Chapter 1

Model Based System Design for Electric Vehicle Conversion

Ananchai Ukaew

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.77265

Abstract

Development of electric vehicle (EV) conversion process can be implemented in a low-cost and time-saving manner, along with the design of actual components. Model-based system design is employed to systematically compute the power flow of the electric vehicle propulsion and dynamic load. Vehicle specification and driving cycles were the two main inputs for the simulation. As a result, the approach is capable of predicting various EV characteristics and design parameters, such as EV performance, driving range, torque speed characteristics, motor power, and battery power charge/discharge, which are the necessity for the design and sizing selection of the main EV components. Furthermore, drive-by-wire (DBW) ECU function can be employed by means of model-based design to improve drivability. For the current setup, the system components are consisted of actual ECU hardware, electric vehicle models, and control area network (CAN) communication. The EV component and system models are virtually simulated simultaneously in real time. Thus, the EV functionalities are verified corresponding to objective requirements. The current methodology can be employed as rapid design tool for ECU and software development. Same methodology can be illustrated to be used for EV tuning and reliability model test in the future.

Keywords: EV conversion, model-based system design, drive-by-wire ECU, real-time application, in-the-loop testing, rapid control design, ECU network, CAN protocol

1. Introduction

Development in EV conversion has been vastly improved in the recent year. However, different vehicle models have different technical specifications, so conversion kits for each one of them have to be customized in order to meet the specific requirement such as range per charge and acceleration performance. Engineers, therefore, have to make the decision on the capacity of batteries and also how many of them are required to meet the driving demand. Moreover, selection of different types of motor is also presented as the main requirement [1]. Normal design process would require high-end expensive software to model the EV system. Furthermore, building the EV without the knowledge of the parameters within the system could costly lead to the failure of the design.

In addition, poor vehicle performance safety and reliability might occur when new electric propulsion characteristics do not match with the characteristics of replaced engine sharing the same chassis.

Therefore, a sub-ECU must be developed to harmonize EV propulsion dynamics and existing vehicle chassis characteristics called drive-by-wire (DBW) [2]. DBW functionality can then improve EV drivability by providing power demand to the electric motor drive according to the driver preference. However, installation of the DBW ECU without appropriate functional safety design and evaluation could induce such system failures or component malfunctions due to unpredicted behaviors during actual driving situations. Therefore, during the initial development process, ECU functions are needed to be established and evaluated against design and functional safety aspects beforehand [3, 4].

To improve EV conversion development process, model-based design process is shown in **Figure 1**. The method would benefit the design engineer in making better decision for the conversion and also saving time and cost by reducing error during the design process [5–7]. The process can be employed to perform system simulation based on different scenarios and technical specification. Embedded system and DBW ECU can be realized by software rapid auto coding to shorten error correction and debugging time. Virtual prototyping test can be employed to validate design requirement and EV conversion specification. The in-the-loop tests can ensure accurate implementation of both software and hardware ECU for the conversion using real-time verification methodology.

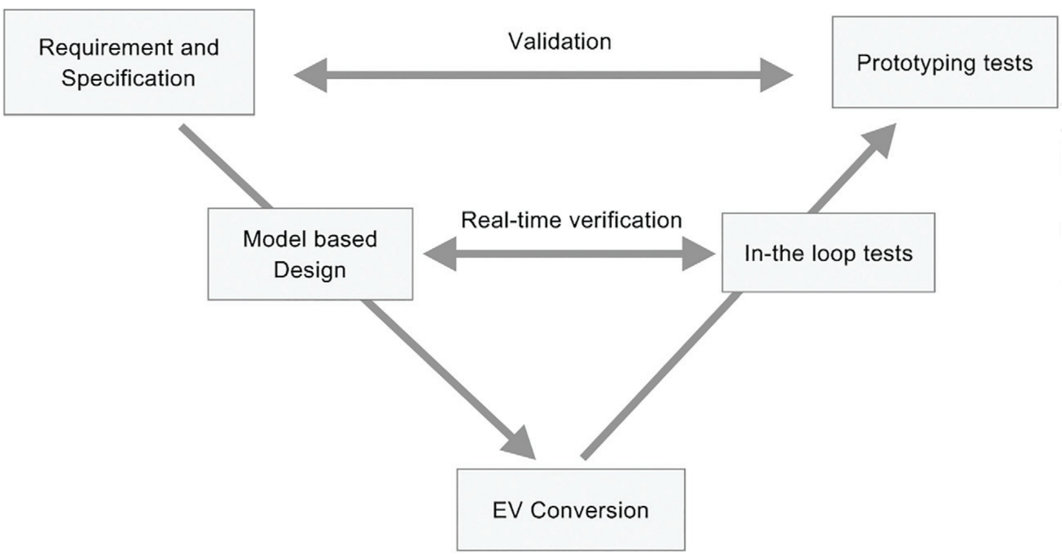

Figure 1. Model-based design process for EV conversion.

http://dx.doi.org/10.5772/intechopen.77265

In this literature, the first model-based design for EV conversion prototyping development, which describes electric vehicle modeling including EV traction, EV components, and power flow models, is defined. Then, drive-by-wire ECU design and in-the-loop testing for EV conversion process are described in details. The last section illustrates versatility of model-based design in EV conversion tuning and diagnostic application.

2. EV conversion prototyping development

2.1. EV system modeling

In order to set up the simulation of EV, mathematical models have to be generated first from the engineering principles and theories. The four core models are traction model, motor model, battery model, and power flow model as follows.

2.1.1. Traction model

Forces acting on the vehicle govern the equation for vehicle traction as seen in **Figure 2**. Those forces comprised of tractive forces (F_{te}), rolling resistance force (F_{rr}), aerodynamic force (F_{ad}), lateral acceleration force (F_{la}), wheel acceleration force (F_{wa}), hill climbing force (F_{hc}) [or component force of vehicle weight which depend on grade (θ)], and the gross weight of the co EV (mg).

The governing relation can be found in Eq. (1) where traction needs to overcome the load that is equal to five other forces:

$$F_{te} = F_{rr} + F_{ad} + F_{hc} + F_{la} + F_{wa} \tag{1}$$

where equation for each force components can be employed from many sources such as reference [2, 4] and other automotive textbooks.

