

# UPCommons

## Portal del coneixement obert de la UPC

<http://upcommons.upc.edu/e-prints>

---

Aquesta és una còpia de la versió *author's final draft* d'un article publicat a la revista [*Títol de la revista*].

URL d'aquest document a UPCommons E-prints:

<http://hdl.handle.net/2117/80103>

---

### **Paper publicar<sup>1</sup> / *Published paper:***

Gutierrez, E., and Sala, X. (2015) Common Rail Diesel Fuel Injection Management based on FPGA Technology. Revista IEEE América Latina, 13. 2078-2084. Doi: 10.1109/TLA.2015.7273761

---

# Common Rail Diesel Fuel Injection Management based on FPGA Technology.

<sup>1</sup> E. Gutierrez, X. Sala.

**Abstract**— This paper describes the development of a computer program using the programming software LabView™. This computer program is capable of performing real-time electronic management of a Common Rail Diesel cycle engine. This management is done openly, ie can be varied in real time execution input parameters and thus experience the engine response depending on the variation of these parameters. The computer program is focused on a real-time controller NI CRIO-9004 model available from National Instruments Co, which conditions and processes the input signals from several engine sensors and generates output signals to control the various engine actuators. The management of many engine concepts in close loop is performed with the aim of obtaining very precise control of these elements and adaptability to rapid transients.

**Keywords**— electronic fuel injection, diesel engine management, diesel common rail, software development

## I. INTRODUCCION

Las exigencias del mercado automovilístico impulsan a los fabricantes de automóviles a desarrollar motores de altas prestaciones y fiabilidad. También la legislación vigente en materia de contaminación, unida a una mayor concienciación por parte de la sociedad y a un aumento del precio de combustible, motivan la búsqueda de una gestión de inyección (entre otros parámetros) que permita que los motores de combustión tengan más prestaciones y sean más limpios y eficientes.

Con el programa desarrollado en este proyecto, el investigador puede experimentar e investigar de forma precisa cuáles son las condiciones en las que debe trabajar el motor para aumentar sus prestaciones, el rendimiento y reducir el consumo de combustible y las emisiones de contaminantes. Una vez logrados los parámetros óptimos de funcionamiento del motor en el banco de pruebas se procede a implementarlos en su unidad de control electrónico.

Este trabajo continúa una de las líneas de investigación que durante años ha seguido el Laboratorio de Motores Térmicos de la ETSEIB de la Universidad Politécnica de Cataluña.

### Abreviaturas

AI	Analogic Input (Entrada Analógica).
AO	Analogic Output (Salida Analógica).
APP	Accelerator Pedal Position (Posición del Pedal del Acelerador).
CKP	Crankshaft Position (Posición del Cigüeñal).
CMP	Camshaft Position (Posición Árbol de Levas).
DIO	Digital Input/Output (Entrada / Salida Digital).
MACI	Motor Alternativo de Combustión Interna.

ECT	Engine Coolant Temperature (Temperatura del Refrigerante del Motor).
EGR	Exhaust Gas Recirculation (Recirculación de Gases de Escape).
FPGA	Field Programmable Gate Array
IAT	Intake Air Temperature (Temperatura del Aire de Admisión).
MAF	Mass Air Flow (Flujo Másico de Aire)
MAP	Mass Air Pressure (Presión Aire Admisión).
PC	Personal Computer (Computador Personal).
PWM	Pulse Width Modulation (Modulación por Ancho de Pulso)
PMI	Punto Muerto Inferior
PMS	Punto Muerto Superior
rpm	Revoluciones Por Minuto
VI	Virtual Instrument (Instrumento Virtual)

## II. SOLUCIÓN PLANTEADA PARA GESTIONAR LA INYECCIÓN ELECTRÓNICA EN TIEMPO REAL DE UN MOTOR DIESEL COMMON RAIL

La solución planteada está diseñada para un entorno de investigación, evaluación y pruebas de MACI, en consecuencia, el sistema de gestión planteado no está orientado para su implementación de serie en los automóviles de fabricación industrial actuales, si bien esto es debido exclusivamente a motivos económicos. En este proyecto, la gestión electrónica se realiza mediante un controlador en tiempo real de programación abierta. La principal ventaja es su capacidad de cálculo y que el controlador en tiempo real empleado permite total flexibilidad, pudiendo controlar y variar todos los parámetros del motor en tiempo real de ejecución.

### A. Componentes utilizados

#### 1) Hardware Informático.

##### a) NI Compact RIO-9004

El NI CRIO 9004 es un controlador embebido en tiempo real. Funciona con un chip de tecnología FPGA que conecta directamente los diferentes módulos captadores con el procesador mediante un bus de alta velocidad y además ejecuta el software compilado directamente en lenguaje de bajo nivel. Este procesador puede ejecutar tareas que funcionan en una escala de tiempo del orden de microsegundos. Una información técnica más detallada de este controlador puede verse en [1].

El NI CRIO-9004 dispone de un chasis al que se conectan todos los módulos de adquisición o generación de señales. A continuación se muestran las características básicas de los módulos utilizados.

