WEX-HIL: Design of a Wireless Extensible Hardware-in-the-loop Real-time Simulator for Electric Vehicle Applications

Yong-Kyu Jung, Fong Mak, Fnu Qingele, and Idrees Alzahid*
Electrical & Computer Engineering Gannon University, 109 University Square, Erie, PA 16541, USA
{jung002, mak001, qingele001}@gannon.edu, and 11422@uotechnology.edu.iq*

***Abstract*—A real-time simulator is an essential tool for swiftly designing and verifying embedded time-sensitive and/or real-time applications in both academia and industry. An FPGA-based wireless extensible hardware-in-the-loop (WEX-HIL) real-time simulator was developed for designing subsystems of an electrical vehicle (EV). The WEX-HIL platform is capable of providing a design and test environment of virtual and real embedded system prototypes for academic research and education. The WEX-HIL platform permits a relatively quick turnaround time for software model development. The FPGA-based virtual hardware prototyping is also effectively done with the WEX-HIL platform by interfacing software models executing in the baseline real-time simulator with the virtual prototypes. Furthermore, the proposed real-time simulation platform provides a user friendly GUI-based model development environment with the commercial software IDEs from high-level to low-level and with the hardware IDEs from simulation to synthesis. The WEX-HIL platform particularly offers an ample means of integrating software and hardware models and virtual and real prototypes via high-speed wire and flexible wireless communications. The WEX-HIL real-time simulation achieved 2.89% of distance error accuracy with the sensing system, 2.5x faster acceleration and 0.3% HIL simulation error, the 50 µs responding timing constraints, and intuitive intra and inter model and prototype integration at different abstract levels in SIL and HIL real-time simulations. With the WEX-HIL platform, a comprehensive hands-on multi-inter-disciplinary course in Electrical and Computer Engineering was created and offered. The course comprises a series of the subjects for successful designing and utilizing the WEX-HIL platform and related applications in Computer, Communication, Electrical and Power-Electronics Engineering. Hands-on laboratories equipped with hardware/software design tools were also developed for further experiments of current and emerging applications in the embedded and real-time systems. The proposed course is expected to alleviate the technology gap developed and utilized between industry and academia.**

Keywords—real-time simulation; hardware-in-the-loop simulation; virtual/real prototyping; electrical vehicle

I. Introduction

Modern transportation systems, including automobiles and locomotives, employ more electrical, electronics, and computing systems embedded for the rapidly increasing demands of safe autonomous operations, control accuracy and complexity, and more user friendly features than traditional transportations. On top of these challenges, the automotive industry and research community in academia also have been faced with swiftly evolving such transportation systems under tight time-to-market pressure. To effectively meet the aforementioned demands, advantages of the electronic design automation (EDA) are feasible for both hardware and software developments of the transportation systems. Although most of the simulation-based approaches are popular for virtual prototyping, traditional simulation-based design environments need to be extended for real-prototyping in automotive applications. Thereby, a real-time simulation with real-prototyping becomes one of the viable solutions for embedded hardware system developers in automotive industry [1].

Real-time simulation inherently permits hardware involvement during the simulation, the so-called hardware-in-the-loop (HIL) simulation. Various HIL simulations [2, 3, 4] were exercised and proved capability of the real-time simulation with hardware prototypes. In addition, the real-time simulations must perform the accurate simulation with existing and developing hardware and software modules. Furthermore, various complex subsystems are seamlessly integrated into a real-time simulation platform while maintaining the intuitive and user-friendly operations, including precise design refinement and effective evaluation.

Since autonomous transportations have been actively researched for the past decade, numerous sensors have been developed for parts of automotive and transportation applications [5]. Cameras are the most popular device for sensing visual information surrounding a transportation system. The GPS systems installed in the transportation system are aimed at finding the location and position of the system. In addition, an automobile is one of the viable applications for being beneficial of Internet-of-things (IoTs), which gather various forms of information from numerous types of sensors via wireless communication and process the information on embedded or application specific processing engines [6]. Thus, more perceptive HIL simulations are generally recommended.

In order to support highly sensitive and sufficiently accurate HIL simulations, a real-time simulation platform provides intuitive input/output configuration, swift extension, precise execution, and systematic verification. In general, a mixture of hardware and software components are interacting with each other in the real-time simulator; both hardware and software developers can promptly evaluate their subsystems installed on the transportation system. The components of the EV can be modeled with a control strategy, various I/O interfaces, different signaling, and signal conditioning. For instance, an electric vehicle (EV) comprises of an engine (i.e., motor) controller, speed controller, battery charger, inverter, and transmission. Some real-time simulators also were developed for electric control unit in EVs [7], fuel cell in hybrid EVs [8], and electric and hydraulic systems in avionics [9]. In lieu of the real-time simulators for the specific applications performed well, the baseline simulation platform must maintain the key features addressed before.

Unlike complicated real-time simulators that have been utilized in industry, relatively simple and limited scope of real-time applications have been dealt with in the real-time simulation environments in academia. The industrial real-time simulations triggered to significantly grow in the usage of real-time simulators in the classroom since the last decade [10]. Academic versions of real-time simulators [11, 12] typically include primary simulation features to handle laboratory exercises inter-operable with equipment, including FPGA/microcontroller boards. More specifically, user-friendly interfaces, the intuitive expansion, and multiple user support are more important than large scale and realistic prototyping. However, this creates an increasing gap between the industrial and academic practices [13].

We have developed a real-time simulation platform in order to reduce the gap in terms of the seamless integration and intuitive interface to the various hardware and models used in the academic research. The proposed real-time simulation platform is an FPGA-based Wireless Extensible hardware-in-the-loop (WEX-HIL) capable real-time simulator. The WEX-HIL has been successfully utilized in an academic environment, especially for applications discovered in the inter-disciplinary research and education in Embedded Systems, Computing, and Communications [14-18]. Particularly, the developed WEX-HIL real-time simulator successfully performs automotive electronics research including an automated collision prevention system installed on an autonomous EV prototype.

