i>Tick Marks 2</i>
Tick Marks		
Type	Straight	Straight
Tick Count	11	21
Skip Tick List		0,2,4,6,8,10,12,14,16,18,20
Label Start Range	0	0
Label End Range	100	100
Straight Ticks		
Direction	First Side	First Side
Start point	7; 60	7; 59,5
End point	49,5; 60	49,5; 59,5
Length	2	1
Label Position	7	7

- Terminada la edición física del medidor, en la sección de layer foreground, donde se encuentra posicionado nuestro contenedor Gauge Velocidad RPM, se localizan todas las opciones de automatización de la herramienta:
 - Un contenedor de objetos, con un medidor dinámico y una aguja (needle) de activación del medidor:

VALORES PARA TPS

Dashboard Object Properties ->

	<i>Rectangle</i>
Rectangle	
Center point	28,25; 62
Size	42,5; 2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(37,37,37)
Opacity	255
Style	Solid
Gradient Style	Radial

	<i>Needle</i>
Needle	
Type	Linear
Range Start	0

Range End	100
Linear Needle	
Type	Offset
Start Point	7; 57
End Point	49,5; 57
Clip Point	50; 40
OBD-II Value	
PID	SAE.LOAD_PCT
Value Type	Value (Number)
Unit System	Metric

- Un medidor numérico dinámico, mediante opción TEXT:

VALORES PARA Text (TPS)

Dashboard Object Properties ->

Text	Text (TPS)
Position	-3,5; 62
Horizontal Center	Left
Vertical Center	Center
Format String	%0f
Use OBD-II value	Yes
OBD-II Value	
PID	SAE.LOAD_PCT
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

d. MEDIDORES DE APP (SENSOR DE POSICIÓN DEL PEDAL DEL ACELERADOR)

Esta herramienta de diagnóstico es de apoyo en situaciones de mal funcionamiento, o pérdida de aceleración de automóviles actuales, donde la aceleración es controlada por este sensor únicamente.

Estéticamente, nuestra herramienta de medición de APP tiene el siguiente aspecto:



Fuente: SkinXL Editor

Fig. 4.130. Diseño de Medidor de APP (Sensor de Posición del pedal del Acelerador)

DISEÑO DE MEDIDOR DE APP (Sensor de Posición de Pedal del Acelerador).

- En el contenedor “Gauge Velocidad RPM”, introducimos los siguientes componentes:
 - 1 rectangle (rectángulo) que servirá de base, medidor y de guía, como en casos anteriores, para la aguja (needle).

VALORES PARA APP

Dashboard Object Properties ->

<i>Rectangle</i>	
Rectangle	
Center point	28,25; 56
Size	42,5; 2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(37,37,37)
Opacity	255
Style	Solid
Gradient Style	Radial

- Text (texto), para el valor numérico porcentual de la cantidad de viaje que se produce en el sensor (%).

VALORES PARA APP

Dashboard Object Properties ->

Text	<i>Text (%)</i>
Position	2,5; 56
Horizontal Center	Left
Vertical Center	Center
Format String	%
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

- Un direccionador de imagen PATH, para la creación de la imagen en esta ocasión del acelerador , en estilo de pixelaje monocromático b/n. La configuración de conformación de figura se la creó con los siguientes valores:

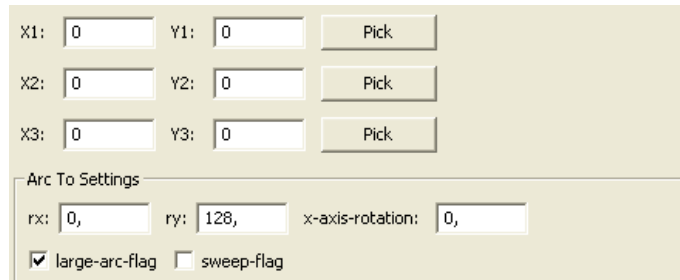


Fig. 4.131. Configuración Path TPS

Además se ingresaron los siguientes valores de gráficos internos:

```

move_to(-9,503, 55,688);
arc_to(0,128, 0,123, 0,, 1, 1, -9,758, 55,688);
arc_to(0,128, 0,123, 0,, 1, 1, -9,503, 55,688);
close_path();
move_to(-9,503, 56,058);
arc_to(0,128, 0,123, 0,, 1, 1, -9,758, 56,058);
arc_to(0,128, 0,123, 0,, 1, 1, -9,503, 56,058);
close_path(); move_to(-9,503, 56,428);
arc_to(0,128, 0,123, 0,, 1, 1, -9,758, 56,428);
arc_to(0,128, 0,123, 0,, 1, 1, -9,503, 56,428);
close_path();
move_to(-9,716, 55,317);
arc_to(0,128, 0,123, 0,, 1, 1, -9,971, 55,317);
arc_to(0,128, 0,123, 0,, 1, 1, -9,716, 55,317);
close_path();
move_to(-8,78, 56,839);
line_to(-8,78, 54,618);
arc_to(0,341, 0,329, -180,, 0, 0, -9,12, 54,289);
line_to(-9,919, 54,289);
arc_to(0,341, 0,329, -180,, 0, 0, -10,257, 54,665);
line_to(-10,06, 55,993);
arc_to(0,341, 0,329, 0,, 0, 1, -10,057, 56,039);
line_to(-10,057, 56,839);
line_to(-8,78, 56,839);
close_path();
move_to(-10,057, 56,921);
arc_to(0,341, 0,329, -180,, 0, 0, -9,716, 57,25);
line_to(-9,12, 57,25);
arc_to(0,341, 0,329, -180,, 0, 0, -8,78, 56,921);
line_to(-10,057, 56,921);
close_path();
move_to(-9,588, 54,947);
arc_to(0,128, 0,123, 0,, 1, 1, -9,844, 54,947);
arc_to(0,128, 0,123, 0,, 1, 1, -9,588, 54,947);

```

```

close_path();
move_to(-9,801, 54,577);
arc_to(0,128, 0,123, 0,, 1, 1, -10,057, 54,577);
arc_to(0,128, 0,123, 0,, 1, 1, -9,801, 54,577);
close_path();
move_to(-8,95, 55,482);
arc_to(0,128, 0,123, 0,, 1, 1, -9,206, 55,482);
arc_to(0,128, 0,123, 0,, 1, 1, -8,95, 55,482);
close_path();
move_to(-9,035, 54,577);
arc_to(0,128, 0,123, 0,, 1, 1, -9,29, 54,577);
arc_to(0,128, 0,123, 0,, 1, 1, -9,035, 54,577);
close_path();
move_to(-9,12, 55,852);
arc_to(0,128, 0,123, 0,, 1, 1, -9,376, 55,852);
arc_to(0,128, 0,123, 0,, 1, 1, -9,12, 55,852);
close_path();
move_to(-9,12, 56,222);
arc_to(0,128, 0,123, 0,, 1, 1, -9,376, 56,222);
arc_to(0,128, 0,123, 0,, 1, 1, -9,12, 56,222);
close_path();
move_to(-9,12, 56,592);
arc_to(0,128, 0,123, 0,, 1, 1, -9,376, 56,592);
arc_to(0,128, 0,123, 0,, 1, 1, -9,12, 56,592);
close_path(); move_to(-9,206, 54,947);
arc_to(0,128, 0,123, 0,, 1, 1, -9,461, 54,947);
arc_to(0,128, 0,123, 0,, 1, 1, -9,206, 54,947);
close_path();
move_to(-9,333, 55,317);
arc_to(0,128, 0,123, 0,, 1, 1, -9,588, 55,317);
arc_to(0,128, 0,123, 0,, 1, 1, -9,333, 55,317);
close_path();
move_to(-9,418, 54,577);
arc_to(0,128, 0,123, 0,, 1, 1, -9,674, 54,577);
arc_to(0,128, 0,123, 0,, 1, 1, -9,418, 54,577);
close_path();

```

- Terminada la edición física del medidor, en la sección de layer foreground, donde se encuentra posicionado nuestro contenedor Gauge Velocidad RPM, se localizan todas las opciones de automatización de la herramienta:
 - Un contenedor de objetos, con un medidor dinámico y una aguja (needle) de activación del medidor:

VALORES PARA APP

Dashboard Object Properties ->

<i>Rectangle</i>	
Rectangle	
<i>Center point</i>	28,25; 56
<i>Size</i>	42,5; 2
Pen	

Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(37,37,37)
Opacity	255
Style	Solid
Gradient Style	Radial

<i>Needle</i>	
Needle	
Type	Linear
Range Start	0
Range End	100
Linear Needle	
Type	Offset
Start Point	7; 57
End Point	49,5; 57
Clip Point	50; 40
OBD-II Value	
PID	SAE.APP_D
Value Type	Value (Number)
Unit System	Metric

- Un medidor numérico dinámico, mediante opción TEXT:

VALORES PARA Text (PEDAL)

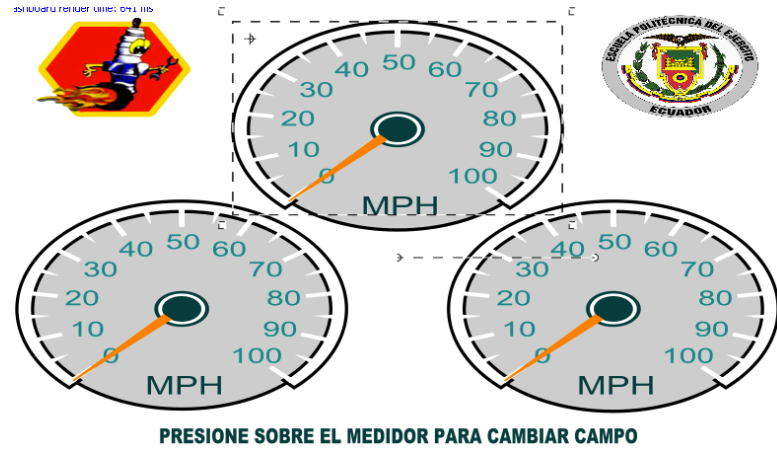
Dashboard Object Properties ->

Text	<i>Text (PEDAL)</i>
Position	-3,5; 56
Horizontal Center	Left
Vertical Center	Center
Format String	%.0f
Use OBD-II value	Yes
OBD-II Value	
PID	SAE.APP_D
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

e. MEDIDOR INTERACTIVO PREDICTIVO (RPM, Gs, FLUJO DE COMBUSTIBLE, ECT, IAT, VOLTAJE, HPs, etc).

La creación de una plantilla interactiva, donde se puedan ubicar los instrumentos de acuerdo a las necesidades del usuario, representa un gran avance dentro de la personalización informática de herramientas de diagnóstico.

El diseño estético de nuestro medidor interactivo predictivo, es diferente a los medidores anteriormente señalados, quedando así su aspecto final:



Fuente: SkinXL Editor

Fig. 4.132. Plantilla dinámica diagnóstico

Esta plantilla dinámica de diagnóstico contiene los siguientes medidores:

- Flujo de combustible.
- ECT
- IAT
- Voltaje
- HPs (Actual y Máxima)
- Torque (Actual y Máximo)
- MAF
- MAP
- Boost (Compresión).

DISEÑO DE MEDIDOR INTERACTIVO PREDICTIVO.

- Adicionamos un nuevo Dashboard, y lo llamamos Gauges GLCDs.
(*Dashboard Screen Settings -> Add Dashboard -> Dashboard Settings -> Dashboard Id = Gauges GLCDs*).

- A partir de las características de una plantilla de partida, en primera instancia adicionamos 3 objetos (botones) que nos van a permitir crear la funcionalidad interactiva de la herramienta.

(Dashboard Object Settings -> Add Object)

VALORES PARA RIGHT_BUTTON, LEFT_BUTTON, CENTER_BUTTON

Gauges GLCDs ->

Object Id	RIGHT_BUTTON	LEFT_BUTTON	CENTER_BUTTON
Object Type	Button	Button	Button
Object Action	ExecuteMultipleObjectActions	ExecuteMultipleObjectActions	ExecuteMultipleObjectActions
Action Value	SwitchRightBG,SwitchRightFG	SwitchLeftBG,SwitchLeftFG	SwitchCenterBG,SwitchCenterFG

- Adherimos los objetivos base de nuestra Action Value de cada Botón anterior, los mismos que cambiarán cada gauge de acuerdo a funcionalidad.

VALORES PARA *SwitchRightBG, SwitchRightFG, SwitchLeftBG, SwitchLeftFG, SwitchCenterBG, SwitchCenterFG.*

Gauges GLCDs ->

Object Id	SwitchRightBG	SwitchRightFG	SwitchLeftBG	SwitchLeftFG	SwitchCenterBG	SwitchCenterFG
Object Type	None	None	None	None	None	None
Object Action	CycleState	CycleState	CycleState	CycleState	CycleState	CycleState
Action Value	RIGHT_BG	RIGHT_FG	LEFT_BG	LEFT_BG	CENTER_BG	CENTER_BG

- Partiendo del diseño gráfico deseado para el medidor, visualizado en la figura Fig. 4.133, debemos seleccionar que objetos incluir en su construcción. Es así que para la obtención de este aspecto, el grupo de objetos Background esta conformado por:

- Un elemento TEXT, que tendrá la leyenda “PRESIONE SOBRE EL MEDIDOR PARA CAMBIAR CAMPO”.
- Tres elipses que darán forma a los medidores.

	Ellipse 1	Ellipse 2	Ellipse 3
Ellipse			
Center point	22,54; 64,516	50; 29,618	77,459; 64,956
Horizontal radius	20; 20	20; 20	20; 20
Vertical radius	20; 20	20; 20	20; 20
Pen			
Width	0,5	0,5	0,5
Color	(0,0,0)	(0,0,0)	(0,0,0)

<i>Opacity</i>	255	255	255
<i>Style</i>	Transparent	Transparent	Transparent
Brush			
<i>Color</i>	(208,208,208)	(208,208,208)	(208,208,208)
<i>Opacity</i>	255	255	255
<i>Style</i>	Solid	Solid	Solid
<i>Gradient Style</i>	Radial	Radial	Radial

- Dos figuras (pictures) con los logos de Ingeniería Automotriz y la Escuela Politécnica del Ejército.
- Tres contenedores con los diferentes medidores para que se cambien las escalas, las medidas, la aguja de medición, y la unidad de medida, variando únicamente la posición de los mismo (derecha, centro e izquierda). Estos contenedores tienen estos elementos:
 - Tres elipses que darán forma a el centro de la aguja.

	<i>Ellipse 1</i>	<i>Ellipse 2</i>	<i>Ellipse 3</i>
Ellipse			
<i>Center point</i>	22,5; 50	22,5; 50	22,5; 50
<i>Horizontal radius</i>	3,5	3	2,5
<i>Vertical radius</i>	3,5	3	2,5
Pen			
<i>Width</i>	0,5	0,5	0,5
<i>Color</i>	(0,0,0)	(0,0,0)	(0,0,0)
<i>Opacity</i>	255	255	255
<i>Style</i>	Transparent	Transparent	Transparent
Brush			
<i>Color</i>	(0,64,64)	(255,255,255)	(0,64,64)
<i>Opacity</i>	255	255	255
<i>Style</i>	Solid	Solid	Solid
<i>Gradient Style</i>	Radial	Radial	Radial

- Un contenedor de cambio de estado, con condición lógica de acuerdo a la pulsación del botón en el foreground, LEFT_BG, CENTER_BG Y RIGHT_BG, dentro del que se encuentra los ítems de estado de cada lector, que a la vez contienen un arco como guía de la aguja de medición, dos contenedores tipo Tick Marks que proveerán la escala, y un texto identificador.

VALORES PARA State Item <FuelFlow> (0 – 10)

Dashboard Object Properties ->

Arc	
Arc	
Center point	22,5; 50
Horizontal radius	20
Horizontal radius	20
Arc Angle	133
Arc Sweep	274
Arc Width	2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial

	Tick Marks (Largo)	Tick Marks (Pequeño)
Tick Marks		
Type	Circular	Circular
Tick Count	6	11
Skip Tick List		
Label Start Range	0	0
Label End Range	10	100
Circular Ticks		
Direction	Inside	Inside
Center Point	22,5; 50	22,5; 50
Horizontal radius	20	20
Vertical radius	20	20
Angle	135	135
Sweep	270	270
Length	20	15
Label Position	35	35

Text	Text
Position	22,5; 65
Horizontal Center	Center
Vertical Center	Center
Format String	Flujo Comb\nl/H
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA State Item <ECT> (175 - 250)

Dashboard Object Properties ->

Arc	
Arc	
Center point	22,5; 50
Horizontal radius	20
Horizontal radius	20
Arc Angle	133
Arc Sweep	274
Arc Width	2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial

	Tick Marks (Largo)	Tick Marks (Pequeño)
Tick Marks		
Type	Circular	Circular
Tick Count	6	16
Skip Tick List		
Label Start Range	175	0
Label End Range	250	100
Circular Ticks		
Direction	Inside	Inside
Center Point	22,5; 50	22,5; 50
Horizontal radius	20	20
Vertical radius	20	20
Angle	135	135
Sweep	270	270
Length	20	15
Label Position	35	35

Text	Text
Position	22,5; 65
Horizontal Center	Center
Vertical Center	Center
Format String	ECT
	°F
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA State Item <IAT> (50 - 200)

Dashboard Object Properties ->

Arc	
Arc	
Center point	22,5; 50
Horizontal radius	20

Horizontal radius	20
Arc Angle	133
Arc Sweep	274
Arc Width	2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial

	Tick Marks (Largo)	Tick Marks (Pequeño)
Tick Marks		
Type	Circular	Circular
Tick Count	6	16
Skip Tick List		
Label Start Range	50	0
Label End Range	200	100
Circular Ticks		
Direction	Inside	Inside
Center Point	22,5; 50	22,5; 50
Horizontal radius	20	20
Vertical radius	20	20
Angle	135	135
Sweep	270	270
Length	20	15
Label Position	35	35

Text	Text
Position	22,5; 65
Horizontal Center	Center
Vertical Center	Center
Format String	IAT °F
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA State Item <Volts> (12 - 16)

Dashboard Object Properties ->

	Arc
Arc	
Center point	22,5; 50
Horizontal radius	20
Horizontal radius	20
Arc Angle	133
Arc Sweep	274
Arc Width	2

Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial

	Tick Marks (Largo)	Tick Marks (Pequeño)
Tick Marks		
Type	Circular	Circular
Tick Count	5	9
Skip Tick List		
Label Start Range	12	0
Label End Range	16	100
Circular Ticks		
Direction	Inside	Inside
Center Point	22,5; 50	22,5; 50
Horizontal radius	20	20
Vertical radius	20	20
Angle	135	135
Sweep	270	270
Length	20	15
Label Position	35	35

Text	
Position	22,5; 65
Horizontal Center	Center
Vertical Center	Center
Format String	Volts
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA State Item <CurrentHP> (0 - 350)

Dashboard Object Properties ->

Arc	
Center point	22,5; 50
Horizontal radius	20
Horizontal radius	20
Arc Angle	133
Arc Sweep	274
Arc Width	2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	

Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial

	Tick Marks (Largo)	Tick Marks (Pequeño)
Tick Marks		
Type	Circular	Circular
Tick Count	8	15
Skip Tick List		
Label Start Range	0	0
Label End Range	350	100
Circular Ticks		
Direction	Inside	Inside
Center Point	22,5; 50	22,5; 50
Horizontal radius	20	20
Vertical radius	20	20
Angle	135	135
Sweep	270	270
Length	20	15
Label Position	35	35

Text	<i>Text</i>
Position	22,5; 65
Horizontal Center	Center
Vertical Center	Center
Format String	Wheel
	HP
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA State Item <MaxHP> (0 - 350)

Dashboard Object Properties ->

	Arc
Arc	
Center point	22,5; 50
Horizontal radius	20
Horizontal radius	20
Arc Angle	133
Arc Sweep	274
Arc Width	2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial

	Tick Marks (Largo)	Tick Marks (Pequeño)
Tick Marks		
Type	Circular	Circular
Tick Count	8	15
Skip Tick List		
Label Start Range	0	0
Label End Range	350	100
Circular Ticks		
Direction	Inside	Inside
Center Point	22,5; 50	22,5; 50
Horizontal radius	20	20
Vertical radius	20	20
Angle	135	135
Sweep	270	270
Length	20	15
Label Position	35	35

Text	<i>Text</i>
Position	22,5; 65
Horizontal Center	Center
Vertical Center	Center
Format String	Max HP
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA State Item <CurrentTorque> (0 - 350)

Dashboard Object Properties ->

	Arc
Arc	
Center point	22,5; 50
Horizontal radius	20
Horizontal radius	20
Arc Angle	133
Arc Sweep	274
Arc Width	2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial

	Tick Marks (Largo)	Tick Marks (Pequeño)
Tick Marks		
Type	Circular	Circular
Tick Count	8	15
Skip Tick List		
Label Start Range	0	0
Label End Range	350	100
Circular Ticks		
Direction	Inside	Inside
Center Point	22,5; 50	22,5; 50
Horizontal radius	20	20
Vertical radius	20	20
Angle	135	135
Sweep	270	270
Length	20	15
Label Position	35	35

Text	Text
Position	22,5; 65
Horizontal Center	Center
Vertical Center	Center
Format String	Wheel Torque
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA State Item <MaxTorque> (0 - 350)

Dashboard Object Properties ->

	Arc
Arc	
Center point	22,5; 50
Horizontal radius	20
Horizontal radius	20
Arc Angle	133
Arc Sweep	274
Arc Width	2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial

	Tick Marks (Largo)	Tick Marks (Pequeño)
Tick Marks		
Type	Circular	Circular
Tick Count	8	15

<i>Skip Tick List</i>		
<i>Label Start Range</i>	0	0
<i>Label End Range</i>	350	100
Circular Ticks		
<i>Direction</i>	Inside	Inside
<i>Center Point</i>	22,5; 50	22,5; 50
<i>Horizontal radius</i>	20	20
<i>Vertical radius</i>	20	20
<i>Angle</i>	135	135
<i>Sweep</i>	270	270
<i>Length</i>	20	15
<i>Label Position</i>	35	35

Text	<i>Text</i>
<i>Position</i>	22,5; 65
<i>Horizontal Center</i>	Center
<i>Vertical Center</i>	Center
<i>Format String</i>	Max Torque
OBD-II Value	
<i>PID</i>	SAE.RPM
<i>Value Type</i>	Value (Number)
<i>Unit System</i>	Metric
Brush	
<i>Color</i>	(0,0,0)
<i>Opacity</i>	255
<i>Style</i>	Solid
<i>Gradient Style</i>	Radial
<i>Hatch Style</i>	Diagonal Cross

VALORES PARA State Item <MAF> (0 - 20)

Dashboard Object Properties ->

<i>Arc</i>	
Arc	
<i>Center point</i>	22,5; 50
<i>Horizontal radius</i>	20
<i>Horizontal radius</i>	20
<i>Arc Angle</i>	133
<i>Arc Sweep</i>	274
<i>Arc Width</i>	2
Pen	
<i>Width</i>	0,5
<i>Color</i>	(0,0,0)
<i>Opacity</i>	255
<i>Style</i>	Transparent
Brush	
<i>Color</i>	(255,255,255)
<i>Opacity</i>	255
<i>Style</i>	Solid
<i>Gradient Style</i>	Radial

<i>Tick Marks (Largo)</i>		<i>Tick Marks (Pequeño)</i>
Tick Marks		
<i>Type</i>	Circular	Circular
<i>Tick Count</i>	5	21
<i>Skip Tick List</i>		
<i>Label Start Range</i>	0	0
<i>Label End Range</i>	20	100
Circular Ticks		

<i>Direction</i>	Inside	Inside
<i>Center Point</i>	22,5; 50	22,5; 50
<i>Horizontal radius</i>	20	20
<i>Vertical radius</i>	20	20
<i>Angle</i>	135	135
<i>Sweep</i>	270	270
<i>Length</i>	20	15
<i>Label Position</i>	35	35

Text	<i>Text</i>
<i>Position</i>	22,5; 65
<i>Horizontal Center</i>	Center
<i>Vertical Center</i>	Center
<i>Format String</i>	MAF g/s
OBD-II Value	
<i>PID</i>	SAE.RPM
<i>Value Type</i>	Value (Number)
<i>Unit System</i>	Metric
Brush	
<i>Color</i>	(0,0,0)
<i>Opacity</i>	255
<i>Style</i>	Solid
<i>Gradient Style</i>	Radial
<i>Hatch Style</i>	Diagonal Cross

VALORES PARA State Item <MAP> (0 - 20)

Dashboard Object Properties ->

	<i>Arc</i>
Arc	
<i>Center point</i>	22,5; 50
<i>Horizontal radius</i>	20
<i>Horizontal radius</i>	20
<i>Arc Angle</i>	133
<i>Arc Sweep</i>	274
<i>Arc Width</i>	2
Pen	
<i>Width</i>	0,5
<i>Color</i>	(0,0,0)
<i>Opacity</i>	255
<i>Style</i>	Transparent
Brush	
<i>Color</i>	(255,255,255)
<i>Opacity</i>	255
<i>Style</i>	Solid
<i>Gradient Style</i>	Radial

	<i>Tick Marks (Largo)</i>	<i>Tick Marks (Pequeño)</i>
Tick Marks		
<i>Type</i>	Circular	Circular
<i>Tick Count</i>	5	21
<i>Skip Tick List</i>		
<i>Label Start Range</i>	0	0
<i>Label End Range</i>	20	100
Circular Ticks		
<i>Direction</i>	Inside	Inside
<i>Center Point</i>	22,5; 50	22,5; 50
<i>Horizontal radius</i>	20	20
<i>Vertical radius</i>	20	20

Angle	135	135
Sweep	270	270
Length	20	15
Label Position	35	35

Text	Text
Position	22,5; 65
Horizontal Center	Center
Vertical Center	Center
Format String	MAP inHg
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA State Item <Boost> (-20 - 20)

Dashboard Object Properties ->

Arc	
Arc	
Center point	22,5; 50
Horizontal radius	20
Horizontal radius	20
Arc Angle	133
Arc Sweep	274
Arc Width	2
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial

	Tick Marks (Largo)	Tick Marks (Pequeño)
Tick Marks		
Type	Circular	Circular
Tick Count	5	21
Skip Tick List		
Label Start Range	-20	0
Label End Range	20	100
Circular Ticks		
Direction	Inside	Inside
Center Point	22,5; 50	22,5; 50
Horizontal radius	20	20
Vertical radius	20	20
Angle	135	135
Sweep	270	270
Length	20	15
Label Position	35	35

Text	<i>Text</i>
Position	22,5; 65
Horizontal Center	Center
Vertical Center	Center
Format String	Boost PSI
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

- Terminada la edición física del medidor, en la sección de layer foreground, donde se encuentra posicionado nuestros objetos complementarios Botón Izquierdo, Botón Derecho, Botón Central, Gauge Izquierdo, Gauge Derecho y Gauge Central , se localizan todas las opciones de automatización de la herramienta:
 - Contenedores de estado para cada Gauge Medidor creado en el background, con una needle (aguja) de medición dinámica:

VALORES PARA State Item <FuelFlow>

Dashboard Object Properties ->

	<i>Needle</i>
Needle	
Type	Circular
Range Start	0
Range End	10
Circular Needle	
Center Point	22,5; 50
Sweep	270
Linear Needle	
Type	Offset
Start Point	20; 40
End Point	80; 40
Clip Point	50; 40
OBD-II Value	
PID	CALC.FUEL_FLOW
Value Type	Value (Number)
Unit System	Metric

VALORES PARA State Item <ECT>

Dashboard Object Properties ->

<i>Needle</i>	
Needle	
Type	Circular
Range Start	0
Range End	10
Circular Needle	
Center Point	22,5; 50
Sweep	270
Linear Needle	
Type	Offset
Start Point	20; 40
End Point	80; 40
Clip Point	50; 40
OBD-II Value	
PID	SAE.ECT
Value Type	Value (Number)
Unit System	English

VALORES PARA State Item <IAT>

Dashboard Object Properties ->

<i>Needle</i>	
Needle	
Type	Circular
Range Start	0
Range End	10
Circular Needle	
Center Point	22,5; 50
Sweep	270
Linear Needle	
Type	Offset
Start Point	20; 40
End Point	80; 40
Clip Point	50; 40
OBD-II Value	
PID	SAE.IAT
Value Type	Value (Number)
Unit System	English

VALORES PARA State Item <Volts>

Dashboard Object Properties ->

<i>Needle</i>	
Needle	
Type	Circular
Range Start	0
Range End	10
Circular Needle	
Center Point	22,5; 50
Sweep	270
Linear Needle	
Type	Offset
Start Point	20; 40
End Point	80; 40

Clip Point	50; 40
OBD-II Value	
PID	VEHICLE.VOLTAGE
Value Type	Value (Number)
Unit System	Metric

VALORES PARA State Item <CurrentHP>

Dashboard Object Properties ->

	Needle
Needle	
Type	Circular
Range Start	0
Range End	10
Circular Needle	
Center Point	22,5; 50
Sweep	270
Linear Needle	
Type	Offset
Start Point	20; 40
End Point	80; 40
Clip Point	50; 40
OBD-II Value	
PID	CALC.ENGINE_POWER
Value Type	Value (Number)
Unit System	English

VALORES PARA State Item <MaxHP>

Dashboard Object Properties ->

	Needle
Needle	
Type	Circular
Range Start	0
Range End	10
Circular Needle	
Center Point	22,5; 50
Sweep	270
Linear Needle	
Type	Offset
Start Point	20; 40
End Point	80; 40
Clip Point	50; 40
OBD-II Value	
PID	CALC.ENGINE_POWER
Value Type	Value (Number)
Unit System	English

VALORES PARA State Item <CurrentTorque>

Dashboard Object Properties ->

	Needle
Needle	
Type	Circular
Range Start	0
Range End	10

Circular Needle	
Center Point	22,5; 50
Sweep	270
Linear Needle	
Type	Offset
Start Point	20; 40
End Point	80; 40
Clip Point	50; 40
OBD-II Value	
PID	CALC.ENGINE_TORQUE
Value Type	Value (Number)
Unit System	English

VALORES PARA State Item <MaxTorque>

Dashboard Object Properties ->

<i>Needle</i>	
Needle	
Type	Circular
Range Start	0
Range End	10
Circular Needle	
Center Point	22,5; 50
Sweep	270
Linear Needle	
Type	Offset
Start Point	20; 40
End Point	80; 40
Clip Point	50; 40
OBD-II Value	
PID	CALC.ENGINE_TORQUE
Value Type	Value (Number)
Unit System	Metric

VALORES PARA State Item <MAF>

Dashboard Object Properties ->

<i>Needle</i>	
Needle	
Type	Circular
Range Start	0
Range End	10
Circular Needle	
Center Point	22,5; 50
Sweep	270
Linear Needle	
Type	Offset
Start Point	20; 40
End Point	80; 40
Clip Point	50; 40
OBD-II Value	
PID	CALC.MAF
Value Type	Value (Number)
Unit System	Metric

VALORES PARA State Item <MAP>

Dashboard Object Properties ->

<i>Needle</i>	
Needle	
Type	Circular
Range Start	0
Range End	10
Circular Needle	
Center Point	22,5; 50
Sweep	270
Linear Needle	
Type	Offset
Start Point	20; 40
End Point	80; 40
Clip Point	50; 40
OBD-II Value	
PID	CALC.MAP
Value Type	Value (Number)
Unit System	Metric

VALORES PARA State Item <Boost>

Dashboard Object Properties ->

<i>Needle</i>	
Needle	
Type	Circular
Range Start	0
Range End	10
Circular Needle	
Center Point	22,5; 50
Sweep	270
Linear Needle	
Type	Offset
Start Point	20; 40
End Point	80; 40
Clip Point	50; 40
OBD-II Value	
PID	CALC.BOOST_PRESSURE
Value Type	Value (Number)
Unit System	English

En los medidores derecho e izquierdo se utilizan estos valores con sistemas Métrico e Inglés de medición cuando se vio la necesidad, pero para mayor comodidad y acceso a otras unidades inmediatas, el gauge central está configurado solamente para que muestre valores en el sistema Inglés.

- Botones de cambio de medidor al siguientes respectivamente según su ubicación en la lista.

VALORES PARA State Item <LEFT_BUTTON>, <CENTER_BUTTON>, <RIGHT_BUTTON>

	<i>NotDepressed</i>	<i>Depressed</i>
Ellipse		
Horizontal Radius	22,5	20
Vertical Radius	22,5	20
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(255,255,255)	(255,255,255)
Opacity	255	255
Style	Transparent	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

f. MEDIDORES DE MAF, MAP Y CARGA COMPRESIÓN

La lectura del estado de componentes fundamentales del automóvil, como lo son los sensores MAF, MAP y el cálculo de Carga Compresión, dentro del diagnóstico preventivo/predictivo es de gran utilidad, por cuanto son influyentes en el análisis funcional.

Por ejemplo el estudio del MAP mide la presión de aire que ingresa al múltiple de admisión del vehículo, entonces según la cantidad que mida este sensor, será la cantidad de gasolina que entregara el inyector. Este sensor funciona en conjunto con el sensor de posición del cigüeñal y juntos envían la señal a la ECU para inyectar la gasolina.

En palabras simples, lo que hace es elaborar una señal sobre cuanta presión de aire hay en la admisión, mas la señal de posición del cigüeñal, se las envían a la computadora y esta ordenará a los inyectores una cantidad optima de combustible.

Mientras que el sensor MAF es el encargado de la medición del flujo de aire que entra por el ducto de la admisión, variable muy importante con relación al funcionamiento del MAP.

Estéticamente, nuestra herramienta de medición de MAP, MAF y Carga Compresión tiene el siguiente aspecto:



Fuente: SkinXL Editor

Fig. 4.133. Diseño de Diagnóstico Flujo de Aire (MAF, MAP, CARGA)

DISEÑO DE MEDIDOR MAF, MAP, CARGA.

