2001 01 1865

ECU DEVELOPMENT AND TESTING THROUGH NUMERICAL OPTIMIZATION AND HARDWARE IN THE LOOP SIMULATIONS

Boatto Paolo, Monacchia Stefano, Vieri Giovanbattista
Ent S.r.l. – Italy

Alessandro Trifiletti
Dipartimento di Ingegneria Elettronica, UniversitàLa Sapienza - Italy

ABSTRACT

A method to speed up and improve the whole development cycle of Electronic Control Units (ECU) is presented.

The proposed procedure can be divided into three steps:

- development and validation of the controlled system dynamic model;
- numerical optimisation of the system control strategy;
- Hardware In the Loop (HIL) simulations to test the electronic control unit.

Implementation of the method has led to the creation of a hardware in the loop system aimed at testing the control unit of electronically controlled transmissions. The system includes a Digital Signal Processing (DSP) board where the dynamic model runs and a console allowing the user to operate the throttle and brake pedals and to actuate the gearshifts. The behaviour of the model can be visualised through a real time Graphical User Interface (GUI) on a PC connected with the DSP board by a serial link.

NOMENCLATURE

Symbols

c	damping
f	function
Fc	chain draught
Fl	rear fork length
FX	horizontal force
FZ	vertical force
g	acceleration of gravity
G	center of gravity
J	moment of inertia
k	stiffness
L	shock absorber's length
r	radius
R	engine-to-wheel ratio
s	slip
T	torque
u	command position
x	horizontal position
z	vertical position

Greek symbols

a	angle of rotation

Subscripts

br	brake
c	crankshaft, chain
cb	cylinder-block
cg	crown gear
cl	clutch
d	drag
f	front, frame, fork
gr	ground
e	engine
et	equivalent transmission
r	rear
s	suspension
sd	spring-drive
t	throttle
v	vehicle
w	wheel

INTRODUCTION

Electronic controls have been utilised more and more into vehicles during the last decade. If on board electronics is by now extensively adopted on cars of all categories, including small city cars, applications to the motorcycle field are much more recent and limited.

In particular electronic engine management has been adopted both on racing bikes and high-end sport tourers, the latter group being often equipped also with Antilock Braking Systems (ABS). Recently, also traction control systems (see e.g. Yamaha) and electronically controlled transmissions (see e.g. Ducati-Marelli) have made their appearance on racing bikes.

Speeding up the development and testing process of ECUs and similar equipment means significant reductions in cost and time to market. So far, though, the development of control systems has been mainly carried out by means of on-board testing on actual vehicles and long experimental campaigns.

The adoption of HIL systems, where the control unit interacts with a model of the controlled process running on a DSP board, drastically reduces costs and risk and allows testing to be carried out even before the completion of the actual vehicle.

On the other hand, model based optimisation of the control logics can improve performance and reduce failures.

ECU MODEL BASED DEVELOPMENT: A THREE STEP APPROACH

A three-steps approach to test control units behaviour in a simulated environment has been devised and then implemented (see figure 1), leading to the creation of a hardware in the loop system aimed at testing the control unit of an electronically controlled transmission.

