

Model Predictive Combustion Control Implementation Using Parallel Computation on an FPGA

Peter Fussey
Ricardo

David Limebeer
University of Oxford

ABSTRACT

The introduction of transient test cycles and the focus on real world driving emissions has increased the importance of ensuring the NO_x and soot emissions are controlled during transient manoeuvres. At the same time, there is a drive to reduce the number of calibration variables used by engine control strategies to reduce development effort and costs.

In this paper, a control orientated combustion model, [1], and model predictive control strategy, [2], that were developed in simulation and reported in earlier papers, are applied to a Diesel engine and demonstrated in a test vehicle.

The paper describes how the control approach developed in simulation was implemented in embedded hardware, using an FPGA to accelerate the emissions calculations. The development of the predictive controller includes the application of a simplified optimisation algorithm to enable a real-time calculation in the test vehicle.

The test vehicle was calibrated over a constant speed step manoeuvre and then demonstrated over the standard US06 drive cycle, where it was found to reduce NO_x and CO_2 emissions.

The paper concludes with a discussion on the merits of an emissions based engine control strategy, where the controller seeks to control emissions directly rather than an indirect quantity such as boost pressure.

CITATION: Fussey, P. and Limebeer, D., "Model Predictive Combustion Control Implementation Using Parallel Computation on an FPGA," *SAE Int. J. Engines* 9(2):2016, doi:10.4271/2016-01-0817.

INTRODUCTION

The emissions legislation for internal combustion engines is being progressively updated to accommodate 'real world' driving conditions; particularly transient manoeuvres. For example, EURO 7 emissions limits are likely to be measured using the proposed World harmonised Light duty Test Procedure (WLTP) drive cycle, which is significantly more dynamic than the previous New European Drive Cycle (NEDC). In addition, off-cycle Real Driving Emissions (RDE) are expected to be regulated with requirements to demonstrate compliance through Portable Emissions Measurements Systems (PEMS).

Most of the opportunities for reducing engine out emissions during a drive cycle by improving engine control relate to transient manoeuvres, since steady-state operating points are already extensively optimised and calibrated. The calibration of all the possible transient conditions would require a significant increase in memory within the Engine Control Unit (ECU). To avoid this situation, there is a move to model based control strategies which use models of the engine systems to estimate transient responses.

In previous publications [1] and [2], the authors have developed a control orientated combustion model and associated predictive control approach in simulation, that minimises NO_x emissions over a transient manoeuvre. This paper describes the implementation of this control approach in a demonstration vehicle.

This study also considers how the control architecture could be reframed in terms of controlling emissions rather than quantities such as air mass flow or EGR ratio. This approach has also been discussed in [3] where a closed loop emissions control is developed for Diesel engines, providing a greater robustness to emissions performance over different driving cycles. This idea is extended in [4] where the engine controller uses the vehicle location to modify the emission/fuel consumption tradeoff within the engine control unit. For example, favouring low emissions in urban areas and low fuel consumption on highway driving.

The paper starts with a re-cap of the modelling and control development, followed by a description of key steps in the implementation of the control strategy in vehicle; control hardware selection, updates to control algorithms to take advantage of control hardware and integration into vehicle control systems. Vehicle test results are presented for a step load change and then over the US06 drive cycle.

BACKGROUND

Diesel Combustion Control

A typical Diesel engine is shown in [Figure 1](#). The engine includes many systems to control the combustion events in the cylinder; Exhaust Gas Recirculation (EGR), turbocharging, and the cooling of intake gases. The systems can be grouped into the fuel path and the air path. The control of the fuel path can influence the next combustion event, since the fuel is injected directly into the cylinder, giving a response of ~ 0.01 seconds [5]. The control of the gaseous mixture entering the cylinder is slower due to the transport delays from the EGR valve to the cylinder (~ 1 second) and boost pressure is slower again, constrained by the inertia of the turbo-charging system ($\sim$ a few seconds).

In general, the control of the combustion process is achieved by varying the fuel injection pressure, the quantity and timing together with the state of the gases coming into the engine; boost pressure and EGR ratio. Without direct cylinder pressure measurement, the combustion is controlled in an open-loop manner, with set-points for the fueling and inlet gases calibrated at steady-state operating points on engine dynamometers. During transient operation, the set-points move according to the driver inputs - typically torque and engine speed - with some transient modifications for features such as smoke limitations. However, the conditions at which the engine is operating during the transient are often different to those conditions used for calibrating the set-points. It is this situation that this work is seeking to address.