- Adicionamos un nuevo Dashboard, y lo llamamos Economía Poder Flujo de Aire.
(Dashboard Screen Settings -> Add Dashboard -> Dashboard Settings -> Dashboard Id = Economía Poder Flujo de Aire).
- Partiendo del diseño gráfico deseado para el medidor, visualizado en la figura Fig. 4.134, debemos seleccionar que objetos incluir en su construcción. Es así que para la obtención de este aspecto, el grupo de objetos Background Container (Contenedor Flujo Aire) esta conformado por:

- 2 rectangles (rectángulos)

VALORES PARA Container (Contenedor Flujo Aire)

Dashboard Object Properties ->

	Rectangle 1	Rectangle 2
Rectangle		
Center point	50; 15	20; 5
Size	90; 20	20; 5
Pen		
Width	0,5	0,5
Color	(128,0,0)	(128,0,0)
Opacity	255	255
Style	Solid	Solid
Brush		
Color	(255,255,255)	(0,0,0)
Opacity	255	255
Style	transparent	Solid
Gradient Style	Radial	Radial

- Text (texto), tanto para Compresión y Vacío, como para la unidad (bar o psi)

VALORES PARA Container (Contenedor Flujo Aire)

Dashboard Object Properties ->

Text	<i>Text</i>
Position	20; 5
Horizontal Center	Center
Vertical Center	Center
Format String	Flujo Aire
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(236,38,26)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

- Un container específico para MAP, MAF y Carga.

VALORES PARA Container (MAF)

Dashboard Object Properties ->

	<i>Rectangle 1</i>	<i>Rectangle 2</i>
Rectangle		
Center point	20; 16,125	20; 16,125
Size	20; 15	15; 8
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(236,38,26)	(236,38,26)
Opacity	255	255
Style	Gradient	Gradient
Gradient Style	Linear	Linear

Text	<i>Text</i>
Position	10,5; 9,25
Horizontal Center	Left
Vertical Center	Top
Format String	MAF
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA Container (MAP)

Dashboard Object Properties ->

	Rectangle 1	Rectangle 2
Rectangle		
Center point	20; 16,125	20; 16,125
Size	20; 15	15; 8
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(236,38,26)	(236,38,26)
Opacity	255	255
Style	Gradient	Gradient
Gradient Style	Linear	Linear

Text	Text
Position	10,5; 9,25
Horizontal Center	Left
Vertical Center	Top
Format String	MAP
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA Container (Carga)

Dashboard Object Properties ->

	Rectangle 1	Rectangle 2
Rectangle		
Center point	20; 16,125	20; 16,125
Size	20; 15	15; 8
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(236,38,26)	(236,38,26)
Opacity	255	255
Style	Gradient	Gradient
Gradient Style	Linear	Linear

Text	Text
Position	10,5; 9,25
Horizontal Center	Left
Vertical Center	Top

<i>Format String</i>	Carga
OBD-II Value	
<i>PID</i>	SAE.RPM
<i>Value Type</i>	Value (Number)
<i>Unit System</i>	Metric
Brush	
<i>Color</i>	(255,255,255)
<i>Opacity</i>	255
<i>Style</i>	Solid
<i>Gradient Style</i>	Radial
<i>Hatch Style</i>	Diagonal Cross

- Terminada la edición física del medidor, en la sección de layer foreground, vamos a configurar los medidores numéricos en el grupo Container (Flujo Aire), dentro del que están los contenedores de MAP, MAF y Carga.

VALORES PARA Container (MAF), Container (MAP), Container (Carga)

Dashboard Object Properties ->

Text	<i>Text (MAF)</i>	<i>Text (MAP)</i>	<i>Text (Carga)</i>
<i>Position</i>	20; 16,125	20; 16,125	20; 16,125
<i>Horizontal Center</i>	Center	Center	Center
<i>Vertical Center</i>	Center	Center	Center
<i>Format String</i>	%0.1f	%0.1f	%0.1f
<i>Use OBD-II Value</i>	Yes	yes	Yes
OBD-II Value			
<i>PID</i>	CALC.MAF	CALC.MAP	CALC.BOOST_PRESSURE
<i>Value Type</i>	Value (Number)	Value (Number)	Value (Number)
<i>Unit System</i>	Metric	Metric	Metric
Brush			
<i>Color</i>	(255,255,255)	(255,255,255)	(255,255,255)
<i>Opacity</i>	255	255	255
<i>Style</i>	Solid	Solid	Solid
<i>Gradient Style</i>	Radial	Radial	Radial
<i>Hatch Style</i>	Diagonal Cross	Diagonal Cross	Diagonal Cross

4.3.3 PERSONALIZACIÓN DE LOS ELEMENTOS DE MEDICIÓN Y TELEMETRÍA

El presente proyecto no fue concebido tan solo con la finalidad de proveer al profesional automotriz un grupo de herramientas de diagnóstico, sino al mismo tiempo desarrollar elementos influyentes en el campo de medición en tiempo real de la mayoría de sistemas disponibles y la aplicación de estas utilidades en la telemetría y, por qué no, en un futuro en la rama de la competición automotriz.

Es así que, en el área de la medición automotriz y la telemetría se incluyeron los siguientes elementos:

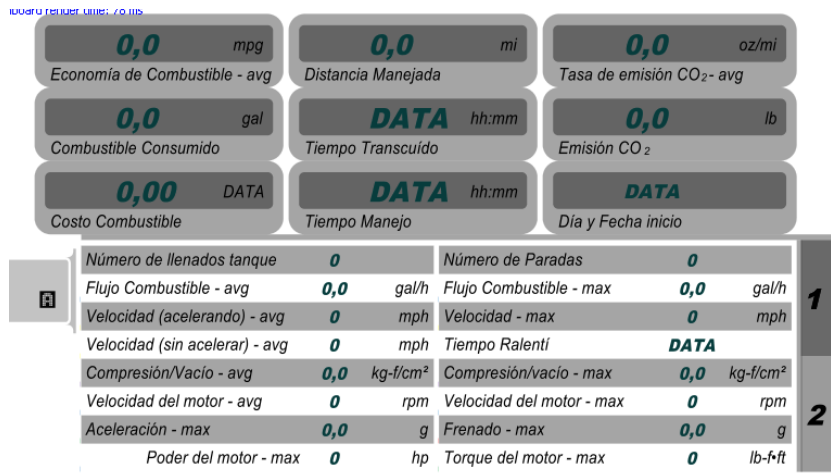
- Economía de combustible (average)
- Combustible consumido
- Costo del combustible
- Distancia manejada
- Tiempo transcurrido
- Tiempo de manejo
- Cantidad de emisión de CO₂ (average)
- Emisión de CO₂
- Tiempo y fecha de inicio.
- Número de llenados del tanque.
- Velocidad promedio
- Aceleración máxima

Adicionando a estos medidores, componentes ya analizados en cuanto a su utilidad y funcionalidad, como son:

- Flujo de combustible (average)
- Velocidad (con acelerador y sin acelerador)
- Compresión / vacío
- Velocidad del motor
- Poder del motor máximo
- Torque máximo
- Frenado máximo.

El uso de este tipo de plantilla, se ve optimizada en el uso de recursos de espacio mínimo visible para cada elemento, es decir, incluir casi todos estos componentes en una sola pantalla, ya que en muchos de los casos, son dependientes unos de otros.

Es así que esta plantilla en su totalidad tendría el siguiente diseño:



Fuente: SkinXL Editor

Fig. 4.134. Diseño de Dashboard Medición y Telemetría

DISEÑO DE DASHBOARD DE ELEMENTOS DE MEDICIÓN Y TELEMETRÍA.

- Adicionamos un nuevo Dashboard, y lo llamamos Medición Telemetría.

(Dashboard Screen Settings -> Add Dashboard -> Dashboard Settings -> Dashboard Id = Medicion Telemetria).

- Adicionamos dos buttons (botones), y les asignamos la denominación Button_Panel1 y Button_Panel2, los cuales nos van a permitir el cambio de paneles de información.

VALORES PARA Button_Panel1, Button_Panel2

Dashboard Object Properties ->

Object Id	Button_Panel1	Button_Panel2
Object Type	Button	Button
Object Action	SetState	SetState
Action Target	PanelStateA	PanelStateA
Action Value	ShowPanel1A	ShowPanel2A

- Partiendo del diseño gráfico deseado para la plantilla, visualizado en la figura Fig. 4.135, debemos seleccionar que objetos incluir en su construcción.

En los ítems Background se crearon cuatro grupos de elementos por contenedores (selección, superior, separador y datos secundarios)

- El grupo de selección tiene dos ítems Path y dos rounded rectangles (rectángulos redondeados).
- El contenedor superior esta compuesto por nueve containers correspondientes a cada lector principal de telemetría y medición automotriz, manejados por condicionadores lógicos.

VALORES PARA Container (Economía de Combustible - avg)

Dashboard Object Properties ->

	<i>Rounded Rectangle 1</i>	<i>Rounded Rectangle 2</i>
Rectangle		
Center point	9; 21,5	9; 20,25
Size	48; 13	44; 6,5
Rounded Rectangle		
Radius	3	1
Uniform Radius	Yes	yes
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(37,37,37)	(0,0,0)
Opacity	255	100
Style	Solid	Solid
Gradient Style	Radial	Radial

	<i>Condition (unidades métricas?)</i>
Condition 1	
Operator	Equal
Condition Value	l/100km
Condition 2	
Logic Operator	None
Operator	Equal
Condition Value	0
OBD-II Value 1	
PID	Fuel economy/consumption
Value Type	Unit/Format Setting (String)
Unit System	Metric
OBD-II Value 1	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

Si la condición es verdadera ->

Text	<i>Text 1</i>	<i>Text 2</i>
Position	-12; 25,75	29; 20,5
Horizontal Center	Left	Right
Vertical Center	Center	Center

Format String	Economía de Combustible - l/100km avg	
OBD-II Value		
PID	SAE.RPM	SAE.RPM
Value Type	Value (Number)	Value (Number)
Unit System	Metric	Metric
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

Si la condición es falsa ->

Text	Text 1	Text 2
Position	-12; 25,75	29; 20,5
Horizontal Center	Left	Right
Vertical Center	Center	Center
Format String	Economía de Combustible - mpg avg	
OBD-II Value		
PID	SAE.RPM	SAE.RPM
Value Type	Value (Number)	Value (Number)
Unit System	Metric	Metric
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Combustible Consumido)

Dashboard Object Properties ->

	<i>Rounded Rectangle 1</i>	<i>Rounded Rectangle 2</i>
Rectangle		
Center point	9; 35,5	9; 34,25
Size	48; 13	44; 6,5
Rounded Rectangle		
Radius	3	1
Uniform Radius	Yes	yes
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(37,37,37)	(0,0,0)
Opacity	255	100
Style	Solid	Solid
Gradient Style	Radial	Radial

	<i>Condition (unidades métricas?)</i>
Condition 1	
Operator	Equal
Condition Value	1
Condition 2	
Logic Operator	None
Operator	Equal
Condition Value	0

OBD-II Value 1	
PID	Volume
Value Type	Unit/Format Setting (String)
Unit System	Metric
OBD-II Value 1	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

Si la condición es ->

Condición		Verdadera	Falsa
Text	Text 1	Text	Text 2
Position	-12; 25,75	29; 34,5	29; 34,5
Horizontal Center	Left	Right	Right
Vertical Center	Center	Center	Center
Format String	Economía de Combustible - I avg	l	gal
OBD-II Value			
PID	SAE.RPM	SAE.RPM	SAE.RPM
Value Type	Value (Number)	Value (Number)	Value (Number)
Unit System	Metric	Metric	Metric
Brush			
Color	(0,0,0)	(0,0,0)	(0,0,0)
Opacity	255	255	255
Style	Solid	Solid	Solid
Gradient Style	Radial	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Costo Combustible)

Dashboard Object Properties ->

	Rounded Rectangle 1	Rounded Rectangle 2
Rectangle		
Center point	9; 49,5	9; 48,25
Size	48; 13	44; 6,5
Rounded Rectangle		
Radius	3	1
Uniform Radius	Yes	yes
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(37,37,37)	(0,0,0)
Opacity	255	100
Style	Solid	Solid
Gradient Style	Radial	Radial

Text	Text 1	Text 2
Position	-12; 53,75	29; 48,5
Horizontal Center	Left	Right
Vertical Center	Center	Center
Format String	Costo Combustible	%s
OBD-II Value		
PID	SAE.RPM	Currency Symbol

<i>Value Type</i>	Value (Number)	Unit/Format
<i>Unit System</i>	Metric	Metric
Brush		
<i>Color</i>	(0,0,0)	(0,0,0)
<i>Opacity</i>	255	255
<i>Style</i>	Solid	Solid
<i>Gradient Style</i>	Radial	Radial
<i>Hatch Style</i>	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Distancia Manejada)

Dashboard Object Properties ->

	<i>Rounded Rectangle 1</i>	<i>Rounded Rectangle 2</i>
Rectangle		
<i>Center point</i>	58; 21,5	58; 20,25
<i>Size</i>	48; 13	44; 6,5
Rounded Rectangle		
<i>Radius</i>	3	1
<i>Uniform Radius</i>	Yes	yes
Pen		
<i>Width</i>	0,5	0,5
<i>Color</i>	(0,0,0)	(0,0,0)
<i>Opacity</i>	255	255
<i>Style</i>	Transparent	Transparent
Brush		
<i>Color</i>	(37,37,37)	(0,0,0)
<i>Opacity</i>	255	100
<i>Style</i>	Solid	Solid
<i>Gradient Style</i>	Radial	Radial

	<i>Condition (unidades métricas?)</i>
Condition 1	
<i>Operator</i>	Equal
<i>Condition Value</i>	km
Condition 2	
<i>Logic Operator</i>	None
<i>Operator</i>	Equal
<i>Condition Value</i>	0
OBD-II Value 1	
<i>PID</i>	Distance
<i>Value Type</i>	Unit/Format Setting (String)
<i>Unit System</i>	Metric
OBD-II Value 1	
<i>PID</i>	SAE.RPM
<i>Value Type</i>	Value (Number)
<i>Unit System</i>	Metric

Si la condición es ->

Condición		<i>Verdadera</i>	<i>Falsa</i>
Text	<i>Text 1</i>	Text	Text 2
<i>Position</i>	37; 25,75	78; 20,5	78; 20,5
<i>Horizontal Center</i>	Left	Right	Right
<i>Vertical Center</i>	Center	Center	Center
<i>Format String</i>	Distancia Manejada	km	mil
OBD-II Value			
<i>PID</i>	SAE.RPM	SAE.RPM	SAE.RPM
<i>Value Type</i>	Value (Number)	Value (Number)	Value (Number)
<i>Unit System</i>	Metric	Metric	Metric

Brush			
Color	(0,0,0)	(0,0,0)	(0,0,0)
Opacity	255	255	255
Style	Solid	Solid	Solid
Gradient Style	Radial	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Tiempo Transcurrido)

Dashboard Object Properties ->

	<i>Rounded Rectangle 1</i>	<i>Rounded Rectangle 2</i>
Rectangle		
Center point	58; 35,5	58; 34,25
Size	48; 13	44; 6,5
Rounded Rectangle		
Radius	3	1
Uniform Radius	Yes	yes
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(37,37,37)	(0,0,0)
Opacity	255	100
Style	Solid	Solid
Gradient Style	Radial	Radial

Text	<i>Text 1</i>	<i>Text 2</i>
Position	37; 39,75	78; 34,5
Horizontal Center	Left	Right
Vertical Center	Center	Center
Format String	Tiempo Transcurrido	hh:mm
OBD-II Value		
PID	SAE.RPM	SAE.RPM
Value Type	Value (Number)	Value (Number)
Unit System	Metric	Metric
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Tiempo de Manejo)

Dashboard Object Properties ->

	<i>Rounded Rectangle 1</i>	<i>Rounded Rectangle 2</i>
Rectangle		
Center point	58; 49,5	58; 48,25
Size	48; 13	44; 6,5
Rounded Rectangle		
Radius	3	1
Uniform Radius	Yes	yes
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(37,37,37)	(0,0,0)
Opacity	255	100

Style	Solid	Solid
Gradient Style	Radial	Radial

Text	Text 1	Text 2
Position	37; 53,75	78; 48,5
Horizontal Center	Left	Right
Vertical Center	Center	Center
Format String	Tiempo Manejo	hh:mm
OBD-II Value		
PID	SAE.RPM	SAE.RPM
Value Type	Value (Number)	Value (Number)
Unit System	Metric	Metric
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Tasa de emisión CO2 - avg)

Dashboard Object Properties ->

	Rounded Rectangle 1	Rounded Rectangle 2
Rectangle		
Center point	107,5; 21,5	107,5; 20,25
Size	48; 13	44; 6,5
Rounded Rectangle		
Radius	3	1
Uniform Radius	Yes	yes
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(37,37,37)	(0,0,0)
Opacity	255	100
Style	Solid	Solid
Gradient Style	Radial	Radial

	Condition (unidades métricas?)
Condition 1	
Operator	Equal
Condition Value	G/km
Condition 2	
Logic Operator	None
Operator	Equal
Condition Value	0
OBD-II Value 1	
PID	CO2 emission
Value Type	Unit/Format Setting (String)
Unit System	Metric
OBD-II Value 1	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

Si la condición es ->

Condición		Verdadera	Falsa
Text	Text 1	Text	Text 2
Position	86; 25,75	128; 20,5	128; 20,5
Horizontal Center	Left	Right	Right
Vertical Center	Center	Center	Center
Format String	Tasa de emisión CO - avg	g/km	oz/mi
OBD-II Value			
PID	SAE.RPM	SAE.RPM	SAE.RPM
Value Type	Value (Number)	Value (Number)	Value (Number)
Unit System	Metric	Metric	Metric
Brush			
Color	(0,0,0)	(0,0,0)	(0,0,0)
Opacity	255	255	255
Style	Solid	Solid	Solid
Gradient Style	Radial	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Emisión CO2)

Dashboard Object Properties ->

	Rounded Rectangle 1	Rounded Rectangle 2
Rectangle		
Center point	107,5; 35,5	107,5; 34,25
Size	48; 13	44; 6,5
Rounded Rectangle		
Radius	3	1
Uniform Radius	Yes	yes
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(37,37,37)	(0,0,0)
Opacity	255	100
Style	Solid	Solid
Gradient Style	Radial	Radial

Condition (unidades métricas?)	
Condition 1	
Operator	Equal
Condition Value	G/km
Condition 2	
Logic Operator	None
Operator	Equal
Condition Value	0
OBD-II Value 1	
PID	CO2 emission
Value Type	Unit/Format Setting (String)
Unit System	Metric
OBD-II Value 1	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

Si la condición es ->

Condición		Verdadera	Falsa
Text	Text 1	Text	Text 2
Position	86; 39,75	128; 34,5	128; 34,5
Horizontal Center	Left	Right	Right

Vertical Center	Center	Center	Center
Format String	Emisión CO	kg	lb
OBD-II Value			
PID	SAE.RPM	SAE.RPM	SAE.RPM
Value Type	Value (Number)	Value (Number)	Value (Number)
Unit System	Metric	Metric	Metric
Brush			
Color	(0,0,0)	(0,0,0)	(0,0,0)
Opacity	255	255	255
Style	Solid	Solid	Solid
Gradient Style	Radial	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Día y Fecha inicio)

Dashboard Object Properties ->

	<i>Rounded Rectangle 1</i>	<i>Rounded Rectangle 2</i>
Rectangle		
Center point	107,5; 49,5	107,5; 48,25
Size	48; 13	44; 6,5
Rounded Rectangle		
Radius	3	1
Uniform Radius	Yes	yes
Pen		
Width	0,5	0,5
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Transparent	Transparent
Brush		
Color	(37,37,37)	(0,0,0)
Opacity	255	100
Style	Solid	Solid
Gradient Style	Radial	Radial

Text	<i>Text 1</i>
Position	86; 53,75
Horizontal Center	Left
Vertical Center	Center
Format String	Día y Fecha inicio
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

- El contenedor separador usa un rectangle (rectángulo) y un rounded rectangle (rectángulo redondeado).
- Los datos secundarios (container), crea ocho rectángulos que nos servirán para identificar las lecturas, unas de otras.

- Terminada la edición física del medidor, en la sección de layer foreground, vamos a editar los ítems referentes a los medidores numéricos dinámicos y las needles de rango para el contenedor superior y el contenedor de datos secundarios.

En el foreground no se encuentra dividido por secciones como en el background, sino que está asignado para cada medidor, siendo la única excepción el container Button_Panel1 y Button_Panel2 que, como son intercambiables por los botones creados para esta plantilla, se encuentran agrupados, y la parte de needles de cada medidor.

VALORES PARA Container (Economía de Combustible - avg)

Dashboard Object Properties ->

<i>Condition (unidades métricas?)</i>	
Condition 1	
Operator	Equal
Condition Value	l/km
Condition 2	
Logic Operator	None
Operator	Equal
Condition Value	0
OBD-II Value 1	
PID	Fuel economy/consumption
Value Type	Unit/Format Setting (String)
Unit System	Metric
OBD-II Value 1	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

Si la condición es ->

Condición	<i>Verdadera</i>	<i>Falsa</i>
Text	Text	Text 2
Position	-12; 21,25	-12; 21,25
Horizontal Center	Left	Left
Vertical Center	Center	Center
Format String	%.1f	%.1f
Use OBD-II Value	Yes	Yes
OBD-II Value		
PID	CALC.TRIP.AFC.A	CALC.TRIP.AFC.A
Value Type	Value (Number)	Value (Number)
Unit System	Metric	English
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Combustible Consumido)

Dashboard Object Properties ->

<i>Condition (unidades métricas?)</i>	
Condition 1	
Operator	Equal
Condition Value	1
Condition 2	
Logic Operator	None
Operator	Equal
Condition Value	0
OBD-II Value 1	
PID	Volume
Value Type	Unit/Format Setting (String)
Unit System	Metric
OBD-II Value 1	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

Si la condición es ->

Condición	<i>Verdadera</i>	<i>Falsa</i>
Text	Text	Text 2
Position	-12; 35,25	-12; 35,25
Horizontal Center	Left	Left
Vertical Center	Center	Center
Format String	%.1f	%.1f
Use OBD-II Value	Yes	Yes
OBD-II Value		
PID	CALC.TRIP.FUEL_USED.A	CALC.TRIP.FUEL_USED.A
Value Type	Value (Number)	Value (Number)
Unit System	Metric	English
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Costo Combustible)

Dashboard Object Properties ->

<i>Condition (unidades métricas?)</i>	
Condition 1	
Operator	Equal
Condition Value	1
Condition 2	
Logic Operator	None
Operator	Equal
Condition Value	0
OBD-II Value 1	
PID	Volume
Value Type	Unit/Format Setting (String)
Unit System	Metric
OBD-II Value 1	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

Si la condición es ->

Condición	Verdadera	Falsa
Text	Text	Text 2
Position	-12; 49,25	-12; 49,25
Horizontal Center	Left	Left
Vertical Center	Center	Center
Format String	%.2f	%.2f
Use OBD-II Value	Yes	Yes
OBD-II Value		
PID	CALC.TRIP.FUEL_COST.A	CALC.TRIP.FUEL_COST.A
Value Type	Value (Number)	Value (Number)
Unit System	Metric	English
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Distancia Manejada)

Dashboard Object Properties ->

	Condition (unidades métricas?)
Condition 1	
Operator	Equal
Condition Value	km
Condition 2	
Logic Operator	None
Operator	Equal
Condition Value	0
OBD-II Value 1	
PID	Distance
Value Type	Unit/Format Setting (String)
Unit System	Metric
OBD-II Value 1	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

Si la condición es ->

Condición	Verdadera	Falsa
Text	Text	Text 2
Position	37; 21,25	37; 21,25
Horizontal Center	Left	Left
Vertical Center	Center	Center
Format String	%.1f	%.1f
Use OBD-II Value	Yes	Yes
OBD-II Value		
PID	CALC.TRIP.DISTANCE.A	CALC.TRIP.DISTANCE.A
Value Type	Value (Number)	Value (Number)
Unit System	Metric	English
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Tiempo transcuido)

Dashboard Object Properties ->

Text	<i>Text</i>
Position	37; 35,25
Horizontal Center	Left
Vertical Center	Center
Format String	%s
Use OBD-II Value	Yes
OBD-II Value	
PID	CALC.TRIP.TIME.ELAPSED.A
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA Container (Tiempo Manejo)

Dashboard Object Properties ->

Text	<i>Text</i>
Position	37; 49,25
Horizontal Center	Left
Vertical Center	Center
Format String	%s
Use OBD-II Value	Yes
OBD-II Value	
PID	CALC.TRIP.TIME.DRIVE.A
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA Container (Tasa de emisión CO2 - avg)

Dashboard Object Properties ->

<i>Condition (unidades métricas?)</i>	
Condition 1	
Operator	Equal
Condition Value	g/km
Condition 2	
Logic Operator	None
Operator	Equal
Condition Value	0
OBD-II Value 1	
PID	CO2 emission
Value Type	Unit/Format Setting (String)
Unit System	Metric
OBD-II Value 1	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

Si la condición es ->

Condición	Verdadera	Falsa
Text	Text	Text 2
Position	86,25; 21,25	86,25; 21,25
Horizontal Center	Left	Left
Vertical Center	Center	Center
Format String	%.1f	%.1f
Use OBD-II Value	Yes	Yes
OBD-II Value		
PID	CALC.TRIP.CO2.RATE.A	CALC.TRIP.CO2.RATE.A
Value Type	Value (Number)	Value (Number)
Unit System	Metric	English
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Emisión CO2)

Dashboard Object Properties ->

	Condition (unidades métricas?)
Condition 1	
Operator	Equal
Condition Value	g/km
Condition 2	
Logic Operator	None
Operator	Equal
Condition Value	0
OBD-II Value 1	
PID	CO2 emission
Value Type	Unit/Format Setting (String)
Unit System	Metric
OBD-II Value 1	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

Si la condición es ->

Condición	Verdadera	Falsa
Text	Text	Text 2
Position	86,25; 35,25	86,25; 35,25
Horizontal Center	Left	Left
Vertical Center	Center	Center
Format String	%.1f	%.1f
Use OBD-II Value	Yes	Yes
OBD-II Value		
PID	CALC.TRIP.CO2.TOTAL.A	CALC.TRIP.CO2.TOTAL.A
Value Type	Value (Number)	Value (Number)
Unit System	Metric	English
Brush		
Color	(0,0,0)	(0,0,0)
Opacity	255	255
Style	Solid	Solid
Gradient Style	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross

VALORES PARA Container (Día y Fecha inicio)

Dashboard Object Properties ->

Text	<i>Text</i>
Position	86,25; 49,25
Horizontal Center	Left
Vertical Center	Center
Format String	%s
Use OBD-II Value	Yes
OBD-II Value	
PID	CALC.TRIP.TIME.START.A
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(0,0,0)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA Container (Button_Panel1)

Dashboard Object Properties ->

	<i>Rectangle</i>
Rectangle	
Center point	128; 68,25
Size	8; 20,5
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(0,0,0)
Opacity	255
Style	Gradient
Gradient Style	Linear

Text	<i>Text</i>
Position	126; 68,25
Horizontal Center	Center
Vertical Center	Center
Format String	1
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

VALORES PARA Container (Button_Panel2)

Dashboard Object Properties ->

	<i>Rectangle</i>
Rectangle	
Center point	128; 88,75

Size	8; 20,5
Pen	
Width	0,5
Color	(0,0,0)
Opacity	255
Style	Transparent
Brush	
Color	(0,0,0)
Opacity	255
Style	Gradient
Gradient Style	Linear

Text	Text
Position	127; 88,75
Horizontal Center	Center
Vertical Center	Center
Format String	2
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric
Brush	
Color	(255,255,255)
Opacity	255
Style	Solid
Gradient Style	Radial
Hatch Style	Diagonal Cross

- El panel de Datos Secundarios es controlado en esta parte del foreground. Se crearon 2 botones de manejo e intercambio de paneles 1 – 2, cada uno con diferentes medidores de telemetría. El intercambio de automatiza por elemento State Tracker o Cambio de Estado de elementos.
- Para el funcionamiento del panel de Datos Secundarios, el conjunto Medidores de aguja, tiene todos los componentes needle necesarios, con configuración de acuerdo a cada operación lógica del tipo CALC.

VALORES PARA Container <Medidores de aguja> Show_Panel1A

Dashboard Object Properties ->

	CALC.TRIP. FILLUPS.A	CALC.TRIP. FUEL_FLOW. AVG.A	CALC.TRIP. SPEED. AVG_NI.A	CALC.TRIP. SPEED. AVG_WI.A	CALC.TRIP. BOOST.AVG.A
Needle					
Type	Circular	Circular	Circular	Circular	Circular
Range Start	0	0	0	0	0
Range End	100	100	100	100	100
Circular Needle					

Center Point	50; 50	50; 50	50; 50	50; 50	50; 50
Sweep	270	270	270	270	270
Linear Needle					
Type	Offset	Offset	Offset	Offset	Offset
Start Point	20; 40	20; 40	20; 40	20; 40	20; 40
End Point	80; 40	80; 40	80; 40	80; 40	80; 40
Clip Point	50; 40	50; 40	50; 40	50; 40	50; 40
OBD-II Value					
PID	CALC.TRIP. FILLUPS.A	CALC.TRIP. FUEL_FLOW. AVG.A	CALC.TRIP. SPEED. AVG_NI.A	CALC.TRIP. SPEED. AVG_WI.A	CALC.TRIP. BOOST.AVG.A
Value Type	Value (Number)	Value (Number)	Value (Number)	Value (Number)	Value (Number)
Unit System	Metric	Metric	Metric	Metric	Metric

	CALC.TRIP. RPM.AVG.A	CALC.TRIP. ACCEL. FORWARD. MAX.A	CALC.TRIP. POWER.MAX. A	CALC.TRIP. STOPS.A	CALC.TRIP. FUEL_FLOW. MAX.A
--	-------------------------	---	-------------------------------	-----------------------	-----------------------------------

Needle					
Type	Circular	Circular	Circular	Circular	Circular
Range Start	0	0	0	0	0
Range End	100	100	100	100	100
Circular Needle					
Center Point	50; 50	50; 50	50; 50	50; 50	50; 50
Sweep	270	270	270	270	270
Linear Needle					
Type	Offset	Offset	Offset	Offset	Offset
Start Point	20; 40	20; 40	20; 40	20; 40	20; 40
End Point	80; 40	80; 40	80; 40	80; 40	80; 40
Clip Point	50; 40	50; 40	50; 40	50; 40	50; 40
OBD-II Value					
PID	CALC.TRIP. RPM.AVG.A	CALC.TRIP. ACCEL. FORWARD. MAX.A	CALC.TRIP. POWER.MAX. A	CALC.TRIP. STOPS.A	CALC.TRIP. FUEL_FLOW. MAX.A
Value Type	Value (Number)	Value (Number)	Value (Number)	Value (Number)	Value (Number)
Unit System	Metric	Metric	Metric	Metric	Metric

	CALC.TRIP. SPEED.MAX.A	CALC.TRIP. TIME.IDLE.A	CALC.TRIP. BOOST.MAX_ BOOST.A	CALC.TRIP. RPM.MAX.A	CALC.TRIP. ACCEL. BRAKING. MAX.A
--	---------------------------	---------------------------	-------------------------------------	-------------------------	---

Needle					
Type	Circular	Circular	Circular	Circular	Circular
Range Start	0	0	0	0	0
Range End	100	100	100	100	100
Circular Needle					
Center Point	50; 50	50; 50	50; 50	50; 50	50; 50
Sweep	270	270	270	270	270
Linear Needle					
Type	Offset	Offset	Offset	Offset	Offset
Start Point	20; 40	20; 40	20; 40	20; 40	20; 40
End Point	80; 40	80; 40	80; 40	80; 40	80; 40
Clip Point	50; 40	50; 40	50; 40	50; 40	50; 40
OBD-II Value					
PID	CALC.TRIP. RPM.AVG.A	CALC.TRIP. ACCEL. FORWARD. MAX.A	CALC.TRIP. POWER.MAX. A	CALC.TRIP. STOPS.A	CALC.TRIP. FUEL_FLOW. MAX.A
Value Type	Value (Number)	Value (Number)	Value (Number)	Value (Number)	Value (Number)
Unit System	Metric	Metric	Metric	Metric	Metric

VALORES PARA Container <Medidores de aguja> Show_Panel2A

Dashboard Object Properties ->

	CALC.TRIP. GEAR.N. DISTANCE_ PCT.A	CALC.TRIP. GEAR.1. DISTANCE_ PCT.A	CALC.TRIP. GEAR.2. DISTANCE_ PCT.A	CALC.TRIP. GEAR.3. DISTANCE_ PCT.A	CALC.TRIP. GEAR.4. DISTANCE_ PCT.A
Needle					
Type	Circular	Circular	Circular	Circular	Circular
Range Start	0	0	0	0	0
Range End	100	100	100	100	100
Circular Needle					
Center Point	50; 50	50; 50	50; 50	50; 50	50; 50
Sweep	270	270	270	270	270
Linear Needle					
Type	Offset	Offset	Offset	Offset	Offset
Start Point	20; 40	20; 40	20; 40	20; 40	20; 40
End Point	80; 40	80; 40	80; 40	80; 40	80; 40
Clip Point	50; 40	50; 40	50; 40	50; 40	50; 40
OBD-II Value					
PID	CALC.TRIP. GEAR.N. DISTANCE_ PCT.A	CALC.TRIP. GEAR.1. DISTANCE_ PCT.A	CALC.TRIP. GEAR.2. DISTANCE_ PCT.A	CALC.TRIP. GEAR.3. DISTANCE_ PCT.A	CALC.TRIP. GEAR.4. DISTANCE_ PCT.A
Value Type	Value (Number)	Value (Number)	Value (Number)	Value (Number)	Value (Number)
Unit System	Metric	Metric	Metric	Metric	Metric

	CALC.TRIP. GEAR.5. DISTANCE_ PCT.A	CALC.TRIP. GEAR.6. DISTANCE_ PCT.A	CALC.TRIP. GEAR.WG. DISTANCE_ PCT.A	CALC.TRIP. GEAR.N.TIME_ _PCT.A	CALC.TRIP. GEAR.1.TIME_ PCT.A
Needle					
Type	Circular	Circular	Circular	Circular	Circular
Range Start	0	0	0	0	0
Range End	100	100	100	100	100
Circular Needle					
Center Point	50; 50	50; 50	50; 50	50; 50	50; 50
Sweep	270	270	270	270	270
Linear Needle					
Type	Offset	Offset	Offset	Offset	Offset
Start Point	20; 40	20; 40	20; 40	20; 40	20; 40
End Point	80; 40	80; 40	80; 40	80; 40	80; 40
Clip Point	50; 40	50; 40	50; 40	50; 40	50; 40
OBD-II Value					
PID	CALC.TRIP. GEAR.5. DISTANCE_ PCT.A	CALC.TRIP. GEAR.6. DISTANCE_ PCT.A	CALC.TRIP. GEAR.WG. DISTANCE_ PCT.A	CALC.TRIP. GEAR.N.TIME_ PCT.A	CALC.TRIP. GEAR.1.TIME_ PCT.A
Value Type	Value (Number)	Value (Number)	Value (Number)	Value (Number)	Value (Number)
Unit System	Metric	Metric	Metric	Metric	Metric

	CALC.TRIP. GEAR.2.TIME_ PCT.A	CALC.TRIP. GEAR.3.TIME_ PCT.A	CALC.TRIP. GEAR.4.TIME_ PCT.A	CALC.TRIP. GEAR.5.TIME_ PCT.A	CALC.TRIP. GEAR.6.TIME_ PCT.A
Needle					
Type	Circular	Circular	Circular	Circular	Circular
Range Start	0	0	0	0	0
Range End	100	100	100	100	100
Circular Needle					

<i>Center Point</i>	50; 50	50; 50	50; 50	50; 50	50; 50
<i>Sweep</i>	270	270	270	270	270
Linear Needle					
<i>Type</i>	Offset	Offset	Offset	Offset	Offset
<i>Start Point</i>	20; 40	20; 40	20; 40	20; 40	20; 40
<i>End Point</i>	80; 40	80; 40	80; 40	80; 40	80; 40
<i>Clip Point</i>	50; 40	50; 40	50; 40	50; 40	50; 40
OBD-II Value					
<i>PID</i>	CALC.TRIP. RPM.AVG.A	CALC.TRIP. ACCEL. FORWARD. MAX.A	CALC.TRIP. POWER.MAX. A	CALC.TRIP. STOPS.A	CALC.TRIP. FUEL_FLOW. MAX.A
<i>Value Type</i>	Value (Number)	Value (Number)	Value (Number)	Value (Number)	Value (Number)
<i>Unit System</i>	Metric	Metric	Metric	Metric	Metric

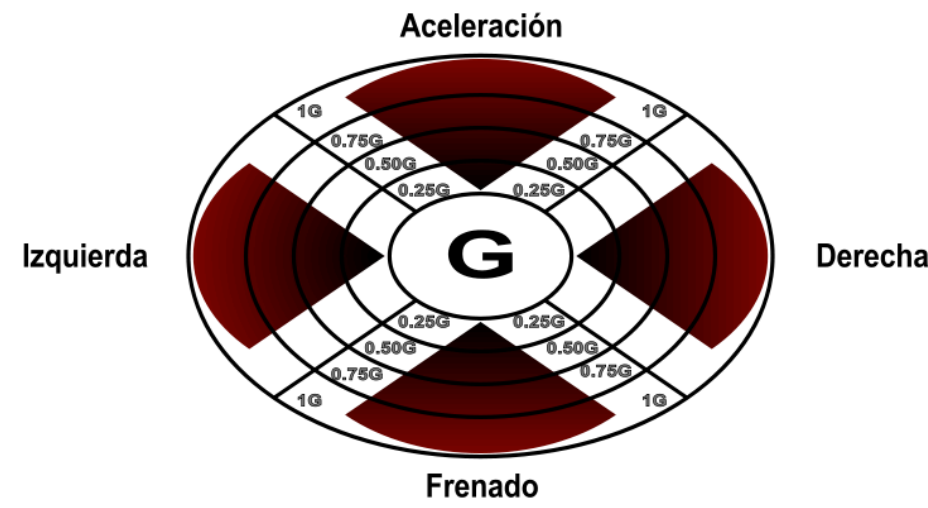
4.3.4 PERSONALIZACIÓN DE LOS INDICADORES TÁCTILES GLCD's

Una innovación dentro de la personalización de dispositivos y elementos de diagnóstico automotriz, es la introducción de indicadores táctiles GLCDs. Un GLCD es un gauge dinámico gráfico (Graphic Liquid Crystal Display) cuya aplicación es la de, en base a programación informática, mostrar en la GUI en tiempo real los valores obtenidos de la ECU del automotor.