Section 2 describes the architecture and operation of the WEX-HIL real-time simulator. Section 3 expresses evaluation of the WEX-HIL real-time simulator integrated to an EV prototype and models via a flexible wireless interface module for the rapid HIL simulations. The evaluation results and analysis of the WEX-HIL real-time simulator with the EV prototype are also described in Section 3. Section 4 elicits the conclusions and future work.

II. Design of the WEX-HIL Real-time Simulation Platform

Fig. 1 illustrates a block diagram of the architecture of the proposed WEX-HIL real-time simulator. The autonomous EV prototype shown in Fig. 1 (a) consists of the sensor modules and a sensor signal processing and wireless communication module, which is shown in Fig. 1 (b). The presented autonomous EV prototype was implemented with ultrasonic/image sensors [19, 20] for detecting objects and measuring distance of the objects

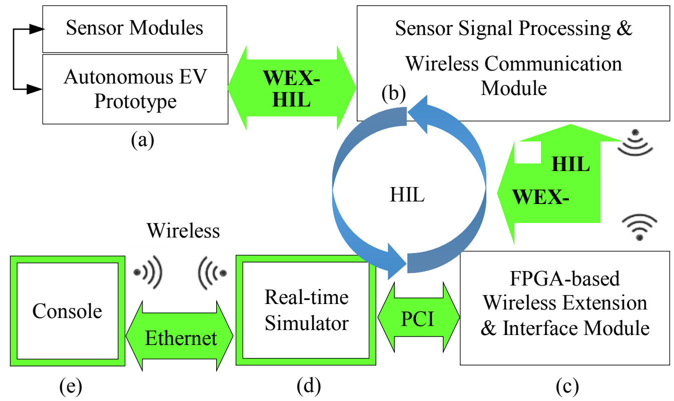

Fig 1. A block diagram of the wireless sensors installed on an autonomous EV prototype for a cost-effective academic FPGA-based Wireless Extensible hardware-in-the-loop (WEX-HIL) real-time simulation; PCI: peripheral component interconnect

and an Arduino microcontroller board [21] for processing signals of the sensors and transmitting information to and receiving them from the flexible wireless interface module. A field-programmable gate array (FPGA)-based interface module illustrated in Fig. 1 (c), is designed with a wireless device for swiftly extending additional modules to the EV prototype and a peripheral component interconnect (PCI) [22] for interfacing to the WEX-HIL real-time simulator.

The WEX-HIL real-time simulator comprises of the QNX Neutrino real-time operating system (RTOS) [23] running on multi-cores in the real-time simulator seen in Fig. 1 (d), shared memory, and an analog/digital input and/or output via the PCI card. Microkernel architecture of the QNX RTOS permits running a number of modularized tasks without changing the RTOS while optionally providing the minimal services of the high-level OS functionality. The QNX also offers adaptive partitioning and a driver for supporting extensive PCI. With these features, the initial console interface via a 100 Mbps Ethernet (i.e., IEEE 802.3u) was extended to an 11 Mbps WiFi (i.e., IEEE 802.11b). Thereby, the real-time simulator is interconnected to a single or multiple consoles shown in Fig. 1 (e) via the high-speed Ethernet and/or the wireless WiFi for dynamically interacting with users while running real-time simulations.

A. *Operation of the WEX-HIL Real-time Simulator*

Fig. 1 also shows an operational flow of the WEX-HIL real-time simulator. An autonomously operable EV needs to be partitioned to virtual software and hardware models and real prototype with the virtual software models directly running on the real-time simulator while the virtual hardware models executing on the FPGA and inter-operating with the real-time simulator via the PCI interface. The EV prototype is concurrently operating with the virtual models running on the real-time simulator and communicating with the virtual models via the wireless extensible interface module for HIL simulation.

1) *Virtual Software & Hardware Model Management*

As seen in Fig. 2 (a) and (b), virtual software and/or hardware models are created through the console by invoking a

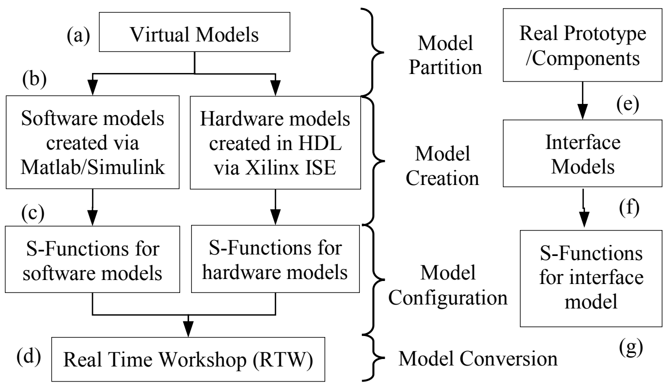

Fig. 2. Virtual software and hardware modeling and interfacing real prototype for the WEX-HIL real-time simulation

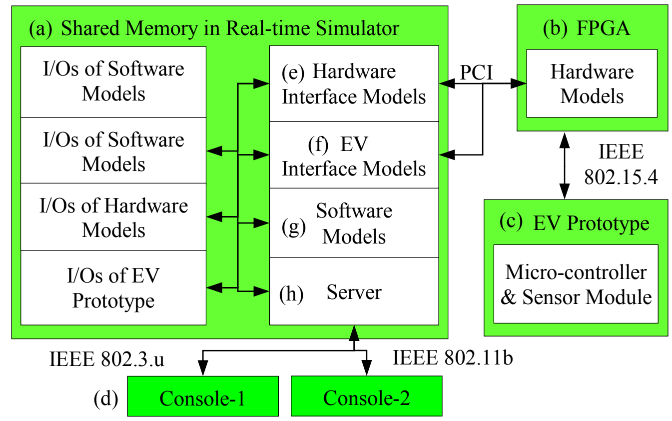

Fig. 3. Inter-operations of the WEX-HIL Real-time Simulations with the Virtual Software Models, Virtual Hardware Models, and EV Prototype