<sup>1</sup> E. Gutierrez, Universidad Politécnica de Catalunya, España, eegutierrez@mmt.upc.edu

X. Sala, Universidad Politécnica de Catalunya, España, sbxavi@gmail.com

(1) *Modulo AI 920: Módulo de la Serie C para entrada analógica de 8 canales a 500 kS/s.*

(2) *Modulo AO 9263: Módulo de salida analógica de actualización simultánea de 4 canales a 100 kS/s.*

(3) *Modulo DIO 9401: Módulo bidireccional de entrada digital de 8 canales a 100 ns. Cada canal es compatible con señales de 5 V/TTL.*

Las características técnicas detalladas de estos módulos pueden consultarse en [2].

#### b) *Computador personal.*

La programación del controlador se realiza mediante un PC, el cual se conecta al controlador vía Ethernet. Como dispone de conexión Ethernet puede conectarse a la red para ser programado y controlado desde cualquier punto del mundo que disponga de conexión a internet. De hecho, para la realización de este proyecto se ha programado el controlador mediante este método en numerosas ocasiones.

Las características del PC utilizado son las siguientes:

- Procesador: Pentium® Dual-Core de 2,5 GHz.
- Memoria: 3,5 GB de memoria RAM.
- Sistema operativo: Microsoft® Windows XP.

#### 2) *Tecnología FPGA.*

Los FPGA se programan con lógica de circuito. Cada elemento lógico consiste en una tabla de la verdad programable con 4-6 entradas, una salida y un flip-flop (1 bit registro) para representar directamente operaciones lógicas simples (AND, OR, etc).

A diferencia de los procesadores, los FPGAs llevan a cabo diferentes operaciones de forma paralela, por lo que estas no necesitan competir por los mismos recursos.

#### 3) *Programación y lenguaje de programación.*

El lenguaje de programación que utiliza el LabVIEW™ se llama lenguaje G, donde G simboliza lenguaje Gráfico. Para programar se utilizan iconos gráficos intuitivos, los cuales se conectan entre sí mediante cables virtuales.

### III. NATURALEZA Y COMPORTAMIENTO DE LAS SEÑALES DE ENTRADA Y SALIDA.

En [2] puede verse el esquema de conexiones (pin\_out) de los diferentes módulos utilizados, las señales utilizadas se resumen a continuación.

#### 1) *Resumen de las conexiones de entrada y salida.*

##### a) *Entradas Analógicas (modulo 9201):*

- Rpm del motor
- Presión del combustible
- Presión del aire de la admisión (MAP)
- Flujo másico de aire (MAF)
- Posición del pedal del acelerador (AAP)
- Temperatura refrigerante del motor (ECT)
- Temperatura del aire de la admisión (IAT)

##### b) *Salidas Analógicas (modulo 9263).*

- Ventilador del radiador
- Electroválvula de regulación de la presión de la sobrealimentación
- Electroválvula reguladora de la presión del combustible
- Electroválvula de la EGR

##### c) *Entradas Digitales (modulo 9401).*

- Arbol de Levas

##### d) *Salidas Digitales (modulo 9401).*

- Inyectores de los cilindros 1,2,3 y 4.

### IV. PROGRAMACIÓN FPGA DEL NI CRIO-9004 CON LABVIEW™

#### 1) *Estructura del programa.*

El programa diseñado consta de múltiples VIs.

En la Fig. 1 se puede observar la ventana del Project Explorer. En ella se puede ver el árbol jerárquico y todos los VIs que forman el programa. Las carpetas "Analog Input", "Analog Output" y "Digital Line Input and Output" corresponden a los tres módulos del NI CRIO-9004 agregados al proyecto.

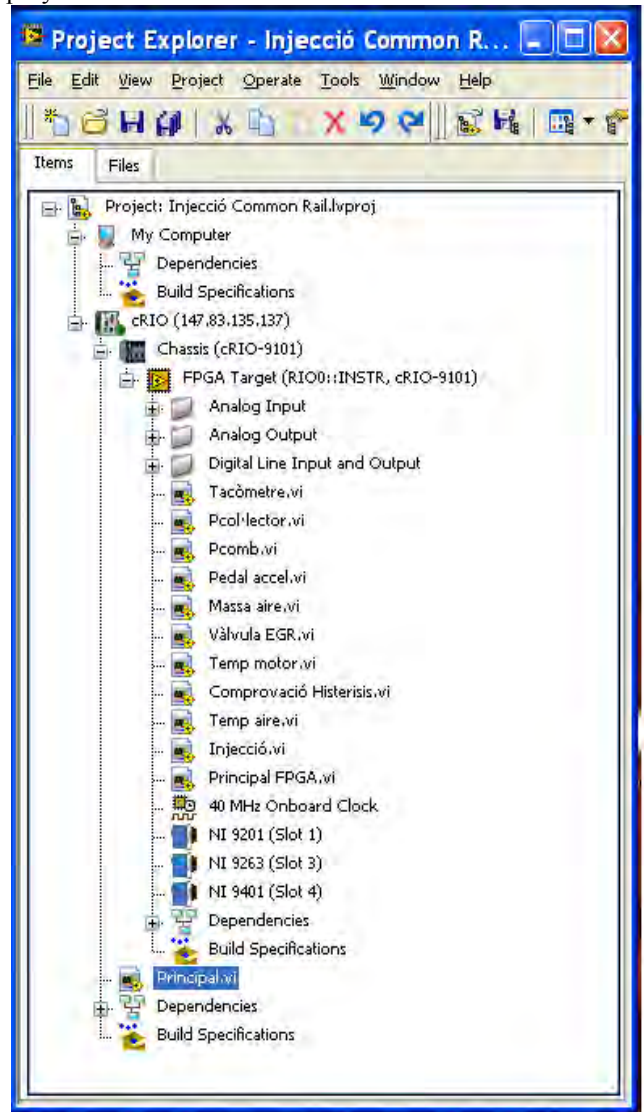


Figura 1. Ventana del Project Explorer

El VI "Principal.vi" recibe y envía información a todos los demás VIs. Este VI llama al VI "Principal FPGA.vi", con la que comparte información. Al mismo tiempo esta última llama a todos los demás VIs, los cuales son ejecutados en paralelo.

