

3. PROYECTO HERCULES-LEON

Este proyecto de fin de carrera forma parte de uno de índole mayor denominado Proyecto Leon, que a su vez está enmarcado en uno de mayor alcance llamado Hercules, en el que participan organismos públicos y empresas privadas. Éste, el **Hercules-Leon**, tiene como objetivo la integración de nuevas tecnologías en el sector de la automoción, de forma que aporten beneficios medioambientales y de eficiencia energética, adaptando un vehículo comercial a otro que utilice hidrógeno a presión como combustible en una pila de combustible así como un motor eléctrico. El resultado final, en resumen, es el desarrollo de un vehículo híbrido con pila de combustible de hidrógeno.

En él participan diferentes entidades que son (por orden alfabético):

- Asociación de Investigación y Cooperación Industrial de Andalucía (AICIA)
- Carburos Metálicos.
- GreenPower Tech.
- Hynergreen Technologies.
- Instituto Nacional de Técnica Aeroespacial (INTA).
- Santana Motor.

3.1 Estructuración del proyecto León

El proyecto León se divide en diferentes tareas de desarrollo para cada uno de los miembros que participan en él. Así, se pueden esquematizar, a grosso modo, como sigue (datos del 2007):

- **Santana Motor y AICIA termotecnia:** especificaciones del vehículo: definición del vehículo base a utilizar, Santana 350, y cálculo de algunos datos como consumos y autonomías esperados.
- **Hynergreen:** diseño del sistema de pila de combustible, incluyendo su propio control y comunicaciones con la ECU.
- **GreenPower y AICIA electrónica:** diseño del sistema de potencia entre baterías, pila de combustible y motor eléctrico, incluyendo el variador de éste último. Desarrollo del control de todos ellos y comunicaciones con la ECU.
- **AICIA automática:** diseño del sistema de control (ECU), del sistema de supervisión con interfaz gráfica para el usuario, del sistema de monitorización de la bancada de ensayos, y definición de los protocolos de comunicaciones
- **Carburos Metálicos:** diseño del sistema de almacenamiento de Hidrógeno.
- **INTA:** comprobación de normativas de seguridad, verificación de requisitos exigidos y dirección del proyecto.

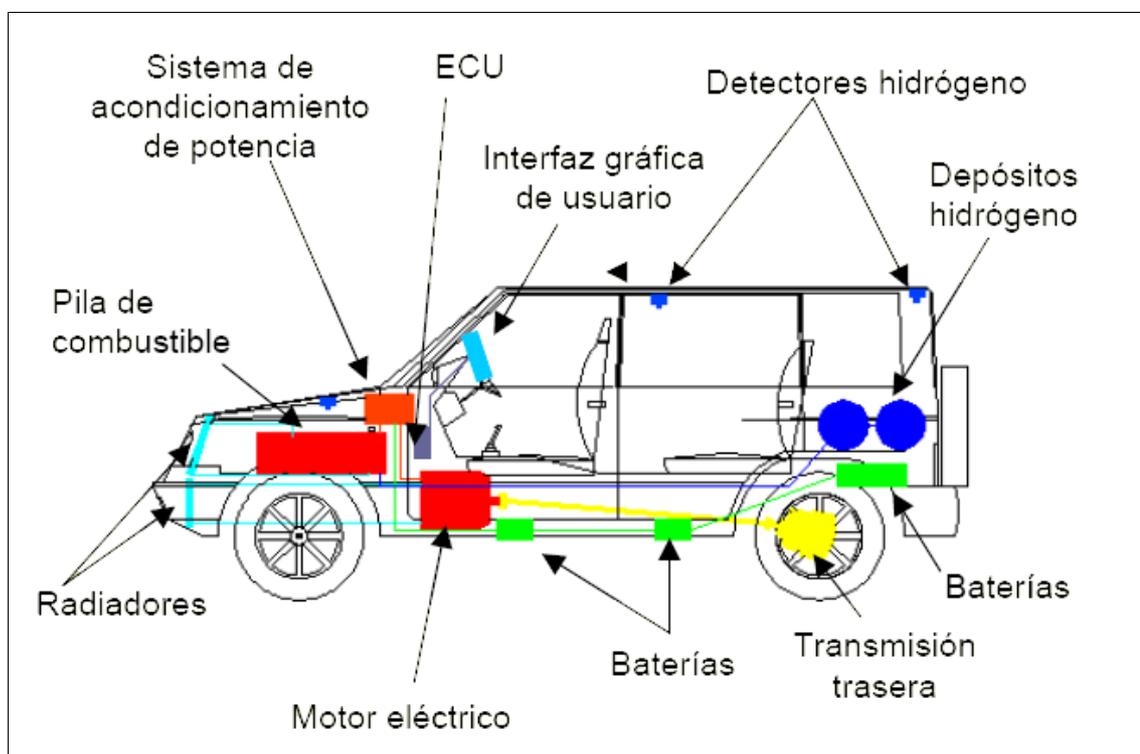


Imagen 25: posiciones de dispositivos en el León

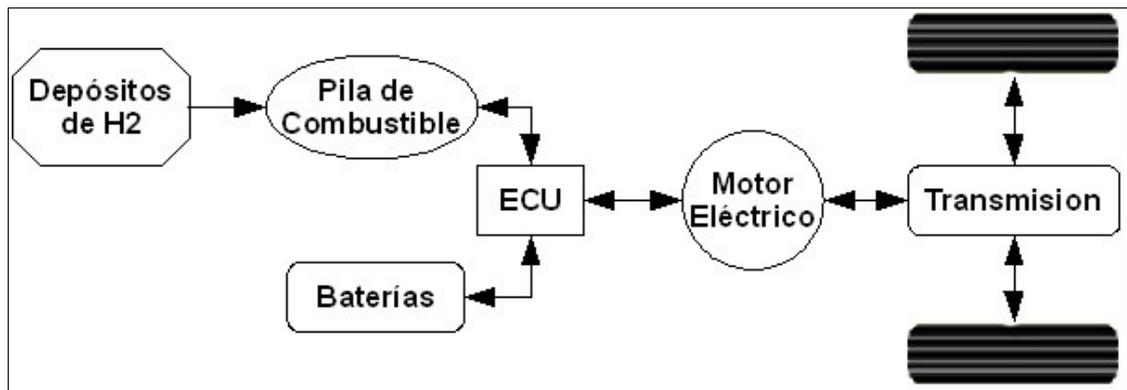


Imagen 26: esquema de dispositivos del León

Este proyecto de fin de carrera está por tanto inmerso en los objetivos del grupo AICIA automática, concretamente el diseño de la ECU y la definición de los protocolos de comunicaciones, entre otras tareas realizadas paralelamente con otros miembros del León.

