Prototyping and Deployment of Real-Time Signal Processing Algorithms for Engine Control and Diagnosis

Fabrice Guillemin, Oliver Grondin, Emmanuel Kuentzmann

IFP
Outline

- Introduction
- Rapid prototyping platform for real-time signal processing algorithms
- Algorithm implementation for combustion analysis
- Deployment on industrial DSP based target
- Conclusion
Outline

- Introduction
- Rapid prototyping platform for real-time signal processing algorithms
- Algorithm implementation for combustion analysis
- Deployment on industrial DSP based target
- Conclusion
Introduction

Today issues:

- Increasing legal requirements (tailpipe emissions, fuel consumption) while keeping engine performances
- Development of innovative combustion concept: LTC, HCCI, CAI
- More sensitive combustion processes according to initial conditions (BGR, temperature, pressure) and fuel properties
- Injection system drift
- No direct control of the start of ignition

A closed loop combustion control is needed to ensure fuel loop robustness

- Combustion indicators (closed loop variables) can be computed from:
  - Cylinder pressure
  - Instantaneous engine speed
  - Ion current
- Need for higher sampling rate and suitable rapid prototyping platform
Real-time combustion analysis issue

- Need of relevant combustion parameters to characterize
  - combustion phasing (efficiency, emissions)
  - combustion noise
- In-cylinder pressure analysis provides significant information
- Acquisition and processing of in-cylinder pressure signals
  - cycle to cycle
  - cylinder to cylinder
  - engine synchronously

in-cylinder pressure

Middle of Combustion
Energy of Combustion
Start of Combustion

[°CA]
[Bar]
Outline

- Introduction
- Rapid prototyping platform for real-time signal processing algorithms
- Algorithm implementation for combustion analysis
- Deployment on industrial DSP based target
- Conclusion
Platform specification

IFP has developed a real-time platform for engine-synchronous algorithms prototyping. It fulfills the following specifications:

- engine events synchronization (TDC or 6°CA)
- eight continuous or multiwindowed acquisition channels
- acquisition at fixed frequency (400 kHz) or at fixed angular resolution (0.1°CA)
- cycle-to-cycle and cylinder-to-cylinder data availability for online signal processing and combustion analysis algorithms
- data recording for database acquisition and post-treatment purpose (over 1000 consecutive cycles)
Platform description

- Industrial PC based acquisition and rapid prototyping platform

Rapid prototyping platform:
Industrial PC with real time kernel

- High frequency acquisitions
- Engine TDC synchronous events
- Algorithm rapid prototyping facility
Platform description

- Industrial PC based acquisition and rapid prototyping platform

![Diagram of the platform with labels for CAN, CPU, RAM, A/D conversion, acquisition trigger, timer board, sensors, cam signal, and crank signal. The diagram also shows connections to the ACEBOX and power supply.]
Platform description: Hardware architecture

- CPU (xPC real time kernel)
- RAM (online data storage)
- HDD (data recording)
- IFP Timer board (PCI): FPGA (ALTERA Stratix)
  - Initially designed for generation of ignition and injection signals driven by ACEBox control system
- IFP Daughter acquisition board
  - Maximum sampling frequency: 400 KHz
  - Minimum sampling period: 0.1 °CA
  - 16 bit resolution (+/-10V with tuneable gain)
Platform description: acquisition functionalities

- Acquisitions frame-based, continuous or windowed in engine cycle
- Acquisitions in time or angle
- Acquired data frames updated engine synchronously (cylinder-to-cylinder and cycle-to-cycle)
- Acquired data frames available for processing with fast recursive algorithms
  - TDC synchronous results sending to external devices (bench supervisors, ECU)
- Data can be saved on PC hard disk drive for a specified number of consecutive engine cycles
  - post-processing purpose
Platform description : software implementation

TDC synchronous task implemented in Simulink

TDC event
Outline

- Introduction
- Rapid prototyping platform for real-time signal processing algorithms
- Algorithm implementation for combustion analysis
- Deployment on industrial DSP based target
- Conclusion
Combustion parameters computation

- Online TDC synchronous computation tasks include combustion analysis:
  - indicated mean effective pressure, indicated Torque
  - maximum pressure gradient and its location
  - peak pressure and its location
  - maximum rate of heat release and its location
  - locations of 10-50-90% of mass fuel burnt

\[
\frac{dQ_{comb}}{d\theta} = \frac{\gamma}{\gamma - 1} \ p_{cyl} \ \frac{dV_{cyl}}{d\theta} + \frac{1}{\gamma - 1} \ V_{cyl} \ \frac{dp_{cyl}}{d\theta} + \frac{dQ_p}{d\theta} = f(p_{cyl}, \theta)
\]