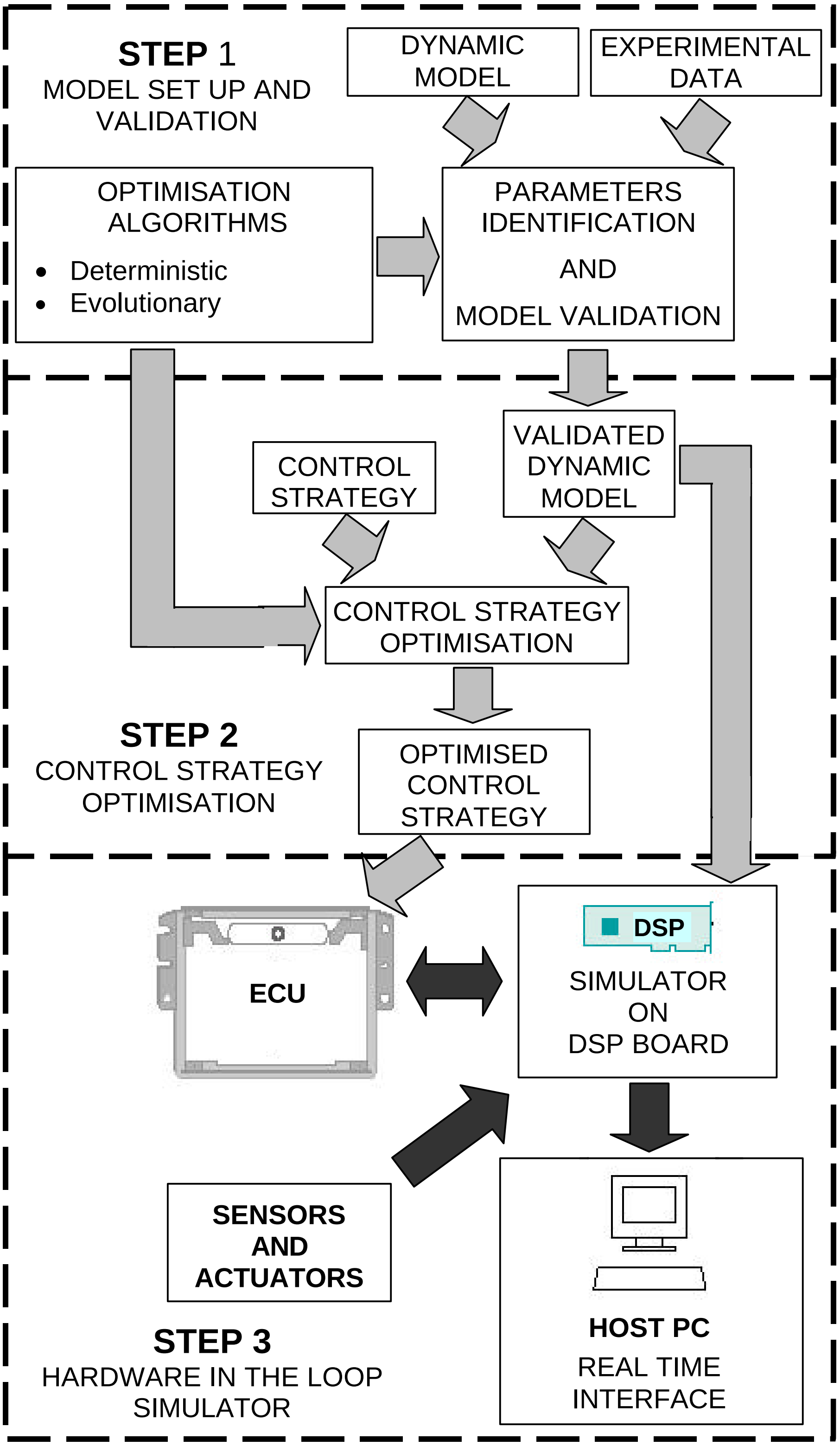

Figure 1 – Three step procedure.

First step of the method is the creation and validation of a virtual dynamic model of the system to be controlled (e.g. engine, transmission, braking system). The model is implemented in a suitable language for real-time applications (typically C) and can be validated by means of experimental data comparison. The identification of some parameters of the real system can be obtained through numerical constrained optimisation again by comparing model response to experimental data. This can be achieved by means of different techniques such as deterministic and evolutionary algorithms.

The second step involves the implementation and optimisation of the control system strategy to be used by the ECU. This can be achieved by means of the above mentioned optimisation techniques adopting proper objective functions and constraints.

The final step is the creation of the HIL system. Such a system includes a DSP board where the virtual model runs, interfaced with the ECU via CAN Bus and a host PC with a real time GUI. The DSP board is equipped with a data acquisition module connected to hardware actuators to let the user operate the model as the pilot would do with the real system (e.g. by use of throttle and brake commands or gearshift buttons). The GUI shows the operator the most significant parameters of the process (speed, rpm, gear, etc.), thus allowing a realistic "driving" simulation.

THE DYNAMIC MODEL

The above described approach has been applied to the electronically controlled transmissions. The dynamic model simulates the rectilinear motion of the vehicle, taking into account its heave and pitch degrees of freedom and the vertical shaking of the unsprung masses. The driveline has five degrees of freedom relevant to the cylinder-block, crankshaft, spring-drive, gearbox and final transmission (modelled as a single equivalent element) and wheel respectively (see figure 2).