2.1.2. Motor efficiency model

In the EV conversion system, the motor replaces the internal combustion engine (ICE) in providing the torque to drive the wheel as shown in **Figure 3**, which also affects the traction

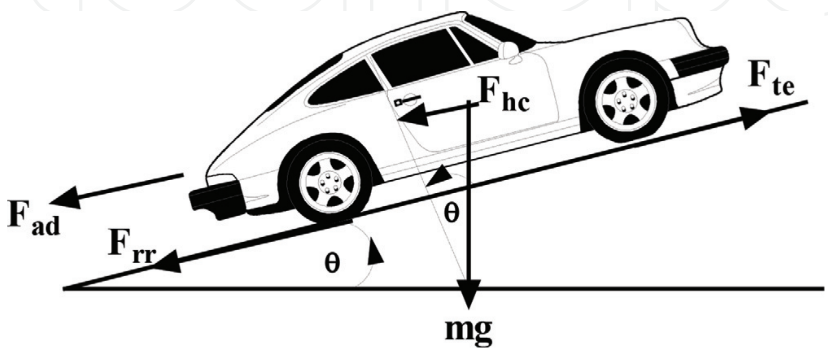

Figure 2. The force components involved in the vehicle traction.

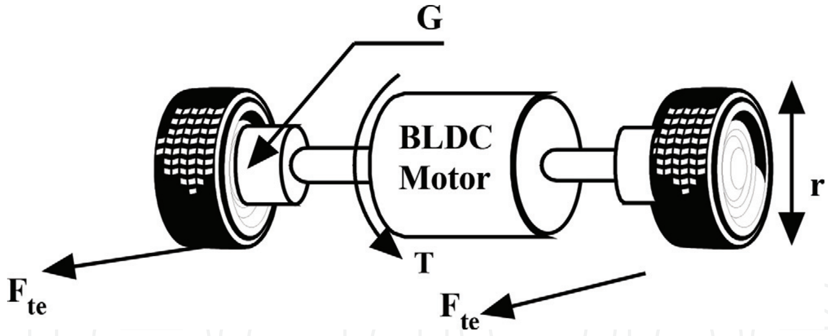

Figure 3. The EV motor provides the traction to the vehicle through transmission.

of the vehicle. The motor torque, speed, and efficiency equation are presented in Eqs. (2), (3), and (4), respectively.

- Motor torque (T)

$$T = \frac{F_{te} r}{G} \quad \text{N.m} \tag{2}$$

- Motor angular speed (ω)

$$\omega = G \frac{v}{r} \quad \text{rad.s}^{-1} \tag{3}$$

- Motor efficiency (η_m)

$$\eta_m = \frac{T\omega}{T\omega + k_c T^2 k_i \omega + k_\omega \omega^3 + C} \tag{4}$$

where k_c is copper losses coefficient, k_i is iron losses coefficient, k_ω is windage loss coefficient, and C is constant loss applied at any speed [6].

2.1.3. Battery discharge model

Battery's dynamic behavior does have a great effect on EV performance and range. Three common types of batteries, which are lead acid, nickel cadmium, and lithium ion batteries, are governed by Eqs. (5), (6), and (7) here, respectively. As seen in [4, 6], open-circuit voltage (E) of the batteries is changed as the state of charge changes and is calculated for each battery type below:

- Lead acid:

$$E = n \cdot \left(2.15 - DoD \cdot [2.15 - 2.00]\right) \tag{5}$$

http://dx.doi.org/10.5772/intechopen.77265

- Nickel cadmium:

$$E = n \cdot \begin{pmatrix} -8.2816DoD^7 + 23.5749DoD^6 - 23.7053DoD^4 - 12.5877DoD^3 + \\ 4.1315DoD^2 - 8.65DoD + 1.37 \end{pmatrix} \tag{6}$$

- Li-ion:

$$E = n \cdot 3.3 \tag{7}$$

(nominal cell voltage = 3.3 V up to 80% *DoD*)

where n is the number of cells and DoD is depth of discharge (0–1).

The open-circuit voltage also affects the battery current (I_B) in both states of charge and discharge as seen in Eqs. (8) and (9).

- Battery current discharge operating at power (P_{bat})

$$I_B = \frac{E - \sqrt{E^2 - 4RP_{bat}}}{2R} \tag{8}$$

- Battery current charge during regenerative braking

$$I_B = \frac{-E + \sqrt{E^2 + 4RP_{bat}}}{2R} \tag{9}$$

where R is the battery resistance. Due to Peukert phenomenon [6], therefore it is necessary to take into account such effect by adding the power to the k value, such as lead acid battery ($k \approx 1.12$) and Lithium ion ($k \approx 1$), when simulation of battery discharge is performed. Battery capacity (CR) is updated for each time step (δt) as shown in Eq. (10) and then used to update the depth of discharge (DoD) in Eq. (11) for discharging state and in Eq. (12) for charging state as following:

- Total charge removed from battery by the *n*th step of the simulation

$$CR_{n+1} = CR_n + \frac{\delta t \cdot I_B{}^k}{3600} \tag{10}$$

- The depth of discharged

$$DoD_n = \frac{CR_n}{C_p} \tag{11}$$

- Charge removed for regenerative braking

$$CR_{n+1} = CR_n - \frac{\delta t \cdot I_B}{3600} \tag{12}$$

2.1.4. EV conversion system power flow model

To complete the simulation, the integrated power flow model is necessary to compute and update the rate of energy going in and out of battery cells, accessories, the motor, gearing components, and wheel to the road and back. Therefore, the model needs to be capable of mathematically simulating the power flow in both driving and braking as shown in **Figure 4**.

Traction model provides the power flow between the vehicle and the road (P_{te}) as shown in Eq. (13). Furthermore, the motor model provides the power going in for both driving and braking mode at the motor/battery connection (P_{mot_in}) and at the motor/transmission connection (P_{mot_out}) as indicated in Eqs. (14) and (15). The power parameters are affected by the motor efficiency (η_m) and the gearing efficiency $\left(\eta_g\right)$. The battery power is also computed and updated (Eq. (16)) during charge and discharge operation using the battery model. Power (P_{ac}) is constantly drawn out of battery due to the use of accessories, such as car stereo and light, which is accounted in Eq. (16) [6, 8].