#### 2) *Gestión de la presión en el colector de admisión.*

Es necesario controlar la presión en el colector de admisión en función de las condiciones de carga y rpm del motor. En la Fig. 2 se muestra y seguido se describen los componentes del

diagrama de bloques del VI "Pcolector.vi", cuya función es leer la señal procedente del sensor de presión del colector de admisión y transformarlo en una presión.

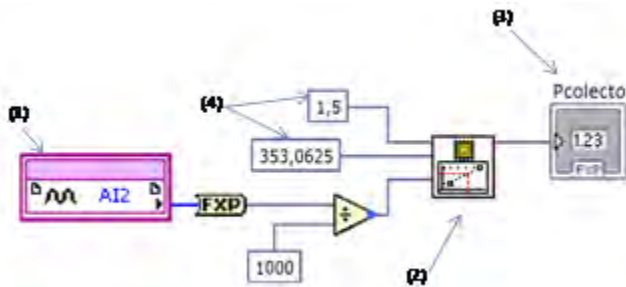


Figura 2. Diagrama de bloques del VI "Pcolector.vi"

- (1) Nodo encargado de realizar la adquisición / lectura de la señal procedente de la entrada analógica AI2, conectada al sensor de presión del colector de admisión.
- (2) Función interpoladora. Lee el valor procedente de (1) y devuelve la presión en kPa.
- (3) Indicador encargado de captar los valores de salida de la función interpoladora. Estos valores son transferidos al VI principal donde serán procesados.
- (4) Constantes de interpolación. Contienen los valores utilizados para construir la recta de interpolación y que se ajustan en este caso particular.

### 3) Gestión de la presión del combustible en el conducto común (Common Rail).

Es necesario controlar la presión en el conducto común en función de las condiciones de carga y rpm del motor. Para ello se lee la presión real a través del sensor de presión del combustible, situado en el conducto común, y se compara con la presión objetivo, que es la presión que se desea tener en el conducto común para las diferentes condiciones de carga y rpm del motor.

En la Fig. 3 se muestra el diagrama de bloques del VI "Pcomb.vi", cuya función es leer la señal procedente del sensor de presión de combustible y transformarla en una presión.

- (1) Nodo encargado de realizar la adquisición / lectura de la señal procedente de la entrada analógica AI1, conectada al sensor de presión de combustible.
- (2) Función interpoladora. Lee el valor procedente de (1) y devuelve la presión en bares.
- (3) Indicador encargado de captar los valores de salida de la función interpoladora. Estos valores son transferidos al VI principal donde serán procesados.

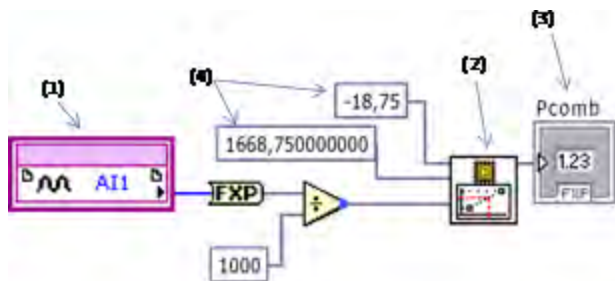


Figura 3. Diagrama de bloques del VI "Pcomb.vi"

- (4) Constantes de interpolación. Son las constantes que contienen los valores utilizados para construir la recta

de interpolación y que se ajustan en este caso particular.

### 4) Gestión del ventilador para la refrigeración del motor

Se ha diseñado un lazo cerrado para controlar el ventilador del radiador para asegurar que el motor siempre se encuentra a la temperatura óptima de trabajo. Para ello se lee el valor de la temperatura del líquido refrigerante a través del sensor de temperatura. Si la temperatura del refrigerante supera el valor máximo establecido por el investigador, el ventilador se activa para reducir la temperatura del refrigerante.

La lectura de la señal de temperatura es enviada a la "Principal FPGA.vi", para finalmente ser transferidos al VI "Principal.vi" donde serán procesados.

### 5) Gestión de los pulsos de inyección.

El programa es capaz de generar hasta tres pulsos de inyección para cada cilindro, los dos primeros corresponden a las dos pre-inyecciones y el último a la inyección principal. El primer tren de pulsos de inyección es enviado al cilindro número 1 y su inicio se realiza unos grados de cigüeñal antes del PMS. Para conocer el PMS del resto de cilindros se suman 180 grados de giro del cigüeñal a cada uno de ellos.

Los retrasos y las duraciones de los pulsos de inyección dependerán de las rpm del motor y de la posición del pedal del acelerador. En la Fig. 4 se muestra la parte del diagrama de bloques del VI "principal.vi" correspondiente a la inyección.

De este diagrama se detallan los elementos más relevantes y se explica su funcionamiento.