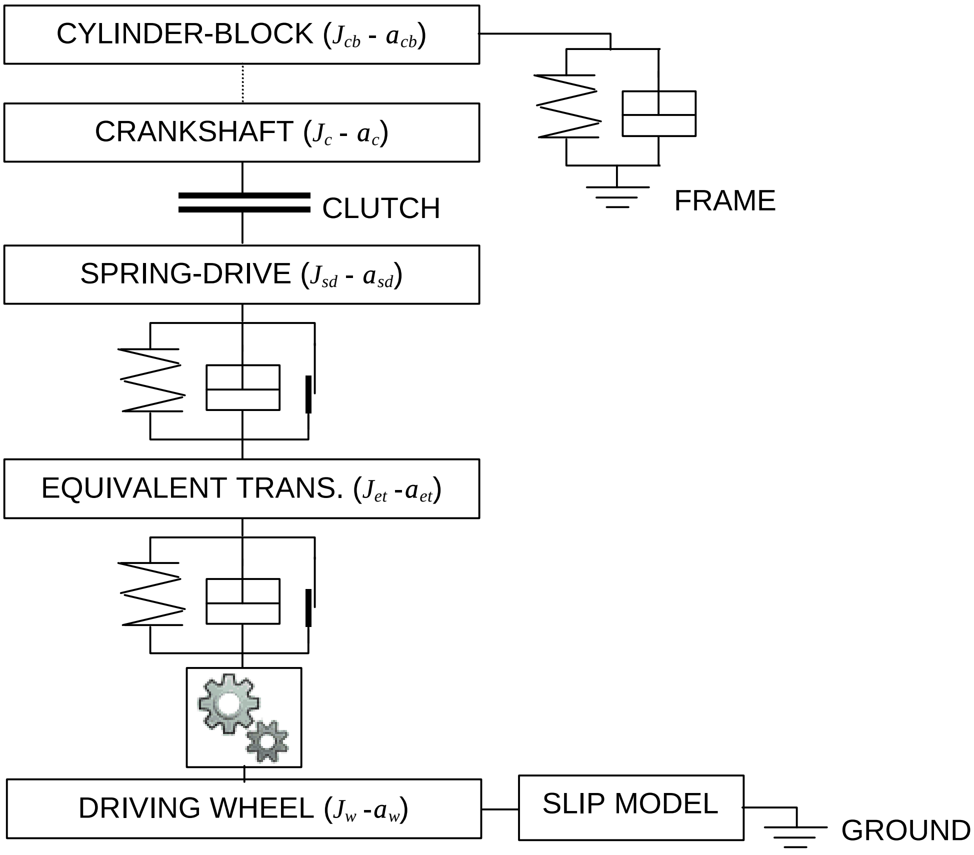
CYLINDER-BLOCK (J_{cb} - a_{cb})
CRANKSHAFT (J_c - a_c)
CLUTCH
FRAME
SPRING-DRIVE (J_{sd} - a_{sd})
EQUIVALENT TRANS. (J_{et} -a_{et})
DRIVING WHEEL (J_w -a_w)
SLIP MODEL
GROUND

Figure 2 – Scheme of the driveline model.

Driveline equations

Cylinder-block

The cylinder-block is subject to the engine reaction torque (-T_e) and to the elastic and damping reactions of its shock-isolating mountings.

(1) $$J_{cb}\ddot{a}_{cb} + c_{cb}\dot{a}_{cb} + k_{cb}a_{cb} = -T_e(\dot{a}_c, u_t)$$

Crankshaft

The actual crankshaft angle of rotation (a_c) is the sum of that of the cylinder-block (a_{cb}) and the relative rotation between crankshaft and cylinder-block (j).

(2) $$a_c = a_{cb} + j$$

The crankshaft is subject to the engine driving torque (T_e) and to the external load from the clutch (T_{cl}).

(3) $$J_c\ddot{a}_c = T_e(\dot{a}_c, u_t) - T_{cl}$$

The engine torque (T_e) is a function of crankshaft speed ($\dot{a}_c$) and throttle position (u_t), and takes into account both the torque resulting from the combustion and that due to the engine internal friction.

The clutch external load (T_{cl}) depends on the operating conditions; two clutch states can be recognized:

1. clutch is sliding (i.e. a relative angular velocity between the engine flywheel and the clutch disk exists or is going to appear). The external load is equal to the maximum torque the clutch can transmit that is given in input as a function of the clutch command position (u_{cl}):

   $$T_{cl} = T_{clMAX} = f(u_{cl}) \tag{4}$$

2. clutch is stuck (i.e. no macroscopic slipping exists between the engine flywheel and the clutch disk). The crankshaft and the spring-drive are treated as a unique shaft (i.e. a state is removed from the system) subject to the engine driving torque on one side and to the torque due to the spring-drive compression (T_{sd}) on the other side (see below).

Clutch is stuck when the two following conditions are contemporaneously satisfied:

1. the engine driving torque minus the torque due to the crankshaft inertia (net torque at the clutch disk) is lower or equal than the maximum torque the clutch can transmit;
2. the relative angular velocity between the engine flywheel and the clutch disk is near zero.

Spring-drive

When the clutch is sliding the following equation states the spring-drive dynamics:

$$J_{sd}\ddot{a}_{sd} = T_{cl} - T_{sd} \tag{5}$$

Torque due to the spring-drive compression (T_{sd}) takes into account the spring stiffness, the internal friction and damping; it is thus expressed by a non-linear function of the angular relative position (a_{et}-a_{ds}) and speed ($\dot{a}_{et} - \dot{a}_{sd}$) between the equivalent transmission and the spring-drive:

$$T_{sd} = T_{sd}\left(a_{et} - a_{sd}, \dot{a}_{et} - \dot{a}_{sd}\right) \tag{6}$$

When the clutch is stuck the crankshaft and spring-drive are treated as a single shaft thus equations (5) and (3) become respectively:

$$\left(J_c + J_{sd}\right)\ddot{a}_{sd} = T_e(\dot{a}_c, u_t) - T_{sd}\left(a_{et} - a_{sd}, \dot{a}_{et} - \dot{a}_{sd}\right) \tag{7}$$

$$\ddot{a}_c = \ddot{a}_{sd} \tag{8}$$

Equivalent transmission

The gearbox and the final transmission have been modelled as a single equivalent element since the natural frequencies of their components are much higher than the typical frequencies of the phenomena under study.