built-in modeling procedure integrated with modeling tool suites, such as Matlab/Simulink for software models and Xilinx ISE for hardware description language (HDL) models. Particularly, the software models created through the Matlab/Simulink are converted to C code via the real time workshop (RTW) as illustrated in Fig. 2 (d). Fig. 2 (c) shows that the virtual models are properly configured, after entire models are completed. The configured virtual models are converted to be compiled and executed before exporting the models to the real-time simulator for the HIL simulation as seen in Fig. 2 (d). In particular, interfacing between virtual models requires configuring outputs and inputs of the models for exchanging the operation results with each other. The virtual models with the associated configurations are compiled for partitioning and scheduling the model simulations by the real-time simulator. The real-time simulator, then, allocates the executable models to the available cores according to the simulation schedule established during the model compilation. Therefore, the real-time simulator is ready for the HIL simulations.

In order to seamlessly integrate virtual software models with each other, and to hardware models and the real EV prototype, custom Simulink system function, S-Function, was developed for intuitive model integration. Thus, RTW generates header information during the model conversion process according to the S-Function configured. RTW, then, utilizes the target language compiler file to generate the inline macro call required to access the shared memory region. The S-Function also provides an effective means for inter-operation of the models and the prototype involving the same HIL real-time simulation simultaneously. Thereby, the virtual models and the real prototype can exchange their inputs and outputs via the shared memory region partitioned under the real-time simulation environment. Similarly, the EV prototype, including sub-modules installed on the EV, must be integrated to the real-time simulator by creating and configuring the associated interface models for the HIL real-time simulation.

2) Operations and Interfaces of Virtual Model and Real Prototype

Fig. 3 illustrates three concurrently operating modules in the virtual software/hardware models and the real prototype for the WEX-HIL real-time simulation. The software models with configuration are executable with each other via the allocated regions in the shared memory as shown in Fig. 3 (a) and (g). However, the virtual hardware models running on the FPGA need the associated interface models configured for assigning the hardware models to the different regions in the shared memory as seen in Fig. 3 (e). Thus, I/Os of the hardware models are virtually interconnected to exchange the operation results between the models running on the FPGA and the models running on the real-time simulator via the PCI. Similarly, the EV prototype, including real hardware modules installed on the EV, can run in parallel with the hardware models running on the FPGA via the wireless communication channel established with the IEEE 802.15.4. To communicate the EV prototype with the software models, an EV interface models shown in Fig. 3 (f) must be configured with the other models. Since the EV prototype can interface with analog and digital signals, a microcontroller board employed by the EV prototype must be capable of converting the analog signals to digital signals and transmitting the digital signals to the FPGA-based interface module.

During the WEX-HIL real-time simulation, the real-time simulator operates as a server seen in Fig. 3 (h). The real-time simulator launches threads to service the software models, hardware interface models, and the prototype interface models according to the interface requests received. The real-time simulator initiates the WEX-HIL real-time simulation of the software models and the interface models according to the configuration. Since the software models and the interface models are distributed to the assigned regions in the shared memory according to the partitioning scenarios, a primarily important function is to share the same thread in different scheduled time. In addition, the concurrently running threads on different cores are monitored for dynamically distributing and scheduling tasks and managing the operation results. Furthermore, the virtual I/Os configured for the threads operating in parallel are established for the WEX-HIL simulation. Consequently, the simulation results collected in temporary files are transmitted to the multiple consoles via a (IEEE 802.11b) connection as seen in Fig. 3 (d). The client-server socket-based Ethernet (IEEE 802.3.u) or WiFi communication layer presented in Fig. 3 (d) is to resolve the FTP overhead, such as maintenance of the clients' IP addresses, as

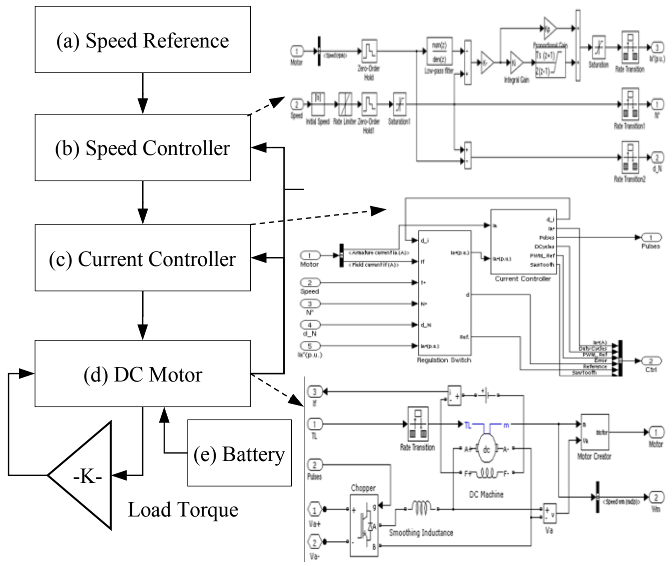

Fig. 4. Generation of software models of an EV with Simulink

well as to provide enhanced flexibility for transmitting custom XML messages to the interface models without further modification of the communication layer.

III. Evaluation Methodology for the WEX-HIL Simulation Platform

To evaluate the WEX-HIL simulation platform as an infrastructure of the real-time simulation, we used a combination of software models, hardware models, and a real prototype. Accordingly, three different real-time simulations, such as SIL, HIL, and WEX-HIL simulations, were tested. For the SIL simulation, a baseline EV model was implemented and simulated with the Matlab/Simulink. The baseline EV model was modified for the HIL simulations. Instead of using a real EV (i.e., golf cart), a miniature driving robot was chosen as an EV prototype.

