RTCE: Real-Time Co-Emulation Framework for EMT-Based Power System and Communication Network on FPGA-MPSoC Hardware Architecture

Tong Duan, *Graduate Student Member, IEEE*, Zhen Huang, *Member, IEEE*, and Venkata Dinavahi, *Fellow, IEEE*

***Abstract*—With the expansion of smart grid infrastructure world-wide, modeling the interaction between power systems and communication networks becomes paramount and has created a new challenge of co-simulating the two domains before commissioning. Existing co-simulation methods mostly concentrate on the off-line software-level interface design to synchronize messages between the simulators of both domains. Instead of simulating in software with a large latency, this article proposes a novel real-time co-emulation (RTCE) framework on FPGA-MPSoC based hardware architecture for a more practical emulation of real-world cyber-physical systems. The *discrete-time* based power system electromagnetic transient (EMT) emulation is executed in programmable hardware units so that the transient-level behaviour can be captured in real-time, while the *discrete-event* based communication network emulation is modeled in abstraction-level or directly executed on the hardware PHY and network ports of the FPGA-MPSoC platform, which can perform the communication networking in real-time. The data exchange between two domains is handled within each platform with an extremely low latency, which is sufficiently fast for real-time interaction; and the multi-board scheme is deployed to practically emulate the communication between different power system areas. The hardware resource cost and emulation latency for the test system and case studies are evaluated to demonstrate the validity and effectiveness of the proposed RTCE framework.**

***Index Terms*—Co-emulation, communication network simulation, cyber-physical systems, electromagnetic transients, field programmable gate arrays, multi-processing, real-time systems, system-on-chip.**

List of Acronyms

CPS	Cyber-Physical System
EMT	Electromagnetic Transient
FPGA	Field Programmable Gate Array
MPSoC	Multi-Processing System-on-Chip
TMU	Transient Data Measurement Unit
TDC/STDC	Transient Data Concentrator/Super TDC
TLN	Transmission-Level Networking
LTE	Local Truncation Error
lwIP	Lightweight Internet Protocol
RTCE	Real-Time Co-Emulation.

Manuscript received May 19, 2020; revised August 21, 2020 and October 8, 2020; accepted November 16, 2020. Date of publication November 19, 2020; date of current version April 21, 2021. This work was supported by the Natural Science and Engineering Research Council of Canada. Paper no. TSG-00764-2020. *(Corresponding author: Tong Duan.)*

The authors are with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 2V4, Canada (e-mail: tduan@ualberta.ca; zh2@ualberta.ca; dinavahi@ualberta.ca).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TSG.2020.3039259.

Digital Object Identifier 10.1109/TSG.2020.3039259

I. Introduction

With the innovations in electric power grids and information and communication technologies (ICT), the traditional power systems are evolving into complex cyber-physical systems (CPS), which increasingly leverages the capabilities of data communication and computation to enhance the flexibility of power transfer [1] in the smart grid (SG). One major concern in smart grid research is to simulate the entire behavior of the system to adequately evaluate the influence of the interplay between digital world and physical equipment. However, since the power system and communication network have distinctly different working principles and simulation tools, simulating the entire behavior of the two domains (so-called co-simulation) is quite challenging [1], [2].

Various co-simulation frameworks have been proposed in the recent past since the first interface for the PSCAD/EMTDC simulator [3] to integrate an agent-based distributed application into the simulation was developed. Most of these works are not to design a complete simulator that could finish all required functions in one software package but attempt interfacing two existing simulators of each domain, because there are already various mature power system and communication network simulators available. Unfortunately, there does not exist universally accepted interfaces for data exchange between simulators of the two domains due to their different working principles or even operation systems. Thus, the main concern of these co-simulator frameworks [4]–[9] is to properly handle data exchange and synchronization for related events in both domains at runtime. Most co-simulators mainly distinguish from each other by the type of power system/communication network simulators used, the methods applied for synchronization, and the application scenarios. For example, EPOCHS [7] is mainly used for wide area monitoring, which uses PSCAD/EMTDC and PSLF for power system simulation, and NS-2 for communication network simulation;

the periodic synchronization mechanism is adopted. GECO [8] interfaces the power system simulator PSLF and network simulator NS-2, and the synchronization is based on a global event driven on-demand mechanism. INSPIRE [9] uses dynamic synchronization points for the interfacing of DigSilent and OPNET simulators. In [10] the MATLAB and C++ based platforms are applied for power system and cyber system simulation respectively, and the how a cyber-contingency affects power system operations was investigated. The co-simulation platform presented in [11] was implemented based on OPAL-RT real-time simulator and Riverbed Modeler to examine the power grid vulnerabilities to cyber-physical attacks.

However, the performance of the software-based simulators is relatively low compared with actual power and communication network devices even without taking the data exchange and synchronization time between two simulators into account. If the electromagnetic transient (EMT) is concerned, the simulation process will be extremely slow due to the small time-step size and massive synchronization requirements. It is therefore difficult to practically simulate and test the adequacy of manufactured protection and control equipment responding to potential damage created by transient events in real-time. Although the real-time power system simulation approaches have been extensively studied in past works such as [12]–[15], the real-time co-simulation of power system and communication network has rarely been investigated. In [16] and [17], the real-time co-simulation frameworks were discussed. Similar with the other software-based co-simulators, in these two works, the interface design and synchronization scheme are still the main concerns. To achieve real-time power system simulation, the commercial simulator RT-LAB was utilized; thus the implementation details of the entire co-simulator are absent. In [18] and [19], the hybrid hardware and software platforms are applied: the communication network simulation ran on the OPNET software simulator on PC computers, while the real-time power system simulation was conducted on the RTDS hardware simulator. The interface design between the two different platforms (dedicated hardware and PC software) is even more complicated with loss of generality.