Such a simplified system is made of a flywheel connected to the spring-drive on one side and to an elastic shaft on the other side; the latter is linked to the wheel through a couple of gears providing the whole engine-to-wheel transmission ratio (R).

The flywheel inertia and the shaft torsional stiffness, damping, friction and slack are different for each gear and take into account the characteristics of all the elements of the gearbox (driving and driven shafts) and chain.

The dynamic equation relevant to the equivalent transmission is:

$$J_{et}\ddot{a}_{et} = T_{sd}\left(a_{et} - a_{sd}, \dot{a}_{et} - \dot{a}_{sd}\right) - k_{et}\left(a_{et} - \frac{a_w}{R}\right) - c_{et}\left(\dot{a}_{et} - \frac{\dot{a}_w}{R}\right) - h\frac{\left|\dot{a}_{et}\right|}{\dot{a}_{et}}\max\left[T_{sd}, k_{et}\left(a_{et} - \frac{a_w}{R}\right)\right] \tag{9}$$

The last term on the right hand side is the torque due to friction; it is assumed to be proportional (constant h) to the maximum value chosen between the torque due to the spring-drive and the elastic torque due to the shaft torsion.

Wheel

Driving wheel is subject to the transmission, ground and brake torques respectively:

$$J_w\ddot{a}_w = R\left[k_{et}\left(a_{et} - \frac{a_w}{R}\right) + c_{et}\left(\dot{a}_{et} - \frac{\dot{a}_w}{R}\right)\right] - T_{gr} - T_{br} \tag{10}$$

Ground torque (T_{gr}) is due both to the tyre tangential force due to friction and to the rolling resistance; the former is calculated by the product between the wheel radius (r_w) and the tangential force (FX_w). Such a force is calculated as a function (given as input) of the tyre slip (s_w) and it is proportional to the vertical force acting on the tyre (FZ_w):

(11) $$s_w = \frac{\dot{a}_w r_w - \dot{x}_G}{\dot{x}_G}$$

(12) $$FX_w = FZ_w f(s_w)$$

Passive torque due to the tyre rolling resistance is a function of wheel velocity and vertical load, while the brake torque (T_{br}) is a function of the brake command position (u_{br}).

Equations relevant to the front wheel don't include the transmission contribution (first term on the right hand side of eq. 10).

Vehicle equations

The scheme adopted for the vehicle dynamics study is given in figure 3.

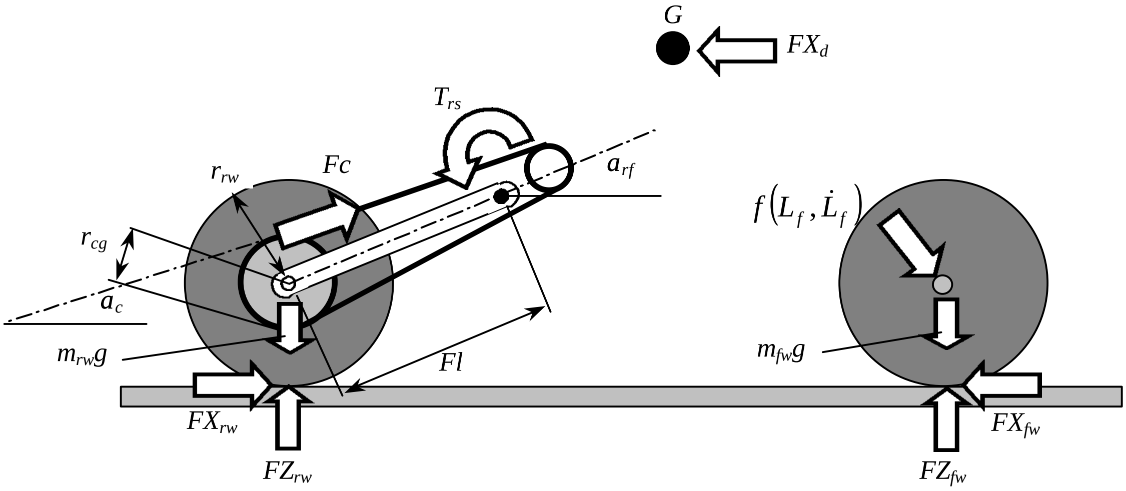

Figure 3 – Scheme of the vehicle model.

Sprung mass

Vehicle horizontal dynamics is influenced by the front (FX_{fw}) and rear (FX_{rw}) tangential forces acting on the tyres and by the aerodynamic drag (FX_d):

(13) $$m_v \ddot{x}_G = FX_{fw} + FX_{rw} - FX_d$$

Tangential forces are calculated by equation (12) while the drag is assumed to be a quadratic function of vehicle velocity.