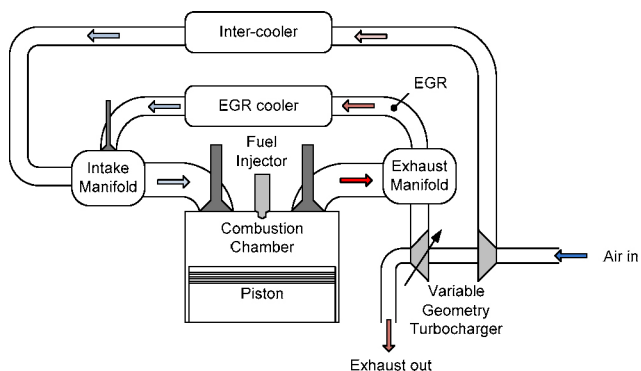


Figure 1. A typical Diesel engine architecture, focused on the combustion system and showing the flow of gases. System includes an EGR circuit, EGR cooler, direct fuel injection, variable geometry turbo-charger (VGT) and an intercooler.

Combustion Model

The online optimization of combustion processes requires a combustion model in order to assess the response of the system to different control actions. The control objectives help define the modelling detail required, for example, if the controller's function is the minimization of emissions, then the model should estimate emissions trends resulting from control changes. Whereas if the controller is designed to keep the emissions below a specified level, the model should include a model of the level of the actual emissions. Since it is difficult for full 3D CFD models to reliably predict absolute exhaust emissions, a pragmatic target for the control orientated model is to predict emissions trends.

A combustion model has been developed, [1], which extends the classical heat release estimations to include a representation of the fuel air mixture. The fuel air mixture is described by a beta probability distribution with the key parameters (mean and variance) evolving as a function of the control input parameters (rail pressure, injection duration and injection timing). This representation allows a time varying, in-cylinder mixture distribution as a function of fuel air ratio. At each timestep, the mass burnt is calculated as a function of fuel air ratio and integrated to give the time varying heat release. The flame temperatures and finally an estimate of the emissions follow on from the heat release calculation and also take the varying fuel air mixture into account. The combustion model is summarised in [Figure 2](#).

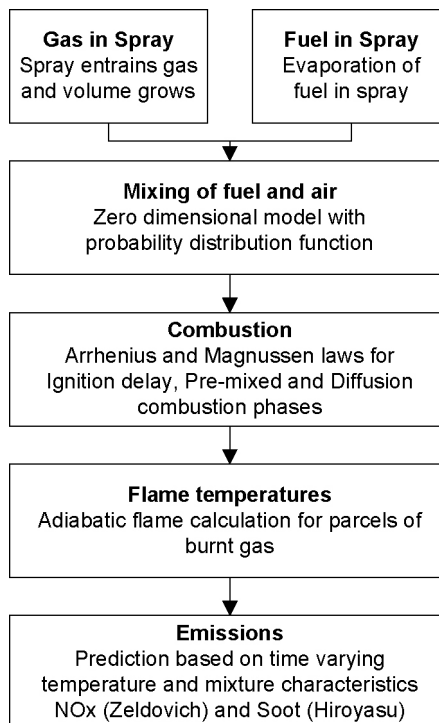


Figure 2. Overview of combustion model.

Control Strategy

The goal of the predictive combustion controller, [2], is to minimise emissions during a transient manoeuvre, using conventional control variables such as start of injection (SOI) and EGR ratio. The

dynamics for the SOI and EGR are relatively fast (compared to boost pressure) so the control problem effectively becomes an optimisation at each timestep.

The objective function J , selected for this optimisation seeks to minimize the difference between the NO_x emissions calculated with the current cylinder conditions and the control action, $y(x, u)$, and the NO_x emissions at the steady-state set-point, $y(\bar{x}, \bar{u})$ plus the difference in control parameters, u , from those at the steady-state set-point, $\bar{u}$

$$J(u) = (y(x, u) - y(\bar{x}, \bar{u}))^2 + Q(u_1 - \bar{u}_1)^2 + R(u_2 - \bar{u}_2)^2$$

where Q and R are weightings on the NO_x emissions error and the differences in the control inputs, x are the in-cylinder states. Note: this objective function will tend to drive the system to the set-point as the engine states stabilize to steady-state conditions, even if the model predicts a better NO_x emission with different control settings. This takes advantage of existing extensive steady-state calibration, whilst allowing a deviation from these points during a transient when the engine states do not match those at the steady-state operating point. Figure 3 shows the predicted reduction in NO_x over a transient, using a Mean Value Engine Model to simulate engine response and the control orientated emissions model to simulate emissions response.

