UNIVERSITÀ DELLA CALABRIA

Dipartimento di Elettronica,
Informatica e Sistemistica

Dottorato di Ricerca in
Ingegneria dei Sistemi e Informatica
XXII ciclo

Tesi di Dottorato

Tools and Methods for Engine Control Systems Development and Validation

Ferdinando De Cristofaro

UNIVERSITÀ DELLA CALABRIA

Dipartimento di Elettronica,
Informatica e Sistemistica

Dottorato di Ricerca in
Ingegneria dei Sistemi e Informatica
XXII ciclo

Tesi di Dottorato

Tools and Methods for Engine Control Systems Development and Validation

Ferdinando De Cristofaro

Coordinatore
Prof. Luigi Palopoli

Supervisore
Prof. Alessandro Casavola

DEIS

DEIS- DIPARTIMENTO DI ELETTRONICA, INFORMATICA E SISTEMISTICA
Novembre

Settore Scientifico Disciplinare: ING-INF/05

Summary

Nowadays the automotive world is changing fast. This extremely competitive market demands excellent performance and the new emissions legislation is imposing very strict constraints, which have both to be met in a really short time. Actually, the vehicle development time is becoming shorter and shorter: time to market imposes a vehicle cycle much smaller than previous years. On the other hand, the expected quality level is increasingly high and safety requirements do not permit any failure. Consequently, the automotive companies have to reconfigure their development processes and identify the best tools to face this “extreme” situation. Without a fast adaptation to the new scenario the companies risk not to survive. Recent experiences have taught everybody involved in cars development that the extraordinary has become the ordinary way and the continuous improvement is no longer a choice but a need.

In this scenario the electronic systems are a key factor. It has been estimated that 90% of automotive innovation includes the electrical and electronics parts. In this situation the electronic system development time has been drastically reduced (i.e. a new engine and its control system development is reduced from 25 months to 18 months).

Furthermore, during my personal work experience (about 15 years in the automotive world) I observed how, up to some years ago, the automotive companies (car makers and their suppliers) adapted their development process to make it suitable with the available tools. In other words, the development processes were driven by tool vendors, who proposed solutions to companies and asked them to adapt their processes and “problems” to be solved with their products. Only recently the same companies understood that their processes should be driven by their specific and real design problems and so the request for specific and non-existing tools became bigger and bigger (AUTOSAR: Automotive Open System Architecture is only the most common example).

The topic of this thesis is the development of tools for supporting and, if possible, automating the most critical phase of an engine control system realization process.

The most part of my research has been carried out inside the company I work for. So the application of the results has been quite immediate, showing the consequent benefits. In particular, a remarkable reduction of the development costs and times have been evaluated: until ten times. Actually, the main results, which will be detailed in the next sections of this thesis, are the

development of engine model, an environment for control systems design and calibration and the design and implementation of a low cost Hardware In the Loop simulator for control systems verification and validation.

The thesis is organized in nine chapters: the first three describe general aspects of the environment in which the thesis is placed, the following are a synthesis of some papers and patents of which I am the author or the inventor (see Chapter 10 for details).

In particular in the first chapter the main aspects of the way the internal combustion engine works, are analyzed. Successively, in the second chapter, some technological enhancements, which increase the fuel economy and decrease the emissions while maintaining or improving the engine's performance, are described, lingering also over the MultiAir technology. In the third chapter, the typical working of the engine control unit, defining some of the principal functions as that of torque control and air-fuel control is considered. Moreover, a description of the specifications and of the working of the sensors and actuators utilized in the engine system is given. The fourth chapter is dedicated to the description of a mean value engine model, by means of a modeling approach for engine dynamics based on electrical analogy. This approach has been described in two papers of mine ([16], [17]). In the fifth chapter the tools for engine control system design and calibration have been described. This chapter is represented by a chapter of the book "Large scale Computation, Embedded Systems Computer Security, ISBN: 9781607413073, Nova Science Publisher, NY, 2010", written by two other authors and myself (see [18]). The sixth chapter describes the SW verification by means of HIL simulator and which introduces the necessity of the object of the seventh chapter that describes the low cost systems for Hardware In the Loop simulation, (which is detailed in some papers of mine, in particular see [58]). Finally the eighth chapter gives an overview of some experimental applications of presented tools and methods to Variable Valve Actuation (VVA) engines: engine model for an investigation on idle speed control that uses the variable valve actuator ([74], [75]); performance and evaluation tools (described in chapter 5) applied to such an engine ([76]); development of new algorithms (two of which have been patented) by means of tools described in chapter 5 ([77], [78] and [79]).

Keywords:
Engine Management System, Engine Control Unit, Engine Model, Calibration Tool, Hardware In the Loop, Variable Valve Timing, MultiAir.

Table of contents

List of figures

Chapter 1 Internal Combustion Engine

The Internal Combustion Spark Ignition Engine is again becoming the most popular engine with respect to the compression ignition engine (well known as diesel engine), because of future emission legislations (for instance Euro 5+ and Euro 6), that will make the compression ignition engines more and more expensive. In this chapter an overview of the Internal Combustion Engine is given. The working cycle of the engine is described with a focus on main engine parameters, in particular on the timing system.

1.1. INTRODUCTION

The engines used for traction can be divided into two main categories (Figure 1.1):

- **Heat engines**: these types of engines derive heat energy from the combustion of fuel or any other source and convert this energy into mechanical work. Heat engines may be classified into two main classes as follows:
  - *External combustion engines*: combustion of fuel takes place outside the cylinder (like steam engines, where the heat of combustion is employed to generate steam which is used to move a piston in a cylinder);
  - *Internal combustion engines*: combustion of the fuel with oxygen of the air occurs within the cylinder of the engine.
- **Electric motors**: they use electric power that is stored in batteries carried on the vehicle. [1], [2]

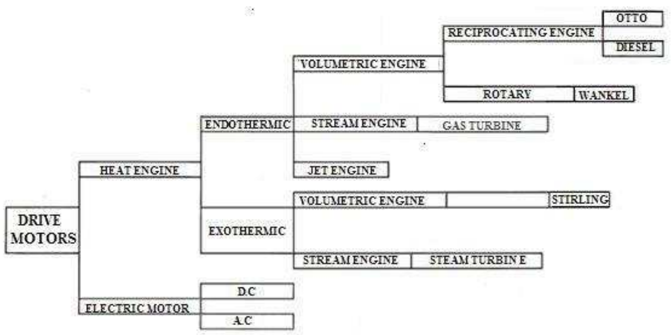

Figure 1.1 Classification of drive motors

In the automotive field, most of the internal combustion engines (endothermic) are volumetric and reciprocating. In particular, the engines are volumetric because the working fluid acts on moving parts that describe cyclically a variable volume and they are alternative because the motion is that of a piston inside a cylinder. According to the method of ignition, the internal combustion engine may be classified as:

- *Spark ignition* (SI): an SI engine starts the combustion process in each cycle by means of a spark plug. The spark plug gives a high-voltage electrical discharge between two electrodes and ignites the air-fuel mixture in the combustion chamber surrounding the plug. Typical fuels for such engines are gasoline, natural gas and sewage and landfill gas.
- *Compression Ignition* (CI): the combustion process in a CI engine starts when the air-fuel mixture self-ignites, due to high temperature in the combustion chamber caused by high compression. Diesel fuel oil is normally used in compression ignition engines, although some are dual-fueled (natural gas is compressed with the combustion air and diesel oil is injected at the top of the compression stroke to initiate combustion)

The internal combustion reciprocating engine (Figure 1.2) consists, essentially, of one or more cylinders in which pistons flow air-tight and with little resistance. Having more cylinders in an engine yields two potential benefits: first the engine can have a larger displacement with smaller individual reciprocating masses (that is, the mass of each piston can be less) thus making a smoother running engine (since the engine tends to vibrate as a result of the pistons moving up and down). Second, with a greater displacement and more pistons, more fuel can be combusted and there can be more combustion events (that is, more power strokes) in a given period of time, meaning that such an engine can generate more torque than a similar engine with fewer cylinders. The pistons are connected to crankshaft via the connecting rod and they convert their rectilinear motion into rotational motion of the crankshaft. The cylinders are filled cyclically from a charge of *fresh fluid* through a valve system that should be controlled. The fluid present in the cylinders is an air- fuel mixture or is constituted of air only to which the fuel is added after placing it in the cylinder. The combustion of the working fluid leads to a significant increase in temperature of the fresh charge, allowing the fluid to make a thermodynamic cycle from which work can be transferred to pistons.

The internal combustion engine is a heat engine that converts chemical energy in the fuel into mechanical energy, usually made available on a rotating output shaft. Chemical energy of the fuel is first converted to thermal energy by means of combustion or oxidation with air inside the engine. This thermal energy raises the temperature and pressure of the gases within the engine, and the high-pressure gas then expands against the mechanical mechanisms of the engine. This expansion is converted by the mechanical linkages of the engine to a rotating crankshaft, which is the output of the engine.

The internal combustion engine consists of fixed and moving parts. The main fixed parts are: engine block , head and oil sump.

The *block* is the body of the engine containing the cylinders and is limited by the head superiorly and by the oil sump inferiorly. It has traditionally been made of gray cast iron because of its good wear resistance and low cost. The *head* seals off the cylinders, usually containing part of the clearance volume of the combustion chamber. It contains the fuel injectors and the intake and exhaust valves with their control elements. Furthermore, it is connected to intake manifold (piping system which delivers incoming air to the cylinders), exhaust gas (piping system which carries exhaust gases away from the engine cylinders), and also to the piping system of the water radiator. The *oil sump* is the lubricating oil tank.

The main moving parts, which transform the reciprocating motion to rotary motion, are: piston, connecting rod and crankshaft.

The *piston*, or plunger, has the combustion chamber at the top, called crown, and on the other side, called skirt, it has the hole for the crank pin and locations for snap seals and scraper rings. The piston both seals the cylinder and transmits the combustion- generated gas pressure to the crank pin via the connecting rod. The *connecting rod* has a socket at the top, called small end of connecting rod, which is for articulation with the piston pin and at the base, called big end of connecting rod, there is a bearing where to fasten to the crankshaft. The connecting rods transmit the reciprocating motion of pistons, turning it into rotary motion, to the crankshaft. The connecting rod is a two-force member; hence it is evident that there are both axial and lateral forces on the piston at crank angles other than 0° and 180°. These lateral forces are, of course, opposed by the cylinder walls. The resulting lateral force component normal to the cylinder wall gives rise to frictional forces between the piston rings and cylinder. It is evident that the normal force, and thus the frictional force, alternates from one side of the piston to the other during each cycle. Thus the piston motion presents a challenging lubrication problem for the control and reduction of both wear and energy loss. The drive shaft, or *crankshaft*, is supported in main bearings. The maximum number of main bearings would be equal to the number of pistons plus one, or one between each set of pistons plus the two ends. On some less powerful engines, the number of main bearings is less than this maximum. At the rear end of the crankshaft, the flywheel (rotating mass with a large moment of inertia) with ring gear for electric motor (starter) is connected. The purpose of the flywheel is to store energy and furnish a large angular momentum that keeps the engine rotating between power strokes and smoothes out engine operation. The crankshaft, through a suitable drive train, controls the tree or trees of distribution, called camshafts; cams, through tappet, command intake and exhaust valves to permit the entry of fresh gas and the expulsion of the residuals of the combustion [1].

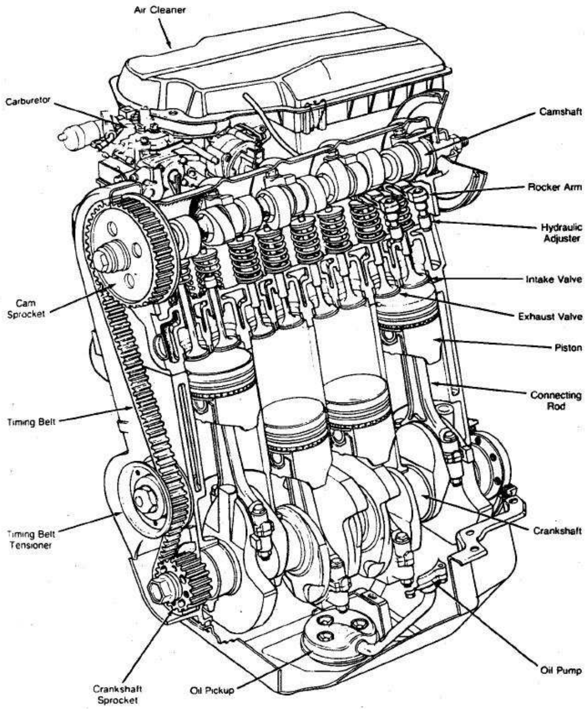

Figure 1.2 Internal Combustion Engine

The name engine cycle means all transformations undergone by the air-fuel mixture inside the cylinder and which are repeated periodically or cyclically. This air-fuel mixture in the cylinder initiates the combustion process by using one or more spark plugs which shoot, under some conditions, in one or more points.

Regarding the motion, while the crank arm rotates through 180°, each piston moves from the position near the upper side of the cylinder, called Top Dead Center (TDC), to the other extreme called Bottom Dead Center (BDC). During this period the piston travels a distance, called the stroke, that is twice the length of the crank. In other words, it is said that the piston is at TDC when it stops at the furthest point away from the crankshaft and reverses the movement and in this place, the cylinder volume is at the minimum (clearance volume). Correspondingly, the position of BDC is assumed by the piston when it stops at the point closest to the drive shaft and reverses the movement and the cylinder volume is at a maximum. The cylinder volume corresponding to the piston stroke is the displacement, while the air volume that exists between the cylinder and the piston head when it is at TDC, is called the combustion chamber. An important parameter for internal combustion engines is the ratio of total volume to clearance volume, known as the compression ratio. Typical values of the compression ratio for spark ignition engines (SI) are 8 to 12 while for compression-ignition engines (CI) they are 12 to 24. The efficiency of the engine is function of the compression ratio. It has a strong influence on the formation of pollutants. The diameter inside the cylinder (and, approximately, also the piston diameter) is called bore. Cylinder bore, stroke, and number of cylinders are usually quoted in engine specifications along with engine displacement (the product of the piston displacement and the number of cylinders).

Each stroke corresponds with half a turn of the crankshaft, hence each cylinder requires four strokes of its piston (two revolutions of the crankshaft, or 720 degrees) to complete the sequence of events which produces one power stroke. These alternative engines, where the cycle requires four strokes of the piston, are called four strokes.

1.2. THE FOUR- STROKE OPERATING CYCLE

The piston, cylinder, crank, and connecting rod provide the geometric basis of the internal combustion engine. The majority of reciprocating engines operate on what is known as the four-stroke cycle. In this engine the piston undergoes two mechanical cycles for each thermodynamic cycle. The intake and compression processes occur in the first two strokes, and the power and exhaust processes in the last two. The purpose of an internal combustion engine is to transform the energy into mechanical power produced by the combustion of the air-fuel mixture. In order to obtain this scope, the cylinders must adequately be charged from the air-fuel mixture; a discharge system must expel the residuals of the combustion in order to permit a new filling of the cylinders and therefore another combustion. Figure 1.3 also shows the four strokes for a spark-ignition engine (Otto cycle):

- *Intake stroke*: the piston travels from TDC to BDC with the intake valve open and the exhaust valve closed. The air-fuel mixture in the manifold enters through the intake valve in the cylinder after the air has passed through the throttle, which regulates the amount of air flow into an SI engine, and the fuel has been introduced by the injectors placed in the intake manifold. In the running down to the BDC, the piston causes a vacuum. The opening of the valves connects the cylinder with the external environment, a strong current of air that passing is mixed with gasoline. When the piston is at BDC, the intake valve closes. In fact, the intake valve opens shortly before TDC and closes shortly after BDC in order to increase the mass drawn into the cylinder and to maximize the inertia even though the piston is rising.
- *Compression stroke*: the piston at the end of the intake stroke climbs from BDC to TDC and starts the compression phase. The intake and exhaust valves are both closed, while the piston, moving from the bottom to the top, compresses the mixture inside the cylinder to a small fraction of its initial volume (combustion chamber). The pressure and the temperature of the mixture in the cylinder increase due to compression. Toward the end of the compression stroke, a spark, between the electrodes of the spark plug, can ignite the mixture quickly. The combustion is initiated. The cylinder pressure rises

rapidly and a boost to the piston carries out the work required. To reduce the work absorbed by the compression, the ignition had better advance so that combustion can occur near TDC.

- *Expansion stroke or Power stroke*: the power stroke, or expansion stroke, is the work producing process of the cycle. Towards the end of the compression phase starts the combustion generated by means of a spark plug. The combustion produces an high increase of pressure and temperature of the fluid contained into the cylinder that carries useful work during the new descendent stroke of the piston. Therefore the high pressure created by the combustion process pushes the piston away from TDC and forces the crank to rotate. As the piston travels from TDC to BDC, cylinder volume is increased, causing pressure and temperature to drop. Obviously, during the power phase, to make the expansion work any leakage of gas from the cylinder having to avoid, both intake and exhaust valves are closed. As the piston approaches BDC the exhaust valve opens to initiate the exhaust process and drop the cylinder pressure to close to the exhaust pressure. In fact, the exhaust valve opens ahead of BDC encouraging the spontaneous release of burnt gas of the cylinder which have a pressure considerably higher than the external environment.
- *Exhaust stroke*: to the end of the expansion phase the piston is at the bottom and the cylinder is occupied from the burned mixture which must be expelled. At this point the exhaust valve is opened and the gases enter into the exhaust manifold for the high pressure reached in the cylinder. The motion of the piston towards the top expels the remaining part of the residuals of the combustion. To the end of the phase the piston is at the top and the burned gases have been completely expelled. Therefore when the piston approaches TDC the exhaust valve is closed and the intake valve is opened: the exhaust valve closes later than TDC, to allow the piston to use fully and effectively the reclimbing stroke. [1],[3]

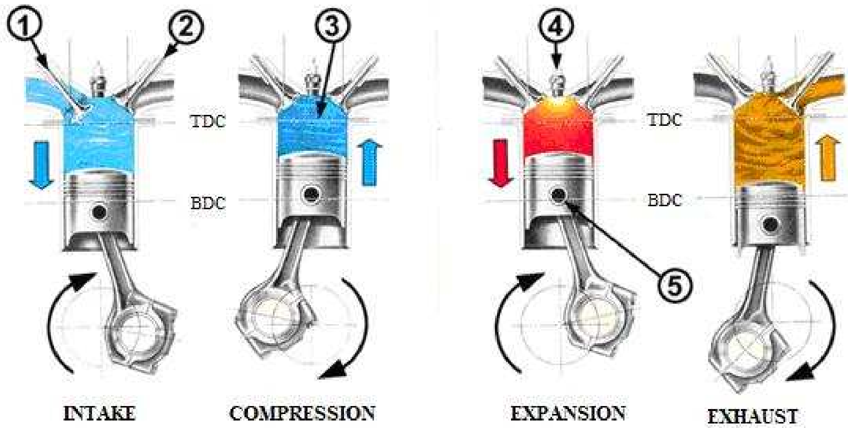

Figure 1.3 The four basic processes in a four-stroke engine: (1) Intake valve; (2) Exhaust valve; (3) Combustion chamber; (4) spark plug; (5) Piston

1.3. IDEAL AND REAL ENGINE CYCLE

The ideal cycle, called **Otto cycle**, is the theoretical cycle commonly used to represent the processes in the spark ignition (SI) internal combustion engine and is generally represented by a Cartesian diagram on which abscissa are the volumes generated by the piston and on the ordinate are the corresponding pressures that the fluid reaches inside the cylinder. It is assumed that a fixed mass of working fluid is confined in the cylinder by a piston that moves from BDC to TDC and back, as shown in Figure 1.4

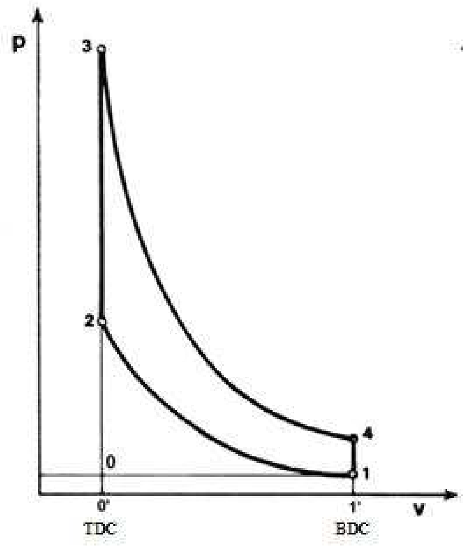

Figure 1.4 Otto cycle

The ideal Otto cycle consists of following reversible (no internal losses) processes:

- **isobar intake** 0 → 1: the intake stroke of the Otto cycle starts with the piston at TDC and is a constant-pressure process (ambient pressure). The piston moves from TDC to BDC producing a change in volume;
- **isentropic compression** 1 → 2: the compression produces a drastic reduction of volume and a significant increase in pressure and temperature. The piston moves from TDC to BDC;
- **isochoric heat addition** 2 → 3: the instantaneous combustion of the mixture happens at constant volume while the piston is stationary at TDC;

- **isentropic expansion** 3 → 4: the expansion produces a drastic reduction of pressure and an increase of volume. The piston moves from TDC to BDC;
- **isochoric instantaneous exhaust** 4 → 1: it is the first part of the exhaust phase during which the pressure reduces while the volume is constant;
- **isobar exhaust** 1 → 0 : the transformation occurs at constant pressure; in this phase the discharge of the residuals of combustion is completed reporting the cycle to the starting position.

The theoretical work developed from the fluid is represented, on *p* - *V* diagram (pressure-volume diagram), by the area(1 – 2 – 3 – 4):

- area (0' – 3 – 3 – 1') represents the expansion work L_1 (positive);
- area (0' – 2 – 1 – 1') represents the compression work L_2 (negative).

The *p* - *V* diagram effectively done from the fluid during engine operation is called **real cycle**, or indicated cycle (Figure 1.5). It plots the cylinder pressure and corresponding cylinder volume throughout the engine cycle. Compared to the ideal cycle, it has rounded tops and a second area that expresses the work transfer between the piston and the cylinder gases during the inlet and exhaust strokes. Figure 1.5 highlights the work delivered from the gas to the piston (*active work A* that is the work correspondent to the positive area of the cycle) and the work done by the piston on the gas (passive work or *pumping work B*). The difference between *A* and *B* allows to obtain the indicated work that is transferred by gas to the piston. Figure 1.6 highlights the substantial difference, both in form and in values of temperatures and pressures, between indicated and actual cycle.

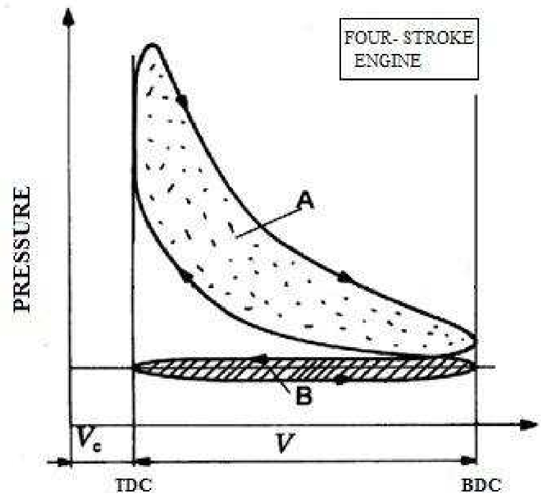

Figure 1.5 Indicated cycle

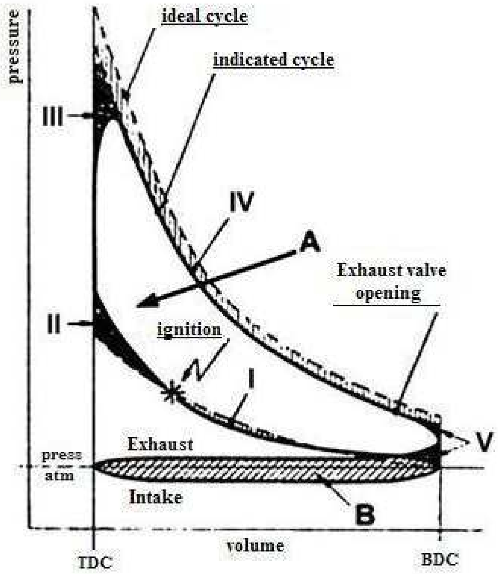

Figure 1.6 Differences between ideal and indicated cycle

The differences between the actual and theoretical cycle are a result of several factors.

- Heat loss to the cylinder walls. There are heat losses during the cycle of a real engine which are neglected in the theoretical cycle. To ensure the smooth piston operation, the cylinder has to be cooled so there is some heat transfer from fluid to the walls. Heat loss during combustion lowers actual peak temperature and pressure from what is predicted. The actual power stroke, therefore, starts at a lower pressure, and work output during expansion is decreased. Heat transfer continues during expansion, and this lowers the temperature and pressure below the ideal isentropic process towards the end of the power stroke. The result of heat transfer is a lower indicated thermal efficiency than predicted by ideal cycle. Heat transfer is also present during compression, which deviates the process from isentropic. However, this is less than during the expansion stroke due to the lower temperatures at this time. Therefore, the compression and expansion lines are not adiabatic but polytrophic because the fluid losses heat. This implies a loss of useful work corresponding to the algebraic sum of the areas *I* and *IV* in Figure 1.6.
- Combustion that is not instantaneous. In theoretical cycle, combustion is assumed instantaneous; in real cycle, instead, combustion requires a short time (but greater than 0) to occur with the result that, in the meantime, the piston moves and the transformation cannot be at constant volume. If the ignition starts at the TDC, the combustion

would proceed during the moving piston from the TDC and the pressure would be lower than expected, leading to loss of useful work. Therefore, the ignition is anticipated so that combustion can occur mostly when the piston is near TDC. This implies a rounding of the theoretical line of introduction of heat and the loss of useful work, represented by areas *II* and *III* (Figure 1.6), but which is much less than that occurs without spark advance.

- Exhaust valve opening. In theoretical cycle the exhaust is assumed that is made, instantly, at the BDC. In fact, this phase lasts for a relatively long time. The exhaust valve should be open in advance to allow a portion of exhaust gases to exit from the cylinder before the piston arrives to the BDC, so that the pressure at the end of the expansion stroke is reduced below that ambient. This causes a loss of useful work represented by areas *V* (Figure 1.6), which is however less than that would occur without the advance. The exhaust valve remains, also, open even at the start of intake stroke to ensure complete expulsion of exhaust gases.
- Cylinder pressure. As Figure 1.6 shows, during the intake stroke the cylinder pressure is less than that which occurs during the exhaust stroke. Except in special cases, the intake is completely depressed while, during the exhaust phase, the pressure is above that ambient setting, as can be seen from the diagram, negative area *B* that corresponds to the pumping work.

1.3.1. Valve timing diagram

The timing of the valves (i.e. their opening and closing with to the travel of the piston) is very important factor for efficient working of the engine. Theoretically the valves open and close at top dead centre (TDC) or at bottom dead centre (BDC) but practically they need time before or after the piston reaches the upper or lower limit of travel. The timing of the valves is graphically represented by means of polar diagrams. In this diagrams the various phases (intake, combustion, expansion and exhaust stroke) have been represented by means of circle arcs, that are subtended by centre angles correspondently to the angles described by the crank. In Figure 1.7 it is shown typical timing valve diagram of a four strokes spark ignition engine.

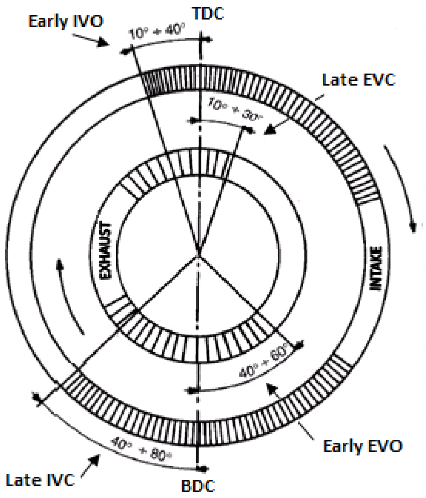

Figure 1.7 Polar diagram for a four spark ignition engine

Analyzing the polar diagram (Figure 1.7), the actual timing of valves can be divided into:

- Early opening of the exhaust valve: exhaust valve opening (EVO) occurs 40° ÷ 60° before BDC, and then before the end of expansion stroke, so that blow down can assist in expelling the exhaust gases. The goal is to reduce cylinder pressure to close at the value of the exhaust manifold pressure as soon as possible after BDC over the full engine speed range. As the piston is close to BDC, most of the push on the piston has ended and nothing is lost by opening the exhaust valve towards the end of the power stroke. This gives the exhaust gases additional time to start leaving the cylinder so that exhaust is then started when the piston passes BDC and start up on the exhaust stroke. The timing at which the exhaust valve is opened is critical. If the valve opens too early, more than necessary work is lost in the last phase of the power stroke. If it opens late, there is still excess pressure in the cylinder at BDC. This pressure resists the piston movement early in the exhaust stroke and increases the negative pumping work of the engine cycle. The ideal time to open the exhaust valve depends on engine speed. Note that the timing of exhaust valve opening affects the cycle efficiency since it determines the effective expansion ratio.
- Late exhaust valve closing: the exhaust valve closing (EVC) ends the exhaust process and determines the duration of the valve overlap period. The exhaust valve remains open for some degrees of crankshaft rotation after that the piston passes TDC and intake stroke has started so that the exhaust gases have some additional time to leave the cylinder. EVC typically falls in the range 8° ÷ 20° after TDC. At idle

and light load, in spark-ignition engines, it regulates the quantity of exhaust gases that flow back into the combustion chamber through the exhaust valve under the influence of intake manifold vacuum. At high engine speeds and loads, it regulates how much of the cylinder burned gases are exhausted. EVC timing should occur sufficiently far after TDC so that the cylinder pressure does not rise near the end of the exhaust stroke. Late EVC favors high power at the expense of low-speed torque and idle combustion quality.

- Early inlet valve opening: the inlet valve opens several degrees of crankshaft-rotation (10° ÷ 40°) before TDC on the exhaust stroke. That is the intake valve begins to open before the exhaust stroke is finished. This gives the valve enough time to reach the fully open position before the intake stroke begin. Then, when the intake stroke starts, the inlet valve is already wide open and air fuel mixture can start to enter the cylinder, immediately. Engine performance is relatively insensitive to this timing point. It should occur sufficiently before TDC so that cylinder pressure does not dip early in the intake stroke.
- Late intake valve closing: the intake valve remains open for few degrees of crankshaft rotation (40° ÷ 80°) after BDC, to provide more time for cylinder filling under conditions where cylinder pressure is below the intake manifold pressure at BDC. When the piston reaches BDC, it starts back towards TDC and in so doing starts to compress the air in the cylinder. Until the air is compressed to a pressure equal to the pressure in the intake manifold, air continues to enter the cylinder. The ideal time for the intake valve to close is when this pressure equalization occurs between the air inside the cylinder and the air in the manifold. If it closes before this point, air that was still entering the cylinder is stopped and a loss of volumetric efficiency is experienced. If the valve is closed after this point, air being compressed by the piston will force some air back out of the cylinder, again with a loss in volumetric efficiency. This delay of intake valve closing (IVC) is mainly used to make maximum use of the kinetic energy of the aspirated mixture that, thanks to the speed gained, continues to flow into the cylinder when the piston begins the compression phase. Of course, at higher engine speed, there is an increase of the kinetic energy and, therefore, late intake valve closing has to be greater. The position where the intake valve closes on most engines is controlled by camshaft and cannot change with engine speed. Therefore, it follows that high values of late IVC improve the filling of cylinder at high speed while they generate ebbs of the fresh charge from cylinder to the inlet at low speed (rejection phenomenon). The diagram in Figure 1.8 shows the values of filling coefficient λ_V, which indicates how much charge is effectively trapped in the cylinder compare to that theoretically possible, depending on engine speed for three values of late IVC.

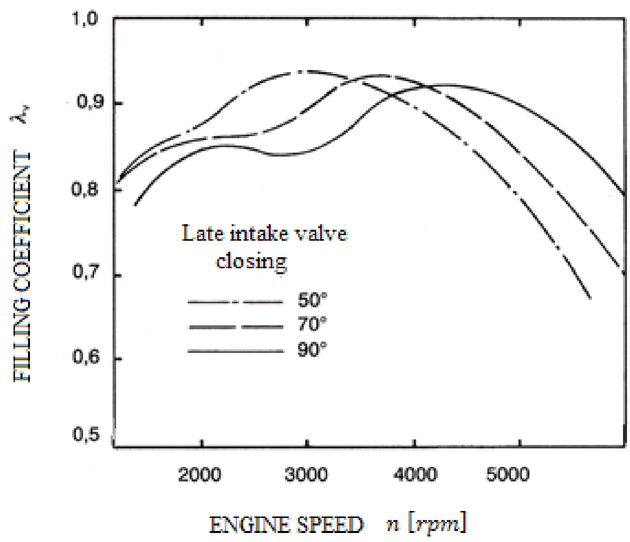

Figure 1.8 Filling coefficient

In other words, the filling coefficient is defined as the ratio between the air mass flow rate inducted by the engine and the air mass inducted the displacement in the reference conditions .In the engines naturally aspired the filling coefficient is generally lower than unit (0.75 ÷ 0.9) for the following considerations:

- there are load losses in the induction system (air filter, throttle body, manifolds, valves) and losses due to the necessary acceleration of the air mass inducted;
- the residual gas have a pressure higher than the same parameter in the initial conditions, and for this reason they create an opposition to the entrance of the fresh mixture as long as theirs pressure is not descends sufficiently thanks to the descendant motion of the piston. For this reason, a portion of the inlet stroke is not possible to use to increase the filling of the cylinder;
- the fresh mixture is heating in the passage through the induction manifold and for contact with the hot walls of the cylinder, with a consequent decrease of the density.

The filling coefficient influences directly the power supplied from an engine but indirectly also the indicate and mechanical efficiency. In fact, the high load losses in the intake manifold realize an important reduction of the filling coefficient, but produce also an increase of the pumping work and consequently a reduction of the indicated efficiency; moreover, if the filling coefficient is low, is low also the mean indicated pressure with negative consequences on the mechanical efficiency [4].

The opening and closing of the intake and exhaust valve affect significantly the correct working of an engine. In fact, the exact choice of the advance of the intake valve opening influences the filling of the cylinder, improving significantly the volumetric efficiency, but it may also have a significant effect on consumption (as well as on the harmful emission of unburned products). Indeed, if the exhaust valve closing occurs on excessive delay, a part of fresh charge, that enters the cylinder through the intake valve, will inevitably be dragged with the unburned gas, going irretrievably lost. Similarly, the excessive advance of inlet valve opening will be harmful too: unburned gases, in addition to flow towards the exhaust, tend to invade the inlet, worsening engine performance significantly.

1.4. TIMING SYSTEM

The timing system represents the whole of openings and closings of intake and exhaust valves, that allow the execution of any engine phase. In other words, the timing system has to regulate the gas exchange processes (intake and exhaust in four- stroke cycle engine) inside the cylinders. The purpose of the exhaust and inlet processes is to remove the burned gases at the end of the power stroke and admit the fresh charge for the next cycle.

Figure 1.9 shows the main parts that make up the timing system. The camshaft, regulates the opening of the valves (4) by the cams (1), that operate the tappets (3). The tappets transmit to the valves the push received by the cams. The purpose of the valves, in internal combustion engine, is to regulate the flow of fresh gas and the reflux of exhaust gases into the cylinder during the different engine phases. For these reasons, there are two valves types: intake and exhaust valve. Therefore, an appropriate air-fuel mixture is supplied from a carburetor through an intake manifold to an intake valve while the combustion gas is discharged through an exhaust valve into an exhaust manifold. The valves return to their closed position by the springs.

The valve type normally used in four-stroke engine is poppet valve. It is made from forged alloy steel which has the necessary high resistance to heat. A poppet valve consists of an head with a stem (cylindrical shape) is the part that moves in a valve guide. The head can be of different diameters to improve combustion. In fact, the head of the exhaust port has a reduced diameter to limit the transmission of heat from burned gases to the valve while that of inlet valve is larger to facilitate the filling of the cylinder. Furthermore, the mass of the valves has to be reduced in order to reach high speeds without excessive stress on the springs and on the drive parts.

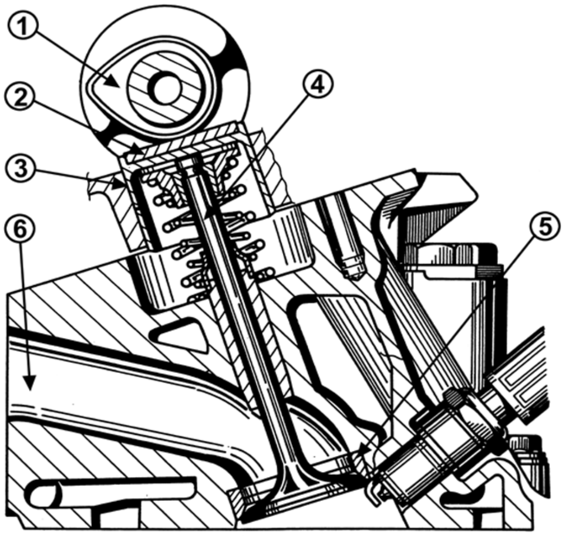

Figure 1.9 Parts of timing system

In four-stroke engines, the complete cycle of opening and closure of intake and exhaust valves occurs in one revolution of the camshaft while the piston strokes (intake, compression, expansion and exhaust stroke) happen in two revolutions of the crankshaft. Therefore, so that the cycle of camshaft is in phasing with that of crankshaft, the camshaft must rotate at a halved frequency compared to that of the crankshaft. The camshaft receives its motion from the crankshaft, at which is connected by means of a belt or chain (timing chain).The type of connection to the crankshaft depends on the position of the camshaft in the engine and on the performance of the engine.

In some applications, which require speed of the crankshaft and camshaft higher than a commercial engine, are used gearing. The use of the chain as a means to connect the crankshaft to the camshaft has a drawback: with the passage of time it tends to lengthen, because of thermal expansion of the head and base. To compensate this chain elongation and the wear of the sprocket, the automatic hydraulic tensioners are employed, that combining the oil pressure with the push of a spring, maintain the optimal level of tension of the chain. The chain runs in an guard and, lubricated by engine oil, can be equipped with sliding block or spring tightener that serve to reduce in some extent the relaxation for wear. Where the engines are very powerful, duplex chains are used. Figure 1.10 shows an application with connection between crankshaft and camshaft by chain. Chain systems are also used for the transmission of the motion of other engine parts, such as oil pumps, balancing countershaft and injection pumps.