3.2 AICIA Automática

El grupo de trabajo está dirigido por los catedráticos Eduardo Fernández Camacho y Carlos Bordons Alba, y por el profesor titular Miguel Ángel Ridaó Carlini. Los miembros son el ingeniero industrial David Marcos Rodríguez, el ingeniero de telecomunicación Carlos Montero, la becaria de investigación Elena González, y el becario de investigación, autor de este proyecto, Manuel Rodríguez.

Los trabajos que se están llevando a cabo dentro del Proyecto Hercules-Leon son:

- **Sistema de supervisión del vehículo:** mostrará datos al conductor para hacer de intermediario entre él y la ECU mediante una pantalla táctil, y tomará registros de las variables más importantes de todo el sistema.
- **Sistema de monitorización para la bancada de ensayo:** mostrará un interfaz para controlar las variables clave del sistema montado en la bancada de ensayo, y ayudará en la supervisión de las pruebas y las comunicaciones, permitiendo el control humano de los ensayos.
- **Unidad de Control Electrónico:** controlará autónomamente el funcionamiento de todo el sistema del vehículo (convertidores, pila de combustible, etc), actuando en respuesta a cualquier error, y gestionando la potencia del vehículo.

Los tres sistemas descritos requieren una toma de decisiones en lo que respecta a hardware, software y comunicaciones:

- **Hardware:** se utiliza PC104 para la ECU y para el Supervisor, y un ordenador Pentium para la monitorización de la bancada de ensayos.
- **Software:** se diseñan los programas sobre C/C++ sobre el sistema operativo en tiempo real QNX, para la ECU y el Supervisor, y sobre el entorno gráfico LabVIEW sobre Windows para el sistema de monitorización de la bancada de ensayos.
- **Comunicaciones:** en los tres sistemas se utiliza el protocolo CAN (Controller Area Network) para comunicarse entre ellos y con el resto de dispositivos del vehículo.

Aparte de diseñar y desarrollar estos tres entornos, este grupo de trabajo tiene el objetivo de definir una serie de estándares del sistema para que todos los dispositivos creados por el resto de grupos de trabajo del proyecto Hercules-Leon se entiendan entre sí y sigan una serie de normas que faciliten la centralización del control, y con ello, el buen funcionamiento de todo el sistema:

- **Protocolo de actuación ante errores:** un método único para todos los dispositivos, que permita un tratamiento rápido y eficiente de los problemas que surjan.

- **Definición de los parámetros de las comunicaciones:** periodicidad de los mensajes, velocidad de transferencia, etc, para conseguir una comunicación fiable.

Además se lleva cabo el diseño y el manejo de otros sistemas que dependerán de la ECU y del sistema de monitorización de la bancada de ensayos:

- **Sensorización y dispositivos de seguridad:** planteamiento de las medidas de seguridad y control de los depósitos de Hidrógeno.
- **Comportamiento del vehículo:** la forma de actuar del vehículo ante los errores, la interacción con todo el sistema, y el funcionamiento del cuadro de mandos disponible.

3.3. Estudios previos al desarrollo

Para llevar a cabo todas las tareas descritas anteriormente, es adecuado realizar algunos análisis sobre las necesidades, y en base a ello, una descripción de la opción elegida para el diseño.

De esta manera, se han llevado a cabo algunos estudios sobre las condiciones del vehículo, partiendo de las elecciones tomadas por el resto de miembros del Proyecto Hercules-Leon, y por las limitaciones del mismo. Así se han realizado cálculos de consumo y de las comunicaciones, y se ha analizado el hardware y software de trabajo junto con el cuadro de mandos del coche.

3.3.1. Datos genéricos del Leon

El proyecto León se monta sobre el chasis de un vehículo modelo Santana 350, y el sistema tiene una serie de elementos básicos que son:

- Un **motor eléctrico**, de una potencia de 60 kW.
- Un conjunto de **8 baterías de Ión Litio** modelo UEV-18XP de la marca Valence Technologies, en serie, con una potencia máxima, en conjunto, de 18 kW.
- Una **pila de combustible** de 56 kW de la marca Nuvera Fuel Cells.
- **Tres convertidores de potencia**, uno para cada uno de los dispositivos anteriores, se encargan de adaptar las tensiones e intensidades a un bus de continua a 600 voltios.
- **Tres depósitos de Hidrógeno** con un volumen total de 91 litros y una presión aproximada de 350 bares (~5000 psi).
- **Cuatro sensores** para medir la concentración de Hidrógeno en aire.
- Un **sistema supervisor** montado sobre un PC104 y con una pantalla táctil para interacción con el conductor.
- Una **unidad de control electrónico** para gestionar el funcionamiento del sistema completo.

Todo el montaje se prueba en una bancada de ensayos antes del montaje en el chasis del Santana 350. En dicha **bancada** se incluyen dispositivos que luego no permanecerán en el vehículo:

- Sensores y actuadores para el control de la pila de combustible.
- Una aplicación para monitorizar las pruebas desde un PC, en un entorno LabVIEW.

- Servomotor para simulación de ciclos de conducción.

El funcionamiento del vehículo, a grosso modo, consiste en presionar el pedal del acelerador, lo que envía una señal hacia la ECU, que gestiona la potencia requerida, y en base a unos parámetros de estado del coche, envía las consignas de potencia a los convertidores y la pila de combustible para que se genere la potencia necesaria para alimentar el motor eléctrico y mover el vehículo.

A continuación se muestra un diagrama donde se representan las comunicaciones principales y el bus de potencia:

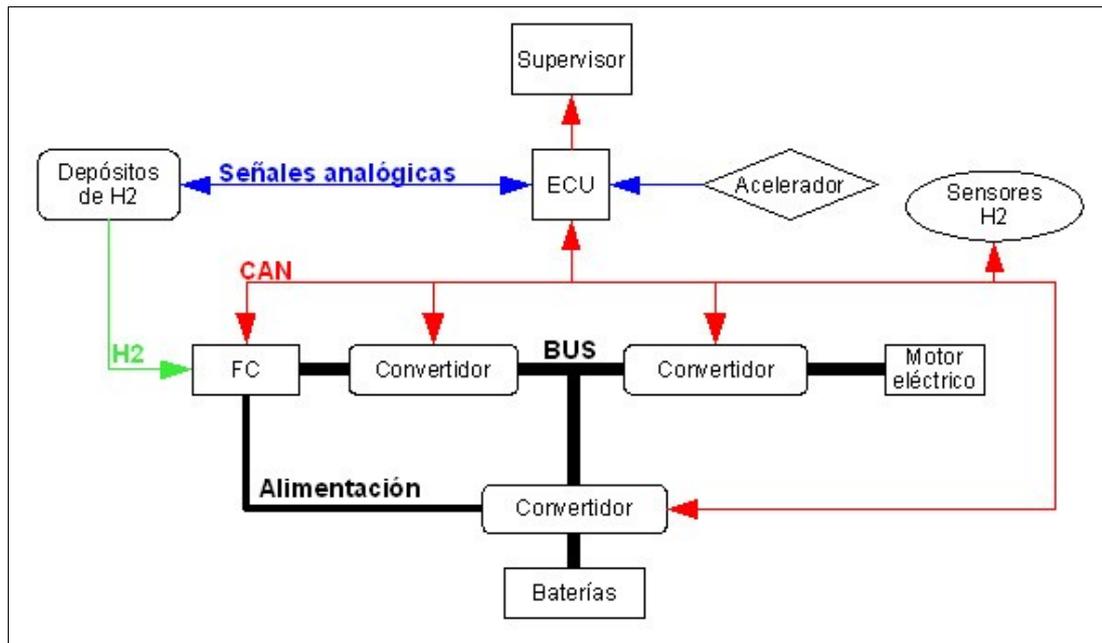


Imagen 27: diagrama de funcionamiento de dispositivos del Leon

Para finalizar se dan algunos datos sobre los diferentes dispositivos del resto de miembros del proyecto Hercules-Leon.

3.3.1.1. Santana 350

El coche que va a ser desarrollado es un Santana 350 proporcionado por la empresa Santana Motor del cual sólo se conservará el chasis y algunos complementos. La apariencia física se mantendrá intacta. Tiene las siguientes características:

Tipo de vehículo	Todoterreno
Número de puertas	4 y portón trasero
Largo/Ancho/Alto	4040/1635/1700 mm
Tara máxima	1850 kg
Velocidad máxima	131 km/h
Aceleración 0-100 km/h	35 s
Autonomía urbana	180 km
Autonomía extraurbana	130 km

Tabla 7: datos del Santana 350



Imagen 28: fotografías del Santana 350

Los datos presentados incluyen cálculos del diseño del proyecto León, no del vehículo comercial en cuestión.

3.3.1.2. Motor

El motor eléctrico tiene como característica fundamental que va a ir acoplado a las dirección trasera mediante una transmisión directa. El motor elegido es un motor eléctrico síncrono de imanes permanentes (PMSM), cuyas características son:

Par máximo	460 N·m
Potencia máxima	66 kW
Velocidad máxima	6000 r.p.m. (150 km/h)
Peso	75 kg

Tabla 8: datos del motor eléctrico

Utiliza un variador de velocidad entre él y su convertidor de potencia, y un servomotor detrás para los ensayos en bancada, que será controlado manualmente o automáticamente para probar ciclos de conducción.

3.3.1.3. Baterías

Las baterías utilizadas son del modelo UEV-18XP de la marca Valence Technologies. Se dispone de 8 de estas baterías, dispuestas en serie. Algunas de sus

características más importantes son:

Capacidad	65 Ah
Tensión nominal	19,2 V (153,6 V)
Máx. corriente en descarga de forma continua	120 A
Máxima corriente en picos de 30 s	200 A
Peso	14,9 kg (119,2 kg)
Largo/Ancho/Alto	268 / 148 / 269 mm
Volumen total	90 litros
Potencia máxima generada de forma continua	18432 W
Potencia generada en pico de 30 s	30720 W
Potencia máxima de recarga	5256 W (~ 2,5 horas para carga completa)

Tabla 9: datos de las baterías UEV 18XP



Imagen 29: fotografía de una batería XP de Valence Technologies

3.3.1.4. Pila de Combustible

Se ha optado por una pila de combustible de tipo PEMFC, de la compañía Nuvera, modelo Andromeda, debido a que funciona a temperaturas bajas entorno a los 80 grados celsius (glosario) y además tiene unas secuencias de arranque y parada muy rápidas comparadas con otras pilas existentes, lo que hace que este tipo de pilas sea muy aptas para aplicaciones automovilísticas.



Imagen 30: fotografía de la pila de combustible de Nuvera

Como vemos en la imagen, el sistema de acondicionamiento está compuesto por un convertidor CC/CC que acondiciona la potencia de la pila de combustible al bus de continua que va a alimentar al motor. Este convertidor es unidireccional puesto que la pila de combustible no puede absorber potencia. Por otra parte encontramos un convertidor de CC/CC bidireccional que une el bus de continua con las baterías, y tiene que permitir la circulación de la intensidad en los dos sentidos para que las baterías puedan cargarse y descargarse. El bus de continua tiene que ser transformado en alterna para poder alimentar al motor. Para ello se ha tenido que implementar un inversor CC/CA, que conecta el bus de continua con el motor y tiene que permitir, al igual que el convertidor CC/CC de las baterías, la circulación de la intensidad en ambos sentidos.

3.3.1.6. Depósitos de Hidrógeno

Para poder determinar el número de depósitos de hidrógeno, su capacidad y su disposición en el coche se han debido estudiar las normativas europeas acerca de ésto y además sujetarse a las especificaciones globales de diseño.

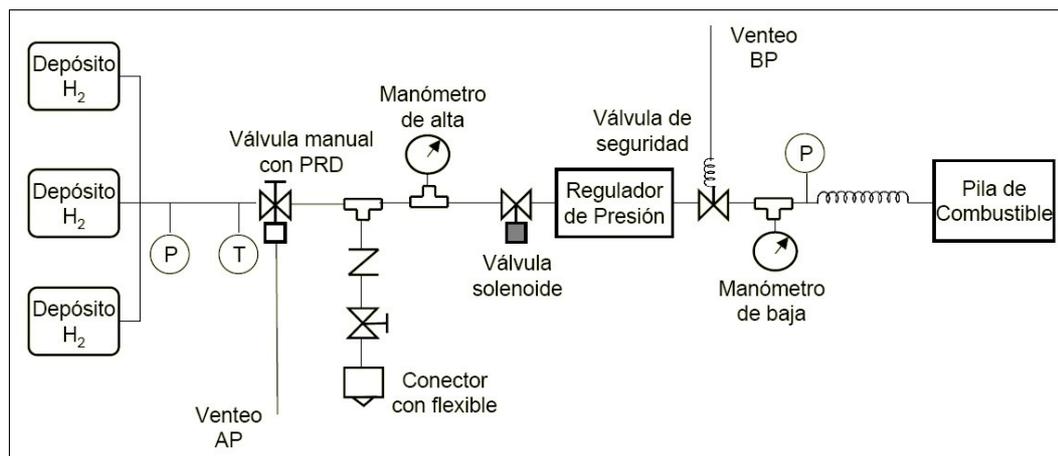


Imagen 32: esquema de los depósitos de hidrógeno

La autonomía del vehículo debe ser 100 km y además siendo la presión de almacenamiento de 350 bar, mediante una serie de simulaciones se llega a que los depósitos tienen que tener una capacidad de 100 litros y 2.4kg de Hidrógeno. Según la normativa no puede haber más de 4 depósitos por vehículo (Directiva europea), por lo que se han montado 3 depósitos: dos de 33 litros y uno de 25 litros. Se exige una certificación, y por último la posición más segura para los tanques es en la parte posterior ya que es la parte donde menos probabilidad hay para la colisión: es menos frecuentemente afectada por choque severo y la energía de deformación es menor.