![Cylinder pressure graph](image)

![Rate of heat release graph](image)

![Mass fuel burnt graph](image)
AVL noise computation

- Noise computation algorithm is implemented using AVL Noisemeter specifications for engine structure filter shape and human ear filter shape
- Root mean square value is computed on filtered frame
- Modeled noise is obtained in dB

Acquired pressure frame

Structure Filter → Ear Filter → RMS → dB → Computed noise
AVL noise computation: filters implementation

- Fixed angular resolution with varying engine speed gives varying sampling frequency
- Filters coefficients updated every TDC, in order to maintain absolute bandwidth
- Approximation: engine speed considered constant during a cycle

\[ H(z) = H(s) \mid z = e^{\frac{2\pi k}{F_s}} \]

where \( k = \frac{2\pi f_s}{\tan(\frac{\pi f_s}{F_s})} \)

and \( F_s \) is the sampling frequency.

**Denominator coefficients**

\[
\begin{align*}
\alpha_1 &= -4k^4 - 2\alpha^2 k^3 + 2\alpha^3 k + 4\omega_c^4 \\
&\quad \frac{k^4 + \alpha^2 k^3 + \beta \omega_c k^2 + \alpha \omega_c^2 k + \omega_c^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4} \\
\alpha_2 &= \frac{6k^4 - 2\beta \omega_c^2 k^3 + 6\omega_c^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4} \\
\alpha_3 &= -4k^4 + 2\alpha \omega_c k^3 - 2\alpha^2 k^3 + 4\omega_c^4 \\
&\quad \frac{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4} \\
\alpha_4 &= \frac{k^4 - \alpha \omega_c k^3 + \beta \omega_c^2 k^2 - \alpha \omega_c^3 k + \omega_c^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4}
\end{align*}
\]

**Numerator coefficients (LP)**

\[
\begin{align*}
b_0 &= \frac{\omega_c^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4} \\
b_1 &= \frac{4\omega_c^2}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4} \\
b_2 &= \frac{6\omega_c^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4} \\
b_3 &= \frac{4\omega_c^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4} \\
b_4 &= \frac{\omega_c^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4}
\end{align*}
\]

**Numerator coefficients (HP)**

\[
\begin{align*}
b_0 &= \frac{k^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4} \\
b_1 &= \frac{-4k^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4} \\
b_2 &= \frac{6k^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4} \\
b_3 &= \frac{-4k^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4} \\
b_4 &= \frac{k^4}{k^4 + \alpha \omega_c k^3 + \beta \omega_c^2 k^2 + \alpha \omega_c^3 k + \omega_c^4}
\end{align*}
\]

where \( \alpha = 2[\cos\left(\frac{\pi}{8}\right) + \cos\left(\frac{3\pi}{8}\right)] \)

\( \beta = 2\left[1 + 2\cos\left(\frac{\pi}{8}\right)\cos\left(\frac{3\pi}{8}\right)\right] \)
AVL noise computation: filters implementation

- Implementation in Simulink, using Signal Processing Blockset and Embedded Matlab functions
AVL noise computation: filters implementation

- Implementation in Simulink, using Signal Processing Blockset and Embedded Matlab functions

![Diagram of AVL noise computation with filters implementation]
Model-based design, implementation and integration

Use of same Simulink model along all development process

- Specification and Simulation
  - Simulink is the receptacle of the functionalities to be developed
  - Simulink allows to test algorithm reliability offline, using simulation

- Development and test of the platform's drivers

- Integration with algorithms in a whole system model
  - System testing in real time, on xPC Target environment

- Validation of the final executable application
  - System validation in HIL conditions, with an engine signals simulator
  - Functional validation on a test bench or in a vehicle
  - Calibration online, using Simulink external mode or GUIs
Model-based design, implementation and integration

- Acquired data exploitation in model-based design
  - The platform's data storage ability permits to feed a full database with a complete engine mapping
  - With an offline analysis of this data, computation algorithms can be tested and pre-tuned from the early development and simulation phases
Outline

- Introduction
- Rapid prototyping platform for real-time signal processing algorithms
- Algorithm implementation for combustion analysis
- Deployment on industrial DSP based target
- Conclusion
Issue and functionalities

- **Need of an embedded industrial and energy-cost effective solution**
  - to deploy developed algorithms rapidly on vehicles and test benches
  - to provide efficient standalone tools for combustion closed-loop control achievement

- **Cost and time effective achievement**
  - Minimize modifications while moving validated application from prototyping platform to deployment hardware
  - Exploit existing targets and IDE
Issue and functionalities