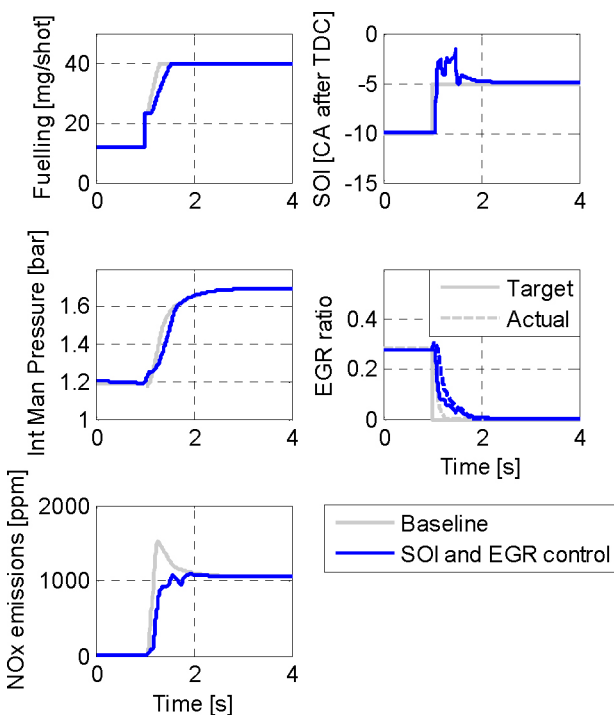


Figure 3. Simulation results from [2] showing SOI and EGR control over a transient

IMPLEMENTATION OF COMBUSTION CONTROLLER

Selection of the Controller Hardware

One challenge for vehicle implementation was to run the combustion model several times every second to carry out an optimisation. The control orientated combustion model had already been simplified as

far as possible whilst retaining the physical based structure and was still too slow for real-time implementation, taking 300ms for one cycle simulation. Rather than simplifying further, this project considered accelerating the calculations using different options for the computational hardware. In particular, the application of a parallel computation engine was studied using a Field Programmable Gate Array (FPGA).

For this project, a combined FPGA and micro-processor (known as a System on a Chip, SOC) was selected to facilitate integration with the vehicle control systems, since communication interfaces are more readily available for the micro-processor. Specifically the ZYNQ SOC from Xilinx, Figure 4, was selected as a realistic hardware option since it has an automotive grade option and an integrated tool-chain from the Mathworks™.

A secondary objective of this project was to explore the integration of FPGA hardware into a model based, autocoding design process.

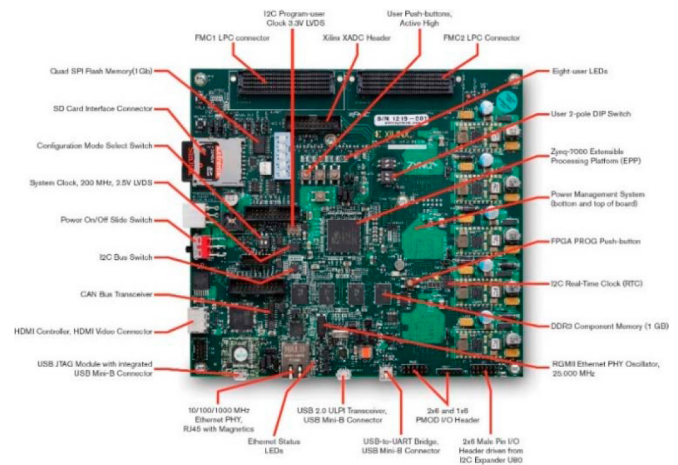


Figure 4. ZYNQ system on a chip ZC702 development board [6]

Integration of FPGA into Model Based Control Development Processes

The ease of use and process integration is a key consideration in the subsequent adoption of new control hardware and software. Model based control development is a widely adopted process for engine control algorithm development, using models of the engine systems, graphical representation of the control algorithms and finally auto-coding tools to generate the embedded software automatically.

The Mathworks™ tool-chain has been expanded to support FPGA code development. In particular the whole control algorithm can be developed and tested within a single Simulink™ model. The individual control modules can be assigned to either the microprocessor or the FPGA, with the tool-chain creating the interface bus (AXI) between the micro-processor and the FPGA. The tool-chain outputs auto-coded C files and VHDL files for flashing onto the micro-processor and FPGA respectively.