F2014-EPT-061

DESIGN AND IMPLEMENTATION OF A REAL-TIME SIMULATOR FOR HARDWARE-IN-THE-LOOP TESTING OF A HYBRID ELECTRIC BUS CENTRAL CONTROL UNIT

1Esfahianian, V.  2Esfandyari, M. J.  3Ha’iri Yazdi, M. R.  4Nehzati, H.
1 Vehicile, Fuel and Environment Research Institute (VFeri), University of Tehran, Tehran, Iran
2 School of Mechanical Engineering, University of Tehran, Tehran, Iran

KEYWORDS – Hybrid Vehicle; Hardware-in-the-Loop; Simulator; Central Control Unit

ABSTRACT – Due to the complexity in the design and cost involved with the testing of automotive control systems, Hardware-in-the-Loop (HiL) simulation plays a pivotal role in the development of such control units. To perform a HiL simulation, a vehicle simulator is needed which can virtually emulate the real vehicle in response to commands from the control unit. This paper presents the design and implementation of a comprehensive simulator for a series hybrid electric city bus which is suitable for HiL testing of the Central Control Unit (CCU). The simulation is implemented in LabVIEW environment. It gives user the ability to act as a real driver and thus, all control functions in different driving regimes can be tested and verified. The designed simulator is tested and validated using real test results and by applying appropriate inputs, it provides the ability to simulate a variety of scenarios that may be too difficult or expensive to perform on a vehicle prototype.

INTRODUCTION

Hybrid Electric Vehicles (HEVs) provide better energy efficiency and significantly reduced vehicle emissions compared to their conventional counterparts [1]. Due to the use of two or more power sources in HEVs, the control system for such a powertrain becomes much more complex than that of a conventional vehicle and this makes developing and testing such control systems more challenging [2].

The adoption of Hardware-in-the-Loop (HiL) simulation as an industry-standard testing process for the development of automotive control systems is a great change in the implementation process of new electronic control units [3], [4]. There are many published research results in the literature that have validated the HiL simulation approach as an efficient tool for control system development and verification [5]. For example, Wang et al. [6] used HiL simulation to test and verify the control functions and performance of the vehicle control unit of a series-parallel hybrid bus using PT-LABCAR as the HiL simulator. In Ref. [7], the control architecture of the hybrid ECU and also performance of the low-level ECUs has been evaluated with the HiL simulation. Also, using a HiL test bench, an energy management strategy for an electric vehicle has been developed and verified in [8], in which the real-time simulation was performed on a target PC using MATLAB xPC Target.

In order to build a HiL simulation platform, a real-time interaction between physical hardware and virtual simulations is needed [9]. Depending on the application, components of the HiL system may differ. Usually, embedded computers are used to run the models in real-time in order to decouple the real-time computing of the HiL system from the host PC [10]. This paper presents the design and implementation of a simulator for a series hybrid electric bus which is suitable for hardware-in-the-loop testing of the hybrid vehicle Central Control Unit (CCU). Design of the simulator is implemented in LabVIEW since it takes advantage ofprogramming multiple tasks that are performed in parallel and also is an extensive support for accessing instrumentation hardware. Using time step of 0.001 seconds, maximum error of one microsecond has been achieved for the elapsed time in each iteration of running which is acceptable for current research for the interaction between the simulator and the CCU. Dynamic behaviour of all components having a communication with the CCU are modelled in the simulator. The user can act as a real driver and experience any driving regime which may happen in reality without being involved with difficulties of testing on an actual vehicle. As a result, any control function which has been implemented in the CCU can be tested and verified and this considerably reduces the validation time. The vehicle which is used for current investigations is the O457 bus which has been converted to a series hybrid vehicle at Vehicle, Fuel and Environment Research Institute (VFeri) at University of Tehran. The simulation model has been validated using real test results. Results show that by applying appropriate inputs, the designed simulator can virtually emulate the real bus in response to commands from the CCU and therefore, it is suitable for HiL simulation. Structure of the paper is as follows. Section 2 describes the drive train of the hybrid bus and briefly introduces the CCU. In section 3, various parts of the designed simulator and their communication with the CCU is explained in detail and the simulation results are presented in section 4.
HYBRID ELECTRIC BUS DRIVETRAIN