- Energy required per second:

$$P_{te} = F_{te} \cdot v \tag{13}$$

- Motor power in driving mode:

$$\begin{aligned} P_{mot\_in} &= \frac{P_{mot\_out}}{\eta_m}, \\ P_{mot\_out} &= \frac{P_{te}}{\eta_g} \end{aligned} \tag{14}$$

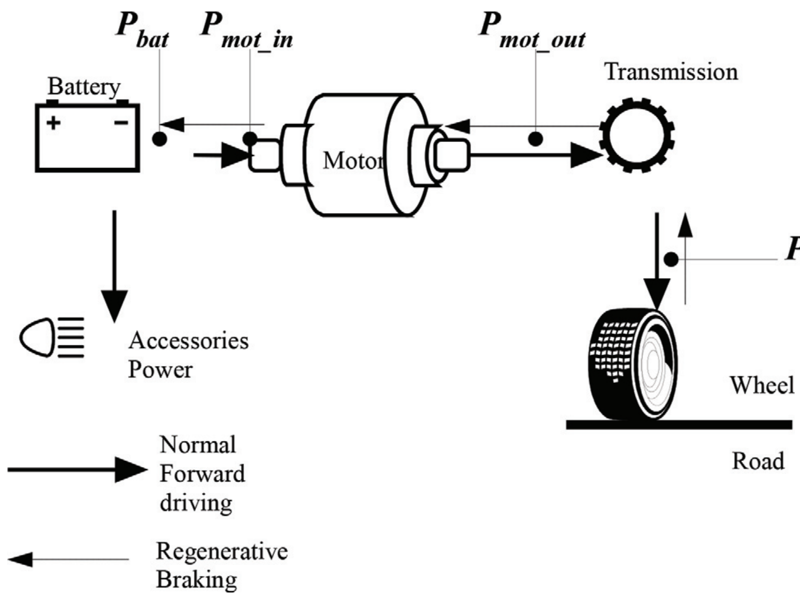

Figure 4. Diagram show power flow in/out components within the EV system for both normal forward driving and regenerative braking operations [6].

- Motor power in braking mode:

$$P_{mot\_in} = P_{mot\_out} \cdot \eta_m, \\ P_{mot\_out} = P_{te} \cdot \eta_g \tag{15}$$

- Battery power:

$$P_{bat} = P_{mot\_in} + P_{ac} \tag{16}$$

2.2. EV conversion simulation test

Models described in the previous section, especially traction model, are employed to simulate the electric vehicle conversion (EVC) performance by obtaining the velocity plot. The vehicle model specifications are approximately used as the input for the simulation. Other inputs are motor specification and road condition where Refs. [2, 4, 8] explained this specification in details.

2.2.1. Programming for simulation

The traction model is reduced to nonlinear first-order differential forms in [6, 8] when all inputs are substituted. Then, differential equation of velocity is numerically solved using the MATLAB script (.m) file for each time step and updates the values in the program arrays. The out velocity can be plotted against time. The EVC performance here is specified as the time for vehicle to accelerate from 0 to 100 km/h.

2.2.2. EV driving simulation

The other important piece of information for the EV design is range per charge, which tells us how far the vehicle can travel before it needs to be recharged again. In order to obtain such information, the motor model, battery model, and power flow model introduced in the previous section are applied here along with additional inputs. Driving cycle needs to be reasonably selected to simulate the driving dynamics. For present simulation, simplified federal urban driving cycle (SFUD) in [6] is chosen since the vehicle is expected to be driven in the urban area most of the time. The main program [6, 8] is employed to call inputs, including vehicle specification and driving cycle, and then execute the power flow model and battery model for each driving cycle and update parameters, such as range and *DoD* simultaneously. The range per charge then can be plotted when the program is done executing the program. Scenarios for EV range design can be explored using this simulation procedure [6, 8].

2.2.3. EV conversion design parameter simulation

EV design parameters shown in the list below can be easily obtained using the simulation done earlier. To obtain such information, we need to simply write the MATLAB commands in EV main program to update our interested parameters and then write the plot command.

An example of such torque speed map plots is shown in **Figure 5**, and the vehicle is still operated within the motor power range and maximum torque of 250 Nm. The constant torque

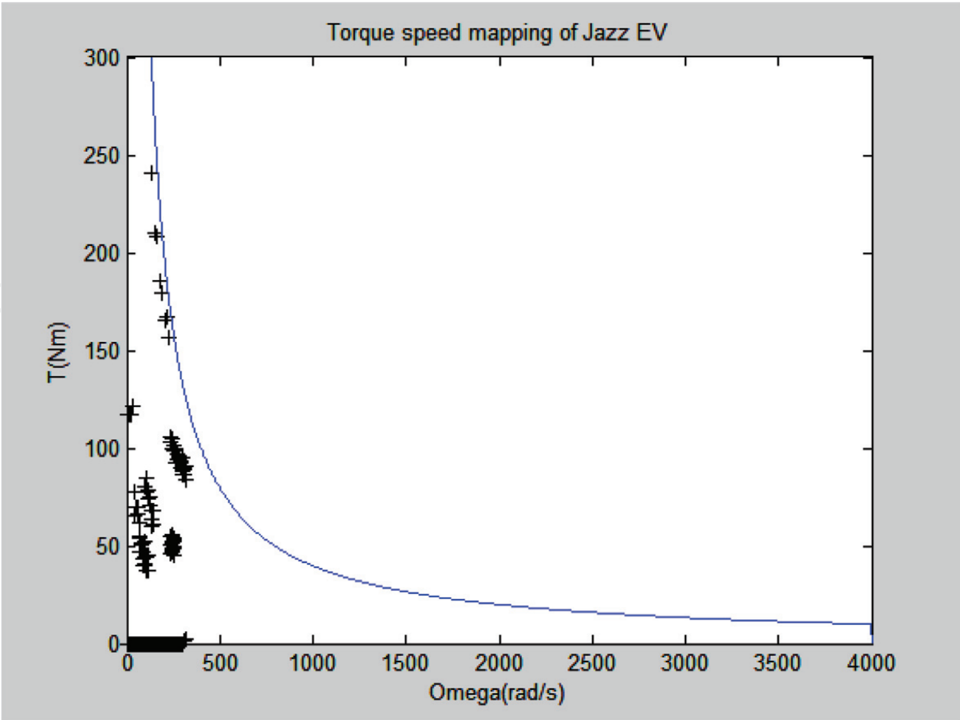

Figure 5. Torque speed map of EVC with SFUD driving cycle and no regenerative braking mode.

region is quite small compared to the field weakening region. The plot also reveals that low motor speed is mostly required when driving in the urban area.

3. EV conversion ECU design and in-the-loop testing

In current EV conversion development as shown in **Figure 6**, drive-by-wire (DBW) functions were developed by means of model-based design approach to synchronize the EV driving characteristics and to improve its drivability. The process starts by setting up parameters and variables for conceptual ECU system requirements. Then, the ECU I/O and signals for communications are formulated. Here, both software functions and embedded hardware design for DBW ECU should be completely determined. The next process is to virtually test DBW ECU against requirements' virtually simulated environment. In this process, the main functionalities along with faulty software or malfunction situation for the ECU can be tested. Bug in the software or communication can be tested and tuned safely with this in-the-loop testing methodology throughout the development process.