- (1) Matrices de dos dimensiones que contienen la duración de los pulsos de inyección.
- (2) Matrices de dos dimensiones que contienen los retrasos de los pulsos de inyección.
- (3) Vectores que representan los índices de rpm y carga del motor utilizados para la construcción de las matrices de dos dimensiones (1) y (2).
- (4) Variable local que contiene el valor instantáneo de las rpm del motor.
- (5) Función de lectura y escritura. En este caso se lee el valor de la carga del motor, procedente de la VI "Principal FPGA.vi".
- (6) Corte de inyección. Control que permite al investigador desde el Panel Frontal del VI "Principal.vi" establecer el valor máximo de rpm a las que se realizará la inyección.
- (7) Parada motor. Pulsador visible desde el Panel Frontal del VI "Principal.vi".
- (8) Case structure. Realiza diferentes acciones en función de cuál sea el valor de la señal de selección.
- (9) Terminal de selección. Selecciona cuál de los casos del Case structure se ejecuta.
- (10) Función de interpolación para matrices de dos dimensiones.
- (11) Función encargada de leer el elemento que ocupa la fila  $i$ , columna  $j$ , de una matriz de dos dimensiones. En este caso se lee el elemento que ocupa la primera posición de la matriz, es decir el elemento con índice  $i = 0$ ,  $j = 0$ . Este elemento es el resultado de la interpolación que ha realizado la función (10).
- (12) Formula Nodo. Por el lado izquierdo recibe las variables de entrada (rpm y el retraso en la inyección expresado en ángulo de cigüeñal), mientras que por el



derecho devuelve las variables deseadas (retraso temporal en la inyección).

expresados en ángulo de cigüeñal, que se están aplicando en cada uno de los pulsos de inyección.

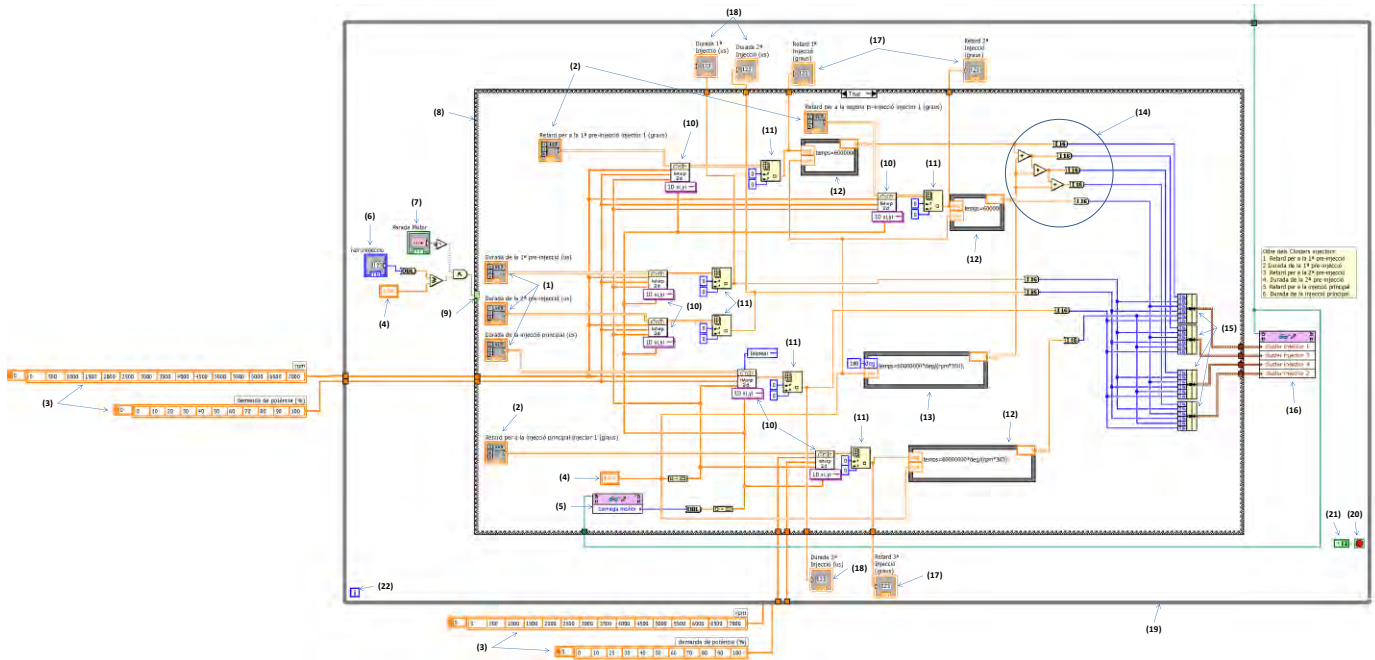


Figura 4. Diagrama de bloques del VI "principal.vi" correspondiente a la inyección

El tiempo que tarda el cigüeñal a recorrer un cierto ángulo dependerá de las rpm a las que gire el motor, por este motivo se ha introducido en el Formula Node la ecuación (1), la cual permite transformar el retraso expresado en ángulo de cigüeñal, en retraso temporal expresado en microsegundos.

$$tiempo = \frac{60 \cdot 10^6 \cdot deg}{rpm \cdot 360} \quad (1)$$

Donde: deg es el retraso expresado en ángulo de cigüeñal, tiempo es el retraso temporal expresado en microsegundos y rpm las revoluciones por minuto del motor.