Frame heave and pitch are influenced by the shock absorber compression/extension according to the two following concise equations (aerodynamic force and torque are neglected):

(14) $$m_f \ddot{z}_G = f(L_f, \dot{L}_f) + f(L_r, \dot{L}_r) + m_f g$$

(15) $$J_f \ddot{a}_f = f(L_f, \dot{L}_f) + f(L_r, \dot{L}_r) + z_G(FX_{fw} + FX_{rw})$$

The calculation of the shock absorber reactions $f(L_f, \dot{L}_f)$ and $f(L_r, \dot{L}_r)$ is carried out taking into account the suspension geometry (progressive linkage), a constant stiffness and a non-linear damping.

Unsprung masses

The motion of the unsprung masses is influenced by the shock absorber and tyre reactions and by the chain draught F_c (rear wheel only).

The dynamic equation of the front wheel vertical translation is:

(16) $$m_{fw} \ddot{z}_{fw} = f(L_f, \dot{L}_f) + FZ_{fw} + m_{fw} g$$

Ground vertical force (FZ_{fw}) is calculated from the front tyre deflection (that is a function of the front wheel pin vertical position z_{fw}), its derivative and by assuming constant tyre vertical stiffness and damping.

The dynamic equation of the rear wheel and fork rotation around the fork pin is:

(17) $$J_{rw} \ddot{a}_{rf} = T_{rs} + (m_{rw} g - FZ_{rw}) Fl \cos a_{rf} + FX_{rw}(r_{rw} + Fl \sin a_{rf}) - Fc\lfloor r_{cg} - Fl \sin(a_{rf} - a_c)\rfloor$$

From the rear suspension geometry and the shock absorber's performances the torque T_{rs} can be expressed as a function of a_{rf} and $\dot{a}_{rf}$. Ground vertical force (FZ_{rw}) is calculated as above described for the front wheel; the tyre deflection and its derivative are computed from the frame kinematics and the rear fork angle (a_{rf}). The chain draught (Fc) is calculated from the transmission and rear wheel rotational dynamics, while the chain angle (a_c) can be easily expressed as a function of the rear fork angle (a_{rf}).

A four steps Runge-Kutta scheme with a 1 [ms] time step is used to solve equations (1-17).

OPTIMISATION TECHNIQUES

Optimisation procedures can be used both to improve the system performances (optimisation of system design and control strategy) and to the identification of the system parameters needed for the model calibration (the direct measurement of some parameters is often very complex and expensive).

Deterministic algorithms are usually much faster than the evolutionary ones but sometimes they can be inadequate; in fact they are highly affected by the starting conditions and thus they can reach a local minimum instead of the global one. Evolutionary algorithms such as the Simulated Annealing (SA) and Genetic (GA) ones can avoid the entrapment in the local minima and they can manage a large number of optimisation parameters without being as time consuming as the fully random algorithms. Since they approach the global minima they are sometimes used to compute the starting solution to be used by a much faster gradient-based routine. The SA approach offers an iterative strategy that allows perturbation to move uphill in a controlled fashion in order to skip the local minima. GA is focused on a search method based on the mechanism of natural selection and genetics.

In the present work both deterministic and evolutionary techniques have been experimented, in particular a recursive quadratic programming (RQP) and the SA algorithms. In fact both SA and GA have shown a high degree of robustness and good performances but GA is also very sensitive to input parameters (number of mutations, elutriations, etc.).

HIL IMPLEMENTATION DETAILS

The HIL implementation and validation has been carried out using a commercial DSP stand-alone board based on the Texas Instruments TMS320F243 CPU. This approach has proven to be extremely efficient since it can provide both the number crunching capability required by the real time simulator (without the drawback related to massive operating system activities which could severely degrade the CPU performance), and fast IO lines.

The proposed control scheme (figure 1) sets the following constraints to the simulator engine:

1. minimal operating system activity overhead, in order to maximise the simulation efficiency and to respect the real time constraints;
2. assembler language software programming capability in order to better exploit the benefit of low level programming and routines optimisation;
3. on board analog to digital conversion capability, in order to avoid the need of sampling clock synchronisation and data exchange with external acquisition board;
4. processing speed compatible with the implemented discrete time differential equations scheme and the chosen simulation period;
5. fast digital IO lines to be used in the model tuning phase which is carried out in conjunction with the controller block.

The aforementioned features can be accomplished by choosing a proper DSP board as the main simulation engine and by a careful optimisation of the timing and synchronisation schemes between the control system blocks.

A PC has been used as a controller (instead of the ECU shown in figure 1) for the HIL testing. The control algorithms have been implemented into the PC and the dynamic model of the vehicle has been implemented into a TMS320F243 EVM DSP based co-processor board. The model running on the DSP uses an integration time step of 1 [ms] while it exchanges data with the PC at a frequency of 100 Hz (typical sampling rate of the transmission control units).