En nuestro proyecto se han incluido algunos medidores gauges GLCDs:

- Velocidad Y Rpm
- Aceleración
- Aceleración %
- Aceleraciones en Gs
- Fuerzas G
- Medidor Hev Battery (Automóviles Híbridos)

En esta parte de la personalización de GLCDs, nos vamos a centrar en la creación del GLCD de medición de Fuerzas G y el Medidor Hev Battery, o medidor de batería, complementando con los diseños finales del resto de GLCDs.



Fuente: SkinXL Editor

Fig. 4.135. Diseño de Dashboard GLCD FUERZAS G

DISEÑO MEDIDOR DE FUERZAS G (GLCD)

- Adicionamos un nuevo Dashboard, y lo llamamos Fuerzas G.
(*Dashboard Screen Settings -> Add Dashboard -> Dashboard Settings -> Dashboard Id = Fuerzas G*).
- Partiendo del diseño gráfico deseado para la plantilla, visualizado en la figura Fig. 4.136, debemos seleccionar que objetos incluir en su construcción.

En los ítems Background se crearon cinco grupos de elementos por contenedores (Aceleración, frenado, izquierda y derecha), todos con un objeto de tipo wedge, que nos servirá en la creación del medidor GLCD, y un texto identificador.

*VALORES PARA Container < Aceleración, frenado, izquierda y derecha >
Dashboard Object Properties ->*

<i>Text</i>	<i>Text (Aceleración)</i>	<i>Text (Frenado)</i>	<i>Text (Izquierda)</i>	<i>Text (Derecha)</i>
<i>Position</i>	50; 15	50; 85	15; 50	127; 88,75
<i>Horizontal Center</i>	Center	Center	Center	Center
<i>Vertical Center</i>	Center	Center	Center	Center
<i>Format String</i>	Aceleración	Frenado	Izquierda	Derecha
<i>OBD-II Value</i>				

<i>PID</i>	<i>SAE.RPM</i>	<i>SAE.RPM</i>	<i>SAE.RPM</i>	<i>SAE.RPM</i>
<i>Value Type</i>	<i>Value (Number)</i>	<i>Value (Number)</i>	<i>Value (Number)</i>	<i>Value (Number)</i>
<i>Unit System</i>	<i>Metric</i>	<i>Metric</i>	<i>Metric</i>	<i>Metric</i>
Brush				
<i>Color</i>	<i>(255,255,255)</i>	<i>(255,255,255)</i>	<i>(255,255,255)</i>	<i>(255,255,255)</i>
<i>Opacity</i>	<i>255</i>	<i>255</i>	<i>255</i>	<i>255</i>
<i>Style</i>	<i>Solid</i>	<i>Solid</i>	<i>Solid</i>	<i>Solid</i>
<i>Gradient Style</i>	<i>Radial</i>	<i>Radial</i>	<i>Radial</i>	<i>Radial</i>
<i>Hatch Style</i>	<i>Diagonal Cross</i>	<i>Diagonal Cross</i>	<i>Diagonal Cross</i>	<i>Diagonal Cross</i>

Wedge	<i>Wedge (Aceleración)</i>	<i>Wedge (Frenado)</i>	<i>Wedge (Izquierda)</i>	<i>Wedge (Derecha)</i>
<i>Position</i>	<i>50; 40</i>	<i>50; 60</i>	<i>40; 50</i>	<i>60; 50</i>
<i>Horizontal Radius</i>	<i>20</i>	<i>20</i>	<i>20</i>	<i>20</i>
<i>Vertical Radius</i>	<i>20</i>	<i>20</i>	<i>20</i>	<i>20</i>
<i>Wedge Angle</i>	<i>225</i>	<i>45</i>	<i>135</i>	<i>-45</i>
<i>Wedge Sweep</i>	<i>90</i>	<i>90</i>	<i>90</i>	<i>90</i>
Brush				
<i>Color</i>	<i>(255,255,255)</i>	<i>(255,255,255)</i>	<i>(255,255,255)</i>	<i>(255,255,255)</i>
<i>Opacity</i>	<i>255</i>	<i>255</i>	<i>255</i>	<i>255</i>
<i>Style</i>	<i>Solid</i>	<i>Solid</i>	<i>Solid</i>	<i>Solid</i>
<i>Gradient Style</i>	<i>Radial</i>	<i>Radial</i>	<i>Radial</i>	<i>Radial</i>
<i>Hatch Style</i>	<i>Diagonal Cross</i>	<i>Diagonal Cross</i>	<i>Diagonal Cross</i>	<i>Diagonal Cross</i>

- Para formar la figura, y los topes de medición del acelerómetro incluido, se hizo una división respectiva del wedge, en proporción a la medición lógica del GLCD, con la finalidad de crear zonas de medición más fáciles de interpretar. Estas hechas por cinco elipses concéntricas, 1 contenedor Tick Marks y 4 containers de texto por escala.
- En la parte del foreground, la diagramación de un GLCD es distinta a la conformación del resto de medidores, por cuanto aquí ya se introducen elementos incorporados en el dispositivo en sí. En este caso en especial el uso del acelerómetro y del inclinómetro, son de participación activa en la plantilla.
 - Los medidores son directamente proporcionales a los wedges creados anteriormente, razón por la cual la medición de los wedges son dependientes entre sí. Por ejemplo si el automóvil está en aceleración, el medidor de frenado no marcaría ningún valor, de igual manera en cuestión de viraje.

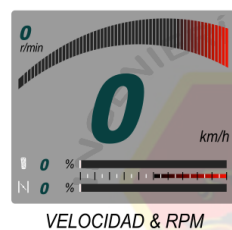
VALORES PARA Container < Aceleración, frenado, izquierda y derecha >

Wedge	Wedge (Aceleración)	Wedge (Frenado)	Wedge (Izquierda)	Wedge (Derecha)
Position	50; 40	50; 60	40; 50	60; 50
Horizontal Radius	20	20	20	20
Vertical Radius	20	20	20	20
Wedge Angle	225	45	135	-45
Wedge Sweep	90	90	90	90
Brush				
Color	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)
Opacity	255	255	255	255
Style	Solid	Solid	Solid	Solid
Gradient Style	Radial	Radial	Radial	Radial
Hatch Style	Diagonal Cross	Diagonal Cross	Diagonal Cross	Diagonal Cross

	Wedge (Aceleración)	Wedge (Frenado)	Wedge (Izquierda)	Wedge (Derecha)
Needle				
Type	Linear	Linear	Linear	Linear
Range Start	0	-1	-1	0
Range End	1	0	0	1
Circular Needle				
Center Point	50; 50	50; 50	50; 50	50; 50
Sweep	270	270	270	270
Linear Needle				
Type	Offset	Offset	Offset	Offset
Start Point	0; 20	0; 20	0; 20	0; 20
End Point	0; 0	0; 0	0; 0	0; 0
Clip Point	50; 40	50; 40	50; 40	50; 40
OBD-II Value				
PID	AUX.ACCEL. FORWARD	AUX.ACCEL. FORWARD	AUX.ACCEL. LATERAL	AUX.ACCEL. LATERAL
Value Type	Value (Number)	Value (Number)	Value (Number)	Value (Number)
Unit System	Metric	Metric	Metric	Metric

DISEÑO MEDIDOR DE VELOCIDAD, RPM Y ACELERACIÓN (GLCD)

Los últimos instrumentos por ser introducidos en nuestro proyecto son los GLCDs correspondientes a medidores de Velocidad, RPM y Aceleración.



Fuente: SkinXL Editor

Fig. 4.136. Diseño de Dashboard GLCD VELOCIDAD y RPM

- En el Dashboard anteriormente creado Lector Principal, debemos seleccionar que objetos incluir en la construcción de los medidores.

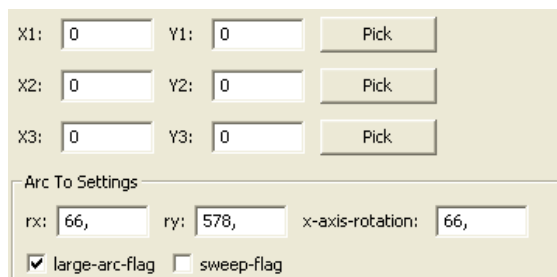
En los ítems Background se creo un container (Gauge Velocidad RPM), el mismo que contiene dos elementos texto con las unidades (km/h y r/min).

VALORES PARA Container < Gauge Velocidad RPM >

Dashboard Object Properties ->

Text	<i>Text (km/h)</i>	<i>Text (r/min)</i>
<i>Position</i>	50; 49	-10,5; 26,5
<i>Horizontal Center</i>	Right	Left
<i>Vertical Center</i>	Center	Center
<i>Format String</i>	km/h	r/min
OBD-II Value		
<i>PID</i>	SAE.RPM	SAE.RPM
<i>Value Type</i>	Value (Number)	Value (Number)
<i>Unit System</i>	Metric	Metric
Brush		
<i>Color</i>	(255,255,255)	(255,255,255)
<i>Opacity</i>	255	255
<i>Style</i>	Solid	Solid
<i>Gradient Style</i>	Radial	Radial
<i>Hatch Style</i>	Diagonal Cross	Diagonal Cross

- En la parte del foreground, medidor GLCD de velocidad y rpm se diferencia del medidor GLCD de Fuerzas G en razón que en vez de utilizar un elemento wedge para la medición, utiliza un caso especial del elemento Path para la forma gráfica dinámica de medición.



Fuente: SkinXL Editor

Fig. 4.137. Diseño elemento Path GLCD Velocidad

La diagramación del elemento Path se encuentra descrita, hablando en lenguaje informático, en el Anexo B, por cuanto su programación es

demasiado extensa como para explicarla en breves rasgos. Su automatización se produce por un needle (medidor aguja) así:

VALORES PARA Container < Gauge Velocidad RPM >

<i>Needle</i>	
Needle	
Type	Linear
Range Start	0
Range End	5000
Circular Needle	
Center Point	50; 50
Sweep	270
Linear Needle	
Type	Offset
Start Point	-11; 28
End Point	50; 28
Clip Point	-11; 28
OBD-II Value	
PID	SAE.RPM
Value Type	Value (Number)
Unit System	Metric

Como aspecto final de la personalización de elementos dentro de la IPA, se crearon elementos como un medidor de aceleración, un medidor de Inst. MPG, Avg MPG, Distancia, medidor de Poder, Torque y Aceleraciones en Gs, para completar las plantillas gráficamente.

CAPÍTULO 5

CONSTRUCCIÓN Y PRUEBAS DE FUNCIONAMIENTO DEL SISTEMA DE DIAGNÓSTICO PREDICTIVO.

5.1 CONSTRUCCION DEL SISTEMA DE RED

El sistema de red WI-FI entre el dispositivo táctil Apple y el generador de señal inalámbrica se conecta mediante la opción WI-FI en la sección de ajustes generales del Tablet siguiendo estos sencillos pasos:

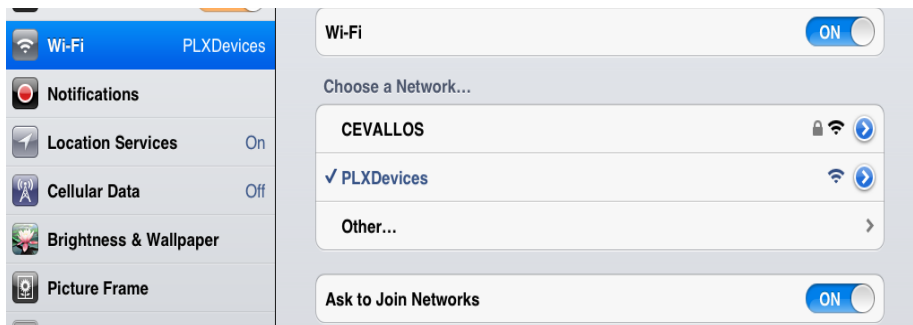
- Conectar la interfaz generadora de señal WI-FI a un puerto OBD-II compatible con los protocolos de información descritos anteriormente. Activar la generación de señal inalámbrica con el switch ubicado a un costado de la carcasa.



Fuente: Grupo de investigación.

Fig. 5.1. Ubicación de puerto OBD-II.

- En la opción WI-FI ubicada en la configuración “Ajustes” del Ipad, activar la pestaña WI-FI. En “Selecciones una red...” damos click en PLXDevices que es nombre del compilador del generador de señal y entramos a la configuración de la red en la flecha de color azul junto al nombre de la red.



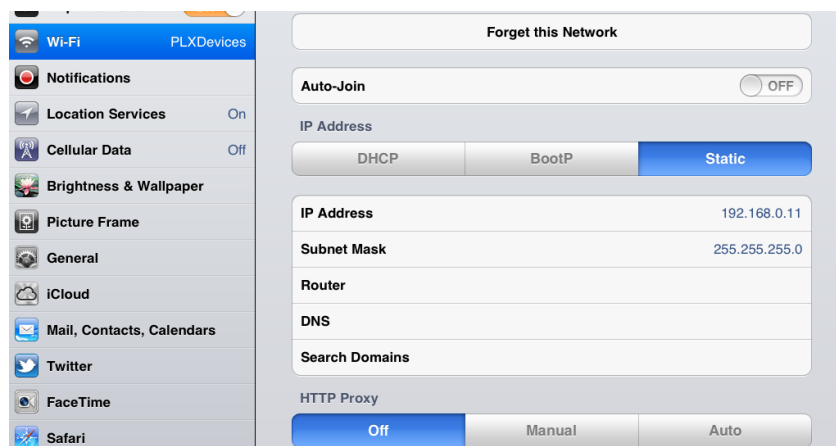
Fuente: Ipad 2 General Settings.

Fig. 5.2. Conexión WI-FI con el generador de señal.

- En la opción “Estática” de “Dirección IP”, ponemos los valores de conexión diseñados de acuerdo a parámetros de seguridad y fiabilidad de señal:

Dirección IP: 192.168.0.11

Máscara de Subred: 255.255.255.0



Fuente: Ipad 2 General Settings.

Fig. 5.3. Configuración de Dirección IP y Máscara de Subnet

- En ocasiones es necesario ingresar la información del puerto que se requiere establecer para mayor seguridad en la conexión:

Port: 35000

5.1.1 PARÁMETROS DE SEGURIDAD DE EQUIPOS INFORMÁTICOS

La seguridad en el envío de la información, el manejo privado del dispositivo y la no filtración de estos datos a otros dispositivos es primordial para la confiabilidad de la IPA en los usuarios potenciales.

En razón de esta búsqueda de protección informática se siguen los siguientes parámetros de seguridad:

- Codificar la información, con lo que se especifica una ruta de acceso única para cada dispositivo conectado.
- Vigilancia de red, en el caso de tablets de la marca Apple en plano secundario, ya que al estar construido bajo sistema Unix, ninguna aplicación de terceros puede realizar alguna acción en la red sin permisos de Administrador.
- Tecnologías repelentes o protectoras, como lo hace un cortafuegos.
- Sistema de Respaldo Remoto o Copia de Seguridad, obtenida mediante Itunes en la opción de "Crear Copia de Seguridad".
- Consideraciones de software, tener instalado en la máquina únicamente el software necesario reduce riesgos. Así mismo tener controlado el software asegura la calidad de la procedencia del mismo (el software obtenido de forma ilegal o sin garantías aumenta los riesgos). En todo caso un inventario de software proporciona un método correcto de asegurar la reinstalación en caso de desastre. El software con métodos de instalación rápidos facilita también la reinstalación en caso de contingencia.

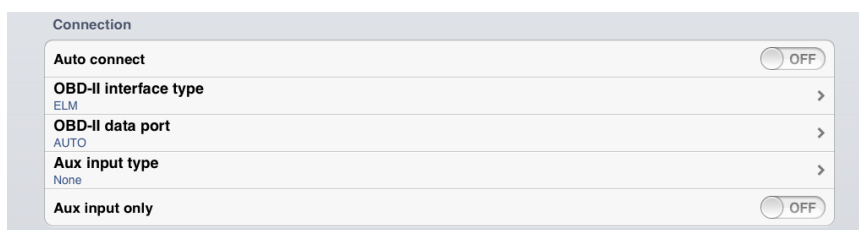
5.1.2 INTERCONEXIÓN DE DISPOSITIVOS DE RED

Lograr la comunicación inalámbrica entre el dispositivo táctil y el generador de señal WI-FI, no indica una conexión final entre la IPA y la ECU del automóvil.

Para que la IPA pueda recibir y enviar datos a la computadora central del automóvil, dentro de la IPA DashCommand v.2.9.1 (Build 2435), aplicación bajo derechos de autor de la marca Palmer Performance Engineering y publicada dentro de la App Store de Apple, se configura:

- En “Settings” → “Connection” tenemos:

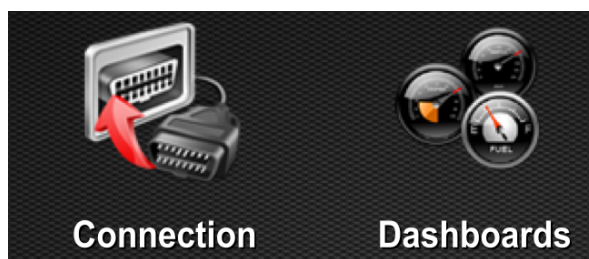
Auto connect	No
OBD-II interface type	ELM
OBD-II data port	AUTO (por default)
Aux input type	None
Aux input only	No



Fuente: DashCommand General Settings.

Fig. 5.4. Configuración del panel Connection

- En el menú principal nos dirigimos a el ícono Connection, con el cuál la IPA procede a realizar la petición de host por parte del generador y la autenticación tanto de configuración como de lista de PIDs disponibles para ese modelo de automóvil.

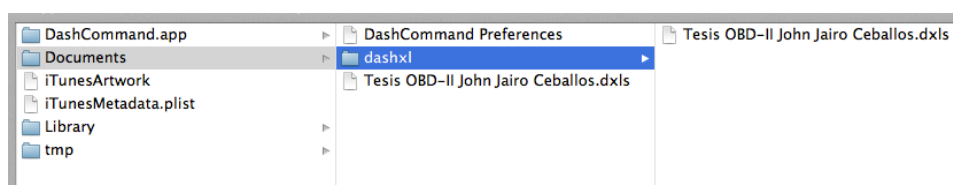


Fuente: DashCommand App.

Fig. 5.5. Iconos de conexión e ingreso a Dashboards

5.1.3 INSTALACIÓN Y CONEXIÓN DE DISPOSITIVOS GLCDS

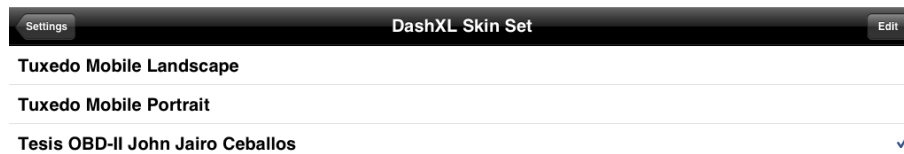
- Los dashboards diseñados para este proyecto (Lector Principal, Gauges GLCDs, Medición y Telemetría, Lectores Digitales, Fuerzas G) mediante el software DashXL se compilan dentro de un solo archivo con formato .dxls, donde además se encuentra toda la información de análisis de PIDs.
- Para hacer uso de este archivo diseñado, mediante comunicación SSH con el Ipad, se translada el archivo a la carpeta dashxl dentro del directorio Documents en DashCommand.app.



Fuente: FunBox SSH.

Fig. 5.6. Conexión SSH con el Directorio DashCommand.app

- Una vez colocado el dxls, en DashCommand “Settings” → “Dashboards” → “Default Dashboard” → seleccionamos el archivo “Tesis Diagnóstico OBD-II John Ceballos”.



Fuente: DashCommand General Settings.

Fig. 5.7. Ubicación archivo .dxls

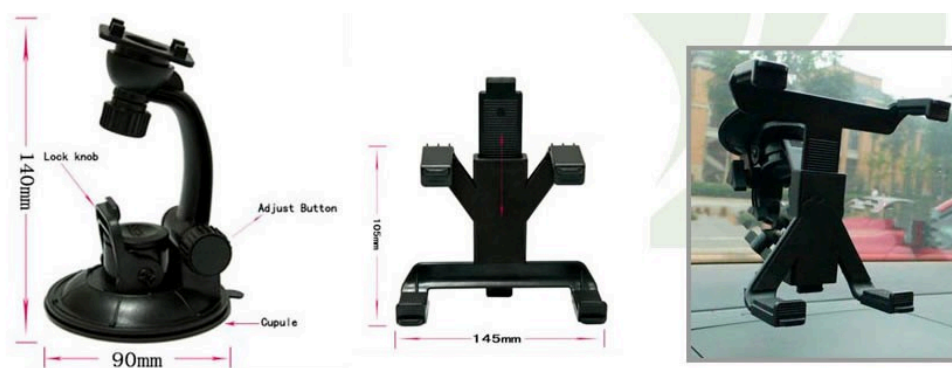
- Para la visualización final de nuestro diseño, en el menú principal damos click en “Dashboards”, adentrándonos posteriormente a main menú del proyecto.

5.1.4 MONTAJE DE DISPOSITIVO TÁCTIL APPLE

Para el montaje del dispositivo táctil Apple, se siguieron primero estos parámetros:

- Ubicación visible del dispositivo.
- Seguridad y soporte en pruebas de carretera.
- Fijación estable a parabrisas o tablero delantero del automóvil.
- Universalidad en la ubicación para cualquier tipo de diseño.

Tomando en cuenta estos puntos de fabricación, se seleccionó un soporte de material plástico, con pestañas de sujeción a la medida y sujetador de vacío al parabrisas. El diseño de este soporte se detalla a continuación:



Fuente: Mercado Libre Ecuador.

Fig. 5.8. Diseño Soporte Tablet Parabrisas

En primera instancia se tomo como posibilidad la ubicación manual del dispositivo, pero en pruebas de manejo o en pruebas en que se requiere una mayor estabilidad del dispositivo (medidor de viraje), esta opción no era viable.

5.2 PRUEBAS DE VERIFICACIÓN DE CONECTORES Y CABLEADO

5.2.1 MÉTODOS DE VERIFICACIÓN

Constatar del excelente estado de comunicación, sobre todo en ocasiones en que se obtienen datos erróneos, se realiza con las siguientes comprobaciones:

- El estado de los pines del conector en el generador de señal OBD-II inalámbrica debe ser impecable. No se deben presentar pines desviados o doblados, ni la presencia de pines rotos o fracturados. Se debe priorizar el estado de los pines de acuerdo al protocolo de información, como se indica:

ISO 9141-2:

Pin 7 (K-line) + (opcional) pin 15 (L-line, inicialización).

ISO 14230-4 (KeyWord Protocol 2000):

Pin 7.

ISO 11519-4 (SAE J1850 PWM), con 2 conexiones:

Pin 2 (Bus+) + pin 10 (Bus-).

ISO 11519-4: (SAE J1850 VPW), con 1 conexión:

Pin 2 (Bus+).

ISO15765-4 /ISO 11898 SAE J2480 (CAN):

Pin 6 CAN_H i + pin 14 CAN_Low

- El momento de activar la generación de señal WI-FI, se debe encender una luz que prenda todo el dispositivo generador, si no ocurre este fenómeno podría ser por falta de energía en el Conector OBD-II, o por daño en la fuente del generador.
- Verificar el nivel de señal obtenido en el los del Ipad, señalado por un ícono de red inalámbrica con 3 figuras que se prenderán o apagarán de acuerdo a:

75%-100% señal: 3 líneas de señal.

50%-75% señal: 2 líneas de señal.

menos del 50% señal: 1 línea de señal.

- En la IPA DashCommand, al momento de ingresar a “Connection” debe pasar tanto la petición como la autenticación de conexión con el servidor OBDNet y obtener la lista de PIDs disponibles por la ECU.

5.2.2 ANÁLISIS COMPROBACIÓN DE CIRCUITOS DE AUTOMÓVILES.

Antes de realizar una comprobación de la lectura de circuitos electrónicos dentro de los sistemas del automóvil, la IPA obtiene la lista de PIDs disponibles por parte de la ECU. El apartado “Data Logging”, mediante las funciones informáticas de enlace, nos indica qué PIDs están disponibles y no disponibles con vistos y equis respectivamente.

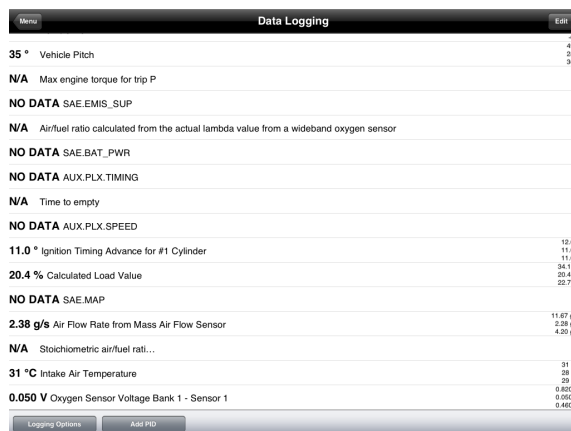
Por defecto contiene las siguientes peticiones:

- SAE.RPM
- SAE.VSS
- Fordward Acceleration
- Lateral Acceleration
- Vehicle Roll

- Vehicle Pitch
- CALC.TRIP.TORQUE.MAX.P
- SAE.EMIS_SUP
- CALC.AFR_ACTUAL
- SAE.BAT_PWR
- AUX.P LX.TIMING
- CALC.TTE
- AUX.P LX.SPEED
- SAE.SPARKADV
- SAE.LOAD_PCT
- SAE.MAP
- SAE.MAF

Si el usuario requiriera un PID en especial, en “Data Logging” se encuentra un botón “Add PID”, el mismo que despliega todos los PIDs disponibles y no disponibles por categorías, como lo explicamos en las listas de prioridad SDK.

En casos de que en el data logging aparezca la leyenda “NO DATA” antes de cada PID, indica que la IPA indica que ese PID no está disponible en la lista de PIDs, o no se ha conectado adecuadamente con la OBDNet, procediendo a la revisión de la conexión posteriormente a este factor.



Fuente: DashCommand General Settings.

Fig. 5.9. Comprobación de conexión por Data Logging

5.3 PRUEBAS DE FUNCIONAMIENTO.

5.3.1 PRUEBAS DE COMUNICACIÓN Y CONECTIVIDAD.

Como en todo software basado en lenguaje de programación, en este caso Objective-C, siempre existe una consola de depuración. En este caso no hay excepción, la IPA tiene un apartado de “Console” con un registro de todas las acciones que se llevan a cabo y la hora en que se efectuaron.

Esta consola de depuración, además consta con un limpiador de acciones y un configurador de opciones de depuración, dentro del cual se activa el Modo de Depuración que viene desactivado por defecto, y se puede enviar un e-mail a el compilador con la finalidad que cualquier error dentro de nuestra compilación sea tomada en cuenta para reparos próximos incluídos en actualizaciones del software de la App.



```
[16:54:11.573] Warning: Not scanning unsupported parameter: SAE.LAMBDA
[16:54:11.577] Warning: Not scanning unsupported parameter: SAE.FLI
[16:54:11.577] Warning: Not scanning unsupported parameter: SAE.AAT
[16:54:11.578] Warning: Not scanning unsupported parameter: SAE.BARO
[16:54:12.576] CONST.TIRE_RESISTANCE: Fault: 'Tire rolling resistance coefficient' setting is undefined.
[16:54:12.576] CONST.FUEL_DENSITY: Fault: 'Fuel type' setting is undefined - Fuel density cannot be determined.
[16:54:12.576] CONST.FUEL_TANK_CAPACITY: Fault: 'Fuel tank capacity' setting is undefined.
[16:54:12.577] CONST.FUEL_CO2_CONTENT: Fault: 'Fuel type' setting is undefined - CO2 content cannot be determined.
[16:54:12.577] CONST.SPEED_CORRECTION: Fault: 'Vehicle speed correction factor' setting is undefined.
[16:54:12.577] CONST.STOICHIOMETRIC_AIR_FUEL_RATIO: Fault: 'Fuel type' setting is undefined - stoichiometric air/fuel ratio cannot be determined.
[16:54:12.577] CONST.ENGINE_DISPLACEMENT: Fault: 'Engine displacement' setting is undefined - engine displacement cannot be determined.
[16:54:12.578] CONST.CURB_WEIGHT: Fault: 'Curb weight' setting is undefined - curb weight cannot be determined.
[16:54:12.578] CONST.ADDITIONAL_WEIGHT: Fault: 'Additional weight' setting is undefined - additional weight cannot be determined.
[16:54:12.578] CONST.DRAG_COEFFICIENT: Fault: 'Drag coefficient' setting is undefined.
[16:54:12.578] CONST.FRONTAL_AREA: Fault: 'Frontal area' setting is undefined - frontal area cannot be determined.
[16:54:12.579] CONST.MAF_PID: Warning: 'MAF PID' setting is undefined or invalid - using CALC.MAF_B.
[16:54:12.579] CONST.MIN_RPM: 'Minimum engine speed' setting is undefined.
[16:54:12.580] CONST.VOLUMETRIC_EFFICIENCY: Fault: 'Volumetric efficiency' setting is undefined.
[16:54:12.580] CONST.FINAL_DRIVE_RATIOS: Fault: 'Final drive ratios' setting is undefined.
[16:54:12.580] CONST.WHEEL_CIRCUMFERENCE: Warning: 'Wheel circumference' setting is undefined - attempting to calculate wheel circumference from 'Tire size specification' setting.
[16:54:12.580] CONST.WHEEL_CIRCUMFERENCE: Fault: 'Tire size specification' setting undefined - cannot calculate a wheel circumference
[16:54:12.580] CONST.LAMBDA_ACTUAL_PID: Warning: 'Lambda (actual) PID' setting is undefined - scanning for suitable PID.
[16:54:12.581] CONST.LAMBDA_ACTUAL_PID: Fault: 'Lambda (actual) PID' setting is undefined - no other Lambda PID found.
[16:54:12.581] CALC.AIR_DENSITY: Warning: default barometric pressure value used, accuracy is compromised.
[16:54:12.582] CALC.MAF_B: Fault: Engine displacement vehicle setting is unavailable - cannot calculate mass air flow (MAF).
[16:54:12.582] CALC.MAF_B: Warning: Volumetric efficiency not specified - using 75%
[16:54:12.583] CALC.FUEL_FLOW: Warning: Air/Fuel ratio unknown, defaulting to the AFR for gasoline.
[16:54:12.583] CALC.FUEL_FLOW: Warning: Fuel density unknown, defaulting to the fuel density of gasoline.
[16:54:12.583] CALC.VSS_C: Warning: 'Vehicle speed correction factor' is undefined - Using default = 1
[16:54:12.596] CALC.FC.IFC: Fault: Fuel flow is unknown - instantaneous fuel consumption cannot be calculated
```

Fuente: DashCommand General Settings.