- (13) Formula Node. Este elemento es idéntico al (12), con la única diferencia de que la variable de entrada correspondiente al retraso expresado en ángulo de cigüeñal ahora es una constante de valor 180 grados, que corresponde al desfase de los pulsos de inyección entre cada uno de los cilindros.
- (14) Realiza la suma al primer tren de pulsos de cada uno de los cilindros sucesores al primero.
- (15) Bundle Function. Función que agrupa los elementos que le entran por la izquierda, formando de esta manera un grupo.
- (16) Función de lectura y escritura. En este caso se escribe el valor procedente de las Bundle Functions (15) sobre las variables del VI "Principal FPGA.vi" llamada "cluster inyector 1", "cluster inyector 2", "cluster inyector 3" y "cluster inyector 4". Cada grupo (cluster) lleva la información de los retrasos y duraciones de los pulsos de inyección para cada uno de los cuatro inyectores.
- (17) Indicadores encargados de mostrar, en el Panel Frontal del VI "Principal.vi", el valor de los retrasos,

- (18) Indicadores encargados de mostrar, en el Panel Frontal del VI "Principal.vi", la duración de cada uno de los pulsos de inyección.
- (19) "While loop", que representa un bucle.
- (20) Terminal de condición del bucle (19).
- (21) Constante booleana de valor "falso".
- (22) Terminal de iteración. Da información de cuál es la iteración del bucle en cada instante.

#### 6) Gestión de la señal del sensor IAT.

El elemento que genera la señal de la temperatura del aire de admisión es el sensor IAT. Esta señal es utilizada para corregir las duraciones de los pulsos de inyección.

La lectura de la señal de temperatura es enviada a la "Principal FPGA.vi", para finalmente ser transferidos al VI "Principal.vi" donde serán procesados.

#### 7) Gestión de la señal del sensor MAF

La señal generada por el sensor MAF proporciona información del caudal másico de aire que circula por el colector de admisión. Esta señal es utilizada para corregir las duraciones de los pulsos de inyección.

La lectura de la señal del sensor MAF es enviada a la "Principal FPGA.vi", para finalmente ser transferido al VI "Principal.vi" donde será procesado.

#### 8) Gestión de la recirculación de los gases de escape (EGR).

Para la recirculación de gases de escape se ha establecido un rango de actuación. La recirculación queda anulada cuando el motor supera el régimen de giro de 3000 rpm y también se anula para temperaturas del refrigerante motor inferiores a 80 °C. Para las demás condiciones de trabajo del motor, la cantidad de gases de escape recirculados se realiza en función de las condiciones de carga y rpm del motor.

Este VI hace la llamada del VI "Válvula EGR.vi", el cual lee el valor del ciclo de trabajo. En la Fig. 5 se muestra el diagrama

de bloques de este VI, y a continuación se explica cómo se genera la señal PWM a partir del valor del ciclo de trabajo.

- (1) Controlador que contiene el valor del ciclo de trabajo de la función generadora de ondas cuadradas. Este ciclo de trabajo es el valor en tanto por uno que se había generado en el VI "Principal.vi".

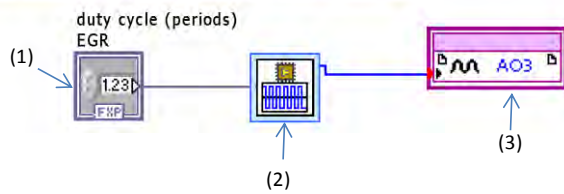


Figura 5. VI "Válvula EGR.vi".

- (2) Función generadora de ondas cuadradas. Esta función generará una onda cuadrada de frecuencia 300 Hz y comprendida entre 0 y 5 V, el ciclo de trabajo de esta onda es variable (PWM) y controlado mediante el controlador (1).
- (3) Nodo encargado de enviar la señal procedente de (2) al canal de salida AO3 del módulo de salidas analógicas AO 9263. En esta salida se conecta la electroválvula EGR.

#### 9) Gestión de la señal del sensor APP.

El sensor APP proporciona la información de cuál es la posición del pedal del acelerador. Esta señal es utilizada para determinar el estado de carga del motor.

En la Fig. 6 se muestra el diagrama de bloques del VI "Pedal accel.vi", cuya función es leer la señal procedente del sensor APP.

- (1) Nodo encargado de realizar la adquisición / lectura de la señal procedente de la entrada analógica AI4, conectada al sensor APP.
- (2) Función interpoladora. Lee el valor procedente de (1) y devuelve la posición del pedal del acelerador desde 0 a 100% de recorrido del pedal.
- (3) Indicador encargado de captar los valores de salida de la función interpoladora. Estos valores son leídos por el VI "principal.vi".

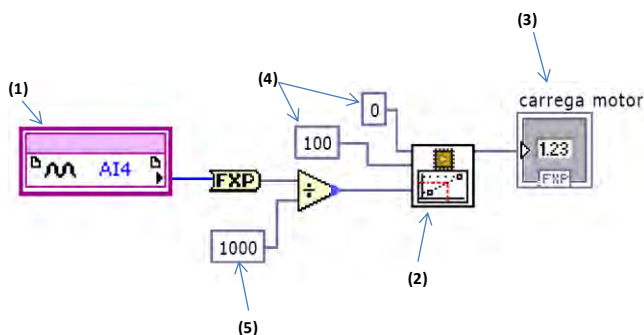


Figura 6. VI "Pedal accel.vi".