Imagen 33: fotografía de los depósitos de hidrógeno

3.3.1.7. Sensores de Hidrógeno

Para la seguridad del conductor y de todo el sistema se repartirán cuatro sensores para detectar concentraciones de hidrógeno en aire, modelo HSS-440 (Hydrogen Safety Sensor) de AppliedSensor. Se van a colocar bajo el capó, en el maletero con los depósitos, y en el habitáculo.

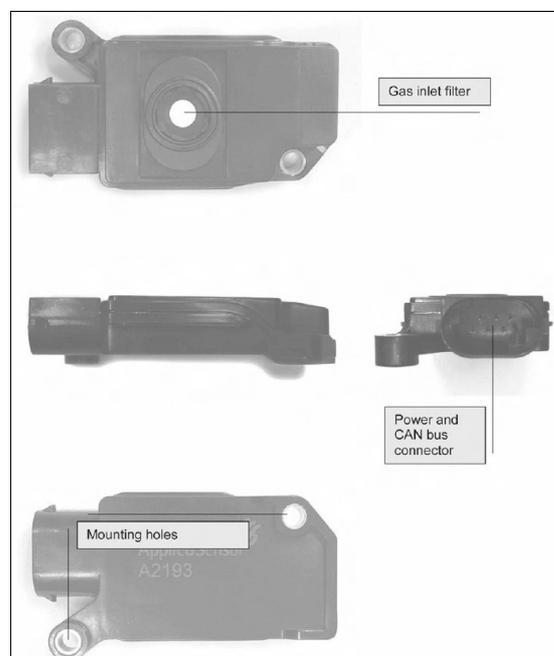


Imagen 34: sensor HS-440

Sus principales características son:

No le afecta la humedad	
Velocidad de respuesta	<2 s
Tiempo de inicio	1000 ms
Precisión	±3000 ppm
Temperatura de operación	-40° C – 110° C
Interfaz de comunicación	CAN 2.0B a 500 kbps
Dimensiones	14,40 x 42,00 x 82,20 mm

Tabla 11: datos de los sensores de hidrógeno



Imagen 35: fotografía de un sensor HSS-440

Los cuatro sensores se conectan al bus CAN de comunicaciones para transferir periódicamente su estado y la concentración de hidrógeno en aire en unidades ppm. El diagrama de pines del conector que utiliza es el siguiente:

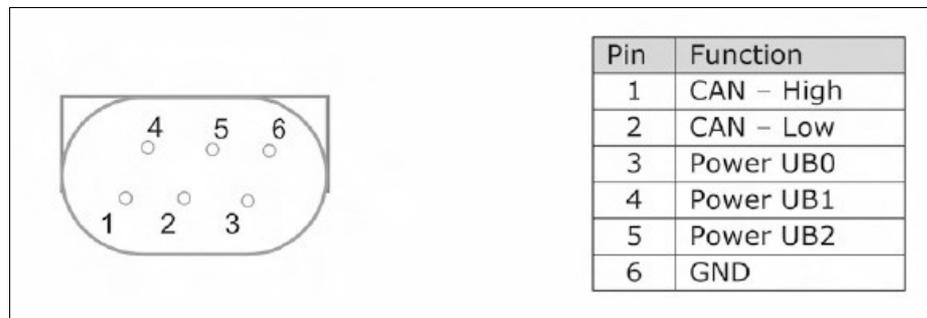


Imagen 36: esquema del conector de sensor HSS-440

Para finalizar, hacer hincapié en que necesita bus CAN con una velocidad de transferencia de 500 kbps para comunicarse, concretamente la versión extendida de este protocolo, el cual veremos más adelante. El resultado será una red de cuatro sensores por bus CAN, que se repartirán por el vehículo para detectar fugas de hidrógeno.

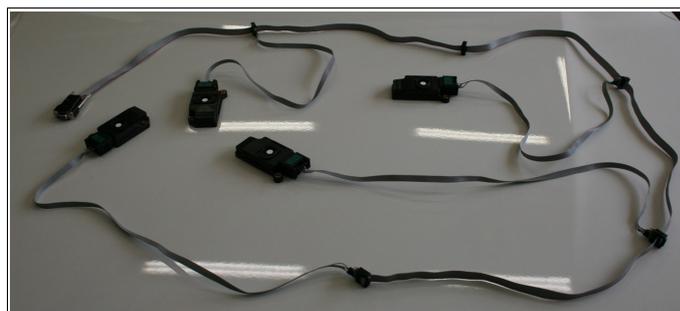


Imagen 37: red de sensores HSS-440

3.3.2. Estimaciones de potencia y consumo

Con todos los datos de los dispositivos en la mano, se pueden realizar algunos **cálculos estimativos** sobre el consumo y algunas **consideraciones** sobre la potencia de la que se dispondrá.

3.3.2.1. Potencia

Es necesario hacer una serie de consideraciones previas al desarrollo de la gestión de potencia del vehículo híbrido en cuestión.

Vamos a disponer de un motor eléctrico de hasta 66 kW (~90 C.V.) de potencia, un conjunto de baterías que podrá entregarnos hasta 18 kW en descarga de forma continua y tomará hasta 5 kW en carga, y una pila de combustible de una potencia de 56 kW pero que dispondrá de unos dispositivos que necesitarán alimentación (compresores). Además, el vehículo dispondrá de consumos auxiliares como puede ser el de los faros, elevelunas eléctricos si los tuviese, indicadores luminosos, etc.

También se contempla la utilización de la frenada regenerativa para cargar las baterías durante las frenadas, además de cargar éstas desde la pila de combustible cuando sea necesario. Ésto requerirá que se introduzca una resistencia de frenado en el bus de potencia, que es donde se realizará todo el intercambio de energía. Además esta estrategia permitirá simular el comportamiento de un motor de combustión en nuestro motor eléctrico, lo que ayudará al conductor, normalmente acostumbrado al primero, a hacerse con el vehículo fácilmente, permitiéndole una conducción fluida y sin sobresaltos.

Y para terminar con estas consideraciones, habrá que tener en cuenta las dinámicas de los diferentes dispositivos de potencia: la velocidad de respuesta de la pila de combustible no será tan rápida como la de las baterías, pero éstas normalmente no serán capaces de entregar toda la potencia necesaria. Por tanto se utilizará la pila para entregar la mayor parte de la potencia, y las baterías para asistir a la generación de energía, especialmente para encargarse de los picos debidos a acelerones repentinos o cualquier situación que exiga una rápida respuesta.