Three primary functional aspects of the WEX-HIL real-time simulation platform identified are (1) intuitive integration of virtual models and a real prototype for providing user-friendly, rapid preparation of the real-time simulations, (2) inter-operability between software and hardware components during the HIL simulation, and (3) the HIL simulation accuracy comparing with the SIL simulation results in terms of the real-time constraint.

A. Developments of Software/Hardware Models for SIL and HIL simulation without Real Prototype

Fig. 4 illustrates a software EV model consisting of four key components, including a speed controller, a current controller, and a DC motor. The software models of the components shown in Fig. 4 (b) to (e) are designed with Simulink and then converted to the C models for the SIL simulation.

As seen in Fig. 5, inputs and outputs of the software models are configured with the S-functions. The C models and the S-functions are transmitted to the real-time simulator for compilation and execution. Because the required DC Motor

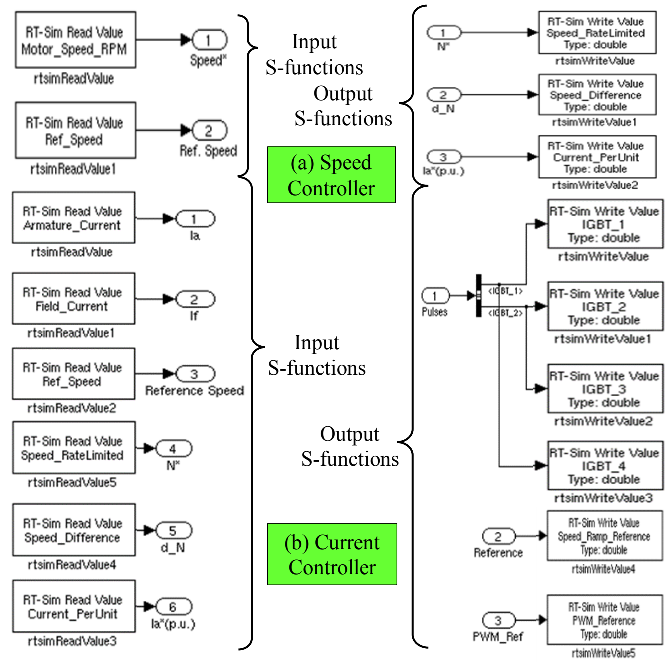

Fig. 5. Integration of software models using S-functions for SIL real-time simulation

speed was not satisfied by the frequency of the PWM signal generated by the current controller model implemented in Simulink, the PWM generator was replaced with the one implemented as the hardware model. Fig. 6 shows modified software models for the HIL simulation with the PWM generator implemented in VHDL and running on the FPGA. The output S-functions of the current controller for the SIL simulation shown in Fig. 5 (b) was changed to the output S-functions of the current controller for the HIL simulation shown in Fig. 6 (b). The DC motor models for the SIL and the HIL simulations are also modified.

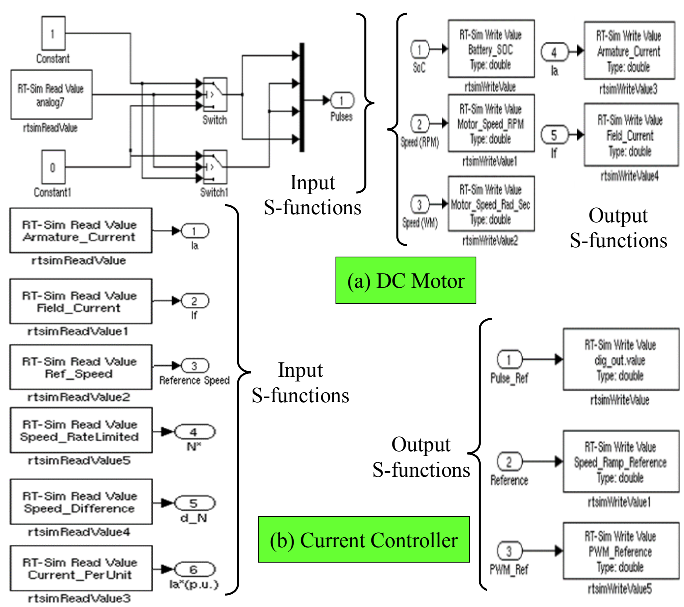

Fig. 6. Integration of software models using S-functions for HIL real-time simulation without the real prototype

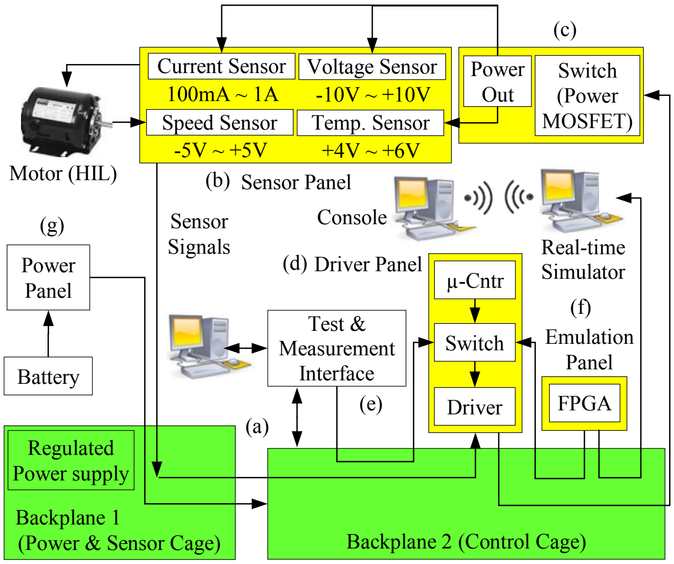

Fig. 7. An EV prototype designed in a modular manner for power electronics, signal acquisition interface, and driving control circuitry with the HIL real-time simulations

B. *Developments of Real Prototype for Integration to the WEX-HIL Real-time Simulation Platform*

After the SIL simulation with the software models, a few important components, especially for the time sensitive components, were identified and transformed to parts of hardware models for the HIL simulation of the target EV. The remaining software models can be implemented into the target EV prototype although some of the software models still can be used for evaluating the parts installed in the prototype. The baseline EV prototype not only provides useful references for evaluating accuracy of the real-time simulation, but also permits exercising various algorithms, including the routing path selection (RPS) and other similar algorithms [24]. Thereby, the SIL and HIL simulations without a real prototype limit the scope of the real-time simulator during the EV development. We are more interested in enhancing performance of the existing algorithms and researching potential solutions of the existing approaches. Therefore, we have been developing two EV prototypes. One EV prototype is for developing driving and controlling circuitries and signal acquisition interface. Another EV prototype is for developing applications running on the autonomous EVs.