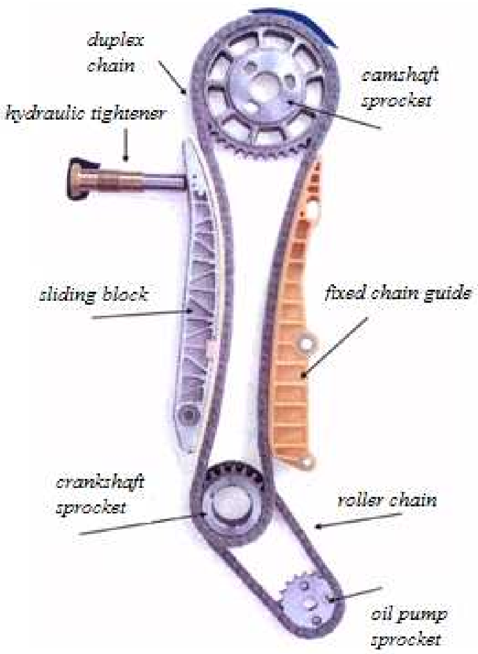

Figure 1.10 Connection between crankshaft and camshaft by chain

Another system is the rubber toothed belt. It is very practical because it is placed externally to the engine, and it does not consume the gears that have slots instead of teeth and is absolutely quieter. Figure 1.11 shows a typical use of the chain in timing system. The chain connects the two sprockets at the back of the camshaft, while the connecting with the engine is through a single pulley, on the front, and moved by a rubber belt. A hydraulic actuator, controlled by oil pressure that is intercepted and modulated by electromagnetic valves, acts on the two branches of the chain, causing a controlled mismatching between the intake and exhaust axis, that is proportional to the length change induced in the branches.

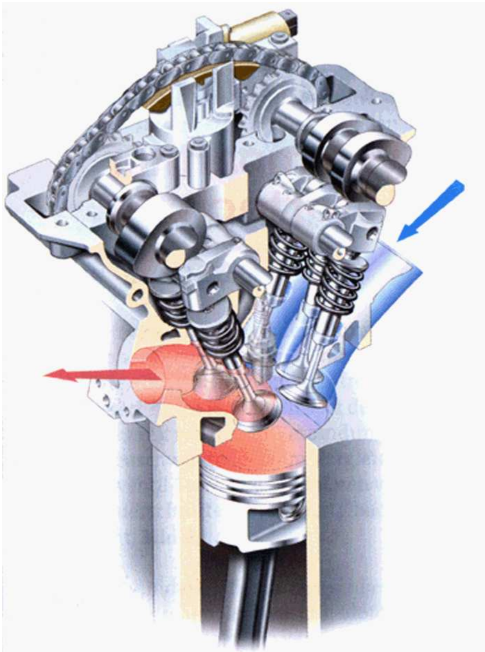

Figure 1.11 Timing system by chain

1.5. COMBUSTION IN SPARK- IGNITION ENGINE

In a conventional spark ignition engine the fuel and air are mixed together in the intake system, inducted through the intake valve into cylinder, where mixing with residual gas takes place, and then compressed. Under normal operating conditions, combustion is initiated towards the end of the compression stroke, near to TDC, at the spark plug by an electric discharge. Following inflammation, a turbulent flame develops, propagates through this essentially premixed fuel, air burned gas mixture, until it reaches the combustion chamber walls, and then extinguishes. In other words, the combustion propagates outwards from the spark plug location gradually thanks to the progress of the flame front that separates the areas, where ignition has started up and where this phenomenon has not yet begun. Figure 1.12 shows the pressure variation during combustion as a function of crank angle. The point (5) in the diagram indicates the position of the crank angle at which an electrical discharge across the electrodes of a spark plug starts the combustion process. This position occurs before TDC in order to enable the development of combustion of the mixture, which takes some time to complete. This diagram shows how the pressure after the ignition, does not rise immediately in the combustion chamber, but it remains for some time (of course very short) at a approximately constant value, as if the fuel needed a period of induction before burning. The delay, with which the increase of pressure shows itself, is due to the fact that initially only a small amount of mixture is involved in the phenomenon. Only when the combustion has affected 20-30% of the volume available to the combustion chamber, appreciable increase in pressure occurs. When the maximum pressure (7) of the cycle has been reached, the final phase of combustion starts during which the mixture terminates to burn with a progressive decrease of pressure due to expansion in progress [2].

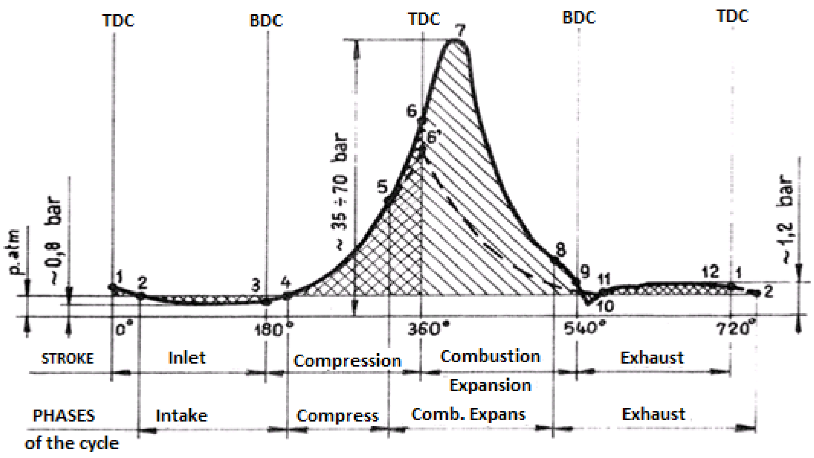

Figure 1.12 Pressure variation

The combustion event must be properly located with respect to top-dead-center in order to obtain the maximum power or torque. Combustion starts before the end of the compression stroke, continues through the early part of the expansion stroke, and ends after the point in the cycle at which the peak cylinder pressure occurs. The pressure versus crank angle curves (shown in Figure 1.13*a*) allow to understand why the engine torque (at given engine speed and intake manifold conditions) varies in function of spark ignition angle (with respect to TDC). If the start of combustion process is progressively advanced, the compression stroke work increases. If the end of the combustion process is progressively delayed by retarding the spark ignition, the peak cylinder pressure occurs later in the expansion stroke and it is reduced in magnitude. These changes reduce the expansion stroke work transfer from the cylinder gases to the piston. The optimum timing which gives the maximum brake torque, called maximum brake torque (MBT) timing, occurs when the magnitudes of these opposing trends just offset each other. Timing which is advanced or retarded from this optimal value gives lower torque. The optimal spark setting will depend on the rate of flame development and propagation, the length of the flame travel path across the combustion chamber, and the details of the flame termination process after it reaches the wall. These parameters depend on design engine design and operating conditions, and on the properties of the fuel, air, burned gas mixture. Figure 1.13*a* shows cylinder pressure versus crank angle for over advanced spark timing (50°), MTB timing (30°), and retarded timing (10°).

Figure 1.13*b* shows the effect of the variations in spark timing on brake torque for a typical spark ignition engine.

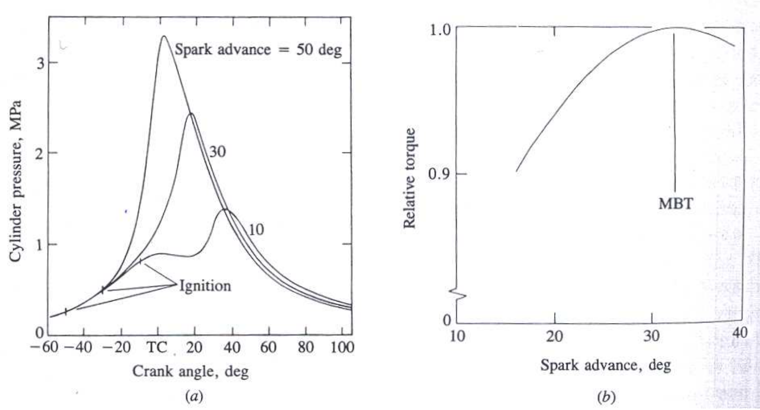

Figure 1.13 Cylinder pressure versus crank angle for variations of spark timing. Effect of spark advance on brake torque

Therefore, the quality of combustion is influenced by the ignition timing, that is the angle of rotation of the crank as to TDC: a poor ignition produces gases only partially burned and combustion is completed in the catalyst with the possibility to damage it, because of the excess heat generated locally. The spark timing is a key parameter for engine management, and it affects performance, fuel economy and emissions. The optimal timing varies according to engine design, its capacity, project objectives and, above all, the engine working point; the timing should therefore be defined to meet the maximum power supply, the minimum specific consumptions, the minimum emissions of unburned gas, and to prevent abnormal combustion phenomena.

A normal combustion is a burning process which is initiated solely by a timed spark and in which the flame front moves completely across the combustion chamber in a uniform manner at a normal velocity. In this case, the oxidation reaction of gas begins in the volume of mixture included between the spark plug electrodes. The discharge between the electrodes initiates the combustion, which then propagates to the rest of the charge. The flame front proceeds with regular speed up to die out to the walls when the whole mixture is burned. However, several factors, as fuel composition, certain engine design and operating parameters, and combustion chamber deposits, may prevent this normal combustion process from occurring. Abnormal combustion is a burning process in which a flame front may be started by hot combustion chamber surfaces either prior to or after spark ignition, or a process in which some part or all of the charge may be consumed at extremely high rates. It reveals itself in many ways. Of the various abnormal combustion processes which are important in practice, the three major phenomena are misfire, knock and surface ignition.

The *misfire* is a faulty ignition. For lean mixtures combustion is always slow and unstable. If the mixture is further impoverished, because of the lack of homogeneity of the fuel in the cylinder, that can happen is that part of it does not burn or that the flame front dies out before to reach the walls. A further impoverishment of the mixture may result, in volume between the spark plug electrodes, a charge so lean to prevent the onset of the oxidation reaction. This is a misfire. The production of torque and power is nothing, and the increase of pollutants expelled into the atmosphere is particularly significant. The same phenomena can be generated if the mixture instead of air is excessively diluted by exhaust gas recycle, EGR, typically used to reduce nitrogen oxides emissions.

Another important abnormal combustion phenomenon is *surface ignition*. It is ignition of the fuel- air charge by overheated valves or spark plugs, by glowing combustion chamber deposits, or by any other hot spot in the engine combustion chamber: it is ignition by any source other than normal spark ignition. In other word, surface ignition is the ignition of the fuel-air mixture by any hot surface, other than the spark discharge, prior to arrival of the flame. In this case, the mixture autoignition occurs and a flame front is generated and is exactly the same of that created by spark plug but it is not controllable either at the time or at the place. It may occur before the spark plug ignites the charge (preignition) or after normal ignition (postignition). It may produce a single flame or many flames. Uncontrolled combustion is most evident and its effects most severe when it results from preignition because it can produce a low torque and, above all, very high temperatures and pressures which can favour, in the following cycles, the generation of knock. However, even when surface ignition occurs after the spark plug fires (postignition), the spark discharge no longer has complete control of the combustion process. Surface ignition is a problem that can be solved by appropriate attention to design engine, and fuel and lubricant quality.

Knock is the most important abnormal combustion phenomenon. Its name comes from the noise which is transmitted through the engine structure when essentially spontaneous ignition of a portion of the end gas (the fuel, air, residual gas, mixture ahead of the propagating flame) occurs. As the flame propagates across the combustion chamber, the end-gas is compressed, causing its pressure, temperature, and density to increase. Therefore, there is an extremely rapid release of most of the chemical energy in the end-gas, causing very high local pressures and the propagation of pressure waves of substantial amplitude across the combustion chamber. The fundamental theories of knock are based on models for the autoignition of the fuel air mixture in the end- gas. Autoignition is the term used for a rapid combustion reaction which is not initiated by any external ignition source. The autoignition of a gaseous fuel- air mixture occurs when the energy released by the reaction as heat is larger than the heat lost to the surroundings; as a result the temperature of the mixture increases, thereby rapidly accelerating, due to their exponential temperature dependence, the rates of the reactions involved. The state at which such spontaneous ignition occurs is called the *self ignition temperature* and the resulting self accelerating event where the pressure and temperature increase rapidly is termed a *thermal explosion*. Knock primarily occurs under wide-open-throttle operating conditions. It is a direct constraint on engine performance. It also constraints engine efficiency, since by effectively limiting the temperature and pressure of the end-gas, it limits the engine compression ratio. The occurrence and severity of knock depend on the knock resistance of the fuel and on the antiknock characteristics of the engine. The ability of a fuel to resist knock is measured by its octane number: higher octane numbers indicate greater resistance to knock. The knock is influenced from the spark advanced: advancing the spark increases the knock severity or intensity and retarding the spark decreases the knock. Therefore, the knock imposes limits to utilizable spark timing. The pressure variation in the cylinder during knocking combustion indicates in more detail what actually occurs. Figure 1.14 shows the cylinder pressure variation in three individual engine cycles, for normal combustion, light knock and heavy knock respectively. When knock occurs, high-frequency pressure fluctuations are observed whose amplitude decays with time [1].

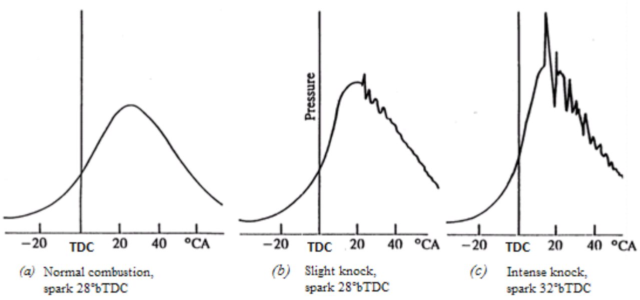

Figure 1.14 Cylinder pressure versus crank angle traces of cycles with normal combustion, light knock, and heavy knock

Chapter 2 New technologies

The recent trend in engine design is the use of low displacement engines (so called “downsizing”) to fulfill the new requirements. Torque and power increase is obviously in contrast with the fuel consumption reduction, so new technologies need to be used in order to induce different engine working conditions. On gasoline engine the industrial research is focused on the development of a new way to let the air into the engine: Variable Valve Timing, or Variable Valve Actuation system (and also the use of Turbo Charger) have the objective to optimize the air inlet, in order to maximize the engine efficiency. In this scenario a revolutionary approach has been introduced by Fiat by means of MultiAir technology, which is detailed in this chapter, that permits to define the desired inlet valve timing.

2.1. VARIABLE VALVE TIMING SYSTEM

In traditional internal combustion engines (ICE), gas exchange valve timing is mechanically fixed with respect to crankshaft position. This timing determines when the valves open and close, thereby affecting the air-fuel mixture and exhaust flow.

Variable Valve Timing (VVT) is a generic term for various concepts that allow changing the advance, overlap, and even (in the case of some import overhead-cam engines) the duration and lift of a four-stroke internal-combustion engine's intake and exhaust valves while the engine is operating. This technology has been under development for more than a century (a variation was tried out on some early steam engines), but it is only within the last twenty years or so with the advent of sophisticated electronic sensors and engine management systems that it has become practical and effective. There are a number of different variable valve timing systems, currently available and under development, to control different valve timing parameters. These systems can be grouped in terms of their operation and they are:

- Phase Changing Systems
- Profile Switching Systems
- Cam Changing and Cam Phasing Systems
- Valve Duration Systems

2.1.1. Phase Changing Systems

Examples of cam-phasing VVT are:

- Toyota’s VVT-i (Variable Valve Timing with Intelligence), which intelligently adjusts the overlap time between the exhaust valve closing and the intake valve opening;

Lexus’s VVT-iE (VVT – intelligent by Electric motor), which consists of the cam phase converter, mounted on the intake camshaft and converts the motor rotational input into the advance and retard of the cam phase, and a brushless electric motor, installed in the engine chain case.

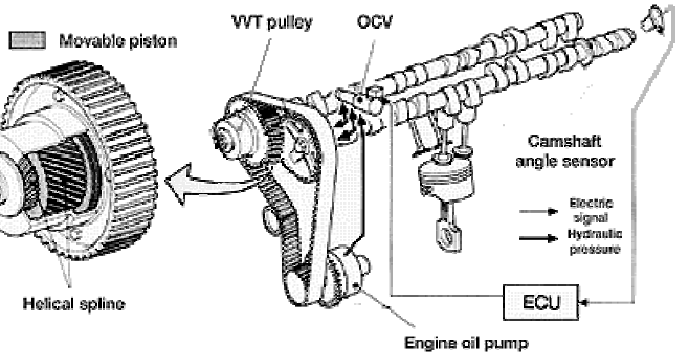

Figure 2.1 Cam- phasing VVT

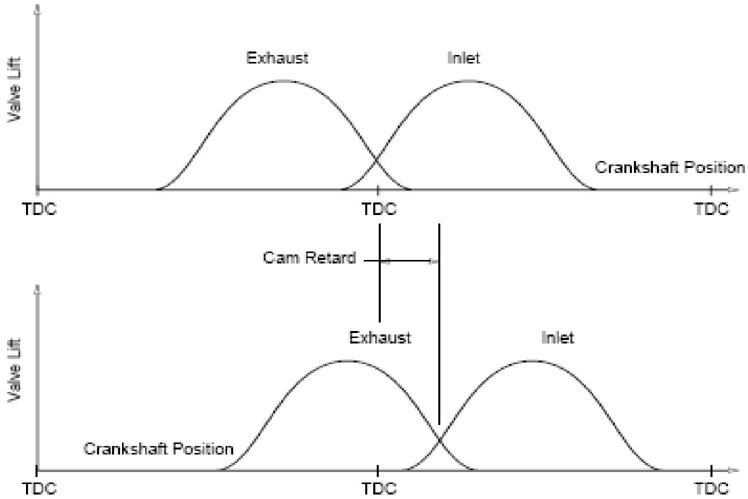

Figure 2.2 Valve Lift for an Engine with Cam Phasing VVT

Phase changing systems have been available on production engines for a number of years but have tended to be applied only to the highest specification engine in a particular range. Phasing of the intake camshaft to gain increased performance with a mechanism that can be moved between two fixed camshaft timings is the most common application with the change in timing normally occurring at a particular engine speed. More recently, there has been a move towards more flexible control systems that allow the camshaft phasing to be maintained at any point between two fixed limits. This has facilitated camshaft phase optimization for different engine speed and load conditions and has allowed exhaust camshaft phasing to be used for internal EGR control.

2.1.2. Profile Switching Systems

This type of variable valve timing system is capable of independently changing valve event timing and valve peak lift. The system switches between two different camshaft profiles on either or both of the camshafts and is normally designed to change at a particular engine speed (Figure 2.3). One profile designed to operate the valves at low engine speeds provides good road manners, low fuel consumption and low emissions output. The second profile is comparable to the profile of a race cam and comes into operation at high engine speeds to provide a large increase in power output. Therefore, cam- changing VVT system acts as if two different cam types at high and low speed are used.

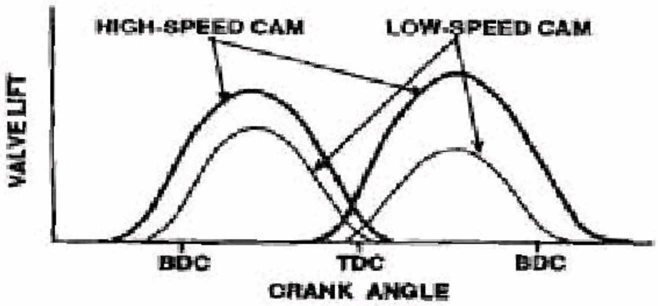

Figure 2.3 Valve Lift for an Engine with Cam Changing VVT

Honda, with VTEC (Variable Valve Timing and Electronic Lift Control) system started production of a system that gives an engine the ability to operate on two completely different cam profiles, eliminating a major compromise in engine design. In its simplest form, VTEC allows the valves to remain open for two different durations: a short opening time for low-speed operation to give good torque and acceleration, and a larger opening time for higher speeds to give more power. To do this, the camshaft has two sets of cam lobes for each valve and a sliding locking pin on the cam follower that determines which lobe is operating the valve. The locking pin is moved by a hydraulic control valve based on the engine speed and power delivery requirements. The two lobe shapes are referred to as fuel economy cams and high power cams, meaning that Honda engines with this technology are really two engines in one - a performance engine and an economical engine.

Due to these systems having an inherently two position operation, they are not suitable for optimizing valve timing parameters under different load conditions, e.g. EGR control. The ability to change valve event timing, lift and duration ensures that these systems are capable of providing very high power output from a given engine whilst still complying with emissions legislation.

2.1.3. Cam Changing and Cam Phasing Systems

Combining cam-changing VVT and cam-phasing VVT could satisfy the requirement of both top-end power and flexibility throughout the whole rev range, but it is inevitably more complex.

A typical example of this system is Toyota's VVTL-i (Figure 2.4). The system can be seen as a combination of the existing VVT-i and Honda's VTEC, although the mechanism for the variable lift is different from Honda. Like VVT-i, the variable valve timing is implemented by shifting the phase angle of the whole camshaft forward or in reverse by means of a hydraulic actuator attached to the end of the camshaft. Like VTEC, Toyota's system uses a single rocker arm follower to actuate both intake valves (or exhaust valves). It also has two cam lobes acting on that rocker arm follower, the lobes have a different profile - one with longer valve-opening duration profile (for high speed), another with shorter valve-opening duration profile (for low speed). At low speed, the slow cam actuates the rocker arm follower via a roller bearing (to reduce friction). The high speed cam does not have any effect on the rocker follower because there is sufficient spacing underneath its hydraulic tappet. At low speed long duration cam idles while when speed has increased to the threshold point, the sliding pin is pushed by hydraulic pressure to fill the spacing. The high speed cam becomes effective. Note that the fast cam provides a longer valve-opening duration while the sliding pin adds valve lift (while for Honda VTEC, both the duration and lift are implemented by the cam lobes). VVTL-i offers variable lift, which lifts its high speed power output a lot. Compared to Honda VTEC, Toyota's system has continuously variable valve timing which helps it far better to achieve low to medium speed flexibility. Therefore it is undoubtedly the best VVT today. However, it is also more complex and probably more expensive to build.

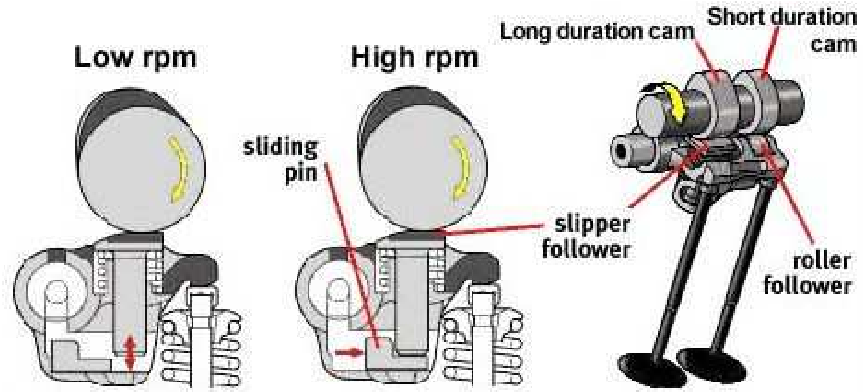

Figure 2.4 Toyota's VVTL-i system

2.1.4. Valve Duration Systems

The basic concept of this system is that of lengthening the duration of the opening of the valves. This system was introduced by Rover who call it Variable Valve Control (VVC). The VVC principle is based on an eccentric rotating disc to drive the inlet valves of every two cylinders. Since eccentric shape creates nonlinear rotation, the opening period of the valves can be varied by controlling the eccentric position of the disc. With VVC the outlet camshaft is not part of the VVC system and is driven normally by the toothed belt from the crankshaft. Figure 2.5 shows an example of VVC system working. When the eccentric wheel, which is connected to crankshaft with revolution speed halved, rotates at 180°, the camshaft turns, for example, only 140°. In the following 180° of eccentric wheel, the camshaft turns instead 220° so that it remains totally in phase with the crankshaft. This variable camshaft speed can change the duration of inlet valves opening. Figure 2.6 shows the valve lift for an engine with VCC Rover.

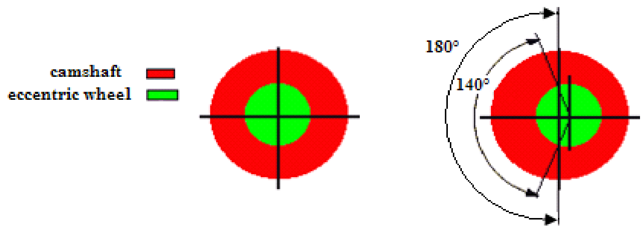

Figure 2.5 VVC system working

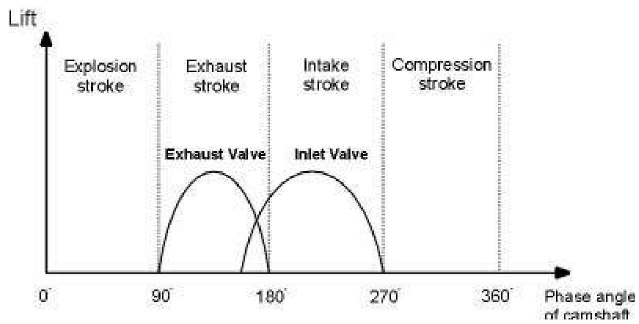

Figure 2.6 Valve Lift for engine with VVC system

2.2. VARIABLE VALVE ACTUATION TECHNOLOGY

The Variable Valve Actuation technology represents the new timing system with variable lift of the intake valves that has been developed in recent years. The VVA has been introduced as a promising technology able to improve the performance of the vehicle in terms of fuel economy, emission reductions and, more generally, the whole efficiency of the system. Contrary to the classical engine, where the intake and exhaust valves are commanded mechanically by the camshaft and so both the timing and the duration of valves opening are fixed by events, the VVA system offers the possibility to vary the valves actuation. Moreover, the innovative VVA system allows to control the air mass flow rate that is inducted in an internal combustion engine without any use of the throttle plate; this last situation has the benefit of having an air pressure upstream of intake valves always constant because the pump losses near the throttle body are zero.

The adopted VVA system is shown schematically in Figure 2.7. The valve actuator consists of a piston connected through an oil chamber to the intake valve, a solenoid valve to regulate the pressure inside the oil chamber and a hydraulic brake to assure the soft landing. When the electro-valve is open, the oil comes out from the high pressure chamber and it can be consequently possible to obtain any condition included between the two following extreme modalities:

- if the solenoid valve is open the intake valves remain closed;
- if the solenoid valve is always closed, the lift of the valves is the same as that of the cams.

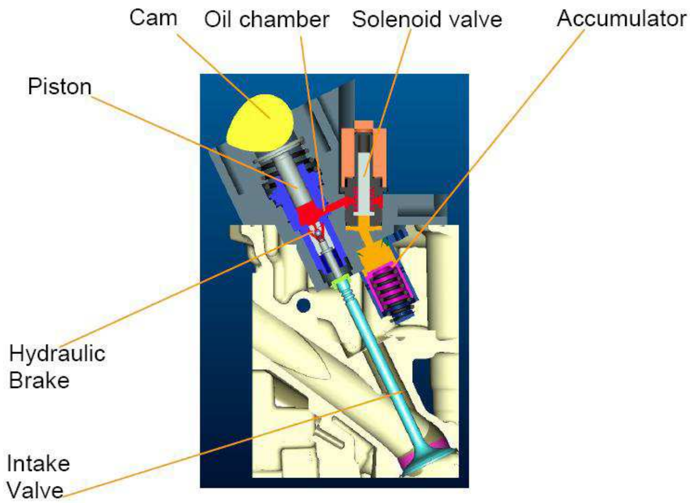

Figure 2.7 VVA system

As reported in Figure 2.8, the valve lift profile can assume different forms depending on the required air mass flow rate and engine speed. In fact, with this particular VVA system many strategies are possible, as is indicated in the following points:

- Full Lift (FL) actuation mode represents the normal functioning of the valves, i.e. commanded mechanically by the camshaft: the solenoid valve remains closed assuring high pressure into the oil chamber and, consequently, assuring a rigid connection between the intake valve and the camshaft through the piston;
- Early Closure (EC) valve mode is obtained by opening the solenoid valve at a certain cam angle, i.e. the control angle, reducing the pressure inside the oil chamber. The motion of the intake valve is then decoupled from the piston and, forced by the valve springs, it starts to

close earlier than in the full-lift mode. Soft landing of the intake valve is controlled by a hydraulic dampening unit (hydraulic brake);

- Late Opening (LO) valve mode can be achieved by regulating the solenoid valve partially opened. In this way, the pressure inside the oil chamber is regulated to a lower pressure than in the full lift mode, obtaining a rigid connection, but with a shorter distance function of the chamber pressure, between the intake valve and the camshaft. Consequently, the valve profile is similar to the full lift mode, but with a smaller time duration;
- Multi Lift (ML)actuation mode is a particular operative actuation mode obtained combining the late opening with the early closure, as is shown in Figure 2.9 This profile is limited by the mechanical cam constrains, in fact the next late opening must be activate before 50% of the full lift cam.

The flexibility of intake valve control offered by the VVA system leads to enhancing the efficiency of the combustion process. More in general, the following advantages can be addressed to the introduction in the vehicle of the VVA system:

- high charge trapping efficiency over the entire speed range through a wide modulation of valve lift;
- throttle-less engine operation, through direct air control at the valves resulting in a reduction of pumping work and fuel consumption;
- dynamic control, cylinder by cylinder and stroke by stroke, of the inlet charge aimed at an improvement of emissions, drivability and fuel consumption in transient operation.

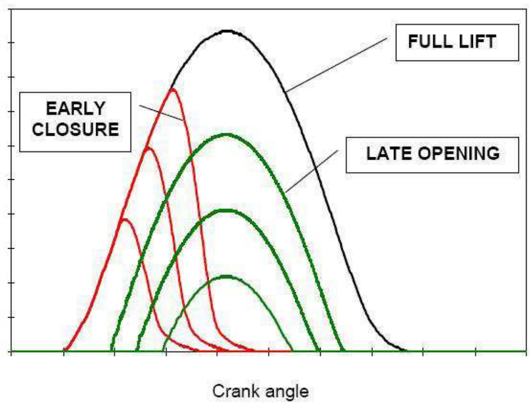

Figure 2.8 Valve lift profiles

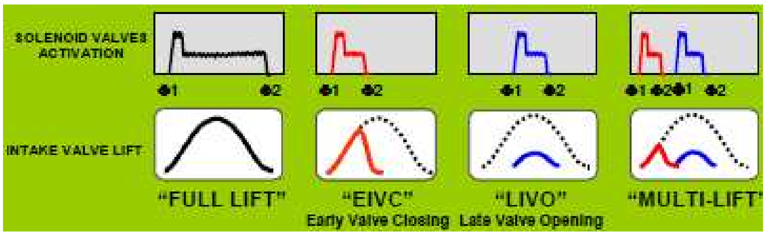

Figure 2.9 Solenoid valves activation and valve lift

The electro-hydraulic actuator that has been developed, moreover, is relatively simple but with high hardiness characteristics and low sensitivity to the critical parameters like a variation of the oil viscosity related to temperatures.

Finally, the control of this innovative VVA system is achieved by a specific electronic control system, that contains model-based strategies, that allows to elaborate the actuation signals of the valves according to the demands of the driver [5].

The MultiAir technology developed by Fiat is an example of application of variable valve actuation system.

2.2.1. The MultiAir technology

In the last decade, the development of the Common Rail technology for Diesel engines marked a breakthrough in the passenger car market. To be competitive also in the field of gasoline engines, Fiat Group decided to follow the same approach and focus on breakthrough technologies. The aim was to provide customers with substantial benefits in terms of fuel economy and fun-to- drive while maintaining the engine intrinsic comfort characteristics, based on a smooth combustion process and on light structures and components.

The key parameter to control Diesel engine combustion and therefore performance, emissions and fuel consumption is the quantity and characteristics of the fuel injected into cylinders. That is the reason why the Common Rail electronic Diesel fuel injection system was such a fundamental breakthrough in Direct Injection Diesel engine technology. The key parameter to control gasoline engine combustion, and therefore performance, emissions and fuel consumption, is the quantity and characteristics of the fresh air charge in the cylinders. In conventional gasoline engines the air mass trapped in the cylinders is controlled by keeping the intake valves opening constantly and adjusting upstream pressure through a throttle valve. One of the drawbacks of this simple conventional mechanical control is that the engine wastes about 10% of the input energy in pumping the air charge from a lower intake pressure to the atmospheric exhaust pressure.

A fundamental breakthrough in air mass control, and therefore in gasoline engine technology, is based on direct air charge metering at the cylinder inlet ports by means of an advanced electronic actuation and control of the intake valves, while maintaining a constant natural upstream pressure.

Research on this key technology started in the eighties, when engine electronic control technologies reached the stage of mature technologies. At the beginning world-wide research efforts were focused on the electromagnetic actuation concept, following which valve opening and closing is obtained by alternatively energizing upper and lower magnets with an armature connected to the valve. This actuating principle had the intrinsic appeal of maximum flexibility and dynamic response in valve control, but despite a decade of significant development efforts the main drawbacks of the concept (its being intrinsically not fail-safe and its high energy absorption) could not be fully overcome. At this point most automotive companies fell back on the development of the simpler, robust and well-known electromechanical concepts, based on the valve lift variation through dedicated mechanisms, usually combined with cam phasers to allow control of both valve lift and phase. The main limitation of these systems is low flexibility in valve opening schedules and a much lower dynamic response; for example, all the cylinders of an engine bank are actuated simultaneously, thereby excluding any cylinder selective actions. Many similar electromechanical valve control systems were then introduced over the past decade.

In the mid 90's Fiat Group research efforts switched to electro-hydraulic actuation, leveraging on the know-how gained during the Common Rail development.

The goal was to reach the desired flexibility of valve opening schedule air mass control on a cylinder-by-cylinder and stroke-by-stroke basis. The electro-hydraulic variable valve actuation technology developed by Fiat was selected for its relative simplicity, low power requirements, intrinsic fail safe nature and low cost potential.

The operating principle of the system, applied to intake valves, is the following: a piston, moved by a mechanical intake cam, is connected to the intake valve through a hydraulic chamber, which is controlled by a normally open on/off solenoid valve.

When the solenoid valve is closed, the oil in the hydraulic chamber behaves like a solid body and transmits to the intake valves the lift schedule imposed by the mechanical intake cam. When the solenoid valve is open, the hydraulic chamber and the intake valves are de-coupled; the intake valves do not follow the intake cam anymore and close under the valve spring action. The final part of the valve closing stroke is controlled by a dedicated hydraulic brake, to ensure a soft and regular landing phase in any engine operating conditions.

Through solenoid valve opening and closing time control, a wide range of optimum intake valve opening schedules can be easily obtained.

For maximum power, the solenoid valve is always closed and full valve opening is achieved completely following the mechanical cam (**Full Lift** mode), which was specifically designed to maximize power at high engine speed (long closing time). For low-rpm torque, the solenoid valve is opened near the end of the cam profile, leading to early intake valve closing (**EIVC** mode). This eliminates unwanted backflow into the manifold and maximizes the air mass trapped in the cylinders. In engine part load, the solenoid valve is opened earlier (before finishing of the cam profile) causing partial valve openings to control the trapped air mass as a function of the required torque (**Partial Load** mode). Alternatively the intake valves can be partially opened by closing the solenoid valve once the mechanical cam action has already started (**LIVO** mode: Late Intake Valve Opening). In this case the air stream into the cylinder is faster and results in higher in-cylinder turbulence. The last two actuation modes can be combined in the same intake stroke, generating a so-called "**Multilift**" mode, which enhances turbulence and combustion rate at very low loads. Figure 2.10 shows possible valve profiles using the multi-air technology.

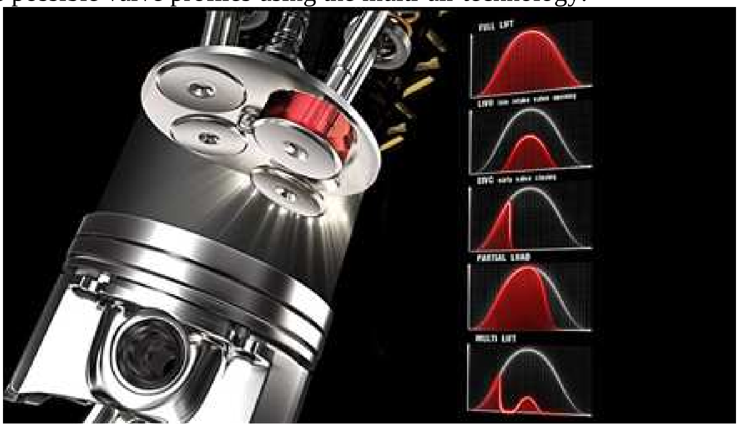

Figure 2.10 Possible valve profiles using the MultiAir technology

The Multiair Technology potential benefits for gasoline engines exploited so far can be summarized as follows:

- Maximum Power is increased by up to 10% thanks to the adoption of a power-oriented mechanical cam profile;
- Low-rpm Torque is improved by up to 15% through early intake valve closing strategies that maximize the air mass trapped in the cylinders;
- Elimination of pumping losses brings a 10% reduction of fuel consumption and CO_2 emissions, both in naturally aspirated and turbocharged engines with the same displacement;
- MultiAir turbocharged and downsized engines can achieve up to 25% fuel economy improvement over conventional naturally aspirated engines with the same level of performance;
- Optimum valve control strategies during engine warm-up and internal Exhaust Gas Recirculation, realized by reopening the intake valves during the exhaust stroke, result in emissions reduction ranging from 40% for HC/CO to 60%for NO_x;
- Constant upstream air pressure, atmospheric for naturally aspirated and higher for turbocharged engines, together with the extremely fast air mass control, cylinder-by-cylinder and stroke-by-stroke, result in a superior dynamic engine response.

The MultiAir technology, a Fiat worldwide premiere in 2009, has introduced further technological evolutions for gasoline engines:

- Integration of the MultiAir direct air mass control with direct gasoline injection to further improve transient response and fuel economy;
- Introduction of more advanced multiple valve opening strategies to further reduce emissions;
- Innovative engine-turbocharger matching to control trapped air mass through combination of optimum boost pressure and valve opening strategies.

While electronic gasoline fuel injection developed in the '70s and Common Rail developed in the '90s were fuel specific breakthrough technologies, the MultiAir Electronic Valve Control technology can be applied to all internal combustion engines whatever fuel they burn.

MultiAir, initially developed for spark ignition engines burning light fuel ranging from gasoline to natural gas and hydrogen, has wide potential also for Diesel engine emissions reduction. Intrinsic NO_x reduction of up to 60% can be obtained by internal Exhaust Gas Recirculation (*iEGR*) realized with intake valves reopening during the exhaust stroke, while optimal valve control strategies during cold start and warm-up bring up to 40% *HC* and *CO* reduction of emissions. Further substantial reduction comes from the more efficient management and regeneration of the diesel particulate filter and NO_x storage catalyst, thanks to the highly dynamic air mass flow control during transient engine operation.

Diesel engine performance improvement is similar to that of the gasoline engine and is based on the same physical principles. Instead, fuel consumption benefits are limited to few percentage points because of the low pumping losses of Diesel engines, one of the reasons of their superior fuel economy. In the future, power train technical evolution might benefit from a progressive unification of gasoline and Diesel engines architectures.

A MultiAir engine cylinder head can therefore be conceived and developed, where both combustion systems can be fully optimized without compromises. Moreover the MultiAir electro-hydraulic actuator is physically the same, with minor machining differences, while internal subcomponents are all carry over from the Fire and SGE (small gasoline engine) applications [6], [7].