Figure 1 shows the drive train of the O457 city bus which has been converted to a series hybrid electric vehicle in Vehicle, Fuel and Environment Research Institute (VFRI) at University of Tehran. The engine/generator set (Genset) consists of a 4-cylinder diesel engine which is coupled to a three-phase generator by means of a constant ratio gearbox in order to keep the consistency of their speeds. The three-phase generator is connected to the high voltage bus using an inverter. Two coupled AC motors drive the vehicle and each has its own inverter. As a result, when the required power is low, one of the traction motors can be switched off and this increases the efficiency of the system [11]. The traction motor can be controlled either as a motor or a generator, and either in forward or reverse motion. The regenerated power from traction motors during braking can be stored in the batteries. For feeding the accessories, another electrical motor and inverter are considered. Each of the mentioned parts of the hybrid bus uses a separate control unit to manage its operation and the CCU is a vehicle-level controller which controls the operation of different components by sending appropriate commands to their local controllers and manages the interaction between these control units. The connection between bus components is made via CAN (Controller Area Network) messages. The CCU is an implementation of Vehicle Control Software (VCS) which consists of five parts: Start Switch Manager (SSM), Drive Motor Management Software (DMMS), Auxiliary Motor Management Software (AMMS), Hybrid Control Software (HCS) and Hybrid Interfacing System (HIS). Figure 2 shows the structure of the VCS and its interface with vehicle components.

![Figure 1: Hybrid electric bus configuration](image1)

![Figure 2: Structure of the VCS and its communication to vehicle components](image2)

THE SERIES HYBRID ELECTRIC BUS SIMULATOR

To perform a HiL simulation, any vehicle behaviour that affects the monitored performance of the control unit needs to be modelled in the simulator. To design such a simulator, input/outputs of the subsystems in the simulator should be identified according to the CAN messages of each component in the real vehicle, especially messages received from the CCU. The interface with the CCU and CAN messages for each subsystem are depicted in Figure 3. As shown, the designed simulator is comprised of six subsystems: Energy Storage System (ESS), Power Distribution System (PDS), Power Generation System (PGS), Traction System (TS), Vehicle and
Auxiliary System (AS). In Figure 4, subsystems of the designed simulator and the input/outputs they transfer to each other is shown.

**Figure 3:** Relevant CAN messages of each part of the simulator

**Figure 4:** Different parts of the hybrid bus simulator and their communication to each other

In the current research, high precision real-time simulation model is not required. Hence, design of the simulator is implemented in LabVIEW environment and the frequency of “while loops” is controlled in such a way that maximum error of one microsecond has been achieved for the elapsed time in each iteration of running which is acceptable for the interaction between the simulator and the CCU. The following subsections give descriptions of each subsystem of the designed simulator.

I. ENERGY STORAGE SYSTEM

The batteries and the Battery Management System (BMS) are modelled in this part. As shown in Figure 3 and Figure 4, the ESS simulator receives charge/discharge current and status of battery contactor and calculates the actual battery current, terminal voltage, SoC and charge/discharge correction coefficients. Since we want to check the performance of the CCU, a detailed battery model may not be required. So, a simple model composed of a voltage source and a resistance is used as the battery model. The open-circuit voltage and the internal resistance in the equivalent circuit are functions of SoC (Figure 5 and Figure 6).
The SoC of the battery is calculated using the following equation:

\[
\text{SoC} = \text{SoC}_0 + \int_0^t \frac{i(t)}{Q} dt
\]

(1)

where \( Q \) and \( \text{SoC} \) are the capacity and initial state of charge of the battery, respectively. When battery pack contactor is close, battery current is zero and the battery terminal voltage equals the open-circuit voltage. When over charge/discharge or over/under voltage conditions occur, BMS uses PI controllers to limit the demanded torque from the traction motors. Block diagram of the designed close-loop control systems for one of these cases is shown in Figure 7. The PI controller output, which is limited to interval \([0, 1]\), is sent to the CCU and known as charge/discharge correction coefficient. The limited torque is then received from the CCU and the traction motors current is added to the current of other components and fed back to the ESS. Table 1 shows gains of the designed PI controllers.