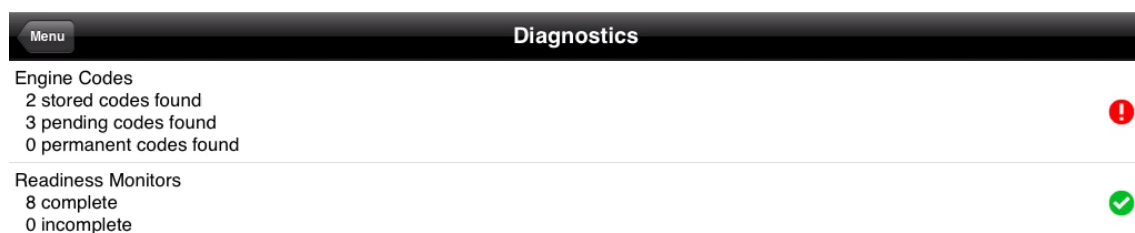
Fig. 5.10. Consola de Depuración de la Compilación.

5.3.2 EXPLORACIÓN DE CÓDIGOS DE AVERÍA Y ESTADO DE VARIABLES PIDS.

Al ser este un proyecto de diagnóstico predictivo del automóvil, directamente tiene que ver con el diseño de un apartado correctamente a este fin. Es en base a esta lógica que la IPA DashCommand tiene una sección exclusiva al diagnóstico del automóvil.

En “Diagnostics”, la IPA mediante el uso de funciones y callbacks, realiza dos gestiones de verificación:

- Engine codes (Códigos de Motor).
- Readiness Monitors (Monitoreos continuos).



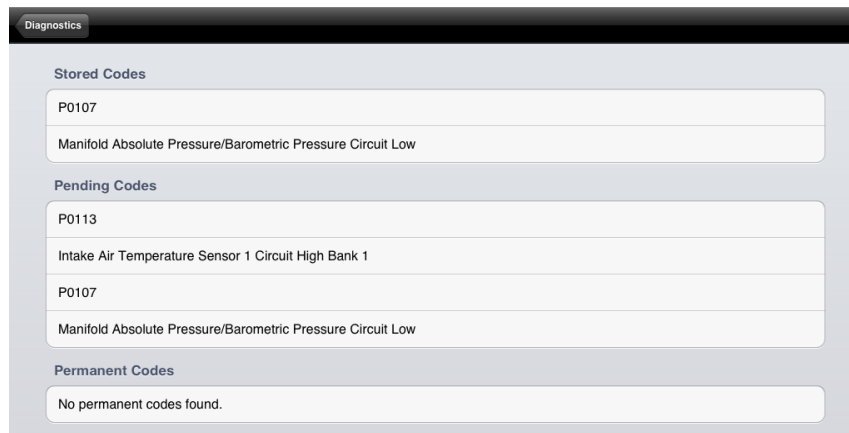
Fuente: DashCommand General Settings.

Fig. 5.11. Engines Codes y Readiness Monitors

En cuanto a Engines codes, como en cualquier sistema de escaneo de DTCs, la ECU si encuentra fallas crea códigos de tipo almacenados, pendientes y permanentes, estos se diferencian en base al nivel de afección que producen estos fallos dentro del correcto funcionamiento del sistema.

Todos los códigos que se lean de la memoria de la ECU, mediante la conexión con la OBDNet, están basados en la norma internacional de codificación OBD-II en cuanto a nombres, simbología y numeración clasificatoria.

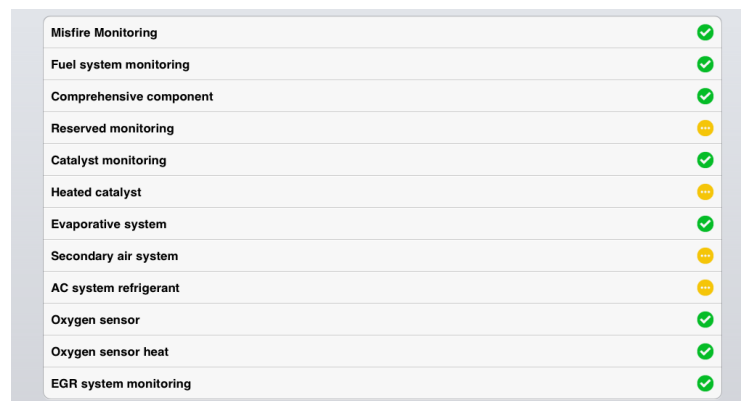
Es así que, como se puede observar en la siguiente figura, al leerse DTCs el dispositivo las clasifica.



Fuente: DashCommand General Settings.

Fig. 5.12. DTCs de tipo almacenados, pendientes y permanentes.

Esta lectura de DTCs se la realizo en un Chevrolet Aveo Activo 1.8cc extrayendo los sensores de MAP y de lectura de IAT, por lo que el automóvil automáticamente al no recibir esta información procede a crear el código de fallo, con el correspondiente encendido de la luz de Check Engine.



Fuente: DashCommand General Settings.

Fig. 5.13. Monitoreos activos de sistemas.

Ahora, es primordial virtualizar no tan sólo individualmente los componentes de sensores y actuadores, sino monitorear los sistemas en conjunto, por lo que “Readiness Monitors” es la función que nos lleva al análisis completo de sistemas. Para el monitoreo se debe priorizar y tomar

en cuenta la meta de la norma OBD-II que es el cuidado del medio ambiente y la reducción de la contaminación ambiental.

El sistema realiza los siguientes monitoreos:

- Misfire Monitoring (Monitoreo de Chispa Perdida)
- Fuel System Monitoring (Monitoreo de Sistema de Combustible)
- Comprehensive Component (Monitoreo de Componentes)
- Reserved Monitoring (Monitoreo)
- Catalyst Monitoring (Monitoreo del Catalizador)
- Heated Catalyst (Calentador del Catalizador)
- Evaporative System (Monitoreo Sistema integrado Evaporación)
- Second Air System (Sistema Inyección Aire Secundario)
- AC system refrigerant (Refrigerante Sistema Aire Acondicionado)
- Oxigen Sensor (Sensor de Oxígeno)
- Oxigen sensor heat (Sensor de oxígeno calefaccionado)
- EGR System Monitoring (Monitoreo sistema de recirculación de gases de escape EGR)

La simbología utilizada por la IPA para señalar si se pasó el monitoreo es un visto en un fondo verde, si el monitoreo no se puede realizar por falta de componentes en el sistema un circulo amarillo con tres punto blancos, y si el monitoreo no pasó la verificación un circulo con fondo rojo con una señal de advertencia.

5.3.3 MEDICIÓN DE MAGNITUDES EN FUNCIONAMIENTO MECÁNICO Y ELECTRÓNICO NORMAL DE AUTOMÓVILES

Con la finalidad de demostrar la universalidad en cuanto al manejo de protocolos, en este capítulo se proceden a realizar las pruebas de medición de magnitudes en tres sistemas diferentes, un sistema americano, europeo y asiático de automóviles. Es por eso que para la lectura de magnitudes se escogieron los siguientes automóviles:

- Chevrolet Optra 1.8L T/M Design 2007 Sedán



Fuente: Grupo de investigación.
Fig. 5.14. Chevrolet Optra 1.8L T/M Design 2007 Sedán.

- Mazda 3 1.6L T/M 2009



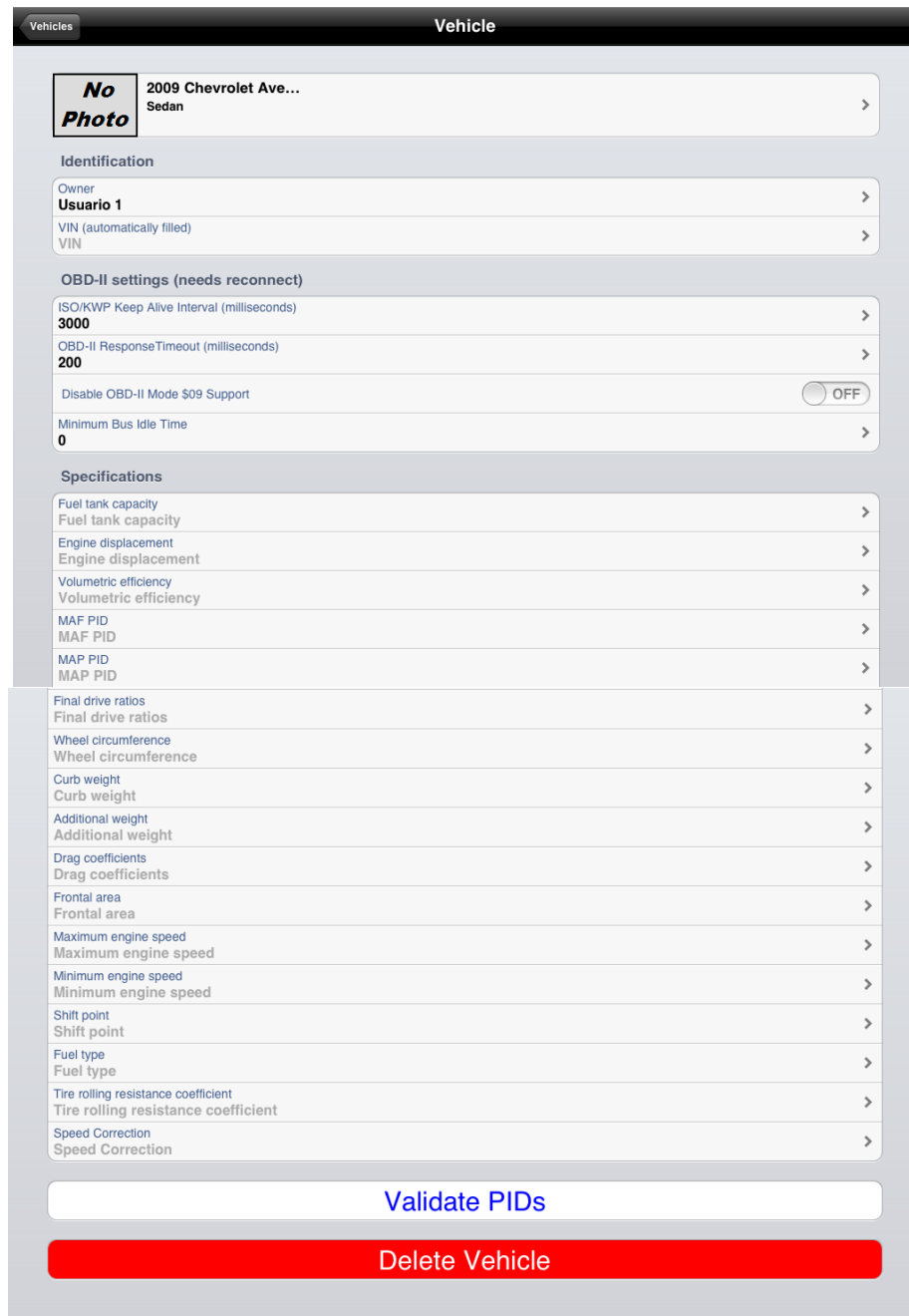
Fuente: Grupo de investigación.
Fig. 5.15. Mazda 3 1.6L T/M 2009

- Peugeot 206 1.4L T/M 2006



Fuente: Grupo de investigación.
Fig. 5.16. Peugeot 206 1.4L T/M 2006

De cada automóvil, primero se crea una base de datos en nuestro dispositivo, la misma que nos indicará los PIDs disponibles y la VIN (número de identificación del vehículo), como se indica en la siguiente figura.



Fuente: DashCommand General Settings.
Fig. 5.17. Hoja de información del Vehículo.

La inclusión de esta base de datos, nos sirve para separar los análisis tomados de cada automóvil, así como para saber los componentes incluidos dentro de la tecnología de control comandada por la ECU. Es indispensable su manejo adecuado, siendo posible hasta el uso de imágenes del estado físico del automotor en todos los casos. Cabe recalcar que cada uno de los PIDs dentro de esta hoja puede ser modificado en cuanto a las unidades que se desean manejar, habiendo la posibilidad del manejo de unidades del Sistema Métrico y del Sistema Inglés.

Posteriormente a este proceso se conecta con la interfaz, en la sección del menú principal de la IPA "Connection", con lo cual el dispositivo táctil procederá a realizar los request, que no son más que peticiones informáticas de la lista de PIDs, el VIN y los DTCs almacenados dentro de la memoria.



Fuente: DashCommand General Settings.

Fig. 5.18. Request de la apertura de comunicación OBD-II.

Una vez aceptadas todas las request por parte del servidor OBDNet, nuestro dispositivo ya recibe la señal de las PIDs inalámbricamente, tanto para nuestro lector principal, como para los gauges GLCDs, medición y telemetría, economía poder flujo de aire y medidor de fuerzas G.

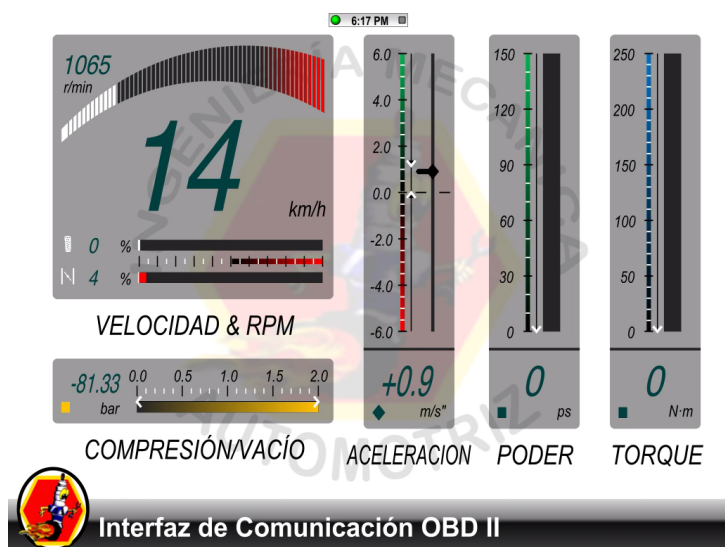
Ahora, en función de la comprobación de los medidores, se realizan dos tipos de pruebas, una prueba con automóvil estacionado y una prueba de manejo para conocer el estado en función de la telemetría del generador. En cada caso se realizan ciertas mediciones, de la siguientes forma:

Tabla 5.1. MEDICIONES DE ESTADO PIDs

MEDICION	INMÓVIL	PRUEBA MANEJO
RPM	Si	Si
APP	Si	Si
TPS	Si	Si
COMPRESIÓN	Si	Si
ACELERACIÓN	No	Si
PODER	No	Si
TORQUE	No	Si
VELOCIDAD	No	Si
ACELERACIONES GS	No	Si
CONS. INSTANTANEO	No	Si
CONS. PROMEDIO	No	Si
FLUJO COMBUSTIBLE	Si	Si
ECT	Si	Si
IAT	Si	Si
VOLTAGE	Si	Si
MAF	Si	Si
MAP	Si	Si

Fuente: Grupo de investigación.

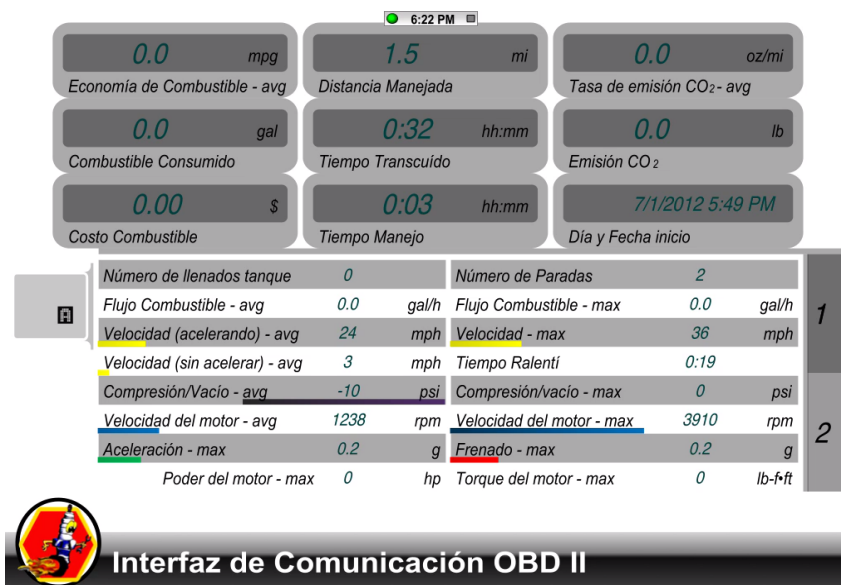
Las medidas tomadas en nuestros dashboards, finalmente son comprobadas con las medidas obtenidas en el data logging para mayor eficacia y veracidad en los resultados expresados en nuestro proyecto.



Fuente: DashXL Editor.

Fig. 5.19. Lector Principal en funcionamiento.

En la figura anterior, se observa la correspondencia entre las revoluciones del motor, con la apertura de la mariposa de aceleración y la consiguiente velocidad que se logra a la salida a las llantas. De igual forma se logra observar una medida de lectura de compresión/vacío que en este caso, por ser un valor negativo corresponde a una medida de vacío.



Fuente: DashXL Editor.

Fig. 5.20. Lectores de Telemetría creando base de datos.

En el caso del medidor de telemetría, mantiene un registro en función del tiempo de las estadísticas obtenidas del performance del automóvil en pruebas de carretera, así como sigue un cálculo determinado de la distancia recorrida en función de la velocidad del automóvil y el tiempo transcurrido.

5.4 VERIFICACIÓN Y TABULACIÓN DE DATOS

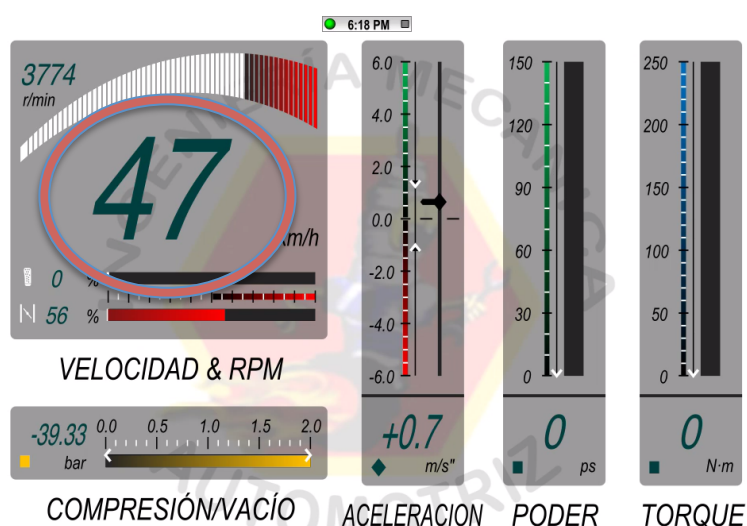
De los procesos de obtención de datos expresados en el apartado anterior, obtenemos la siguiente información de cada una de las mediciones diseñadas dentro de los grupos de dashboards, tomando como punto en común que todas estas mediciones se hicieron a una velocidad del motor de 3000 rpm:

Velocidad

Tabla 5.2. TABLA DE MEDICIÓN DE VELOCIDAD

Velocidad (km/h)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	0 km/h	0 km/h	0 km/h
Prueba de manejo	1era – 15 km/h	1era – 19 km/h	1era – 13 km/h
	2da – 24 km/h	2da – 36 km/h	2da – 30 km/h
	3ra – 54 km/h	3ra – 76 km/h	3ra – 57 km/h
	4ta – 67 km/h	4ta – 89 km/h	4ta – 72 km/h

Fuente: Grupo de investigación.



Fuente: DashCommand Dashboards.

Fig. 5.21. Medición de velocidad en Prueba de Manejo (Chevrolet Optra 1.8L T/M Design)

Sensor de Posición del Pedal del Acelerador

Tabla 5.3. TABLA DE MEDICIÓN DE SENSOR APP

APP (%)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	NA	15%	NA
Prueba de manejo	NA	19%	NA

Fuente: Grupo de investigación.



Fuente: DashCommand Dashboards.

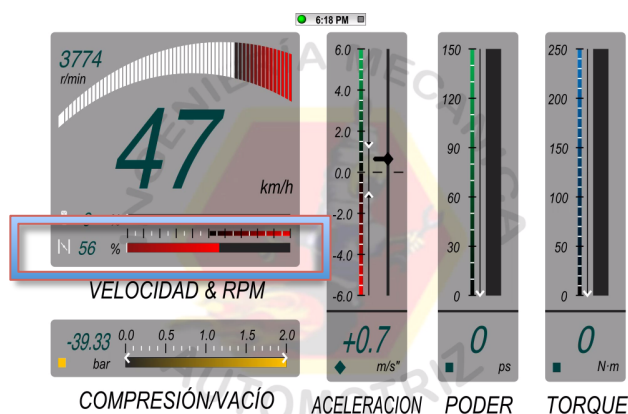
Fig. 5.22. Medición de Sensor APP Prueba Estacionado (Mazda 3 1.6L T/M 2009)

Sensor de Posición de la mariposa de Aceleración

Tabla 5.4. TABLA DE MEDICIÓN DE SENSOR TPS

TPS (%)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	13%	27%	16%
Prueba de manejo	56%	35%	48%

Fuente: Grupo de investigación.



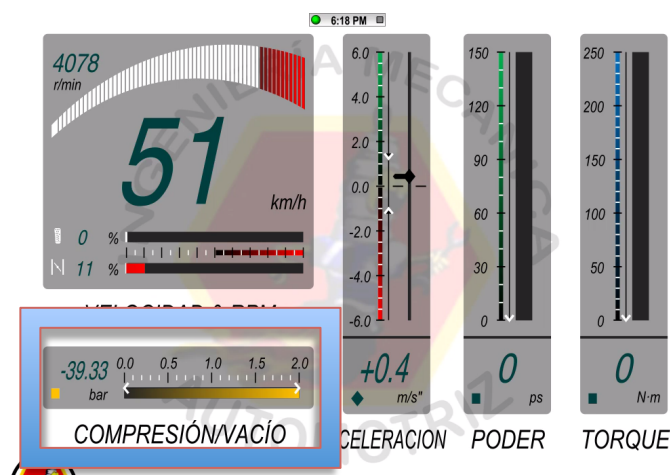
Fuente: DashCommand Dashboards.

Fig. 5.23. Medición de Sensor TPS Prueba de Manejo (Chevrolet Optra 1.8L T/M Design)

Compresión y Vacío

Tabla 5.5. TABLA DE MEDICIÓN DE COMPRESIÓN Y VACÍO

Compresión/Vacío (kPa)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	- 78.33 kPa	- 74.22 kPa	-73.42 kPa
Prueba de manejo	Aceleración - 40.24 kPa	Aceleración - 42.26 kPa	Aceleración - 41.65 kPa
	Desaceleración - 85.32 kPa	Desaceleración - 86.21 kPa	Desaceleración - 82.34 kPa



Fuente: DashCommand Dashboards.

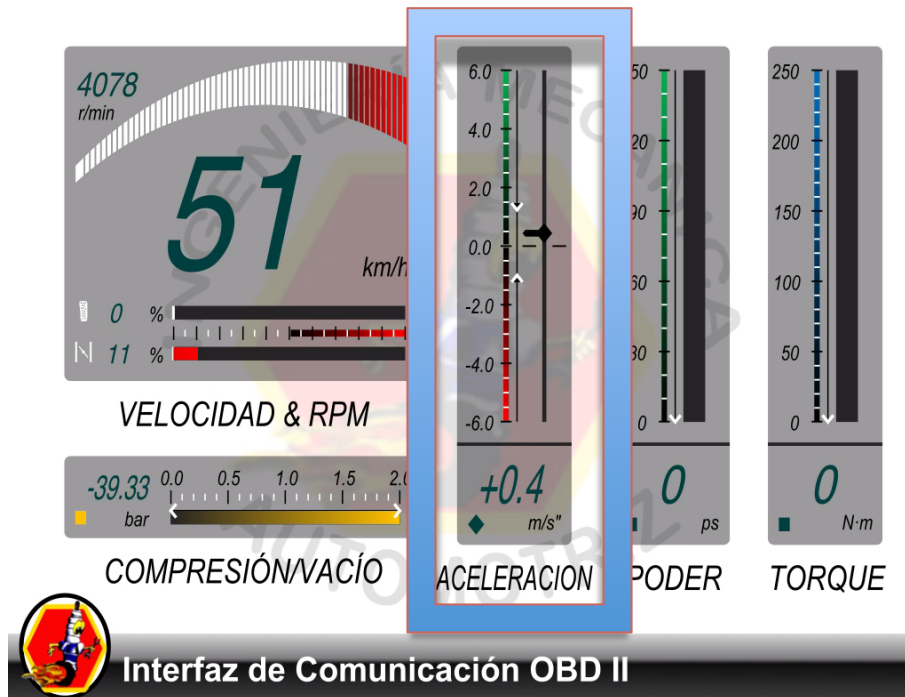
Fig. 5.24. Medición de Compresión/Vacío en Prueba de Manejo (Chevrolet Optra 1.8L T/M Design)

Aceleración (En base a acelerómetro integrado)

Tabla 5.6. TABLA DE MEDICIÓN DE ACELERACIÓN

Aceleración (m/s ²)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	0 m/s ²	0 m/s ²	0 m/s ²
Prueba de manejo	Aceleración 1.4 m/s ²	Aceleración 2.1 m/s ²	Aceleración 1.6 m/s ²
	Desaceleración - 1.8 m/s ²	Desaceleración - 2.6 m/s ²	Desaceleración - 2.2 m/s ²

Fuente: Grupo de investigación.



Fuente: DashCommand Dashboards.

Fig. 5.25. Medición de Aceleración en Prueba de Manejo (Chevrolet Optra 1.8L T/M Design)

Poder

Tabla 5.7. TABLA DE MEDICIÓN DE PODER

Poder (ps)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	NA	NA	NA
Prueba de manejo	NA	NA	NA

Fuente: Grupo de investigación.

Torque

Tabla 5.8. TABLA DE MEDICIÓN DE TORQUE

Torque (N.m)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	NA	NA	NA
Prueba de manejo	NA	NA	NA

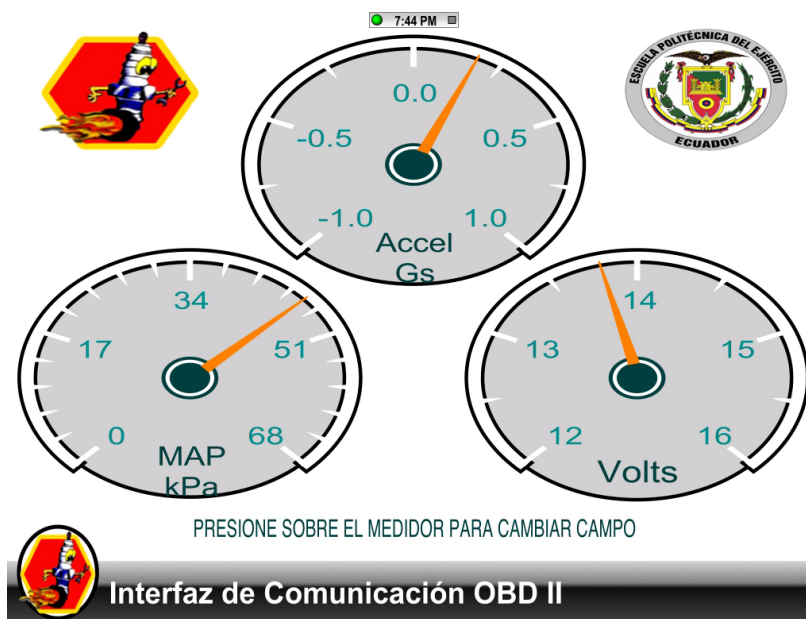
Fuente: Grupo de investigación.

Aceleraciones en Gs

Tabla 5.9. TABLA DE MEDICIÓN DE ACELERACIÓN EN Gs

Aceleración (Gs)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	0 Gs	0 Gs	0 Gs
Prueba de manejo	-0,5 < Gs < 0.6	-0,6 < Gs < 0.6	-0,5 < Gs < 0.5

Fuente: Grupo de investigación.



Fuente: DashCommand Dashboards.

**Fig. 5.26. Medición de Aceleraciones en Gs
(Chevrolet Optra 1.8L T/M Design)**

Consumo instantáneo de Combustible

Tabla 5.10. TABLA DE CONSUMO INSTANTÁNEO DE COMBUSTIBLE

Inst. l/100km	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	NA	NA	NA
Prueba de manejo	NA	NA	NA

Fuente: Grupo de investigación.

Consumo promedio de Combustible

Tabla 5.11. TABLA DE MEDICIÓN DE CONSUMO PROMEDIO DE COMBUSTIBLE

Avg (l/100km)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	NA	NA	NA
Prueba de manejo	NA	NA	NA

Fuente: Grupo de investigación.

Flujo de Combustible

Tabla 5.12. TABLA DE MEDICIÓN DE FLUJO DE COMBUSTIBLE

Flujo Comb (l/h)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	NA	NA	NA
Prueba de manejo	NA	NA	NA

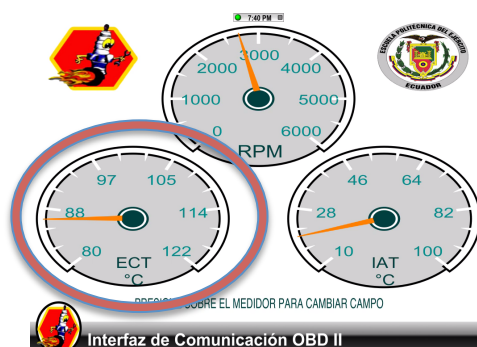
Fuente: Grupo de investigación.

Temperatura del refrigerante del motor

Tabla 5.13. TABLA DE MEDICIÓN DE SENSOR ECT

ECT (°C)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	40°C (frío)	30°C(frío)	35°C(frío)
Prueba de manejo	86°C(estable)	82°C(estable)	88°C(estable)

Fuente: Grupo de investigación.



Fuente: DashCommand Dashboards.

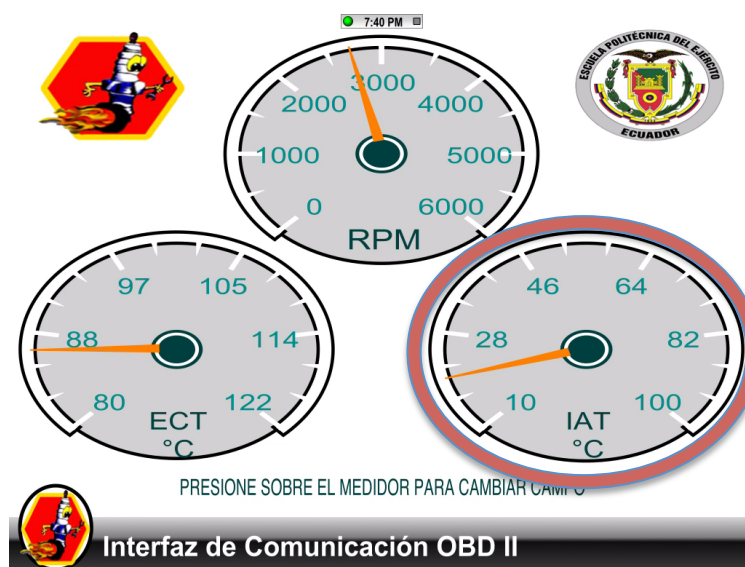
Fig. 5.27. Medición de Sensor ECT (Engine Coolant Temperature) (Chevrolet Optra 1.8L T/M Design)

Temperatura del Aire en admisión

Tabla 5.14. TABLA DE MEDICIÓN DE SENSOR IAT

IAT (°C)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	21°C	21°C	22°C
Prueba de manejo	23°C	22°C	23°C

Fuente: Grupo de investigación.



Fuente: DashCommand Dashboards.

**Fig. 5.28. Medición de Sensor IAT (Intake Air Temperature)
(Chevrolet Optra 1.8L T/M Design)**

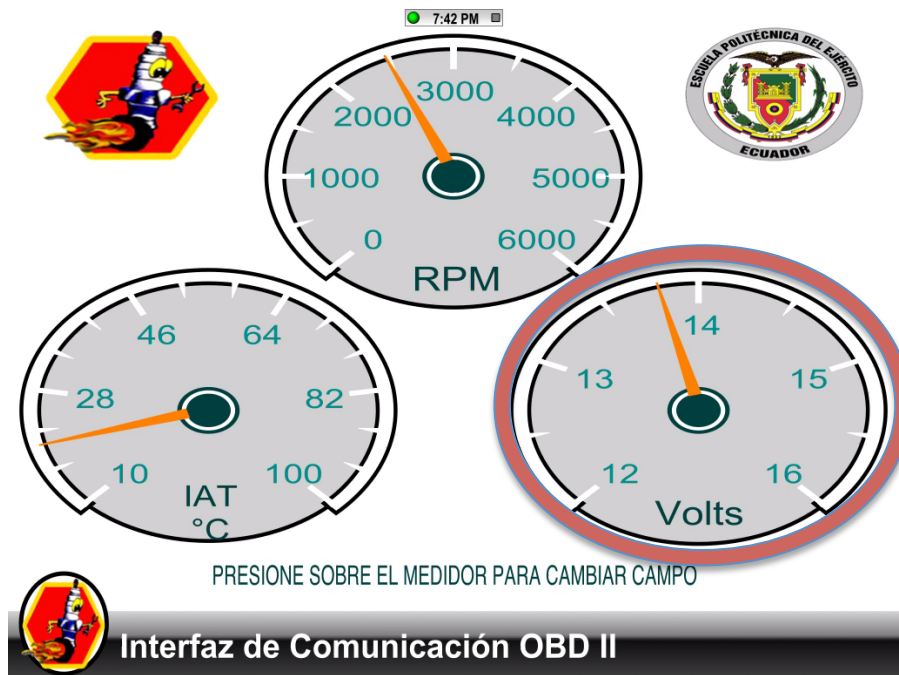
Voltaje de la Batería

(Referencial al voltaje de entrega en el puerto OBD-II)

Tabla 5.15. TABLA DE MEDICIÓN DE VOLTAJE DE LA BATERÍA

Volts (V)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	14V	13,6V	12,9V
Prueba de manejo	13,2V	13,1V	13,0V

Fuente: Grupo de investigación.



Interfaz de Comunicación OBD II

Fuente: DashCommand Dashboards.

Fig. 5.29. Medición de Voltaje (Referencial)
(Chevrolet Optra 1.8L T/M Design)

Poder en HPs

Tabla 5.16. TABLA DE MEDICIÓN DE PODER EN HPs

Wheel (HP)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	NA	NA	NA
Prueba de manejo	NA	NA	NA

Fuente: Grupo de investigación.