3.1. Drive-by-wire ECU design

The main function of conceptual drive-by-wire ECU developed by [2] is to determine power demand from the driver, through vehicle supervisory control ECU, based on the pedal ratio in percentage as shown in **Figure 7** and software algorithm. Next, percent pedal kickdown signal is sent to DWW ECU for torque command and regenerative percentage setting based on power

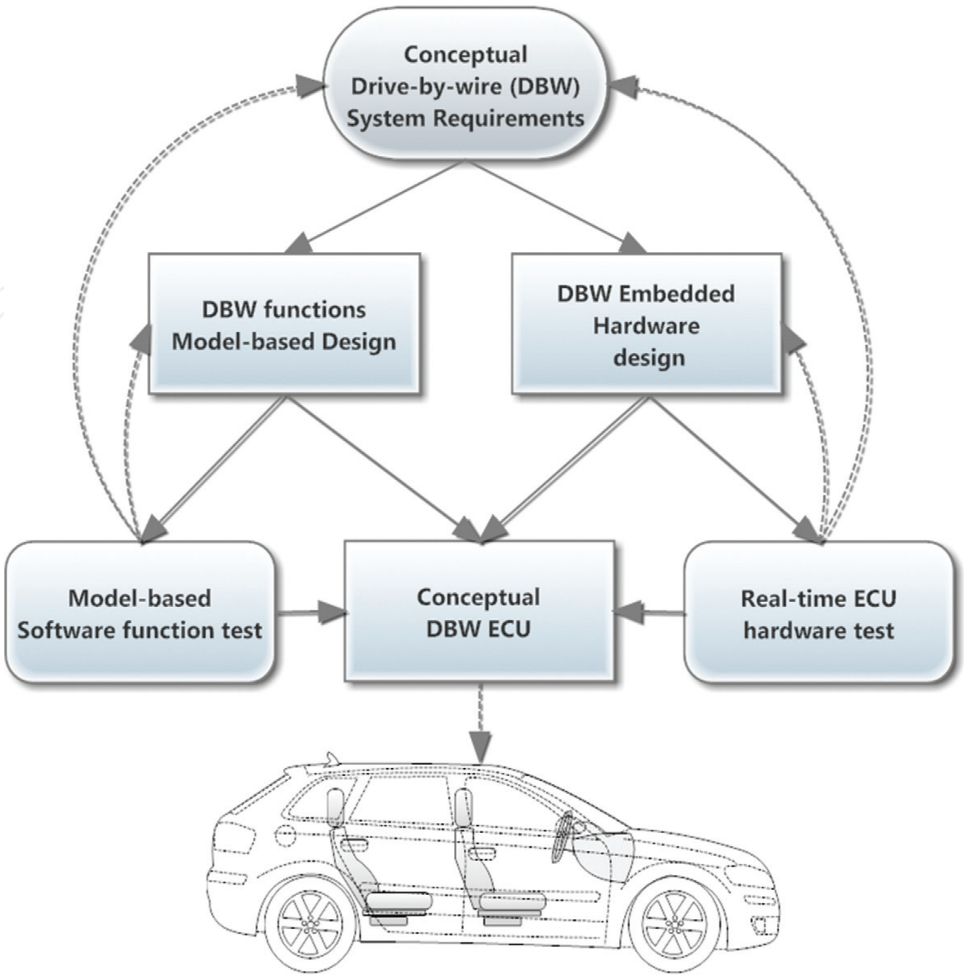

Figure 6. Novel methodology for rapid and safe EV software and hardware development [9].

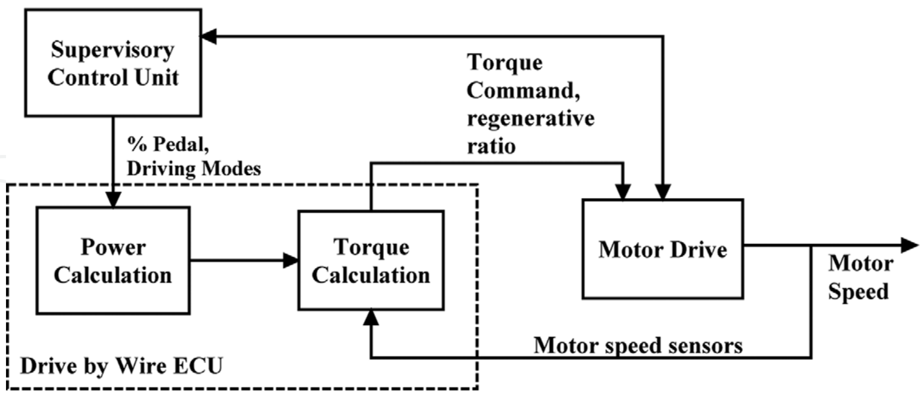

Figure 7. Drive-by-wire ECU functions and signal connection to the supervisory ECU and the motor drive unit [2].

to torque calculation and motor speed signals [9] in rule-based control algorithm. The input and output (I/O) parameters employed for DBW software and ECU are presented in **Figure 8**.

Since new characteristics from EV propulsion are applied to the old chassis. New torque map shall be calculated to compensate EV conversion performance. The design process can be

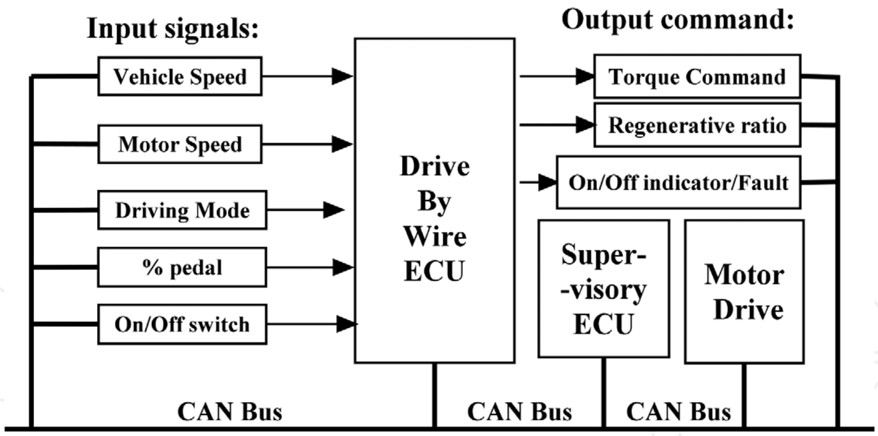

Figure 8. Input and output signal flow of the drive-by-wire ECU with CAN bus interface [2].

reviewed in Ref. [2]. The basic principle is to determine torque setting for various EV driving situations in four quadrants of torque speed map. This methodology can enable the design of more advanced features, such as driving modes, and other advanced driver assistance system (ADAS).