2.3. INTELLIGENT ALTERNATOR CONTROL SYSTEM

An alternator is an electromechanical device that converts kinetic energy into electrical energy and is used in modern automobiles to charge the battery and to power a car's electric system when its engine is running. It is connected to the engine via a belt that transmits the motion of rotation that within the alternator is used to produce alternating electric current (AC). Automotive alternators use a set of diodes to convert AC to DC (direct current). A voltage regulator makes for the voltage output from the alternator to remain constant.

Intelligent Alternator Control System (IAC) allows some energy to recover, through "intelligent" use of the alternator, which during braking is dissipated as heat in brake discs. The IAC system generates electric power for a car's on board network exclusively in over-run and during braking to transform the kinetic energy resulting from the inertia of the vehicle into electrical energy that

is transferred to the battery becoming an energy surplus, which can return available in the following phases of acceleration when it is used to feed the electrical components of the vehicle, thus decreasing the work required at the alternator at this stage. This results in more torque being produced to the wheels. Normally, the engine control system determines the power that is used to activate the alternator --through a torque-based model.
The IAC system establishes that:

- in the acceleration phase, the alternator output voltage is set so that it is equal or close to the level of battery voltage, which, in this way, provides alone for the electricity needs of the car;
- in over-run or during breaking, the alternator output voltage is set to a value higher than the battery voltage, so the alternator is not only able to cover the entire electricity demand of the vehicle, but at the same time to charge the battery.

In the latter case, the alternator is actually a load on the crankshaft and thereby exerts a braking effect on the vehicle which represents precisely the recovery of part of the energy dissipated in the brakes. For the IAC system it’s necessary to have precise information about the charging status and battery usage which are acquired by using a sensor IBS (Intelligent Battery Sensor). The battery is charged to only about 80% of its capacity as long as the engine is propelling the car, depending on ambient conditions. A reserve charge that is adequate for the consumption of power while the car is at a standstill and enabling the driver to start at any time, is maintained under all circumstances. Battery charge exceeding the 80% threshold is generated only in over-run and while the driver is applying the brakes. Since the number of charge cycles increases as a function of such battery management, IAC uses modern AGM (absorbent glass mat) batteries able to handle greater loads than conventional lead acid batteries.

The IAC system improves the overall efficiency of a vehicle by decreasing ancillary loads on the engine and by recuperating more of the waste heat energy so as to reduce fuel consumption.

Chapter 3 Engine Management System

An Engine Management System (EMS) is an HW and SW system (microcontroller based board and all necessary sensors and actuators) that implements all engine control and management functions that are necessary to make the engine work properly. The importance of such a system has recently been increased, starting from the first requirements regarding the pollutant emission legislation, becoming a crucial factor for more and more ambitious and restrictive requirements, such as emissions (new €5+ and €6 legislation is coming), fuel consumption, fun to drive and also driver safety.

3.1. GENERAL DESCRIPTION

The engine management system consists of:

- Sensors that monitor and measure the operative conditions of the engine;
- Engine control unit (ECU) that, on the basis of the input received from sensors and from other systems in vehicles and of calculations and tables which realize the control strategies, generates control signals for the actuators;
- Actuators that, commanded by ECU, perform actions as a result of the values measured by the sensors.

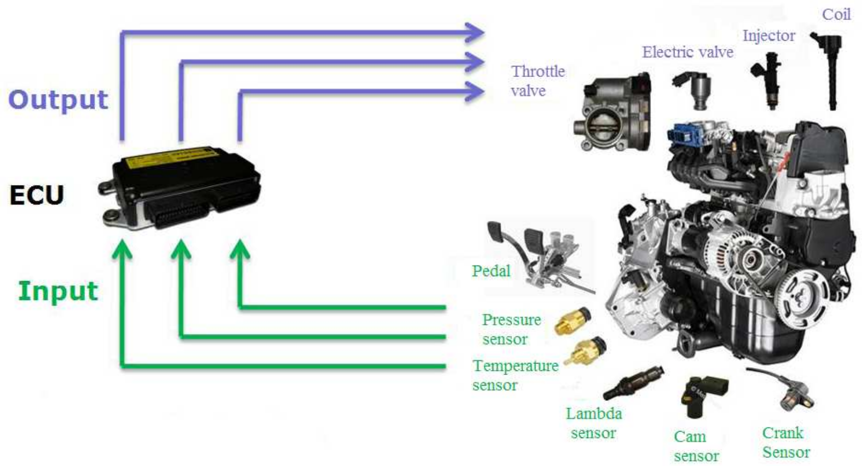

Figure 3.1 Engine management system

As shown in Figure 3.1, the control center of EMS is the engine control unit which detects all input signals and generates control signals for the actuators.

The main objectives in the planning of these control systems are:

- To maximize the performances, meeting the request of torque of the driver, providing an optimal drivability in all operative conditions and reducing fuel consumption;
- To respect the limits of exhaust gas emission. This objective requires an exact air- fuel ratio and a good compensation during transient engine operation;
- To perform a suitable diagnosis and recovery of system in case of bad functioning.

Figure 3.2 shows a scheme of a turbocharged spark ignition, indirect injection, and a variable timing engine management system. In this scheme, it is possible to observe the position of main sensors and actuators inside the engine.

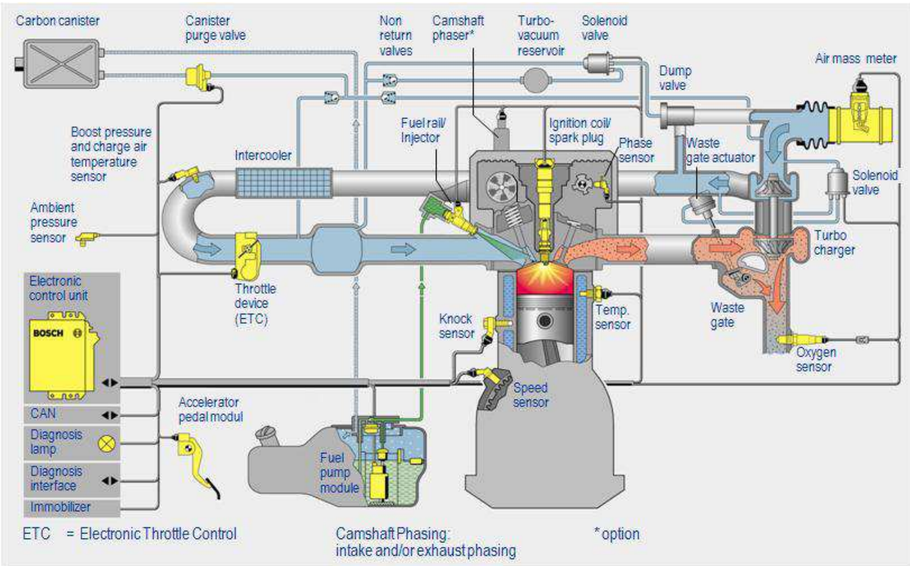

Figure 3.2 Components used for electronic control in an engine management system

3.2. ENGINE CONTROL FUNCTIONS

The heart of engine management systems are the strategies and controls which allow the exact engine working. The main four engine control cycles are:

1. Feed forward cycle of injected fuel;
2. Feedback cycle of air- fuel ratio;
3. Spark advance cycle;
4. Knock cycle;

Moreover, in the most ECUs , the following control loop is also presented:

- Title control
- Idle control
- Exhaust gas recirculation (EGR) control

- Purge canister control
- Turbocharger control
- Speed limiting device

3.2.1. Torque control

The torque of a spark ignition engine is controlled by the air- fuel quantity inside the cylinder. This quantity is modified changing the manifold pressure and, therefore, the density of the air- fuel mixture. Even though this solution is comparatively easy and efficacious it generates some pumping losses which influence the efficiency of the engine. From the point of view of control, the gasoline engine is controlled by the air provided and therefore, the torque regulation occurs, first of all, through the throttle valve and then by the injection time. Moreover, to have a rapid transient response, it is regulated by the spark advance. The SI engine is characterized by a powered throttle valve (drive by wire), i.e. the driver doesn't order it directly through the gas pedal, but the input is translated throughout a potentiometer in an electric signal and then sent to the ECU. This signal is interpreted as a torque request. This request is also influenced by some subsystems (ESP, cruise control, etc). For this reason a technique is used in the software structures called torque-based, which de-couples the strategic decisions from the functions of low level. The manager of torque collects all torque requests and considers all other information. Then only one signal is transmitted to the module that generates the suitable commands to the actuators so that the desired torque is generated. Figure 3.3 shows the block diagram in the ECU that manages the torque control. The torque-based method expects the calculation of three fundamental parameters:

- Throttle angle: it regulates the opening and closing of the throttle;
- Spark advance: it permits to regulate the charge time of the coils (dwell time);
- Injection time: it establishes the duration of the fuel quantity to inject.

The actuation of the desired torque also considers the air dynamic. This presents a time constant not negligible. Therefore, besides acting on the throttle valve, to have a rapid response, the spark advance is regulated. The torque generated in the first way is called *slow* torque while that obtained in the second way is called *fast* torque. Figure 3.4 shows a block scheme relevant to the actuation of the torque. The actuation of the slow torque, which is of a certain throttle angle, defines an engine operative point (manifold pressure), while that of the fast torque, which is of a certain spark advance, describes the combustion efficiency and has the scope to coincide the realized torque with the one desired [8].

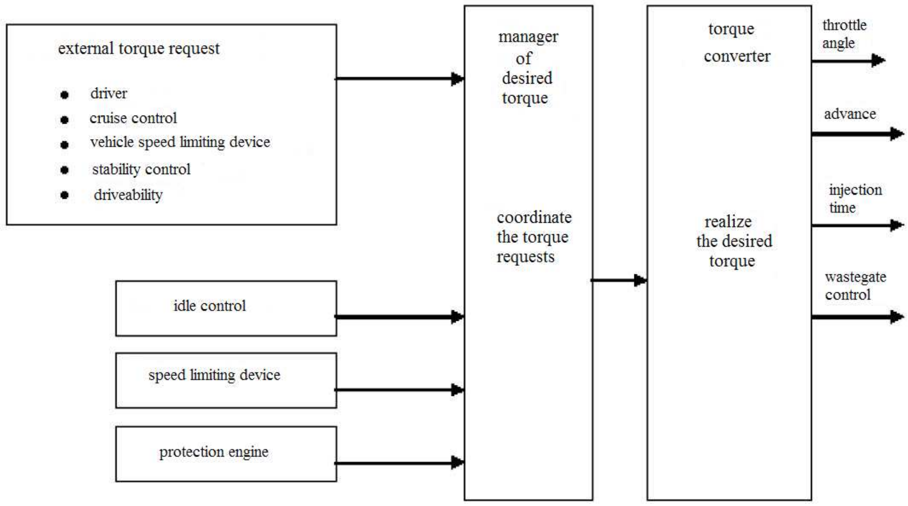

Figure 3.3 Torque Control

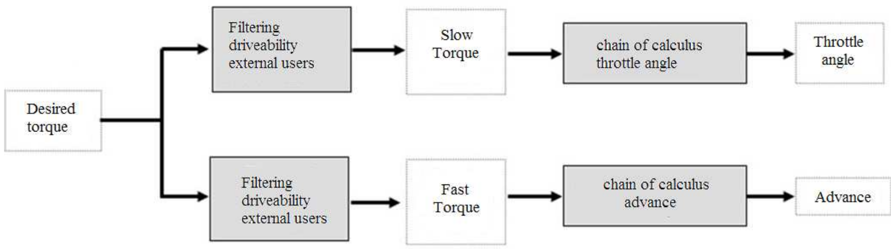

Figure 3.4 Actuation of slow and fast torque

3.2.2. Air- fuel ratio control

The air- fuel control is a very important control cycle, because the three way catalytic converter works in an efficient way only when the engine presents a stoichiometric ratio. This control system is composed of a slow but exact feedback action, founded on the lambda sensor, and from a fast but rough feed-forward action, as shown in Figure 3.5:

- Feed-forward control: Without a feed-forward action, the controller is too slow while in the transient condition it is very important to determine the exact fuel quantity in a rapid way. This is possible only considering the manifold dynamic. The information on throttle angle, manifold pressure, or air flow passing for the manifold, has to be used to predict the air mass flow rate entering the cylinder and on the basis of this information the fuel quantity to inject is determined. One also has to consider the well- wetting phenomenon that has a typical dynamic to be compensated.

- Feedback control: Even though the three way catalytic converter can operate with different air-fuel ratios, a small steady- state error is required. For this reason, the feed forward control has to be completed with a feedback control. A possible control can be performed on the air- fuel ratio through a switch-type sensor placed before the catalyser. Typically, the output of the sensor is converted in a binary signal, lean mixture for voltage lower than 450mV, rich mixture for voltage higher than 450mV. In this way, a control structure like a PI is realized (Figure 3.6): the correction of injection time is obtained by multiplying the injection time for the output of the controller. The advantage derives from multiplicative characteristic of air- fuel ratio. [8]

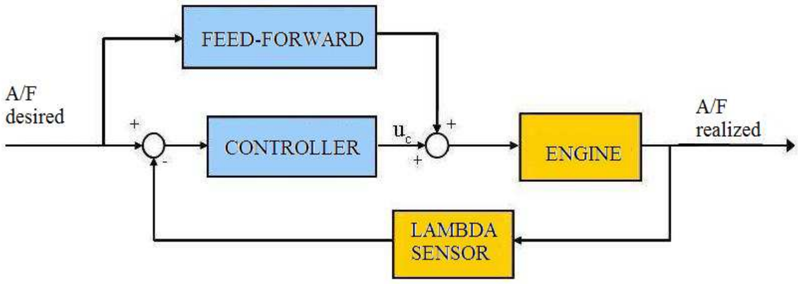

Figure 3.5 Control system of air-fuel ratio

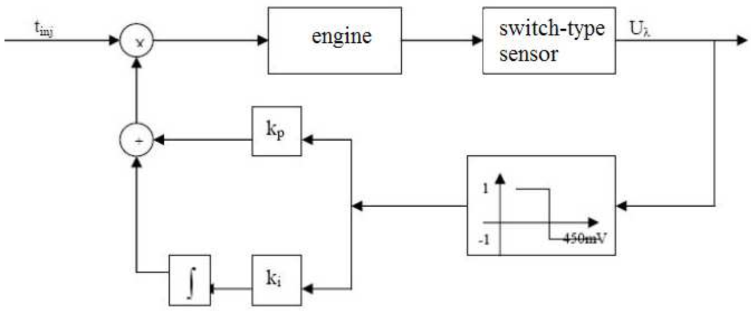

Figure 3.6 Control system of air-fuel ratio with switch sensor

The air- fuel ratio influences the combustion process. The air quantity m_a entering the cylinder depends on some effects: the throttle valve, the aerodynamic resistances, the resonance in the manifold, etc. The theoretical air quantity m_{ath} entering volume V_d with a standardized pressure (p_0 = 1.013 *bar*) and an air density (ρ_0 = 1.29 kg/m^3) is $m_{a,th} = \rho_0 V_d$. The ratio between real value and theoretical value of air mass is equal

$$\lambda_a = \frac{m_a}{m_{a,th}}$$

while as regards fuel mass m_f:

$$\lambda_f = \frac{m_f}{m_{f,th}}$$

The theoretical fuel mass $m_{f,th}$ is equal to the necessary mass for an ideal stoichiometric combustion. Under normal conditions the stoichiometric ratio is:

$$L_{st} = \frac{m_{a,th}}{m_{f,th}} = 14{,}67$$

And λ is defined as:

$$\lambda = \frac{\lambda_a}{\lambda_f}$$

Therefore, substituting the previous relations, λ can be defined as:

$$\lambda = \frac{m_a}{m_f}\frac{m_{f,th}}{m_{a,th}} = \frac{1}{L_{st}}\frac{m_a}{m_f}$$

For an ideal stoichiometric combustion, this ratio is equal to 1.

In the internal combustion engine, the air-fuel ratio is influenced by the variation of λ_a (that is established from driver) for a λ_f given [9]:

- Lean mixture ($\lambda>1$): the injected fuel is less than that necessary for a stoichiometric combustion. If λ belongs to the range $1<\lambda<1.1$, the thermodynamic efficiency increases all the same, because of the high temperature of combustion. This produces high nitrogen oxides emissions NO_x. For bigger λ, the efficiency is moving downward.
- Rich mixture ($\lambda<1$): the injected fuel is more than that necessary for a stoichiometric combustion. If $\lambda<0.9$, an unfinished combustion produces high hydrocarbons emissions in the exhaust gas and decreases the effective work. If $\lambda<1$, the thermodynamic efficiency is moving downward.

3.2.3. Spark advance control

The purpose of engine control as regards the spark advance, is to provide an advance that permits, depending on the different zones of engine working, to optimize the engine torque, the emissions, the fuel consumption and drivability. Moreover, it has to be able to prevent the knock phenomenon. The strategy is composed of some tables containing the basic advance in function of load and engine speed. These tables are stored in the ROM memory of the ECU and are defined experimentally through bench tests. Some corrections are produced to the basic advance to manage the effects of the engine temperature, EGR, the ambient pressure, and the knock.

The spark advance is used to rapidly vary the engine torque in the idle control and to increase the quality of gear change in the automatic transmissions. Figure 3.7 shows a typical scheme of spark advance control.

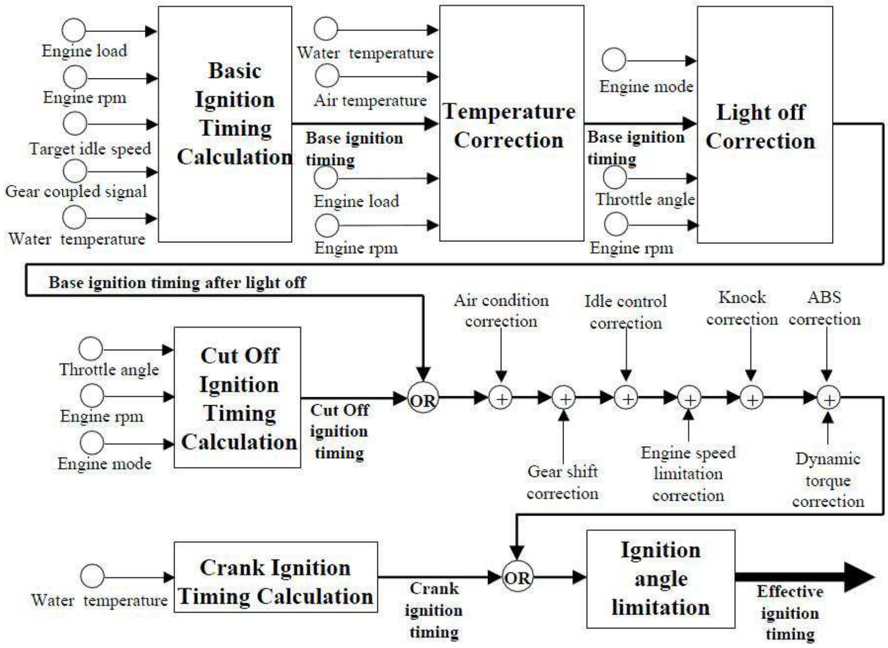

Figure 3.7 Advance control

3.2.4. Knock control

The internal combustion engine efficiency is also limited by the knock phenomenon. To avoid the knock phenomenon, the compression ratio has to be below a certain value and the ignition has to be optimized both off-line and on-line. The task of knock control is to identify the knock with sufficient sensibility and to control the advance with rapidity so that the engine works to the maximum, which divides the zone of normal combustion from that of knock. The first optimization is realized in the phase of engine development. The nominal ignition times are stored in the engine control unit. These times have to be corrected considering the fuel quality and the engine characteristics. For this reason, a knock sensor is fundamental. This sensor is fixed to the engine block and it is able to inform when the combustion becomes rough because of knock. Therefore, the sensor sends a signal to the ECU which intervenes by retarding the ignition (as to decrease the advance), until when the combustion doesn't resume the normal progress.

The knock control is very important in the zone of high load and high speed. In fact, on the one hand in this region the knock phenomenon is very dangerous while on the other hand it is important to reach a high level performance.

Moreover, consumptions and exhaust temperatures, which are always critical at the high loads, are decreased [8].

3.3. DIAGNOSIS

The diagnosis is the ability of the electronic system to verify cyclically the good functioning of external components (sensors and actuator) or of the system itself and to distinguish between failures and noises. It is actuated through the knowledge of the working condition of the control system and of the correspondence between this information and the measured values. The diagnosis was born for different motives:

- Normative requirements;
- Safety of the people and prevention of the risks;
- Monitoring of the quality of the product and satisfaction of the requirement customer market;

One of the main targets is to provide strong and reliable products and/or services. A product is *strong* if it is little sensitive to the causes of variability of the performance which can be generated from:

- Control factor;
- Environmental factors and conditions of usage;
- Noise factor (parameters of prior controllable process but difficult or expensive to control);

A product is *reliable* if it satisfies all the customer's expectations regarding to the functions, maintaining the answers in the range of acceptability expected by the costumer.

In the past, the systems were quite easy and at each failure a specific behavior of the system was referable. With the progress of the electronics and the increase of the complexity of the systems in the vehicle, an abnormal behavior was not always referable to an exact cause. For this reason, the necessity to use the information presented in the electronic systems and to render it available, the diagnostic tool is born. The diagnostic system receives some input from the control system in terms of activation conditions and of parameters of engine working (Figure 3.8).

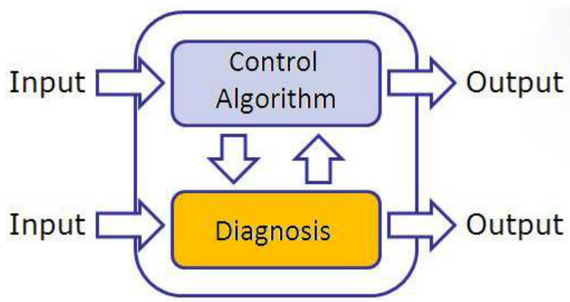

Figure 3.8 Diagnostic system

The diagnoses can be of three types:

- *Electrical diagnosis*: algorithm based on the identification of limits of out range for physical signals;
- *Diagnosis of congruence or plausibility*: algorithm based on physical congruence relations presented between different types of input (this is possible only if in the system there are more measures of the same physical phenomenon);
- *Functional diagnosis*: algorithm of congruence performed between different types of information of the system which present functional relations.

The diagnosis system of the ECU, after having verified the signal status, points the failure to the start or during the working (recovery). The recovery strategies are algorithms which, in the case of fault, permit to operate with decayed performances. If a sensor doesn't operate, the ECU replaces the value provided by the sensor itself with a value of recovery, calculated in an indirect way, that is through values obtained from other sensors. The purpose of these strategies is:

- To avoid other failures to the system;
- To guarantee the safety of the passengers;
- To assure the achievement of the welfare network;

The operating procedures of the faults expect:

- The identification of the conditions of existence of a fault;
- Filtering state of the fault;
- The storage DTC (Diagnostic Trouble Code) in the control unit;

The current normative expects that the faults together with other data stored in ECU have to be easily acquired from the outside through a diagnostic connector. After having pointed out any failure of the system, the ECU enters filtering state, where it persists for some validation time and during this phase the DTC aren't stored yet in the ECU memory so that it is able to isolate a generic noise from a real fault of the system. At the end of this state, if the fault is present still, it is stored in the not volatile memory.

Each ECU is connected to:

- Some sensors which permit the control unit to acquire information from the outside;
- Some actuators which permit the control unit to perform particular operations;
- CAN (Controller Area Network) communication network which permits the control unit to meet the other ECUs and, moreover, to perform the diagnosis operations through suitable tools.

The purpose of the diagnosis algorithms is to control the exact working of all the actors (hardware and software) involved in the control system. These algorithms work in the background during the normal operating and provide output only if some conditions of activation are verified. In general, they have to:

- Recognize the fault type;
- Establish the possible causes;

- Avoid the functional deterioration originated from fault;

Signal the fault to the driver through lamp M.I.L (Malfunction Indicator Lamp)

3.4. ECU SOFTWARE AND HARDWARE ARCHITECTURE

The ECU software is written typically in assembler code with real-time kernel. Over the last few years, there has been a strong trend to the high level programming (for example MATLAB/Simulink software) and then to utilize suitable tools which permit the translation in assembler code for the desired platform.

A typical ECU hardware architecture includes (Figure 3.9):

- Some standard microcontrollers and at least one TPU (Time processing Unit) which synchronizes the commands of engine control with the reciprocal engine actions;
- RAM memory to store the variables;
- EEPROM memory to store programs and tables of calibration;
- A/D converter which converts and conditions the signals from analog to digital coming from sensors;
- D/A converter which converts and conditions the signals from digital to analog to send in to the actuators;
- CAN communication interface;

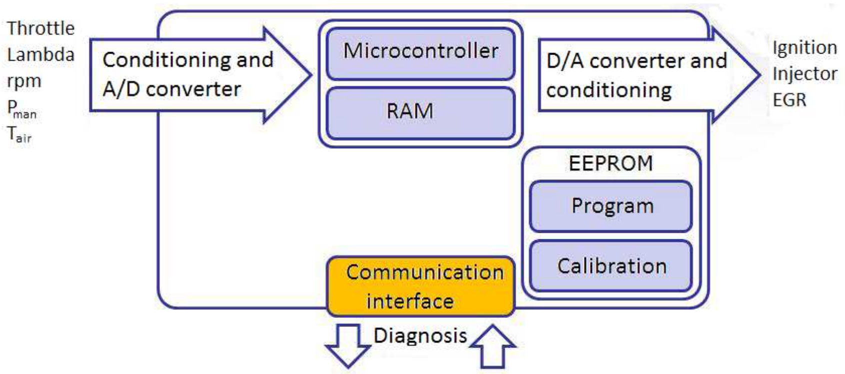

Figure 3.9 ECU hardware

Therefore, the "command center" of the ECU is a small microcomputer with a program memory which stored all the algorithms for the control processes. The input variables that derived from the data received from sensors influenced the algorithm calculations and, consequently, the control signals for the actuators. The actuators convert the electrical signals into physical quantities. Moreover, the ECU can also exchange data via the CAN with other electronic systems such as the ESP (Electronic Stability Program). In this way, the engine management system can be integrated in the overall vehicle management system.

3.5. SENSORS

The engine management system employs sensors to gather the operational data required for open and closed-loop control of the engine. By monitoring physical and chemical parameters the sensors are able to furnish information on the engine's current operating status. Examples of these monitored parameters are: manifold pressure, camshaft position, crankshaft rotation rate, etc.

3.5.1. Phase and engine speed sensors

The signals generated by engine speed sensor and by phase sensor permit the ECU to identify univocally the crankshaft position and the phase of the cylinders. At the start, the ECU proceeds to the recognition of the phasing of the injection and of the ignition, which are fundamental for the functioning of all the strategies. This identification is performed on the basis of the interpretation of the sequence of signals coming from the phase sensor, placed in proximity of the trigger wheel, and from the phase sensor, placed on the camshaft. The number of teeth on the trigger wheel is 60 although 2 teeth are omitted as to create a discontinuity: the angle between two consecutive teeth is 6° while the angle relative to two absent teeth is 12° more than the range between two consecutive teeth. The two absent teeth allow the synchronization with the camshaft and to have a reference mark for the measure of the angular position. The crankshaft sensor is an inductive sensor. Therefore it generates a sinusoidal signal which is proportional to the rate of change of the magnetic flux. The level of the magnetic flux depends on whether the sensor is opposite a trigger wheel tooth or a gap: if the distance between the trigger wheel and the sensor is minimum, that is a tooth is faced with sensor, the magnetic flux increases, while if the sensor is opposite a tooth gap, the flux decreases. The amplitude of the sinusoidal voltage increases strongly along with increasing trigger wheel speed. The signal generated from crankshaft sensor has to be converted in a digital signal through an A/D converter. The phase sensor is a Hall effect sensor that outputs a waveform square.

The outputs of two sensors in function of engine speed define the synchronized signals (Figure 3.10).

3.5.2. Manifold pressure and temperature sensor

A manifold pressure sensor sends a voltage value to the engine control unit, proportional to the pressure measured. The sensor is a debimeter based on hot film or hot wire air mass meter. It is constituted by a platinum thin plate heated electrically and crossed by an air flow that tends to cool it. A circuit controls, in feedback, the temperature of the thin plate, maintaining it at a constant value, and the current which serves to make this, represents a measure of air mass flow rate that crosses the sensor. This signal is converted into a voltage to send to ECU. This sensor is able to evaluate the flow direction using two temperature sensors present on the thin plate, which also provide the value of air crossing manifold.

The sensor has the advantage to perform a direct measure with rapid response time (below 15ms). It also presents drawbacks: decay of the performances caused from soiling, cost and sensibility to pulsation of flow.

In engines with turbocharger a sensor to measure the pressure afterwards is installed. The sensor is of the same type as that for measuring the manifold pressure. The information of the afterward pressure turbocharger is used for turbocharger control and for the management of the transitions of load (as information on the pressure to the bottom of throttle valve to advance the progress of the manifold pressure).

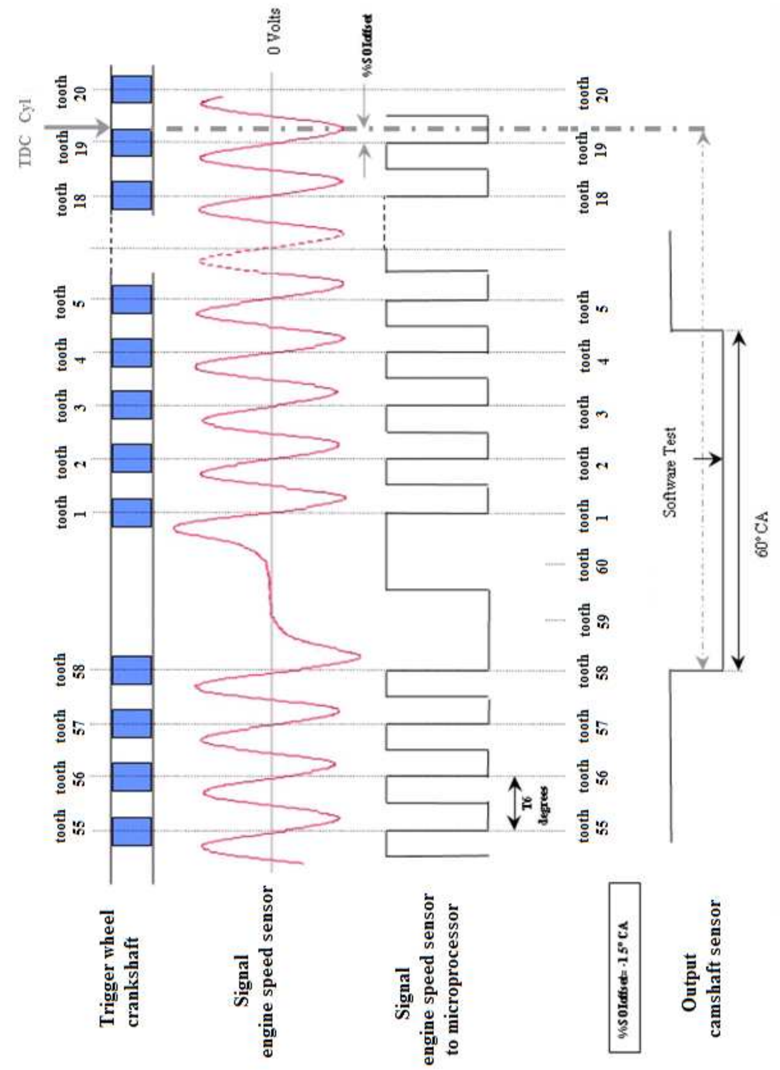

Figure 3.10 Synchronized signals

3.5.3. Accelerator- pedal sensor

The driver has the possibility to send some requests to the engine- vehicle system and the engine control unit has the task to interpret these requests that directly affect the working of the engine. On today’s electronic engine management systems, the driver’s accelerator inputs are transmitted to the ECU by an accelerator pedal sensor which registers the accelerator pedal travel, or the pedal's angular setting, and sends this to the engine ECU by means of an electric signal. The core of this sensor is the potentiometer across which a voltage is developed which is the function of the accelerator pedal setting. A second (redundant) sensor is incorporated for diagnosis purposes and for use in case of malfunctions. The voltage across this potentiometer is always half of that across the first potentiometer. The gas-pedal position, together with information about the actual working condition of the system (engine and vehicle speed) permits the ECU to calculate the engine torque required by the driver through the gas pedal [11].

3.6. ACTUATORS

After receiving the electric signals transmitted by the sensors, the ECU processes these data in order to generate control signals for the actuators. The actuators can be commanded in various ways:

- Through application of a constant voltage (relays and low-power loads);
- Through PWM (Pulse Width Modulation) signals;
- Through analog signals in current (rarest, used for immunity at the noises);

3.6.1. Coils and Injectors

The injection phase, defined as the angular time of end injected that is measured in comparison with the end of intake phase of correspondence cylinder, is calculated in function of the engine speed and of the intake efficiency. To guarantee that the combustion begins at the desired engine angle, it is necessary to establish with how much advance to start the charge of the coils, so that these latter have the sufficient power to generate the spark at the desired time. Thanks to this information (injection time, spark advance, time to charge coils) it is possible to generate the signals to guide the coil and injector drivers. Coils and injectors are guided by some low-side drivers, which are driven by a digital signal so that they can operate as a controlled switch. The time to recharge the coils measured in engine degree is called *dwell angle*. This angle is function of the engine speed and battery voltage. It is such as to allow the achievement of the desired current value on the primary circuit of the coil immediately before the time in which the spark goes off. This has to be done to assure the necessary charge, also in case of rapid transient response.

In the gasoline engine, the Time-Angle operation is used, which is characterized by two modes of working:

- *Time- priority*: in this mode the active duration of signal (dwell time DT) doesn't change; the duration of low-pulse is constant. This means that if the period, with which the wheel turns, changes, the leading edge occurs at a different angle (for example because of a deceleration);
- *Angle-priority*: this mode is used, instead, when it is required that the signal has the leading edge to an exact angle.

For the injectors the Time-Angle with time priority mode is used, because the duration of the injection has priority while for the coils the Time-Angle with angle-priority mode is used because the priority is the ignition angle.

3.6.2. Throttle Valve

In the traditional system, a mechanical connection transmits the movement of the gas pedal to the throttle valve. In this case, the air quantity taken in by the engine depends exclusively on the action of the driver on the pedal and the task of the ECU reduces itself only to measure the air mass flow rate through sensors and to inject the exact fuel quantity.

Today the drive-by-wire systems are used. They are provided by an electronic throttle control (ETC) and in these the pedal position is measured by a potentiometer and is converted in an electrical signal which is sent to ECU. In this way, the connection between pedal and throttle is eliminated so the ECU will manage the air mass flow rate.

The electronic throttle control consists of a direct current electric motor that, through a mechanism of reduction, permits to animate the throttle pan (Figure 3.11). The shaft, on which the pan is fitted, feels of the action of two return springs. These springs act separately: one exclusively on the small angles (that is the angles which permit to manage the idling) the other on the further angles. However, there is a position which suffers the combined action of both springs and it is a position of repose in which the throttle sets itself when the DC-motor is switched off. This position, which represents a limp home condition for engine system, is called NLP.

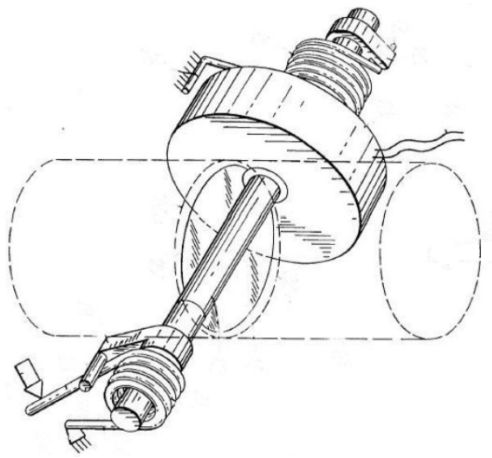

Figure 3.11 Electronic Throttle Control

The main operation statements of the ETC are:

- Response time for a maximum excursion inferior to 100*ms;*
- Lack of under and over elongation;

The electric motor is managed from a H bridge controlled in duty-cycle. The frequency of the driver is connected to the noise level. In fact, over a threshold frequency (10 kHz, the noise emitted by the electric motor is no more audible. The duty cycle of command can be managed in two ways:

- *Slow Decay*: in this mode the terminal voltage of the electric motor commutes effectively from value of supply during T_{on} to zero during T_{off};
- *Fast Decay*: in this mode the terminal voltage of the electric motor commutes from value of supply in one way during T_{on} to the value of supply changed of sign (reversal of positive pole with that negative) during T_{off};

The signal sent by the accelerator pedal sensor is elaborated by the ECU and transformed into a signal going to the electric motor which animates and sets, in correspondence with desired opening angle, the throttle valve. A greater accuracy in the positioning is guaranteed by the feedback: a sensor identifies the exact position of the throttle valve and, in correspondence with it, sends a signal to the ECU, which is able to evaluate the difference between the desired and the real position. Therefore, the sensor that measures the throttle opening angle assumes a central role, since at this angle the control loop on the throttle valve is closed. Given the importance of this information, to guarantee a certain safety level two sensors are used, this both to realize the functional crossed diagnoses and to have a functional redundancy. To obtain the two analog signals suitably treated, on the throttle device two potentiometers are present. These potentiometers provide two voltage values directly proportional to the angle of the throttle valve opening. They have a voltage characteristic with opposite slope. Each 4*ms*, the values of voltage are read to the terminal potentiometers and then, after having inverted the characteristics, one performs the comparison. If these voltage values diverge from a certain value, the system goes to recovery and a safety angle is given as output. In other words, the driver input is registered by two potentiometers and then taking into account the engine's actual operating status, the ECU calculates the throttle valve opening (which corresponds with the driver input) and converts it into a triggering signal for the throttle valve drive.

In the gasoline engine, the power is thus modulated from the throttle valve. The usage of throttle valve therefore causes to lose power to the engine, increasing the consumptions and the emissions exactly when the valve is opened a little, that is, when little power to the engine is required as occurs typically in town. The solution to this problem is to regulate the air aspirated from the engine, modifying the opening times of the intake valves. This is an intelligent solution but it is complex to realize. In this context, the MultiAir is born. The solution developed by Fiat is a middle course between a system without cams and a hydraulic control of the valves. The functioning can be summarized in the following way: the cam moves a hydraulic piston which in its turn controls the intake valve. The oil is practically irrepressible and, therefore, it acts as a rigid body and the system is thus exactly as a classic working of the valves. The trick is in an ON/OFF valve which opens the oil chamber putting it in communication with a small accumulator placed nearby. When this valve is opened, the oil, put in pressure by the cam, goes to the accumulator and doesn't press the cam which is closed through a spring. To obtain a delayed opening it is sufficient to close the ON/OFF valve with a small delay, while to close in advance it is enough to open the valve in advance. Through the management of the valve opening times, the air mass flow rate to the engine is regulated apart from the throttle valve. In theory, the MultiAir engine doesn't necessitate the throttle valve, but in reality the throttle is still present only now it is electro-actuated. Therefore it is no longer managed by the driver through the gas pedal. The reason to still have the throttle valve is connected to safety motives in case of malfunctioning. In reality, there is another aspect of interest which is the impossibility to measure and to know the position attained by the valve. In fact, the system permits to select the valve when sending the signal to open or to close but it is not possible to know exactly when this occurs and, it is not possible either to measure the position attained by the valve. Instead, if it had been in this way, the throttle valve could have been eliminated completely. This aspect underlines that the compressibility of the oil changes with the temperature, which influences the delay between the command and the effective opening. This can make the management of the air difficult in some conditions. Therefore, the presence of a throttle valve helps in the critical conditions. The strategies of the opening of the valves are defined on the basis of the engine working. In particular, at low speed with little accelerator, the valves are opened with delay and closed in advance to obtain a little raise and thus a little flow rate. When the engine torque is required to start at a low speed, the maximum raise is obtained but the valve closure is advanced. This permits reducing the crossing with the exhaust valve, improving the air charge at low speed. In the intermediate working zone, the system can execute the duplex openings (multi-lift) which create a greater turbulence supporting the mixing of air and fuel to obtain a good combustion.