Máximo Poder en HPs

Tabla 5.17. TABLA DE MEDICIÓN DE MAXIMO PODER

Max (HP)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	NA	NA	NA
Prueba de manejo	NA	NA	NA

Fuente: Grupo de investigación.

Máximo Torque de Salida

Tabla 5.18. TABLA DE MEDICIÓN DE TORQUE DE SALIDA

Velocidad (km/h)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	NA	NA	NA
Prueba de manejo	NA	NA	NA

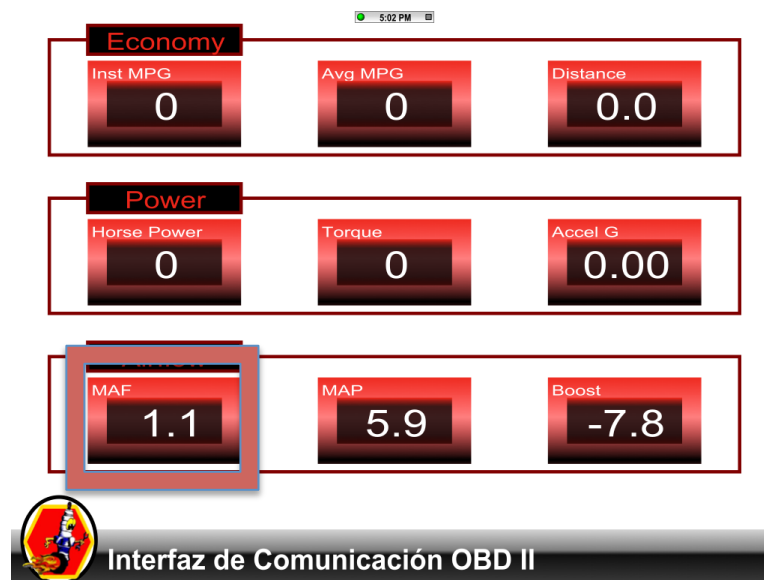
Fuente: Grupo de investigación.

MAF

Tabla 5.19. TABLA DE MEDICIÓN DE MAF

MAF (lbs/min)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	NA	1.1 lbs/min	NA
Prueba de manejo	NA	1.7 lbs/min	NA

Fuente: Grupo de investigación.



Fuente: DashCommand Dashboards.

**Fig. 5.30. Medición de MAF
(Mazda 3 1.6L T/M 2009)**

MAP

Tabla 5.20. TABLA DE MEDICIÓN DE MAP

MAP (kPa)	Chevrolet Optra 1.8L T/M Design	Mazda 3 1.6L T/M 2009	Peugeot 206 1.4L T/M 2006
Estacionado	26 kPa	16,93 kPa	20,3 kPa
Prueba de manejo	17 - 34 kPa (acel. Leves)	18 - 30 kPa (acel. Leves)	17 - 28 kPa (acel. Leves)
	17 - 70 kPa (acel. Bruscas)	16 - 65 kPa (acel. Bruscas)	16 - 76 kPa (acel. Bruscas)

Fuente: Grupo de investigación.



Fuente: DashCommand Dashboards.

**Fig. 5.31. Medición de MAP en prueba de manejo
(Chevrolet Optra 1.8L T/M Design)**

5.5 ANÁLISIS DE RESULTADOS

Tomando todos los datos obtenidos mediante nuestro proyecto, se realiza un análisis detenido de como se comportan los componentes individualmente, en primera instancia, para posteriormente realizar el estudio de correspondencia entre componentes dentro de un mismo sistema.

Es así que, comenzando por la lectura de velocidad, en pruebas de manejo es referencial al trabajo del TPS y las revoluciones que se den al motor, como se puede observar en la tabla 5.2, viéndose una mínima diferencia entre cada automóvil, debiéndose este hecho al comportamiento de la caja manual y la relación de compresión que se maneje en cada automóvil.

En referencia al sensor APP, en los automóviles Chevrolet Optra y Peugeot 206, no obtenemos datos de este sensor por cuanto aún utilizan aceleradores de tipo cable de alambre, mientras que el Mazda ya posee un acelerador con sensor de este tipo, que en porcentaje señala cuanto presionamos el acelerador, y es proporcional al control del TPS en este caso en su correcto funcionamiento, con diferencia de porcentajes en pruebas estacionado y de manejo. (Tabla 5.3).

El sensor TPS, tiene una apertura menor en el caso del Chevrolet Optra, subiendo proporcionalmente en aumento de revoluciones, mientras que el Mazda tiene una apertura mayor porcentualmente en las pruebas estacionadas, esto nos lleva a una deducción en cuanto al consumo de combustible que realiza cada motor en un ciclo de trabajo, llegando a la conclusión que en cuanto a economía de combustible (Medidor telemétrico), el Chevrolet Optra es un 15% más económico que el Mazda 3.

La medida de Compresión-Vacío es referencial, ya que para comprobar el correcto estado de compresión individual en cada cilindro, se necesita hacer una prueba manual de compresión y vacío del motor. Esta medida hace referencia a el comportamiento de carga Boost que provee el vehículo, que en los tres casos es vacío por darnos una medida negativa. La tabla 5.5 nos da una clara idea del comportamiento de carga que provee el motor, habiendo una diferencia mínima entre los tres casos estudiados.

En la tabla 5.6 se hace una medición de aceleración, que no es necesariamente la aceleración del motor, sino la aceleración que sufre la carrocería de sus centro de gravedad, por lo cual varía entre 6 m/s^2 y -6

m/s". Esta medida es tomada del comportamiento del acelerómetro incorporado en el dispositivo táctil, siendo positiva en casos de aceleración, y negativa en el caso contrario. Esta medida de aceleración también puede ser señalada en Gs como unidades, para el registro de fuerzas Gs que sufre el automóvil, conductor y ocupantes (Tabla 5.9).

Los medidores referentes a Poder, Torque, Consumo instantáneo, consumo promedio y flujo de combustible, no obtuvieron datos de la ECU en los tres casos en razón de que, al ser automóviles en versiones estándar, no incluyeron características que solamente son necesarias en automotores de alto performance. De la misma manera la mayoría de estas medidas son estudiadas en ECUs que manejan cajas de transmisión automática, sobre todo el poder y torque que el sistema provee a las llantas.

El análisis predictivo de sensores de temperatura (ECT e IAT), son fundamentales en casos de sobrecalentamiento del motor, de daños en el sistema de refrigeración, en el comportamiento erróneo de componentes como es el caso del termostato, y en casos que el ventilador se enciende irregularmente o permanece encendido en todo el funcionamiento del sistema. Es así que en la tabla 5.3 el sensor ECT señala cuán frío puede llegar a estar el motor antes de encenderlo, en pruebas estacionado, lo que nos ayuda a comprender casos de encendido en frío y problemas de encendido por esta razón, llegando a una temperatura estable de 82°C a 86°C.

La tabla 5.14 nos señala con exactitud la medida del sensor IAT, indispensable para el correcto funcionamiento del sistema motor en cambios de ambientes de temperatura, por ejemplo, un automóvil se comporta de diferente manera en ambientes templados, que en ambientes donde la temperatura del aire es mucho mayor. Este sensor ayuda a la ECU a hacer las respectivas correcciones en cuanto a la dosificación de combustible y otros factores que implican en la combustión.

El medidor de Voltaje de la Batería, es tomado en este caso de la cantidad de voltaje que llega al puerto OBDII, y va disminuyendo en cuanto mayor cantidad de componentes se vayan utilizando dentro del habitáculo. La medida no debe reducir por debajo de los 12 Volts, y señala el buen o mal estado del alternador en cada ciclo de trabajo, y si el sistema de carga está realizando correctamente su labor. En pruebas estacionadas, o antes del encendido se puede conocer, en ciertos casos, el nivel de carga de la batería, y si está en buen estado o si necesita mantenimiento.

Finalmente, los sensores indispensables del análisis dentro de nuestro proyecto son el sensor MAF y el sensor MAP. En el caso del sensor MAF, solo se encuentra presente en el vehículo Mazda 3, aumentando en exactitud las medidas de elementos influyentes dentro de la combustión, como es el caso del aire. El Sensor MAF nos señala que ingresan aproximadamente 1.1 libras de aire por minuto en la admisión. Ahora en nuestro proyecto, se utilizan dos formas de análisis de MAF, una directa de la lista de PIDs de la ECU, y la segunda forma es mediante cálculo por método CALC.MAF_B como se explicó en la Fig. 4.38.

El sensor MAP, viene de la lectura directa del sistema sin pasar por correcciones o scripts de regulación, si en el diseño de tomo la OBDII value SAE.MAP, pero como también fue en el caso del MAF, puede ser estudiado de dos maneras adicionales que son el CALC.MAP_A (Fig. 4.39), que involucra los valores de SAE.MAF, SAE.IAT, SAE.RPM, Eficiencia volumétrica y el desplazamiento del motor, o mediante el CALC.MAP_B (Fig 4.40) que involucra en cambio al CALC.MAF_A.

La utilización de los métodos CALC, nos abre una compuerta a la creación de sensores virtuales de medición es decir, aún cuando estos componentes no se encuentren físicamente dentro del sistema, pueden ser tomados mediante la formulación de otros componentes que intervienen en su trabajo.

Como datos adicionales, los medidores de telemetría llevan un registro de los valores tomados por nuestro medidores anteriores, pudiendo ser comparados posteriormente para mejorar la performance del automóvil, como se lo indica en su respectiva plantilla dentro del dispositivo táctil.

CAPÍTULO 6

MARCO ADMINISTRATIVO

6.1. RECURSOS:

Con el fin de desarrollar este proyecto de forma planificada, se plantea el presente capítulo con el cual se analizará el aspecto técnico-operativo del mismo para así comprender todo aquello que tenga relación con el funcionamiento y la operatividad del propio proyecto de aquí que los recursos tanto humanos, tecnológicos y materiales son fundamentales para la puesta en marcha.

La planificación tiene como objetivo la optimización de recursos, para lo que es necesario diseñar un plan y asegurarse que se cumpla.

6.1.1. RECURSOS HUMANOS:

Para realizar el proyecto de tesis titulado **“DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE DIAGNÓSTICO PREDICTIVO PARA AUTOMOTORES BAJO NORMA OBD II, CON CONEXIÓN INALÁMBRICA WI-FI Y CONTROLADORES TÁCTILES DE LA MARCA APPLE”**, lo más importante constituyó el proceso, tanto de diseño, programación, diagramación y corrección de problemas hecho por su servidor, John Jairo Ceballos, que desempeñó el papel de investigador, programador y en cierto modo de autoeducador. Así también se contó el asesoramiento del Ing. Germán Erazo designado como Director, quién con su extensa experiencia en el campo automotriz, promovió la investigación científica y la puesta en marcha del proyecto. Además hubo el apoyo tecnológico de la Ing. Alexandra Corral, que en calidad de Codirectora colaboró y guió toda la parte tecnológica innovativa incluida en el desarrollo de este proyecto.

6.1.2. RECURSOS TECNOLÓGICOS:

La innovación tecnológica, su desarrollo y aplicación expuestas en este trabajo, es un gran aporte como recurso tecnológico para proyectos de investigación futuros. La incursión en la programación Objective-C, cuya exploración es nueva en el campo automotriz, fue de gran ayuda para poder lograr los objetivos antes planteados.

No hay que olvidar que cualquier recurso es válido, como es el caso de herramientas de compilación de IPAs SDK, Cocoa-Lab, y demás compilaciones textuales informáticas, más aún cuando se pretende desarrollar un proyecto de investigación como es el caso, pues se necesita recopilar información que muchas veces se muestra irrelevante, pero para catalogarla se encuentran los investigadores.

6.1.3. RECURSOS MATERIALES:

Los recursos materiales corresponden a todos los elementos físicos que fueron necesarios para la materialización del proyecto y estos son: laptop Apple MacBook Pro, Ipad 2, conector OBD-II, lector generador de puerto OBD-II con señal WI-FI, soportes, scanner, osciloscopio y herramientas de diagnóstico.

6.2. PRESUPUESTO:

Para cumplir con la meta prevista, a continuación se muestra valores económicos junto con su asignación, que sirvieron como control financiero del proyecto, al mismo tiempo que generará una idea de la inversión que se realizó.

**Tabla 6.1.
Presupuesto.**

ORDEN	DETALLE	TOTAL USD
1	Generador WI-FI	500,00
2	Ipad 2	1000,00
3	Soporte Ipad	100,00
4	Protector Ipad	50,00
5	Smart Cover	70,00
6	Programación	500,00
7	Actualización	40,00
8	Construcción	100,00
TOTAL USD		2360,00

Fuente: Grupo de investigación.

6.3. FINANCIAMIENTO:

El financiamiento se presentó por parte del realizador del proyecto: John Jairo Ceballos Marcillo.

6.4 CRONOGRAMA

El cronograma utilizado para este proyecto de innovación y aplicación tecnológica se adjunta en el Anexo C.

CONCLUSIONES

Al concluir el presente trabajo de investigación, ponemos a consideración las siguientes conclusiones y recomendaciones, con la finalidad de que este material esté a disposición general como fuente de consulta.

Se diseñó y construyó un Sistema de Diagnóstico Predictivo Universal, con la utilización de un sistema de programación informática de uso nativo de la marca Apple, permitiendo optimizar la utilización de recursos y la creación de una nueva herramienta de análisis para el profesional automotriz.

El lenguaje de programación Objective-C, nativo en la creación de IPAs, fomenta la reducción en el uso de memoria y recursos informáticos en cuanto a recepción de señales OBD-II se refiere.

La OBDNet ya no es de estudio exclusivo de fabricantes y productores escáneres comerciales, ya que en este proyecto se señala su estructura de datos y los procesos que realiza.

La opción de personalización de herramientas de medición, GLCDs, y diagnóstico predictivo preventivo, ayuda a la comprensión final de la interacción de componentes electrónicos, y la prioridad por defecto dentro de cada sistema.

Un ambiente inalámbrico de estudio de PIDs y DTCs, es necesario para la reducción en tiempos y costos de implementos físicos de estudio.

Este proyecto fomenta la inducción en la creación de NUI y GUI, en cualquier campo de aplicación automotriz, siendo este campo de acción de gran desarrollo potencial actualmente.

RECOMENDACIONES.

Comprender el lenguaje Objective-C desde nivel principiante, para que en el proceso de diagramación y programación no existan “huecos” estructurales que no permitan la compilación de la IPA.

Estudiar los diagramas de bloques SDK y las tablas de prioridad SDK detenidamente antes de personalizar las herramientas de diagnóstico.

Establecer en el software de compilación Visual Basic el tipo de unidad que se va a usar, métrica o inglesa.

Realizar correctamente la configuración de conexión entre la Tablet y el generador ya que, como ocurrió en el desarrollo de este proyecto, si no es configurada correctamente existe mal interpretación de los datos o conexión nula, perdiendo seguridad en el sistema.

Comprobar la ruta de acceso y la MAC Adress de cada dispositivo, sobre todo en casos de aplicación en competición, para que la tasa de bits en telemetría no se desvíe o se filtre en otros dispositivos desconocidos.

BIBLIOGRAFÍA

LIBROS:

Actron (2004). OBD-II AutoScanner. Estados Unidos: Actron USA

Altenberg, B. (2008). Become an Xcoder. Estados Unidos: CocoaLab

Berzal, F. (2008). Interfaces gráficas de usuario. España: Independiente

Euro Systems (2006). OBD-II Specifications and Connections. Estados Unidos: Euro Systems Automotive Training Inc.

Hernández, J (2006). Circuitos electrónicos auxiliares del vehículo. Chile: Fundación Universidad de Atacama.

Krueger, N. (2008). Implementación de RFID activa y WI-FI para la excelencia operativa en el automóvil. Alemania: Magazine

MacProgramadores (2008). El lenguaje Objective-C para programadores C++ y Java. España: MacProgramadores.org

Marrero, C. (2006). Interfaz Gráfica de usuario. España: Universidad de la Laguna.

Molina, J (2003). Especificación de interfaz de usuario. España: Universidad Politécnica de Valencia.

Morales, C (2010). Scanner Automotriz Interfaz PC. México: UPN.

Pérez, J. (2005). Comunicaciones móviles e inalámbricas. España:
Gaptel

Pinto, J. (2007). Arquitectura de redes y servicios. España: Uniovi.

ANEXO A: CODIGOS OBD II

4.3.5.1.1.1. POWER TRAIN ERROR CODES

P0010...."A" Camshaft Position Actuator circuit (Bank 1)
P0011...."A" Camshaft Position - Timing over-advanced (Bank 1)
P0012...."A" Camshaft Position - Timing over-retarded (Bank 1)
P0013...."B" Camshaft Position Actuator circuit (Bank 2)
P0014...."B" Camshaft Position - Timing over-advanced (Bank 1)
P0015...."B" Camshaft Position - Timing over-retarded (Bank 1)
P0020...."A" Camshaft Position Actuator circuit (Bank 2)
P0021...."A" Camshaft Position - Timing over-advanced (Bank 2)
P0022...."A" Camshaft Position - Timing over-retarded (Bank 2)
P0023...."B" Camshaft Position Actuator circuit (Bank 2)
P0024...."B" Camshaft Position - Timing over-advanced (Bank 2)
P0025...."B" Camshaft Position - Timing over-retarded (Bank 2)
P0030....O2 Sensor heater control circuit (Bank 1 Sensor 1)
P0031....O2 Sensor heater control circuit low (Bank 1 Sensor 1)
P0032....O2 Sensor heater control circuit high (Bank 1 Sensor 1)
P0033....Turbocharger bypass valve control circuit
P0034....Turbocharger bypass valve control circuit low
P0035....Turbocharger bypass valve control circuit high
P0036....O2 Sensor heater control circuit (Bank 1 Sensor 2)
P0037....O2 Sensor heater control circuit low (Bank 1 Sensor 2)
P0038....O2 Sensor heater control circuit high (Bank 1 Sensor 2)
P0042....O2 Sensor heater control circuit (Bank 1 Sensor 3)
P0043....O2 Sensor heater control circuit low (Bank 1 Sensor 3)
P0044....O2 Sensor heater control circuit high (Bank 1 Sensor 3)
P0050....O2 Sensor heater control circuit (Bank 2 Sensor 1)
P0051....O2 Sensor heater control circuit low (Bank 2 Sensor 1)
P0052....O2 Sensor heater control circuit high (Bank 2 Sensor 1)
P0056....O2 Sensor heater control circuit (Bank 2 Sensor 2)
P0057....O2 Sensor heater control circuit low (Bank 2 Sensor 2)
P0058....O2 Sensor heater control circuit high (Bank 2 Sensor 2)
P0062....O2 Sensor heater control circuit (Bank 2 Sensor 3)
P0063....O2 Sensor heater control circuit low (Bank 2 Sensor 3)
P0064....O2 Sensor heater control circuit high (Bank 2 Sensor 3)
P0065....Air Assisted Injector control range/performance problem
P0066....Air Assisted Injector control circuit low
P0067....Air Assisted Injector control circuit high
P0070....Ambient Air Temperature Sensor circuit
P0071....Ambient Air Temperature Sensor range/performance problem
P0072....Ambient Air Temperature Sensor circuit low input
P0073....Ambient Air Temperature Sensor circuit high input
P0074....Ambient Air Temperature Sensor circuit intermittent
P0075....Intake Valve control Solenoid circuit (Bank 1)
P0076....Intake Valve control Solenoid circuit low (Bank 1)
P0077....Intake Valve control Solenoid circuit high (Bank 1)

P0078....Exhaust Valve control Solenoid circuit (Bank 1)
P0079....Exhaust Valve control Solenoid circuit low (Bank 1)
P0080....Exhaust Valve control Solenoid circuit high (Bank 1)
P0081....Intake Valve control Solenoid circuit (Bank 2)
P0082....Intake Valve control Solenoid circuit low (Bank 2)
P0083....Intake Valve control Solenoid circuit high (Bank 2)
P0084....Exhaust Valve control Solenoid circuit (Bank 2)
P0085....Exhaust Valve control Solenoid circuit low (Bank 2)
P0086....Exhaust Valve control Solenoid circuit high (Bank 2)
P0100....Mass or Volume Air Flow circuit malfunction
P0101....Mass or Volume Air Flow circuit range/performance problem
P0102....Mass or Volume Air Flow circuit low input
P0103....Mass or Volume Air Flow circuit high input
P0104....Mass or Volume Air Flow circuit intermittent
P0105....Manifold Absolute Pressure/Barometric Pressure circuit malfunction
P0106....Manifold Absolute Pressure/Barometric Pressure circuit range/performance problem
P0107....Manifold Absolute Pressure/Barometric Pressure circuit low input
P0108....Manifold Absolute Pressure/Barometric Pressure circuit high input
P0109....Manifold Absolute Pressure/Barometric Pressure circuit intermittent
P0110....Intake Air Temperature circuit malfunction
P0111....Intake Air Temperature circuit range/performance problem
P0112....Intake Air Temperature circuit low input
P0113....Intake Air Temperature circuit high input
P0114....Intake Air Temperature circuit intermittent
P0115....Engine Coolant Temperature circuit malfunction
P0116....Engine Coolant Temperature circuit range/performance problem
P0117....Engine Coolant Temperature circuit low input
P0118....Engine Coolant Temperature circuit high input
P0119....Engine Coolant Temperature circuit intermittent
P0120....Throttle/Petal Position Sensor/Switch A circuit malfunction
P0121....Throttle/Petal Position Sensor/Switch A circuit range/performance problem
P0122....Throttle/Petal Position Sensor/Switch A circuit low input
P0123....Throttle/Petal Position Sensor/Switch A circuit high input
P0124....Throttle/Petal Position Sensor/Switch A circuit intermittent
P0125....Insufficient Coolant Temperature for Closed Loop Fuel control
P0126....Insufficient Coolant Temperature for stable operation
P0127....Intake Air Temperature too high
P0128....Coolant Thermostat (coolant below normal temp range)
P0130....O2 Sensor circuit malfunction (Bank 1 Sensor 1)
P0131....O2 Sensor circuit low Voltage (Bank 1 Sensor 1)
P0132....O2 Sensor circuit high Voltage (Bank 1 Sensor 1)
P0133....O2 Sensor circuit Slow Response (Bank 1 Sensor 1)
P0134....O2 Sensor circuit No Activity detected (Bank 1 Sensor 1)
P0135....O2 Sensor Heater circuit malfunction (Bank 1 Sensor 1)
P0136....O2 Sensor circuit malfunction (Bank 1 Sensor 2)

P0137....O2 Sensor circuit low Voltage (Bank 1 Sensor 2)
P0138....O2 Sensor circuit high Voltage (Bank 1 Sensor 2)
P0139....O2 Sensor circuit Slow Response (Bank 1 Sensor 2)
P0140....O2 Sensor circuit No Activity detected (Bank 1 Sensor 2)
P0141....O2 Sensor Heater circuit malfunction (Bank 1 Sensor 2)
P0142....O2 Sensor circuit malfunction (Bank 1 Sensor 3)
P0143....O2 Sensor circuit low Voltage (Bank 1 Sensor 3)
P0144....O2 Sensor circuit high Voltage (Bank 1 Sensor 3)
P0145....O2 Sensor circuit Slow Response (Bank 1 Sensor 3)
P0146....O2 Sensor circuit No Activity detected (Bank 1 Sensor 3)
P0147....O2 Sensor Heater circuit malfunction (Bank 1 Sensor 3)
P0148....Fuel delivery error
P0149....Fuel timing error
P0150....O2 Sensor circuit malfunction (Bank 2 Sensor 1)
P0151....O2 Sensor circuit low Voltage (Bank 2 Sensor 1)
P0152....O2 Sensor circuit high Voltage (Bank 2 Sensor 1)
P0153....O2 Sensor circuit Slow Response (Bank 2 Sensor 1)
P0154....O2 Sensor circuit No Activity detected (Bank 2 Sensor 1)
P0155....O2 Sensor Heater circuit malfunction (Bank 2 Sensor 1)
P0156....O2 Sensor circuit malfunction (Bank 2 Sensor 2)
P0157....O2 Sensor circuit low Voltage (Bank 2 Sensor 2)
P0158....O2 Sensor circuit high Voltage (Bank 2 Sensor 2)
P0159....O2 Sensor circuit Slow Response (Bank 2 Sensor 2)
P0160....O2 Sensor circuit No Activity detected (Bank 2 Sensor 2)
P0161....O2 Sensor Heater circuit malfunction (Bank 2 Sensor 2)
P0162....O2 Sensor circuit malfunction (Bank 2 Sensor 3)
P0163....O2 Sensor circuit low Voltage (Bank 2 Sensor 3)
P0164....O2 Sensor circuit high Voltage (Bank 2 Sensor 3)
P0165....O2 Sensor circuit Slow Response (Bank 2 Sensor 3)
P0166....O2 Sensor circuit No Activity detected (Bank 2 Sensor 3)
P0167....O2 Sensor Heater circuit malfunction (Bank 2 Sensor 3)
P0168....Fuel temperature too high
P0169....Incorrect fuel composition (flex fuel vehicles)
P0170....Fuel Trim malfunction (Bank 1)
P0171....System too Lean (Bank 1)
P0172....System too Rich (Bank 1)
P0173....Fuel Trim malfunction (Bank 2)
P0174....System too Lean (Bank 2)
P0175....System too Rich (Bank 2)
P0176....Fuel Composition Sensor circuit malfunction
P0177....Fuel Composition Sensor circuit range/performance
P0178....Fuel Composition Sensor circuit low input
P0179....Fuel Composition Sensor circuit high input
P0180....Fuel Temperature Sensor A circuit malfunction
P0181....Fuel Temperature Sensor A circuit range/performance
P0182....Fuel Temperature Sensor A circuit low input

P0183....Fuel Temperature Sensor A circuit high input
P0184....Fuel Temperature Sensor A circuit intermittent
P0185....Fuel Temperature Sensor B circuit malfunction
P0186....Fuel Temperature Sensor B circuit range/performance
P0187....Fuel Temperature Sensor B circuit low input
P0188....Fuel Temperature Sensor B circuit high input
P0189....Fuel Temperature Sensor B circuit intermittent
P0190....Fuel Rail Pressure Sensor circuit malfunction
P0191....Fuel Rail Pressure Sensor circuit range/performance
P0192....Fuel Rail Pressure Sensor circuit low input
P0193....Fuel Rail Pressure Sensor circuit high input
P0194....Fuel Rail Pressure Sensor circuit intermittent
P0195....Engine Oil Temperature Sensor malfunction
P0196....Engine Oil Temperature Sensor range/performance
P0197....Engine Oil Temperature Sensor low
P0198....Engine Oil Temperature Sensor high
P0199....Engine Oil Temperature Sensor intermittent
P0200....Injector circuit malfunction
P0201....Injector circuit malfunction – Cylinder 1
P0202....Injector circuit malfunction – Cylinder 2
P0203....Injector circuit malfunction – Cylinder 3
P0204....Injector circuit malfunction – Cylinder 4
P0205....Injector circuit malfunction – Cylinder 5
P0206....Injector circuit malfunction – Cylinder 6
P0207....Injector circuit malfunction – Cylinder 7
P0208....Injector circuit malfunction – Cylinder 8
P0209....Injector circuit malfunction – Cylinder 9
P0210....Injector circuit malfunction – Cylinder 10
P0211....Injector circuit malfunction – Cylinder 11
P0212....Injector circuit malfunction – Cylinder 12
P0213....Cold Start Injector 1 malfunction
P0214....Cold Start Injector 2 malfunction
P0215....Engine Shutoff Solenoid malfunction
P0216....Injection Timing control circuit malfunction
P0217....Engine Overtemp Condition
P0218....Transmission Over Temperature Condition
P0219....Engine Overspeed Condition
P0220....Throttle/Pedal Position Sensor/Switch B circuit malfunction
P0221....Throttle/Pedal Position Sensor/Switch B circuit range/performance
problem
P0222....Throttle/Pedal Position Sensor/Switch B circuit low input
P0223....Throttle/Pedal Position Sensor/Switch B circuit high input
P0224....Throttle/Petal Position Sensor/Switch B circuit intermittent
P0225....Throttle/Petal Position Sensor/Switch C circuit malfunction
P0226....Throttle/Petal Position Sensor/Switch C circuit range/performance
problem

P0227....Throttle/Petal Position Sensor/Switch C circuit low input
 P0228....Throttle/Petal Position Sensor/Switch C circuit high input
 P0229....Throttle/Petal Position Sensor/Switch C circuit intermittent
 P0230....Fuel Pump Primary circuit malfunction
 P0231....Fuel Pump Secondary circuit low
 P0232....Fuel Pump Secondary circuit high
 P0233....Fuel Pump Secondary circuit intermittent
 P0234....Engine Overboost Condition
 P0235....Turbocharger Boost Sensor A circuit malfunction
 P0236....Turbocharger Boost Sensor A circuit range/performance
 P0237....Turbocharger Boost Sensor A circuit low
 P0238....Turbocharger Boost Sensor A circuit high
 P0239....Turbocharger Boost Sensor B malfunction
 P0240....Turbocharger Boost Sensor B circuit range/performance
 P0241....Turbocharger Boost Sensor B circuit low
 P0242....Turbocharger Boost Sensor B circuit high
 P0243....Turbocharger Wastegate Solenoid A malfunction
 P0244....Turbocharger Wastegate Solenoid A range/performance
 P0245....Turbocharger Wastegate Solenoid A low
 P0246....Turbocharger Wastegate Solenoid A high
 P0247....Turbocharger Wastegate Solenoid B malfunction
 P0248....Turbocharger Wastegate Solenoid B range/performance
 P0249....Turbocharger Wastegate Solenoid B low
 P0250....Turbocharger Wastegate Solenoid B high
 P0251....Injection Pump Fuel Metering control "A" malfunction
 (Cam/Rotor/Injector)
 P0252....Injection Pump Fuel Metering control "A" range/performance
 (Cam/Rotor/Injector)
 P0253....Injection Pump Fuel Metering control "A" low (Cam/Rotor/Injector)
 P0254....Injection Pump Fuel Metering control "A" high (Cam/Rotor/Injector)
 P0255....Injection Pump Fuel Metering control "A" intermittent
 (Cam/Rotor/Injector)
 P0256....Injection Pump Fuel Metering control "B" malfunction
 (Cam/Rotor/Injector)
 P0257....Injection Pump Fuel Metering control "B" range/performance
 (Cam/Rotor/Injector)
 P0258....Injection Pump Fuel Metering control "B" low (Cam/Rotor/Injector)
 P0259....Injection Pump Fuel Metering control "B" high (Cam/Rotor/Injector)
 P0260....Injection Pump Fuel Metering control "B" intermittent
 (Cam/Rotor/Injector)
 P0261....Cylinder 1 Injector circuit low
 P0262....Cylinder 1 Injector circuit high
 P0263....Cylinder 1 Contribution/Balance Fault
 P0264....Cylinder 2 Injector circuit low
 P0265....Cylinder 2 Injector circuit high
 P0266....Cylinder 2 Contribution/Balance Fault
 P0267....Cylinder 3 Injector circuit low

P0268....Cylinder 3 Injector circuit high
P0269....Cylinder 3 Contribution/Balance Fault
P0270....Cylinder 4 Injector circuit low
P0271....Cylinder 4 Injector circuit high
P0272....Cylinder 4 Contribution/Balance Fault
P0273....Cylinder 5 Injector circuit low
P0274....Cylinder 5 Injector circuit high
P0275....Cylinder 5 Contribution/Balance Fault
P0276....Cylinder 6 Injector circuit low
P0277....Cylinder 6 Injector circuit high
P0278....Cylinder 6 Contribution/Balance Fault
P0279....Cylinder 7 Injector circuit low
P0280....Cylinder 7 Injector circuit high
P0281....Cylinder 7 Contribution/Balance Fault
P0282....Cylinder 8 Injector circuit low
P0283....Cylinder 8 Injector circuit high
P0284....Cylinder 8 Contribution/Balance Fault
P0285....Cylinder 9 Injector circuit low
P0286....Cylinder 9 Injector circuit high
P0287....Cylinder 9 Contribution/Balance Fault
P0288....Cylinder 10 Injector circuit low
P0289....Cylinder 10 Injector circuit high
P0290....Cylinder 10 Contribution/Balance Fault
P0291....Cylinder 11 Injector circuit low
P0292....Cylinder 11 Injector circuit high
P0293....Cylinder 11 Contribution/Balance Fault
P0294....Cylinder 12 Injector circuit low
P0295....Cylinder 12 Injector circuit high
P0296....Cylinder 12 Contribution/range Fault
P0300....Random/Multiple Cylinder misfire detected
P0301....Cylinder 1 misfire detected
P0302....Cylinder 2 misfire detected
P0303....Cylinder 3 misfire detected
P0304....Cylinder 4 misfire detected
P0305....Cylinder 5 misfire detected
P0306....Cylinder 6 misfire detected
P0307....Cylinder 7 misfire detected
P0308....Cylinder 8 misfire detected
P0309....Cylinder 9 misfire detected
P0311....Cylinder 11 misfire detected
P0312....Cylinder 12 misfire detected
P0313....misfire detected with low fuel
P0314....Single cylinder misfire (cylinder not specified)
P0320....Ignition/Distributor Engine Speed input circuit malfunction
P0321....Ignition/Distributor Engine Speed input circuit range/performance
P0322....Ignition/Distributor Engine Speed input circuit No Signal