The above card is equipped with a data acquisition module thus the user can “drive” the model running on the DSP through the throttle and brake pedals and the buttons that actuates the gearshifts.

The behaviour of the model can be visualised through a Graphical User Interface on a second PC connected with the DSP board by a serial interface.

The datapath in HIL is shown in the following figure 4:

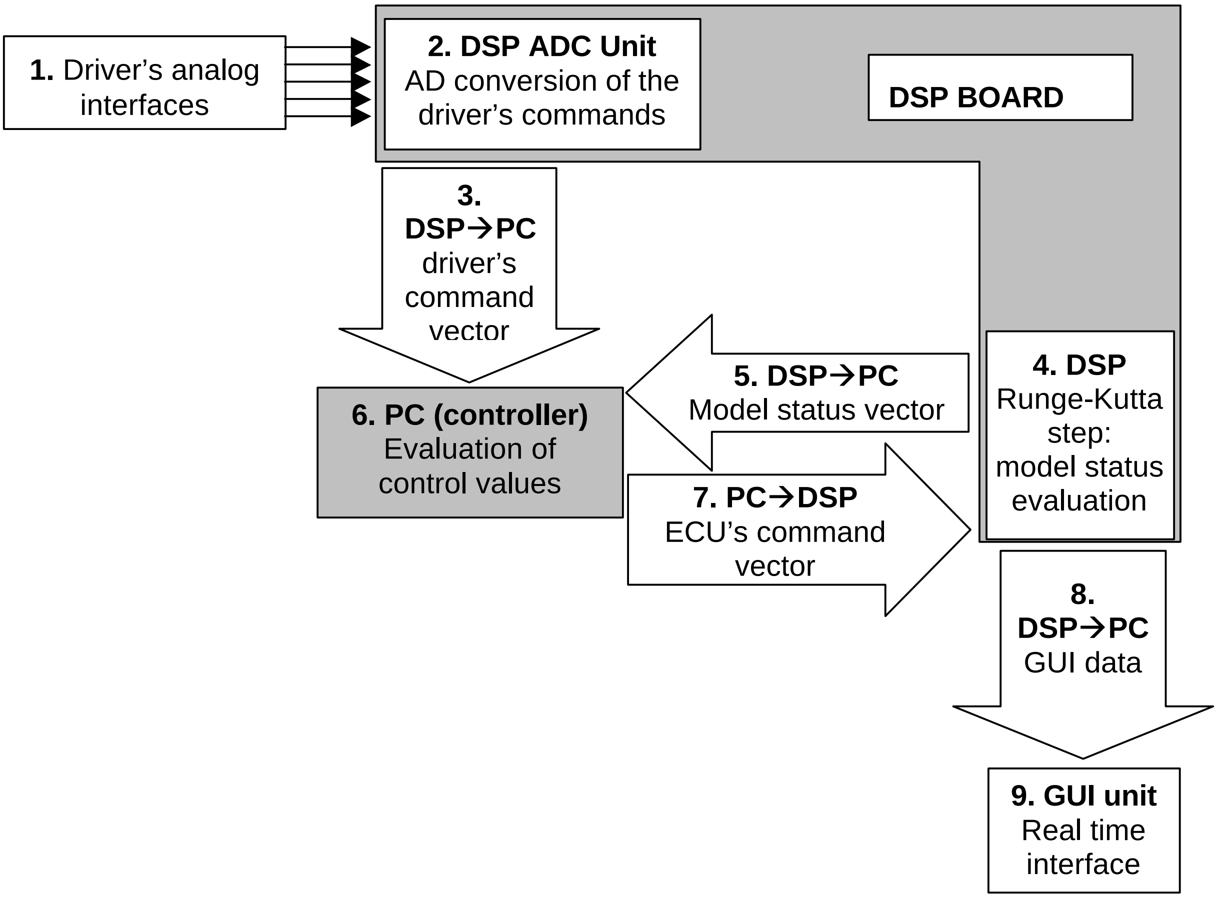

Figure 4 - Datapath in the present HIL system.

1. The driver operates the vehicle control commands (i.e. throttle, front and rear brakes, gearshift button);
2. the DSP board performs the analog signal acquisition and the analog-to-digital conversion;
3. the data vector (driver's command vector) resulting from the above conversion is sent to the PC that performs the control function (controller);
4. the DSP board computes the model actual state variables by numerical integration of the dynamic equation system;
5. the vector of the updated model state variables (model status vector) is sent to the controller;
6. the controller computes the new set of control commands (ECU's command vector);
7. the ECU's command vector is sent to the DSP board;
8. the status vector is addressed to a second PC performing the GUI function to allow the observation of the system dynamics.
9. the real time interface is updated on the GUI unit.