Chapter 4 Engine Model

In this chapter a multi-cylinder, internal combustion engine model is presented. The chapter describes a new modular, physically based and lumped parameter (zero dimensional) approach, leading to a complete and coherent model structure. The model is conceived to be used with general purpose simulation software, as MATLAB/Simulink. The mean value outputs of the model (pressure, temperature, mass flow, torque) are compared with experimental data, collected by real engines (some new Fiat engine examples are used), in order to perform a standard identification and validation procedure.

4.1. ENGINE MODEL DYNAMICS

The internal combustion engine can be seen as a nonlinear process with multi-input and multi-output, provided both with controllable inputs (i.e. fuel mass flow rate and spark advance) and with uncontrollable but measurable inputs (i.e. throttle opening angle) and with uncontrollable and unmeasurable inputs (i.e. load torque). The complexity of the models, involves the physical and chemical aspects of the combustion process (mono-zone or multi-zone), the air flow dynamics (wave effects) or the liquid deposition and vaporization dynamics (wall wetting).

Widely accepted method for modeling Naturally Aspirated (NA) Spark Ignition (SI) engines is the Mean Value Engine Model (MVEM) thanks to its great flexibility, good performance and low power computation [12]. It describes the average dynamics of engine physical variables over time periods which are long compared to the dominant time constants of the engine, such as crank shaft speed, manifold pressure and air/fuel ratio [13],[14], [15]. Nevertheless, MVEM does not include explicitly a description of the intake, exhaust or combustion processes, but simply represents the overall result. A major advantage of such a kind of models is their simple structure: no more than three nonlinear differential equations, related to the air flow through the intake manifold, the well-wetting (film fluid) on manifold wall, and the engine crankshaft dynamics, respectively and certain number of algebraic equations.

4.1.1. Air Dynamics

The air dynamic describes the development of pressure manifold and, accordingly, air mass flow rate entering the cylinders. The system inputs are throttle angle (α) and crankshaft speed (n) while the outputs are manifold pressure (p_{man}) and air mass flow rate entering cylinders (m_{acyl}). As shown Figure 4.1, the air dynamic is composite from three blocks:

- *Throttle valve block* calculates air mass flow rate that passes through the throttle valve. The throttle valve controls air flow rate entering cylinder as that it can regulate engine torque. Supposing one-dimensional isentropic flow through a duct of compressible fluid, the throttle valve model is:

$$\dot{m}_{th} = \begin{cases} C_d \cdot A_{th}(\alpha) \cdot \dfrac{p_{amb}}{\sqrt{R\,T_{amb}}} \cdot \left(\dfrac{p_{man}}{p_{amb}}\right)^{\frac{1}{k}} \cdot \left\{\dfrac{2k}{k-1} \cdot \left[1-\left(\dfrac{p_{man}}{p_{amb}}\right)^{\frac{k-1}{k}}\right]\right\}^{\frac{1}{2}} \text{ se } \dfrac{p_{man}}{p_{amb}} > p_{CR} = \left(\dfrac{2}{k+1}\right)^{\frac{k}{k-1}} \\ \\ C_d \cdot A_{th}(\alpha) \cdot \dfrac{p_{amb}}{\sqrt{R\,T_{amb}}} \cdot (k)^{\frac{1}{2}} \cdot \left(\dfrac{2k}{k+1}\right)^{\frac{k+1}{2(k-1)}} \text{ se } \dfrac{p_{man}}{p_{amb}} \leq p_{CR} = \left(\dfrac{2}{k+1}\right)^{\frac{k}{k-1}} \end{cases}$$

Where C_d is the is charge coefficient that has been introduced to consider non-isentropic flow and is determined experimentally, A_{th} is the throttle plate open area and is a function of angle α for the throttle plate geometry, P_{amb} and T_{amb} are ambient pressure and temperature respectively, P_{CR} is critical pressure ratio; $k=c_p/\ c_v$ specific heat ratio, R gas constant.

- *Intake manifold block* estimates the manifold pressure. Several models of the flow in an intake manifold have been proposed. One simple manifold model that describes many of the above phenomena is the filling and emptying model. It is based on the assumption that at any given time the manifold pressure is uniform. Further, the filling and emptying model is a mean value model which ignores, therefore, periodic components of pressure generated from motion of intake valve. In all working state, the mass of air, intaken from engine, depends on environmental conditions to the bottom of throttle, that is pressure and temperature, that are supposed the same of those outsides, on throttle angle, on physical conditions presented in the manifold, on engine operative point given from engine speed. The equation for this model is:

$$\dot{p}_{man} = -\frac{n}{120} \cdot \frac{V_d}{V} \cdot \lambda_v\, p_{man} + \frac{R\,T_{man}}{V} \cdot \dot{m}_{th}(\alpha, p_{man})$$

Where V_d is the engine displacement, V is the manifold volume, λ_v is the filling coefficient and T_{man} is the manifold temperature.

- *Speed density block* calculates the air mass flow rate entering the cylinder that is function of the crankshaft speed, of the manifold pressure and of the throttle angle. The equation of the model is:

$$\dot{m}_{acyl} = \frac{n}{120}\,\frac{V_d\,\lambda_v}{R\,T_{man}}\,p_{man}$$

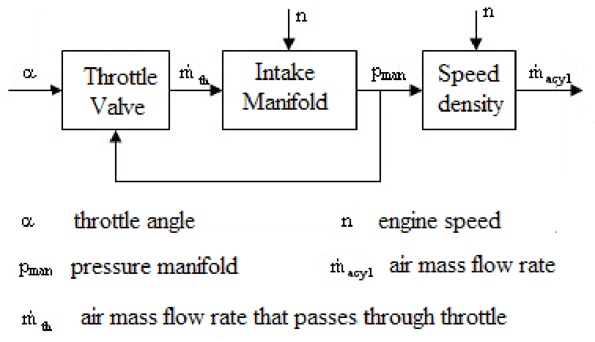

Figure 4.1 Air dynamics

4.1.2. Fuel Dynamics

In internal combustion engine, the amount of injected fuel is determinate from appropriate algorithms in function of air mass flow rate expected in all engine operative point. Typically, inside the engine control unit some maps are memorized which provide, in function of the manifold pressure, of the engine speed, of the engine temperature and of the other state variable of system, the amount of fuel to inject to obtain an optimal air- fuel ratio in that engine operative point. The fuel enters the air stream as a liquid jet. When the fuel is injected in the manifold, the air flow at high speed atomizes it into droplets. The smaller droplets vaporize completely according to the equations that regulate the mass and heat transport process between the fluid droplets and the gas flow, or remain suspended in the flow. The largest droplets, instead, impact on the manifold walls making so called well-wetting (film fluid). The fuel film, that is deposited on manifold walls, vaporizes therefore with a time constant that depends from the temperature of the walls and from the air in the manifold. The system input is the fuel mass flow rate while the system output is the fuel mass flow rate entering cylinder (Figure 4.2).

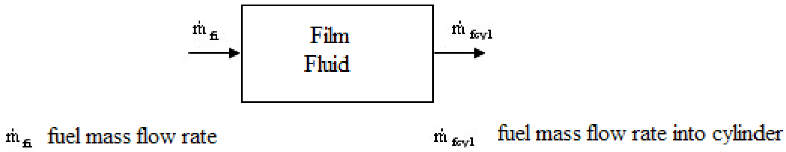

Figure 4.2 Fuel dynamics

Several models of the behavior of liquid- fuel wall- films have been developed. An mean value engine model is used to describe the fuel dynamic. The equations that describe this dynamics are :

$$\dot{m}_{ff} = -\frac{m_{ff}}{\tau} + x \cdot \dot{m}_{fi}$$

$$\dot{m}_{fv} = \frac{m_{ff}}{\tau} - \frac{m_{fv}}{\tau_m} + (1 - x) \cdot \dot{m}_{fi}$$

$$\dot{m}_{fcyl} = \frac{m_{fv}}{\tau_m}$$

Where m_{ff} is film fluid, m_{fv} is the fuel hanging in air at gas or liquid state in the manifold, x is the fraction of fuel that deposits on the manifold walls, τ is the evaporation constant of film fluid and τ_m is the characteristic time of the intake process.

4.1.3. Crankshaft Dynamics

The crankshaft dynamic is composed from two parties: combustion process and crankshaft speed. The combustion system models the process of transformation of chemical energy into mechanical torque and heat by the combustion of air-fuel mixture into the cylinder. In Figure 4.3, the blocks composing the crankshaft dynamics of the engine model are presented. The combustion process is composed from following blocks:

- *Burned Fuel block* calculates the fuel charge actually burned during the combustion. It is supposed that only the air-fuel mixture that is in a stoichiometric ratio (14.6 part of air for 1 part of fuel) participates actively to the combustion process. If the air-fuel mixture is lean ($\lambda \geq 1$) all the fuel in the cylinder takes part at the combustion; while, in rich condition ($\lambda<1$), only the 14.6 – *th*-part of the cylinder air is burned. Thus, the burned fuel is computed as follows:

$$\dot{m}_{fe} = \frac{\dot{m}_a}{\lambda_{ST}} \cdot \begin{cases} \frac{1}{\lambda} & \lambda \geq 1 \\ 1 & \lambda < 1 \end{cases}$$

  Where λ_{ST} is the stoichiometric value of the air-fuel ratio.

- *Combustion efficiency block* estimates the efficiency of the engine (η_f) in transforming the chemical energy of the fuel into mechanical energy through the combustion. It is modeled as a product of two components η_λ, that computes the combustion efficiency as a function of the engine operating point, and η_{AV}, that modulates the combustion efficiency as a function of the distance between the real and the nominal value of the spark advance:

$$\eta_\lambda = c_0 + \theta(c_1 \lambda^2 - c_2 \lambda)$$

$$\eta_{AV} = 1 - c_3(\theta - \theta^*)^2$$

$$\eta_f = \eta_\lambda \cdot \eta_{AV}$$

Where θ is the spark advance, θ^* is its nominal value for the production of the torque from the combustion and it is a nonlinear function of the engine operative point:

$$\theta^* = f(n, \dot{m}_a, T_{cool}),$$

Where n is engine speed and T_{cool} is coolant temperature.

- Combustion block estimates the effective torque generated by the combustion (T) and the heat produced by the combustion warming the exhaust gas:

$$T = \frac{\eta_f \, \dot{m}_{fe} Q_{HV}}{n}$$

$$\dot{Q}_u = \beta(t)(1 - \eta_f)\dot{m}_{fe} Q_{HV}$$

Where Q_{HV} is the low heat value of the fuel and β describes the time- varying partition of the thermal energy directed toward both the engine mechanical components and exhaust gas

- *Thermal dynamic block* models the dynamical behavior of the exhaust gas temperature:

$$\dot{T}_{FG} = a_0 \dot{Q}_u - a_1 n \left(T_{FG} - T_{cool}\right)$$

- *Unburned hydrocarbons block* calculates the total unburned hydrocarbons (THC_{pre}) at the exhaust pipe as a function of the feedgas temperature (T_{FG}), air mass flow rate, spark advance (θ), engine speed (n) and air/fuel ratio (λ).

The *crankshaft block* has a behavior comparable with that of a first order system, which time constant is a nonlinear function of the engine operative point. It calculates the engine speed.

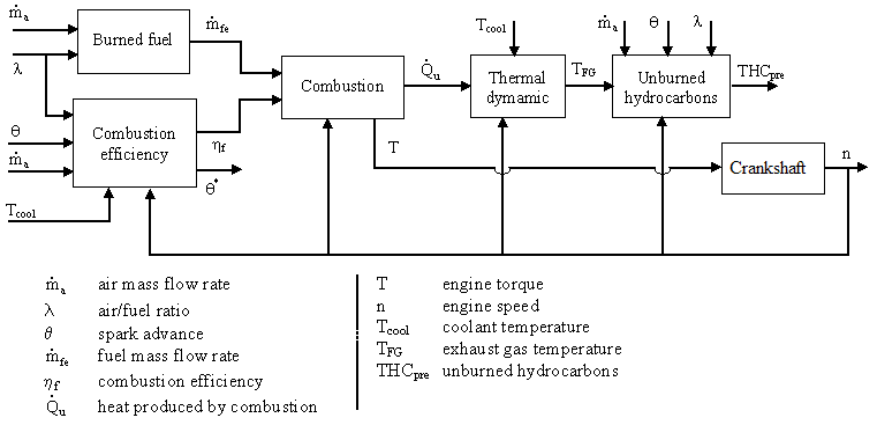

Figure 4.3 Crankshaft dynamics

4.2. AN ELECTRICAL INTERPRETATION OF ENGINE MODEL

The adopted modeling methodology, starting from an analogy with electrical systems, simplifies the approach eliminating the space dynamics (multi-zone combustion and wave effects), while preserving the time dynamics (resonance and tuning phenomena). In this way, it is obtained an engine description similar to an electrical (although not linear) circuit, with all the useful consequences in term of existence and numerical availability of the solution. The advantages are in the specific comparison that is found between the engine components and variables (as throttle valve, cylinder, inertial flows), with electrical counterparts (current, voltage, resistance).

The main benefit achievable is the simplicity to compose the whole engine model and customize it including all the latest devices. So it is possible to easily implement both a baseline engine and a high complex automotive system (see [17]).

The application domain of the engine model, that is possible to obtain with this methodology, is the analysis of dynamical behavior of engine, too difficult to investigate on the real system, an example scavenging or exhaust gas recirculation, allowing to test the engine dynamics under different operative conditions with a good level of reliability and accuracy.

There is an analogy between mechanical and electrical system [16]. A physical system can be decomposed in elementary subsystems or elements. For each elementary subsystem is possible to define three variables named:

- q quantity, i.e. variable of the element
- i flow, equal to dq/dt, i.e. the variable that flows across the element
- v forcing, i.e. the variable acting at the extremes of the element

The Figure 4.4 reports these variables for electric, hydraulic and mechanical elements. The main features is that one variable can be considered constant despite the others or, alternatively, can be considered constant the following ratios:

$$\frac{v}{i}, \frac{v}{\frac{di}{dt}}, \frac{i}{\frac{dv}{dt}}$$

as shown in Table2of Figure 4.4.

Table 1. Variables

element	q	i	v
electric	charge q $[C]$	current i $[A]$	voltage v $[V]$
hydraulic	volume V m^3	flow rate $\dot{m}$ $[kg/sec]$	pressure p $[N/m^2]$
mechanic	angle θ $[rad]$	speed ω $[rad/s]$	torque T $[Nm]$

Table 2. Parameters

element	parameters		
electric	resistance $R = \frac{v}{i}$	inductance $L = \frac{v}{\frac{di}{dt}}$	capacity $C = \frac{i}{\frac{dv}{dt}}$
hydraulic	pneumatic resistance $R = \frac{p}{\dot{m}}$		pneumatic capacity $C = \frac{\dot{m}}{\frac{dp}{dt}}$
mechanical	friction $B = \frac{T}{\omega}$	inertia $J = \frac{T}{\frac{d\omega}{dt}}$	elasticity $\frac{1}{K} = \frac{\omega}{\frac{dT}{dt}}$

Figure 4.4 Tables of variables and of parameters

For a network of elements and in particular conditions, the variables are related thought Kirchhoff laws, as follows:

- for each node of the network, the algebraic sum of the flow variables is zero

$$\sum_h i_h = 0$$

  where i_h is the flow across the *h-th*-element.

- for each mesh of the network, the algebraic sum of the forcing variables is zero

$$\sum_h v_h = 0$$

  Where v_h is the forcing acting on the *h-th*-element.

Considering the engine formed by mechanical components, as throttle valve, manifolds, cylinders and crankshaft, crossed by a gas, the analogy among the electric and the mechanic and hydraulic elements is presented. Regarding the mechanic systems an analogy, named "Maxwell's analogy" can be found among speed and torque respectively with electric current and voltage. It results in considering the mechanical friction B as an electric resistance R and, similarly, the inertia J as an electric inductance L and the inverse of the elasticity K as a capacitor C. The same considerations can be done for the hydraulic elements. Here the analogies are between the gas flow rate with the current, the pressure with the voltage and the volume V with the charge q. Then, the electrical resistance corresponds to the pneumatic resistance, that is the resistance of gas flowing across an orifice, and the volume to an electric capacitor. All the parts composing the engine can find an equivalent electrical circuit or element. Moreover, in order to exactly describe the operation of electrical circuit, it is necessary to use the Maxwell equations. These can capture both the dynamics of the electrical quantities, such as currents and voltages, and the related electromagnetic phenomena, as transmission and radiation. Fortunately, if the size of the circuit is small compared to the wavelength of the electrical variables (i.e. the ratio between the light speed and the frequency of the pulsating events), these electromagnetic phenomena can be neglected. As a consequence, the partial differential relationships of the Maxwell equations can be simplified to the widely used electric equations, that are the Kirchhoff laws and the current/voltage relationships of circuit components. Similarly, the same approach can be extended to the internal combustion engine, providing that the wavelength (in this case the ratio between the sound speed and the frequency of its pulsating events) is large enough compared to the length size of the engine.

The analogy between the mechanical and electrical components can be extended to the relationship governing the relative quantities. In fact, the first Kirchhoff equation has a specific counterpart in the mass conservation law and, similarly, the Bernoulli equation can correctly replace the second Kirchhoff equation.

4.3. ENGINE COMPONENTS

The engine is seen as an array of cylinders, having common connections with an intake and an exhaust manifold. The connections are regulated by valves opening. the elements composing the engine can be classified in the following categories: volumes, orifices, inertial effects and combustion. In the following, each category is introduced and the relationship among the interested variables are reported.

4.3.1. Volumes

The intake and exhaust manifold and cylinders, are grouped respectively as constant and variable volumes. The electric counterpart is the quantity of charge stored in a capacitor, as shown in Figure 4.5 reporting the corresponding circuit.

Applying the corresponding current/voltage relationship and considering the analogies with pressure and temperature inside the volume, it is possible to obtain the classical equations. Starting from ideal gas equations:

$$p\,V = m\,R\,T$$

Where R is the specific gas constant and m is the mass of gas, the equations are:

$$\dot{p} = \frac{R\,\gamma}{V}\left[\sum_i \dot{m}_i\,T_i - T\sum_j \dot{m}_j + \frac{\gamma-1}{R\,\gamma}\,\dot{Q}_{ext} - \frac{p\,\dot{V}}{R}\right]$$

$$\dot{T} = \frac{R\,\gamma\,T}{p\,V}\left[\sum_i \dot{m}_i\,T_i\left(1-\frac{T}{\gamma T_i}\right) - T\sum_j \dot{m}_j\left(1-\frac{T}{\gamma T_j}\right) + \frac{\gamma-1}{R\,\gamma}\,\dot{Q}_{ext} - \frac{p\,\dot{V}}{R}\left(1-\frac{1}{\gamma T}\right)\right]$$

Where i represents the entering mass flow and j the outgoing mass flow.

It is remarked that, regarding the intake and exhaust manifolds, since the volume V is constant, the derivative terms in the equation disappear.

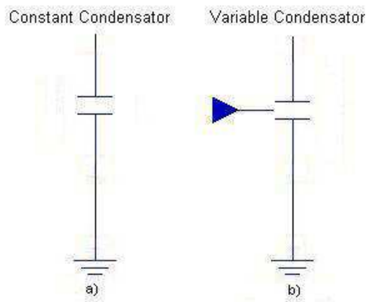

Figure 4.5 Volume equivalent circuit: a) constant capacity; b) variable capacity.

4.3.2. Orifices

The orifices are responsible of the pressure drops along the gas path. They are modeled as variable resistances causing equivalent voltage drops, as illustrated in Figure 4.6. The size of the orifice is variable and regulated by valve opening, as throttle valve, air bypass, intake and exhaust valves[1].

The electrical resistance is governed by a static relationship between voltage and current, corresponding to a static relationship between the analogue variables, i.e. pressure and flow rate, in according to the well known equations [14]:

$$\dot{m}_c = \frac{C_D A_T p_0}{\sqrt{R\,T}}\,\gamma_{\frac{1}{2}}\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{2\,(\gamma-1)}} \qquad \frac{p_T}{p_0} > 1$$

$$\dot{m}_{nc} = \frac{C_D A_T p_0}{\sqrt{R\,T}}\left(\frac{p_T}{p_0}\right)^{\frac{1}{\gamma}}\left\{\frac{2\gamma}{\gamma-1}\left[1-\left(\frac{p_T}{p_0}\right)^{\frac{\gamma-1}{\gamma}}\right]\right\} \qquad \frac{p_T}{p_0} \le 1$$

Where p_τ and p_0 are respectively the pressure upstream and downstream the orifice.

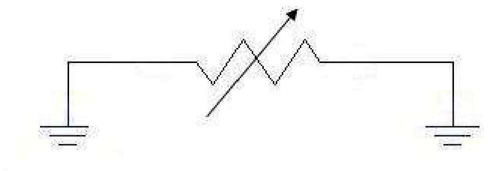

Figure 4.6 Orifice equivalent circuit.

4.3.3. Inertial effects

The inertial phenomena can be considered as minor efforts but not completely negligible. They describe the reduction or the increase of the pressure upstream the valve of a quantity proportional to the derivative of the mass flow through the same valve. They are modeled as an linear inductance (Figure 4.7) regulated by a differential relationship between voltage and current, corresponding to the following equation:

$$p_{corr} = p - k\,\ddot{m}$$

where p_{corr} is the manifold pressure, m is a generic mass and k is a parameter to be set.

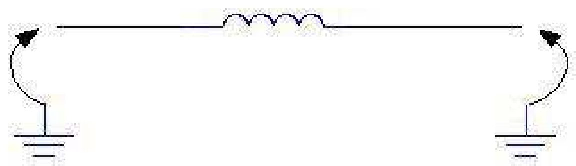

Figure 4.7 Inertial effects equivalent circuit.

4.3.4. Combustion description

The combustion process constitutes the most meaningful and complex phenomenon occurring into the engine. In order to model the in-cylinder cycle pressure, an equivalent electric circuit has been adopted, as shown in Figure 4.7. The circuit is formed by a variable capacitor, representing the cylinder volume equipped by an impulsive voltage generator. This causes an impulsive increase of the voltage at the capacitor extremities and, consequently, generates a current flow thought the capacitor. This phenomenon corresponds to the well known combustion process, i.e. an impulsive increase of the in cylinder pressure caused by the combustion resulting in a torque generation and in mass flow through the exhaust valves. The equation regarding this process is described by the following relationship:

$$\dot{Q}_{ext} = h_l A_l(\theta_{SA})(T_{sl} - T) + h_b A_b (T_{sb} - T) + m_b \eta_{cb} \eta_{burn}(\lambda, p)\ S\ (\theta_{SA}, p) Q_{HV}$$

Where h_l and h_b are parameters to be set and A_l and A_b the lateral and base area respectively. This equation represents the heat power generated by the combustion affecting the pressure and temperature . Its cyclic variation has been implemented as a function of the operating conditions of engine and of random factors needed to model the combustion cycle-by-cycle and cylinder-by-cylinder irregularity. m_b represents the fuel burned during the combustion, calculated as the product of injected fuel mass and mass fraction burned. Injected fuel mass can be calculated from fuel flow rate and fuel injection duration, while different mathematical functions can be used to empirically model the mass fraction burned.

An accepted model to fit the combustion cycle is the Wiebe function given by the equation:

$$m_b = m_{fuel} x_w$$

where $x_w$is the fraction of the total fuel m_{fuel} inside the cylinder

$$x_w = \frac{1 - e^{\tau \theta^{\sigma+1}}}{1 - e^{-\tau}}$$

with

$$\theta = \frac{\theta - \theta_s(\theta_{SA})}{\theta_{\Sigma}(\theta_{SA}) - \theta_s(\theta_{SA})}$$

The fraction of fuel, x_w,is function of crankshaft angle θ and varies between 0, when θ is equal to $\theta_s(\theta_{sA})$ (the angle that determines the combustion begin), and 1, when θ is equal to $\theta_{\Sigma}(\theta_{sA})$ (the angle that determines the combustion end). Both $\theta_s(\theta_{sA})$ and $\theta_{\Sigma}(\theta_{sA})$ are function of spark advance angle through static map obtained experimentally. The Wiebe function provides a convenient approach to model the mass burned fraction, computed directly from the combustion data using two only parameters: τ, σexperimentally selected

Finally, it is possible to estimate the corresponding combustion heat release, $S(\theta_{sA},p)$, for the firing cylinder, given by a single zone combustion map modeled as function of spark advance angle and cylinder pressure.

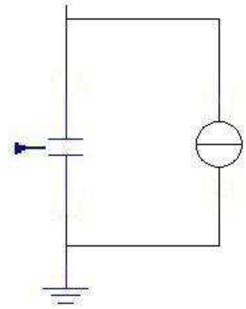

Figure 4.8 Combustion equivalent circuit

4.3.5. Equivalent circuit of the whole engine model

Based on the analogies depicted in the previous section, the entire engine can be represented by the circuit shown in Figure 4.8.

The model starts describing the dynamic of the air crossing the intake manifold, i.e. driven by the ambient pressure (a current generator), the air mass passes the filter (a resistance) and the throttle body (a variable resistance) and arrives into the cylinder through the intake valves (a new variable resistance). The cylinder are described by a parallel of *n* combustion equivalent circuit, with *n* the number of cylinders composing the engine. Finally, the gas mixture is discharged into the exhaust manifold through the exhaust valves (a variable resistance) and ends into the ambient crossing the muffler (a resistance). For sake of completeness, it is possible to introduce the inductors that naturally represent the inertial effects of the current and are able to describe the analogous effects of the fluid columns.

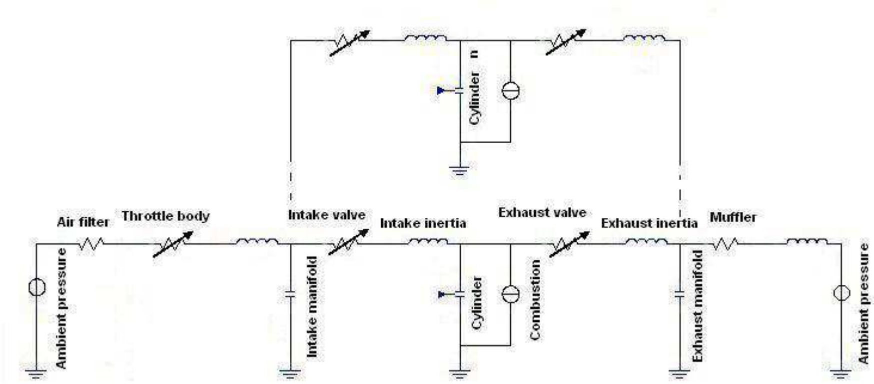

Figure 4.9 Internal combustion engine equivalent circuit

Chapter 5 Control systems design and calibration

Today motor vehicles are equipped with a set of embedded control systems with an increasing number of software implemented features. The design of such systems is a challenging problem because of the functions complexity to be implemented, for the constraints which are due to the tight interaction between mechanical and electrical components and for safety reasons.

Current software development is aided by simulation tools which allow block diagrams generation providing the customers with a graphical environment that supports the design and the simulation activities and can be run on PCs.

However, much more is required for the complete development of embedded control systems. The current lack of automated tools (see [19] and [20]), methods and models make the whole process tedious, time-consuming and potentially affected by errors and omissions because most steps are handmade, especially in the earlier stages.

The main goal of this chapter is to present some ideas for automating the whole embedded control system design process. As a result, many of the above ideas have been implemented in the F.I.R.E. tool which supports the embedded control systems modelling, design and simulation at a high abstraction level, allowing control engineers and software developers to focus on the control system aspects of the problem instead of the platform[1] ones.

Another relevant aspect when dealing with modern internal combustion engine control systems is the availability of new robust and multi-objective engine calibration methods and tools, which potentially allow substantial new flexibility and performance with respect to the traditional calibration practice. In this work, a novel general purpose model-based calibration methodology will also be described which merges statistical concepts, like the robust design theory of experiments, numerical optimization techniques and Engine Control Unit (ECU) algorithm modelling. This approach exploits software tools in order to support the calibration of estimation algorithms used in the ECU. The proposed tools are:

- non-linear multivariate regression;
- discrete regression;
- multi-map optimization;
- graphical user interface for calibration, validation and change;
- calibration performance meters.

An application of these tools to the “basic engine” calibration process of an actual real thermal engine is presented. With the term “basic-engine” calibration

[1] The specific hardware and the operating system on which the control system should be executed.

we mean the calibration activity dealing with the ECU algorithms involved in the engine operations which do not depend on a particular vehicle application like charge estimation, injector model, spark advance computation, torque estimation, catalyst protection and so on.

The use of this methodology has been integrated in the control algorithm design process thus speeding up, simplifying and improving the whole process. This tool suite for control algorithms design and calibration is one of the most important results of my research [18].

The application of this approach, compared with the best techniques used in industry, has produced really interesting results:

- reduction of experimental test bench design effort (more than 50% of reduction);
- more accurate estimation (almost doubled);
- more robust behaviour, with respect to engine to engine variability and environmental conditions.

These tools have been developed by using MathWorks' MATLAB® /Simulink® software, which is a de-facto standard in the embedded system industry.

5.1. F.I.R.E. TOOL CONTEXT USE

The development of embedded control systems is usually a complex process, because of the of the system size and the shared resources amount ([21],[22] and [22]). The modelling of these systems has to include the knowledge of the top level structure, the distribution of the functionalities amongst the system resources, the links and the data transmission amongst the components and the system response to asynchronous or synchronous events [21].

Such different information cannot be expressed by a single graphical notation. As an example, a single graphical view can show some abstract functional aspects, and another one shows the control software organization or the physical structure. An actual design methodology is based on the idea that every control system can be observed by using three different views: the *functional*, the *implementation* and the *re-usability* views.

5.1.1. Functional View

Following a model-based approach for embedded system design, during the functional design phase, where the customer requirements are analyzed, the control engineer plans a logical-functional view (Figure 5.1) of the control system, in which the functions of the different parts of the project, e.g. signal acquisition, error value computation or command signal actuation, are highlighted ([21] and [22]). Such a view focuses on the data flow, that is the process of identifying, modelling and reporting how data flows around the system. In this graphical notation, each activity is data-driven, i.e. every block undertakes its elaborations as soon as input data are available on its input ports, without waiting for any timing, priority or CPU availability.

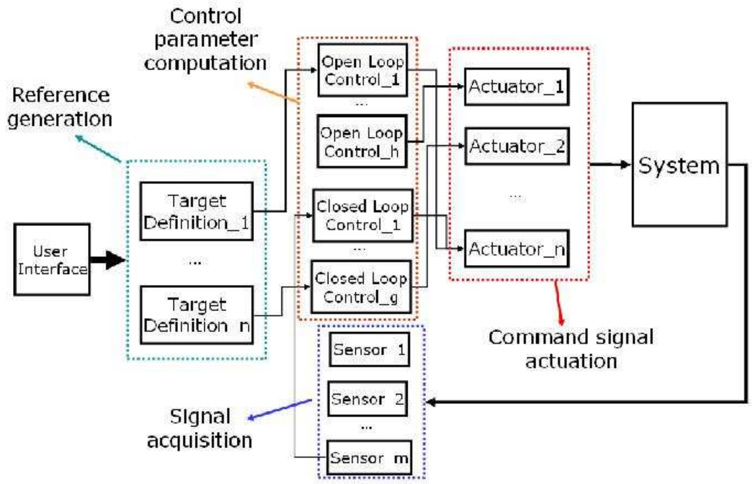

Figure 5.1 Functional view

5.1.2. Implementation View

The next step in system planning is the ANSI-C code implementation of the functional models onto the specific target to be installed into the vehicle Electronic Control Unit, which is usually accomplished by means of some automated production code generator ([26] and [27]).

A functional view is not suitable for dealing with the implementation aspects deriving by the use of a real-time operating system in the ECU, such as task timing and scheduling, or asynchronous events handling ([28] and [29]). Such aspects can be better analyzed in an *implementation* view (Figure 5.2) of the same model where the control flow[2] is also highlighted. In this view, the customer considers the effect of introducing prescribed timing requirements in the execution of each Simulink® block. This is accomplished by introducing an additional scheduler block which simulates the real behaviour of the control system when implemented on a RTOS (Real-Time Operating System) for embedded systems.

[2] The timing control of the activities that have being processed by the control unit.

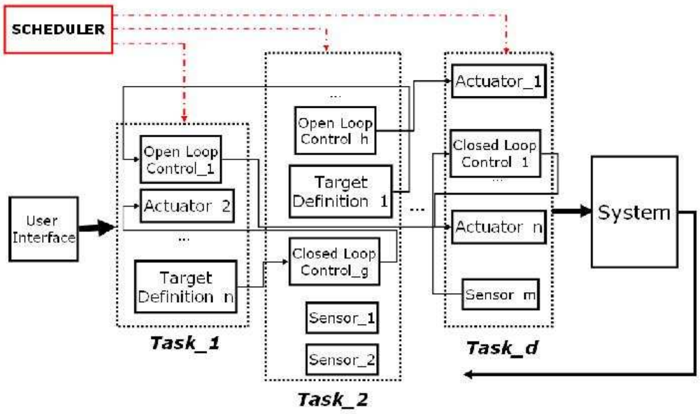

Figure 5.2 Implementation view

5.1.3. Re-usability View

After generating the code, it can be helpful, for taking care of possible future changes, to characterize the subsystems by their hardware dependence in a so-called *re-usability* view (Figure 5.3), in order to identify the HLSW (High-Level SoftWare) and the LLSW (Low-Level SoftWare) part of the generated code.

This view improves the re-usability of a control algorithm, because it provides a physical representation of the control system in which the process resources physical location is illustrated.

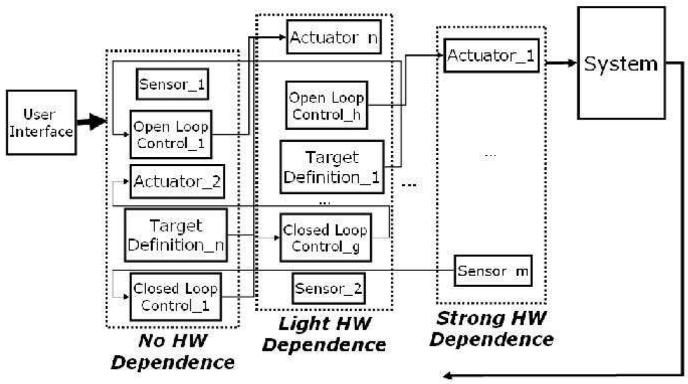

Figure 5.3 Re-usability view

5.2. F.I.R.E. TOOL OBJECTIVES AND MAIN FUNCTIONALITIES

The development of an embedded system involves the cooperation of different planning activities. In particular, the interaction between the design and algorithm implementation phases requires a (fragile) information sharing between control and software engineering groups. On the other hand, the control and software engineers have different points of view, design approaches, terminology and development tools.

Up to now, the system designer is the person in charge of choosing the most suitable view to model a control system, and all transformations of the Simulink® models, realized in collaboration with the software engineer, are manually accomplished. Such an activity is slow and affected by possible unavoidable errors and omissions.

F.I.R.E[3] has been created to automate these transformations, ensuring data consistency in each view of the same system. F.I.R.E. is a software tool which allows the designer to develop, to analyze and to simulate embedded control systems at a high abstraction level. It helps to realize more correct and efficient implementations of the functional graphical models, it supports the early steps of the *V-cycle* design methodology [22], assures more flexibility and re-usability of the Simulink® models and improves the development cost and time. Morevoer, F.I.R.E. assures that the implementation view is realized according to the Auto-Code Generators (ACG) requirements [26].

The Simulink® library has been extended to support the design of the F.I.R.E. models thus providing a user-friendly graphical interface. From these models, the control engineer can carry out an early investigation of the timing problems potentially affecting the control system, such as delays in the data flow, jitters, potential data-loss in buffers, and priority task handling ([29] and [30]). A simulation with different sampling times on the same system is also possible. In this way the results of such an analysis can be used by the software engineers to produce more correct ECU codes.

It is important to notice that the transformation from a functional view to a re-usability one consists in a graphical blocks rearrangement, which is useful during the reverse engineering phase, when focusing on some particular hardware dependencies of the control system is of interest. Moreover, transit from a functional view to an implementation one offers a logical tasks reorganization based on their timing requirements. On the other hand, the implementation view makes it possible to analyze different multi-rate simulations by acting on the scheduler block.

5.3. F.I.R.E. BLOCKSET

In order to have a user friendly tool, a customized block library is supplied and integrated with the other Simulink® built-in libraries. The F.I.R.E. block set containing all the components required to create new F.I.R.E. models (Figure 5.4). The customer can also introduce new templates of asynchronous event generators and update a MAT-file containing the hardware dependencies of the subsystems inside the functional view.

[3] Functional Implementation Re-usability Environment.

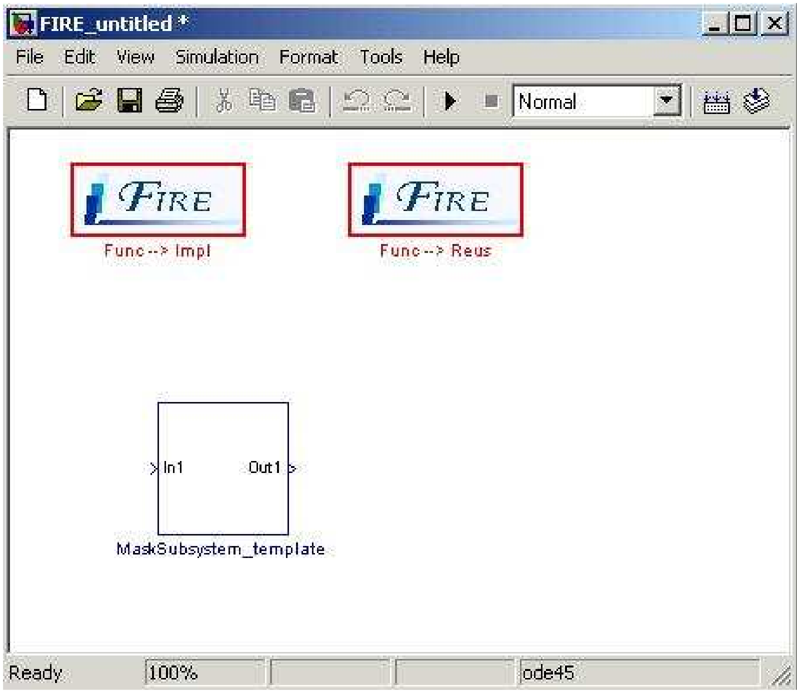

Figure 5.4 Building a new F.I.R.E. model

In order to carry out the transformations, the user must introduce some information into the model about the task timings or hardware dependencies. In order to introduce these properties in a standard and easy way, F.I.R.E. supplies a *MaskSubsystem_template* block (Figure 5.5), in which the customer can set the required information, like the sampling time or the declaration of an asynchronous events, and the hardware dependency.