P0323....Ignition/Distributor Engine Speed input circuit intermittent
 P0325....Knock Sensor 1 circuit malfunction (Bank 1 or Single Sensor)
 P0326....Knock Sensor 1 circuit range/performance (Bank 1 or Single Sensor)
 P0327....Knock Sensor 1 circuit low input (Bank 1 or Single Sensor)
 P0328....Knock Sensor 1 circuit high input (Bank 1 or Single Sensor)
 P0329....Knock Sensor 1 circuit intermittent (Bank 1 or Single Sensor)
 P0330....Knock Sensor 2 circuit malfunction (Bank 2)
 P0331....Knock Sensor 2 circuit range/performance (Bank 2)
 P0332....Knock Sensor 2 circuit low input (Bank 2)
 P0333....Knock Sensor 2 circuit high input (Bank 2)
 P0334....Knock Sensor 2 circuit intermittent (Bank 2)
 P0335....Crankshaft Position Sensor A circuit malfunction
 P0336....Crankshaft Position Sensor A circuit range/performance
 P0337....Crankshaft Position Sensor A circuit low input
 P0338....Crankshaft Position Sensor A circuit high input
 P0339....Crankshaft Position Sensor A circuit intermittent
 P0340....Camshaft Position Sensor circuit malfunction
 P0341....Camshaft Position Sensor circuit range/performance
 P0342....Camshaft Position Sensor circuit low input
 P0343....Camshaft Position Sensor circuit high input
 P0344....Camshaft Position Sensor circuit intermittent
 P0345....Camshaft Position Sensor "A" circuit (Bank 2)
 P0346....Camshaft Position Sensor "A" circuit out of range (Bank 2)
 P0347....Camshaft Position Sensor "A" circuit low input (Bank 2)
 P0348....Camshaft Position Sensor "A" circuit high input (Bank 2)
 P0349....Camshaft Position Sensor "A" circuit intermittent (Bank 2)
 P0350....Ignition Coil Primary/Secondary circuit malfunction
 P0351....Ignition Coil A Primary/Secondary circuit malfunction
 P0352....Ignition Coil B Primary/Secondary circuit malfunction
 P0353....Ignition Coil C Primary/Secondary circuit malfunction
 P0354....Ignition Coil D Primary/Secondary circuit malfunction
 P0355....Ignition Coil E Primary/Secondary circuit malfunction
 P0356....Ignition Coil F Primary/Secondary circuit malfunction
 P0357....Ignition Coil G Primary/Secondary circuit malfunction
 P0358....Ignition Coil H Primary/Secondary circuit malfunction
 P0359....Ignition Coil I Primary/Secondary circuit malfunction
 P0360....Ignition Coil J Primary/Secondary circuit malfunction
 P0361....Ignition Coil K Primary/Secondary circuit malfunction
 P0362....Ignition Coil L Primary/Secondary circuit malfunction
 P0365....Camshaft Position Sensor "B" circuit (Bank 1)
 P0366....Camshaft Position Sensor "B" circuit out of range (Bank 1)
 P0367....Camshaft Position Sensor "B" circuit low input (Bank 1)
 P0368....Camshaft Position Sensor "B" circuit high input (Bank 1)
 P0369....Camshaft Position Sensor "B" circuit intermittent (Bank 1)
 P0370....Timing Reference high Resolution Signal A malfunction
 P0371....Timing Reference high Resolution Signal A Too Many Pulses

P0372....Timing Reference high Resolution Signal A Too Few Pulses
P0373....Timing Reference high Resolution Signal A intermittent/Erratic Pulses
P0374....Timing Reference high Resolution Signal A No Pulses
P0375....Timing Reference high Resolution Signal B malfunction
P0376....Timing Reference high Resolution Signal B Too Many Pulses
P0377....Timing Reference high Resolution Signal B Too Few Pulses
P0378....Timing Reference high Resolution Signal B intermittent/Erratic Pulses
P0379....Timing Reference high Resolution Signal B No Pulses
P0380....Glow Plug/Heater circuit "A" malfunction
P0381....Glow Plug/Heater Indicator circuit malfunction
P0382....Exhaust Gas Recirculation Flow malfunction
P0385....Crankshaft Position Sensor B circuit malfunction
P0386....Crankshaft Position Sensor B circuit range/performance
P0387....Crankshaft Position Sensor B circuit low input
P0388....Crankshaft Position Sensor B circuit high input
P0389....Crankshaft Position Sensor B circuit intermittent
P0390....Camshaft Position Sensor "B" circuit intermittent
P0391....Camshaft Position Sensor "B" circuit out of range (Bank 2)
P0392....Camshaft Position Sensor "B" circuit low input (Bank 2)
P0393....Camshaft Position Sensor "B" circuit high input (Bank 2)
P0394....Camshaft Position Sensor "B" circuit intermittent (Bank 2)
P0400....Exhaust Gas Recirculation Flow malfunction
P0401....Exhaust Gas Recirculation Flow Insufficient detected
P0402....Exhaust Gas Recirculation Flow Excessive detected
P0403....Exhaust Gas Recirculation circuit malfunction
P0404....Exhaust Gas Recirculation circuit range/performance
P0405....Exhaust Gas Recirculation Sensor A circuit low
P0406....Exhaust Gas Recirculation Sensor A circuit high
P0407....Exhaust Gas Recirculation Sensor B circuit low
P0408....Exhaust Gas Recirculation Sensor B circuit high
P0410....Secondary Air Injection System malfunction
P0411....Secondary Air Injection System Incorrect Flow detected
P0412....Secondary Air Injection System Switching Valve A circuit malfunction
P0413....Secondary Air Injection System Switching Valve A circuit open
P0414....Secondary Air Injection System Switching Valve A circuit shorted
P0415....Secondary Air Injection System Switching Valve B circuit malfunction
P0416....Secondary Air Injection System Switching Valve B circuit open
P0417....Secondary Air Injection System Switching Valve B circuit shorted
P0418....Secondary Air Injection System Relay "A" circuit malfunction
P0419....Secondary Air Injection System Relay "B" circuit malfunction
P0420....Catalyst System Efficiency Below Threshold (Bank 1)
P0421....Warm Up Catalyst Efficiency Below Threshold (Bank 1)
P0422....Main Catalyst Efficiency Below Threshold (Bank 1)
P0423....Heated Catalyst Efficiency Below Threshold (Bank 1)

P0424....Heated Catalyst Temperature Below Threshold (Bank 1)
 P0425....Catalyst Temperature Sensor (Bank 1)
 P0426....Catalyst Temperature Sensor out of range (Bank 1)
 P0427....Catalyst Temperature Sensor low input (Bank 1)
 P0428....Catalyst Temperature Sensor high input (Bank 1)
 P0429....Catalyst Heater control circuit (Bank 1)
 P0430....Catalyst System Efficiency Below Threshold (Bank 2)
 P0431....Warm Up Catalyst Efficiency Below Threshold (Bank 2)
 P0432....Main Catalyst Efficiency Below Threshold (Bank 2)
 P0433....Heated Catalyst Efficiency Below Threshold (Bank 2)
 P0434....Heated Catalyst Temperature Below Threshold (Bank 2)
 P0435....Catalyst Temperature Sensor (Bank 2)
 P0436....Catalyst Temperature Sensor out of range (Bank 2)
 P0437....Catalyst Temperature Sensor low input (Bank 2)
 P0438....Catalyst Temperature Sensor high input (Bank 2)
 P0439....Catalyst Heater control circuit (Bank 2)
 P0440....Evaporative Emission control System malfunction
 P0441....Evaporative Emission control System Incorrect Purge Flow
 P0442....Evaporative Emission control System Leak detected (small leak)
 P0443....Evaporative Emission control System Purge control Valve circuit malfunction
 P0444....Evaporative Emission control System Purge control Valve circuit open
 P0445....Evaporative Emission control System Purge control Valve circuit shorted
 P0446....Evaporative Emission control System Vent control circuit malfunction
 P0447....Evaporative Emission control System Vent control circuit open
 P0448....Evaporative Emission control System Vent control circuit shorted
 P0449....Evaporative Emission control System Vent Valve/Solenoid circuit malfunction
 P0450....Evaporative Emission control System Pressure Sensor malfunction
 P0451....Evaporative Emission control System Pressure Sensor range/performance
 P0452....Evaporative Emission control System Pressure Sensor low input
 P0453....Evaporative Emission control System Pressure Sensor high input
 P0454....Evaporative Emission control System Pressure Sensor intermittent
 P0455....Evaporative Emission control System Leak detected (gross leak)
 P0456....Evaporative Emission control System leak detected (small leak)
 P0457....Evaporative Emission control System leak detected (loose filler cap)
 P0460....Fuel Level Sensor circuit malfunction P0461....Fuel Level Sensor circuit range/performance
 P0462....Fuel Level Sensor circuit low input
 P0463....Fuel Level Sensor circuit high input
 P0464....Fuel Level Sensor circuit intermittent
 P0465....Purge Flow Sensor circuit malfunction
 P0466....Purge Flow Sensor circuit range/performance
 P0467....Purge Flow Sensor circuit low input

P0468....Purge Flow Sensor circuit high input
P0469....Purge Flow Sensor circuit intermittent
P0470....Exhaust Pressure Sensor malfunction
P0471....Exhaust Pressure Sensor range/performance
P0472....Exhaust Pressure Sensor low
P0473....Exhaust Pressure Sensor high
P0474....Exhaust Pressure Sensor intermittent
P0475....Exhaust Pressure control Valve malfunction
P0476....Exhaust Pressure control Valve range/performance
P0477....Exhaust Pressure control Valve low
P0478....Exhaust Pressure control Valve high
P0479....Exhaust Pressure control Valve intermittent
P0480....Cooling Fan 1 control circuit malfunction
P0481....Cooling Fan 2 control circuit malfunction
P0482....Cooling Fan 3 control circuit malfunction
P0483....Cooling Fan Rationality Check malfunction
P0484....Cooling Fan circuit Over Current
P0485....Cooling Fan Power/Ground circuit malfunction
P0486....Exhaust Gas Recirculation Sensor "B" circuit
P0487....Exhaust Gas Recirculation Throttle Position control circuit
P0488....Exhaust Gas Recirculation Throttle Position control out of range
P0491....Secondary Air Injection System (Bank 1)
P0492....Secondary Air Injection System (Bank 2)
P0500....Vehicle Speed Sensor malfunction
P0501....Vehicle Speed Sensor range/performance
P0502....Vehicle Speed Sensor low input
P0503....Vehicle Speed Sensor intermittent/Erratic/high
P0505....Idle control System malfunction
P0506....Idle control System RPM lower Than Expected
P0507....Idle control System RPM higher Than Expected
P0508....Idle control System circuit low
P0509....Idle control System circuit high
P0510....Closed Throttle Position Switch malfunction
P0512....Starter request circuit
P0513....Incorrect Immobilizer key
P0515....Battery Temperature Sensor circuit
P0516....Battery Temperature Sensor circuit low
P0517....Battery Temperature Sensor circuit high
P0520....Engine Oil Pressure Sensor/Switch circuit malfunction
P0521....Engine Oil Pressure Sensor/Switch circuit range/performance
P0522....Engine Oil Pressure Sensor/Switch circuit low Voltage
P0523....Engine Oil Pressure Sensor/Switch circuit high Voltage
P0524....Engine oil pressure too low
P0530....A/C Refrigerant Pressure Sensor circuit malfunction
P0531....A/C Refrigerant Pressure Sensor circuit out of range
P0532....A/C Refrigerant Pressure Sensor circuit low input

P0533....A/C Refrigerant Pressure Sensor circuit high input
P0534....Air Conditioner Refrigerant Charge Loss
P0540....Intake Air Heater circuit
P0541....Intake Air Heater circuit low
P0542....Intake Air Heater circuit high
P0544....Exhaust Gas Temperature Sensor circuit (Bank 1)
P0545....Exhaust Gas Temperature Sensor circuit low (Bank 1)
P0546....Exhaust Gas Temperature Sensor circuit high (Bank 1)
P0547....Exhaust Gas Temperature Sensor circuit (Bank 2)
P0548....Exhaust Gas Temperature Sensor circuit low (Bank 2)
P0549....Exhaust Gas Temperature Sensor circuit high (Bank 2)
P0550....Power Steering Pressure Sensor circuit malfunction
P0551....Power Steering Pressure Sensor circuit range/performance
P0552....Power Steering Pressure Sensor circuit low input
P0553....Power Steering Pressure Sensor circuit high input
P0554....Power Steering Pressure Sensor circuit intermittent
P0560....System Voltage malfunction
P0561....System Voltage Unstable
P0562....System Voltage low P0563....System Voltage high
P0564....Cruise control multi-function input signal
P0565....Cruise control On Signal malfunction
P0566....Cruise control Off Signal malfunction
P0567....Cruise control Resume Signal malfunction
P0568....Cruise control Set Signal malfunction
P0569....Cruise control Coast Signal malfunction
P0570....Cruise control Accel Signal malfunction
P0571....Cruise control/Brake Switch A circuit malfunction
P0572....Cruise control/Brake Switch A circuit low
P0573....Cruise control/Brake Switch A circuit high
P0574....Cruise control system - vehicle speed too high
P0575....Cruise control Input circuit
P0576....Cruise control Input circuit low
P0577....Cruise control Input circuit high
P0578....Cruise control Related malfunction
P0579....Cruise control Related malfunction
P0580....Cruise control Related malfunction
P0600....Serial Communication Link malfunction
P0601....Internal control Module Memory Check Sum Error
P0602....control Module Programming Error
P0603....Internal control Module Keep Alive Memory (KAM) Error
P0604....Internal control Module Random Access Memory (RAM) Error
P0605....Internal control Module Read Only Memory (ROM) Error
P0606....PCM Processor Fault
P0607....control Module performance
P0608....control Module VSS Output "A" malfunction
P0609....control Module VSS Output "B" malfunction

P0610....control Module vehicle options error
P0615....Starter Relay circuit
P0616....Starter Relay circuit low
P0617....Starter Relay circuit high
P0618....Alternative Fuel control Module KAM error
P0619....Alternative Fuel control Module RAM/ROM error
P0620....Generator control circuit malfunction
P0621....Generator Lamp "L" control circuit malfunction
P0622....Generator Field "F" control circuit malfunction
P0623....Generator Lamp control circuit
P0624....Fuel Cap Lamp control circuit
P0630....VIN not programmed or mismatch - ECM/PCM
P0631....VIN not programmed or mismatch - TCM
P0635....Power Steering control circuit
P0636....Power Steering control circuit low
P0637....Power Steering control circuit high
P0638....Throttle Actuator control out of range (Bank 1)
P0639....Throttle Actuator control out of range (Bank 2)
P0640....Intake Air Heater control circuit P0645....A/C Clutch Relay control circuit
P0646....A/C Clutch Relay control circuit high
P0647....A/C Clutch Relay control circuit high
P0648....Immobilizer Lamp control circuit
P0649....Speed control Lamp control circuit
P0650....Malfunction Indicator Lamp (MIL) control circuit malfunction
P0654....Engine RPM Output circuit malfunction
P0655....Engine Hot Lamp Output control circuit malfunction
P0656....Fuel Level Output circuit malfunction
P0660....Intake Manifold Tuning Valve control circuit (Bank 1)
P0661....Intake Manifold Tuning Valve control circuit low (Bank 1)
P0662....Intake Manifold Tuning Valve control circuit high (Bank 1)
P0663....Intake Manifold Tuning Valve control circuit (Bank 2)
P0664....Intake Manifold Tuning Valve control circuit low (Bank 2)
P0665....Intake Manifold Tuning Valve control circuit high (Bank 2)
P0700....Transmission control System malfunction
P0701....Transmission control System out of range
P0702....Transmission control System Electrical
P0703....Torque Converter/Brake Switch B circuit malfunction
P0704....Clutch Switch input circuit malfunction
P0705....Transmission range Sensor circuit malfunction (PRNDL input)
P0706....Transmission range Sensor circuit range/performance
P0707....Transmission range Sensor circuit low input
P0708 Transmission range Sensor circuit high input
P0709 Transmission range Sensor circuit intermittent
P0710....Transmission Fluid Temperature Sensor circuit malfunction
P0711....Transmission Fluid Temperature Sensor circuit range/performance

P0712....Transmission Fluid Temperature Sensor circuit low input
P0713....Transmission Fluid Temperature Sensor circuit high input
P0714....Transmission Fluid Temperature Sensor circuit intermittent
P0715....input/Turbine Speed Sensor circuit malfunction
P0716....input/Turbine Speed Sensor circuit range/performance
P0717....input/Turbine Speed Sensor circuit No Signal
P0718....input/Turbine Speed Sensor circuit intermittent
P0719....Torque Converter/Brake Switch B circuit low
P0720....Output Speed Sensor circuit malfunction
P0721....Output Speed Sensor range/performance
P0722....Output Speed Sensor No Signal
P0723....Output Speed Sensor intermittent
P0724....Torque Converter/Brake Switch B circuit high
P0725....Engine Speed input circuit malfunction
P0726....Engine Speed input circuit range/performance
P0727....Engine Speed input circuit No Signal
P0728....Engine Speed input circuit intermittent
P0730....Incorrect Gear Ratio
P0731....Gear 1 Incorrect ratio
P0732....Gear 2 Incorrect ratio
P0733....Gear 3 Incorrect ratio
P0734....Gear 4 Incorrect ratio
P0735....Gear 5 Incorrect ratio
P0736....Reverse incorrect gear ratio
P0737....TCM engine speed output circuit
P0738....TCM engine speed output circuit low
P0739....TCM engine speed output circuit high
P0740....Torque Converter Clutch circuit malfunction
P0741....Torque Converter Clutch circuit performance or stuck off
P0742....Torque Converter Clutch circuit stuck on
P0743....Torque Converter Clutch circuit electrical
P0744....Torque Converter Clutch circuit intermittent
P0745....Pressure control Solenoid malfunction
P0746....Pressure control Solenoid performance or Stuck Off
P0747....Pressure control Solenoid Stuck On
P0748....Pressure control Solenoid Electrical
P0749....Pressure control Solenoid intermittent
P0750....Shift Solenoid A malfunction
P0751....Shift Solenoid A performance or Stuck Off
P0752....Shift Solenoid A Stuck On
P0753....Shift Solenoid A Electrical
P0754....Shift Solenoid A intermittent
P0755....Shift Solenoid B malfunction
P0756....Shift Solenoid B performance or Stuck Off
P0757....Shift Solenoid B Stuck On
P0758....Shift Solenoid B Electrical

P0759....Shift Solenoid B intermittent
 P0760....Shift Solenoid C malfunction
 P0761....Shift Solenoid C performance or Stuck Off
 P0762....Shift Solenoid C Stuck On
 P0763....Shift Solenoid C Electrical
 P0764....Shift Solenoid C intermittent
 P0765....Shift Solenoid D malfunction
 P0766....Shift Solenoid D performance or Stuck Off
 P0767....Shift Solenoid D Stuck On
 P0768....Shift Solenoid D Electrical
 P0769....Shift Solenoid D intermittent
 P0770....Shift Solenoid E malfunction
 P0771....Shift Solenoid E performance or Stuck Off
 P0772....Shift Solenoid E Stuck On
 P0773....Shift Solenoid E Electrical
 P0774....Shift Solenoid E intermittent
 P0775....Pressure Control Solenoid "B"
 P0776....Pressure Control Solenoid "B" performance or stuck off
 P0777....Pressure Control Solenoid "B" stuck on
 P0778....Pressure Control Solenoid "B" electrical
 P0779....Pressure Control Solenoid "B" intermittent
 P0780....Shift malfunction
 P0781....1-2 Shift malfunction
 P0782....2-3 Shift malfunction
 P0783....3-4 Shift malfunction
 P0784....4-5 Shift malfunction
 P0785....Shift/Timing Solenoid malfunction
 P0786....Shift/Timing Solenoid range/performance
 P0787....Shift/Timing Solenoid low
 P0788....Shift/Timing Solenoid high
 P0789....Shift/Timing Solenoid intermittent
 P0790....Normal/performance Switch circuit malfunction
 P0791....Intermediate Shaft Speed Sensor circuit
 P0792....Intermediate Shaft Speed Sensor circuit out of range
 P0793....Intermediate Shaft Speed Sensor circuit no signal
 P0794....Intermediate Shaft Speed Sensor circuit intermittent
 P0795....Pressure Control Solenoid "C"
 P0796....Pressure Control Solenoid "C" out of range
 P0797....Pressure Control Solenoid "C" stuck on
 P0798....Pressure Control Solenoid "C" electrical
 P0799....Pressure Control Solenoid "C" intermittent
 P0801....Reverse Inhibit control circuit malfunction
 P0803....1-4 Upshift (Skip Shift) Solenoid control circuit malfunction
 P0804....1-4 Upshift (Skip Shift) Lamp control circuit malfunction
 P0805....Clutch Position Sensor circuit P0806....Clutch Position Sensor circuit
 out of range

P0807....Clutch Position Sensor circuit low
P0808....Clutch Position Sensor circuit high
P0809....Clutch Position Sensor circuit intermittent
P0810....Clutch Position control error
P0811....Excessive Clutch Slippage
P0812....Reverse Input circuit
P0813....Reverse Output circuit
P0815....Upshift Switch circuit
P0816....Downshift Switch circuit
P0817....Starter Disable circuit
P0818....Driveline Disconnect Switch input circuit
P0820....Gear Lever X-Y Position Sensor circuit
P0821....Gear Lever X Position circuit
P0822....Gear Lever Y Position circuit
P0823....Gear Lever X Position circuit intermittent
P0824....Gear Lever Y Position circuit intermittent
P0830....Clutch Pedal Switch "A" circuit
P0831....Clutch Pedal Switch "A" circuit low
P0832....Clutch Pedal Switch "A" circuit high
P0833....Clutch Pedal Switch "B" circuit
P0834....Clutch Pedal Switch "B" circuit low
P0835....Clutch Pedal Switch "B" circuit high
P0837....Four Wheel Drive Switch circuit
P0838....Four Wheel Drive Switch circuit low
P0839....four Wheel Drive Switch circuit high
P0840....Transmission Fluid Pressure Sensor/Switch "A" circuit
P0841....Transmission Fluid Pressure Sensor/Switch "A" circuit out of range
P0842...Transmission Fluid Pressure Sensor/Switch "A" circuit low
P0843....Transmission Fluid Pressure Sensor/Switch "A" circuit high
P0844....Transmission Fluid Pressure Sensor/Switch "A" circuit intermittent
P0845....Transmission Fluid Pressure Sensor/Switch "B" circuit
P0846....Transmission Fluid Pressure Sensor/Switch "B" circuit out of range
P0847...Transmission Fluid Pressure Sensor/Switch "B" circuit low
P0848....Transmission Fluid Pressure Sensor/Switch "B" circuit high
P0849....Transmission Fluid Pressure Sensor/Switch "B" circuit intermittent
C0000 - Vehicle Speed Information Circuit Malfunction

4.3.5.1.1.2. BODY ERROR CODES

Los códigos de error en la carrocería del vehículo no se encuentran estandarizados por normativa, aunque existen unos códigos de GM, que generalmente se usan como estandarizados.

4.3.5.1.1.3. CHASSIS ERROR CODES

C0035 - Left Front Wheel Speed Circuit Malfunction
C0040 - Right Front Wheel Speed Circuit Malfunction
C0041 - Right Front Wheel Speed Sensor Circuit Range/Performance (EBCM)
C0045 - Left Rear Wheel Speed Circuit Malfunction
C0046 - Left Rear Wheel Speed Sensor Circuit Range/Performance (EBCM)
C0050 - Right Rear Wheel Speed Circuit Malfunction
C0051 - LF Wheel Speed Sensor Circuit Range/Performance (EBCM)
C0060 - Left Front ABS Solenoid #1 Circuit Malfunction
C0065 - Left Front ABS Solenoid #2 Circuit Malfunction
C0070 - Right Front ABS Solenoid #1 Circuit Malfunction
C0075 - Right Front ABS Solenoid #2 Circuit Malfunction
C0080 - Left Rear ABS Solenoid #1 Circuit Malfunction
C0085 - Left Rear ABS Solenoid #2 Circuit Malfunction
C0090 - Right Rear ABS Solenoid #1 Circuit Malfunction
C0095 - Right Rear ABS Solenoid #2 Circuit Malfunction
C0110 - Pump Motor Circuit Malfunction
C0121 - Valve Relay Circuit Malfunction
C0128 - Low Brake Fluid Circuit Low
C0141 - Left TCS Solenoid #1 Circuit Malfunction
C0146 - Left TCS Solenoid #2 Circuit Malfunction
C0151 - Right TCS Solenoid #1 Circuit Malfunction
C0156 - Right TCS Solenoid #2 Circuit Malfunction
C0161 - ABS/TCS Brake Switch Circuit Malfunction
C0221 - Right Front Wheel Speed Sensor Circuit Open
C0222 - Right Front Wheel Speed Signal Missing
C0223 - Right Front Wheel Speed Signal Erratic
C0225 - Left Front Wheel Speed Sensor Circuit Open
C0226 - Left Front Wheel Speed Signal Missing
C0227 - Left Front Wheel Speed Signal Erratic
C0229 - Drop Out of Front Wheel Speed Signals
C0235 - Rear Wheel Speed Signal Circuit Open
C0236 - Rear Wheel Speed Signal Circuit Missing
C0237 - Rear Wheel Speed Signal Erratic
C0238 - Wheel Speed Mismatch
C0241 - EBCM Control Valve Circuit
C0245 - Wheel Speed Sensor Frequency Error
C0254 - EBCM Control Valve Circuit
C0265 - EBCM Relay Circuit
C0266 - EBCM Relay Circuit
C0267 - Pump Motor Circuit Open/Shorted
C0268 - Pump Motor Circuit Open/Shorted
C0269 - Excessive Dump/Isolation Time
C0271 - EBCM Malfunction

C0272 - EBCM Malfunction
C0273 - EBCM Malfunction
C0274 - Excessive Dump/Isolation Time
C0279 - Powertrain Configuration Not Valid
C0281 - Brake Switch Circuit
C0283 - Traction Switch Shorted to Ground
C0284 - EBCM Malfunction
C0286 - ABS Indicator Lamp Circuit Shorted to B+
C0287 - Delivered Torque Circuit
C0288 - Brake Warning Lamp Circuit Shorted to B+
C0290 - Lost Communications With PCM
C0292 - Lost Communications With PCM
C0291 - Lost Communications With BCM
C0297 - Powertrain Configuration Data Not Received
C0298 - Powertrain Indicated Traction Control Malfunction
C0300 - Rear Speed Sensor Malfunction
C0305 - Front Speed Sensor Malfunction
C0306 - Motor A or B Circuit
C0308 - Motor A/B Circuit Low
C0309 - Motor A/B Circuit High
C0310 - Motor A/B Circuit Open
C0315 - Motor Ground Circuit Open
C0321 - Transfer Case Lock Circuit
C0323 - T-Case Lock Circuit Low
C0324 - T-Case Lock Circuit High
C0327 - Encoder Circuit Malfunction
C0357 - Park Switch Circuit High
C0359 - Four Wheel Drive Low Range (4LO) Discrete Output Circuit
C0362 - 4LO Discrete Output Circuit High
C0367 - Front Axle Control Circuit High
C0374 - General System Malfunction
C0376 - Front/Rear Shaft Speed Mismatch
C0379 - Front Axle System
C0387 - Unable to Perform Shift
C0472 - Steering Handwheel Speed Sensor Signal V Low
C0473 - Steering Handwheel Speed Sensor Signal V High
C0495 - EVO Tracking Error
C0498 - Steering Assist Control Actuator Feed Circuit Low
C0499 - Steering Assist Control Solenoid Feed Circuit High
C0503 - Steering Assist Control Solenoid Return Circuit Low
C0504 - Steering Assist Control Solenoid Return Circuit High
C0550 - ECU Malfunction - internal write / checksum malfunction
C0559 - EEPROM Checksum Error
C0563 - Calibration ROM Checksum Error
C0577 - Left Front Solenoid Circuit Low
C0578 - Left Front Solenoid Circuit High

C0579 - Left Front Solenoid Circuit Open
 C0582 - Right Front Solenoid Circuit Low
 C0583 - Right Front Solenoid Circuit High
 C0584 - Right Front Solenoid Circuit Open
 C0587 - Left Rear Solenoid Circuit Low
 C0588 - Left Rear Solenoid Circuit High
 C0589 - Left Rear Solenoid Circuit Open
 C0592 - Right Rear Solenoid Circuit Low
 C0593 - Right Rear Solenoid Circuit High
 C0594 - Right Rear Solenoid Circuit Open
 C0611 - VIN Information Error
 C0615 - Left Front Position Sensor Malfunction
 C0620 - Right Front Position Sensor Malfunction
 C0625 - Left Rear Position Sensor Malfunction
 C0628 - Level Control Position Sensor Circuit High
 C0630 - Right Rear Position Sensor Malfunction
 C0635 - Left Front Normal Force Circuit Malfunction
 C0638 - Left Front Normal Force Circuit High
 C0640 - Right Front Normal Force Circuit Malfunction
 C0643 - Right Front Normal Force Circuit High
 C0655 - Level Control Compressor Relay Malfunction
 C0657 - Level Control Compressor Circuit Low
 C0658 - Level Control Compressor Circuit High
 C0660 - Level Control Exhaust Valve Circuit Malfunction
 C0662 - Level Control Exhaust Valve Circuit Low
 C0663 - Level Control Exhaust Valve Circuit High
 C0665 - Chassis Pitch Signal Circuit
 C0690 - Damper Control Relay Circuit Malfunction
 C0691 - Damper Control Relay Circuit Range
 C0693 - Damper Control Relay Circuit High
 C0695 - Position Sensor Overcurrent (8 volt supply)
 C0696 - Position Sensor Overcurrent (5 volt supply)
 C0710 - Steering Position Signal Malfunction
 C0750 - Tire Pressure Monitor (TPM) system sensor not transmitting
 C0755 - Tire Pressure Monitor (TPM) system sensor not transmitting
 C0760 - Tire Pressure Monitor (TPM) system sensor not transmitting
 C0765 - Tire Pressure Monitor (TPM) system sensor not transmitting
 C0800 - Device Power #1 Circuit Malfunction
 C0896 - Electronic Suspension Control (ESC) voltaje is outside the normal range of 9 to 15.5 volts.

4.3.5.1.1.4. NETWORK ERROR CODES

U0001 High Speed CAN Communication Bus
 U0002 High Speed CAN Communication Bus Performance
 U0003 High Speed CAN Communication Bus (+) Open

U0004 High Speed CAN Communication Bus (+) Low
U0005 High Speed CAN Communication Bus (+) High
U0006 High Speed CAN Communication Bus (-) Open
U0007 High Speed CAN Communication Bus (-) Low
U0008 High Speed CAN Communication Bus (-) High
U0009 High Speed CAN Communication Bus (-) shorted to Bus (+)
U0010 Medium Speed CAN Communication Bus
U0011 Medium Speed CAN Communication Bus Performance
U0012 Medium Speed CAN Communication Bus (+) Open
U0013 Medium Speed CAN Communication Bus (+) Low
U0014 Medium Speed CAN Communication Bus (+) High
U0015 Medium Speed CAN Communication Bus (-) Open
U0016 Medium Speed CAN Communication Bus (-) Low
U0017 Medium Speed CAN Communication Bus (-) High
U0018 Medium Speed CAN Communication Bus (-) shorted to Bus
U0019 Low Speed CAN Communication Bus
U0020 Low Speed CAN Communication Bus Performance
U0021 Low Speed CAN Communication Bus (+) Open
U0022 Low Speed CAN Communication Bus (+) Low
U0023 Low Speed CAN Communication Bus (+) High
U0024 Low Speed CAN Communication Bus (-) Open
U0025 Low Speed CAN Communication Bus (-) Low
U0026 Low Speed CAN Communication Bus (-) High
U0027 Low Speed CAN Communication Bus (-) shorted to Bus (+)
U0028 Vehicle Communication Bus A
U0029 Vehicle Communication Bus A Performance
U0030 Vehicle Communication Bus A (+) Open
U0031 Vehicle Communication Bus A (+) Low
U0032 Vehicle Communication Bus A (+) High
U0033 Vehicle Communication Bus A (-) Open
U0034 Vehicle Communication Bus A (-) Low
U0035 Vehicle Communication Bus A (-) high
U0036 Vehicle Communication Bus A (-) shorted to Bus A (+)
U0037 Vehicle Communication Bus B
U0038 Vehicle Communication Bus B Performance
U0039 Vehicle Communication Bus B (+) Open
U0040 Vehicle Communication Bus B (+) Low
U0041 Vehicle Communication Bus B (+) High
U0042 Vehicle Communication Bus B (-) Open
U0043 Vehicle Communication Bus B (-) Low
U0044 Vehicle Communication Bus B (-) High
U0045 Vehicle Communication Bus B (-) shorted to Bus B (+)
U0046 Vehicle Communication Bus C
U0047 Vehicle Communication Bus C Performance
U0048 Vehicle Communication Bus C (+) Open
U0049 Vehicle Communication Bus C (+) Low

U0050 Vehicle Communication Bus C (+) High
U0051 Vehicle Communication Bus C (-) Open
U0052 Vehicle Communication Bus C (-) Low
U0053 Vehicle Communication Bus C (-) High
U0054 Vehicle Communication Bus C (-) shorted to Bus C (+)
U0055 Vehicle Communication Bus D
U0056 Vehicle Communication Bus D Performance
U0057 Vehicle Communication Bus D (+) Open
U0058 Vehicle Communication Bus D (+) Low
U0059 Vehicle Communication Bus D (+) High
U0060 Vehicle Communication Bus D (-) Open
U0061 Vehicle Communication Bus D (-) Low
U0062 Vehicle Communication Bus D (-) High
U0063 Vehicle Communication Bus D (-) shorted to Bus D (+)
U0064 Vehicle Communication Bus E
U0065 Vehicle Communication Bus E Performance
U0066 Vehicle Communication Bus E (+) Open
U0067 Vehicle Communication Bus E (+) Low
U0068 Vehicle Communication Bus E (+) High
U0069 Vehicle Communication Bus E (-) Open
U0070 Vehicle Communication Bus E (-) Low
U0071 Vehicle Communication Bus E (-) High
U0072 Vehicle Communication Bus E (-) shorted to Bus E (+)
U0073 Control Module Communication Bus Off
U0074 Reserved by Document
U0075 Reserved by Document
U0076 Reserved by Document
U0077 Reserved by Document
U0078 Reserved by Document
U0079 Reserved by Document
U0080 Reserved by Document
U0081 Reserved by Document
U0082 Reserved by Document
U0083 Reserved by Document
U0084 Reserved by Document
U0085 Reserved by Document
U0086 Reserved by Document
U0087 Reserved by Document
U0088 Reserved by Document
U0089 Reserved by Document
U0090 Reserved by Document
U0091 Reserved by Document
U0092 Reserved by Document
U0093 Reserved by Document
U0094 Reserved by Document
U0095 Reserved by Document