Haciendo un cálculo rápidamente ($56 \text{ kW} + 18 \text{ kW} > 66 \text{ kW}$), se obtiene que funcionando conjuntamente con la pila y las baterías se supera en casi 10 kW la potencia máxima del motor. Ésto nos permitirá no tener nunca el sistema a máximo rendimiento, lo que nos ayudará a tener mayor autonomía y a la vez poder alimentar todos los dispositivos auxiliares del vehículo sin rebajar las prestaciones energéticas del mismo. Además nos permite cierto rango en kW para los diferentes ciclos de carga y descarga de las baterías.

3.3.2.2. Consumo

Haciendo uso de los datos de los depósitos y de la pila de combustible se pueden calcular datos de consumo, de almacenamiento de hidrógeno y de autonomía, en tiempo. Aunque estas estimaciones ya han sido realizadas por otros miembros del proyecto, se ha desarrollado una aplicación sobre LabVIEW que permite introducir los datos de estado para obtener estos parámetros de consumo rápidamente.

La aplicación es muy sencilla y se basa en un **modelo cuasi-estático de la pila** de combustible que nos permite calcular el consumo de hidrógeno en gramos/segundo:

$$h(t) = m_{H_2} \cdot n \cdot \{ [i_{st}(t)] \div [2 \cdot F] \}$$

h(t): consumo de hidrógeno (g/s)

m_{H₂}: masa molar del hidrógeno

n: número de células de la pila

F: constante de Faraday

i_{st}(t): intensidad de la pila

Para el cálculo de la cantidad de hidrógeno en los depósitos se hace uso de las **Leyes de los Gases Ideales**, y para el cálculo de la autonomía bastará calcular el tiempo que tardarán en agotarse los depósitos con ese consumo estimado anteriormente.

El resultado es el siguiente:

PILA DE COMBUSTIBLE:

I(t) Perdidas

Depósito:
 -92 litros:
 2 deps de 33 litros
 1 dep de 26 litros
 -Máxima presión: 350 bares.

P H2 T H2 °C T H2 °K

RESULTADOS:

H(t) g/s
 Consumo: g/s
 Mh(t) g
 H2 en depósito: g

Resto

Segundos Totales
 Minutos Totales
 Horas Totales

Datos Stack Hércules				
Corriente(A)	Densidad Corriente(A/m2)	Tensión Stack (V)	Tensión Celda (V)	Potencia (kW)
0	0	247.105	0.985	0.0
18	500	224.786	0.878	4.0
36	1000	218.601	0.846	7.8
54	1500	214.420	0.838	11.6
72	2000	204.577	0.799	14.7
108	3000	197.725	0.772	21.4
144	4000	191.773	0.749	27.6
180	5000	184.185	0.719	33.1
216	6000	178.744	0.698	38.6
252	7000	172.238	0.673	43.4
288	8000	166.384	0.650	47.9
324	9000	159.708	0.624	51.7
360	10000	155.739	0.608	56.1

Imagen 38: aplicación LabVIEW para estimaciones de consumo

Para su uso basta **introducir** la corriente a la que está funcionando la pila de combustible, las posibles pérdidas en gramos por segundo, que incluyen cualquier evento como puede ser una fuga o un venteo, y luego la presión y la temperatura de los depósitos de hidrógeno.

El **resultado** que se obtiene es la cantidad de hidrógeno que hay en los depósitos, en gramos, el consumo de la pila en gramos por segundo, y el tiempo que tardará en agotarse el hidrógeno con ese ritmo de funcionamiento.

Para el **caso** de tener los depósitos a 350 bares de presión, una temperatura de 30° celsius, y la pila funcionando a casi su máximo, 300 amperios, que equivale aproximadamente a 50 kW, se tendría el siguiente resultado:

- **Hidrógeno almacenado:** aproximadamente 2570 gramos.
- **Consumo:** casi 0,8 gramos por segundo.
- **Autonomía:** casi una hora, exactamente 53 minutos y medio.

La aplicación se incluye en el contenido software del proyecto bajo el nombre de "**HERCULES consumo de hidrogeno.exe**". Requiere tener instalado el programa LabVIEW versión 7.1. Se ha incluido también el archivo vi (virtual instrument) para permitir modificaciones en el código.

3.3.3. Protocolo de comunicaciones: CAN

El bus de campo CAN (Controller Area Network) fue diseñado en la década de los 80 para su aplicación como **red de control** para los automóviles. Desde entonces su desarrollo ha afianzado su utilización en la industria automovilística y está actualmente normalizado por la norma ISO 11898. Entre sus características principales destaca su **fiabilidad** ante errores lo que lo hace perfecto para el tipo de entorno ruidoso de un coche.

En los vehículos existen muchos dispositivos que se comunican mediante CAN, desde el control de la potencia o el cuadro de instrumentos, hasta el desplazamiento de los asientos o el cierre centralizado, aunque se pueden usar otros protocolos como LIN (Local Interconnect Network) o MOST (Media-oriented System Transport):

- LIN: es un protocolo maestro-esclavo que se utiliza para controlar dispositivos de ON-OFF como el cierre de las puertas, espejos, sensores de lluvia, etc. Es de baja tasa de bits y puede usarse con una red CAN de alta velocidad.
- MOST: es un protocolo sobre fibra óptica y por tanto gran ancho de banda, que permite la transmisión de datos multimedia para ordenadores, televisores, o sistemas de sonido.

- La señalización en CAN se lleva a cabo por un **par de cables** (CAN_L y CAN_H) y se considera la tensión diferencial para identificar los bits de datos. Ésta es una de las razones por las que este protocolo es especialmente inmune al ruido. Suele utilizarse un conector de tipo DB9.



Imagen 40: conector DB9

- El número máximo de **nodos** en una red CAN es de 32, o 64 si no usa repetidores.
- Una red CAN se puede **configurar** a distintas velocidades de transferencia, lo que implica fijar también el tiempo de bit y la longitud máxima de los cables:

Velocidad	Tiempo de bit	Longitud máxima
1 Mbps	1 us	30 m
800 kbps	1,25 us	50 m
500 kbps	2 us	100 m
250 kbps	4 us	250 m
125 kbps	8 us	500 m
50 kbps	20 us	1000 m
20 kbps	50 us	2500 m
10 kbps	100 us	5000 m

Tabla 12: tabla de relación velocidad, tiempo de bit, longitud, en CAN

- Los mensajes pueden ser de CAN **estándar** (2.0 A) o de CAN **extendido** (2.0 B). La diferencia más destacada es que en el primero el identificador tiene 11 bits, y en el segundo tiene 29 bits. En una red, un nodo que maneje CAN extendido puede manejar CAN estándar, pero no al revés: un nodo que solo maneje CAN estándar no entenderá CAN extendido.
- Los mensajes CAN se componen de varios **campos** de información, sin embargo la mayoría son campos de control que no nos interesan para el formato de los diferentes mensajes de CAN. Los importantes son:
 - **Identificador:** un número de hasta 11 bits si es CAN estándar o 29 bits si es CAN extendido. Cuanto menor es el número, mayor prioridad.