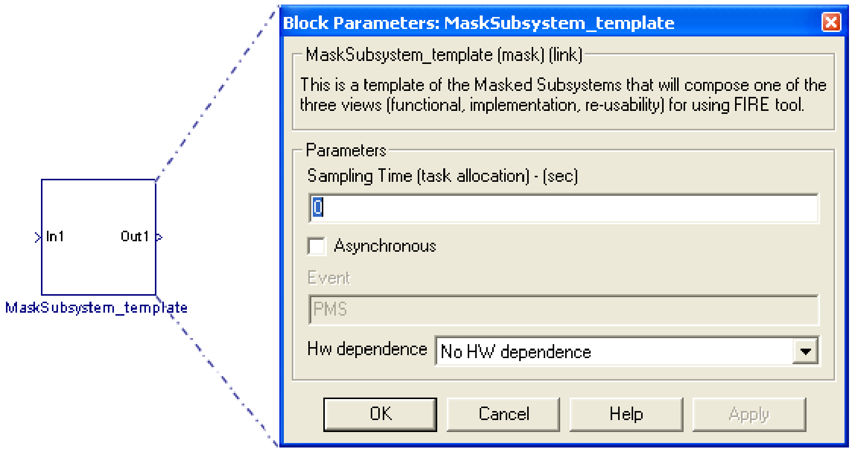

Figure 5.5 MaskSubsystem_template block.

5.3.1. Implementation Aspects

The operation provided by the F.I.R.E. tool is accomplished via a text low-level parsing procedure acting on the Simulink files related to the functional view. The parsing is undertaken by means of some MATLAB® scripts which analyze and re-write these files on the basis of the desired transformation chosen by the user.

Each F.I.R.E. transformation is composed by two different sections: a low-level one where the parameters characterizing the new view to build up are extracted from the functional view masked subsystems, and a high-level one, in which the new file is completed with the aid of MATLAB® *Simulink Model Construction* commands, using the information extracted during the low-level phase. The customer has to set task priorities, used by the scheduler block in the implementation view to solve task activation conflicts, by means of a Graphical User Interface.

5.4. EXAMPLES OF F.I.R.E. APPLICATIONS

This software has been tested and validated on two different real control system design problems; a simplified Drive-by-Wire (DBW) [31] and Variable Valve Timing (VVT) [1] units and a real engine management system [32]. Specifically, the F.I.R.E. tool has been used to analyze the timing problems of the control system and to find a correct tasks allocation.

The goal of the first case study is to get a controlled step response in terms of settling time and overshoot, both for first and second order models, which respectively represent their simplified behaviours. In the functional view, the control engineer designs the control algorithm by tuning the controller parameters, like a proportional or an integral gain, but he/she cannot estimate easily the impact of a scheduling choice on the overall control system behaviour. With the help of the F.I.R.E. tool, the tasks allocation can be more easily analyzed. The control engineer decides the timing of the tasks in the functional view, quickly transforms this view in the implementation one and then analyzes the results of different tasks allocation and scheduling choices on the control system responses. Next Figure 5.6 highlights how different task allocations of the same control system can change the control performance. In particular, see how changing the VVT control task activation from a constant 100 ms sampling time (too slow) to an asynchronous TDC synchronized (Top Dead Center)[4] policy reduces both the settling time and the overshoot, and the same happens for the DBW control module.

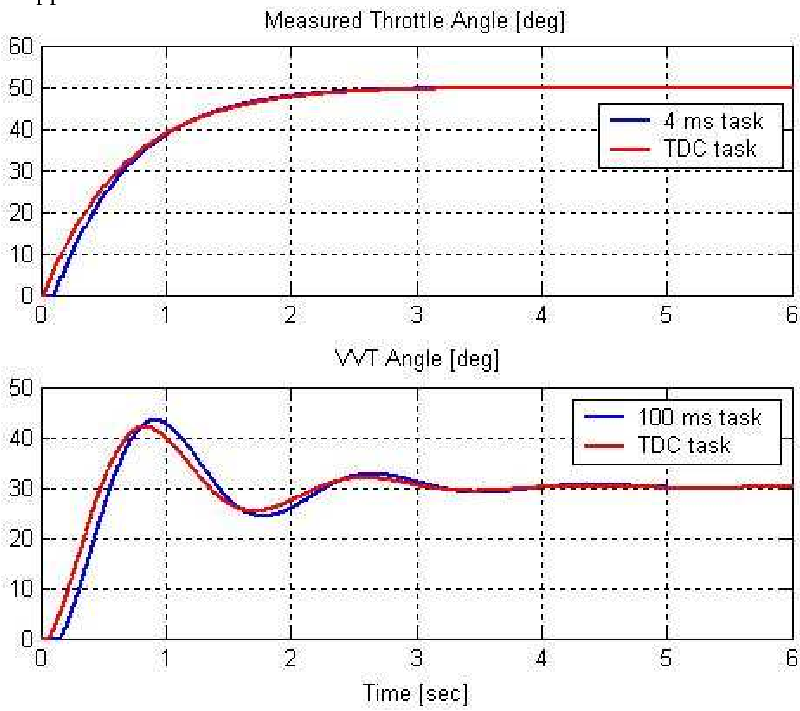

Figure 5.6 Different time slicing on a control system.

From this example is also easy to understand the advantage offered by this tool in autocode generation, because the implementation view provided by F.I.R.E. is ready to be transformed in a TargetLink block diagram.

In the second case study, the F.I.R.E. tool was used for studying the timing behaviour of the overall engine control system before an autocode generation.

In order to carry out such an analysis, four software simulation tests were undertaken in a loop virtual environment, which integrates an engine/vehicle Simulink® model with an Engine Management System model for performing

[4] At an engine speed of 2000 rpm TDC event occurs each 15 ms.

both MIL (Model In the Loop) and HIL (Hardware In the Loop) simulation studies. In particular, the following tests have been carried out:

- Air Conditioner Test
- Misfire Test
- Tip In/Tip Out Test
- Efficient Catalyst Test

The relevance of the F.I.R.E. tool in allowing multi-rate simulations to be easily performed before generating the code and performing HIL simulations, can be judged by comparing the simulation results achieved by the Air Conditioner Test, when the same EMS model is described either by the functional (Figure 5.7) or by the implementation (Figures 8 and 9) views.

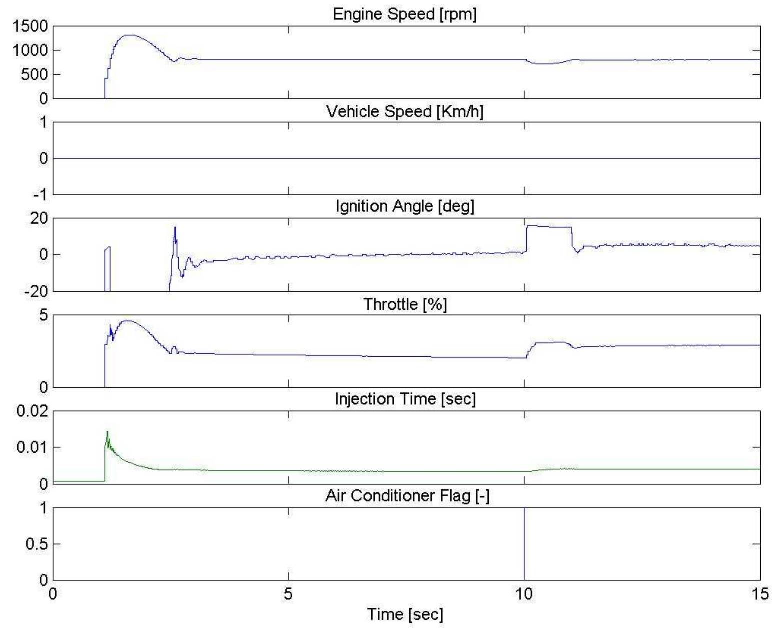

Figure 5.7 Air Conditioner Test with ECU's functional view

In fact, although the idle speed control in Figure 5.7seems to perform well, the system is not stable and in fact there are oscillations in the engine speed (Figure 5.8) because the erroneous task allocation chosen results in a huge number of scheduling constraints and task execution significant delays.

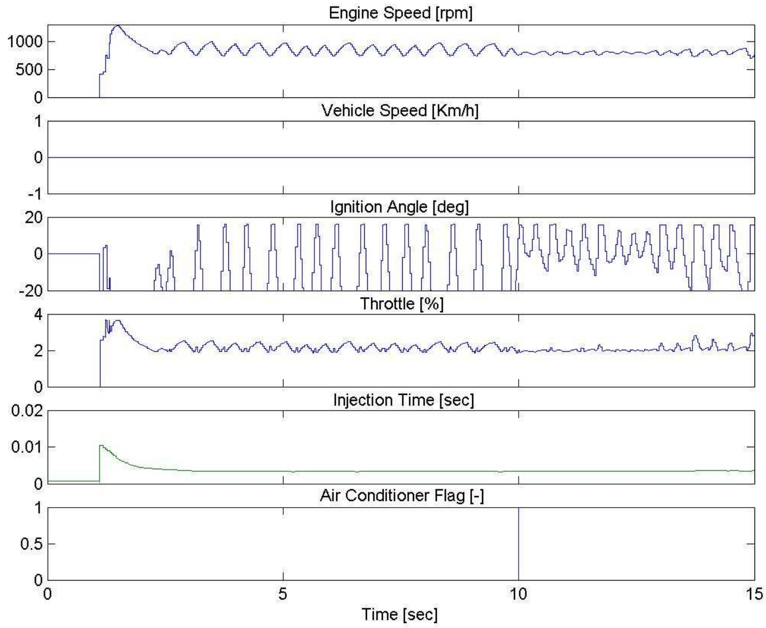

Figure 5.8 Air Conditioner Test with ECU's implementation view

However, repeating the simulation with a new allocation tasks (Figure 5.9), in which the idle speed control task allocation is changed from 100 ms to TDC[5], the control responses become indistinguishable from the prescribed nominal ones reported in Figure 5.7. The repeated simulations have been made possible, before the final production code generation, by executing another *Functional to Implementation* transformation by means of the F.I.R.E. tool. The simulation results obtained with the ECU's re-usability view do not differ significantly from the ones achieved with the ECU's functional one, because that view consists only in a graphical reorganization of the blocks, without any semantic change in the overall model.

[5] **At an engine speed of 800 rpm TDC event occurs each about 40 ms.**

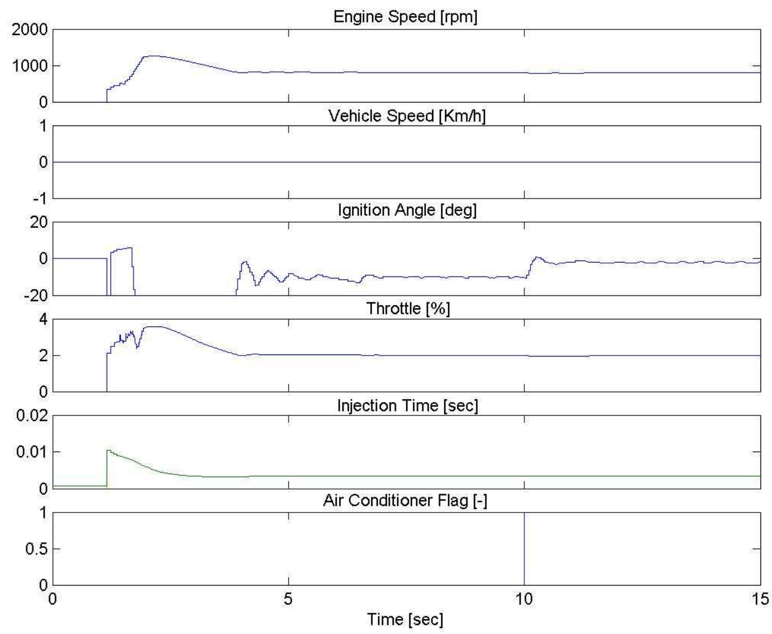

Figure 5.9 Air Conditioner Test with ECU's implementation view with a new task allocation

5.5. BASIC ENGINE CALIBRATION OBJECTIVES

Once the algorithm has been implemented and tested, it is possible to perform an accurate model calibration by using advanced calibration tools integrated in the F.I.R.E. environment. The parameters calibration of the basic-engine control algorithms consists of the identification of the parameter values which, codified in maps and vectors, best describe the engine behaviour in a defined working range.

- The calibration process of a single algorithm (process) can be divided in:
- Bench experimental test design and execution
- Data analysis and algorithm calibration
- Bench test verification

These phases are usually repeated until the target precision is reached. Data analysis consists in the transformation of the experimental results in maps and vectors which will be used to describe the behaviour of the engine in the ECU software.

The entire process can be speeded up by using statistical techniques which reduce the number of experiments to undertake and maximize the informative contents of each test. Using the proposed advanced calibration techniques, again implemented in the MATLAB® environment, it is possible to automatically transform test information in calibration maps directly usable by the ECU. The higher precision achievable, especially when dealing with multidimensional actuation (throttle body, cam phaser for intake or exhaust camshaft, fuel injectors…), correspondingly reflects on better control performance (lower fuel consumption, larger maximum power and so on) and on a more robust characterization of the phenomena for the whole engine family, not only for the "tested engine". This is also important for reducing the influence of noise factors on the engine performance and the generation of diagnostic false alarms. Moreover, the availability of automatic calibration tools speeds up the development process of new engine control algorithms ([1]).

5.6. GENERAL PURPOSE CALIBRATION TOOLS

In a model-based software development process, the simulation models of the processes are often available. A complete collection of general purpose calibration tools has been developed in the MATLAB® /Simulink® environment. These tools are described below.

5.6.1. Continuous multivariable non-linear regression models

In order to describe a relationship of the type z=f(x,y), it is possible to use a regression model which relates the experimental points having x, y and z coordinates. The model is imposed by the ECU algorithms and/or by the physics underlying the process.

In Figure 5.10, the coloured surface represents the volumetric efficiency of the engine, at a defined speed, which depends by the manifold pressure and cam phaser position. This surface minimizes the mean percentage square error from the experimental points represented by blue circles.

The regression model is quadratic in the cam phaser variable and linear in the manifold pressure until the breaking pressure, usually at a value of 950 mbar, is reached whereas it has a quadratic relationship over that pressure value. This switching regressor describes the natural supercharge effect ([2]and [34]), which is particularly evident at 2700-3300 rpm in this real engine application. The traditional linear regressors produce an error up to 8% in the air charge estimation at full load, worsening other actuations like spark advance and mixture title.

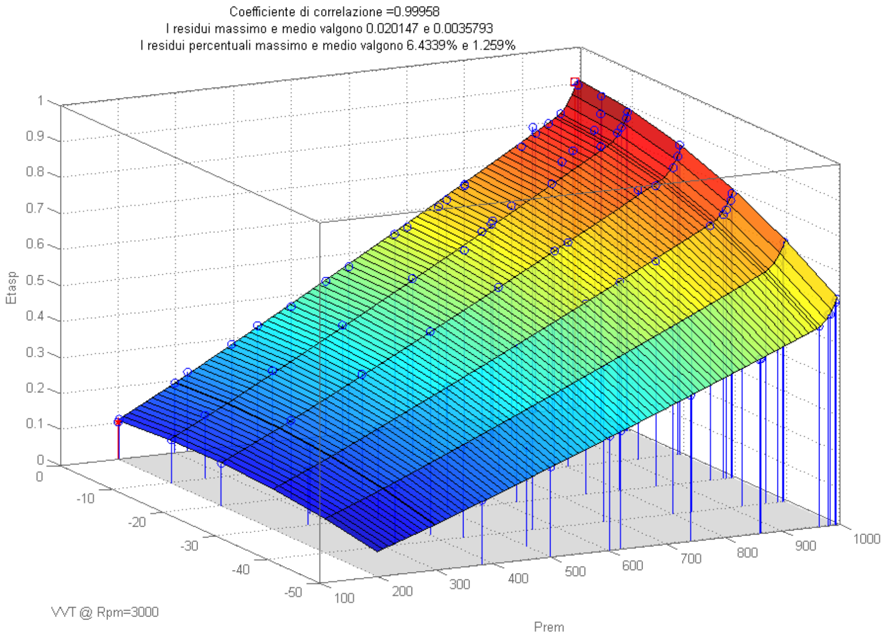

Figure 5.10 Multivariable switching regression model example

5.6.2. Discrete regression model

To describe the relationship of the type z=f(x,y), is possible to use a map which represents the z value for every point on a discrete x-y grid, defined by two breakpoint vectors. The output value of the map in the points that do not belong to this grid, is calculated using a bilinear interpolation method, like in the Engine Control Unit.

The *Discrete Regression* tool has been developed to meet this need. It computes the values of the map that minimize the mean square error between the experimental data and the surface, described as the bilinear interpolation of the map. In Figure 5.10, an example is reported. The experimental points are plotted as red dots, while the map is represented in transparent blue.

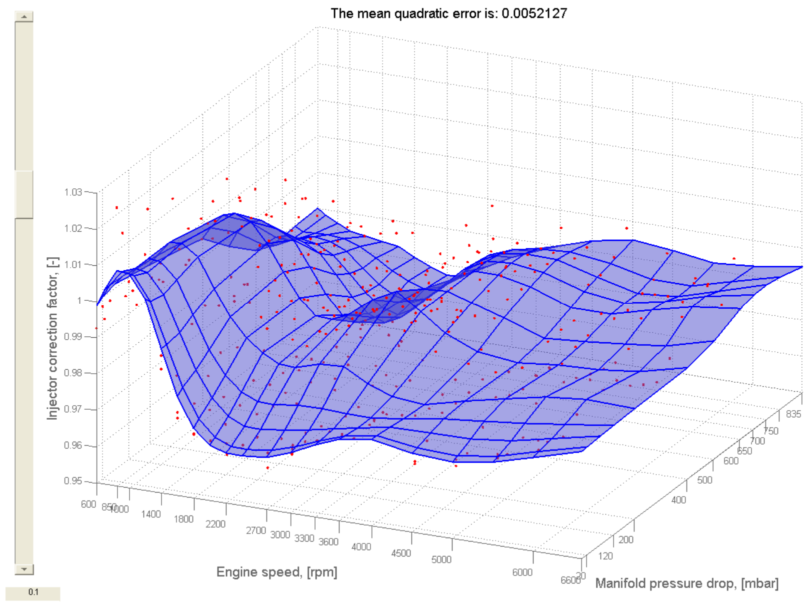

Figure 5.11 Discrete Regression tool example

On the left side of the Figure 5.11, there is a slider that increases the surface stiffness. It works like a spring which stretches the surface edges. A high stiffness generates a plain surface, increasing the mean square error. The map continuity is one of the requirements in the calibration phase. The discontinuities can be caused by a large measurement error in some points. A discontinuous map can generate different output values for very similar input values. This is dangerous in the use of the map, because an error in the input variable or its estimation can generate a big variation on its output, causing instability in the control loop. The stiffness can also be used to impose a rule for the extrapolation of the map in not experimented x and y coordinates.

The discrete regression allows one to be free in the acquisition dataset choice, because it is not any longer necessary, for example, to acquire data exactly in the breakpoint intersection points. It doesn't need to split the problem into sub problems as with the traditional calibration tools.

The advantages are essentially the same of the continuous regression model with the addition of the following ones:

- The model implemented in the ECU is often the output of a map. By acquiring data not only in breakpoint intersections, it is possible to obtain a map which minimizes also the model error, not only the measurement one. This aspect is fully explained below.
- The possibility to interactively stiff the map produces more physically realistic input/output relationships thus avoiding the overfitting problem.

5.6.3. Discrete regression algorithm

The aim of the discrete regression algorithm is to minimize the sum of the squares of the distances amongst the map and the experimental points. In order to achieve this goal, an analogy with a mechanical phenomenon can be done: every experimental point is fixed in the space while the map can slide along z-axis like shown in Figure 5.12. Every experimental point, which is red coloured in the figure, is linked by a spring, in blue, whose stiffness is the same for each point. A damper is also present. The damping value may be critical. The map segments, surfaces in a three dimensional space, have a mass and react to the spring and damper forces according to the second dynamic principle. The forces move the map toward the points. The equilibrium will be reached in the map along a configuration that minimizes the energy of the spring system. The expression of this energy is:

$$E = \sum_{i}^{N} K * \Delta z_i^2 \qquad \textbf{(1)}$$

Where: K is the stiffness of the spring

Δz_i is the z-axis distance between the i^{th} experimental point and the map

N is the number of experimental points

Such a formula represents the sum of the square errors between the experimental points and the map.

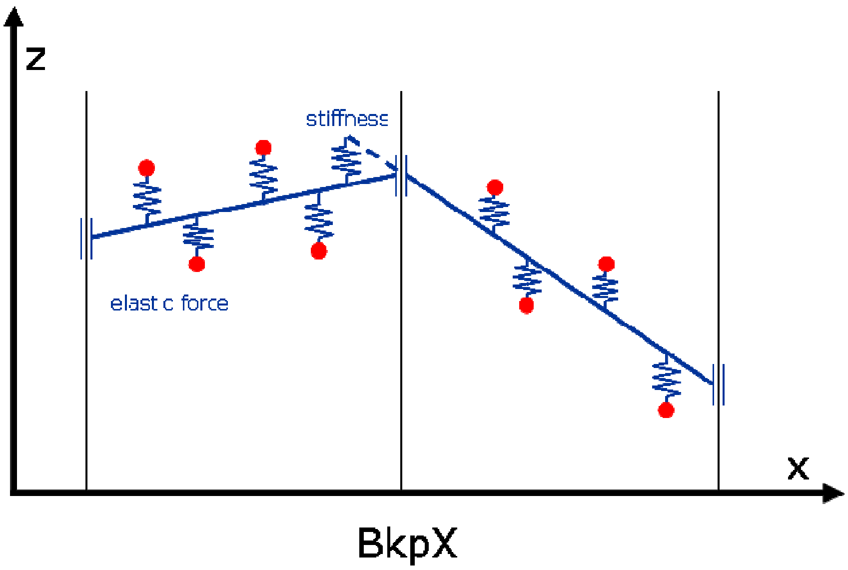

Figure 5.12 DiscreteRegression explanation

The mechanical system dynamic has been discretized and MATLAB® coded with a discrete difference equation. Some programming technicalities have been used to avoid possible instabilities and to enhance the simulation speed. As a result, over 200.000 points can be handled in a few minutes.. Moreover, the outgoing maps are smoother, more coherent with the description of the physical phenomenon and can be interactively judged and modified by the calibration engineer.

5.6.4. Multi Map Optimization

A MATLAB® / Simulink® model of the algorithm has been implemented. This model, fed by experimental acquisitions, produces an output that depends on the calibration requirements. The multi-map optimization changes the calibration results in several aspects. It changes the values of the vectors and scalars, elements of the maps, until the minimum mean square error or mean percentage square error between measured output and estimated ones is reached.

The implemented optimization method is based on a steepest descent algorithm (see [40], [41] and [42]). The algorithm is a local optimization procedure, but some tricks are used to reduce the probability of getting stuck in a local minimum. The multi-map optimization is extremely fast and can optimize, on a 1.6 GHz Intel Centrino processor, for example, three 12x21 maps, in almost ten minutes.

During the elaboration, the optimized maps are shown, see Figure 5.14, together with predicted vs observed graphs and some statistics, e.g. the mean square error.

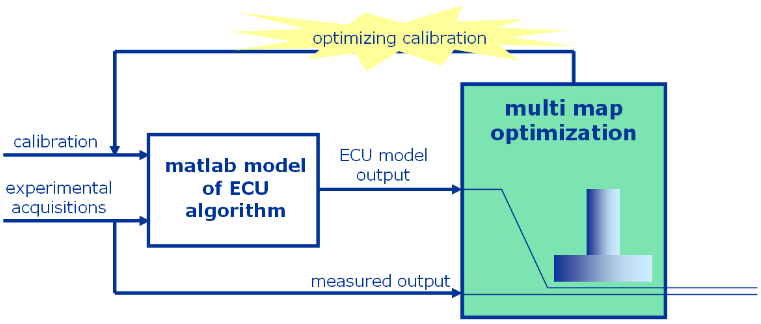

Figure 5.13 Multi map optimization working scheme

The accuracy of the result is up to 4 times better than the traditional techniques. The maps are also smoother because the optimization algorithm is instructed to pick, amongst many solutions with the same error, the smoothest.

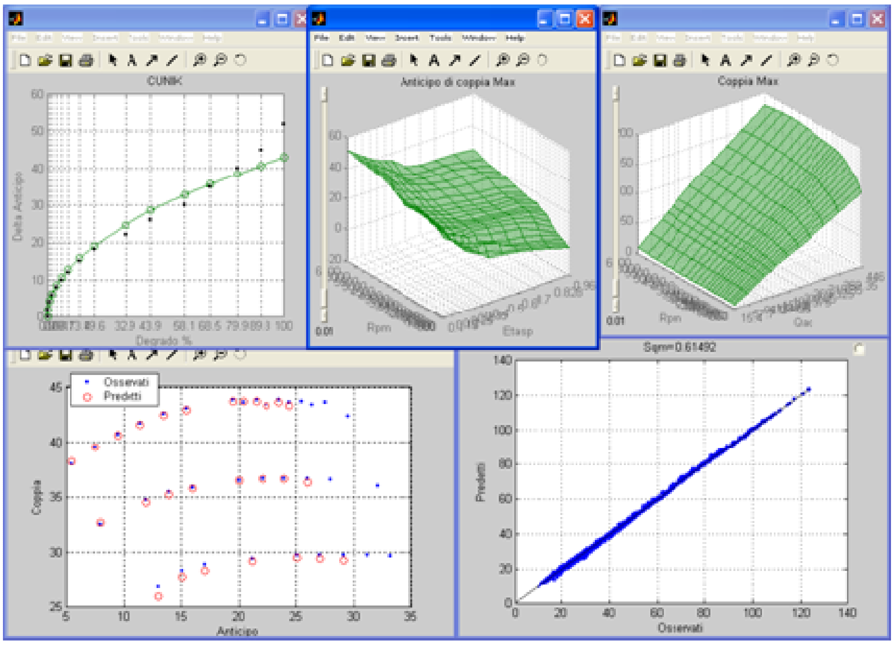

Figure 5.14 Multimap optimization example

In the past, once a first calibration result was achieved over a map, if the breakpoints changed for some reasons no automatic ways to reuse the old calibration were available and only a time-consuming manual retuning, usually undertaken by using Excel sheets, was possible. On the contrary, the proposed multi-map optimization approach frees one from this limitation.

Other advantages include:

- Many maps can be simultaneously optimized. The solution minimizes the total error, not the error of a single map so the results are usually better.
- The accuracy and the continuity of the maps are better than those resulting from traditional methods.

Tunable algorithms

The described optimization techniques can be used mainly to calibrate automotive estimation algorithms. These algorithms are, as a first step, validated through specific sensor measurement by means of test bench instrumentation, usually not present in commercial automotive engines.

The algorithms have to be memory-less, like e.g. the torque estimation algorithm. It is difficult to use the above mentioned optimization techniques for dynamic algorithms, like gas temperature estimation; for these dynamic algorithms, only the stationary part is calibrated using multi map optimization, while the dynamic part has to be calibrated in a different way.

It is furthermore impossible to calibrate algorithms which require the control of a non simulated variable, e.g. the mixture title for the calibration of wall wetting compensation strategy, where the wall wetting is the fuel quantity lying into manifold walls next to the injector that will contribute to combustion process in runtime. In this case, the verification tools presented below can anyway help the calibration process.

Performance measurement

The optimized maps are usually evaluated by computing the mean square error, denoted hereafter with σ, of the predicted values related to the experimental ones: the lower the error, the better the representation of the experimental points given by the maps. The σ statistic, multiplied by 3, is a good estimation of the experimental data variation range around the surface described by the map. To be more precise, the experimental data fall in the described range with a probability of 99.97% in the hypothesis of normal error distribution.

For similar applications, the percentage mean square error is more interesting. It is obtained by dividing the square error with the experimental one and by multiplying it by 100. In this way the statistic is more correlated with the final performance; e.g. in the air charge estimation, the percentage mean square error is correlated with the error on the mixture title actuation.

The error of an estimation algorithm can be divided in two parts: model error and measurement error. The model error is caused by the utilization of a model which is not enough complex to describe the examined phenomenon. It can be reduced by using a different model equation, increasing the number or changing the breakpoints of a map.

The measurement error is due to the limited precision of the data acquisition and elaboration instruments. An experimental data measured with a value which is far from the regression model, generally denoted as an outlier, has to be found, eliminated and, if possible, measured again, to enhance the model precision. As a result, the optimized maps are then more robust against measurement errors than the ones manually optimized, where the values are exactly the ones measured at each specific breakpoint. This statement, based on the Chebyshev inequality [40], justifies the improvement in the calibration quality achieved by using multivariable regression models, which are more robust against measurement errors than those based on a single variable. By forcing the continuity on the dependant variables in more dimensions, the resulting model depends on more experimental points, reducing the probability of interpreting a measurement error as a phenomenon.

5.7. CALIBRATION PERFORMANCE VERIFICATION TOOL

For each calibration problem, a specific tool has been developed which graphically and numerically shows the effects of a calibration variation on the engine control algorithm accuracy. This tool is useful to rapidly verify that the calibration agrees with the imposed criteria and to make a final fine tuning, following the calibration engineer experience. The result of an automatic calibration process has to be verified by an expert for two reasons:

The calibration tool, if there are missing data, extrapolates the behaviour of nearest points. An expert calibrator can do this task better, using his/her knowledge of the phenomenon.

The calibration tool optimizes with respect to a variable that is usually the most important, but in some operating conditions, it may not be the only variable of interest. The calibration engineer can correct the optimization process in such cases; with a user-friendly graphical interface this task is speeded up and the use of bench tests diminished.

In Figure 5.15, it is shown the main screen of the spark advance and torque estimation tool, used for the calibration of a real engine. The main graph shows the so called “umbrella” curves, which show the mean torque depending on the spark advance at a predefined engine speeds and cam phaser positions.

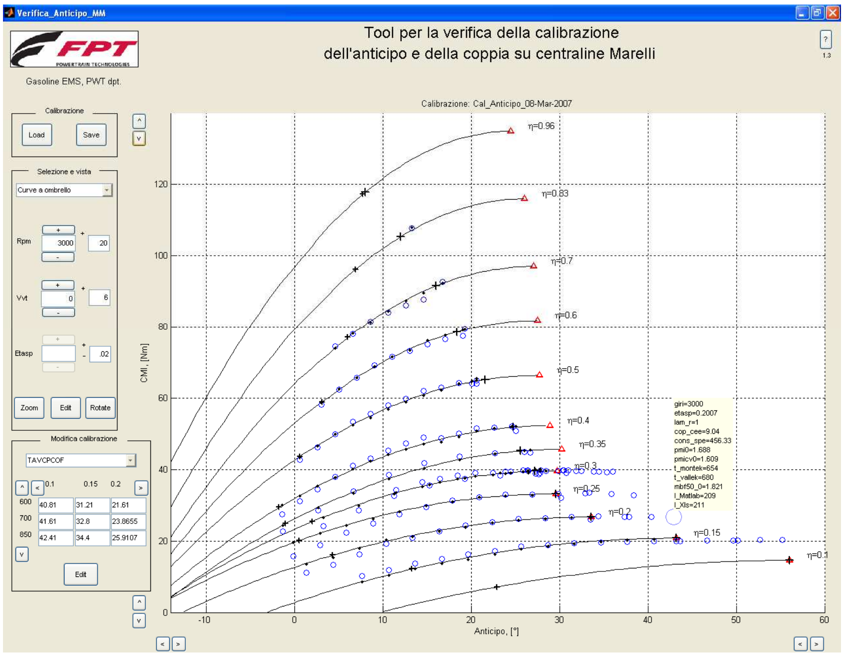

Figure 5.15 Torque interface verification tool

Standard files

To make easy the information exchange, some standard file formats and contents have been defined.

Engine bench data

The engine bench data file is generated at the bench test. It is an Excel sheet that contains, in the first row, the names of the acquired variables while in the second row there are the measurement units, and, from the third row, the acquired data values. Every row represents an operating stationary point, characterized by the measure of input-output values after a stabilizing time. The measure lasts for a predetermined amount of time. All the acquisition process can be automated.

Usually the data are acquired in a first phase for calibration purposes and in a second phase for calibration verification.

Calibration

The calibration file has to contain information about the ECU parameters related to the calibrating algorithm.

5.8. DEVELOPED TOOLS: APPLICATION TO A REAL ENGINE CALIBRATION

The following tools have been developed in the MATLAB®7 / Simulink environment. They are currently used by the calibration engineers for real engine applications.

5.8.1. Air charge estimation

Before the introduction of the above described tools, the experimental plan took 15 days (24 hours per day). This time effort had to be multiplied by the number of significant changes in the hardware during the calibration phase, which are usually up to four. The data analysis requires almost 5 days for each plan. The resulting maps present a significant number of discontinuities and the estimation error is not satisfactory.

The calibration tools were introduced in the development process at the start of one real calibration phase. The experimental plan time has been halved, now taking 8 days, 24 hours a day. The data analysis requires now a few hours. The estimation error is almost halved and it is now fully satisfactory.

Automatic calibration - The criteria exploited by the calibration engineers have been implemented in this tool. The synergies between the calibration department and the engine test department produce the continuous enhancement of the automatic calibration tools.

The inputs are:

- A calibration file, to gather information about the dimensions and the breakpoints to be used in the maps calibration.
- One or more engine bench test files containing the necessary channels.

For each engine speed breakpoint, a multivariable switching regression model is calculated (Figure 5.16) to best describe the manifold pressure - cam phaser position - volumetric efficiency relationship. The used model has been developed by taking in account the ECU reproducibility of the relationship and the physical behaviour of the phenomenon. The volumetric efficiency depends squarely by the cam phaser position, linearly by manifold pressure. Over the breaking pressure, which is a regression parameter, the linear dependency becomes quadratic without first order derivative discontinuity, while, under 300 mbar the slope can change to best fit the data.

The regression model has been then made discrete and transformed, without information loss, in the ECU algorithm model, made by 4 maps, for a total of 1092 parameters

Calibration verification - This software allows one to graphically and numerically visualize the relationship between experimental points and corresponding ECU estimated points. The working algorithm can be explored with the following points of view:

- Iso intake manifold pressure efficiency curves
- Iso cam phaser position efficiency curves
- 3D surface of volumetric efficiency depending on pressure and cam phaser
- Iso volumetric efficiency curves depending on intake manifold pressure and cam phaser
- Percentage error of the total air charge estimation

It is possible to modify the maps, graphically or numerically, interactively for verifying the effect in one of the possible views.

A left mouse click on an experimental point shows additional information, while a right mouse click opens a contextual menu which permits to eliminate the point or open the source data file, highlighting the corresponding row.

By opening multiple instances of the tool, it is possible to compare different calibrations or different bench test data, speeding up the verification and refinement of the calibration.

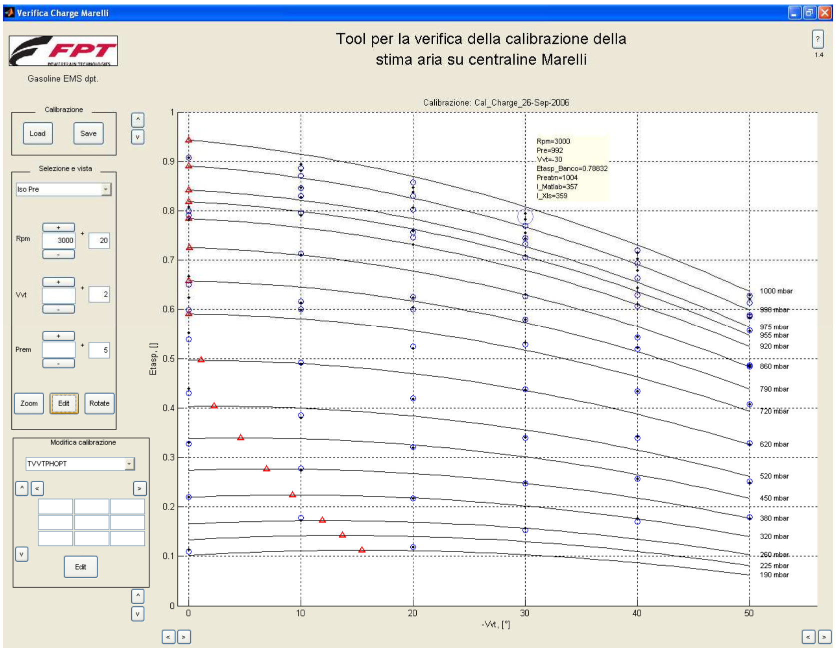

Figure 5.16 Charge estimation calibration verification tool. Inlet efficiency curves depending on cam phaser position, at defined intake manifold pressures

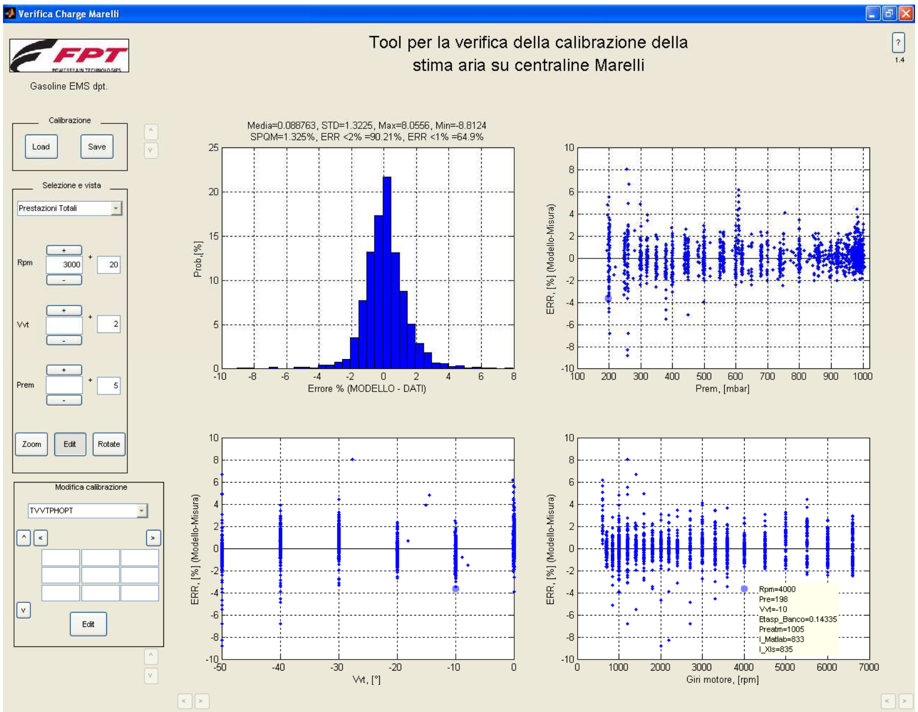

Figure 5.17 Air charge estimation calibration verification tool. Total performance, statistics.