Fig. 7 illustrates an overview of the mini-EV, which was modified from a golf cart. The motor in the mini-EV is the main driving source. In order to drive the motor, the power-electronics switches shown in Fig. 7 (b) and the sensor board shown in Fig. 7 (c) were designed based on constraints imposed by the motor. The driver circuit seen in Fig. 7 (d) is for driving the power-electronics switches board. The test and measurement board and the FPGA board shown in Fig. 7 (e) and (f), respectively, allow controlling the data acquisition and drive circuit. The power panel board in Fig. 7 (g) is to use the battery array and to provide required DC power to the boards. As seen in Fig. 7 (a), the boards are interconnected via the back plane boards installed on the mini-EV.

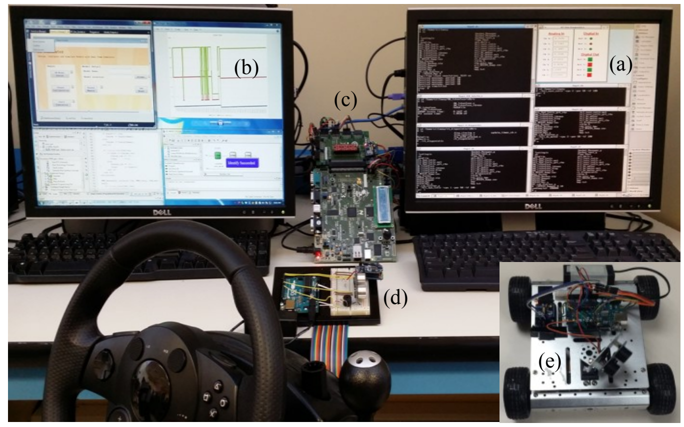

Fig. 8. An EV prototype designed for application and driving software research with the WEX-HIL real-time simulations

The autonomously driving EV prototype used for the WEX-HIL real-time simulation is shown in Fig. 8 (e). The EV prototype is useful to identify the simulation error between SIL and HIL simulations as well as to swiftly test various application hardware and software although the HIL simulation is still limited the further evaluation of the EV prototype, including embedded software testing. The EV prototype erupted wireless capability enhances remote accessibility and intuitive interface for the HIL simulations under the realistic test conditions. The EV prototype comprising in a wireless device (i.e., IEEE 802.15.4) interfaced with movable sensors (i.e., +/- 45 ^{o} spinning of ultrasonic and/or image sensors) mounted on a servo and another wireless device installed on an FPGA-based interface module shown in Fig. 8 (c) for detecting moving objects and for measuring the distance between the objects and the EV prototype. The EV prototype is capable of independently driving all wheels and operating with the RPS algorithm programmed. The EV prototype offers the high flexibility and compatibility with further extension. Three identified hierarchical real-time evaluation layers, such as SIL, HIL, and WEX-HIL, convince for users to intuitively apply their embedded systems to the WEX-HIL real-time simulation platform.

Fig. 8 also illustrates the WEX-HIL real-time simulation platform consisting of (a) a baseline real-time simulator for SIL/HIL simulations, (b) a console for user interface, and (c) an FPGA-based hardware and wire/wireless interface for HIL /WEX-HIL simulations. In particular, the WEX-HIL real-time simulation provides additional flexibility to expand simultaneous and distributed HIL operations without overheads of threads partitioning and scheduling in the baseline real-time simulator.

IV. New Rapid Embedded System Design Course with Real-time Simulation

A new embedded system design course with the proposed WEX-HIL real-time simulation platform was developed. This course aims to offer the concepts and hardware/software modeling techniques to delivery real-time data processing and signal acquisition on real-time applications. The subjects discussed in the course include embedded system modeling based on Matlab/Simulink, FPGA-based hardware

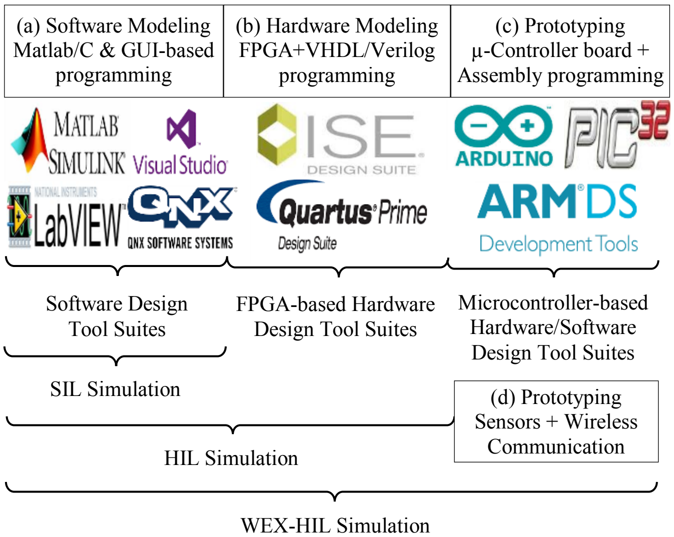

Fig. 9. Applicability of the WEX-HIL real-time simulation platform for embedded system and computing education

implementation in VHDL, and an EV prototype comprising of microcontrollers and sensors. Labview-based console frontend design is also expected to enhance GUI-based user interface to the real-time simulation platform. In particular, a miniature EV prototype serves a relatively realistic opportunity to explore various aspects of real-time application development, embedded software and hardware design, and understanding and establishing the wire and wireless communication channels under one real-time simulation platform.