<table>
<thead>
<tr>
<th>Conditions</th>
<th>Proportional Gain</th>
<th>Integral Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>over-discharge</td>
<td>0.001</td>
<td>0.008</td>
</tr>
<tr>
<td>over-charge</td>
<td>0.3</td>
<td>0.3</td>
</tr>
<tr>
<td>under-voltage</td>
<td>0.001</td>
<td>0.3</td>
</tr>
<tr>
<td>over-voltage</td>
<td>0.1</td>
<td>0.001</td>
</tr>
</tbody>
</table>

II. POWER DISTRIBUTION SYSTEM

The PDS provides the requested electric energy of various components of the vehicle from the batteries. It consists of a contactor box and some safety components to prevent or alert improper operation of the electrical components. As shown in Figure 3, the PDS simulator receives contactors on/off command from the CCU and after ensuring the safe operation of the vehicle components, it gives the commands to relevant parts of the simulator. Also, by receiving currents of various subsystems, the PDS simulator calculates the charge/discharge current of the batteries using Kirchhoff's Current Law (KCL).
III. POWER GENERATION SYSTEM

The PGS consists of the diesel engine, a constant ratio gearbox, the generator and the inverter. The HCS determines when and how to turn on or turn off the Genset according to the applied control strategy. According to the control mode of the engine and generator, each can receive torque or speed references from the CCU (see Figure 3). Block diagram of the designed model for the PGS is depicted in Figure 8. Design of a model which can exactly simulate the dynamic behaviour of the engine and generator is difficult and together with other parts of the simulator makes the real-time simulation hard. Hence, to simulate the output torque of the engine and generator, two PID controllers are designed in the case that the engine or generator is in speed-control mode in which output of the controller determines the torque produced by each component. Table 2 gives gains of the designed PID controllers. Depending on the sign of the generator reference torque, two controllers are designed for the engine. In order to simulate the dynamic process to build-up the commanded torque, a response time is introduced for the engine and generator. In order to taking into account the engine power losses due to friction, etc., characteristic maps of the engine have been used which gives the power losses vs. the engine speed. Also, the engine fuel consumption is calculated using the engine characteristic maps.

For each of the inverters (i.e. inverters of the traction motors, auxiliary motor and generator), a contactor closes the current flow from the batteries and when it is open, charging of a capacitor in the inverter starts. An RC circuit is used to simulate this precharge process (Figure 9). When the capacitor charging process finished, the precharge contactor is closed and the main contactor is opened and the required electrical energy of vehicle components is provided from the batteries.

<table>
<thead>
<tr>
<th>Component</th>
<th>Proportional</th>
<th>Integral</th>
<th>Derivative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Engine No. 1</td>
<td>3</td>
<td>0.01</td>
<td>1.3</td>
</tr>
<tr>
<td>Engine No. 2</td>
<td>12</td>
<td>5</td>
<td>2.2</td>
</tr>
<tr>
<td>Generator</td>
<td>5</td>
<td>7</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Table 2: Engine and generator PID controllers in speed-control mode

Figure 8: Block diagram of the PGS simulator

Figure 9: The equivalent RC circuit for simulating the precharge process

IV. TRACTION SYSTEM

Two traction motors with their inverters make the TS. The accelerator pedal or the brake pedal pressure is converted to torque command in the CCU. The TS model receives the torque command from the CCU and applies a response time to build up the received torque. The output torque is limited to the maximum allowable torque using the characteristic maps of the motor and the produced mechanical power is delivered to the wheels to drive the vehicle. For forward motion, positive and negative torque references indicate the acceleration and deceleration (in the case of regenerative braking) conditions, respectively. For the backward motion it is reversed.

V. VEHICLE

This part consists of two submodels: Driver and Vehicle Dynamics (VD). Driver commands include acceleration and brake pedals, regenerative brake permission switch, electric vehicle mode switch, retarder position, hand
brake and forward or backward motion request (Figure 10). Vehicle longitudinal dynamics is solved in VD to obtain vehicle and traction motors speed. Electric motors tractive force has to overcome the opposing forces of aerodynamic drag, rolling resistance and road grade force as illustrated by the following equation:

$$\delta M \frac{dv}{dt} = F_t - F_m - \frac{1}{2} \rho C_D A_f v^2 - Mg \sin \theta - Mg f_r \cos \theta : (v > 0)$$

(2)

$$\delta M \frac{dv}{dt} = F_t + F_m + \frac{1}{2} \rho C_D A_f v^2 - Mg \sin \theta + Mg f_r \cos \theta : (v < 0)$$

(3)

where $M$ is the bus total mass, $F_{mb}$ indicates the mechanical brake force, $A_f$ is the frontal area of the vehicle, $f_r$ is the rolling resistance coefficient, $\Theta$ indicates the road slope, $C_D$ is aerodynamic drag coefficient and $\delta$ is the mass factor that equivalently converts the rotational inertias of rotating components into translational mass [12]. Backward motion of the vehicle is also considered in the simulation. When moving forward, the tractive force ($F_t$) is positive during acceleration and it becomes negative in the case of regenerative braking and for backward motion, signs are reversed. Solving equation (2) and (3), vehicle speed and therefore the angular velocity of traction motors can be obtained. Table 3 shows the vehicle data.