3.2. Model-in-the-loop test

Initial concept of EV software functions can be tested by simulating EV system components and virtual environment of model-based software function test as seen in **Figure 9** [4, 10, 11]. Model-based system design of EV component and drive-by-wire algorithm development can be consulted in details in Ref. [2]. In-the-loop models of driving test profile, supervisory control, DBW function, and motor are developed by using simulation software such as MATLAB/Simulink or others to emulate EV parameters and communication between the ECU and the driving load from vehicle dynamic model. System design requirement can be verified in this MBSF testing stage, such as toque map, and driving mode tests, which can be

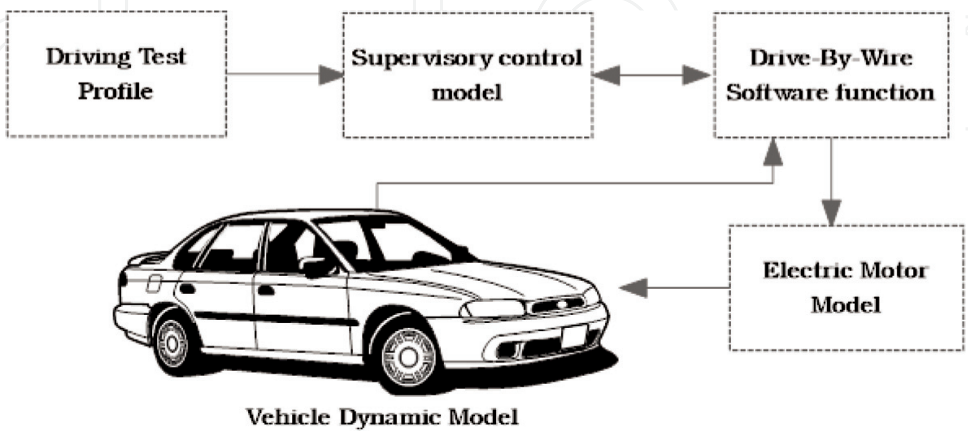

Figure 9. Model-based software function test setup where drive-by-wire function model is connected within the loop with other joint models [9].

done by setting up driving test profile and run the simulation for analysis. However, its capability is not as effective as real-time simulation, which is presented in Section 3.4.

3.3. Simulation analysis

Simulation results from model-in-the-loop test can be analyzed to verify whether system requirements are met. Driving profile in **Figure 14** can be set in several driving schedules as seen in **Figure 10**. After simulation is performed for DBW function, the parameters such as torque speed curve can be analyzed to check EV output such as performance in driving quadrants in **Figure 10**.

Without actual driving, DBW parameters resulted from simulation can be analyzed in different scenarios such as forward driving and reverse driving. Major error can be corrected at this stage along with fault-tolerant function test such as limp home mode in case the DBW is disconnected or malfunctioned.

3.4. Hardware-in-the-loop test

When ECU hardware is ready for testing, software can embed into the ECU to operate in real-time environment. The process is called hardware-in-the-loop (HIL) test when the drive-by-wire software algorithm is replaced by a physical ECU hardware while still connected to virtual environment as seen in **Figure 11**. Thus, HIL components consisted of an actual hardware, real-time interface, and virtual environments (models). It requires a capable communication protocol to handle real-time signal process where CAN protocol is chosen for this HIL System [12–17]. The overall specification of HIL system can be found in **Table 1**.

Arrangement of HIL configuration allows the engineers to conduct test for DBW ECU where it is difficult to perform with the actual vehicle. EV fault and malfunction scenarios can be simulated within the system to check ECU resiliency and fault-tolerant setting. Test repetition and automation can simply be done by scheduling HIL system. Therefore, it helps to reduce testing time and test cases required for real driving. ECU performance testing can be

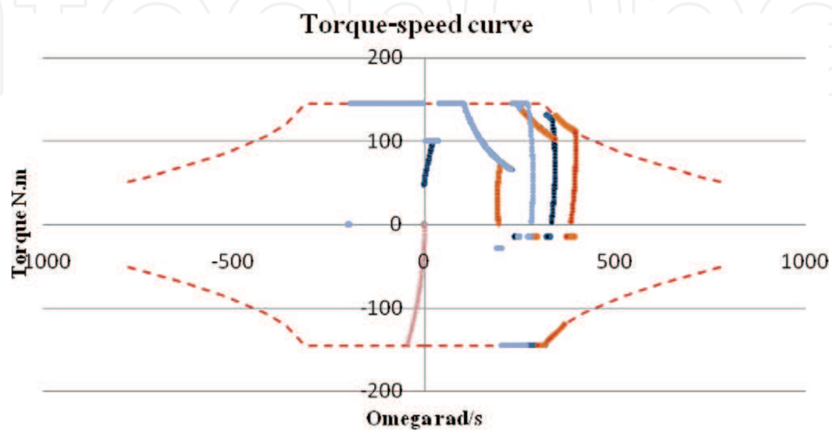

Figure 10. EV torque speed in four-quadrant driving results for analysis by means of model-in-the-loop simulation [8].

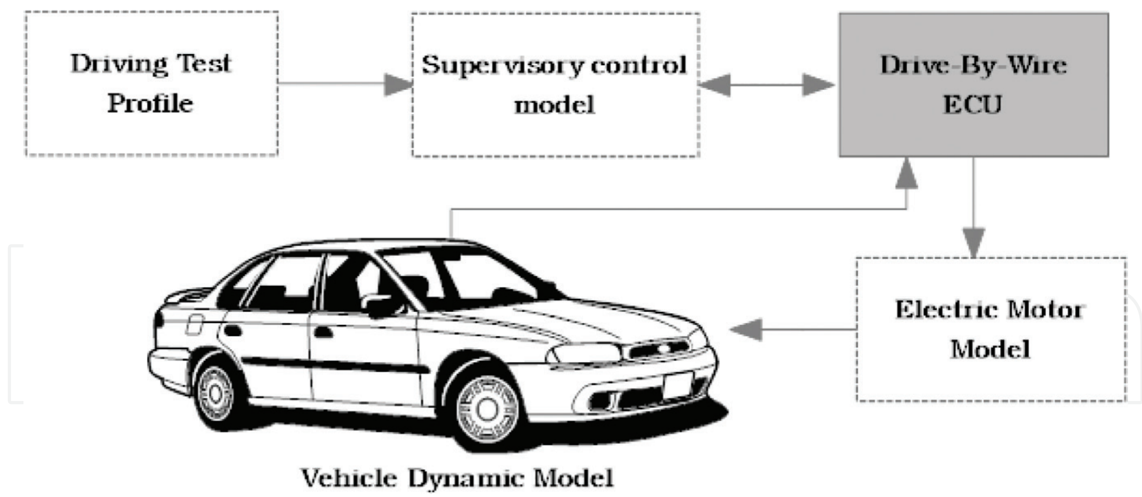

Figure 11. Hardware-in-the-loop (HIL) test configuration for drive-by-wire (DBW) ECU [9].