- (4) Constantes de interpolación. La constante 0 es el valor, en tanto por ciento, que devuelve la función por una entrada de cero voltios, mientras que la constante 100 es el valor, en tanto por ciento, que devuelve la función por una entrada de 5 voltios.

- (5) Constante. Divide por 1000 la señal de entrada a la función, y la convierte en un valor comprendido entre 0 y 1, ya que la función interpoladora sólo acepta valores comprendidos entre 0 y 1.

#### 10) Gestión de la señal del sensor CKP

La señal del sensor CKP se utiliza para determinar el régimen de giro del motor. Esta señal es muy importante, y en su ausencia el programa interpreta que el motor está detenido, y por tanto no se inyecta combustible.

El programa dedicado a adquirir y transformar la señal del sensor CKP en revoluciones por minuto es muy complejo. Se han utilizado cuatro VIs para dividir la complejidad del problema inicial en otras no tan complejas.

La primera dificultad es que al tratarse de una señal inductiva el valor de la amplitud varía en función de la velocidad de la rueda fónica. La segunda dificultad añadida ha sido el ruido que incorpora la señal inductiva.

En la Fig. 7 se muestra el diagrama de bloques del VI "Principal FPGA.vi" correspondiente al tratamiento de las rpm, cuya función es leer y determinar el período de la señal del sensor CKP. Se han numerado los elementos que lo componen y se explica su funcionamiento.

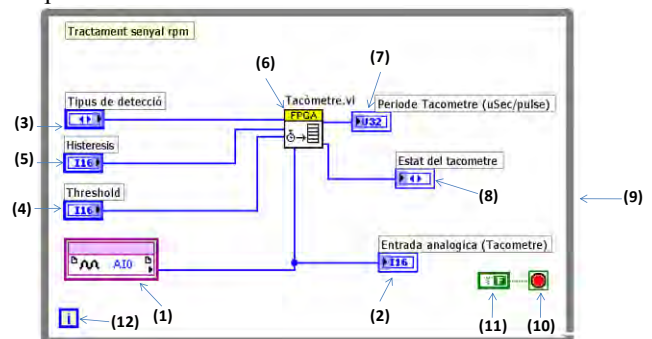


Figura 7. VI "Principal FPGA.vi" para el tratamiento de las rpm.

- (1) Nodo encargado de realizar la adquisición / lectura de la señal procedente de la entrada analógica AI0, conectada al sensor CKP.
- (2) Entrada analógica (Tacómetro). Indicador informativo, en el que se puede ver en tiempo real la entrada leída por el nodo (1).
- (3) Tipos de detección. Control mediante el cual se selecciona la forma en que el VI "Tacómetro.vi" detecta el período de la señal de entrada, su valor puede ser "Subida" o "Bajada" y es transferido al control (3) del VI "Tacómetro.vi". En este caso se ha configurado como "Subida".
- (4) Threshold. Control numérico mediante el cual se fija el nivel al que está centrado la señal inductiva del sensor CKP. Su valor es transferido al control (4) del VI "Tacómetro.vi". En este caso se ha fijado a cero voltios.
- (5) Histéresis. Control numérico mediante el cual se fija la histéresis de la señal inductiva del sensor CKP. Su valor es transferido al control (5) del VI "Tacómetro.vi". En este caso se ha fijado a 0,5 V.
- (6) VI "Tacómetro.vi". Se hace la llamada de este VI en cada iteración, escribiendo y leyendo información mediante las variables conectadas a ella.

- (7) Período Tacómetro (uSec / pulse). Indicador que contiene el valor instantáneo del período de la señal CKP, expresado en microsegundos.
- (8) Estado del Tacómetro. Indicador que informa del estado en el que se encuentra el VI "Tacometre.vi". Su valor es leído del indicador (3) del VI "Tacometre.vi".
- (9) "While loop". Bucle.
- (10) Terminal de condición. Cuando toma por valor "verdadero" el bucle terminará la ejecución en curso.
- (11) Constante booleana de valor "falso".
- (12) Terminal de iteración. Da información de cuál es la iteración en la que se encuentra el bucle a cada instante comenzando por cero, por lo tanto en la enésima ejecución del bucle, el terminal de iteración tomará por valor n-1.

En la Fig. 8 se muestra el diagrama de bloques del VI "Principal.vi" correspondiente a la obtención de las rpm, cuya función es determinar las rpm del motor a partir de la señal generada por el VI "Principal FPGA.vi". Se numeran los elementos que lo componen y se explica su funcionamiento.

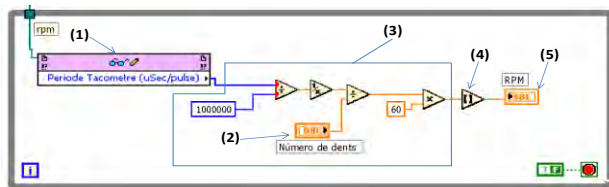


Figura 8. VI "Principal FPGA.vi" para la obtención de las rpm.

- (1) Función de lectura y escritura. En este caso se lee el valor del indicador Período Tacómetro (uSec / pulse) del VI "Principal FPGA.vi", el cual contiene el valor del período de la señal CKP.