The time-critical data exchange have been performed in the present design using the CAN bus which can be accessed as a peripheral on the DSP board. To better use the bandwidth of serial IO lines of the DSP board an interleaved data exchange protocol has been implemented. Using this approach a significant part of the data are sent offline with respect to the main simulation loop, without affecting convergence of control and simulation algorithms.

EXAMPLES OF OFFLINE SIMULATIONS

Results relevant to the following two cases are here reported:

1. wide open throttle acceleration starting from about 100 [km/h] with the third gear engaged; two gearshifts (3°→4° and 4°→5°) are executed during the simulated time interval (15 [s]).
2. maximum acceleration from standing start; first gear is engaged during the whole simulated time interval (2 [s]).

Data supplied to the model are relevant to a high displacement sport bike.

Case 1

The normalised throttle and clutch command positions versus time during the two gearshifts are represented in figure 5; time scale spans the whole gearshift sequence. The old and the new gears are instantaneously disengaged/engaged (torques due to synchronization are neglected) at the same fixed instant (vertical lines) and the correct synchronized speed is assigned to the equivalent transmission flywheel.

The engine speed corresponding to the gearshift start (11676 [rpm] and 11465 [rpm], about 800 and 600 [rpm] beyond the maximum power speed) has been optimised to maximise the covered distance; both the simulated annealing and the recursive quadratic algorithms have reached the same results.

Vehicle horizontal kinematics is represented in figure 6. The two gearshifts are well visible from the acceleration behaviour; oscillations at the simulation start are due to the initial conditions.

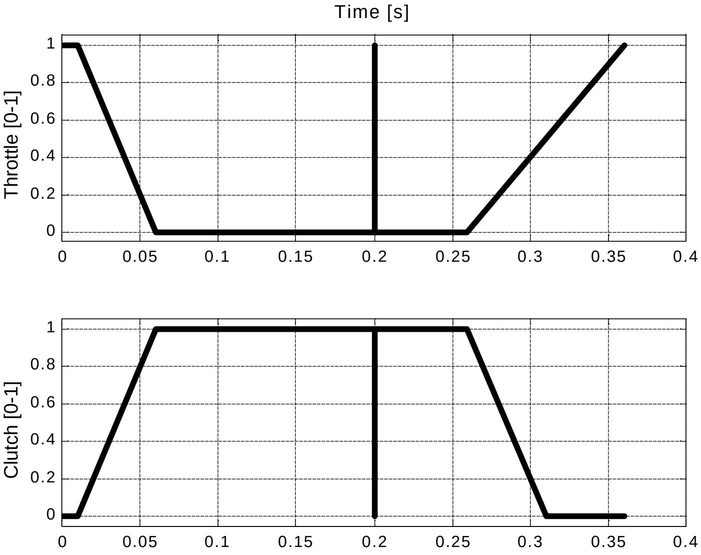

Figure 5 – Case 1: throttle and clutch command positions versus time during gearshifts.

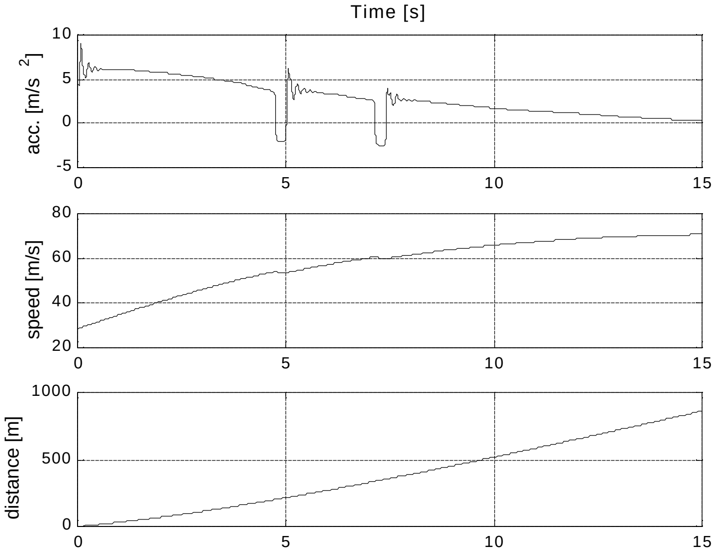

Figure 6 – Case 1: vehicle kinematics versus time.

Case 2

Throttle and clutch command positions during the whole simulated time interval have been optimised to maximise the covered distance. 31 points for each of the two commands have been fixed on the time scale 0-2 [s] (points are more dense near the origin); the normalised command positions corresponding to the above points have been used as optimisation variables. An inequality constraint on the front wheel vertical position has been introduced to avoid the vehicle overturn.

Results have been obtained by the simulated annealing algorithm since the deterministic method could not reach a significant solution. The latter aspect is due to the highly non-linear influence of the clutch command position on the objective function: when the clutch is stuck and it can transmit a torque notably higher than the engine one, small variations of the command position don’t affect the transmitted torque (it keeps stuck, thus it transmits the whole engine torque) and then the objective function. If during the iterations the above condition is reached, the calculated gradient of the objective function with respect to the clutch command position is zero and consequently the algorithm is no longer able to reach the optimal solution.