- **Número de bytes de datos (DLC):** indica la longitud de los datos en octetos, mínimo de 1 y máximo de 8. Un mensaje que no interese para mandar datos deben tener la longitud a 1 y el dato cualquiera.
- **Datos:** de 1 a 8 octetos de información, formateado de la forma que se desee, por ejemplo, 4 campos de información de 2 octetos cada uno.

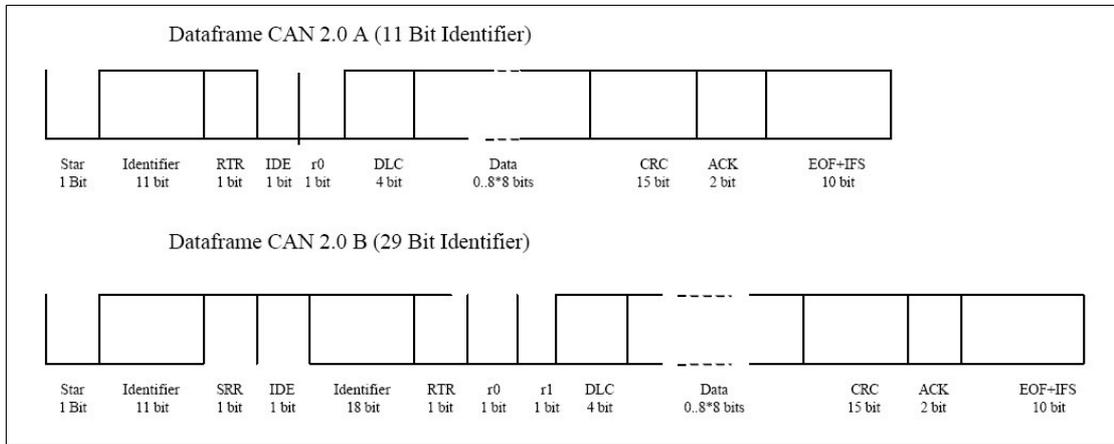


Imagen 41: campos de mensajes CAN de tipo A y B

➤ Los mensajes CAN no pueden tener más de 8 bytes de datos, lo que nos limita el **tamaño máximo** de cada tipo de mensaje o trama CAN a los siguientes valores:

	Trama 2.0A	Trama 2.0B
Tamaño máximo en bits	130 bits	154 bits

Tabla 13: tamaño de cada tipo de trama en bits

➤ Entre los nodos es necesaria una constante resincronización del tiempo de bit para evitar desajustes que incurran en errores. Para ello se usa una señalización asíncrona con **codificación unipolar NRZ** (sin retorno a cero).

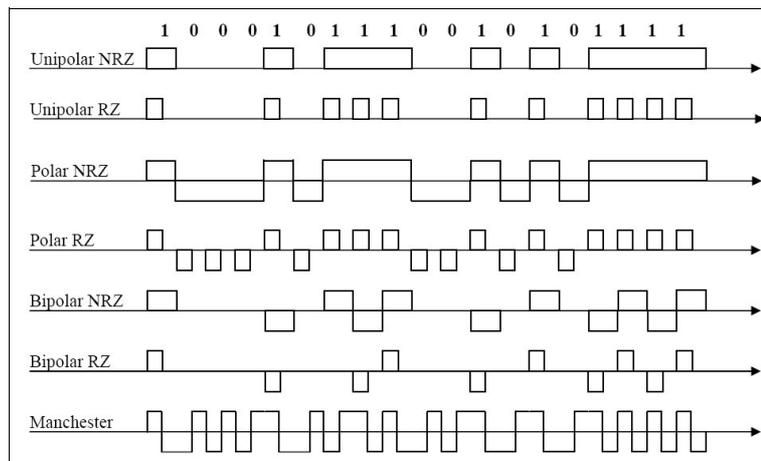


Tabla 14: comparativa de codificaciones

- CAN dispone de cinco mecanismos de **detección de errores**:
 - Error de bit: el nodo que está transmitiendo monitorea simultáneamente el bus, y comprueba constantemente si lo que transmite se está enviando correctamente.
 - Error de relleno: se considera este tipo de error cuando se reciben 6 bits consecutivos iguales, ya que rompen la regla de “bit stuffing” (glosario).
 - Error de CRC: cuando el cálculo del CRC en un receptor no coincide con el recibido en la trama.
 - Error de forma: cuando un campo de formato fijo de bits se recibe alterado en algún bit.
 - Error de reconocimiento: cuando ningún nodo cambia a dominante el bit de reconocimiento (ACK).
- Cuando se detecta un error en un nodo, éste avisa a los demás con una trama de error, y se inicia la retransmisión del mensaje.
- Si un nodo tiene demasiados errores acabaría enviando demasiadas tramas de error que condicionarían el funcionamiento del resto. En este caso el nodo es “desactivado” de la red (estado anulado) y solo se reactivaría tras 128 secuencias de 11 bits recesivos (bit recesivo equivale a un tiempo de bit sin que nadie transmita en la red).
- Hay **5 tipos de tramas** en una red CAN:
 - Trama de datos: estándar o extendido.
 - Trama remota: es una trama con el bit RTR a 1, y se usa para que un nodo solicite a otro el envío de una trama de datos concreta. No tiene campo de datos.
 - Trama de error: es la trama que genera un nodo al detectar un error. Dispone de dos campos: delimitador de error (8 bits recesivos) e indicador de error (entre 6 y 12 bits).
 - Espacio entre tramas: el espacio inter-trama consta de un mínimo de 3 bits recesivos.
 - Trama de sobrecarga: es similar a la trama de error, pero se usa cuando un nodo necesita alargar el tiempo inter-trama porque no está en condiciones de recibir ninguna. Consta de 20 bits (campos de identificador y delimitador).

3.3.3.2. Estimaciones

Para la estimación del funcionamiento de CAN buscaremos calcular el porcentaje de errores en la red según la velocidad de transferencia, tasas de errores y número de mensajes en la misma. Consideraremos dos casos, uno de **alta velocidad** y otro de **baja velocidad**, y en ambos supondremos las tramas de mensajes de tipo 2.0B (extendido) y de tamaño máximo (154 bits). Las tramas de intermisión (espacio entre tramas) las supondremos con el valor por defecto de 3 bits. No usaremos tramas remotas ni tramas de sobrecarga, y las de error tendrán el máximo de 20 bits. Todo este contexto representa un caso común con tramas de mensajes de longitud máxima.