- (2) Número de dientes. Se puede modificar su valor desde el Panel Frontal del VI "Principal.vi". Su valor representa el número de dientes que tiene la rueda fónica solidaria al eje del cigüeñal y su valor es utilizado para realizar el cálculo de las rpm.
- (3) La función de este grupo de elementos es convertir el período (microsegundos) en una velocidad angular (revoluciones por minuto). La operación matemática que realizan es la mostrada en la ecuación (2).

$$rpm = \frac{10^6 \cdot 60}{T \cdot (\text{Número de dientes})} \quad (2)$$

Donde: rpm son las revoluciones por minuto, T es el período expresado en microsegundos, y número de dientes es el número de dientes que contiene la rueda fónica solidaria al cigüeñal.

- (4) Redondeo a la unidad. Elemento que devuelve el valor de entrada redondeado a la unidad.
- (5) RPM. Indicador que contiene el valor instantáneo de las rpm del motor.

### 11) Interfaz interactiva del programa.

En este apartado se muestra la interfaz que permite la comunicación entre el programa desarrollado en este proyecto y el investigador. Esta interfaz es el Panel Frontal del VI "Principal.vi", la cual se muestra en la Fig. 9.

- (1) Temperatura refrigerante. En esta sección se pueden establecer los valores para los que el ventilador se activará y se desactivará. También se visualiza el valor de la temperatura del refrigerante motor. Se activa una señal luminosa cuando el ventilador está en funcionamiento.
- (2) Presión colector de admisión. Muestra la presión del aire en el colector de admisión. Debajo del indicador de aguja se visualiza numéricamente la presión objetivo del aire en el colector de admisión.

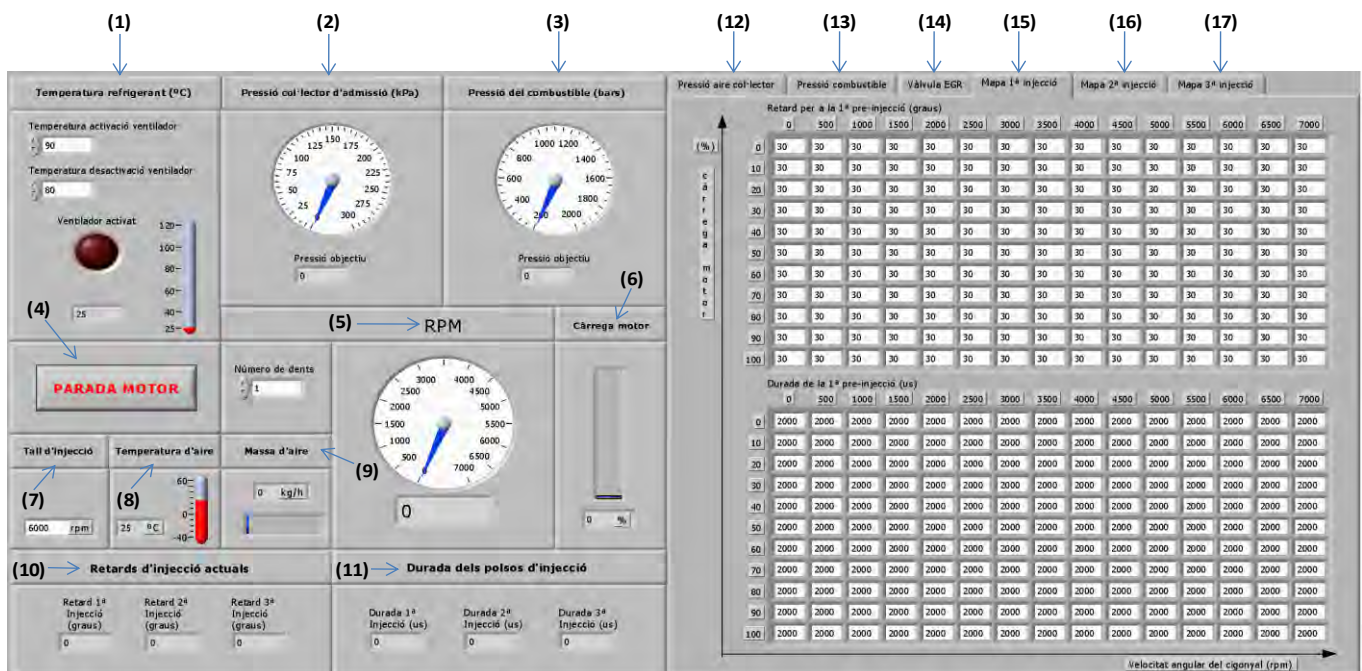


Figura 9. Interfaz del VI "Principal.vi"



- (3) Presión del combustible. Muestra la presión del combustible en el conducto común. Debajo del indicador de aguja se visualiza la presión objetivo del combustible en el conducto común.
- (4) Parada motor. Pulsando este botón se corta la inyección de combustible para detener el motor.
- (5) RPM. Muestra las revoluciones por minuto a las que gira el cigüeñal del motor (rpm).
- (6) Carga motor. Muestra la posición del pedal del acelerador.
- (7) Corte de inyección. Establece el valor de rpm a partir del cual, por seguridad, se detendrá la inyección de combustible.
- (8) Temperatura de aire. Muestra la temperatura del aire en el colector de admisión.
- (9) Masa de aire. Muestra la masa de aire en el colector de admisión en kilogramos por hora.
- (10) Retrasos de inyección actuales. Muestran el valor, en ángulo de cigüeñal, de los respectivos retrasos en la inyección que el programa está utilizando.
- (11) Duración de los pulsos de inyección. Muestran la duración de cada uno de los tres pulsos de inyección que el programa está ejecutando, expresados en microsegundos.
- (12) Presión del aire en el colector. Tabla de las presiones objetivos del aire en el colector de admisión.
- (13) Presión del combustible. Tabla de las presiones objetivos del combustible en el conducto común.
- (14) Válvula EGR. Tabla de los ciclos de trabajo enviados a la electroválvula EGR.
- (15) Mapa 1<sup>a</sup> inyección. La tabla superior contiene los retrasos de inyección para la primera de las tres inyecciones. La tabla inferior contiene las duraciones del primer pulso de inyección.
- (16) Mapa 2<sup>a</sup> inyección. La tabla superior contiene los retrasos de inyección para la segunda de las tres inyecciones. La tabla inferior contiene las duraciones del segundo pulso de inyección.
- (17) Mapa 3<sup>a</sup> inyección. La tabla superior contiene los retrasos de inyección para la tercera de las tres inyecciones. La tabla inferior contiene las duraciones del tercer pulso de inyección.