As mentioned earlier, the main objective of this course conjunction with a series of laboratory experiments is to increase individual and team's rapid design opportunities and hands-on experiences. Consequently, near-future professionals are motivated with a systematic approach for system-level design and verification within relatively short period (i.e., one semester). The other objectives are to (1) understand real-time simulator design concept and flow; (2) develop skills for FPGA-based hardware modeling and programming language-based software modeling and integration of the models; and (3) develop proficiency with contemporary tools, including Matlab/Simulink, FPGA Integrated Synthesis Environment, Labview-based GUI and signal acquisition, and wire/wireless communication between heterogeneous operating systems (i.e., microkernel-based RTOS and monolithic kernel-based OS). The presented course with the WEX-HIL real-time simulation platform, therefore, is expected to expedite learning and implementation of relatively large scope of system-level embedded hardware and software design. As a result, students not only learn software and hardware design utilizing various popular tool suites, they also obtain knowledge on rapid design procedures, including system partitioning and design reuse techniques [25].

A. New Integration-oriented Design Methodology for Embedded System and Computing Education

To teach rapid and comprehensive embedded system design with the WEX-HIL real-time simulation platform within a single semester, the key subjects taught to the undergraduate students in Electrical and Computer Engineering major have been reviewed. Since embedded computing and system requires software and hardware components, (1) GUI-based or high-level language programming, (2) hardware description language (HDL)-based virtual prototyping, and (3) a mixed assembly and high-level programming in embedded software and microcontroller-based real prototyping are included as primary subjects. On top of the main subjects, interface and/or integration (1) between software and software, such as programming with different languages, (2) between virtual and real prototypes, such as FPGA and microcontroller, and (3) between embedded software and hardware with microcontroller boards, such as Arduino. In particular, subjects related to interface between heterogeneous operating system frameworks, such as small kernels used in RTOS and large kernels used in OS, need to be covered.

As seen in Fig. 9, three major software and hardware frameworks—(1) software modeling, (2) hardware modeling, and (3) system prototyping—are classified according to the embedded system design with the WEX-HIL real-time simulation platform. In Fig. 9 (a), the most commonly used programming languages, Matlab, C/C++, or JAVA, in engineering education are used. Particularly, GUI-based intuitive programming with Simulink and Labview are used for software modeling and console designing for the baseline real-time simulator. For the virtual prototyping with HDL, FPGA-based integrated design tools, such as Xilinx ISE, are used as seen in Fig. 9 (b). There are several microcontroller boards that are available and used in embedded systems. For instance, Arduino microcontroller boards and development environment (IDE), Microchip's PIC series and MPLAB, and ARM's DS-5 development studio can be used according to users' preference as shown in Fig. 9 (c). The additional real hardware prototyping, including sensor systems and wireless communication devices for IoT applications, can be exercised under the WEX-HIL simulation environment shown in Fig. 9 (d).

TABLE I. SUMMARY OF EMBEDDED SYSTEM DESIGN WITH THE WEX-HIL REAL-TIME SIMULATION PLATFORM

Attainments & Results of Integration-oriented Design Methodology	Types of Real-time Simulation		
	SIL	*HIL*	*WEX-HIL*
Software models in Matlab/C & Simulink/Labview	Yes	Yes	Yes
Modified software models and hardware models in VHLD/Vrilog with FPGA & design suites	No	Yes	Yes
Real Prototype with microcontroller boards & deisgn suites	No	No	Yes
Simulation Accuracy/Integration Complexity	Low	Mid	High

Software-/Hardware-in-the-loop (SIL/HIL) simulations

Table I summarizes required design attainments under the proposed integration-oriented embedded system design methodology. Design activities based on three real-time simulations, such as SIL, HIL, and WEX-HIL simulations, are listed in the first column of Table I. Design and simulation results are also included in the same column. The remaining

TABLE II. STUDENT OUTCOMES ASSESSMENT (2014~15)

Student Outcomes (SOs)	Correlated Course Outcomes	EAMU Average	
		2015	*2014*
Evaluate and utilize research methodologies appropriate to the discipline	Understand real-time simulator design concept and flow	3.95	4.11
Master the skills, methods, and knowledge appropriate to the discipline	Develop skills for FPGA-based real-time HW/SW modeling and integration	4.30	3.56
Access, analyze, and evaluate information	Develop proficiency with contemporary tools	5.00	4.22

EAMU: E (Excellent), A (Adequate), M (Minimal), U (Unsatisfactory)

three columns depict required inputs and outputs of the real-time simulations. For the SIL simulation, only software models are required, but simulation accuracy and model integration complexity are lower than the hardware components involved in other HIL simulations. The HIL simulation with virtual prototype generally needs simpler model integration between software and hardware models via an interface hardware module than those employed in the WEX-HIL simulation. The proposed WEX-HIL simulation has extended coverage of the real-time simulation with various built-in interfacing infrastructures to operating both software and hardware models as virtual prototypes and real prototypes of target embedded system while alleviating difficulty of the integration complexity.

B. Course Outline and Assesement

The proposed design methodology and the real-time simulation platform developed have been exercised for the past few semesters. The aforementioned modeling with commercial design tools were used for preparing real-time simulations with the WEX-HIL simulation platform. Then, model integration between software, hardware, and real-prototype are executed throughout the WEX-HIL simulation platform. Software model integration for SIL simulation is performed by I/O configuration of the models with the in-house S-functions and model partition and schedule in the baseline real-time simulator. In order to execute HIL simulation with virtual prototype, students exercise simplified hardware/software co-design and verification tasks. Finally, a comprehensive rapid EV prototyping assignment is carried out.