U0096 Reserved by Document
U0097 Reserved by Document
U0098 Reserved by Document
U0099 Reserved by Document
U0100 Lost Communication With ECM/PCM "A"
U0101 Lost Communication with TCM
U0102 Lost Communication with Transfer Case Control Module
U0103 Lost Communication With Gear Shift Module
U0104 Lost Communication With Cruise Control Module
U0105 Lost Communication With Fuel Injector Control Module
U0106 Lost Communication With Glow Plug Control Module
U0107 Lost Communication With Throttle Actuator Control Module
U0108 Lost Communication With Alternative Fuel Control Module
U0109 Lost Communication With Fuel Pump Control Module
U0110 Lost Communication With Drive Motor Control Module
U0111 Lost Communication With Battery Energy Control Module "A"
U0112 Lost Communication With Battery Energy Control Module "B"
U0113 Lost Communication With Emissions Critical Control Information
U0114 Lost Communication With Four-Wheel Drive Clutch Control Module
U0115 Lost Communication With ECM/PCM "B"
U0116 Reserved by Document
U0117 Reserved by Document
U0118 Reserved by Document
U0119 Reserved by Document
U0120 Reserved by Document
U0121 Lost Communication With Anti-Lock Brake System (ABS) Control Module
U0122 Lost Communication With Vehicle Dynamics Control Module
U0123 Lost Communication With Yaw Rate Sensor Module
U0124 Lost Communication With Lateral Acceleration Sensor Module
U0125 Lost Communication With Multi-axis Acceleration Sensor Module
U0126 Lost Communication With Steering Angle Sensor Module
U0127 Lost Communication With Tire Pressure Monitor Module
U0128 Lost Communication With Park Brake Control Module
U0129 Lost Communication With Brake System Control Module
U0130 Lost Communication With Steering Effort Control Module
U0131 Lost Communication With Power Steering Control Module
U0132 Lost Communication With Ride Level Control Module
U0133 Reserved by Document
U0134 Reserved by Document
U0135 Reserved by Document
U0136 Reserved by Document
U0137 Reserved by Document
U0138 Reserved by Document
U0139 Reserved by Document
U0140 Lost Communication With Body Control Module

U0141 Lost Communication With Body Control Module "A"
U0142 Lost Communication With Body Control Module "B"
U0143 Lost Communication With Body Control Module "C"
U0144 Lost Communication With Body Control Module "D"
U0145 Lost Communication With Body Control Module "E"
U0146 Lost Communication With Gateway "A"
U0147 Lost Communication With Gateway "B"
U0148 Lost Communication With Gateway "C"
U0149 Lost Communication With Gateway "D"
U0150 Lost Communication With Gateway "E"
U0151 Lost Communication With Restraints Control Module
U0152 Lost Communication With Side Restraints Control Module
U0153 Lost Communication With Side Restraints Control Module
U0154 Lost Communication With Restraints Occupant Sensing Control
Module
U0155 Lost Communication With Instrument Panel Cluster (IPC) Control
Module
U0156 Lost Communication With Information Center "A"
U0157 Lost Communication With Information Center "B"
U0158 Lost Communication With Head Up Display
U0159 Lost Communication With Parking Assist Control Module
U0160 Lost Communication With Audible Alert Control Module
U0161 Lost Communication With Compass Module
U0162 Lost Communication With Navigation Display Module
U0163 Lost Communication With Navigation Control Module
U0164 Lost Communication With HVAC Control Module
U0165 Lost Communication With HVAC Control Module
U0166 Lost Communication With Auxiliary Heater Control Module
U0167 Lost Communication With Vehicle Immobilizer Control Module
U0168 Lost Communication With Vehicle Security Control Module
U0169 Lost Communication With Sunroof Control Module
U0170 Lost Communication With "Restraints System Sensor A"
U0171 Lost Communication With "Restraints System Sensor B"
U0172 Lost Communication With "Restraints System Sensor C"
U0173 Lost Communication With "Restraints System Sensor D"
U0174 Lost Communication With "Restraints System Sensor E"
U0175 Lost Communication With "Restraints System Sensor F"
U0176 Lost Communication With "Restraints System Sensor G"
U0177 Lost Communication With "Restraints System Sensor H"
U0178 Lost Communication With "Restraints System Sensor I"
U0179 Lost Communication With "Restraints System Sensor J"
U0180 Lost Communication With Automatic Lighting Control Module
U0181 Lost Communication With Headlamp Leveling Control Module
U0182 Lost Communication With Lighting Control Module
U0183 Lost Communication With Lighting Control Module
U0184 Lost Communication With Radio

U0185 Lost Communication With Antenna Control Module
U0186 Lost Communication With Audio Amplifier
U0187 Lost Communication With Digital Disc Player/Changer Module "A"
U0188 Lost Communication With Digital Disc Player/Changer Module "B"
U0189 Lost Communication With Digital Disc Player/Changer Module "C"
U0190 Lost Communication With Digital Disc Player/Changer Module "D"
U0191 Lost Communication With Television
U0192 Lost Communication With Personal Computer
U0193 Lost Communication With "Digital Audio Control Module A"
U0194 Lost Communication With "Digital Audio Control Module B"
U0195 Lost Communication With Subscription Entertainment Receiver
Module
U0196 Lost Communication With Rear Seat Entertainment Control Module
U0197 Lost Communication With Telephone Control Module
U0198 Lost Communication With Telematic Control Module
U0199 Lost Communication With "Door Control Module A"
U0200 Lost Communication With "Door Control Module B"
U0201 Lost Communication With "Door Control Module C"
U0202 Lost Communication With "Door Control Module D"
U0203 Lost Communication With "Door Control Module E"
U0204 Lost Communication With "Door Control Module F"
U0205 Lost Communication With "Door Control Module G"
U0206 Lost Communication With Folding Top Control Module
U0207 Lost Communication With Movable Roof Control Module
U0208 Lost Communication With "Seat Control Module A"
U0209 Lost Communication With "Seat Control Module B"
U0210 Lost Communication With "Seat Control Module C"
U0211 Lost Communication With "Seat Control Module D"
U0212 Lost Communication With Steering Column Control Module
U0213 Lost Communication With Mirror Control Module
U0214 Lost Communication With Remote Function Actuation
U0215 Lost Communication With "Door Switch A"
U0216 Lost Communication With "Door Switch B"
U0217 Lost Communication With "Door Switch C"
U0218 Lost Communication With "Door Switch D"
U0219 Lost Communication With "Door Switch E"
U0220 Lost Communication With "Door Switch F"
U0221 Lost Communication With "Door Switch G"
U0222 Lost Communication With "Door Window Motor A"
U0223 Lost Communication With "Door Window Motor B"
U0224 Lost Communication With "Door Window Motor C"
U0225 Lost Communication With "Door Window Motor D"
U0226 Lost Communication With "Door Window Motor E"
U0227 Lost Communication With "Door Window Motor F"
U0228 Lost Communication With "Door Window Motor G"
U0229 Lost Communication With Heated Steering Wheel Module

U0230 Lost Communication With Rear Gate Module
U0231 Lost Communication With Rain Sensing Module
U0232 Lost Communication With Side Obstacle Detection Control Module
U0233 Lost Communication With Side Obstacle Detection Control Module
U0234 Lost Communication With Convenience Recall Module
U0235 Lost Communication With Cruise Control Front Distance Range Sensor
U0300 Internal Control Module Software Incompatibility
U0301 Software Incompatibility with ECM/PCM
U0302 Software Incompatibility with Transmission Control Module
U0303 Software Incompatibility with Transfer Case Control Module
U0304 Software Incompatibility with Gear Shift Control Module
U0305 Software Incompatibility with Cruise Control Module
U0306 Software Incompatibility with Fuel Injector Control Module
U0307 Software Incompatibility with Glow Plug Control Module
U0308 Software Incompatibility with Throttle Actuator Control Module
U0309 Software Incompatibility with Alternative Fuel Control Module
U0310 Software Incompatibility with Fuel Pump Control Module
U0311 Software Incompatibility with Drive Motor Control Module
U0312 Software Incompatibility with Battery Energy Control Module A
U0313 Software Incompatibility with Battery Energy Control Module B
U0314 Software Incompatibility with Four-Wheel Drive Clutch Control Module
U0315 Software Incompatibility with Anti-Lock Brake System Control Module
U0316 Software Incompatibility with Vehicle Dynamics Control Module
U0317 Software Incompatibility with Park Brake Control Module
U0318 Software Incompatibility with Brake System Control Module
U0319 Software Incompatibility with Steering Effort Control Module
U0320 Software Incompatibility with Power Steering Control Module
U0321 Software Incompatibility with Ride Level Control Module
U0322 Software Incompatibility with Body Control Module
U0323 Software Incompatibility with Instrument Panel Control Module
U0324 Software Incompatibility with HV AC Control Module
U0325 Software Incompatibility with Auxiliary Heater Control Module
U0326 Software Incompatibility with Vehicle Immobilizer Control Module
U0327 Software Incompatibility with Vehicle Security Control Module
U0328 Software Incompatibility with Steering Angle Sensor Module
U0329 Software Incompatibility with Steering Column Control Module
U0330 Software Incompatibility with Tire Pressure Monitor Module
U0331 Software Incompatibility with Body Control Module "A"
U0400 Invalid Data Received
U0401 Invalid Data Received From ECM/PCM
U0402 Invalid Data Received From Transmission Control Module
U0403 Invalid Data Received From Transfer Case Control Module
U0404 Invalid Data Received From Gear Shift Control Module
U0405 Invalid Data Received From Cruise Control Module
U0406 Invalid Data Received From Fuel Injector Control Module

U0407 Invalid Data Received From Glow Plug Control Module
U0408 Invalid Data Received From Throttle Actuator Control Module
U0409 Invalid Data Received From Alternative Fuel Control Module
U0410 Invalid Data Received From Fuel Pump Control Module
U0411 Invalid Data Received From Drive Motor Control Module
U0412 Invalid Data Received From Battery Energy Control Module A
U0413 Invalid Data Received From Battery Energy Control Module B
U0414 Invalid Data Received From Four-Wheel Drive Clutch Control Module
U0415 Invalid Data Received From Anti-Lock Brake System Control Module
U0416 Invalid Data Received From Vehicle Dynamics Control Module
U0417 Invalid Data Received From Park Brake Control Module
U0418 Invalid Data Received From Brake System Control Module
U0419 Invalid Data Received From Steering Effort Control Module
U0420 Invalid Data Received From Power Steering Control Module
U0421 Invalid Data Received From Ride Level Control Module
U0422 Invalid Data Received From Body Control Module
U0423 Invalid Data Received From Instrument Panel Control Module
U0424 Invalid Data Received From HVAC Control Module
U0425 Invalid Data Received From Auxiliary Heater Control Module
U0426 Invalid Data Received From Vehicle Immobilizer Control Module
U0427 Invalid Data Received From Vehicle Security Control Module
U0428 Invalid Data Received From Steering Angle Sensor Module
U0429 Invalid Data Received From Steering Column Control Module
U0430 Invalid Data Received From Tire Pressure Monitor Module
U0431 Invalid Data Received From Body Control Module "A"

ANEXO B: ELEMENTO PATH GLCD

```

move_to(-4,162, 32,715);
arc_to(59,484, 59,387, 86,61, 0, 1, -3,407, 32,315);
line_to(-3,407, 29,014);
arc_to(66,578, 66,47, -91,21, 0, 0, -4,162, 29,488);
line_to(-4,162, 32,715);
close_path();
move_to(-4,414, 29,65);
arc_to(66,578, 66,47, -91,21, 0, 0, -5,168, 30,144);
line_to(-5,168, 33,272);
arc_to(59,484, 59,387, 86,61, 0, 1, -4,414, 32,852);
line_to(-4,414, 29,65);
close_path();
move_to(-5,419, 30,312);
arc_to(66,578, 66,47, -91,21, 0, 0, -6,174, 30,826);
line_to(-6,174, 33,854);
arc_to(59,484, 59,387, 86,61, 0, 1, -5,419, 33,415);
line_to(-5,419, 30,312);
close_path();
move_to(-6,425, 31,001);
arc_to(66,578, 66,47, -91,21, 0, 0, -7,18, 31,535);
line_to(-7,18, 34,461);
arc_to(59,484, 59,387, 86,61, 0, 1, -6,425, 34,002);
line_to(-6,425, 31,001);
close_path();
move_to(-7,431, 31,718);
arc_to(66,578, 66,47, -91,21, 0, 0, -8,186, 32,274);
line_to(-8,186, 35,098);
arc_to(59,484, 59,387, 86,61, 0, 1, -7,431, 34,619);
line_to(-7,431, 31,718);
close_path();
move_to(-8,437, 32,463);
arc_to(66,578, 66,47, -91,21, 0, 0, -9,192, 33,041);
line_to(-9,192, 35,764);
arc_to(59,484, 59,387, 86,61, 0, 1, -8,437, 35,261);
line_to(-8,437, 32,463);
close_path();
move_to(-9,443, 33,238);
arc_to(66,578, 66,47, -91,21, 0, 0, -10,952, 34,46);
line_to(-10,952, 37,);
arc_to(59,484, 59,387, 86,61, 0, 1, -9,443, 35,934);
line_to(-9,443, 33,238);
close_path();
move_to(0,617, 26,708);
arc_to(66,578, 66,47, -91,21, 0, 0, -0,137, 27,112);
line_to(-0,137, 30,735);
arc_to(59,484, 59,387, 86,61, 0, 1, 0,617, 30,405);
line_to(0,617, 26,708);
close_path();
move_to(-0,389, 27,249);
arc_to(66,578, 66,47, -91,21, 0, 0, -1,143, 27,671);
line_to(-1,143, 31,195);

```



```

arc_to(59,484, 59,387, 86,61, 0, 1, -0,389, 30,848);
line_to(-0,389, 27,249);
close_path();
move_to(-1,395, 27,814);
arc_to(66,578, 66,47, -91,21, 0, 0, -2,149, 28,253);
line_to(-2,149, 31,678);
arc_to(59,484, 59,387, 86,61, 0, 1, -1,395, 31,313);
line_to(-1,395, 27,814);
close_path();
move_to(-2,401, 28,402);
arc_to(66,578, 66,47, -91,21, 0, 0, -3,155, 28,858);
line_to(-3,155, 32,185);
arc_to(59,484, 59,387, 86,61, 0, 1, -2,401, 31,802);
line_to(-2,401, 28,402);
close_path();
move_to(4,641, 24,756);
arc_to(66,578, 66,47, -91,21, 0, 0, 3,886, 25,096);
line_to(3,886, 29,114);
arc_to(59,484, 59,387, 86,61, 0, 1, 4,641, 28,847);
line_to(4,641, 24,756);
close_path();
move_to(3,635, 25,213);
arc_to(66,578, 66,47, -91,21, 0, 0, 2,88, 25,569);
line_to(2,88, 29,488);
arc_to(59,484, 59,387, 86,61, 0, 1, 3,635, 29,205);
line_to(3,635, 25,213);
close_path();
move_to(2,629, 25,69);
arc_to(66,578, 66,47, -91,21, 0, 0, 1,874, 26,062);
line_to(1,874, 29,882);
arc_to(59,484, 59,387, 86,61, 0, 1, 2,629, 29,584);
line_to(2,629, 25,69);
close_path();
move_to(1,623, 26,188);
arc_to(66,578, 66,47, -91,21, 0, 0, 0,869, 26,576);
line_to(0,869, 30,297);
arc_to(59,484, 59,387, 86,61, 0, 1, 1,623, 29,984);
line_to(1,623, 26,188);
close_path();
move_to(8,665, 23,123);
arc_to(66,578, 66,47, -91,21, 0, 0, 7,911, 23,406);
line_to(7,911, 27,821);
arc_to(59,484, 59,387, 86,61, 0, 1, 8,665, 27,613);
line_to(8,665, 23,123);
close_path();
move_to(7,659, 23,503);
arc_to(66,578, 66,47, -91,21, 0, 0, 6,905, 23,8);
line_to(6,905, 28,115);
arc_to(59,484, 59,387, 86,61, 0, 1, 7,659, 27,893);
line_to(7,659, 23,503);
close_path();

```

```

move_to(6,654, 23,902);
arc_to(66,578, 66,47, -91,21, 0, 0, 5,899, 24,213);
line_to(5,899, 28,428);
arc_to(59,484, 59,387, 86,61, 0, 1, 6,654, 28,191);
line_to(6,654, 23,902);
close_path();
move_to(5,647, 24,319);
arc_to(66,578, 66,47, -91,21, 0, 0, 4,892, 24,644);
line_to(4,892, 28,761);
arc_to(59,484, 59,387, 86,61, 0, 1, 5,647, 28,51);
line_to(5,647, 24,319);
close_path();
move_to(12,689, 21,787);
arc_to(66,578, 66,47, -91,21, 0, 0, 11,935, 22,016);
line_to(11,935, 26,836);
arc_to(59,484, 59,387, 86,61, 0, 1, 12,689, 26,684);
line_to(12,689, 21,787);
close_path();
move_to(11,683, 22,095);
arc_to(66,578, 66,47, -91,21, 0, 0, 10,929, 22,337);
line_to(10,929, 27,055);
arc_to(59,484, 59,387, 86,61, 0, 1, 11,683, 26,889);
line_to(11,683, 22,095);
close_path();
move_to(10,677, 22,42);
arc_to(66,578, 66,47, -91,21, 0, 0, 9,923, 22,675);
line_to(9,923, 27,292);
arc_to(59,484, 59,387, 86,61, 0, 1, 10,677, 27,112);
line_to(10,677, 22,42);
close_path();
move_to(9,671, 22,762);
arc_to(66,578, 66,47, -91,21, 0, 0, 8,917, 23,032);
line_to(8,917, 27,547);
arc_to(59,484, 59,387, 86,61, 0, 1, 9,671, 27,353);
line_to(9,671, 22,762);
close_path();
move_to(19,731, 20,108);
arc_to(66,578, 66,47, -91,21, 0, 0, 18,977, 20,249);
line_to(18,977, 25,806);
arc_to(59,484, 59,387, 86,61, 0, 1, 19,731, 25,747);
line_to(19,731, 20,108);
close_path();
move_to(18,726, 20,298);
arc_to(66,578, 66,47, -91,21, 0, 0, 17,971, 20,452);
line_to(17,971, 25,9);
arc_to(59,484, 59,387, 86,61, 0, 1, 18,726, 25,828);
line_to(18,726, 20,298);
close_path();
move_to(17,72, 20,505);
arc_to(66,578, 66,47, -91,21, 0, 0, 16,965, 20,67);
line_to(16,965, 26,013);

```

```
arc_to(59,484, 59,387, 86,61, 0, 1, 17,72, 25,927);
line_to(17,72, 20,505);
close_path();
move_to(16,714, 20,728);
arc_to(66,578, 66,47, -91,21, 0, 0, 15,959, 20,906);
line_to(15,959, 26,142);
arc_to(59,484, 59,387, 86,61, 0, 1, 16,714, 26,044);
line_to(16,714, 20,728);
close_path();
move_to(15,708, 20,967);
arc_to(66,578, 66,47, -91,21, 0, 0, 14,953, 21,158);
line_to(14,953, 26,289);
arc_to(59,484, 59,387, 86,61, 0, 1, 15,708, 26,177);
line_to(15,708, 20,967);
close_path();
move_to(14,701, 21,223);
arc_to(66,578, 66,47, -91,21, 0, 0, 13,947, 21,427);
line_to(13,947, 26,453);
arc_to(59,484, 59,387, 86,61, 0, 1, 14,701, 26,328);
line_to(14,701, 21,223);
close_path();
move_to(13,695, 21,497);
arc_to(66,578, 66,47, -91,21, 0, 0, 12,941, 21,713);
line_to(12,941, 26,636);
arc_to(59,484, 59,387, 86,61, 0, 1, 13,695, 26,497);
line_to(13,695, 21,497);
close_path();
move_to(41,864, 19,841);
arc_to(66,578, 66,47, -91,21, 0, 0, 41,109, 19,727);
line_to(41,109, 28,108);
arc_to(59,484, 59,387, 86,61, 0, 1, 41,864, 28,34);
line_to(41,864, 19,841);
close_path();
move_to(40,103, 27,815);
arc_to(59,484, 59,387, 86,61, 0, 1, 40,858, 28,032);
line_to(40,858, 19,691);
arc_to(66,578, 66,47, -91,21, 0, 0, 40,103, 19,589);
line_to(40,103, 27,815);
close_path();
move_to(39,852, 19,557);
arc_to(66,578, 66,47, -91,21, 0, 0, 39,097, 19,466);
line_to(39,097, 27,541);
arc_to(59,484, 59,387, 86,61, 0, 1, 39,852, 27,744);
line_to(39,852, 19,557);
close_path();
move_to(38,091, 27,286);
arc_to(59,484, 59,387, 86,61, 0, 1, 38,846, 27,475);
line_to(38,846, 19,438);
arc_to(66,578, 66,47, -91,21, 0, 0, 38,091, 19,359);
line_to(38,091, 27,286);
close_path();
```

```
move_to(37,84, 19,335);
arc_to(66,578, 66,47, -91,21, 0, 0, 37,086, 19,269);
line_to(37,086, 27,049);
arc_to(59,484, 59,387, 86,61, 0, 1, 37,84, 27,225);
line_to(37,84, 19,335);
close_path();
move_to(36,08, 26,831);
arc_to(59,484, 59,387, 86,61, 0, 1, 36,834, 26,993);
line_to(36,834, 19,248);
arc_to(66,578, 66,47, -91,21, 0, 0, 36,08, 19,192);
line_to(36,08, 26,831);
close_path();
move_to(32,055, 26,138);
arc_to(59,484, 59,387, 86,61, 0, 1, 32,81, 26,246);
line_to(32,81, 19,052);
arc_to(66,578, 66,47, -91,21, 0, 0, 32,055, 19,042);
line_to(32,055, 26,138);
close_path();
move_to(31,803, 19,04);
arc_to(66,578, 66,47, -91,21, 0, 0, 31,049, 19,042);
line_to(31,049, 26,009);
arc_to(59,484, 59,387, 86,61, 0, 1, 31,803, 26,105);
line_to(31,803, 19,04);
line_to(31,803, 19,04);
close_path();
move_to(30,043, 25,898);
arc_to(59,484, 59,387, 86,61, 0, 1, 30,798, 25,98);
line_to(30,798, 19,04);
arc_to(56,391, 56,299, 88,79, 0, 1, 30,043, 19,058);
line_to(30,043, 25,898);
close_path();
move_to(29,792, 19,064);
arc_to(66,578, 66,47, -91,21, 0, 0, 29,037, 19,089);
line_to(29,037, 25,804);
arc_to(59,484, 59,387, 86,61, 0, 1, 29,792, 25,873);
line_to(29,792, 19,064);
close_path();
move_to(28,786, 19,099);
arc_to(66,578, 66,47, -91,21, 0, 0, 28,031, 19,136);
line_to(28,031, 25,728);
arc_to(59,484, 59,387, 86,61, 0, 1, 28,786, 25,783);
line_to(28,786, 19,099);
close_path();
move_to(24,762, 19,392);
arc_to(66,578, 66,47, -91,21, 0, 0, 24,007, 19,474);
line_to(24,007, 25,591);
arc_to(59,484, 59,387, 86,61, 0, 1, 24,762, 25,596);
line_to(24,762, 19,392);
close_path();
move_to(23,756, 19,504);
arc_to(66,578, 66,47, -91,21, 0, 0, 23,001, 19,598);
```

```

line_to(23,001, 25,6);
arc_to(59,484, 59,387, 86,61, 0, 1, 23,756, 25,592);
line_to(23,756, 19,504);
close_path();
move_to(21,995, 25,626);
arc_to(59,484, 59,387, 86,61, 0, 1, 22,749, 25,605);
line_to(22,749, 19,631);
arc_to(66,578, 66,47, -91,21, 0, 0, 21,995, 19,737);
line_to(21,995, 25,626);
close_path();
move_to(21,743, 19,774);
arc_to(66,578, 66,47, -91,21, 0, 0, 20,989, 19,891);
line_to(20,989, 25,669);
arc_to(59,484, 59,387, 86,61, 0, 1, 21,743, 25,635);
line_to(21,743, 19,774);
close_path();
move_to(20,737, 19,933);
arc_to(66,578, 66,47, -91,21, 0, 0, 19,983, 20,062);
line_to(19,983, 25,729);
arc_to(59,484, 59,387, 86,61, 0, 1, 20,737, 25,682);
line_to(20,737, 19,933);
close_path();
move_to(27,78, 19,149);
arc_to(66,578, 66,47, -91,21, 0, 0, 27,025, 19,197);
line_to(27,025, 25,668);
arc_to(59,484, 59,387, 86,61, 0, 1, 27,78, 25,711);
line_to(27,78, 19,149);
close_path();
move_to(26,774, 19,215);
arc_to(66,578, 66,47, -91,21, 0, 0, 26,019, 19,274);
line_to(26,019, 25,625);
arc_to(59,484, 59,387, 86,61, 0, 1, 26,774, 25,655);
line_to(26,774, 19,215);
close_path();
move_to(25,768, 19,295);
arc_to(66,578, 66,47, -91,21, 0, 0, 25,014, 19,366);
line_to(25,014, 25,6);
arc_to(59,484, 59,387, 86,61, 0, 1, 25,768, 25,617);
line_to(25,768, 19,295);
close_path();
move_to(35,828, 19,175);
arc_to(66,578, 66,47, -91,21, 0, 0, 35,074, 19,131);
line_to(35,074, 26,631);
arc_to(59,484, 59,387, 86,61, 0, 1, 35,828, 26,779);
line_to(35,828, 19,175);
close_path();
move_to(34,068, 26,449);
arc_to(59,484, 59,387, 86,61, 0, 1, 34,822, 26,584);
line_to(34,822, 19,119);
arc_to(66,578, 66,47, -91,21, 0, 0, 34,068, 19,086);
line_to(34,068, 26,449);

```

```

close_path();
move_to(33,816, 19,077);
arc_to(66,578, 66,47, -91,21, 0, 0, 33,062, 19,057);
line_to(33,062, 26,285);
arc_to(59,484, 59,387, 86,61, 0, 1, 33,816, 26,406);
line_to(33,816, 19,077);
close_path();
move_to(49,158, 31,183);
arc_to(59,484, 59,387, 86,61, 0, 1, 49,912, 31,543);
line_to(49,912, 21,62);
arc_to(66,578, 66,47, -91,21, 0, 0, 49,158, 21,409);
line_to(49,158, 31,183);
close_path();
move_to(48,906, 21,34);
arc_to(66,578, 66,47, -91,21, 0, 0, 48,152, 21,141);
line_to(48,152, 30,724);
arc_to(59,484, 59,387, 86,61, 0, 1, 48,906, 31,067);
line_to(48,906, 21,34);
close_path();
move_to(47,9, 21,076);
arc_to(66,578, 66,47, -91,21, 0, 0, 47,146, 20,89);
line_to(47,146, 30,287);
arc_to(59,484, 59,387, 86,61, 0, 1, 47,9, 30,613);
line_to(47,9, 21,076);
close_path();
move_to(46,14, 29,872);
arc_to(59,484, 59,387, 86,61, 0, 1, 46,894, 30,181);
line_to(46,894, 20,829);
arc_to(66,578, 66,47, -91,21, 0, 0, 46,14, 20,655);
line_to(46,14, 29,872);
close_path();
move_to(45,888, 20,6);
arc_to(66,578, 66,47, -91,21, 0, 0, 45,134, 20,438);
line_to(45,134, 29,478);
arc_to(59,484, 59,387, 86,61, 0, 1, 45,888, 29,771);
line_to(45,888, 20,6);
close_path();
move_to(44,128, 29,105);
arc_to(59,484, 59,387, 86,61, 0, 1, 44,882, 29,383);
line_to(44,882, 20,385);
arc_to(66,578, 66,47, -91,21, 0, 0, 44,128, 20,236);
line_to(44,128, 29,105);
close_path();
move_to(43,876, 20,188);
arc_to(66,578, 66,47, -91,21, 0, 0, 43,122, 20,05);
line_to(43,122, 28,752);
arc_to(59,484, 59,387, 86,61, 0, 1, 43,876, 29,015);
line_to(43,876, 20,188);
close_path();
move_to(42,116, 28,42);
arc_to(59,484, 59,387, 86,61, 0, 1, 42,871, 28,668);

```

```

line_to(42,871, 20,006);
arc_to(66,578, 66,47, -91,21, 0, 0, 42,116, 19,881);
line_to(42,116, 28,42);
close_path();
move_to(-4,162, 32,715);
arc_to(59,484, 59,387, 86,61, 0, 1, -3,407, 32,315);
line_to(-3,407, 29,014);
arc_to(66,578, 66,47, -91,21, 0, 0, -4,162, 29,488);
line_to(-4,162, 32,715);
move_to(-4,414, 29,65);
arc_to(66,578, 66,47, -91,21, 0, 0, -5,168, 30,144);
line_to(-5,168, 33,272);
arc_to(59,484, 59,387, 86,61, 0, 1, -4,414, 32,852);
line_to(-4,414, 29,65);
move_to(-5,419, 30,312);
arc_to(66,578, 66,47, -91,21, 0, 0, -6,174, 30,826);
line_to(-6,174, 33,854);
arc_to(59,484, 59,387, 86,61, 0, 1, -5,419, 33,415);
line_to(-5,419, 30,312);
move_to(-6,425, 31,001);
arc_to(66,578, 66,47, -91,21, 0, 0, -7,18, 31,535);
line_to(-7,18, 34,461);
arc_to(59,484, 59,387, 86,61, 0, 1, -6,425, 34,002);
line_to(-6,425, 31,001);
move_to(-7,431, 31,718);
arc_to(66,578, 66,47, -91,21, 0, 0, -8,186, 32,274);
line_to(-8,186, 35,098);
arc_to(59,484, 59,387, 86,61, 0, 1, -7,431, 34,619);
line_to(-7,431, 31,718);
move_to(-8,437, 32,463);
arc_to(66,578, 66,47, -91,21, 0, 0, -9,192, 33,041);
line_to(-9,192, 35,764);
arc_to(59,484, 59,387, 86,61, 0, 1, -8,437, 35,261);
line_to(-8,437, 32,463);
move_to(-9,443, 33,238);
arc_to(66,578, 66,47, -91,21, 0, 0, -10,952, 34,46);
line_to(-10,952, 37,);
arc_to(59,484, 59,387, 86,61, 0, 1, -9,443, 35,934);
line_to(-9,443, 33,238);
move_to(0,617, 26,708);
arc_to(66,578, 66,47, -91,21, 0, 0, -0,137, 27,112);
line_to(-0,137, 30,735);
arc_to(59,484, 59,387, 86,61, 0, 1, 0,617, 30,405);
line_to(0,617, 26,708);
move_to(-0,389, 27,249);
arc_to(66,578, 66,47, -91,21, 0, 0, -1,143, 27,671);
line_to(-1,143, 31,195);
arc_to(59,484, 59,387, 86,61, 0, 1, -0,389, 30,848);
line_to(-0,389, 27,249);
move_to(-1,395, 27,814);
arc_to(66,578, 66,47, -91,21, 0, 0, -2,149, 28,253);