Todos los valores almacenados en las tablas explicadas anteriormente se pueden modificar en tiempo real durante la ejecución del programa.

## V. CONCLUSIONES.

Se ha desarrollado un sistema de inyección electrónica abierto y flexible capaz de evaluar y poner a punto la inyección de combustible de un motor Diesel Common Rail. También es muy útil cuando el objetivo es evaluar el comportamiento de un motor después de alguna modificación realizada.

Se describe la programación informática del sistema de forma que pueda ser utilizada por cualquier investigador de esta temática o, al menos, utilizarla como punto de partida para el desarrollo de sistemas similares pero específicos de acuerdo con sus propio entorno investigativo, teniendo siempre presente que optimizando la gestión electrónica de un MACI se logra mejorar su comportamiento, sus prestaciones, su rendimiento y reducir el consumo de combustible y emisiones de contaminantes.

El sistema desarrollado es completamente abierto, lo que lo hace fácilmente configurable, permitiendo su uso en cualquier

tipo de MACI que disponga de inyectores de combustible accionados electrónicamente.

El coste económico del proyecto no es excesivo, sobre todo si es a nivel institucional, por lo que pensamos que puede ser implementado por cualquier equipo de trabajo.

Por último, se puede afirmar que la evaluación de este sistema a nivel de laboratorio ha sido completamente satisfactorio. Se ha evaluado en un motor virtual, esto es, mediante equipos generadores de señales, osciloscopios y sistemas de adquisición de datos se ha comprobado su correcto funcionamiento.

## AGRADECIMIENTOS

Queremos agradecer al departamento de Máquinas y Motores Térmicos de la ETSEIB los recursos que ha dejado a nuestra disposición para la realización de este proyecto de investigación.

## REFERENCIAS

- [1] <http://sine.ni.com/ds/app/doc/p/id/ds-200/lang/es>
- [2] <http://sine.ni.com/nips/cds/view/p/lang/es/nid/14147>
- [3] AUDI AG. *Sistema de inyección Common Rail para motor V8 TDI de 3,3 ltr. Diseño y funcionamiento*. Programa didáctico 227. D-85045 Ingolstadt, Depto. IVK-5.
- [4] Heywood, J.B., *Internal Combustion Engine Fundamentals*. McGraw-Hill Book Co., New York, N.Y., pp.20-23, 1988
- [5] Experimental investigation on performance and emission characteristics of dual fuel split injection of ethanol and diesel in CI engine, Babu, P.T., Srivathsan, P.R. ; Banugopan, V.N. ; Prabhakar, S. ; Annamalai, K. *Frontiers in Automobile and Mechanical Engineering (FAME)*, 2010, IEEE Conference Publications.
- [6] Bosch, "automotive electrics automotive electronics" Bentley publishers, U.S.A, 2004
- [7] Fuel-injection control system design and experiments of a diesel engine. Yuanchun Li; Guangjun Liu; Xiao Zhou. *Control Systems Technology, IEEE Transactions on* Volume: 11, Issue: 4, 2003, Page(s): 565 – 570.
- [8] M. Odaka, Y. Tsukamoto, H. Yagi, and T. Hashizume, "Development of automatic engine control system with predictive control and its application for heavy duty diesel engine," SAE SP, no. 819, 1990
- [9] Common rail injection system on-line parameter calibration for precise injection quantity control. Fengjun Yan; Junmin Wang. *American Control Conference (ACC)*, 2010, Page(s): 2248 – 2253, IEEE Conference Publications
- [10] G. Stumpp, and M. Ricco, "Common rail – an attractive fuel injection system for passenger car DI Diesel engines," SAE paper 960870, 1996
- [11] K. Huhtala and M. Vilenius, "Study of a common rail fuel injection system," SAE paper, 2001-01-3184
- [12] P. Lino, B. Majone and A. Rizzo, "A control-oriented model of a common rail injection system for Diesel engines," *Emerging Technologies and Factory Automation, ETFA 2005*, 10th IEEE conference.



**Ernesto Gutiérrez.** PhD in Industrial Engineering from the Polytechnic University of Catalonia, Spain. Research Professor at the Laboratory of Thermal Engines in the Superior Technical School of Industrial Engineering of Barcelona. His main area of research is the electronic management of internal combustion engines, of which he is the author of several scientific papers and publications.



**Xavier Sala.** Industrial Engineering from the Polytechnic University of Catalonia, Spain. His main area of interest is thermal machines and the electronic management of the fuel injection.