Components	**Specification**
Drive-by-wire ECU	Real-time rapid prototyping board
	CPU: ARM Cortex-M4 32bits 168 MHz
	RAM: 8 Mb
Vehicle dynamic and driving profile real-time applications	Real-time processor board
	CPU: ARM Cortex-M4 32bits 168 MHz
	RAM: 8 Mb
Real-time platform	MathWork Simulink real-time workshop
Interface	CAN bus 2.0 (high speed)
	Baud rate: 500 kBaud
Physical connection	CAN: DB9 connector
Power supply	12 V terminal
Protocol sampling time	10 ms

Table 1. Details of HIL test system specification [9].

conducted for EV high speed where it is difficult for real driving test. All model and ECU parameters can be adjusted simultaneously during the test, in real time, enabling more accurate parameter tuning. Therefore, system requirements can be verified in real time in this process.

3.5. Real-time ECU test analysis

To perform DBW ECU HIL test for this work, Simulink real-time workshop toolbox is chosen along with real-time application module for driving profile and vehicle dynamics, and CAN

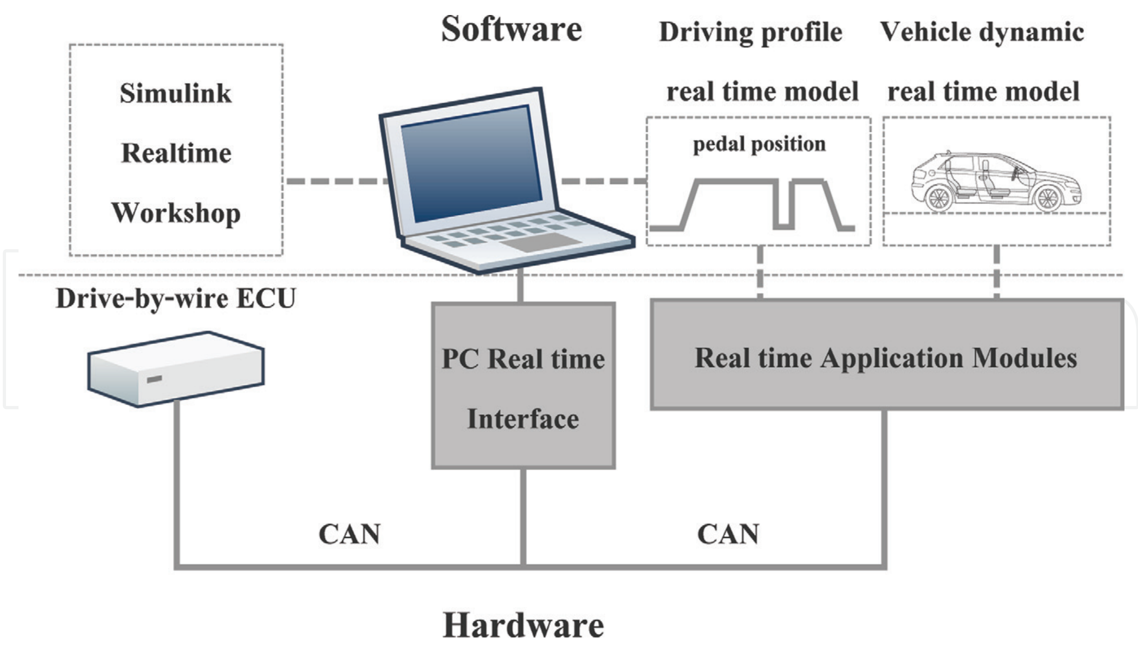

Figure 12. Hardware component integration setup for ECU function tests [9].

protocol is set for PC and ECU real-time interface as seen in **Figure 12**. Simple driving profile for this specific DBW test consists of different driving patterns to represent accelerator pressing by the driver in **Figure 13**. More complicated driving profile can be designated based on test scenarios and particular interest.

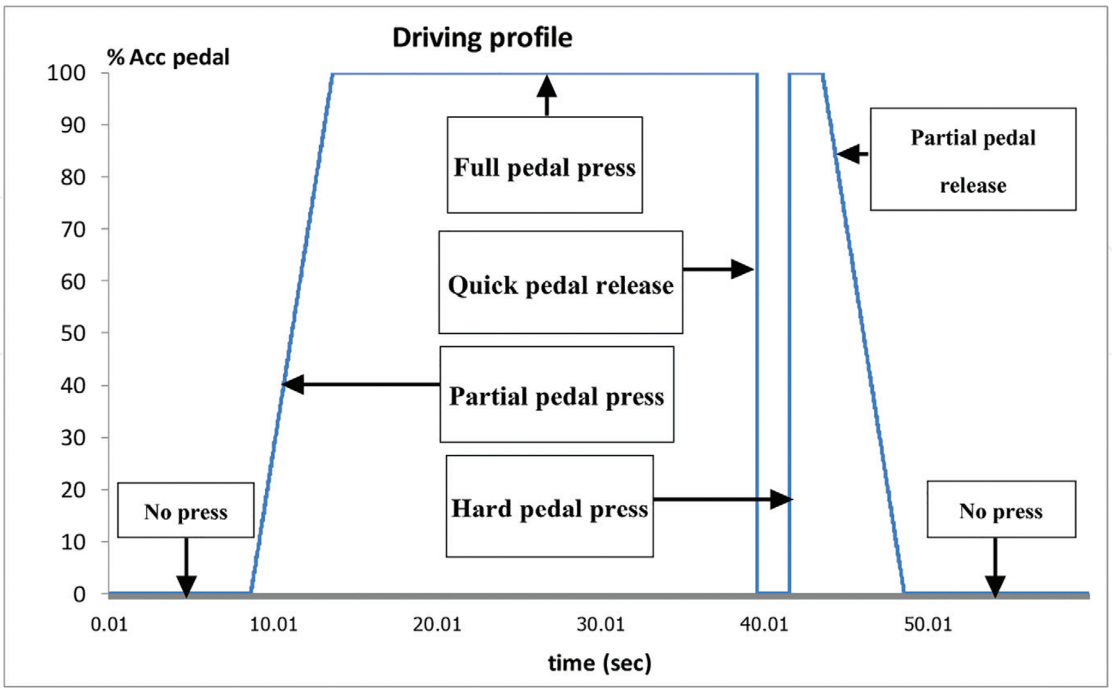

Figure 13. Driving profile test profile based on accelerator pedal position [9].