```

line_to(-2,149, 31,678);
arc_to(59,484, 59,387, 86,61, 0, 1, -1,395, 31,313);
line_to(-1,395, 27,814);
move_to(-2,401, 28,402);
arc_to(66,578, 66,47, -91,21, 0, 0, -3,155, 28,858);
line_to(-3,155, 32,185);
arc_to(59,484, 59,387, 86,61, 0, 1, -2,401, 31,802);
line_to(-2,401, 28,402);
move_to(4,641, 24,756);
arc_to(66,578, 66,47, -91,21, 0, 0, 3,886, 25,096);
line_to(3,886, 29,114);
arc_to(59,484, 59,387, 86,61, 0, 1, 4,641, 28,847);
line_to(4,641, 24,756);
move_to(3,635, 25,213);
arc_to(66,578, 66,47, -91,21, 0, 0, 2,88, 25,569);
line_to(2,88, 29,488);
arc_to(59,484, 59,387, 86,61, 0, 1, 3,635, 29,205);
line_to(3,635, 25,213);
move_to(2,629, 25,69);
arc_to(66,578, 66,47, -91,21, 0, 0, 1,874, 26,062);
line_to(1,874, 29,882);
arc_to(59,484, 59,387, 86,61, 0, 1, 2,629, 29,584);
line_to(2,629, 25,69);
move_to(1,623, 26,188);
arc_to(66,578, 66,47, -91,21, 0, 0, 0,869, 26,576);
line_to(0,869, 30,297);
arc_to(59,484, 59,387, 86,61, 0, 1, 1,623, 29,984);
line_to(1,623, 26,188);
move_to(8,665, 23,123);
arc_to(66,578, 66,47, -91,21, 0, 0, 7,911, 23,406);
line_to(7,911, 27,821);
arc_to(59,484, 59,387, 86,61, 0, 1, 8,665, 27,613);
line_to(8,665, 23,123);
move_to(7,659, 23,503);
arc_to(66,578, 66,47, -91,21, 0, 0, 6,905, 23,8);
line_to(6,905, 28,115);
arc_to(59,484, 59,387, 86,61, 0, 1, 7,659, 27,893);
line_to(7,659, 23,503);
move_to(6,654, 23,902);
arc_to(66,578, 66,47, -91,21, 0, 0, 5,899, 24,213);
line_to(5,899, 28,428);
arc_to(59,484, 59,387, 86,61, 0, 1, 6,654, 28,191);
line_to(6,654, 23,902);
move_to(5,647, 24,319);
arc_to(66,578, 66,47, -91,21, 0, 0, 4,892, 24,644);
line_to(4,892, 28,761);
arc_to(59,484, 59,387, 86,61, 0, 1, 5,647, 28,51);
line_to(5,647, 24,319);
move_to(12,689, 21,787);
arc_to(66,578, 66,47, -91,21, 0, 0, 11,935, 22,016);
line_to(11,935, 26,836);

arc_to(59,484, 59,387, 86,61, 0, 1, 12,689, 26,684);
line_to(12,689, 21,787);
move_to(11,683, 22,095);
arc_to(66,578, 66,47, -91,21, 0, 0, 10,929, 22,337);
line_to(10,929, 27,055);
arc_to(59,484, 59,387, 86,61, 0, 1, 11,683, 26,889);
line_to(11,683, 22,095);
move_to(10,677, 22,42);
arc_to(66,578, 66,47, -91,21, 0, 0, 9,923, 22,675);
line_to(9,923, 27,292);
arc_to(59,484, 59,387, 86,61, 0, 1, 10,677, 27,112);
line_to(10,677, 22,42);
move_to(9,671, 22,762);
arc_to(66,578, 66,47, -91,21, 0, 0, 8,917, 23,032);
line_to(8,917, 27,547);
arc_to(59,484, 59,387, 86,61, 0, 1, 9,671, 27,353);
line_to(9,671, 22,762);
move_to(19,731, 20,108);
arc_to(66,578, 66,47, -91,21, 0, 0, 18,977, 20,249);
line_to(18,977, 25,806);
arc_to(59,484, 59,387, 86,61, 0, 1, 19,731, 25,747);
line_to(19,731, 20,108);
move_to(18,726, 20,298);
arc_to(66,578, 66,47, -91,21, 0, 0, 17,971, 20,452);
line_to(17,971, 25,9);
arc_to(59,484, 59,387, 86,61, 0, 1, 18,726, 25,828);
line_to(18,726, 20,298);
move_to(17,72, 20,505);
arc_to(66,578, 66,47, -91,21, 0, 0, 16,965, 20,67);
line_to(16,965, 26,013);
arc_to(59,484, 59,387, 86,61, 0, 1, 17,72, 25,927);
line_to(17,72, 20,505);
move_to(16,714, 20,728);
arc_to(66,578, 66,47, -91,21, 0, 0, 15,959, 20,906);
line_to(15,959, 26,142);
arc_to(59,484, 59,387, 86,61, 0, 1, 16,714, 26,044);
line_to(16,714, 20,728);
move_to(15,708, 20,967);
arc_to(66,578, 66,47, -91,21, 0, 0, 14,953, 21,158);
line_to(14,953, 26,289);
arc_to(59,484, 59,387, 86,61, 0, 1, 15,708, 26,177);
line_to(15,708, 20,967);
move_to(14,701, 21,223);
arc_to(66,578, 66,47, -91,21, 0, 0, 13,947, 21,427);
line_to(13,947, 26,453);
arc_to(59,484, 59,387, 86,61, 0, 1, 14,701, 26,328);
line_to(14,701, 21,223);
move_to(13,695, 21,497);
arc_to(66,578, 66,47, -91,21, 0, 0, 12,941, 21,713);
line_to(12,941, 26,636);
arc_to(59,484, 59,387, 86,61, 0, 1, 13,695, 26,497);

line_to(13,695, 21,497);
move_to(41,864, 19,841);
arc_to(66,578, 66,47, -91,21, 0, 0, 41,109, 19,727);
line_to(41,109, 28,108);
arc_to(59,484, 59,387, 86,61, 0, 1, 41,864, 28,34);
line_to(41,864, 19,841);
move_to(40,103, 27,815);
arc_to(59,484, 59,387, 86,61, 0, 1, 40,858, 28,032);
line_to(40,858, 19,691);
arc_to(66,578, 66,47, -91,21, 0, 0, 40,103, 19,589);
line_to(40,103, 27,815);
move_to(39,852, 19,557);
arc_to(66,578, 66,47, -91,21, 0, 0, 39,097, 19,466);
line_to(39,097, 27,541);
arc_to(59,484, 59,387, 86,61, 0, 1, 39,852, 27,744);
line_to(39,852, 19,557);
move_to(38,091, 27,286);
arc_to(59,484, 59,387, 86,61, 0, 1, 38,846, 27,475);
line_to(38,846, 19,438);
arc_to(66,578, 66,47, -91,21, 0, 0, 38,091, 19,359);
line_to(38,091, 27,286);
move_to(37,84, 19,335);
arc_to(66,578, 66,47, -91,21, 0, 0, 37,086, 19,269);
line_to(37,086, 27,049);
arc_to(59,484, 59,387, 86,61, 0, 1, 37,84, 27,225);
line_to(37,84, 19,335);
move_to(36,08, 26,831);
arc_to(59,484, 59,387, 86,61, 0, 1, 36,834, 26,993);
line_to(36,834, 19,248);
arc_to(66,578, 66,47, -91,21, 0, 0, 36,08, 19,192);
line_to(36,08, 26,831);
move_to(32,055, 26,138);
arc_to(59,484, 59,387, 86,61, 0, 1, 32,81, 26,246);
line_to(32,81, 19,052);
arc_to(66,578, 66,47, -91,21, 0, 0, 32,055, 19,042);
line_to(32,055, 26,138);
move_to(31,803, 19,04);
arc_to(66,578, 66,47, -91,21, 0, 0, 31,049, 19,042);
line_to(31,049, 26,009);
arc_to(59,484, 59,387, 86,61, 0, 1, 31,803, 26,105);
line_to(31,803, 19,04);
move_to(30,043, 25,898);
arc_to(59,484, 59,387, 86,61, 0, 1, 30,798, 25,98);
line_to(30,798, 19,04);
arc_to(56,391, 56,299, 88,79, 0, 1, 30,043, 19,058);
line_to(30,043, 25,898);
move_to(29,792, 19,064);
arc_to(66,578, 66,47, -91,21, 0, 0, 29,037, 19,089);
line_to(29,037, 25,804);
arc_to(59,484, 59,387, 86,61, 0, 1, 29,792, 25,873);
line_to(29,792, 19,064);

move_to(28,786, 19,099);
arc_to(66,578, 66,47, -91,21, 0, 0, 28,031, 19,136);
line_to(28,031, 25,728);
arc_to(59,484, 59,387, 86,61, 0, 1, 28,786, 25,783);
line_to(28,786, 19,099);
move_to(24,762, 19,392);
arc_to(66,578, 66,47, -91,21, 0, 0, 24,007, 19,474);
line_to(24,007, 25,591);
arc_to(59,484, 59,387, 86,61, 0, 1, 24,762, 25,596);
line_to(24,762, 19,392);
move_to(23,756, 19,504);
arc_to(66,578, 66,47, -91,21, 0, 0, 23,001, 19,598);
line_to(23,001, 25,6);
arc_to(59,484, 59,387, 86,61, 0, 1, 23,756, 25,592);
line_to(23,756, 19,504);
move_to(21,995, 25,626);
arc_to(59,484, 59,387, 86,61, 0, 1, 22,749, 25,605);
line_to(22,749, 19,631);
arc_to(66,578, 66,47, -91,21, 0, 0, 21,995, 19,737);
line_to(21,995, 25,626);
move_to(21,743, 19,774);
arc_to(66,578, 66,47, -91,21, 0, 0, 20,989, 19,891);
line_to(20,989, 25,669);
arc_to(59,484, 59,387, 86,61, 0, 1, 21,743, 25,635);
line_to(21,743, 19,774);
move_to(20,737, 19,933);
arc_to(66,578, 66,47, -91,21, 0, 0, 19,983, 20,062);
line_to(19,983, 25,729);
arc_to(59,484, 59,387, 86,61, 0, 1, 20,737, 25,682);
line_to(20,737, 19,933);
move_to(27,78, 19,149);
arc_to(66,578, 66,47, -91,21, 0, 0, 27,025, 19,197);
line_to(27,025, 25,668);
arc_to(59,484, 59,387, 86,61, 0, 1, 27,78, 25,711);
line_to(27,78, 19,149);
move_to(26,774, 19,215);
arc_to(66,578, 66,47, -91,21, 0, 0, 26,019, 19,274);
line_to(26,019, 25,625);
arc_to(59,484, 59,387, 86,61, 0, 1, 26,774, 25,655);
line_to(26,774, 19,215);
move_to(25,768, 19,295);
arc_to(66,578, 66,47, -91,21, 0, 0, 25,014, 19,366);
line_to(25,014, 25,6);
arc_to(59,484, 59,387, 86,61, 0, 1, 25,768, 25,617);
line_to(25,768, 19,295);
move_to(35,828, 19,175);
arc_to(66,578, 66,47, -91,21, 0, 0, 35,074, 19,131);
line_to(35,074, 26,631);
arc_to(59,484, 59,387, 86,61, 0, 1, 35,828, 26,779);
line_to(35,828, 19,175);
move_to(34,068, 26,449);

```
arc_to(59,484, 59,387, 86,61, 0, 1, 34,822, 26,584);
line_to(34,822, 19,119);
arc_to(66,578, 66,47, -91,21, 0, 0, 34,068, 19,086);
line_to(34,068, 26,449);
move_to(33,816, 19,077);
arc_to(66,578, 66,47, -91,21, 0, 0, 33,062, 19,057);
line_to(33,062, 26,285);
arc_to(59,484, 59,387, 86,61, 0, 1, 33,816, 26,406);
line_to(33,816, 19,077);
move_to(49,158, 31,183);
arc_to(59,484, 59,387, 86,61, 0, 1, 49,912, 31,543);
line_to(49,912, 21,62);
arc_to(66,578, 66,47, -91,21, 0, 0, 49,158, 21,409);
line_to(49,158, 31,183);
move_to(48,906, 21,34);
arc_to(66,578, 66,47, -91,21, 0, 0, 48,152, 21,141);
line_to(48,152, 30,724);
arc_to(59,484, 59,387, 86,61, 0, 1, 48,906, 31,067);
line_to(48,906, 21,34);
move_to(47,9, 21,076);
arc_to(66,578, 66,47, -91,21, 0, 0, 47,146, 20,89);
line_to(47,146, 30,287);
arc_to(59,484, 59,387, 86,61, 0, 1, 47,9, 30,613);
line_to(47,9, 21,076);
move_to(46,14, 29,872);
arc_to(59,484, 59,387, 86,61, 0, 1, 46,894, 30,181);
line_to(46,894, 20,829);
arc_to(66,578, 66,47, -91,21, 0, 0, 46,14, 20,655);
line_to(46,14, 29,872);
move_to(45,888, 20,6);
arc_to(66,578, 66,47, -91,21, 0, 0, 45,134, 20,438);
line_to(45,134, 29,478);
arc_to(59,484, 59,387, 86,61, 0, 1, 45,888, 29,771);
line_to(45,888, 20,6);
move_to(44,128, 29,105);
arc_to(59,484, 59,387, 86,61, 0, 1, 44,882, 29,383);
line_to(44,882, 20,385);
arc_to(66,578, 66,47, -91,21, 0, 0, 44,128, 20,236);
line_to(44,128, 29,105);
move_to(43,876, 20,188);
arc_to(66,578, 66,47, -91,21, 0, 0, 43,122, 20,05);
line_to(43,122, 28,752);
arc_to(59,484, 59,387, 86,61, 0, 1, 43,876, 29,015);
line_to(43,876, 20,188);
move_to(42,116, 28,42);
arc_to(59,484, 59,387, 86,61, 0, 1, 42,871, 28,668);
line_to(42,871, 20,006);
arc_to(66,578, 66,47, -91,21, 0, 0, 42,116, 19,881);
line_to(42,116, 28,42);
close_path();
```

ANEXO C: CRONOGRAMA

ANEXO D: ARTICULO DEL PROYECTO

DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE DIAGNÓSTICO PREDICTIVO PARA AUTOMOTORES BAJO NORMA OBD II, CON CONEXIÓN INALÁMBRICA WI-FI Y CONTROLADORES TÁCTILES DE LA MARCA APPLE.

Ing. John Jairo Ceballos, Ing. Germán Erazo, Ing. Alexandra Corral
Departamento de Ciencias de la Energía y Mecánica
Quijano y Ordoñez y Marquez de Maéza s/n
Latacunga - Ecuador
Email : rave_angel_2107@hotmail.com



Resumen.

El presente proyecto facilitará el estudio, análisis e interpretación de los diferentes sistemas eléctricos, electrónicos y mecánicos de los diferentes tipos y marcas de automóviles, que mantengan un sistema de comunicación OBD II. Los profesionales y estudiantes tendrían una herramienta táctil, con una interfaz gráfica (GUI) muy amigable y fácil de manejar, comunicación universal con los tipos específicos de conexión normados internacionalmente, bajo el desarrollo de programación informática nativa de Apple Objective-C.

I. INTRODUCCIÓN.

El incremento en el número de componentes del vehículo controlados electrónicamente mediante ECU es continuo. Si bien, la cooperación entre esas ECU y el acceso a la información que circula por ellas, o entre ellas y el conjunto de sensores y actuadores, sigue quedando restringido a los fabricantes de vehículos y las grandes empresas fabricantes de componentes, es necesario la creación de nuevas interfaces intuitivas y de rápido acceso de comunicación con estos componentes, para la utilización de ingenieros y aficionados en el campo automotriz, en las áreas correspondientes al diagnóstico y el confort.

En este concepto, la utilización de sistemas de comunicación con redes de tipo inalámbricas, nos sería el tipo de enlace apto para el intercambio de información, tomando en cuenta los tiempos de respuesta, que en este caso son muy altos, y los medios en los cuales van a ser instalados.

La introducción de sistemas táctiles de la marca Apple, ha contribuido a la mejora de los sistemas inalámbricos, con aplicaciones desarrolladas para actividades específicas de industrias especiales. En este caso, en

la industria automotriz, permite mantener un sistema de información organizada y universal, así como nos permite establecer una base de datos determinada para cada unidad y marca, la misma que será utilizada en casos de revisiones, o de diagnósticos de tipo preventivo, cuando el usuario así lo requiera.

Toda la estructura se basa en el manejo del protocolo OBDII respectivo (ISO 9141-2, SAE J1850 VPW, SAE J1850 PWM, ISO 15765 (CAN), etc), por lo que es requerimiento indispensable establecer un lenguaje informático optimizado para su enlace, en este caso inalámbrico.

La utilización del lenguaje Objective-C es la herramienta funcional adecuada debido a sus características de alto desempeño dentro de conexiones wireless, y su tasa rápida de respuesta tanto en conexión de subida como de bajada.

Físicamente, el esquema de conexión, está compuesto por una interfaz de lectura de códigos de tipo OBD II con generador de señal de emisión de datos Wi-Fi, un software de aplicación desarrollado en una interfaz única de usuario (NUI) y compilado en Objective-C, un dispositivo táctil de la marca Apple (iPhone, Ipad), y una red establecida de registro de información periódica e individual para cada modelo y tipo de automotor analizado.

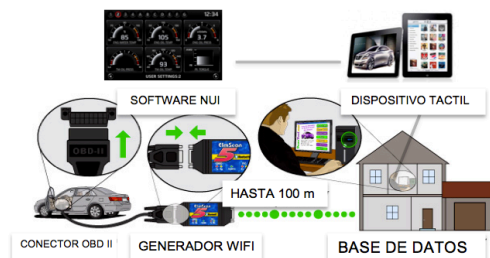


Figura 1. Diagrama de Bloques de un Sistema de Comunicación Inalámbrica

Para los docentes y técnicos especializados en los laboratorios de Mecánica y Prototipos, además de los especialistas en el área de Autotrónica, este sistema les permitiría, aparte de la comunicación directa con el automóvil, la interpretación de códigos DTC's, con su respectivo despliegue y borrado. La adición de un módulo de seguridad integral del dispositivo táctil, al mismo tiempo permitiría una investigación más exhaustiva en el área de seguridad automotriz.

Aprovechando el diseño del que fue hecho el dispositivo táctil, podemos utilizar herramientas tecnológicas de medición de avanzada, como son los acelerómetros micrométricos y los inclinómetros y giroscopios digitales, de gran importancia en el análisis tanto de vibraciones y oscilaciones causadas por el movimiento y la velocidad, las fuerzas G que soporta tanto en aceleración y frenado, en el caso del acelerómetro; y el nivel de sujeción al suelo, y viraje del automóvil tanto frontal, posterior y lateralmente.

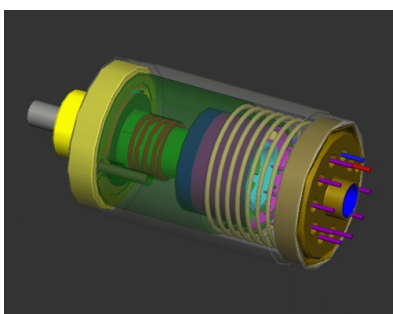


Figura 2. Acelerómetro digital micrométrico

La utilización del diseño realizado nos permite recibir información de los siguientes parámetros controlados por la ECU, dependiendo del nivel de tecnología del que este compuesto:

- Velocidad del motor.
- Compresión y vacío.
- Aceleración, poder y torque.
- Consumo de combustible.
- Avance al encendido.
- Temperatura del refrigerante.
- MAP
- MAF
- Ajuste del combustible.
- Presión del combustible.
- Relación aire/combustible, etc.

II. DISEÑO DE LA RED DE COMUNICACIÓN

Nuestra red de enlace, entre dispositivo táctil e interfaz de generación de señal, se ve optimizada, tanto en comunicación de envío como de recepción, con la utilización de la interconexión de tipo WI-FI ya que, en comparación de el resto de tipos de tecnologías de comunicación, nos da las siguientes características:

- **Alcance** hasta 100 mts en lugares abiertos, dependiendo de estado meteorológicos y otros factores de recepción, como alcance del generador de señal, alcance del receptor, etc.
- **Fiabilidad** alta, con la utilización de una máscara de subnet y una dirección IP estática propia, y un puerto de comunicación dedicado para el envío y recepción de bits de información.
- **Ancho de Banda** de 3 Mbps nominal, más que suficiente para cadenas de datos que van desde 120kb hasta 1Mb como tope de información.
- **Obstáculos e interferencias** que ésta red puede sobrepasar, los mismos que son mínimos, ya que el automóvil no está diseñado como un espacio totalmente cerrado para la comunicación de radiofrecuencias.
- **Seguridad de transmisión** mediana para ser una medio inalámbrico.
- **Programación** compatible con la empleada en el dispositivo, que es de tipo Objective-C, nativa de aplicaciones UNIX para dispositivos táctiles.
- **Protocolos de comunicación** IEEE 802.11b, óptimos con el receptor y el generador de señal, que utilizan protocolos de tipo estándar 802.11a/b/g. Protocolos de comunicación.

Determinado el ambiente en el que se va a desarrollar el enlace, para la conexión entre interfaz y módulo de control, primero se debe inicializar el OBDNet o red informática encriptada OBDII. Para realizar este proceso en Objective-C se llama a la función OBDNetInitialize, la misma que analiza y establece que tipo de lenguaje de programación informática se esta

utilizando, para enviar información que sea de lectura para nuestro respectivo protocolo.

Después de haber inicializado OBDNet, se necesita crear una nueva función con el nombre OBDNET_HANDLE para identificar nuestra conexión, en este caso inalámbrica WI-FI. Con ésta nosotros podemos extender nuestra red punto a punto a una red en malla, para no solamente tener un dispositivo táctil al alcance, sino varios tablets recibiendo y enviando información al mismo tiempo.

Para finalizar el proceso, nuestra IPA llama las siguientes funciones complementarias:

- **OBDNetCreate** para crear una red inicial con valores en tiempo real fundamental para que, en el manejo de varios vehículos de estudio, no permita la conjunción de datos entre unidades analizadas.
- **OBDNetWorker** para designar parte de la programación a cada campo de análisis que nuestra interfaz requiera, por ejemplo el estudio de PIDs, GLCDs, medidores de tiempo real, y parte de diagnóstico predictivo.

III. DESENCRIPTACIÓN DE INFORMACIÓN EN OBJETIVE-C

Ya inicializado nuestro enlace, la programación se debe conectar al respectivo servidor de envío de información, creado entre nuestro generador de señal y la salida del puerto OBDII del automóvil. Este servidor de envío es diferente de acuerdo a la complejidad de la ECU, y a la cantidad de dispositivos electrónicos que esta maneja.

La función OBDNetConnection nos permite entrar en este servidor, cuyo único requisito es que tengamos OBDNET_HANDLE activo como parte de la red.

Como paso inicial de la obtención de datos, la ECU siempre envía primero una lista completa de PIDs activos, si este fuera el caso, como primer parámetro de análisis. La función OBDNetGetPidList realiza este estudio, con información procedente de datos enviados de cada PID por la función OBDNET_CALLBACK_PID_LIST_ITEM.

Como ejemplo de la descriptación de información, tomaremos un valor tipo de gran utilidad, en este caso el OBDNET_PID_VALUE

USO	
<p>Establece el valor real de lectura informática en bits del parámetro PID que se requiera. Se puede obtener mediante la utilización de la función OBDNetParameterGetCurrentValue o mediante el callback de definición #define OBDNET_CALLBACK_STREAMING_PID_VALUE.</p>	
SINTAXIS	
<pre>typedef struct { unsigned long PidValueType; unsigned long TimeStamp; double EnglishValue; double MetricValue; const char * EnglishValueString; const char * MetricValueString; const char * UniqueId; } OBDNET_PID_VALUE;</pre>	
PARAMETROS	
<ul style="list-style-type: none"> • PidValueType es el tipo de dato que la estructura contiene. Puede ser OBDNET_PID_VALUETYPE_NUMBER, en caso de valores numéricos únicos, POBDII_PID_VALUETYPE_STRING en caso de una cadena de valores específicos OBDII. Además indica si este valor es valido para EnglishValue - EnglishValueString en el sistema inglés de medición, o MetricValue - MetricValueString en el sistema métrico. • TimeStamp establece el número de milisegundos desde que la petición de información. • EnglishValue, el valor en el Sistema Inglés, si el PidValueType es POBDII_PID_VALUETYPE_NUMBER. • MetricValue, el valor en el Sistema Métrico, si el PidValueType es POBDII_PID_VALUETYPE_NUMBER. • EnglishValueString, una cadena de valores en el sistema inglés, si el PidValueType es POBDII_PID_VALUETYPE_STRING. 	

- MetricValueString, una cadena de valores en el sistema métrico, si el PidValueType es POBDIL_PID_VALUETYPE_STRING.
- Uniqueld identifica el PID específico. Por ejemplo “SAE.RPM”, etc.

Así se procede con las demás funciones incluidas en la programación, dependiendo de la lectura que se quiera obtener, del PID que se quiera analizar y del instrumento que se aplique en ese momento, teniendo en ocho grandes grupos estas funciones:

- Función de inicialización.
- Función de creación.
- Función de estado.
- Función de corrección.
- Función de conexión.
- Funciones complementarias.
- Funciones de parámetros.
- Funciones de enlace.

Ya definidas nuestras funciones, la información enviada sigue un esquema de análisis en etapas, como se lo demuestra en este esquema:

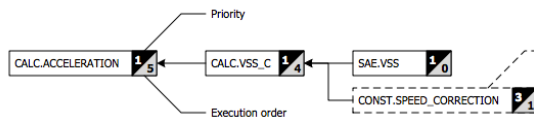


Figura 3. Diagrama de procesos OBDII

En esta guía, se sigue una serie de procesos, antes de llegar a la información requerida por la IPA, que en el caso del diagrama anterior, es el cálculo de la aceleración en tiempo real, que depende en primera instancia de la corrección en la constante de la velocidad, y de los cálculos de las variables SAE.VSS y CALC.VSS_C, cuyos parámetros provienen directamente de la OBDNet.

IV. PERSONALIZACIÓN DE LA INTERFAZ GRAFICA DE USUARIO (NUI)

Para proveer al usuario un sistema amigable de uso y de interpretación, parte del diseño final comprende la personalización de una interfaz gráfica de usuario, la misma que por lenguaje de programación de objetos complementario Visual Basic SDK, hace uso del dispositivo

táctil y del resto de complementos tecnológicos correspondientes a cada estudio.

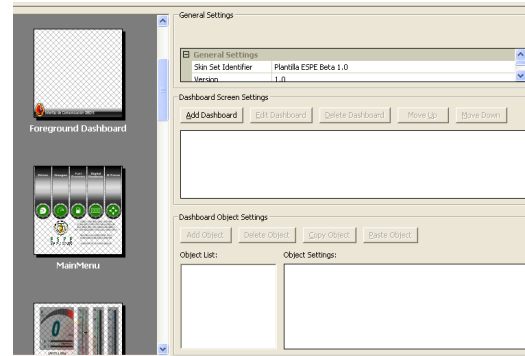


Figura 4. Programador – Compilador de objetos Visual Basic SDK

Como cualquier proyecto basado en lenguaje Visual Basic, este programador compilador tiene la peculiaridad que al mismo momento que realiza la edición de características y funcionalidad, realiza toda la secuencia informática requerida por cada acción ya en Objective-C, que a diferencia de los lenguajes C y C++, optimiza la utilización de memoria, y el uso de recurso tanto en las llamadas de variables, como en las peticiones de retorno o Callback.

El software nos divide a nuestra plantilla en varias grupos de acción, medidores, gauges digitales, GLCDs dinámicos de alta velocidad de respuesta, visualizadores de inclinómetros, lectores de acelerómetros y valores alfanuméricos correspondientes, los mismos que pueden ser interpretados tanto en el sistema inglés como en el sistema métrico de medición.

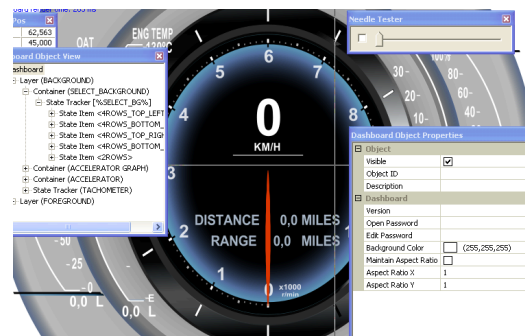


Figura 5. Editor de GLCDs dinámicos y gauges electrónicos

En lo relacionado a las características de la acción de cada objeto, se despliegan dos grupos que modifican, en primera instancia, los detalles gráficos del botón correspondiente (color, tamaño, fondo, imagen, etc), y los parámetros intuitivos en

tiempo real (cambios de tamaño, color, valores, ubicación, etc), todo automatizadamente, sin que intervenga la acción del usuario a no ser que sea estrictamente necesario.



Figura 7. Lectura de datos mediante GLCDs.

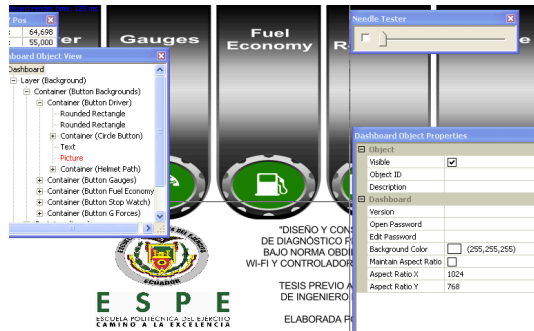


Figura 5. Edición de acción y reacción de botones y medidores.

Para finalizar en el proceso de personalización de la herramienta de diagnóstico y medición, se deben establecer los valores de recepción y envío de la información tomada desde el servidor OBDNet, de acuerdo al respectivo protocolo OBDII conforme al fabricante, en el apartado de edición de funciones y callbacks SDK como se lo señala en el siguiente gráfico:

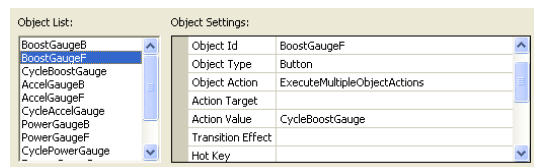


Figura 6. Funciones y callbacks SDK de la IPA.

Terminado el proceso de personalización, solo nos queda introducir nuestra interfaz gráfica en la IPA de comunicación dentro del dispositivo táctil Apple escogido (IPAD, IPHONE, etc).

V. CONEXIÓN, PRUEBAS Y RESULTADOS.

Para comprobar el excelente funcionamiento de nuestro diseño ya introducido en la IPA, vamos a conectarnos inalámbricamente con los servidores de datos OBDNet de dos diferentes vehiculos, un Mazda 3 1.6 full, y un Chevrolet Optra 1.8 estándar. Cada uno con listas de PIDs diferentes, en cuanto a componentes y a sensores, actuadores que intervienen en el sistema mecánico - electrónico.

Como primer punto de nuestro enlace físico entre el generador de señal WI-FI y nuestro dispositivo táctil con la respectiva IPA diseñada, es establecer nuestra dirección IP y pronunciar la identificación DHCP de la Tablet.

La tecnología WI-FI tiene varios parámetros (Dirección IP, Máscara de Subnet y dirección de router), los mismos que establecen una identidad virtual que ubica a cada dispositivo como receptor o generador dentro de la comunicación.

En nuestro proyecto utilizaremos los siguientes identificadores:

Dirección IP: 192.168.0.11

Máscara de Subnet: 255.255.255.0

Se utiliza un dirección IP con unidad 11 para evitar interferencias y que los envíos no se multipliquen en el ambiente, ya que por lo general, las redes de tipo infraestructura usan unidades que varían desde el 1 hasta el 10.

Conectado al servidor, nuestra IPA lee toda la lista de PIDs disponibles para el automóvil analizado, pudiendo nosotros adicionar PIDs de lectura adicionales si el caso así fuera necesario.

lateral Acceleration	0.00 Gs
Vehicle Roll	-0°
Vehicle Pitch	-0°
Max engine torque for trip P	N/A
SAE EMBS_SUP	NO DATA
Air/fuel ratio calculated from the actual lambda value from a wideband oxygen sensor	14.48 None
SAE_BAT_PWR	NO DATA
AUX_P1_X_TIMING	NO DATA
Time to empty	N/A
AUX_P1_X_SPEED	NO DATA
Ignition Timing Advance for #1 Cylinder	36.0°
Calculated Load Value	25.9 %
Intake Manifold Absolute Pressure	0.30 MPa 30 kPa
Air Flow Rate from Mass Air Flow Sensor	1.04 l/min 7.68 g/s

Figura 8. Lista de PIDs estándar de lectura para el Mazda 3 1.6 full.

Definida la lista de PIDs que deseamos que el servidor nos informe, es momento de utilizar nuestro diseño personalizado que contiene los medidores, GLCDs y gauges digitales para comprobar la tasa de envío de bits.

El servidor OBDNet del Mazda 3 1.6 full, en gauges digitales, nos dio valores correspondientes a las RPM, el TPS o sensor de posición de la mariposa de aceleración, el APP o sensor de posición del pedal de aceleración respectivo, cuyos valores son:

- RPM: 3211
- TPS: 27%
- APP: 15%

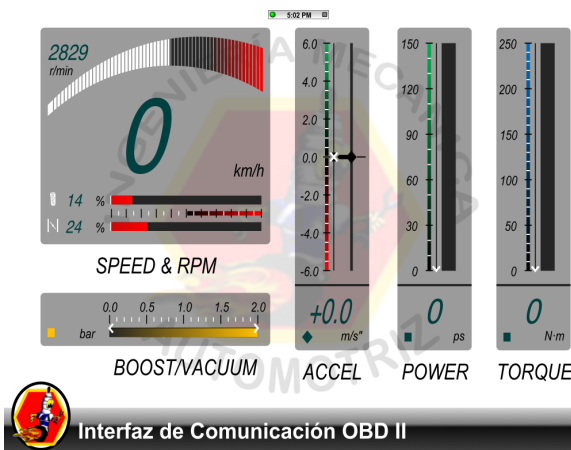


Figura 7. Lectura de gauges digitales.

Así vamos comprobando los PIDs que va recibiendo por medio de la interfaz, que en el caso del Mazda 3 nos da valores de MAP, MAF y Boost que demuestran el buen estados de estos componentes, parte fundamental de un buen diagnóstico preventivo y predictivo.



Figura 8. Lectura de flujo de aire, poder y economía de combustible.

Como device adicional al comportamiento del automotor, y utilizando tando acelerómetro como el giroscopio incorporador en el dispositivo táctil de la marca Apple, se diseño un analizador de viraje e inclinación del carro con respecto al nivel del suelo total tierra.



Figura 8. Lista de PIDs estándar de lectura para el Mazda 3 1.6 full.

Del análisis de ambos automotores, podemos sacar una tabla de composición de componentes, con lo que podemos señalar en primera instancia el nivel de tecnología del que esta compuesto cada sistema de cada automotor, la complejidad en la totalidad de PIDs disponibles por cada ECU, la compatibilidad universal con diferentes tipos y marcas, y el estado de los componentes que son nuestra meta a cumplir en el campo del diagnóstico predictivo.

	Mazda 3 1.6	Optra 1.8
Compresion	170 PSI average	160 PSI
VSS	SI 0	SI 0
RPM	SI 2829rpm	Si 2910rpm
TPS	SI 24%	SI 27%
APP	SI 14%	NA

Tabla 1. Valores medidos de los Gauges y GLCDs.

En la tabla 1 se observan los valores obtenidos por medio de los gauges personalizados previamente, comprenden a los valores de compresión promedio, que en ambos casos están en el rango normal de acuerdo al fabricante, al vss o sensor de velocidad del vehículo que como se encuentra inmóvil en el momento del

estudio nos da el valor 0, las RPMs en las que se hizo la medición, el nivel de apertura de la mariposa de aceleración TPS en esas revoluciones y el nivel de presión aplicada sobre el pedal del acelerador como en el caso único del Mazda.

	Mazda 3 1.6	Optra 1.8
COOLANT TEMP	Si 80°C	Si 75°C
LOAD	23.9%	13.7%
MAF	Si 1.1	Si 0.27
MAP	Si 5.9 inHg	Si 7.1 inHg

Tabla 2. Valores medidos mediante el data logging y sensores actuadores.

En esta última tabla se visualizan los valores de temperatura del refrigerante, la carga que soporta el motor el momento del estudio, el estado del MAF y la medición en pulgadas de mercurio inHG del MAP.



V. CONCLUSIONES.

- El estudio de la OBDNet, mediante este proyecto, ya no es de uso exclusivo de fabricantes de automóviles, scanners exclusivos, ni de programadores privados automotrices.
- La utilización de IPAs programadas en lenguaje Objective-C son óptimas para el análisis y el estudio del campo informático de sistemas OBDII.
- La personalización de plantillas con objetos activos del tipo Visual Basic, permite al usuario crear una guía propia con las mediciones que en verdad se crean necesarias en cada caso, y la forma como se presentan en la interfaz gráfica de usuario.
- El uso de un dispositivo móvil de la marca Apple, facilita en gran cantidad los procesos de identificación de ordenes de software a hardware,

ayudando en gran parte a la reducción de conflicto en el ancho de banda de la red inalámbrica WI-FI, con el consiguiente beneficio en la mejora de la tasa de envío y recepción de bits.

- El uso de nuestro diseño, es potencialmente aplicable a las áreas de diagnóstico predictivo y preventivo de automotores, telemetría, competición y de docencia aprendizaje en el área de la informática automotriz muy poco explorada actualmente.
- Nuestro diseño es compatible para la mayoría de marcas disponibles en el mercado actualmente, siempre y cuando tengan una infraestructura tecnológica informática de tipo OBDII, con un protocolo compatible de los anteriormente señalados, y un puerto que sea adaptable a nuestro generador de señal WI-FI.

VI. BIBLIOGRAFÍA.

- Altenberg, A (2008). *Become an Xcoder*. Estados Unidos: Cocalab Edit.
- Chevrolet (2000). *Manual de Servicio Optra Chevy*. Brasil.
- Hernández, L (2008). *Lenguaje Objective-C. El Principio*. Madrid: MacProgramadores.

Latacunga, Septiembre de 2012.

Ceballos Marcillo John Jairo

Director de la Carrera de Ingeniería Automotriz

Ing. Juan Castro

Director de la Unidad de Admisión y Registro

Dr. Rodrigo Vaca