Performance measurement - This tool measures the total performance, in estimation accuracy sense, of the pair [Calibration – Engine bench test data]. It is possible to compare calibrations from different sources, to rapidly find critical points, and to evaluate different versions of the algorithm. This function is integrated also in the air charge calibration verification tool.

5.8.2. *Gasoline injector model*

Automatic calibration - This tool automatically calibrates the gasoline injector model. It uses the same engine bench test data used for charge estimation. The discrete regression tool easily calculates the map which minimizes the error between the injected gasoline estimation, done by the ECU, and the measured one.

5.8.3. *Spark advance calculation and torque estimation*

Automatic calibration - The torque supplied by the engine is estimated using mainly the engine speed, air inlet efficiency, cam phaser position and spark advance ([13] and [14]). In an engine without the cam phaser, the algorithm uses only two maps and one vector:

- MTA(speed, eta), called the maximum torque advance map: it describes, for each engine speed - air inlet efficiency point, the spark advance that maximizes the torque. If the detonation occurs before reaching the real maximum, an extrapolated value is used to best fit the data. In Figure 5.15, the x-coordinate of the red triangles represents the maximum torque.
- MT(speed, eta), called the maximum torque map: it describes the indicated torque measured at the maximum torque advance. In it is represented by the y-coordinate of the red triangles.
- UC(advance - MTA(speed, eta)), called the unique curve: it describes how the distance between the spark advance and the maximum torque spark advance reduces the torque. Its output is 1 if the input is 0. The output decreases while the input difference increases. It is very similar to a parabolic curve and it has the property to fit well the experimental data in the equation

- TORQUE = MT(*speed*, *eta*) * UC(*advance* – MTA(*speed*, *eta*)) (2)

  Where:
  - *speed* is the engine speed
  - *eta* is the air inlet efficiency
  - *advance* is the actuated spark advance

The hyper surface that describes the torque delivery has to be continuous, because it describes a physical phenomenon. The multi-map optimization automatically finds the values of the maps which best fit the experimental data. The result is very continuous. The error on the torque estimation, 0.61 Nm, is almost a quarter of the error obtained with traditional calibration methods, that calculated the maximum torque spark advance by analyzing only the points at the same engine speed and air inlet efficiency, thus resulting extremely sensitive to experimental errors.

By using different implementations of the multi-map optimization and discrete regression, 10 maps can be calibrated, taking into account the cam phaser position dependencies, for a total of 2284 parameters.

<u>Calibration verification</u> - This tool assists the calibration engineer in the verification of the correctness of the calibration of the spark advance calculation and torque estimation. The inputs are:

- A calibration file, in Excel format, generated by the calibration tool
- The engine bench test data

The main graph shows the trend of the “umbrella” curves, CMI(advance), at a fixed engine speed and cam phaser position. Acquired data are represented by blue circles, while calibration estimated corresponding points are the black dots. The red triangle represents the maximum torque point for each load breakpoint, while the big black plus symbol is the working spark advance. Clicking over an experimental point, additional information are shown, like fuel consumption, temperatures and so on. Clicking over an ECU calculated point, instead, the relative formula adopted will be shown, with input maps values.

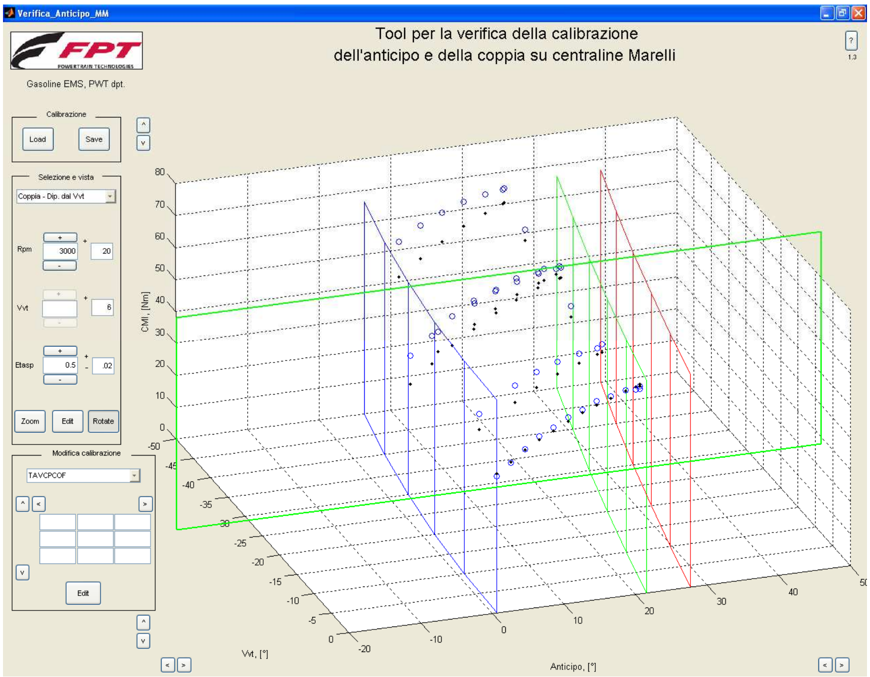

Figure 5.18 Torque verification tool, cam phaser position dependency

On the left panel it is possible to choose the engine speed, the cam phaser position and the load. The cam phaser can be specified with the *obj* string, to select only the points at objective cam phaser position, characterizing the steady state operating points. On the bottom left side it is possible to interactively modify the maps, numerically or graphically and to control the effect in the main graph.

In Figure 5.18, the CMI (advance, cam phaser position) graph is shown, which is of use to verify the phenomena from another point of view.

Other three views are available, to verify that the working advance actuation is correct for every cam phaser position. The modified calibration can be exported in Excel format, ready to be copied in ECU software.

5.9. CONCLUSION

In this chapter an analysis and simulation powerful tool for embedded control system design has been presented. A discussion regarding the related automotive industry demands and the relevance of software support tools for control system design has been provided. From these considerations, F.I.R.E. has been built up to provide multiple views of a Simulink® model, because control system developers have to consider different aspects, not all visible with a single graphical notation. It mainly allows a graphical reorganization and an early timing analysis of the control system, according to the Auto-Code Generator requirements, with the objective of automatically generating the production code for the ECU directly from these models.

This software has been tested and validated on real applications and the relevance of this tool has been illustrated by investigating different closed-loop system behaviours which result from bad scheduling choices and by the tool assistance in finding the most correct time slicing for each task. Future developments include:

- automatic discretization of continuous-time blocks in the Functional to Implementation transformation;
- joint comparison of simulation results between functional and implementation views;
- inverse transformations and overall consistency checks amongst the views;
- explicit specification of the execution order of subtasks within each task in the Functional to Implementation transformation.

Another important feature which can be added are the possibility to undertake more detailed investigations and comparisons between different dynamic scheduling policies[6] (e.g. priority-based preemptive scheduling and Earliest Deadline First (EDF) scheduling) and a deeper analysis about the synchronization amongst tasks that use shared data for their computation. It could be also interesting to investigate the delay due to common data access and the transmission rate into the network by integrating the F.I.R.E. tool with handmade scripts or other blocks representing different schedulers, networks and monitors for synchronization, as in the TrueTime platform.

On the other hand, the integration of this environment with accurate calibration instruments guarantees the Engine Management System product quality, that is more and more influencing the costs and it has to be accurately managed since the first phases of development. In order to take under control many strictly, even conflicting, requirements the development environment has to allow a strong integration between every single phase. This satisfaction has a big impact on the characteristics of the tools to be used and on the necessary skills of people involved in development. An important goal achieved is that of being able to merge the skills coming from different engineering departments (i.e. engine application, engine control system development) and experiences from different projects to guarantee in a predictable way the required product requirements in future projects.

[6] Simulink® simulation engine uses a fixed-priority Rate Monotonic (RM) scheduling to simulate multitasking models.

Chapter 6 Control system SW verification

The verification and validation process of an embedded system has been always a crucial aspect in the overall development cycle. This consideration is particularly true for engine control systems for which the complexity is more and more increasing and, on the other hand, the time to market is on the contrary more and more decreasing.

The use of tools able to accelerate the single verification phases is really opportune and its integration within the overall development process has to be strongly encouraged. The development process for Engine Management Systems consists of several phases each one of which is related to engine development one. During these phases an incremental approach for SW development is used. So a particular focus has to be given to the non regression of next SW version with respect to previous ones: the absence of undesirable impact of new requirements on the rest of SW has to be verified. In this chapter the non regression test activity and the corresponding tools have been detailed

This chapter describes a process for SW verification and validation. The described process is based on a very simple idea: every SW change has to correspond to a new incremental requirement. From this idea a very simple, but on the other hand, very powerful and effective tool has been proposed. The application of this tool is able to reduce the automotive development lifecycle and to maintain high quality products. The use of this methodology is sometimes obstructed by the necessity of a high experience and knowledge level, but the use of this process has shown all the benefits presented above and represents a beginning point to automate the process of SW verification and validation

6.1. SW TEST ACTIVITIES

The product quality is more and more influencing the costs and so it must be accurately managed since the first phases of development. In order to take under control the big amount of “variables” coming from very high system complexity and many different requirements and constraints to be met, the development process has to provide a strong integration between every single phase. This requirement has a big impact on the characteristics of tools to be used and on the necessary skills of people involved in development . Actually a discontinuity in the development process can cause a lack of time and a loss of control of product quality. This implies a necessity of a systematic cooperation between developers in different phases and a systematic exchange of information. This sharing activity is possible only if the development environment is able to manage all the product development phases. Another important issue is the systematic use of “lesson learned” in order to predict errors that have been only corrected in the past. This issue imposes the definition and use of a database of problems, tricks and all information regarding problems faced in the previous projects.

Many paper have been described the necessity of new tools for SW testing and more in general for ECU development ([21], [25], [32] and [43])

In this chapter the testing phase of engine management system development is described and a new tool for a particular sub-phase here called *Non Regression Test* is presented.

The Testing phase can be divided in tree main steps:

- module testing
- integration testing
- functional testing

6.1.1. Module testing

The module testing (also called unit testing) is related to a single SW module (function) and has the objective to verify the correctness of SW implementation. It is in particular useful when auto-code generation tools are used, with which the code is automatically generated by an algorithm modelling (i.e. a Simulink model can be used with Target Link auto code generator). In this case a comparison between the algorithm model simulation results and signals coming from the EMS can be easily and quickly performed. The results of this kind of testing is, for example, implementation errors like overflow, underflow, round off or non alignment between algorithms models and EMS code.

6.1.2. Integration testing

The integration testing has the target to verify that the singly tested module is correctly integrated in the overall SW architecture. This kind of test has the objective to verify that the new algorithms or modifications to the existing ones do not affect undesirably the behavior of other modules. A part of this activity is the verification that the SW is not regressed compared with the previous version. This kind of test is called *Non Regression Test* and will be described in details in the next section.

6.1.3. Functional testing

The functional testing (also called validation phase) is the final testing activity that has the target to validate the algorithms with the respect to functional requirements and it has to be carried out normally after the calibration phase. For this kind of testing activity specific tools have been developed (see Chapter 5).

6.2. NON REGRESSION TEST ACTIVITY

The Non Regression testing, on which this chapter is focused, is a part of integration testing and it consists of following steps:

1. benchmark SW definition;
2. selection of test maneuvers from Data Base and definition of the engine parameters to be monitored;
3. execution of selected maneuvers on benchmark SW;
4. execution of selected maneuvers on SW under verification;
5. post processing of the executed tests in order to identify the differences between the behavior of the SW under verification compared with the benchmark one;
6. analysis of differences between benchmark SW and the one under verification, differences can be
7. desired, if caused by new requirements added in this release of SW, or
8. not desired, if not related to new requirements, in this case a regression of SW can be possible.
9. SW delivery

In the following the listed phases will be detailed.

6.2.1. Benchmark SW definition

The definition of benchmark SW is really important because it represents the baseline which the new SW release has to be compared with. The previous version of SW is normally used as benchmark. In this case the possible differences must be related to new requirements effects.

6.2.2. Selection of test maneuvers

The second phase is the selection of test maneuvers from Data Base and the definition of engine parameters to be monitored. A data base has been developed in which the maneuvers, belonging to the following categories, have been included:

- all the ones that have highlighted problems on previous projects;
- all the ones that come from the field experience;
- standard and legislative ones;
- maneuvers that affect specific functions (i.e. diagnosis ones).

Some examples of maneuvers contained in database are:

- driving cycle derived by ECE-EUDC cycle;

- cold temperature cranking;
- take off with big undershoot;
- maximum engine speed overtaking.

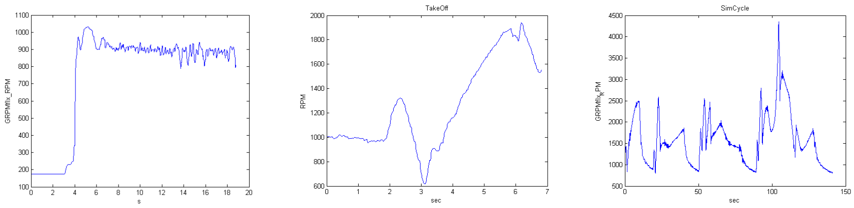

Figure 6.1 Example of time-history acquired by car (RPM channel): Cranking, TakeOff, SimCycle.

6.2.3. Execution of selected maneuvers on benchmark SW

The third phase consists of the execution of the selected maneuvers with the benchmark SW. As said before, it is required that the SW under verification has the identical behavior of benchmark one. Every difference has to be accurately traced with the respect to new requirements.

6.2.4. Execution of selected maneuvers on SW under verification

In the fourth phase the same maneuvers have to be reproduced on the SW under verification. These tests have to be led using the same input used in the previous phase and starting from the same initial conditions.

6.2.5. Post processing

The fifth phase consists of the post processing of results of third and forth phases in order to identify automatically the differences between benchmark SW and the one under verification. This phase is supported by a tool developed in Matlab environment, which will be detailed in the following.

6.2.6. Analysis of differences

The sixth phase is the analysis of the differences identified in the previous one. Each difference has to be analyzed in order to understand if it is due by a regression of the SW under verification with the respect to the benchmark, or it is related to a new requirement implemented in the new release of SW. In order to understand the correctness of identified differences, some further tests can be necessary. These last ones have to be specified just for the specific requirements, while the test cases until now realized are not related to specific requirements.

6.2.7. SW delivery

The last phase consists of the conformity declaration of the SW under verification or in the definition of non conformities.

6.3. USE OF SPECIFIC TOOLS TO SUPPORT EACH PHASE

The process described can be adopted if and only if some necessary tools are available. In the section below, a brief overview of the HIL simulator will be given and the post processing tool will be described.

6.3.1. HIL Simulator

The process of verification and validation of an embedded system needs an accurate and repetitive engine and network test bench. One of the used processes for the verification and validation of an embedded system is the Hardware in the loop simulation.

The hardware used for the HIL testing consists mainly of a computer with the GUI software, an *I/O board* dedicated to each signals “to” and “from” the ECU, and an *Processor board* where the physical model is loaded.

The sensors, which are necessary to be sent to the ECU to provide the engine simulation, are generated via an automotive dedicated board which is able to reproduce:

- Water and air temperature sensors via resistor channels
- Manifold, ambient pressure sensors and pedal potentiometer via analogue output
- Crankshaft, camshaft and knock sensor via DSP (custom automotive)
- CAN network via transceiver
- Additional digital signals (e.g., brake switch, clutch…)

The commands of the actuators from the ECU are connected to real load or simulated ones such as resistors, capacities, inductive loads. The simulator reads the commands and send it to the model in a physical value such as time of injection and Ignition angle point. For each applications the set up process must be performed. This process consists of two different parts which are the hardware configuration and the model calibration. For the hardware configuration process, the electrical schema, the sensor and actuators electrical interface and the can network settings are necessary. The input channels for the ECU are output channels for the simulator such as temperature sensor, wheel speed sensor and knock generation. The output channels for the ECU are input for the simulator such as Injectors and ignition coils. The automotive custom board of the simulator provides all the needed signals for the car simulation and can network. In the setup process the model must be provided by all the necessary TA block for the fault injection and the Non regression test.

Why use hardware-in-the-loop simulation?

In many cases, the most effective way to develop an embedded system is to connect the embedded system to the real plant.

In the most part of the cases, HIL simulation is more efficient.

The metric of development and test efficiency is typically a formula that includes the following factors:

1. Cost
2. Duration
3. Safety

Cost of the approach will be a measure of the cost of all tools and effort. The duration of development and test affects the time-to-market for a planned product. The safety factor and duration are typically equated to a cost measure.

The tight development schedules associated with most new automotive programs do not allow embedded system testing to wait for a prototype availability.

In fact, most new development schedules assume that HIL simulation will be used in parallel with the development of the plant. There are different levels of simulator related to the different level of the project.

Simulation is an important method in the design and validation process of complex hardware/software systems like electronic control units (ECU) for automotive applications. Modelling of such systems is a trade-off between high speed, high accuracy and low effort. It is easy to create a model covering two of these three attributes but it is nearly impossible to build high speed and high accuracy models spending only low effort.

6.3.2. Post processing tool

In order to highlight the differences between the two SW (the benchmark one and the one under verification), a specific tool has been developed, to speed up and improve the verification and validation process. The tool allows, through a Graphical User Interface, to choose for every channel, the acceptable range of the error and the type, between absolute or relative (i.e. 10Nm or 10 percent). We have moreover to outline the possibility to acquire signals logged from different sources; that gives the flexibility to use different SW platforms.

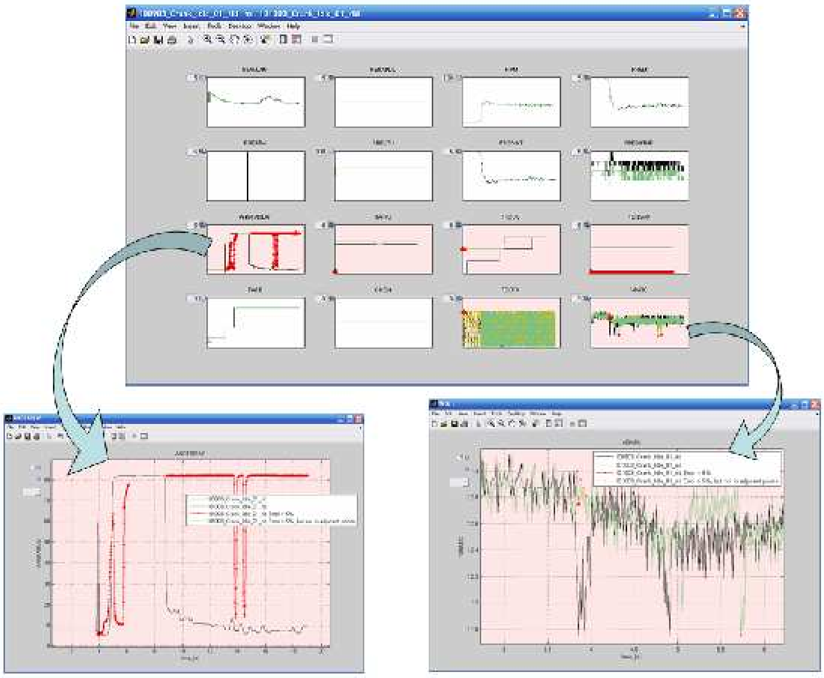

Figure 6.2 Tool interface

The result of the post-processing is a screen like the one shown in Figure 6.2, in which different colors highlight the points where the values of the variable under examinations, coming from two acquisitions of the same maneuvers (one for each SW), differ for more than the preset error.

In particular the tool takes into account also the error "quality", pointing out with a marker color, the errors that are inside the admitted threshold in the next and previous sampling step and so negligible.

After the analysis phase there is the recording of the already done compares. This serves as a history database for future SW analysis. This database of test results is easily accessible to interested people.

6.4. CONCLUSIONS

Scheduled with beginning goals, the new process points out benefits as follows:

- reduction SW validation time (as validation cost);
- maneuvers repeatability;
- possibility to effect test highlighting on fixed SW function;
- easy SW bug detection thanks to the process capability to highlight any SW non conformities.

Implementing new process of verification and validation pointed out also limit into it, to keep in mind during post processing analysis, but at the same time to consider as future development points. In particular we can resume these points as follows:

- often it is necessary to have a further "vehicle test check";
- the supplier has to be able to make the same test as comparison;
- it must be considered that the "noise factor" influenced any signal.

Finally a considerations comes out about the necessary instrumentation to be used. HIL described in 6.3.1, induces to a question: how is it convenient the use of a such powerful tool for reproducing time histories for EMS? This question was the idea that drove to the development of new Hardware In the Loop system that will be discussed into the next chapter.

Chapter 7 Low cost systems for Hardware In the Loop simulation

In-vehicle driving tests for evaluating performance and diagnostic functionalities of the engine control system are often time-consuming, expensive and not reproducible. Using Hardware-In-the-Loop (HIL) simulation approach, new control strategies and diagnostic functions on controller area network (CAN) line can be easily tested in real time, in order to reduce the effort and the cost of the testing phase. Today spark ignition engines are controlled by an electronic control unit with a large number of embedded sensors and actuators. In order to meet the rising demand of lower emissions and fuel consumption, an increasing number of control functions are added.

The use of Hardware In the Loop simulation is a crucial factor in this scenario. On the other hand the availability and cost of such kind of system imposes a thrifty use and it drove the development of a lower cost HIL. This chapter aims at presenting a portable electronic environment system, suited for HIL simulations, in order to test the non-regression software and the diagnostic functionality on CAN line. The performances of the proposed electronic device, named Micro Hardware-In-the-Loop system are finally presented through the testing of the engine management system software of a Fiat gasoline engine with variable valve timing, both for the production and the development version.

7.1. EXTENSIVE TESTING BY MEANS OF HARDWARE IN THE LOOP SIMULATION

The Hardware-In-the-Loop (HIL) simulation methodology is currently recognized as a useful and effective approach in testing the automotive control strategies and diagnostic functionalities. Nowadays the requirements of an Engine Control Unit (ECU) are hard to meet due to the more and more stringent emission normative and ambitious performance in terms of fuel consumption and power request. On the other hand, the high competitiveness among car makers is stressed because of a continuous reduction of time to market and development costs. Therefore, the need of higher efficient methodologies arises, aimed at extensive testing of hardware and software components of the ECU [44], [45]. For this purpose, HIL simulation approach is widely adopted thanks to its compatibility to replace significant portions of test procedures for different control systems. It incorporates hardware components in numerical simulation environment, yielding results with better credibility than pure numerical simulations. HIL experiments run in real time and also make test procedures, which are difficult or even impossible with the real systems, possible. The procedure based on the HIL approach has been widely studied and successfully applied in different engineering fields. In [46] and [47] an efficient real-time HIL testing approach for control design in power electronics applications has been proposed. In particular, in [46] the authors have developed a digital power controller for aerospace applications, validated through hardware-in-the-loop testing using Virtual Test Bed Real-Time (VTB-RT), whereas in [47] the environment system, realized with HIL configuration, has been applied in two power electronic application examples, a boost converter and an H-bridge inverter. Moreover, modelling and simulation through HIL approach has been adopted for improving wind energy system control strategies. In [48], how a newly established real-time HIL test facility can be utilized for wind energy research, is exploited. The test site uses two dynamometers, a variable voltage and frequency converter to emulate a realistic dynamic environment, both from a mechanical and electrical point of view. About the wind energy research, in [49] and [50], an experimental failure diagnosis system based on FPGA-in-the-loop hardware prototyping has been discussed for verifying the performances of the fault tolerant wind energy conversion system.

In this chapter, attention is addressed to the car engine control system to test the new control strategies and the diagnostic functions. Engine control tasks, that were classically solved mechanically, are now being replaced by electronic control systems, and the design and implementation of control and diagnostic algorithms is a crucial element in the development of automotive engine-control systems [5], [51]. Furthermore, the engine system and its components must be constantly monitored in order to comply the exhaust emissions limit specified by international regulations in all driving situations. So, in the last years, the car makers have intensified their actions on innovative functionalities and diagnosis to respect and monitor the emissions related to the components and the whole system. In particular, the development and calibration of ECU functions require a more accurate tuning within the imposed constraints of costs and time to market. Among the simulation methods, HIL testing represents the most common procedure, where a computer with a real time simulation model of the engine or vehicle system is connected to a real ECU in order to test the final embedded software [52], [53] and [54].

The advantages of adopting the HIL system are evident and have been described above, where the main characteristics have been highlighted. On the other side, the main drawback is the necessity to dispose of accurate engine models for real time simulations [55], [56] and [57] in order to be able to describe the dynamics of system components and their interactions. Moreover, the hardware which is necessary to run appropriate simulations is complex to setup and extremely expensive, requiring specific know-how of the people involved in testing.

In this scenario the proposed device, named Micro HIL (MHIL), has been developed to perform those experiments where the commercial HIL simulator is unproductive in terms of time to setup the experiments and costs of the devices. Therefore, the main advantages of the MHIL are the simplicity of its use and the portability [58].

The chapter is organized as follows. Firstly, an overview of the HIL background and the motivation for developing a new system are detailed. The proposed electronic system is then described in section 3. Experimental results about the diagnostic functionalities and software validation are discussed, respectively, in sections 7.4 and 7.5, and finally, conclusions end the chapter.

7.2. BACKGROUND AND MOTIVATION

A typical Hardware-In-the-Loop system is reported in Figure 7.1. Here the engine is modelled using Mat-lab/Simulink software and it is simulated through a dedicated real-time hardware simulator. This provides all electrical signals to fully exercise a real Electronic Control Unit (ECU), connected to the simulator, where the control strategies are running.

The use of dSpace devices for HIL applications is commonly adopted (see, among others, [59] and [60]). In Figure 7.1 is shown the dSpace full-size simulator. The simulator is equipped with a real-time processor board and I/O unit. The processor is a DS1006 Board, equipped with an AMD Opteron Processor at 2.2 GHz. This board computes the model components for engine dynamics simulation. To handle the simulator inputs and outputs, a DS2211 HIL I/O Board is used. This is the standard board for dSPACE HIL applications providing the entire I/O signals for the simulation of a typical 8-cylinders engine, i.e. crankshaft angle synchronous signals, CAN communication, and analogue, digital and PWM I/O, including the necessary signals conditioning. Moreover, a rack where, at the bottom, there is a remote-controlled power supply unit simulates the battery and allows to vary the voltage during the real-time simulation (under voltage and overvoltage tests, voltage drops during engine start, etc.).

Figure 7.1 Typical Hardware-In-the-Loop system from dSPACE.

HIL approach is used by design and test engineers to evaluate and validate components during the development of new control systems [61]. Rather than utilizing them in test bench experiments, the HIL system allows the testing of new components and prototypes while communicating with software models that are simulating the rest of the system. It results in a significant reduction of the costs and time to setup the experiments and it increases the flexibility and efficiency of the testing phase.

In literature, several HIL applications can be found to prove the effectiveness of the methodology [62], [63]. As an example, in [64] an HIL system has been developed to test commercial Antilock Braking System (ABS) and Electronic Stability Program (ESP) ECU. Using the developed HIL system, the performance of a commercial brake ECU were evaluated for a virtual vehicle under various driving conditions. The system has been built in a laboratory providing convenient and reliable means for testing multiple ABS and ESP modules.

The benefits to use HIL system are evident both for time saving and as a consequence in cost saving, and for the performance that it is possible to reach. But, on the other side, the drawbacks are the high costs of the hardware simulator and the elevated skill and know-how necessary to configure and run the complex simulations, including the needs to dispose of accurate mathematical models, sufficiently accurate for the kind of experiments. Therefore, since the majority of the experiments are simple tests regarding the validation of ECU software, i.e. Non Regression Tests and diagnosis functionalities, it makes the HIL system unsuited for this kind of experiments.

Non Regression Test is aimed at ensuring that a modified control software still meets the specifications and it is detailed in 6.2. Conversely, a selective approach is based on the choice of a subset of the test pool that can provide sufficient confidence of the system [65], [66]: a test case will be selected if and only if it executes at least one of the modified functions that influences the behaviour of the program [67], [68]. Moreover, in [69] it is present an overview of the major issues involved in software regression testing, an analysis of the state of the research and the state of the practice in regression testing in both academia and industry, and a discussion of the main open challenges for regression testing software.

Regarding the automotive embedded system, a crucial aspect for the engine development is to avoid regression of new ECU software versions compared with previous ones, i.e. the absence of undesirable impact of new requirements on the part of software unchanged. This consideration is particularly true for engine control systems, where the complexity is increasing and the time to market is decreasing.

Similarly, regarding the diagnosis, the automotive industry has to provide on-board health monitoring capabilities to meet legislated diagnostic requirements for engine management systems. Actually, there is an exhaustive literature about the diagnosis problem, that is analyzed from different point of view [70], [71] and [72].

In particular the real-time diagnostic functionalities of engine management system are devoted to monitor the performance of the electronic throttle body, variable valve timing, injectors and ignition control systems and to detect and identify a suite of anomalies.

So the need to dispose of a device able to automatically perform this kind of tests. Again, this device should be easy to use and fast to setup. Consequently, it must not be based on mathematical models but rather then on measurements so to generate the correct signals to provide to the ECU in order to verify the embedded algorithm.

Finally, the solution proposed in this chapter cannot replace an expensive and complex HIL simulator, as those described above, but it is a low cost alternative for common and simpler applications, as regressive test and diagnostic functionalities tests.

7.3. MICRO HARDWARE IN THE LOOP SYSTEM

The models for HIL simulation have an important role when there is the need to close the control loop. In the new testing device, here presented, the key aspect is that the experiments to perform are open loop test and consequently don't require models. MHIL has been thought as a system able to generate input signals to an ECU and analyze the outputs. Starting from data collected at test bench or, alternatively, at HIL during selected manoeuvres, these can be regenerated by the KBOX (a purposely designed signals generator) to stimulate the ECU under test. Then, the signals generated by KBOX are elaborated by the ECU control algorithms producing the commands for the engine actuators, i.e. valves, injectors, and so on. Finally, these commands are acquired and compared with a benchmark in order to verify the correct functioning of the control algorithms performed by the ECU. If any variation occurs, it can be attributed to a different firmware version of the electronic control unit or to unpredicted side effects due to some modifications of the control algorithms. A prototype of MHIL is reported in Figure 7.2. It is formed by two different hardware components; the core of the system, KBOX; devoted to signals generation; a console containing the actuators (external loads, see Figure 7.3) to be connected to the ECU. The console is equipped by: a power supply, to connect the MHIL to the electric commercial network; a suited console with led and interrupters to simulate different vehicle status as key-on/key-off, clutch inserted/not inserted; a throttle valve; injectors, lambda probes and electro-hydraulic valves. Moreover, the MHIL console is equipped with a fan coil to reduce the heat increase in the console, a RS232 port interface and an Ethernet port to link the MHIL to the host computer.

Figure 7.2 The MHIL hardware components: a console containing the actuators (external loads), the ECU under test and the signals generator (KBOX).

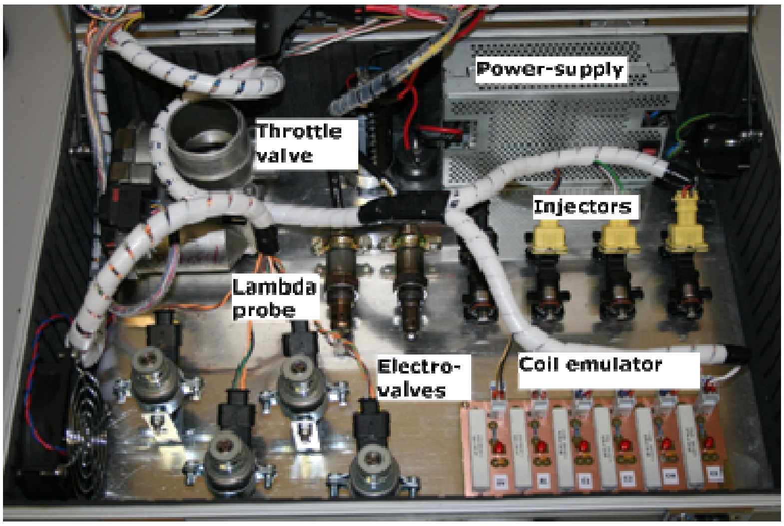

Figure 7.3 External loads

The KBOX characteristics have been designed to make the testing of engine control system flexible and suitable for future development (i.e additional sensors/actuators or extended functions). In order to stimulate all kind of ECU inputs, as pressure and temperature sensors, crankshaft and camshaft sensors, both inductive and Hall effects [8], the KBOX is formed by:

- 16 analogue outputs with a range between 0 and 5V (Max 10 mA); 16 digital outputs (max 100 mA);
- 8 outputs frequency, as camshaft/crankshaft signals, configurable as VRS type (i.e. waveform square + / - 12 volt with zero crossing) or Hall effect type (0 / 12 Volt);
- 2 outputs for knock signal (for variable setting of frequency and width) 2 outputs for lambda probe UEGO;
- 2 CAN Bus 2.0

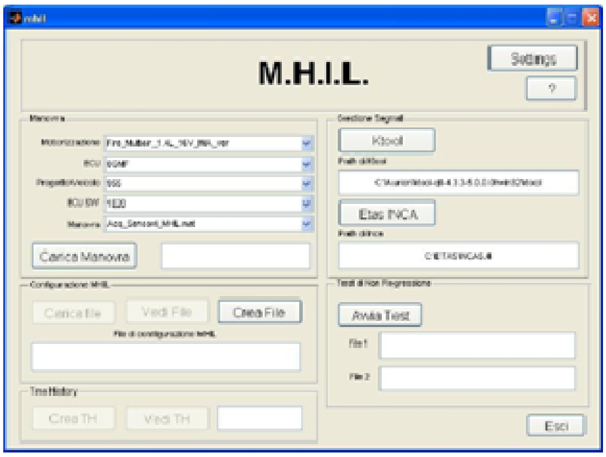

Figure 7.4 MHIL user interface

The user interface has been developed in Matlab environment. In Figure 7.4 is shown the main panel. Here, the first step is to choose the manoeuvre to run and upload the relative time history, i.e. the collection of signals to generate, to the KBOX. Then it is possible to con figure the analogue and digital channels to match the KBOX outputs with the ECU inputs, and run the experiments. Finally the data can be collected and analyzed by means of the Non Regression Test (NRT) module.

7.4. DIAGNOSTIC FUNCTIONALITIES TESTING

The rise of electronics in the automobile and the increased complexity of modern fuel injection and emission system place high demand on the diagnostic problem, though the monitoring of the vehicle during all operations. Commonly diagnostic functions are included in the ECU as standard component in the electronic engine management system. In fact, during the normal functioning, it is operated a diagnosis of sensors and actuators connected to ECU. The diagnosis can be electrical and logical. Regarding the last, the ECU checks only the correctness of the incoming signals, instead, for the electrical diagnosis, the ECU controls also the electrical load scheduled at its extremities for that device. Therefore, in order to prevent that the ECU goes in alarm and, hence, stops the experiment, the MHIL has to provide both the correct signals and loads to overcome the diagnosis. To this aim, external loads, connected directly to the ECU, are foreseen for those sensors and actuators performing electrical diagnosis.

For the other sensors, the signals are simulated by the KBOX. In particular, crankshaft and camshaft position sensor signals are created o -line. The crank and camshaft sensor waveforms are fixed (except for noise effects) as a function of crankshaft angle. These two waveforms are created and stored before the real time simulation. A separate program, with a graphical user interface, has been designed to create these two signals. This program also allows the user to create crank and camshaft signals with a variety of faults, using different frequency channel for differential and Hall effect sensor waveforms. Possible faults which can be created by the program include missing peaks in the crank or camshaft sensor, changes in width or height of chosen parts of the signal (usually the peaks) and the addition of sensor noise. It is also possible to inject these faults while the real time simulation is running. Any errors or faults detected are stored in the ECU memory, and stored fault information can be read via a serial interface.

Moreover, the communication with other ECUs present in the vehicle (as an example the Electronic Stability Program-ESP-ECU) is performed, over the Controller Area Network (CAN) bus. The benefit is that CAN protocol contains control mechanism to detect malfunctions, resulting that transmission errors are even detectable by CAN module. Since the majority of CAN messages are sent at regular intervals by the individual ECU, the failure of a CAN controller is detectable by testing at regular intervals.

CAN is one of the communication standard defined both in European and American OBD standard. In 2008 the CAN protocol will be the only permitted interface for OBD II diagnostics in the USA.

The CAN bus allows multiple devices to be linked together. A typical vehicle architecture is illustrated in Figure 7.5. The communication protocol shows several advantages for car makers. Firstly, CAN uses a two-wire solution (CAN Hi and CAN Lo); this enables higher data rate than the solution with a single wire, such as K-Line, that is a common serial communication standard. In fact EOBD specifies a maximum CAN data rate of 500,000 bits per second compared with the K-Line data rate of 10,400 bit/second. Secondly, CAN has extensive error checking built into the format of each packet composing the message.

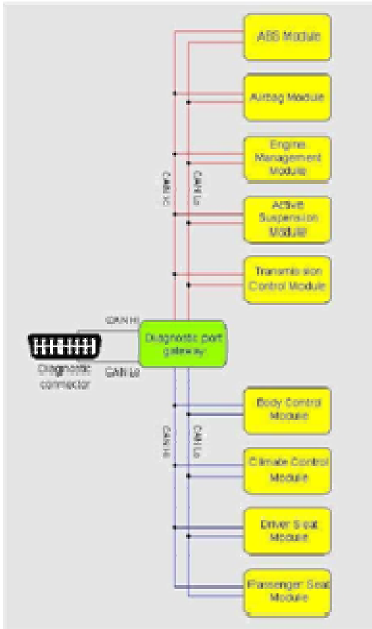

Figure 7.5 Typical vehicle architecture

Conventional tests nowadays performed can evaluate the network management, gateway functionality and CAN physical level, but unfortunately there are many restrictions: the tests can only be performed manually and are not reproducible; there isn't an automatic test report generation; and test coverage is incomplete, so that, the CAN communication cannot be checked thoroughly.

Now, considering that new vehicles generally use CAN to provide EOBD diagnosis, a key issue of the proposed MHIL is the capability to communicate with ECU under test through the CAN bus. In this way, the device can fulfil the following requirements necessary to test the diagnosis functionalities:

- read all pertinent ECU power drivers and signal outputs; logging of all CAN messages;
- interface to diagnostic serial line;
- test for both development and production ECU

In Figure 7.6 it is reported an experiment in order to validate the diagnostic functionalities provided by MHIL. In particular, in Figure 7.6 is shown the monitoring panel of the DIanalyzer, a diagnostic software tool of Fiat Group, where are present two errors due to lambda probe and electro-valve of the canister disconnected. This kind of experiment is necessary both to check hardware functionalities and to evidence eventual bugs during the code generation process on a different hardware.