La **probabilidad de error de bit** que consideraremos depende del ruido en el canal, que desconocemos. La probabilidad de que un error sea detectado es prácticamente del 100% gracias a los cinco métodos de detección que posee este protocolo, por lo que podemos relacionar directamente la probabilidad de error de bit con la **probabilidad de que una trama de datos tenga un error y sea detectado**.

Probabilidad de error de bit (BER)

Para calcularla usaremos la probabilidad de error aproximada por la cota de unión (glosario), para la cual necesitamos la distancia mínima (d_{min}) y la potencia de ruido en el canal (N_0).

Las dos señales que se envían y que corresponden al bit dominante (1) y al bit recesivo (0) se pueden aproximar por señales rectangulares de duración el tiempo de bit (T_{bit}) y amplitud 2 voltios para el bit dominante y 0 voltios para el bit recesivo (unipolar NRZ). Por tanto podemos deducir inmediatamente que la distancia mínima es la raíz cuadrada de la energía del bit dominante, es decir, 2 multiplicado por la raíz cuadrada del tiempo de bit:

$$d_{min} = 2 * \sqrt{T_{bit}}$$

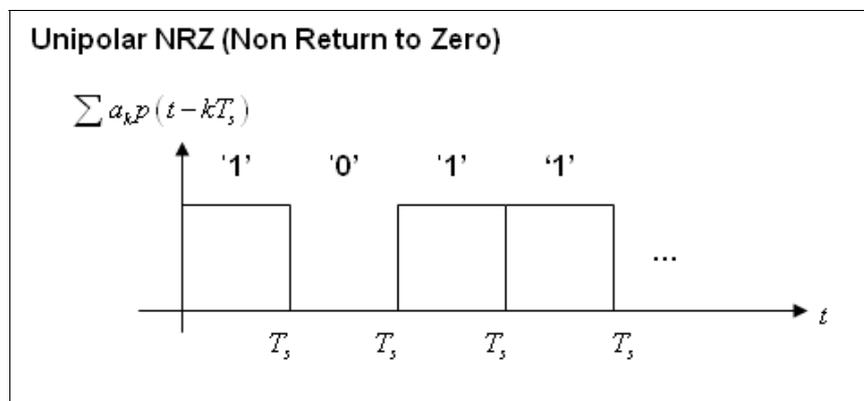


Imagen 42: codificación NRZ
($T_s = T_{bit}$)

La potencia del ruido en el canal lo estimamos según su espectro de densidad de potencia y considerándolo ruido blanco, gaussiano y de media nula:

$$S_x = N_o/2$$

La probabilidad de error según la cota de unión queda:

$$P_e = Q(d_{min}/\sqrt{2 \cdot N_o})$$

$$P_e = Q(\sqrt{(4 \cdot T_{bit})/(2 \cdot N_o)})$$

$$P_e = Q(\sqrt{(2 \cdot T_{bit})/(N_o)})$$

Vamos a realizar la estimación de la probabilidad de error según tres valores de potencia de ruido, y las dos velocidades de transmisión, de 500 kbps y de 125 kbps. El resultado se representa en la siguiente tabla:

Probabilidad de error de bit	$S_x = 2 * 10^{-6}$	$S_x = 10^{-6}$	$S_x = 5 * 10^{-7}$
500 kbps = 2 us	0,15865	0,02275	0,00003
125 kbps = 8 us	0,00003	~ 0	~ 0

Tabla 15: estimaciones de la probabilidad de error según la potencia de ruido

Estas tasas de error (BER) son demasiado altas, pero nos sirven para comparar la situación entre ambas velocidades de transferencia. Cuando a 125 kbps tenemos una tasa de casi 0, a 500 kbps podemos tener un valor de 10^{-6} que es a su vez una cantidad considerada aceptable. Recordemos la especial inmunidad del protocolo CAN al ruido, por lo que podemos considerar que la potencia del mismo no será excesiva. De ahora en adelante tomaremos la BER de 10^{-6} para los sucesivos cálculos.

La razón por la cual a una tasa de transferencia más pequeña la comunicación es más inmune al ruido es porque a menor tasa, mayor es el tiempo de bit, y por tanto mayor energía tiene, lo que influye directamente en una mayor relación señal a ruido (SNR).

Probabilidad de error en una trama:

Un error en una trama es como mínimo un bit erróneo en la misma, y ésta tiene un total de 154 bits. Como hemos considerado que todos los errores se detectan, podemos considerar que una trama que no genera trama de error es aquella que no tiene errores (parece obvio, pero si la detección no fuese perfecta, no sería así).

Considerando la BER de 10^{-6} , la probabilidad de que una trama de datos de 154 bits no tenga errores es igual a:

$$(1 - 10^{-6})^{154} = 0,999846$$

Por tanto la probabilidad de que una trama genere un error es de:

$$1 - 0,999846 = 0,000154$$

En la siguiente tabla se representan las probabilidades de que una trama genere una trama de error en base a la BER para el caso de los 500 kbps:

Probabilidad de trama de error	$BER = 10^{-5}$	$BER = 10^{-6}$	$BER = 10^{-7}$
500 kbps	0,154%	0,0154%	0,00154%

Tabla 16: probabilidad de que una trama CAN genere una trama de error

Hay que recordar que la BER en el caso de 125 kbps será siempre más baja que para el caso de 500 kbps, por lo que la probabilidad será mucho menor.

Estimación de las limitaciones:

Ya solo falta calcular el número aproximado de tramas de datos por segundo, considerando el tamaño de la trama de datos (154 bits), el tamaño de la trama de error (20 bits), la velocidad de transferencia en ambos casos, y las distintas probabilidades de errores en base a la BER. Con esto se obtendrán distintas situaciones para este contexto.

$$N_{datos} * 157 * T_{bit} + N_{error} * 180 * T_{bit} = 1 [segundo]$$

$$N_{error} = P_{error} * N_{tramas}$$

$$N_{datos} = (1 - P_{error}) * N_{tramas}$$

El primer término se refiere a la duración de todas las tramas de datos por segundo, más los 3 bits de intermission (154+3). El segundo término corresponde a la duración de las tramas de datos con errores, más la trama de error generada, y los 2 tiempos de intermission correspondientes (154+3+20+3). En función de la BER se calcula el número de tramas por segundo, y de éstas, cuántas son de datos.