Table 3: Vehicle data

<table>
<thead>
<tr>
<th>Vehicle data</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Effective mass</td>
<td>11500 kg</td>
</tr>
<tr>
<td>Frontal area</td>
<td>6.75 m²</td>
</tr>
<tr>
<td>Drag coefficient</td>
<td>0.79</td>
</tr>
<tr>
<td>Rolling resistance coefficient</td>
<td>0.01</td>
</tr>
<tr>
<td>Wheel radius</td>
<td>0.466 m</td>
</tr>
<tr>
<td>Overall gearbox and differential efficiency</td>
<td>0.9</td>
</tr>
<tr>
<td>Overall gearbox and differential speed ratio</td>
<td>17.415</td>
</tr>
</tbody>
</table>

VI. AUXILIARY SYSTEM

As mentioned before, according to the applied control strategy, sometimes the engine is off. Therefore, an auxiliary electric motor is needed to drive components like compressor, oil pump, water pump, etc. The AS consists of the auxiliary electric motor and its inverter. The designed model for AS receives speed command from the CCU and the input voltage from the ESS and calculates the actual speed, actual torque and also current.
of the auxiliary electric motor. To simplify the simulation, a constant load is defined for the auxiliary components and a PID controller is designed to simulate the output torque of the auxiliary motor. The designed proportional, integral and derivative gains of the controller are 1, 2 and 0.1, respectively. Block diagram of the auxiliary system is depicted in Figure 11.

![Block diagram of the auxiliary system](image1)

Figure 11: AS block diagram

**RESULTS**

To show the precision of real-time simulation for the designed simulator, for time steps of one millisecond, error in the duration of each iteration compared to exact real-time simulation is calculated for the designed simulator and plotted versus time in Figure 12. As shown, maximum error is about one microsecond and is decreasing as the time goes on. This error in real-time simulation is acceptable for the current research. For validating the designed simulation model, some speed references to the engine through the ADM21, results are saved and compared with results of the PGS simulator. Figure 13 shows results of this test. Also, by applying a charge/discharge current according to Figure 14, the battery terminal voltage is shown in Figure 15. These results illustrate the degree of alignment between the real and simulated results.

![Error vs. time](image2)

Figure 12: Each iteration duration error compared with exact real-time simulation (time step=1ms)

![Engine speed](image3)

Figure 13: Engine speed in PGS validation test

![Charge/discharge current](image4)

Figure 14: Charge/discharge current in ESS validation test

![Battery terminal voltage](image5)

Figure 15: Battery terminal voltage in ESS validation test

In order to show the performance of the designed simulator, some inputs are applied to imitate the CCU commands and the simulator results are shown. As a sample, by applying a reference torque to the traction motor, the actual torque, dc current and speed of traction motor are shown in Figure 16 to Figure 19. As shown,

---

1 Adaptation Module (the engine CAN interface)
by applying negative reference torque in forward motion, which indicates the regenerative braking condition, the vehicle starts to decelerate and keeping to apply this negative torque, vehicle starts to move backward. This result show that it is possible to check the CCU operation in the case of regenerative braking when the vehicle speed falls below a predefined value due to the braking condition and this may lead to vehicle backward motion due to wrong traction motor reference torque. When the reference torque becomes zero, vehicle speed gradually reaches zero by the effect of resistance forces. These and other verification results show that the designed simulator can virtually emulate the real vehicle in response to commands from the CCU and can be used in a HiL test bench to test the performance of the CCU.

CONCLUSION

HiL simulation has been increasingly used for testing and verification of complex electronic control units. This paper presents the design and implementation of a comprehensive simulator for a series hybrid electric city bus which is suitable for HiL testing of the Central Control Unit (CCU). Design of the simulator is performed in LabVIEW environment and any vehicle behaviour that affects the monitored performance of the CCU is modelled in the simulator. Results show that the designed vehicle simulator can emulate the behaviour of the real vehicle in response to commands from the CCU. Using this simulator, any control function which has been implemented in the CCU can be tested and verified in a HiL test bench and this in turn leads to a simple and cheap way of testing such control units. It is possible to immerse some of the other physical components in the HiL test bench by replacing the input/outputs of the relevant simulation models with the CAN messages of the real hardware which is a matter for future research.

ACKNOWLEDGMENT

This work was supported by the Vehicle, Fuel and Environment Research Institute (VFERI) of the University of Tehran.
REFERENCES