- IFP has developed an industrial solution based on FPGA-DSP (TI C6727) hardware
- The functionalities to be embedded are:
  - in-cylinder pressure acquisition and sampling
  - combustion analysis and noise computation
  - cylinder-to-cylinder and cycle-to-cycle updating of combustion parameters
  - result availability to external devices
  - algorithm parameterization from external devices
Hardware description

This device is composed of:

- A customized timer board
  - includes an FPGA, which manages:
    - engine and angular coder signals, such as cycle trigger and 0.1°CA trigger
    - acquisitions
    - communication protocols with external devices (mainly CAN)
    - memory mapping

- An 8-channel acquisition board
  - 8 dedicated 16-bit ADCs
  - high-frequency acquisition (800 kHz or 0.1°CA)

- A TI C6727 DSP module
  - targeted by Real-Time Workshop to execute a Simulink modeled application
Hardware description

- Deployment platform scheme

Targeted DSP
TI C 6727

FPGA

Digital conditioning

Analog conditioning

CAN

Synchronization signals:
- angular coder
- camshaft
- reference cylinder TDC

Sensors:
- cylinder pressure
- AFR
- Fuel pressure

External devices:
- ECU
- bench supervisor
- prototyping platform
- Monitoring facility

USB 2.0 interface
LAN 10/100M
User interface LCD display
Software implementation

- Aim: bring the application from a PC-based rapid prototyping environment to an industrial DSP-based target
- Direct use of the Simulink code implemented on prototyping platform
- Additional tools exploited in the code generation process:
  - Real-Time Workshop® Embedded Coder™
  - Target Support Package™ TC6 (for TI’s C6000™ DSP) and its corresponding target function library (TFL)
  - Embedded IDE Link™ CC (for TI’s Code Composer Studio™)
Software implementation

- Real-Time Workshop Embedded Coder configuration using TI C67x library
- Using TFL instead of complete ANSI C code generation has improved code execution performance by a factor of 5, for frame-based AVL computation algorithm
Software implementation

- The application embedded in the DSP gets acquired data and gives computation results
  - Data accessibility management by mapping DSP memory
  - Memory addresses specified by code variables
  - Add of specific headers and initialization code in the model and in the custom target configuration.
Outline

- Introduction
- Rapid prototyping platform for real-time signal processing algorithms
- Algorithm implementation for combustion analysis
- Deployment on industrial DSP based target
- Conclusion
Conclusions & Perspectives

- Future engine technologies applying new combustion systems require a closed loop combustion control
- IFP has developed a rapid prototyping platform for high frequency data acquisition and signal processing algorithm development
- By acquiring in-cylinder pressures, online analysis is possible in real-time TDC synchronously, allowing for the estimation of combustion phasing and noise
- Validated algorithms are being deployed on a standalone industrial DSP-based solution developed at IFP
- Other issues addressed with signal processing platform:
  - Fuel pressure measurement for diagnostic and control purpose
  - AFR rapid measurements for control purpose
  - Instantaneous engine speed measurement for torque estimation
  - CAI applications
- Implementation, integration, and calibration phases of processing algorithms have been simplified and accelerated thanks to the use of The MathWorks toolchain and model based design approach
Thank you!

fabrice.guillemin@ifp.fr
Annex : ACEbox Control System

Rapid Prototyping based on Mathworks xPC Target®

Rapid Prototyping of control algorithms in real time on an engine test bench...

ToolBox xPC Target &
Real-Time WorkShop
(MathWorks)

PowerTrain Control

ACEbox II

Matlab/Simulink
(MathWorks)

The ACEbox II system includes all the necessary software components for rapid prototyping of control algorithms. The system integrates a gasoline and diesel torque structure with all Simulink’s drivers allowing to configure and modify control parameter in only one Drag & Drop.

Main Characteristics of the ACEbox® II System

- Operation of gasoline or diesel engines from 1 to 6 cylinders.
- Multiple injection & Spark events (from 1 to 8 events per engine cycle).
- Accurate operation of solenoid valves and motored throttle valves.
- Engine synchronisation on :
  - 2 AAC inputs of any type (with measuring the out of phase),
  - the engine crank by a captor of vehicle’s target 60-2 or by a crank angle encoder with different resolutions.
- OMERE® user Interface with Windows XP on PC supervisor linked by TCP/IP 100Mbits with central unit ACEbox® II.
- ACEbox® II based on PC architecture with timer card dedicated to engine applications.
- Optimised overall dimensions and fitted to its industrial integration on an engine test bench.
- Easy connection with captors and actuators cables of engine by separate connectors including power pins for all input.
- ACEbox® II powered by 220VAC/50Hz and interface’s modules with captor and actuators cables of engine powered by 12VCC.