En la siguiente tabla se representa la estimación final del número de tramas de datos por segundo según la BER (no se muestra el nº de tramas de error):

P_{error}	0,00154%	0,0154%	0,15%	1,5%	14%	78%
500 kbps	3185	3184	3179	3130	2638	628
125 kbps	797	796	794	782	670	157
BER	10^{-7}	10^{-6}	10^{-5}	10^{-4}	10^{-3}	10^{-2}

Tabla 17: estimación según el número de tramas de datos por segundo

3.3.3.3. Conclusiones

Podemos observar que incluso para una BER de más de 10^{-6} el protocolo CAN sigue siendo bastante resistente a los errores gracias a su casi 100% de detección de errores y a su inmunidad al ruido en el canal, y es mejor cuanto menor sea la tasa de transferencia, aunque obviamente el número de mensajes posibles por segundo será más reducido.

Para el caso de **500 kbps**, y como se usaría para tareas en tiempo real, tomando como ejemplo la situación en la que tengamos que enviar mensajes **cada 50 ms**, con una BER de 10^{-6} , podríamos manejar hasta **159 mensajes distintos**.

Para el caso de **125 kbps**, y como se usaría para tareas que requerirían menos exigencia, tomando como ejemplo la situación en la que tengamos que enviar mensajes **cada medio segundo**, con una BER de 10^{-7} , podríamos manejar hasta **398 mensajes distintos**.

Todas las conclusiones y estimaciones son aproximadas e incluyen algunas consideraciones que no serán ciertas en todas las situaciones, por lo que no se debe ajustar el tráfico de la red según estos cálculos. Solo deben ser usados como guía para evitar posibles congestiones que retrasen el funcionamiento de los dispositivos de la red CAN.

3.4. Supervisor

El sistema supervisor, también diseñado y desarrollado por AICIA Automática, se encargará de mostrar en un **interfaz gráfico e interactivo** los datos de estado del vehículo, especialmente aquellos datos referidos al consumo de la potencia y de autonomía.

Además de esta tarea de supervisión interactiva con un interfaz gráfico, el supervisor también lleva a cabo el registro de todos los parámetros del vehículo que se comunican por el bus CAN. Así, se realiza un **triple historial** en archivos de texto:

- Un historial de mensajes CAN.
- Un historial de entradas analógicas.
- Un historial de entradas digitales.

Este sistema ha sido desarrollado por uno de los miembros del grupo, David Marcos Rodríguez.

3.4.1. Diseño de la aplicación

Ha sido diseñado en lenguaje de programación C, para montarlo sobre un PC/104 con sistema operativo en tiempo real QNX. La plataforma es igual a la de la ECU, pero añade a la vez una **pantalla táctil** que será incorporada en el salpicadero del vehículo.

Los procesos que se realizan en el Supervisor son:

- **Guardar USB:** este proceso autodetecta si se ha conectado un pendrive para guardar los historiales en él.
- **Adquisición de datos:** maneja la tarjeta de adquisición de datos, es decir, todas las entradas analógicas y digitales, para adquirir los datos y enviarlos mediante colas de mensajes al proceso de Interfaz y de Historial.
- **Receptor CAN:** maneja la tarjeta de bus CAN para adquirir todos los mensajes que van llegando y enviarlos mediante colas de mensajes al proceso de Interfaz y de Historial.
- **Historial:** se encarga de guardar historiales en archivos de texto de las entradas analógicas, las entradas digitales y de los mensajes CAN recibidos.
- **Interfaz:** muestra en la pantalla táctil los datos más importantes del sistema para la supervisión por parte del conductor, y a su vez maneja la interactividad de dicha pantalla para que se pueda controlar, por ejemplo, el guardado de los historiales.

3.4.2. Interfaz de usuario

Está diseñado para una pantalla modelo G084SN03 de la marca AU Optronics Corporation con un controlador de la marca DATA MODUL, y un sensor táctil de ELO TouchSystems modelo 2701.

El interfaz indicará diversos parámetros importantes del vehículo repartidos en diferentes paneles de supervisión, donde poder visualizar estas variables de forma organizada. Así mismo dispondrá de diferentes botones para controlar algunos parámetros de funcionamiento, sobre todo durante la etapa de desarrollo.

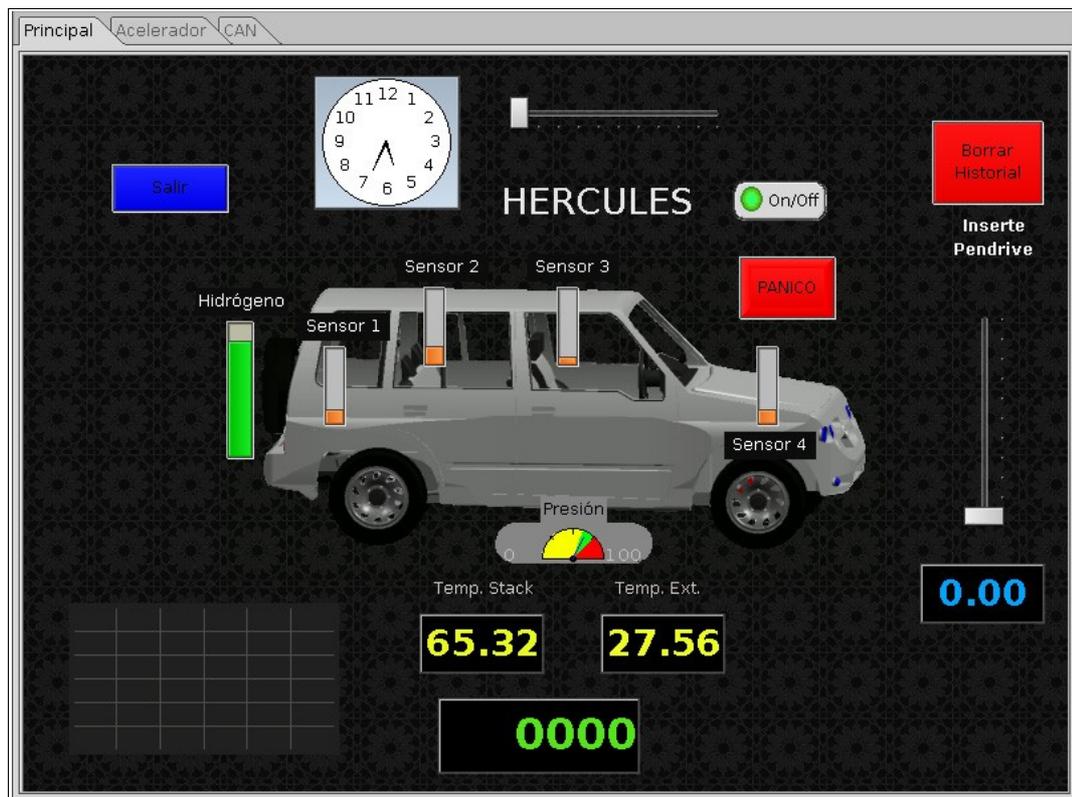


Imagen 43: panel principal del interfaz gráfico del Supervisor en su etapa de desarrollo