Based on the driving profile, the test results can be analyzed. In-depth test analysis can be consulted in Ref. [9]. In this case, performance parameters such as vehicle speed and acceleration are monitored for ECU validation.

4. EV conversion tuning and diagnostics

Model-based design can be employed to simulate scenarios where problem might occur during EV conversion process or to figure out the root cause of the problem [18]. The problem can then be realized beforehand by scenario test run to prevent the damage of the EV components and saving time to reinstall new part. The examples of model-based design for tuning and diagnostics can be demonstrated below.

4.1. Motor sizing mismatch scenario

This problem can occur during test run of EV conversion prototype where the motor drive converter becomes too hot during initial run of the EV. This problem forces the drive component to shut down to prevent circuit overheat. Therefore, the EV has to stop early to be checked. The torque map simulation can be executed to check whether there is mismatch between driving load and motor sizing as seen in **Figure 14** or there is some serge in current during EV driving simulation due to poor EV conversion system design as seen **Figure 15**.

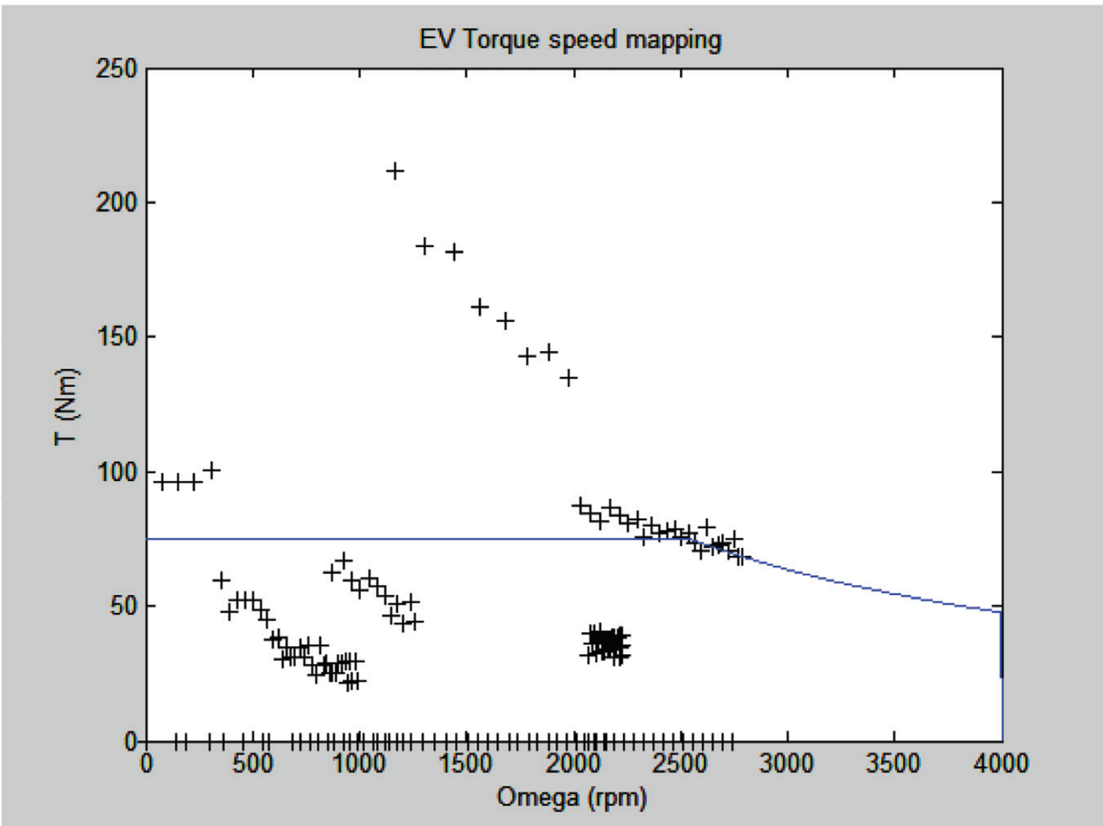

Figure 14. EV torque map results show mismatch in driving load vs. motor power.

http://dx.doi.org/10.5772/intechopen.77265

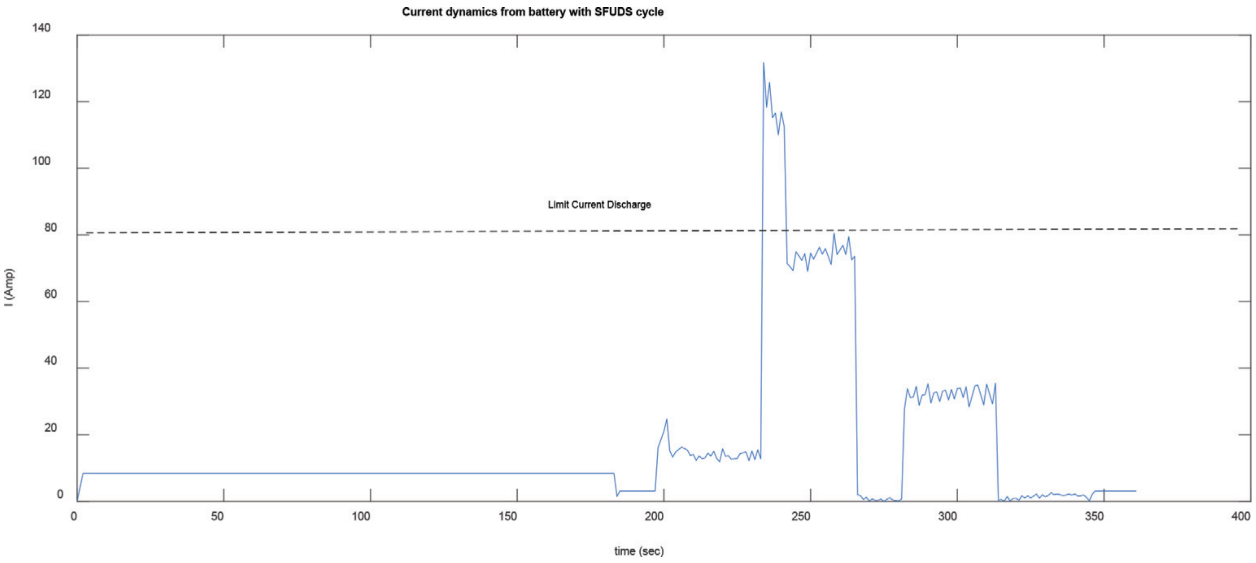

Figure 15. EV simulation show excess current drawn from batteries during some period of EV driving.