Based on the course assessment method, a few key assignments were used for monitoring and measuring the performance of students. Three course outcomes mentioned in the previous section are justified. The students' performance has been measured through the assessment strategies implemented on an online assessment tool.

1) Course Outline

The manuscript review and presentation aim to comprehend real-time applications and simulation for design and testing various embedded systems. Laboratory experiments are for exercising the knowledge of components and techniques in both software and hardware modeling and simulation flow. In addition, contemporary design suites shown in Fig. 9 are utilized to enhance the understanding of hardware and software integration under the real-time simulation environment. In order to provide a system-level design experience with both virtual and real prototypes and to achieve proficiency with contemporary design and analysis tools, a design project aimed for demonstrating the ability to deliver a real-time simulator design solution via more than one way to meet the design requirement under the time constraints.

The key assignment for understanding the real-time simulator design and concept provides an exploration opportunity to comprehend the capability of SIL/HIL operations. Thereby, the "ability to apply knowledge of engineering" can be measured. Another key assignment is to cover analysis and optimization exercise of the FPGA-based hardware model simulation and synthesis and model integration for measuring programming skills and data interpreting ability. Therefore, the student outcome, "master the skills, methods, and knowledge appropriate to the discipline," can be assessed. A comprehensive project is aimed to evaluate students' proficiency in hardware and software design and analysis with contemporary tools.

2) Course Assesement Method

The presented embedded system design course comprises inter- and intra-assessment of the learning outcomes in each key subject and across the frameworks in the integration-oriented methodology described in Section IV-A. In general, many comprehensive-level design courses offer open-ended project. However, the proposed course currently does not include any open-ended project because of the tight course schedule, which is three hours per week in a semester. A similar scale of design experiment, which links the rapid design and verification concepts learned and delivers essential embedded system design and integration skill sets under the WEX-HIL real-time simulation environment, is performed by groups of students.

a) Intra-assesment of the Course

Internal assessment is based on the assessment of outcomes in key subjects, which are identified according to the three primary frameworks classified in the course outline. The assessment is performed via the key assignments given as (1) research manuscript review and presentation, (2) laboratory experiments on model development and verification with SIL/HIL/WEX-HIL real-time simulations, and (3) an EV system prototyping with the WEX-HIL platform, for the students during the specific periods. The key assignment outcomes and the corresponding ABET student learning outcomes are assessed by the construction of the heuristic rule-based EAMU performance vector, which consists of Excellent ($\geq$ 90%), Adequate ($\geq$ 75%), Minimal ($\geq$ 60%), and Unsatisfactory ($<$ 60%). The key assignments are judged by the EAMU vector, which measures and analyzes students' performance.

Table 2 shows the assessment results collected from the courses that enrolled 15 and 19 students and was offered in 2014 and 2015, respectively. All key assignments are designed for groups consisting of three or four students. The other assignments are used for individual assessment. As seen in the EAMU scores (i.e., 3.95 & 4.11) of the first student outcome, our students needed more knowledge on real-time applications and tools. After completion of laboratory experiments, the software/hardware modeling and integration for the real-time simulations were performed more efficiently. The 20.79% EAMU score increase of the second student outcome reflected that our students became more familiar with modeling and integration activities. Furthermore, all five teams' assessment results of the final project in 2015 were higher than 90%, which is the equivalent EAMU score of 5.0. Consequently, the proposed course objectives were satisfied with a contingency of research methodology utilization effectiveness, based on the analysis of the course assessment results. Various sensor systems will be introduced as additional parts to the existing project in order to expend the current scope of embedded system design to the emerging IoT infrastructure.

b) Inter-assesment of the Course

Based on the intra-assessment results, the overall course outcomes and ABET student learning outcomes are measured and analyzed with the **f**aculty **c**ourse **a**ssessment **r**eport (FCAR) [26]. FCAR consists of formative and summative components. Students' performance is monitored through the formative component, which allows the instructor to identify and meet the needs of students. On the other hand, the summative component is to determine the attainment of the course outcomes and ABET student learning outcomes from the EAMU performance vector measured from each key assignment. Therefore, FCAR is used for completing the self-assessment of the course outcomes. The FCAR is a part of the online assessment tool, EvalTools® [27].

V. Conclusions

We developed the WEX-HIL real-time simulation platform for designing various subsystems of an EV in a rapid and intuitive integration manner. The WEX-HIL platform provides a flexible means to perform real-time simulation with various embedded system prototypes and models for academic research and education. The WEX-HIL platform consists of hardware and software subsystems. The FPGA-based hardware subsystem is capable of wirelessly interfacing software models executing in the real-time simulator to the target hardware prototypes, such as an autonomous EV equipped with sensor modules which employed wireless communication devices. In the WEX-HIL platform, the software subsystem consisting of the sensor/wireless communication drivers and the EV models developed in Matlab/Simulink/C and assembly languages executes with the hardware subsystem in virtual and/or real prototypes. Our evaluations of the WEX-HIL real-time simulation proves (1) accuracy of the sensing system (i.e., less than 2.89% of distance error per cm), (2) fast and accurate HIL simulation (i.e., 2.5 times faster acceleration than Simulink acceleration and 0.3% HIL simulation difference compared to Simulink simulation), (3) satisfaction of the 50 μs responding timing constraints of the HIL real-time simulation, and (4) intuitive integration into the real-time simulator at different abstract levels from functional SIL to hardware prototyping HIL simulations.

A comprehensive hands-on multi-interdisciplinary course in Electrical and Computer Engineering was created and offered to graduate students with our development of the WEX-HIL real-time simulation platform. The course comprises of a series of lectures related to the subjects for successful designing and utilizing the WEX-HIL platform and related applications in Computer, Communication, Electrical, and Power-Electronics Engineering. Hands-on laboratories equipped with hardware/software design tools were also developed for further experiments of current and emerging applications in the embedded and real-time systems. The proposed course is expected to alleviate the technology gap developed and utilized between industry and academia. It will also be beneficial for students to prepare for their professional careers in the rapidly evolving and synthesizing technology development world.