Figure 7 shows the normalised throttle and clutch command positions, and the parameter that sets the clutch operating conditions. The latter is equal to zero when the clutch is sliding while it is equal to one when it is stuck; from the figure it can be seen that the clutch slips almost all the time. Throttle and clutch commands changes rapidly, especially at the beginning of the simulation, to ensure the maximum acceleration of the vehicle and keep the front wheel on the ground. Figure 8 shows the vehicle horizontal kinematics.

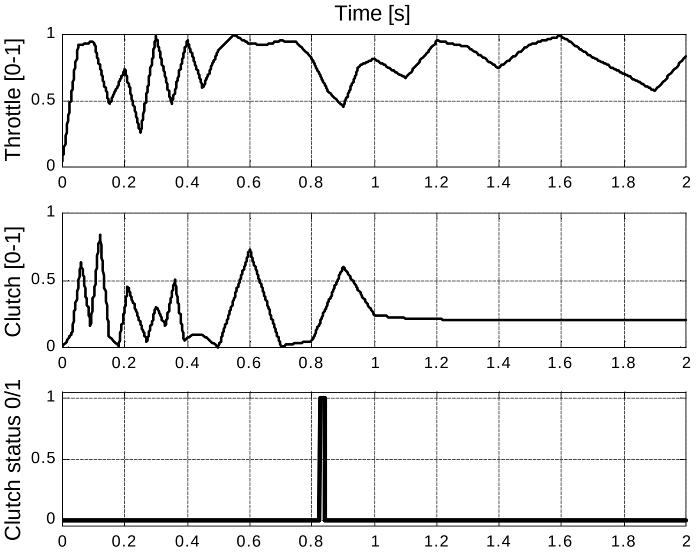

Figure 7 – Case 2: optimised throttle and clutch command positions, and clutch status versus time.

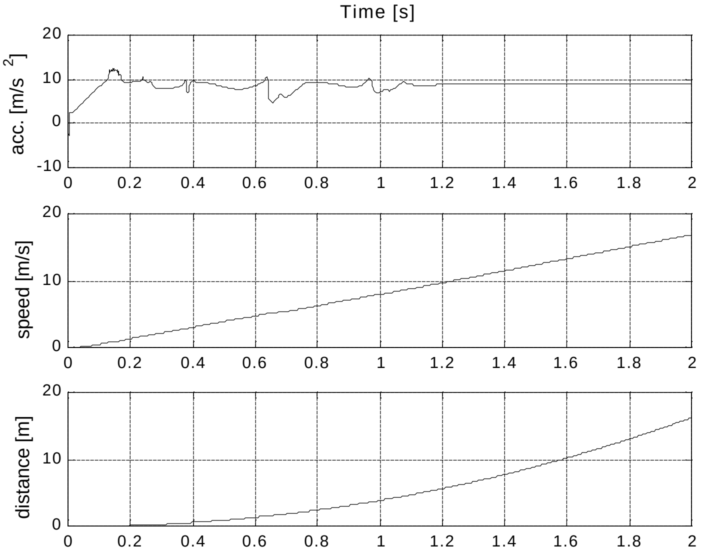

Figure 8 – Case 2: vehicle kinematics versus time.

CONCLUSION

In the present work the main phases of the implementation of a Hardware In the Loop experiment have been successfully exploited, assuming an electronically controlled transmission as a case study. The overall control problem has been split into three parts, which can be solved independently by each other. The model calibration and validation have been carried out using available data relevant to a high displacement sport bike. Optimisation algorithms based on evolutionary and deterministic techniques have been used to establish control strategies suitable for different targets. The HIL implementation has been optimised in order to use commercial electronic components thus simplifying and speeding up the system verification. The proposed approach appears to be general and it can be used as a general solution scheme for ECUs testing through HIL simulations.

REFERENCES

1. Cossalter, V., Cinematica e dinamica della motocicletta, Edizioni Progetto, 1999.
2. Kienke, U, Nielsen, L, Automotive Control Systems, Springer, 2000.
3. Jurgen, K. R., Electronic Transmission Controls, SAE PT-79, Society of Automotive Engineers, Inc., Warrendale, PA, 2000.
4. Automotive Handbook, Bosch, 4th Edition.
5. Milliken, W.F., Milliken, D.L., Race Car Vehicle Dynamics, SAE R-146, Society of Automotive Engineers, Inc., Warrendale, PA, 1995.
6. Goffe, Ferrier and Rogers, Global Optimization of Statistical Functions with Simulated Annealing, *Journal of Econometrics*, vol. 60, no. 1/2, Jan./Feb. 1994, pp. 65-100.
7. Goldberg D.E., Genetic Algorithms in Search Optimisation and Machine Learning, Addison-Wesley, 1989.