4.2. EV conversion driving range and energy storage technology

For the simplicity of modeling, PEM fuel cell with the lower heating value (LHV) plant will be reasonably modeled as a system with approximated value of efficiency [6]. Mass of liquid hydrogen supplied from cryogenic tank is calibrated to provide the same amount of energy supplied when compared to battery source. Because the fuel cell system was employed to entirely replace the batteries, regenerative braking option was not available. Comprehensive review for other types of fuel cell system and hydrogen storage for electric vehicle can be found in Larminie and Lowry's literature [4, 6]. To model the power flow model for fuel cell electric pickup truck, the battery unit is simply replaced by fuel cell system for one way flow of power since there is no regenerative braking option available. The analysis can be seen in Ref. [8].

4.3. Fault-tolerant simulation

Another important aspect for EV reliability is the fault-tolerant function. This is the safety feature when the main components of the EV, such as ECU or sensors, are malfunctioned. It ensures that EV can still provide safe operation, although in an inefficient manner. One scenario is limp home mode where EV can still be driven under limited speed and performance. By employing model-based design, limp home mode scenarios can be simulated and analyzed. As a result, accelerator redundant system can be designed and tested to compensate when fault is detected within the EV system.

4.4. Noise and vibration simulation

Improper installation or configuration of EV components could seriously induce noise and vibration, resulting in damage or shorter lifespan of EV components or undesirable noise level. EV mechanical system model can be simulated at early stage to validate the level of noise, and vibration is acceptable before the actual bench testing. Vibration characteristics such as resonance and force inducted due to rotating unbalance of the motor can be investigated and analyzed by means of model-based system design approach as well. Thus, proper noise and vibration handling method for EV system can be applied to protect and maintain EV components.

5. Conclusion

Model-based system analysis can be helpful to improve EV conversion system design, software and ECU development, testing, tuning, and diagnostic processes along with the actual EV prototype fabrication. It can also be used as a decision-making tool for future EV customization process. This methodology would benefit for EV conversion not only in terms of saving cost but also to shorten development period. As a result, reliable and low-cost EV conversion can be established.

Acknowledgements

This work cannot be accomplished without the support from several organizations. I would like to thank my engineering faculty at Naresuan University for establishing DRIVE center and providing budgets for equipment. Furthermore, I am grateful and appreciative for continuously supporting this R&D program from the corporations of EGAT (Electricity Generating Authority of Thailand) and NSTDA (National Science and Technology Development Agency).

Author details

Ananchai Ukaew

Address all correspondence to: ananchaiu@nu.ac.th

Development and Research of Innovative Vehicle Engineering (DRIVE) Center, Faculty of Engineering, Naresuan University, Thailand

References

[1] Fowler KR. What Every Engineer Should Know about Developing Real-Time Embedded Product. Boca Raton FL: CRC Press; 2008. Ch. 1-3

[2] Ukaew A. Model based system design of conceptual drive-by-wire ECU functions for electric vehicle conversion. SAE International Journal of Passenger Cars—Electronic and Electrical Systems. 2013;**6**(2):411-418. DOI: 10.4271/2013-01-0426

[3] Mueller S, Langjahr P. Efficient testing and cost awareness: Low-cost versus HIL-system. SAE Technical Paper 2012-01-0933. 2012. DOI: 10.4271/2012-01-0933

[4] Vuli P, Badalament M, Jaikamal V. Maximizing test asset re-use across MiL, SiL, and HiL development platforms. SAE Technical Paper 2010–01-0660. 2010. DOI: 10.4271/2010-01-0660

[5] Himmler A, Lamberg K, Beine M. Hardware-in-the-loop testing in the context of ISO 26262. SAE Technical Paper 2012-01-0035. 2012. DOI: 10.4271/2012-01-0035

[6] Larminie J, Lowry J. Electric Vehicle Technology Explained. UK: John Wiley & Sons; 2013. ISBN: 0-470-85163-5

[7] Patil K, Muli M, Zhu Z. Model-based development and production implementation of motor drive controller for hybrid electric vehicle. SAE Technical Paper 2013-01-0158. 2013. DOI: 10.4271/2013-01-0158

[8] Ukaew A. Model based system design of urban fleet electric vehicle conversion. SAE Technical Paper 2014-01-2005. 2014. DOI: 10.4271/2014-01-2005.946

[9] Ukaew A, Chauypen C. Implementation of conceptual real-time embedded functional design via drive-by-wire ECU development. International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering. 2015;**9**(6):940-946

[10] Choi Y, Kim Y, Moon H, Son Y. Model based design and real-time simulation of the electric bike using RT-LAB and Simulink. SAE Technical Paper 2013-01-0110. 2012. DOI: 10.4271/2013-01-0110

[11] King J, Nelson D. Model-based design of a plug-in hybrid electric vehicle control strategy. SAE Technical Paper 2013–01-1753. 2013. DOI: 10.4271/2013-01-1753

[12] Vandi G, Moro D, Ponti F, Parenti R. Vehicle dynamics modeling for real-time simulation. SAE Technical Paper 2013–24-0144. 2013. DOI: 10.4271/2013-24-0144

[13] Yeaton M. Managing the challenges of automotive embedded software development using model-based methods for design and specification. SAE Technical Paper 2004–01-0720. 2004. DOI: 10.4271/2004-01-0720

[14] William R. Real-Time Systems Development. Burlington, MA: Butterworth-Heinemann; 2006. Ch. 1–3

[15] Vincentelli AS, Zeng H, Natale MD, Marwedel P. Embedded System Development from Functional Models to Implementations. NY: Springer-Science; 2013. pp. 8-16

[16] Laplante PA. Real-Time System Design and Analysis. 3rd ed. NJ: IEEE Press, Wiley-Interscience; 2004. Ch 1, 5

[17] Lee I, Leung JYT, Son SH. Handbook of Real-time and Embedded Systems. Boca Raton, FL: Chapman & Hall/CRC; 2008. pp. 2.1-2.16

[18] Michaels L, Pagerit S, Rousseau A, Sharer P. Model-based systems engineering and control system development via virtual hardware-in-the-loop simulation. SAE Technical Paper 2010-01-2325. 2010. DOI: 10.4271/2010-01-2325