Acknowledgment

This work was supported in part by donations from QNX and Xilinx university programs.

References

[1] P. Ranky, "Advanced Digital Automobile Sensor Applications," Sensor Reivew, Vol. 22, No. 3, pp. 213–217, 2002.

[2] G. Parma, and V. Dinahavi, "Real-Time Digital Hardware Simulation of Power Electronics and Drives," IEEE Transactions on Power Delivery, pp. 1235–1246, 2007.

[3] P. Menghal and A. Jaya laxmi, "Real Time Simulation: A Novel Approach in Engineering Education," Proceedings of the IEEE International Conference on Electronics Computer Technology, pp. 215–219, 2011.

[4] R. McNeal and M. Belkhayat, "Standard Tools for Hardware-in-the-Loop (HIL) Modeling and Simulation," Proceedings of the IEEE Electric Ship Technologies Symposium, pp. 130–137, 2007.

[5] H. Salarian, K. Chin, and F. Naghdy, "An energy-efficient mobile sink path selection strategy for wireless sensor networks," IEEE Transactions on Vehicular Technology, Vol. 63, No. 5, pp. 2407–2419, 2014.

[6] D. Georgoulas, "E. In-Motes: A Real Time Application for Automobiles in Wireless Sensor Networks," Wireless Sensor Network, Vol.3, No. 5, pp.158–166, 2011.

[7] L. Cheng and Z. Lipeng, "Hardware-in-the-Loop Simulation and Its Application in Electric Vehicle Development," Proceedings of the IEEE Vehicle Power and Propulsion Conference, 2008.

[8] C. Dufour, S. Abourida, and J. Belanger, "Hardware-in-the-Loop Simulation of Power Drives with RT-LAB," Proceedings of the IEEE International Conference on Power Electronics and Drives Systems, pp. 1646–1651, 2005.

[9] J. Casteres and T. Ramaheriariny, "Aircraft integration real-time simulator Modeling with AADL for architecture tradeoffs," Proceedings of the IEEE Design, Automation & Test in Europe Conference & Exhibition, pp. 346–351, 2009.

[10] A. Rodriguez, R. Metzger, O. Cifdaloz, and T. Dhirasakdanon, "Description of a Modeling, Simulation, Animation,and Real-Time Control (MoSART) Environment for a Class of Electromechanical Systems," IEEE Transactions on Education, Vol. 48, No. 3, pp. 359–374, 2005.

[11] P. Menghal and A. Jaya Laxmi, "Real Time Simulation: A Novel Approach in Engineering Education," Proceedings of the IEEE International Conference on Electronics Computer Technology, pp. 215–219, 2011.

[12] Baogui Zuo, "Research on computer simulation theory and experimental verification technology of the working process of automobile airbag," Proceedings of the International Conference on Management, Education, Information and Control, pp. 625–630, 2015.

[13] P. Venne, J. Paquin, J. Bélanger, "The What, Where and Why of Real-Time Simulation," Technical Report of Opal-RT Technologies, pp.37–49, 2010.

[14] O. Mohammed, N. Abed, and S. Ganu, "Real-Time Simulations of Electrical Machine Drives with Hardware-in-the-Loop," Proceedings of the IEEE Power Engineering Society General Meeting, pp. 1–6, 2007.

[15] S. Abourida, C. Dufour, J. Bélanger, and V. Lapointe, “Real-Time, PC-Based Simulator of Electric Systems and Drives,” Proceedings of the International Conference on Power Systems Transients, pp. 1–6, 2003.

[16] R. Sam, M. Tan, and M. Ismail, “Quad-Copter using ATmega328 Microcontroller,” Proceedings of the International Conference on Electrical Machines and Systems, pp. 566–570, 2013.

[17] Yong Chen, “Application of 5 Kinds of Simulation Analysis in Automobile Engineering,” Science and Technology Review, Vol. 25, No. 17, pp. 65–73, 2007.

[18] T. Silloway, Y. Jung, D. Mackellar, F. Mak, “Design of a Real-Time Simulator for an Electric Vehicle,” Proceedings of the International Conference on Modeling, Simulation and Visualization Methods, pp. 163–168, 2012.

[19] R. Dhole, V. Undre, C. Solanki, S. Pawale, “Smart Traffic Signal Using Ultrasonic Sensor,” Proceedings of the IEEE Int. Conf. on Green Computing Communication and Electrical Engineering, pp. 1–4, 2014.

[20] L. Junsong, et al, “FPGA Based Wireless Sensor Node with Customizable Event-Driven Architecture,” EURASIP Journal on Embedded Systems, pp. 1–11, 2013.

[21] C. Bell, Beginning Sensor Networks with Arduino and Raspberry Pi, New York, NY: Apress, 2013.

[22] C. Chris, “High-Performance PCI Card: Main/Lifestyle Edition,” New Straits Times: 10. 2007.

[23] M. Hou, F. Ni, M. Jin, L. Jiang, H. Liu, Y. Jin, and M. Hou, “The application of real-time operating system QNX in the computer modeling and simulation,” Proceedings of the IEEE International Conference on Artificial Intelligence, Management Science and Electronic Commerce, pp. 6808–6811, 2011.

[24] B. Suh and S. Berber, “Rendezvous points and routing path-selection strategies for wireless sensor networks with mobile sink,” IET Journals & Magazines Electronics Letters, Vol. 52, No. 2, pp. 167–169, 2016.

[25] M. Fujita, S. Sasaki, and K. Matsui, “Object-oriented analysis and design of hardware/software co-designs with dependence analysis for design reuse,” Proceedings of the IEEE International Conference on Information Reuse and Integration, pp.318–325, 2005.

[26] Fong Mak, “Assessing and Tracking Instructional Effectiveness,” ABET symposium 2010.

[27] EvalTools®, http://www.makteam.com.