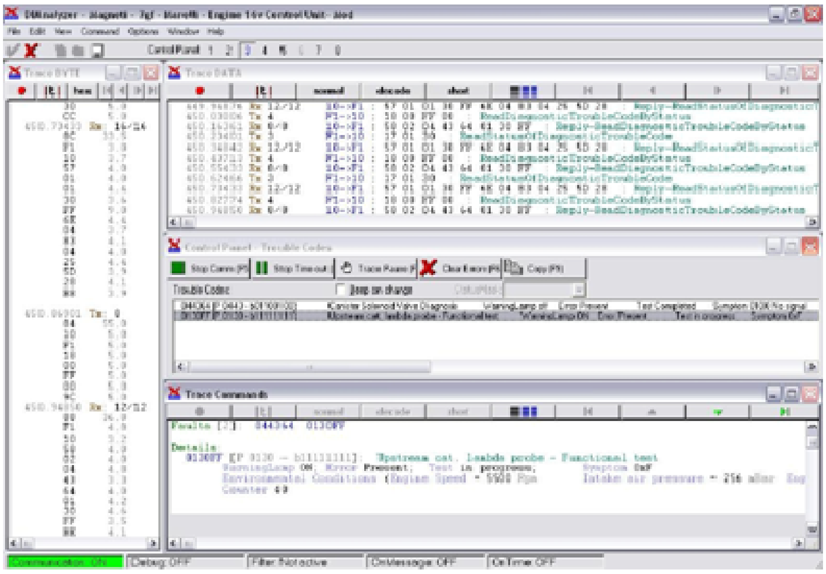

Figure 7.6 Diagnostic analysis

7.5. EXPERIMENTAL SOFTWARE VALIDATION

In engine control system development, the final release software is the result of a sequence of releases, each one introducing new requirements. As described in previous chapter the testing phase of the final release can be divided into three main steps.

- Module Testing:
- Integration testing, that includes the verification that the software is not regressed compared with the previous version (NRT).
- Functional testing.

Due to the very easiness use and to the optimized performance-cost ratio, the mHIL is really suitable for Non Regression Test execution.

In order to verify experimentally the performance of MHIL as testing system, two kinds of non-regression tests have been conducted to evaluate the differences of control software for development ECU. The first compares two different releases software, related to a four cylinder Spark Injection engine equipped with Variable Valve Timing (VVT) system. The second is performed for the same release software on two development ECU with different processor, having different hardware characteristics. This checks the possibility to change the ECU processor and to verify that all signals are maintained equal.

To this aim the first step is to collect data during a selected manoeuvre in order to obtain time history for MHIL device. In Figure 7.7 is reported the engine speed during the cycle. Here is also shown a comparison of the RPM signals during the two acquisitions necessary to guarantee that the ECU is always excited by the same input data set. The crankshaft position sensor generates 60 peaks per revolution, meaning a resolution of 6 degrees/peak. For indexing purposes, two of these peaks are empty, resulting in 58 peaks and 2 null outputs for revolution. The camshaft position sensor voltage output has 4 peaks corresponding to the 4 cylinders. Both signals are first created offline as a function of crankshaft angle and after downloaded on KBOX system.

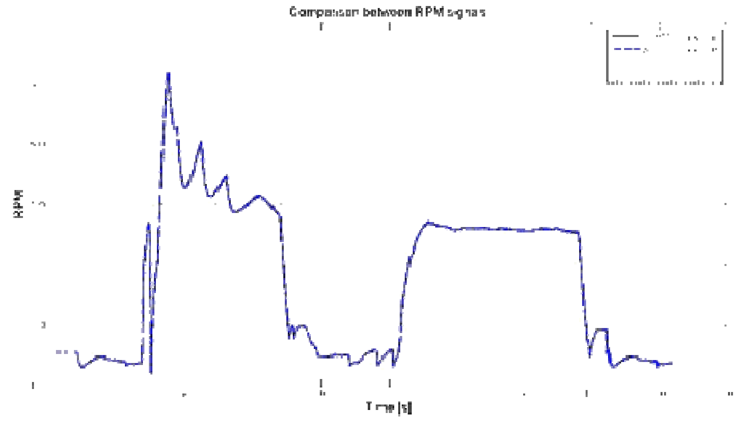

Figure 7.7 Engine Speed channel: comparison between TH acquired in vehicle and TH generated by the MHIL

Once the data are collected and the manoeuvre has been uploaded on the KBOX, it is possible to run the experiment stimulating the ECU under test and monitoring its outputs. Finally, a specific software tool has been purposely designed to perform NRT on the set of data previously obtained: the measured signals are firstly synchronized and then compared simply superimposing them. The tool allows to choose, for each signal, the acceptable range of the error and the error type (i.e. for the torque can be 5 Nm or 5 percent while for spark advance is 1 of absolute error). The result of the post-processing is a picture, in which different colours highlight if the variables under examination differ more than the selected error thresholds. In particular, the tool takes into account also the error quality, pointing out with different colours the errors that are inside the admitted threshold. For each plot the NRT tool underlines in red the differences between the two compared channels while the matching parts are in green. The plot is yellow if the channels are out of alignment.

The results of the two experiments are reported from Figure 7.8 to Figure 7.11. In Figure 7.8 and Figure 7.9 are illustrated the results of the first experiment, i.e. a comparison of engine torque and volumetric efficiency, for two different release software but implemented on the same ECU. In this experiment, both signals, inner variables calculated by the ECU, are very similar, Therefore it is possible to affirm that regression doesn't occur.

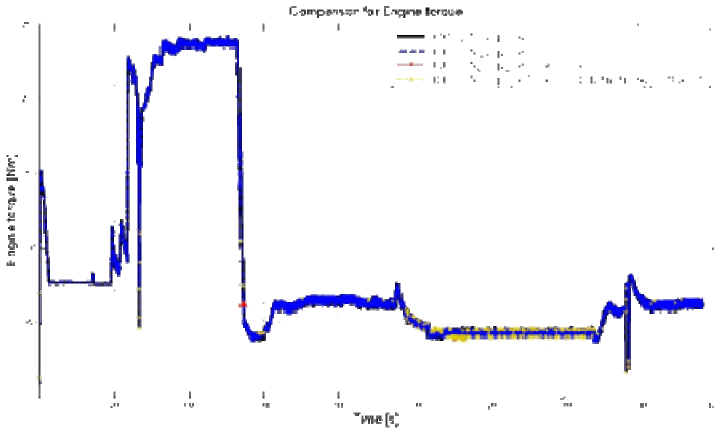

Figure 7.8 First experiment. Engine torque: comparison between data acquired from the same ECU but with two different release control software

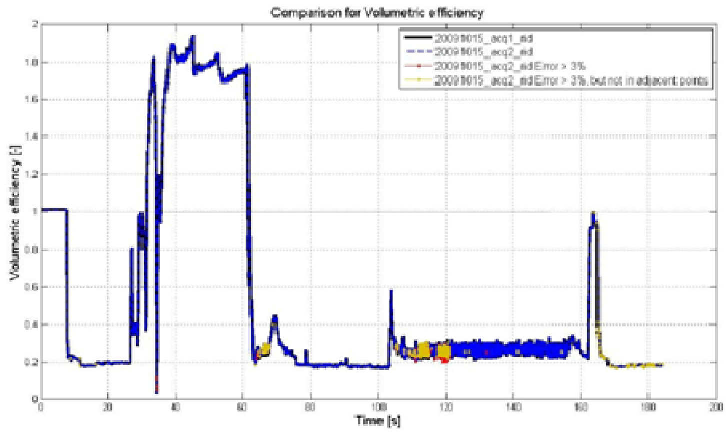

Figure 7.9 First experiment. Volumetric efficiency: comparison between data acquired from the same ECU with two different software release.

Conversely, in Figure 7.10 and Figure 7.11 showing again the engine torque and volumetric efficiency during the second experiment (same release software, implemented on two different ECUs, having different hardware characteristic), it checks some differences between signals. It means that some regression are present and, as a consequence, the test fails. Details on data are omitted for confidential reasons.

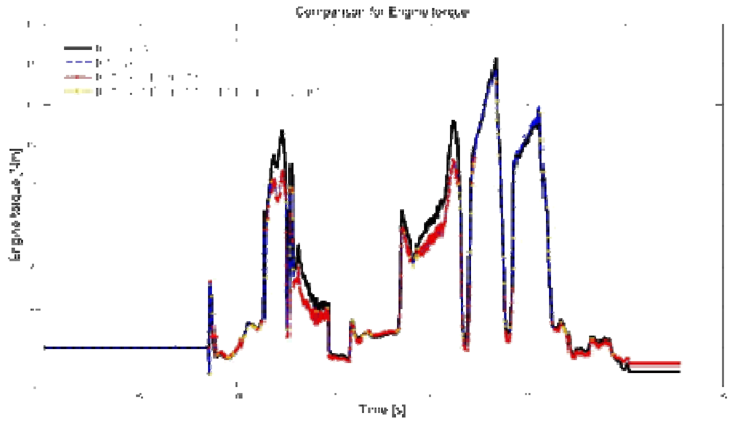

Figure 7.10 Second experiment. Engine torque: comparison between data acquired from the same ECU with two different software release

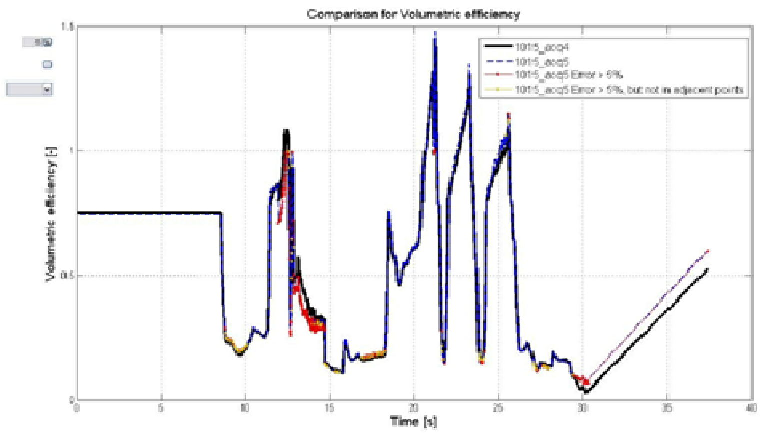

Figure 7.11 Second experiment. Volumetric effciency: comparison between data acquired from the same ECU with two different software releases.

Finally, in this chapter it is evident that the performances obtained both with MHIL and tradition HIL system [73] about the NRT procedure on several releases software are similar. The real advantage of the new device are the cost, the portability and the easy setup. The cost of the MHIL, equipped with load platform and independent power supply is a tenth of the traditional system, the MHIL is portable so to guarantee the test executions on the desk in different laboratories and the easy setup allows the control testing phase by not skilled poeple.

Chapter 8 Experimental application of presented tools and methods to Variable Valve Actuation (VVA) engine

In this chapter that concludes my thesis, some examples of experimental application of tools developed as results of my research work will be described. In particular the tools have been applied to an engine that represents a new technology frontier for gasoline engine: the MultiAir Engine that has been awarded as the 2010 engine of the year.

This kind of engine has been described in chapter 2. The only thing which is worth remarking from a control point of view, is the presence of a new additional actuator that opens to very challenging opportunities for the control community.

The first section gives an overview on the application of the modeling approach described in chapter 3 to MultiAir engine in order to evaluate the capability of the VVA actuator to improve the idle speed control performance (for a more exhaustive description see [74]). Actually, the traditional spark ignition engines have two control variables in charge to change the engine torque and, consequently, regulate the engine speed to the required value: the throttle position and spark advance. The engine torque dynamics corresponding to these two variables are strongly different: about 200ms for the throttle valve (via the air manifold filling/empting dynamics) and about 20ms for the spark advance. The presence of the Variable Valve Actuation system allows the regulation of the inlet air charge by means of different positions of the inlet valves (see 2.2). The torque changes, due to inlet air charge change, in a way that is much faster than traditional engines. Under this new situation, the control algorithms have to be reviewed in order for them to optimally exploit this new degree of freedom and improving the performance.

The second section is related to the application of calibration tools described in chapter 5 to MultiAir engine [76].

The third section describes some new control algorithms for VVA engines developed by means of the tools described in chapter 5. In particular, in the third section the following algorithms are described:

- a new spark advance calculation algorithm, that fits the specific needs of the MultiAir engine [77];
- an algorithm for gear shift suggestion [78];
- an algorithm for inlet air charge estimation [79].

8.1. A SIMULATION ANALYSIS FOR VVA AND IDLE CONTROL STRATEGIES FOR A GASOLINE ENGINE

In this section the engine model used to analyze the interaction between the idle control and the VVA system is derived by the methodology detailed in

Chapter 4. The use of the model for the VVA engine has been accurately detailed in [5] (for further details on possible use of model see [17]). In this section the focus has been given to the use of this model in order to achieve possible performance improvements of the idle speed control for VVA engines.

8.1.1. Mid-Ranging Idle Control

In the control field, there are numerous practical examples of control algorithms where, in order to meet the control objectives, two inputs must be manipulated to control a single output [81]. In some cases, this may be achieved by manipulating one input at the time. Such strategies are often referred to as split ranging. In other situations, it may be desirable or even necessary to simultaneously manipulate the inputs. For example, consider the situation shown in Figure 8.1, where the speed ω is controlled by a combination of two controllers in parallel. In particular, a slow control loop driven by R1 forces the control input u2 to a steady-state desired value. This is called Mid-Ranging technique.

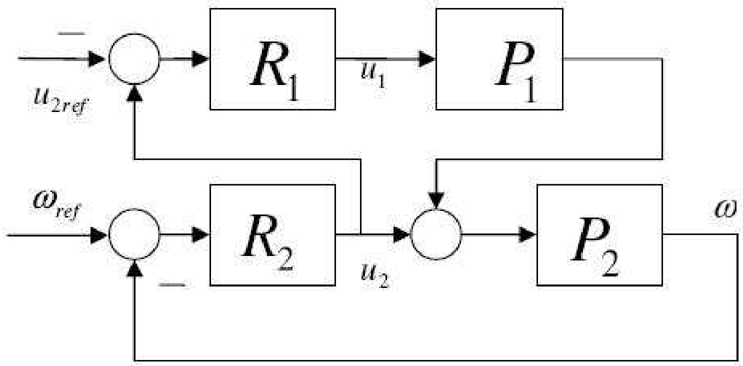

Figure 8.1 Mid-ranging control scheme. A slow control loop driven by R1 forces the control input u2 to a steady-state desired value

Today, the Mid-Ranging technique is largely used for idle speed control of spark ignition engines, usually controlled by commercial Electronic Control Unit (ECU) based on ”Torque Based” architecture [82]. Idling is one of the most often used functionalities in modern cars. This is especially the case in city traffic, where there are frequent stop and go situations. Therefore, improvements of the control performance for the idle speed control unit have always been a high priority. That is, keep the engine speed at a desired setpoint value and ensure good disturbance rejection while maintaining low fuel consumption.

Typical disturbances that are to be rejected by the controller are loads from the air-conditioning system or powersteering. Obviously, the ECU compensates such disturbances on engine torque by using the throttle valve. However, due to the slow dynamics of the air mass in the intake manifold, this would generate an unacceptably slow disturbance rejection. For this reason the spark advance is used as a second control signal, by advancing or retarding the ignition and obtaining an instantaneous torque variation from the engine. However, a deviation from the optimal spark ignition will result in higher fuel consumption.

Thus, the use of this signal should be kept at minimum and used only for improving the speed of the disturbance rejection. From a control point of view, this is a difficult problem since the system in question is nonlinear, multivariable (two inputs) and time varying. Moreover, the throttle control channel has a slower dynamics than the spark advance event. In the literature, this problem is usually approached by treating the two control channels separately (one control signal is set to constant while the other is modified), leading to performance degradation. Some other approaches consider linearized models resulting in local designs. There exist approaches where both control signals are treated in the same time (multivariable control), however the resulting controllers are highly complex and difficult to tune.

This section proposes the usage of a simple technique originally used in process control, called Mid-Ranging 'series', to analyze the idle speed control problem. The particular of this scheme is that the throttle is governed directly by the error between the desired and actual spark advance, while the spark advance is governed by the engine speed error, as reported in Figure 8.2. Moreover, the technique is particularly suitable for processes where one of the inputs has faster dynamics than the other; this is precisely true for the idle speed control problem in traditional SI engines. In the Figure 8.2, the spark advance control loop is the faster control loop. It takes as reference value the desired engine speed. The second loop contains the slower dynamics, where the air path controller adjusts the throttle angle such that in stationarity the spark advance will converge to the desired value. Traditionally the midranging schemes are based on PID controllers, which will be used here too. This idea is correctly based on experimental observation, the time delay between the spark advance application and the engine torque response is relatively small, symmetrical and predictable, if compared with the engine torque response from the intake manifold filling/emptying gas dynamics. This control concept, depicted in Figure 8.2, is simple and robust for a standard spark ignition engine, as demonstrated in [82] and in the commercial ECU available on the market.

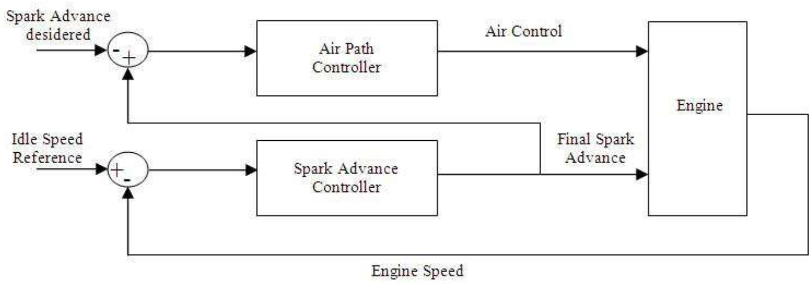

Figure 8.2 The Mid Ranging Idle Speed Control scheme

8.1.2. Mid-Ranging Algorithm With VVA System

Starting from the comparison with a Mid-Ranging ’series’ scheme, a similar control scheme for an engine model equipped with a VVA system has been realized. In the traditional engines, the air path is considered the slow way to regulate the engine torque, otherwise, in the modern engine equipped with VVA system the engine torque response, for any valve closing change, is able to exhibit a time delay comparable with the engine torque response to spark advance variation. For this reason, the Mid Ranging control scheme has been changed from its traditional form with two input in a ’parallel’ scheme with only an input, as shown in Figure 8.3. The advantage of this scheme is to avoid severe interaction between the inner and outer control loops of the serial scheme. In this way, both the control loops are driven from the same rpm error without control interactions, improving the advantages about the authority, quickness and steady state performance.

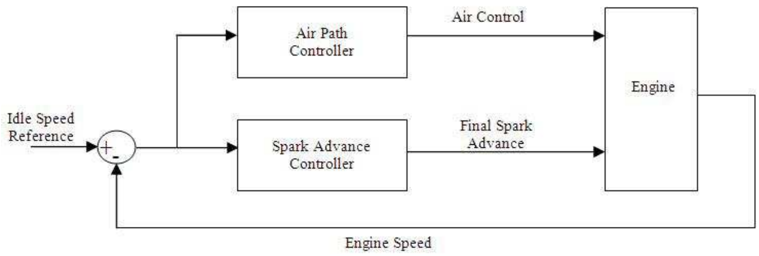

Figure 8.3 The Modified Mid Ranging Idle Control Scheme

For the experiments the previously described with VVA engine model has been used along with the same standard spark advance and air path controllers. In this experiment, the parallel controllers tuning has required similar difficulty of the serial scheme, without any loss of performance. The experiments have been accomplished by applying the control strategy to the four stroke internal combustion engine model. In Figure 8.4 a simulation result for engine speed based on two Mid-Ranging control schemes is shown. The results depict the quality and robustness of the idle speed control for both techniques.

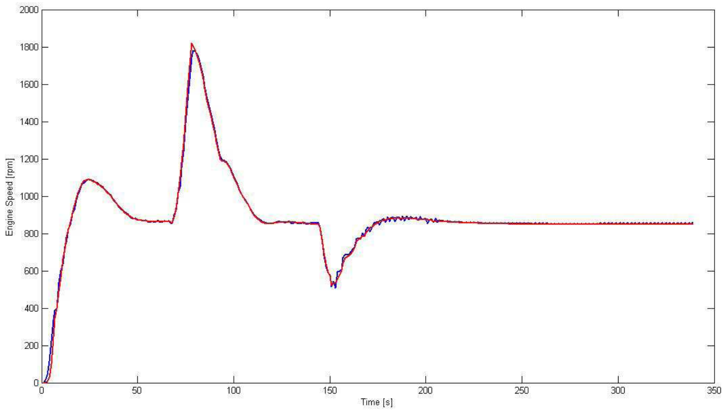

Figure 8.4 Engine speed: comparison between the Mid-Ranging 'parallel' scheme (red line) and Mid-Ranging 'series' scheme (blue line)

The good quality of the idle control and engine behaviour has been obtained by using only a Proportional- Derivative controller for the spark advance control loop and a Proportional-Integral controller for the engine air path control loop. In Figure 8.5 the spark advance behaviour of two kind of control schemes is depicted, showing good robustness, slightly improved by the opportunity to use more high spark advance during the engine work than the system usually allows. This can be seen in Figure 8.6, where it is possible to note that the throttle valve is used only for taking care of cut-off occurrence and improves the fuel consumption. Actually in the Mid-Ranging 'parallel' scheme proposed in this section, the mean value of the spark advance is higher than in the Mid-Ranging 'series' control scheme, guaranteeing fuel economy. Moreover, it should be noted that the general improvement partly depends by the better cylinder filling efficiency of the un-throttled VVA engines.

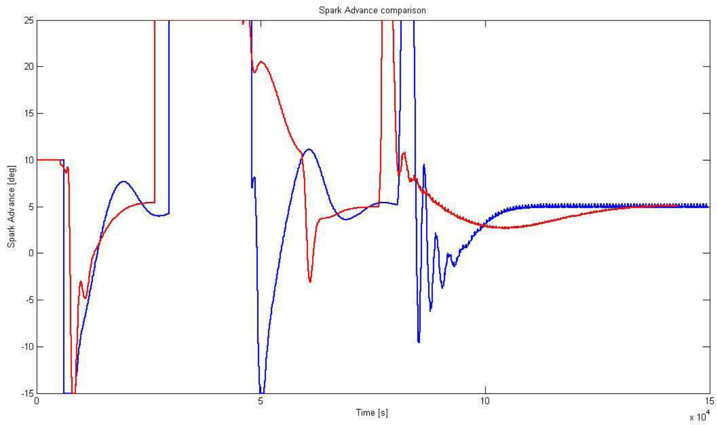

Figure 8.5 Spark advance: comparison between the Mid-Ranging 'parallel' scheme (red line) and Mid-Ranging 'series' scheme (blue line)

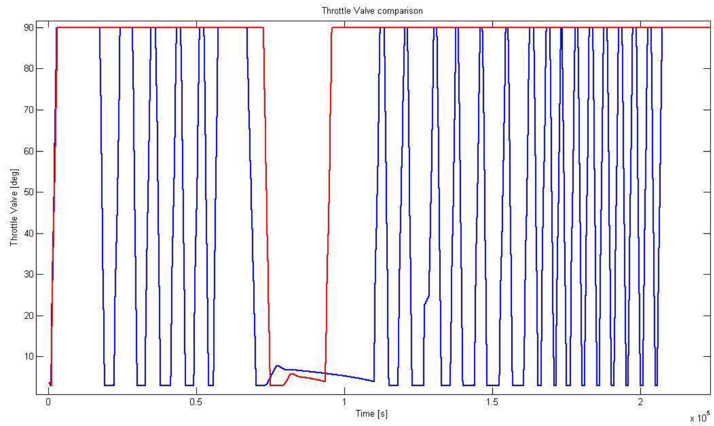

Figure 8.6 Throttle valve opening: comparison between the Mid-Ranging 'parallel' scheme (red line) and Mid-Ranging 'series' scheme (blue line)

8.1.3. Conclusions

A standard Mid Ranging scheme for idle speed control in spark ignition engines has been investigated on a realistic four cylinder engine model. A similar control scheme, here named "parallel" Mid Ranging scheme, specific for engine equipped with VVA has been deducted and tested in similar conditions, showing similar and slightly better properties in terms of performance quality and tuning easiness. The ability of the parallel Mid Ranging control scheme for idle speed control in internal combustion engines inspire to think that even other kinds of control variables, such as instantaneous air/fuel ratio, could be added in order to obtain a more robust idle controller, without increases the tuning procedure. In such a way, the idle control for a spark ignited engine will be more hopefully able to fulfill the future requirements in terms of emission reduction, fuel consumption, disturb rejection, and driver satisfaction. With three charge/torque control variables (air, spark advance, air/fuel ratio) or even more (injection phase for direct fuel injection systems), the challenge will be in the implementation of relatively simple, modular and separated control loops, with simple and almost separated tuning procedures.

8.2. APPLICATION OF CALIBRATION TOOLS TO MULTIAIR ENGINE

In this section an overview of the application of tools described in Chapter 5 to MultiAir engines is given. This tool assists the off-line calibration of the *base* control algorithms (i.e. charge and torque determination, injector model, spark advance calculation, exhaust temperature estimation for catalyst protection and so on). The use of optimization tools, to speed up the calibration and verification phases, has been proved effective and it is currently at same time a hot issue for the research community and already an industrial practice. The Automatic Calibration Tool, conceived for the Engine Control Unit (ECU) using MathWorks Matlab software, consists of 33 automatic calibration tools. The application of this approach, compared with the best competing techniques, has reduced the experimental test bench effort and the calibration time and has improved the estimation precision up to 4 times.

The described tools have been used for the calibration of the FIRE 1.4l 105 hp, 135 hp TC, 170 hp TC engines with VVA technology.

8.2.1. Base engine calibration objectives

With the term “Base Engine” calibration we mean the calibration activity dealing with the ECU algorithms involved in the engine operations which do not depend on a particular vehicle application, like charge estimation, injector model, spark advance calculation, torque estimation, catalyst protection and so on. In a model-based software development process, the simulation models of the algorithms are often available. Once the algorithm has been implemented and tested, it is possible to perform an accurate model calibration. The parameters calibration of the base-engine control algorithms consists in the identification of the parameter values which, coded in scalars, maps and vectors, best describe the engine behavior in a defined working range.

The calibration of a single algorithm process can be divided in bench experimental test design and execution, data analysis, algorithm calibration, bench test verification

These phases are usually repeated until the target precision is reached. Data analysis consists in the transformation of the experimental results into maps and vectors which will be used to describe the behavior of the engine in the ECU software.

The whole process can be speeded up using statistical techniques which reduce the number of needed experiments and maximize the informative contents of each test.

Using the proposed advanced calibration techniques, implemented in the MATLAB® environment, it is possible to automatically transform test information in calibration maps directly usable by the ECU. The higher precision achievable, especially when dealing with multidimensional actuation (throttle body, cam phaser for intake or exhaust camshaft, fuel injectors and so on), correspondingly reflects on better control performance (lower fuel consumption, higher maximum power and so on) and on a more robust characterization of the phenomena for the whole engine family, not only for the tested engine. This is also important to reduce the influence of noise factors on the engine performance and on the generation of diagnostic false alarms. Moreover, the availability of automatic calibration tools speeds up the development process of new engine control algorithms.

8.2.2. Automatic Calibration Tool (TCA) environment

The main ECU algorithms are coded in Matlab scripts, which simulate the behaviour of the embedded software. The modeled algorithm are:

- air charge estimation for each valve actuation mode;
- injection model;
- exhaust backpressure estimation;
- exhaust temperature estimation;
- delivered torque estimation;
- spark advance calculation;
- engine friction and pumping losses estimation.

Almost sixty maps and vectors are necessary to calibrate these algorithms for each application; it means thousands of scalar parameters. In order to satisfy this target, 33 automatic calibration tools have been developed, integrated in TCA environment. In Figure 8.7 the main TCA interface is shown.

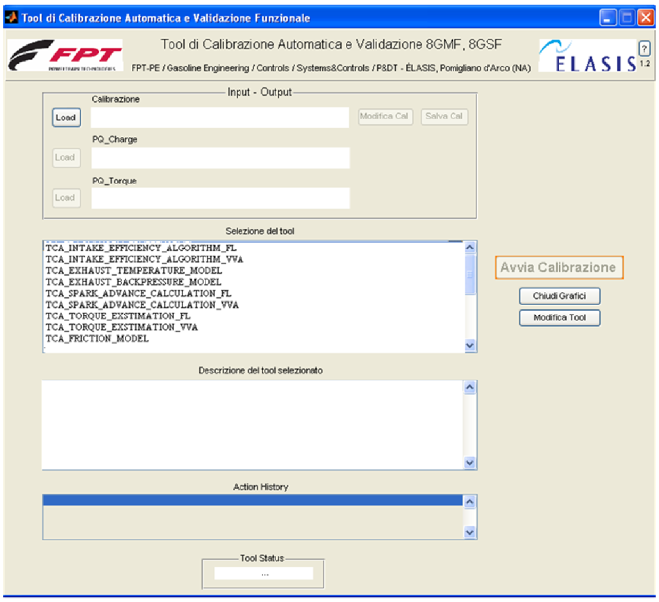

Figure 8.7 TCA Tool, user interface

The inputs are:

- a calibration file, to gather information about the dimensions and the breakpoints to be used in the maps calibration;
- engine bench test data files, specific for air charge estimation;
- engine bench test data files, specific for torque estimation, containing also the spark advance sweeps.

After the data loading, the user have to select a tool: all the necessary information to correctly execute the tool are displayed, like which maps have to be calibrated before running the tool and so on. Useful hints are also displayed that take into account the best practices to use the tool, suggested by the application team.

If only one map has to be calibrated, the Discrete Regression Tool is used (see Figure 8.8). The mentioned tool has been developed to calculate the values of the map that minimize the average square error between the experimental data and the surface, described as the bilinear interpolation of the map, using the same algorithm embedded in the ECU. The experimental data are plotted as red dots, while the map is represented in transparent blue [73].

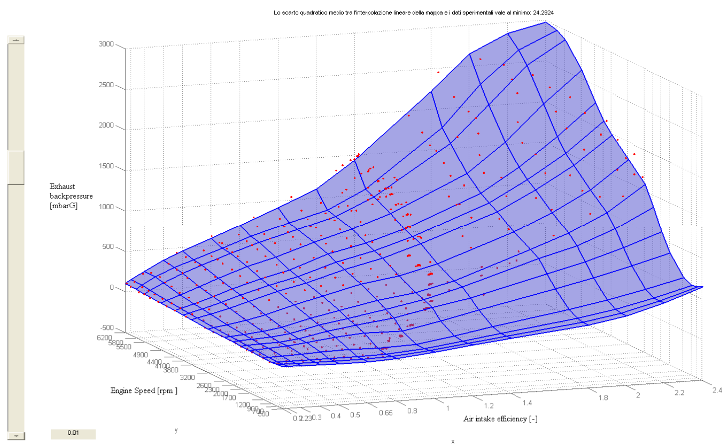

Figure 8.8 Relationship between Engine Speed, Air intake efficiency and exhaust backpressure.

It's possible to interactively modify the surface stiffness, using the slider as it's shown on the left in Fig.4, to produce more physical maps, avoiding to overfit the data. A high stiffness generates a plain surface, increasing, by one side, the average square error, and increasing, by the other side, the physical behaviour. The discontinuities can be caused by a measure error in some points. A non-continuous map can generate different output values for very similar input values. This is dangerous because an error in the input variable measure or estimation can generate a big variation on the output map value, causing instability in a control algorithm. The stiffness can be also used to impose a rule for the extrapolation of the map in not experimented points.

When the algorithm's output depends on more maps or vectors that interacts each other, it's better to use the Multi Map Optimization Tool ([73] and [77]).

The Multi Map Optimization tool modifies the calibration, varying the values of the maps, vectors and scalars that compose it, until the minimum average square error, or the average percentage square error, between measured outputs and estimated ones, is reached.

This tool solves a multidimensional optimization problem, using as optimization function the absolute error or the percentage square error.

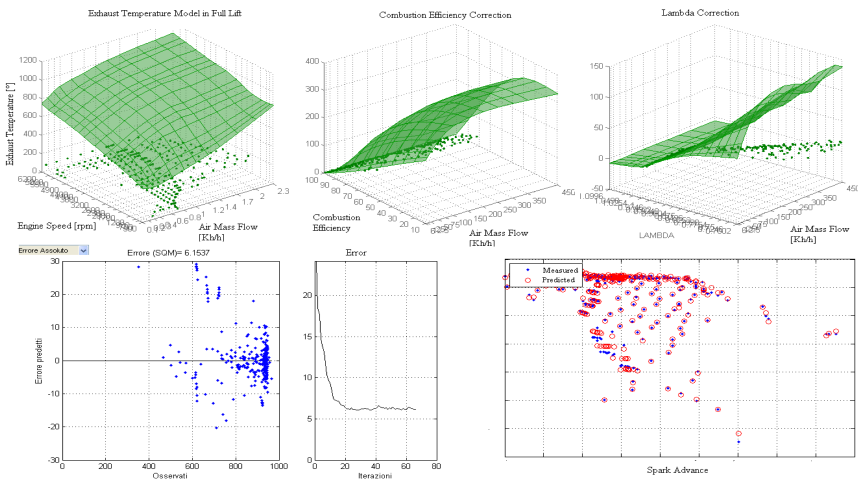

Figure 8.9 Automatic calibration of exhaust temperature model parameters

8.2.3. Performance evaluation

Once the calibration has been accomplished it is useful to check, with an expert eye, the achieved performance on different data in order to analyze the behaviour of the automatically calibrated map with different engines. This method is useful to consider also the dispersion between different applications and decide if the results are robust. In any case, this practice can put in evidence non desired hardware configurations between different testing engines, e.g. exhaust line or aspiration one.

The optimized maps are usually evaluated computing the average square error, denoted hereafter with σ, of the predicted values related to the experimental ones: the lower the error, the better the representation of the experimental points. The σ statistic, multiplied by 3, is a good estimation of the experimental error variation range. To be more precise, the experimental data fall in the described range with a probability of 99.97% in the hypothesis of normal error distribution.

For similar applications, the percentage average square error is more interesting. It is obtained dividing the square error by the experimental one and multiplying it by 100. In this way the statistic is more correlated with the final performance; e.g. in the air charge estimation, the percentage average square error is correlated with the error on the mixture title actuation.

Moreover, for each calibration problem, a specific tool has been developed, which graphically and numerically shows the results of a calibration variation on the ECU estimation. This tool is useful to rapidly verify that the calibration agrees with the imposed criteria, and to make the fine tuning, following the calibration engineer experience. The result of an automatic calibration has to be verified by an expert for two reasons:

- the calibration tool, if there are missing data, extrapolates the behaviour of nearest points;

- the calibration tool optimizes a performance that is usually the most important, but in some operating conditions it isn't the only one of interest.

In Figure 8.10, it is shown the air charge estimation trend in early closing mode at fixed engine speed. Moreover it's possible to analyze the charge algorithm performance using the following points of view:

- Fixed engine speed efficiency curves
- Air charge estimation percentage error
- Fixed engine speed exhaust temperature curves
- Exhaust temperature percentage error
- Fixed engine speed exhaust backpressure curves
- Exhaust backpressure percentage error

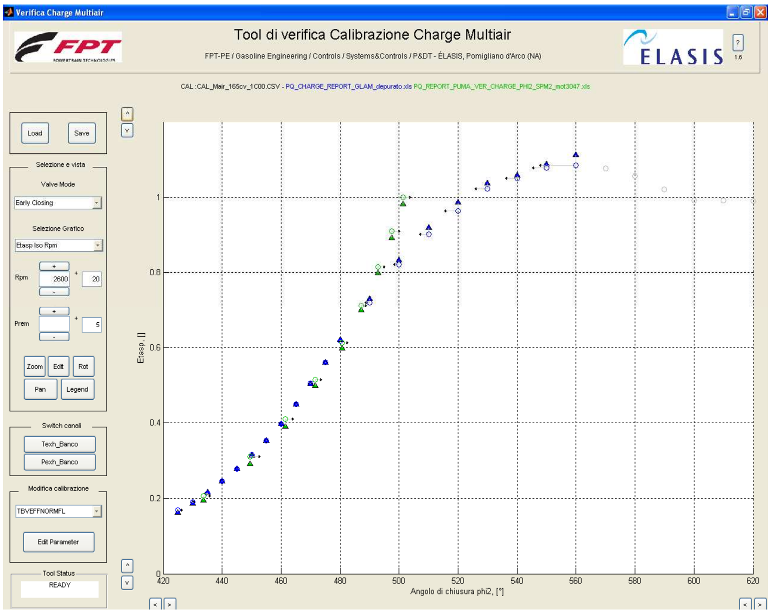

Figure 8.10 Air charge estimation trend in early closing mode

This tool solves a multidimensional optimization problem, having as optimizing function the absolute error or the percentage square error.

The advantages are the same of the discrete regression, plus the following:

- many maps can be optimized simultaneously, the solution minimizes the total error, not the single map one, so the result better fits the data;
- the precision and the continuity of the maps are better than using other methods.

In Figure 8.10 the circles represents experimental data, while the black dots represents ECU VVA angle actuation using the current calibration. The triangles represents the air charge consequently estimated. If the calibration is not precise, the distance between triangles and dots, and between circles and dots, represent, respectively, the air error and the angle actuation error. The symbol colour identifies different engines.

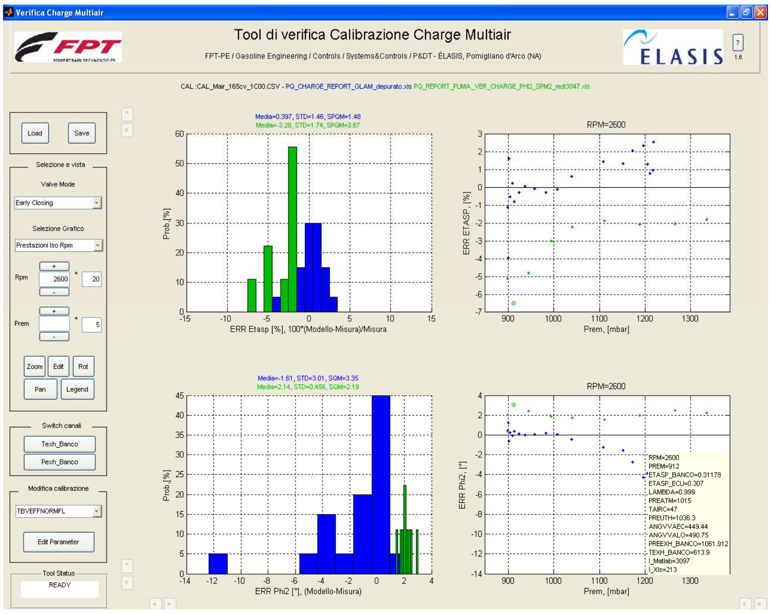

Figure 8.11 Air charge estimation percentage error in early closing mode

In Figure 8.11 a statistical analysis of the air estimation precision is done, using histograms and scatter plots.

In Figure 8.12 the main screen of the spark advance calculation and torque estimation tool is shown. The main graph shows the so called umbrella curves, which represents the mean indicated torque depending on spark advance, at a predefined engine speed and at defined mixture.

Acquired data are represented by circles, while calibration estimated corresponding points are the black dots. The red triangle represents the maximum torque point for each load breakpoint, while the big black plus symbol is the working spark advance. Clicking over an experimental point, additional information will be shown, like fuel consumption, temperatures and so on. Clicking over an ECU calculated point, instead, the relative formula adopted will be shown, with input maps values.

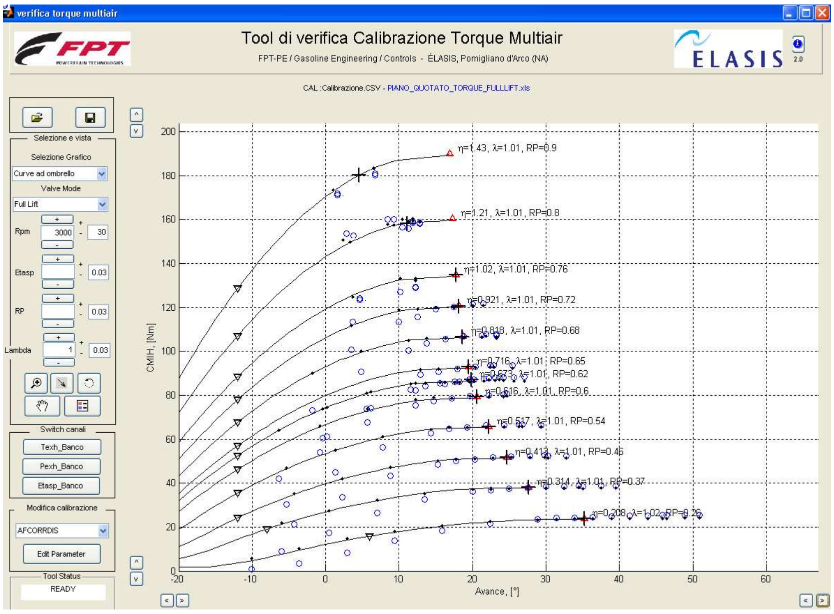

Figure 8.12 Torque estimation performance

It's possible to deeply analyse also the torque algorithm performance using the following points of view:

- umbrella curves;
- indicated Torque estimation percentage error;
- fixed engine speed Effective Torque curves in function of spark advance;
- effective Torque estimation percentage error;
- fixed engine speed friction and pumping losses curves;
- friction and pumping losses percentage error.

8.2.4. Conclusion

The innovative FIAT Powertrain Technologies engines equipped with VVA technology have been calibrated using the methodology and the tools exposed in the present work. The reduction of the bench test effort and of the calibration time have been a key factor for the improvement of the calibration process quality, increasing the time available for performance refinement and algorithms improvement. The calibration know how has been successfully transferred from the first engine application to the others, making this approach to the calibration a key factor in the engine development process

8.3. NEW CONTROL ALGORITHMS

8.3.1. A new spark advance calculation algorithm, that fits the specific needs of MultiAir engine

It is well known that in spark ignition engine the spark advance is one of the actuation variables to be calculated in order to produce the desired torque. In a Variable Valve Actuation engine the spark advance calculation is much more complicated because of all available freedom degrees.

In this section a new algorithm for spark advance calculation for such a kind of engines is described.

The torque supplied by the engine is estimated using mainly engine speed, air inlet efficiency, valve lift profiles and spark advance. In an engine without the VVA, the algorithm uses only two maps and one vector:

1. MTA(speed, eta), called the maximum torque advance map: it describes, for each engine speed - air inlet efficiency point, the spark advance that maximizes the torque. If the detonation occurs before reaching the real maximum, an extrapolated value is used to best fit data. In Figure 8.13 the red triangles x-coordinate represent the maximum torque.
2. 2. MT(speed, eta), called the maximum torque map: it describes the indicated torque measured at the maximum torque advance. In Figure 8.13 it is the red triangles y-coordinate.
3. UC(advance - MTA(speed, eta)), called the unique curve: it describes how the distance between the spark advance and the maximum torque spark advance reduces the torque. Its output is 1 if the input is 0. The output decreases while the input difference increases. It is very similar to a parabolic curve and it has the property to fit well the experimental data in the equation

*TORQUE=MT(speed, eta) * UC(advance – MTA(speed, eta))*

where:
speed is the engine speed
eta is the air inlet efficiency
advance is the actuated spark advance

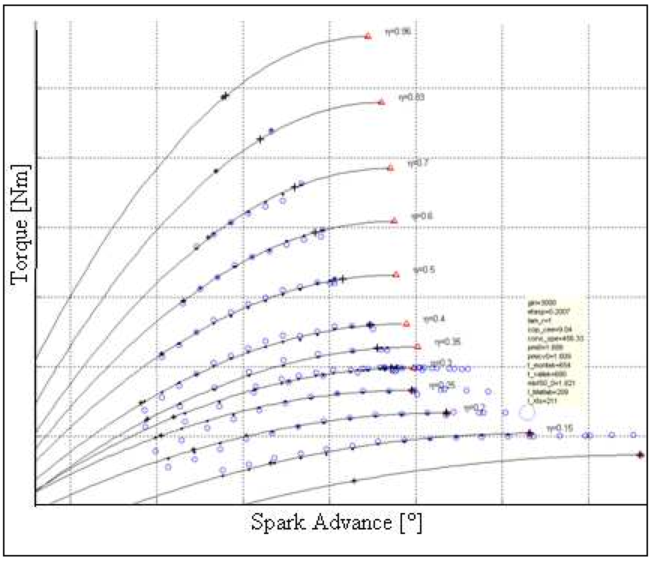

Figure 8.13 Torque interface

In engines with the VVA system, this algorithm implies one map and four vectors in Full Lift mode. Generally, an spark advance map is a two-dimensional map dataset consisting of two variables such as engine speed and load.

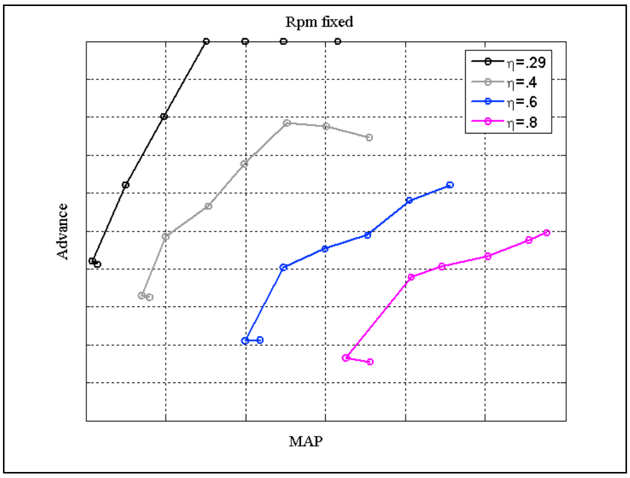

Figure 8.14 Advance versus manifold Pressure at Rpm fixed for different air inlet load

In EIVC or LIVO mode, at fixed speed and load, the spark advance depends also on the manifold pressure (Figure 8.14). A higher pressure implies a lower

Early Closing angle, reducing the effective compression ratio and the end of compression temperature; This reduces the knock phenomenon, increasing the maximum allowed spark advance.

The Figure 8.14 shows, in EIVC mode, the spark advance trend versus the manifold pressure at fixed speed, for some values of air inlet efficiency (the values have been omitted for confidentiality reason).

Therefore, while in Full Lift mode, is yet valid that:

Advance = f(Speed, Load),

in VVA mode it has to be considered the dependency from pressure:

Advance = f(Speed, Load, Manifold Pressure)

The MMO has allowed to manage the phenomenon, and different proposed algorithms have been automatically calibrated and compared in a few time.

The Figure 8.15shows the algorithm with the best performance.

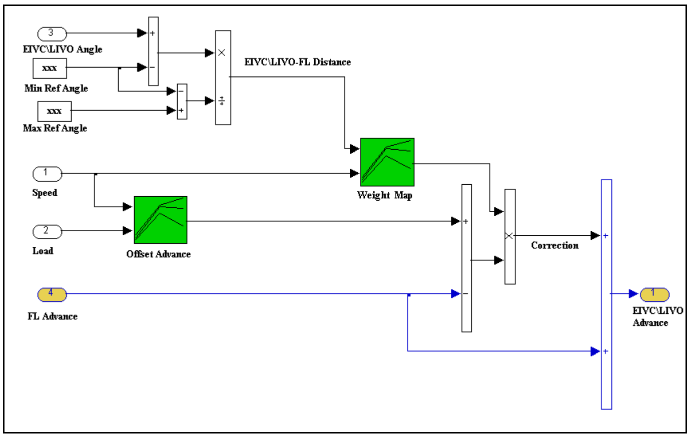

Figure 8.15 Spark advance algorithm

In EIVC or LIVO mode, the spark advance is calculated as the sum of the spark advance in FL mode and the correction for VVA system. This correction is the weighted average between the FL spark advance and the reference angle spark advance. The weight factor map is function of the speed and the EIVC\LIVO-FL Distance, calculated as follow:

$$\mathrm{EIVC \setminus LIVO - FLDistance} = \frac{\mathrm{Current\ EIVC \setminus LIVO\ Angle - MinAngleRef}}{\mathrm{MaxAngleRef\ - MinAngleRef}}$$

The *MinAngleRef* is the minimum realizable cam angle for each mode, while the *MaxAngleRef* is the maximum early closure angle for EIVC and the maximum late opening angle for the LIVO on the cam profile (Figure 8.16).

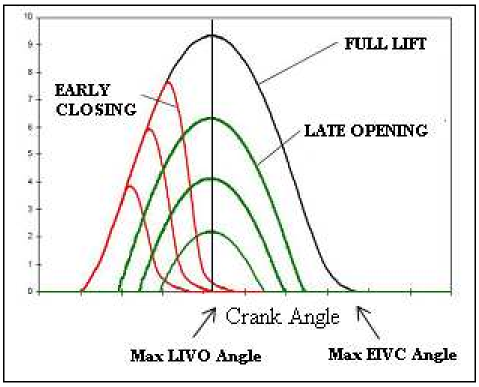

Figure 8.16 Max Angle Ref for EIVC and LIVO mode

Using the Multi Map Optimization (Figure 8.17), the resulting optimized maps are smooth and the mean square error meets the precision target (see also the error distribution on Fig.12) and the predicted versus observed graph shows the validity of the chosen model. The experiments have been conducted on test bench for a gasoline engine Fiat SI Turbo 1.4 l 135 Hp with VVA.

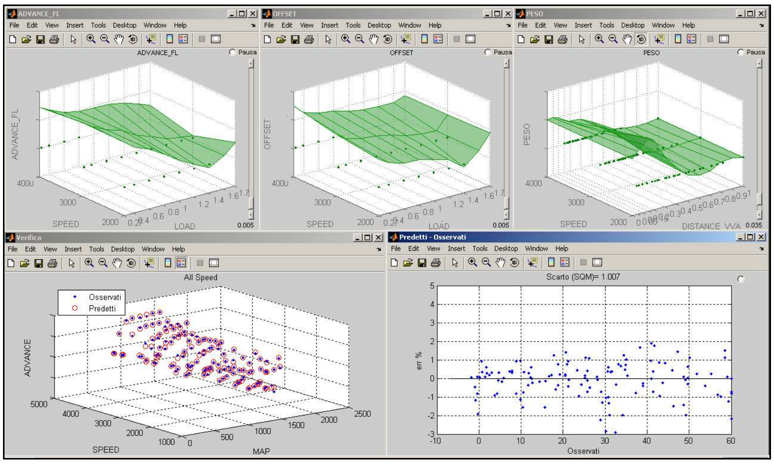

Figure 8.17 The MMO for spark advance algorithm

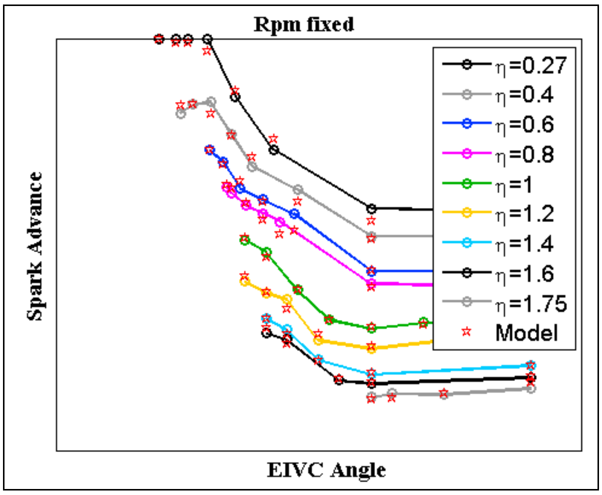

Figure 8.18 Graph predicted vs observed

The Figure 8.18 shows the spark advance trend versus the early closing angle for different values of air inlet efficiency at fixed Speed. The circle points are experimental, while the star points are the points predicted by the model.

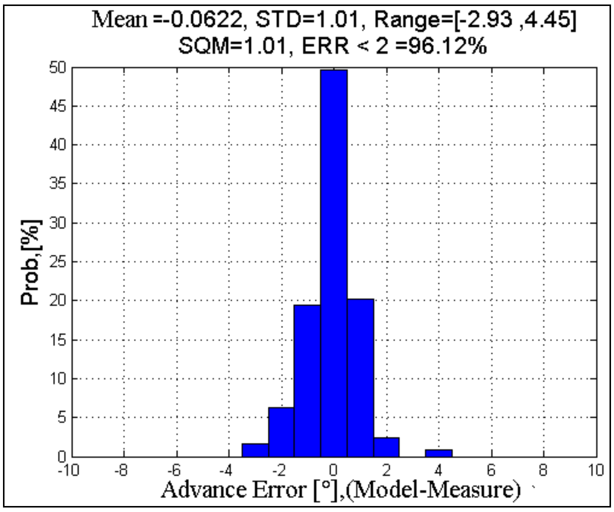

Figure 8.19 Error Distribution

The described algorithm is now equipping FIAT Group automobiles. The good accuracy of the described algorithm has permitted to improve the dynamic response of the engine, taking full advantage of the VVA technology. The described development tools are currently used by the calibration engineers in worldwide FPT engine test departments.

8.3.2. An Algorithm for gear shift suggestion

The gear shift suggestion is an helpful tool in order to achieve a further reduction of CO_2. The basic idea is to suggest to the driver to put the engine in conditions that minimize the fuel consumption. The existing algorithm is strongly depending on the vehicle characteristics and then requires a specific calibration for any different vehicle, even if it is equipped with the same engine. The proposed algorithm (patented with number EP2017505 IT), allows a single calibration for an engine and then reduces the calibration effort and saves CPU and memory resources.

The objective of this algorithm is a suggestion that does not induce a difference in vehicle speed. This means that with the suggested gear the torque at vehicle wheel needs to be the same one that is applied with the current gear. So this condition is the constraint to be considered in an optimization problem, where the function to be optimized is the fuel consumption.

In the Figure 8.20 a diagram indicating the engine torque at the current gear (x-axis) and the fuel consumption (y-axis), where the suggestion zones have been indicated. The part above the black line (that indicates the fuel consumption with the current gear), corresponds to an up-shift suggestion, the part under the black line is related to a down-shift. For this reason the algorithm needs only two maps (look up tables) to be implemented. Both maps are addressed by current gear and engine speed: the first map (UpShift Map) gives in output the torque under the which an up-shift suggestion has to be given, the second one (DownShift Map) gives in output the torque above the which a down-shift suggestion has to be given.

The described algorithm used the calibration tools described in 5.6 to obtain the UpShift Map and DownShift Map.

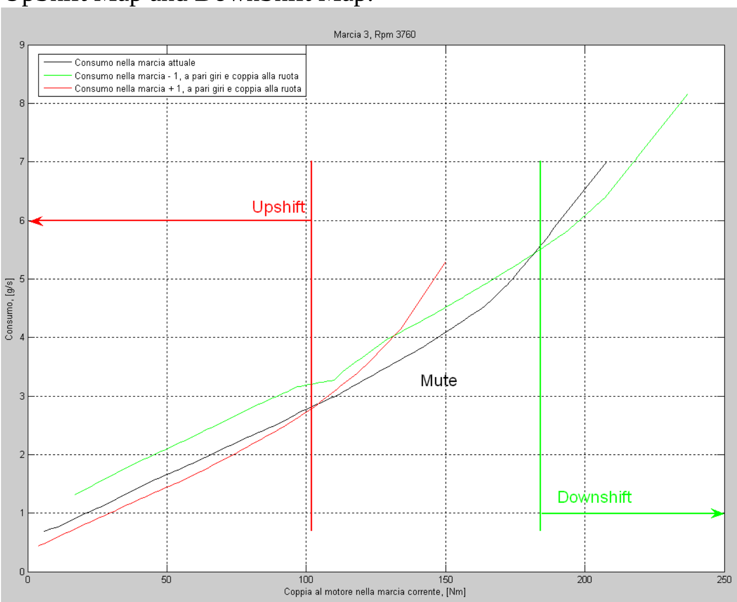

Figure 8.20 Torque and fuel consumption

- a new spark advance calculation algorithm, that fits the specific needs of MultiAir engine;
- an algorithm for gear shift suggestion;

8.3.3. An algorithm for inlet air charge estimation

In Spark Ignition engine control system the inlet air charge estimation is a very important factor to fulfill engine performance. Currently different solutions are implemented by means of specific sensor or estimation algorithm. In second case two different approaches are proposed: Saint Venant equation or filling and emptying model. The algorithm, proposed in ([79]) and patented, uses a combination of both approaches: the use of the one of the other is depending on the ratio between the value of pressure downstream the throttle body and the upstream value. This algorithm is particular suitable for Variable Valve Actuation engines where the inlet air charge is harder to estimate than in a conventional engine.

In the Figure 8.21 the Simulink diagram is shown. The details of the implementation are available in [79].

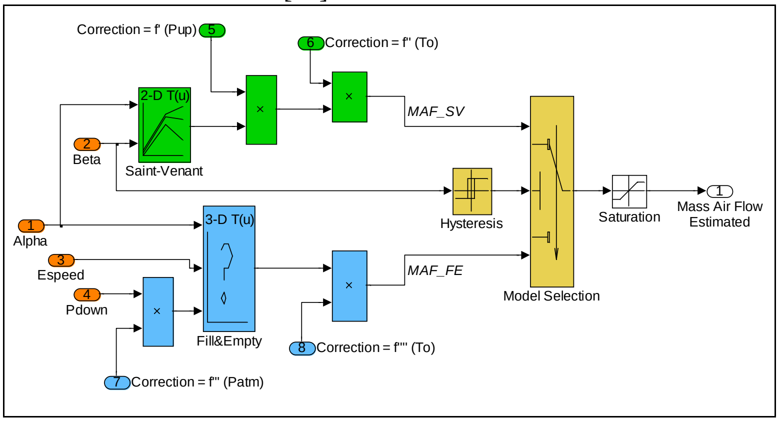

Figure 8.21 Inlet air charge estimation algorithm - Simulink diagram

Chapter 9 Conclusions

The automotive industry is facing a very challenging phase. Actually the coming scenario consists of more and more ambitious requirements (regarding fuel consumption and fun to drive) and new constraints due to new emissions legislation (Euro 5+ and in the following Euro 6 are going to be mandatory for every new vehicle homologation starting from September 2011 and for every vehicle registration from January 2014. The most part of new requirements is affecting the powertrain and implementation involves almost always electronics. Consequently the engine control systems are becoming more and more complex. In the figure the complexity of control system has been expressed in terms of new components and number of control parameters (to be calibrated).

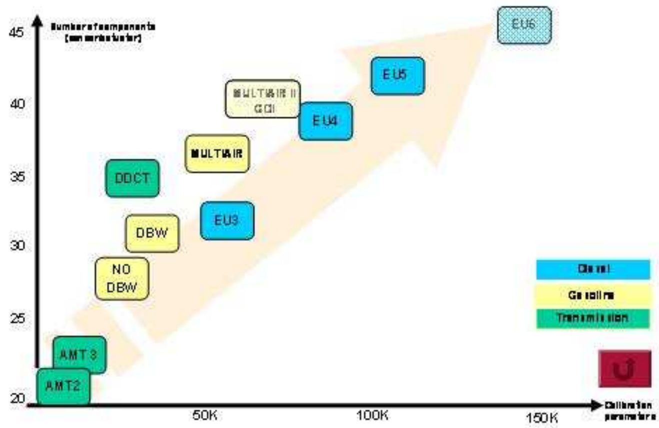

Figure 9.1 Control system complexity

On the other hand time to market is reducing more and more and consequently development time is becoming shorter and shorter: for instance development of new engine and related control system passes from 25 months to 18 months.

The peculiarities of automotive industry is that it is based on large scale production and so any kind of problem can affects thousands and sometimes millions of models: there is less time to develop system but no errors are admitted. Several cases regarding call campaigns have been published also for the most important car makers. In this situation to meet requirements and to guarantee the necessary quality level (Index Per Thousand Vehicles indicators are used to define the expected level of quality: some units are the current targets), a development process needs to be effective, predictable and fast.

A quite known development process can be represented by a V-cycle (Figure 9.2).

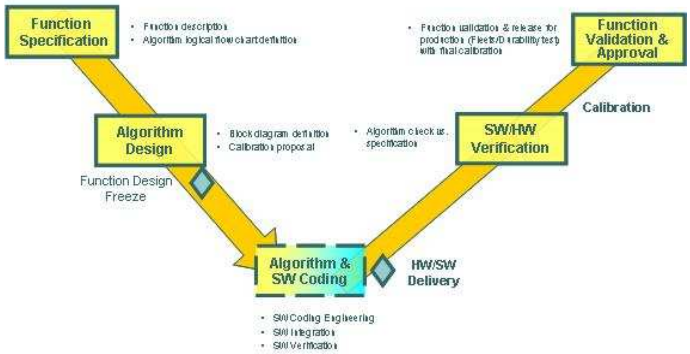

Figure 9.2 Control system development cycle

The development of complex systems in shorter time to meet more ambitious requirements can be faced only by means of specific tools.

The research has been focused on the realization and application of new tools to support and some times make automatic the most critical phases of development process of an engine control system. In particular following tools have been developed and experimented in real cases:

- engine model
- engine control systems design and calibration environment
- low-cost Hardware-In-the-Loop system

In particular the:

- engine model can be used during:
  - the first phases of control specification, to define the system targets;
  - algorithm design, to verify in simulation the control system performance;
  - HW/SW verification, by means of a hardware in the loop simulator, to verify the control implementation with respect to control algorithm models;
- engine control system design and calibration environment assists the design and implementation phases and makes automatic the final calibration activity;
- low cost HIL has been used during the verification phase in particular for the integration phase and the low cost makes possible a wide application of the tool.

The described tools have been applied during development of control systems for new variable Valve Actuation engines (see Figure 9.3 and Chapter 2 for more details).

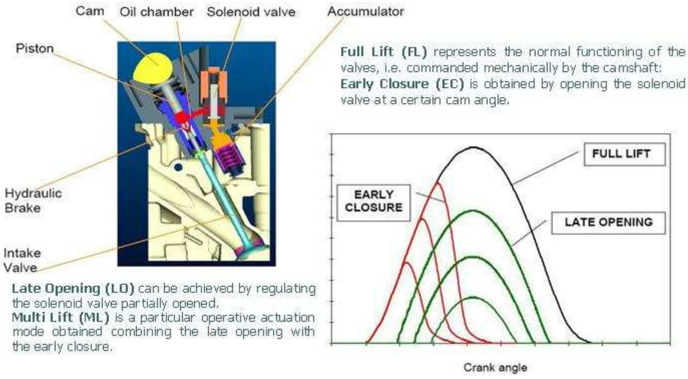

Figure 9.3 Variable Valve Actuation

Such a kind of engine introduces a new freedom degree that can be used for reduce fuel consumption and emissions, but on the other hand increases the complexity of the overall system and consequently the development and validation time for control system.

A rough estimation of benefits of using these tools, has shown a remarkable reduction of development time that in some cases (i.e. the calibration time has been reduced from 20 to 1!). Another consideration is that the complexity of systems makes the availability of such kind of tools a must and not an option.

Chapter 10 Bibliography

[1] J. Heywood, "*Internal combustion engine fundamentals*", McGraw Hill, 1988.

[2] R. Della volpe, M. Migliaccio, "*Motori a combustione interna per autotrazione*, Liguori Editori, 2000.

[3] C. F. Taylor, "*The internal combustion engine in theory and practice*", The M.I.T Press, 1966.

[4] P. Scalori, "*L'autoveicolo e la sua evoluzione*", Politecnico di Torino, 2000

[5] A. Palladino, G. Fiengo, F. Cristofaro, A. Casavola, and L. Glielmo, "In cylinder air charge prediction for VVA system: Experimental validation", *Multi-conference on Systems and Control*, 2008.

[6] www.fptpowertrain.com

[7] www.fiatgroupautomobilexpress.com

[8] L. Guzzella, C.H. Onder, "*Introduction to Modeling and Control of Internal Combustion Engine Systems*", Springer, 2004.

[9] U. Kiencke, L. Nielsen, "*Automotive Control Systems for engine, driveline, and vehicle*", Springer, 2000.

[10] T. Denton, "*Advanced automotive fault diagnosis*", Elsevier Butterworth-Heinemann, 2006.

[11] Bosch, "*Gasoline- Engine Management*", Bosch GmbH, Technology, 2004

[12] M. Muller, E. Hendricks, and S.C. Sorenson, "Mean value modeling of turbocharged spark ignition engine", *SAE Paper* 980784, 1998.

[13] E. Hendricks, "Isothermal vs. adiabatic mean value SI engine models", *3rd IFAC Workshop Advances in Automotive Control*, 2001.

[14] E. Hendricks, "The analysis of mean value engine models", *SAE Paper* 980563, 1989.

[15] J. Deur, D. Hrovat, and J. Asgari, "Analysis of mean value engine model with emphasis on intake manifold thermal effects", *Conference on Control Applications*, 2003.

[16] A. Palma, A. Palladino, G. Fiengo, F. De Cristofaro, F. Garofalo and L. Glielmo, "A modeling approach for engine dynamics based on electrical analogy", *IFAC* 2008

[17] F. De Cristofaro, G. Fiengo, A. Palladino, A. Palma "Inner phenomena modeling occurring into SI-ICE combustion chamber: Knock, EGR and Scavenging", *International Journal of Modeling, Identification and Control* 2009

[18] A. Casavola, F. De Cristofaro, I. Montalto "A new Development Environment for Embedded Control Systems Design and Interactive Optimization Methodologies for Robust Calibration of Automotive Engines" *Large scale Computation, Embedded Systems and Computer Security*, ISBN: 9781607413073, Nova Science Publisher, NY, 2010

[19] Henriksson, D., Cervin, A. and Årzén, K.E. “TrueTime: Real-time Control System Simulation with MATLAB/Simulink”, *Proceedings of the Nordic MATLAB Conference*, Copenhagen, Denmark, 2003.
[20] Ohlin, M., Henriksson, D. and Cervin, A. “*TrueTime 1.4 – Reference Manual*”, Department of Automatic Control, Lund Institute of Technology, Sweden, 2006.
[21] Calvez, J.P. “*Embedded real-time systems – A specification and design methodology*”, 1st Edition, John Wiley & Sons, Chichester, England, 1993.
[22] Gaviani, G., Gentile, G., Stara, G., Romagnoli, L., Thomsen, T. and Ferrari, A. “From conception to implementation: a model based design approach”, *Proceedings of 1st IFAC Symposium on Advances in Automotive Control*, Salerno, Italy, 2004.
[23] Oshana, R. “Introduction to embedded and real-time systems”, *Proceedings of Embedded Systems Conference*, Boston, USA, 2003.
[24] Kiencke, U. and Nielsen, L. “*Automotive Control System – For Engine, Driveline and Vehicle*”, Springer, Berlin, Germany, 2000.
[25] Törngren, M., Elkhoury, J., Sanfridson, M. and Redell, O. “*Modelling and Simulation of Embedded Computer Control Systems: Problem Formulation*”, Technical report, Mechatronics Lab, Department of Machine Design, Royal Institute of Technology, Stockholm, Sweden, 2001.
[26] Köster, L., Thomsen, T. and Stracke, R. “Connecting Simulink to OSEK: Automatic Code Generation for Real-Time Operating Systems with TargetLink”, *Embedded Intelligence*, Nuremberg, Germany, 2001.
[27] *TargetLink – Production Code Generation Guide*, for TargetLink 2.0, dSPACE GmbH, Paderborn, Germany, 2004.
[28] Buttazzo, G. “Sistemi Real-Time per il controllo automatico: problemi e nuove soluzioni”, *Automazione e Strumentazione*, Anno XLVIII, N. 5, pp. 107-116, 2000.
[29] Redell, O. “*Global Scheduling in Distributed Real-Time Computer Systems – An Automatic Control Perspective*”, Technical Report, Mechatronics Lab, Department of Machine Design, Royal Institute of Technology, Stockholm, Sweden, 1998.
[30] Liu, C.L. and Layland, J.W. “Scheduling algorithms for multiprogramming in a hard real-time environment”, *Journal of ACM*, Vol. 20, N. 1, pp. 46-61, 1973.
[31] “*Automotive Electrics Automotive Electronics*”, Fourth Edition, Robert Bosch GmbH, Plochingen, Germany, 2004.
[32] Annunziata, M., De Cristofaro, F., Flauti, G. and Scala, S. “Development of a Virtual Test Environment for Engine Control Systems: ESILE, Elasis Software In the Loop Environment”, *FISITA* 2004.
[33] De Cristofaro F., Riegel A., Di Martino U., De Sisto G., "Estimation of drawn air mass on a ICE: implementation of a new algorithm to better estimate the atmospheric pressure influence", *Proceedings of SAE International* 2007-01-1344, Detroit, 2007

[34] Box, G.E.P., Draper, N.R. "*Empirical Model-Building and Response Surfaces*", Wiley & Sons, New York, 1987
[35] Montgomery, D.C., "*Il Controllo Statistico della Qualità*", McGraw-Hill, New York, 2000
[36] Erto, P., "*Probabilità e statistica per le scienze e l'ingegneria*", McGraw-Hill, Italia, 2004
[37] Box, G.E.P.; Hunter, J.S. ; Hunter, W.G., "*Statistics for Experimenters*", Wiley & Sons, 2005
[38] Peter W.M.J. , "*Statistical Design and Analysis of Experiments*", Classics in Applied Mathematics, Vol. 22, 1998
[39] Taguchi. G., "*System of Experimental Design*", Kraus International Pub., New York, USA, 1987
[40] Wu, C.F.J., Hamada, M., "*Experiments. Planning, Analysis, and Parameter Design Optimization*", Wiley & Sons, 2000
[41] Rao S.,"*Engineering Optimisation Theory and Practice*", Wiley & Sons, 1996
[42] Miettinen K., "*Nonlinear Multiobjective Optimisation*", Kuwler Academic Publishers, 1999
[43] Caraceni A, De Cristofaro F, Ferrara F, Philipp O, Scala S "Benefits of using a real-time engine model during engine ECU development" *Proceeding of SAE* 2003
[44] S.Raman, N. Sivashankar, W. Milam, W. Stuart, S. Nabi, "Design and implementation of HIL simulators for powertrain control system software development", *Proceding of the American Control Conference*.
[45] W. Lee, M. Yoon, M. Sunwoo, "A cost- and time-effective hardware-in-the-loop simulation platform for automotive engine control systems", *Proceedings of the Institution of Mechanical Engineers*, Part D: Journal of Automobile Engineering 217 (1), pp. 41-52
[46] Z. Jiang, R. Dougal, R. Leonard, H. Figueroa, A. Monti, "Hardware-in-the-loop testing of digital power controllers", *IEEE Applied Power Electronics Conference and Exposition* - APEC, 2006, art. no. 1620645, pp. 901-906
[47] B. Lu, X. Wu, H. Figueroa, A. Monti, "A low-cost real-time hardware-in-the-loop testing approach of power electronics controls", *IEEE Transactions on Industrial Electronics*, 54 (2), pp. 919-931
[48] H. Li, M. Steurer, K. Shi, S. Woodru , D. Zhang, "Development of a unified design, test, and research platform for wind energy systems based on hardware-in-the-loop real-time simulation", *IEEE Transactions on Industrial Electronics* 55 (4), pp. 1144-1151.
[49] S. Karimi, A. Gaillard, P. Poure, S. Saadate, "FPGA-based real-time power converter failure diagnosis for wind energy conversion systems", *IEEE Transactions on Industrial Electronics* 55 (12), pp. 4299-4308.
[50] O. Lpez, J. lvarez, J. Doval-Gandoy, F. Freijedo, A. Nogueiras, A. Lago, C. Pealver, "Comparison of the FPGA implementation of two

multilevel space vector pwm algorithms", *IEEE Transactions on Industrial Electronics* 55 (4), pp. 1537-1547.

[51] S. C. Oh, "Evaluation of motor characteristics for hybrid electric vehicles using the hardware-in-the-loop concept", *IEEE Transactions on Vehicular Technology* 54 (3), pp. 817-824.

[52] W. Ren, M. Steurer, T. Baldwin, "Improve the stability and the accuracy of power hardware-in-the-loop simulation by selecting appropriate interface algorithms", *IEEE Transactions on Industry Applications* 44 (4), pp. 1286-1294.

[53] A. Bouscayrol, "Different types of hardware-in-the-loop simulation for electric drives", *IEEE International Symposium on Industrial Electronics*, art. no. 4677304, pp. 2146-2151.

[54] P. Gawthrop, D. Virden, S. Neild, D. Wagg, "Emulator-based control for actuator-based hardware-in-the-loop testing", *Control Engineering Practice* 16 (8), pp. 897-908.

[55] S. Ayasun, R. Fischl, S. Vallieu, J. Braun, D. Cadirli, "Modeling and stability analysis of a simulation-stimulation interface for hardware-in-the-loop applications", Simulation *Modelling Practice and Theory* 15 (6), pp. 734-746.

[56] M. Yoon, W. Lee, M. Sunwoo, "Development and implementation of distributed hardware-in-the-loop simulator for auto-motive engine control systems", *International Journal of Automotive Technology* 6 (2), pp. 107-117.

[57] W. Zhu, S. Pekarek, J. Jatskevich, O. Wasynczuk, D. Delisle, "A model-in-the-loop interface to emulate source dynamics in a zonal dc distribution system", *IEEE Transactions on Power Electronics*, 20 (2), pp. 438-445.

[58] A.Palladino, G.Fiengo, F. Giovagnini, D. Lanzo, "A micro hardware-in-the-loop test system", *IEEE European Control Conference*.

[59] F. Costanzo, M. D. Manes, G. D. Mare, F. Ferrara, A. Montieri, "Testing networked ECUs in a virtual car environment", *Proceeding of SAE* 2004-01-1724.

[60] H. Schuette, P. Waeltermann, "Hardware-in-the-loop testing of vehicle dynamics controllers a technical survey", *Proceeding of SAE* 2005

[61] F. Baronti, F. Lenzi, R. Roncella, R. Saletti, O. D. Tanna, "Electronic control of a motorcycle suspension for preload self-adjustment", *IEEE Transactions on Industrial Electronics* 55 (7), pp. 2832-2837.

[62] J. Du, Y. Wang, C. Yang, H. Wang, "Hardware-in-the-loop simulation approach to testing controller of sequential tur-bocharging system", *Proceedings of the IEEE International Conference on Automation and Logistics*.

[63] A. Cebi, L. Guvenc, M. Demirci, C. Karadeniz, K. Kanar, E. Guraslan, "A low cost, portable engine electronic control unit hardware-in-the-loop test system", *Proceedings of the IEEE International Symposium on Industrial Electronics*.

[64] T. Hwang, J. Rohl, K. Park, J.Hwang, K. Lee, L. K, K. Y.-J., "Development of hils systems for active brake control systems", *SICE-ICASE International Joint Conference.*

[65] A. Ali, A. Nadeem, M. Z. Z. Iqbal, M. Usman, "Regression testing based on uml design models", *IEEE International Symposium on Paci c Rim Dependable Computing.*

[66] B. Korel, G. Koutsogiannakis, L. Tahat, "Application of system models in regression test suite prioritization", *IEEE Inter-national Conference on Software Maintenance*, ICSM, art. no. 4658073, pp. 247-256.

[67] Y. Wu, M.-H. C. H. Kao, "Regression testing on object-oriented programs", *Symposium on Software Reliability Engineering*.

[68] S. Elbaum, H. Chin, M. Dwyer, M. Jorde, "Carving and replaying di erential unit test cases from system test cases", *IEEE Transactions on Software Engineering* 35 (1), pp. 29-45.

[69] M. Harrold, A. Orso, "Retesting software during development and maintenance", *Proceedings of the 2008 Frontiers of Software Maintenance, FoSM* 2008, art. no. 4659253, pp. 99-108.

[70] G. Rizzoni, P. Min, "Detection of sensor failures in automotive engines", *IEEE Transactions on Vehicular Technology*, Issue 2, Page(s):487 - 50.

[71] R. Conatsera, J. Wagner, S. Gantab, I. Walkerb, "Diagnosis of automotive electronic throttle control systems", *Control Engineering Practice*, Volume 12, Issue 1, Pages 23-30.

[72] Q. Butt, A. Bhatti, "Estimation of gasoline-engine parameters using higher order sliding mode", *IEEE Transactions on Industrial Electronics*, Volume 55, Issue 11, Page(s):3891 - 3898.

[73] A. Riegel, F. D. Cristofaro, I. Montaldo, "Interactive optimization methodology for robust base engine calibration", *EAC* 2007.

[74] A. Casavola, F. De Cristofaro, G. Fiengo, F. Garofalo, A. Palladino, "A Simulation Analysis for VVA and Idle Control Strategies", *18th Mediterranean Conference on Control and Automation* 2010

[75] F. De Cristofaro, G. Fiengo, A. Guzzo, A. Palladino, A. Palma "A Simulation based Investigation of Interaction between VVA and Idle Control for SI Engines", *Virtual Conference Control* 2010

[76] S. Bova, A. Casavola , F. De Cristofaro, A. Guzzo, I. Montalto, A. Riegel "Variable Valve Actuation: performance evaluation and optimization tools", *FISITA* 2010

[77] S. Bova, A. Casavola , F. De Cristofaro, A. Guzzo, I. Montalto, A. Riegel "Development of new algorithm for spark advance calculation for gasoline engine with variable valve actuation", *FISITA* 2010

[78] F. De Cristofaro, A. Riegel, E. Sepe "*Sistema per il suggerimento di un cambio di marcia per un autoveicolo provvisto di un motore e di un cambio di velocità a rapporti discreti*" patent number: EP2017505 (IT)

[79] F. De Cristofaro, A. Riegel, L. Tosato "*Metodo e dispositivo di stima della portata d'aria aspirata da un motore a combustione interna*", patent number: EP2055918 (IT)

[80] A. Zagrai, “Electro-mechanical analogies for modeling the structural impedance response”, *Proceedings of SPIE - The International Society for Optical Engineering*, No. 65320F, 2007
[81] B. J. Allison and A.J. Isaksson, “Design and performance of midranging controllers”, *Journal of Process Control*, Volume 8, Issue 5, pp. 469474, 2009
[82] S. Solyom and S. Eriksson, “Mid-Ranging Scheme for Idle Speed Control of SI Engines”, *SAE Technical Paper*, No. 2006-01-0608, 2006