Different from the pure software-based or hybrid hardware and software based *co-simulation* discussed above, there is no relevant work available for the more practical hardware based *co-emulation* on FPGAs, due to the complexity and specificity of the cyber physical power system and FPGA/MPSoC platforms. Although the FPGA-MPSoC platform that enables flexible programmability and highly paralleled computing environment has been used in multi-rate mixed-solver [20], it was purely applied for power system EMT simulation and no communication networks were involved. If the entire resources including the fast and parallel computation capabilities of programmable logics and soft-processors, and the physical ethernet network ports could be leveraged for the co-emulation of the power system and communication network, it can be expected that the testing and evaluation process of cyber-physical energy systems would be more practical and be accelerated significantly.

Based on the above observations, this work proposes a real-time co-emulation (RTCE) framework executed on hardware FPGA-MPSoC platform instead of interfacing two software based simulators. To the best of our knowledge, this is the first work that conducts real-time co-emulation on hybrid FPGA-MPSoC hardware platform. In this work, the entire hardware architecture and implementation details for co-emulating the power system and communication network are given, which utilize all the programmable logics, soft-processor and networking port resources on the FPGA and MPSoC platform, and provide a new vision of co-emulation of emerging cyber-physical systems. The advantages and features of the proposed real-time co-emulator are as follows:

1) Power system EMT emulation is carried out on programmable hardware units so that transients can be captured in real-time;
2) Communication network emulation is modeled in abstraction for transmission-level networking and directly conducted on real-world physical ports to form the function-level networking;
3) Power system and communication network emulation modules are embedded within each FPGA/MPSoC board so that the data exchange and synchronization between the two domains are sufficiently fast for real-time interaction;
4) System monitoring and control applications are executed within the soft processors of FPGA-MPSoC platform so that the system control can be programmed flexibly to respond to the physical system events such as over-currents or communication breakdown such as link failures.

Based on the above advantages, the proposed RTCE framework can achieve real-time co-emulation of power system and communication network, which resembles a real-world cyber-physical system. The IEEE 39-bus system is chosen as the test case emulated on the hybrid FPGA-MPSoC hardware platform, and the real-time emulation results are captured for the over-current and communication link failure case studies. The rest of this article is organized as follows: Section II describes the concept and commonly-used methods of co-simulation. Section III proposes the hardware-based real-time co-emulation framework and emulation principles. Section IV introduces the hardware implementation details of RTCE emulator for the test system. Section V presents the hardware resource cost and emulation results. Finally, the conclusion is drawn in Section VI.

II. Co-Simulation Background

This section introduces the basic concepts and commonly used methods in both simulation domains, and then presents the main challenges of power-communication co-simulation.

A. Power System Simulation

In EMT simulation, each power equipment can be modeled into an equivalent circuit using basic power elements such as the resistor, capacitor and current source. Then the whole power system can be described using a set of differential equations obtained by gathering all the power equipment models and applying circuit laws, where the state variables

of differential equations are to be solved. The numerical integration algorithms such as Trapezoidal Rule and Backward Euler methods are usually applied to those differential equations to solve the state variables within each discrete time-step. Therefore, the EMT power system simulation is discrete-time based simulation, and a small time-step size (typically at micro-second level) is always required to capture the transient level behaviours.

B. Communication Network Simulation

Different from power systems, the elements in communication networks are modeled not based on physical principles but on their *functionality*, and thus there are no differential equations or matrix equations to be solved. Instead, the components in communication networks simply receive, modify and send data packets, and thus the process of each network element is always modeled into a sequence of *discrete events* that occur unevenly in time, where each event represents an operation of packet processing or transmission. In this context, the communication network simulator refers to a program running on software or hardware to simulate the behaviour of a user-defined network topology, and each node in the simulator executes the packet processing procedure similar to real-world equipment.

The basic network component in the cyber-physical power system is the sampling and reporting device, or a smart meter, which is responsible to measure the electrical values of the power system and report the measurements to the system controller. The phasor measurement unit (PMU) [22] is a commonly used basic network component that is deployed in each single process bus or substation model to measure the electrical quantities; and after sampling the data values it computes the corresponding phasor values and then reports them to the phasor data concentrator (PDC), which is responsible to monitor a power area and collects the measurements from the PMUs in the area. In fact, the main function of the communication network is to provide a two-way communication between smart meters, data concentrators and controllers to modify the related circuit parameters according to the service requirements.

C. Co-Simulation

Power system and communication network co-simulation refers to simulating the power system operations while taking the communication network layer into consideration, so that the impact of interaction between the two domains can be evaluated and the entire system features can be revealed. However, power system simulation is time-continuous and the simulation proceeds step-by-step in discrete time, while the communication network simulation is event-driven where the events are unevenly distributed in time. Thus, the main challenge in co-simulation is how to interface the two domain simulators with significantly different work principles.

The interaction between the two domains includes the sampling and reporting operations. The sampling operation refers to measuring and digitizing the values of the concerned

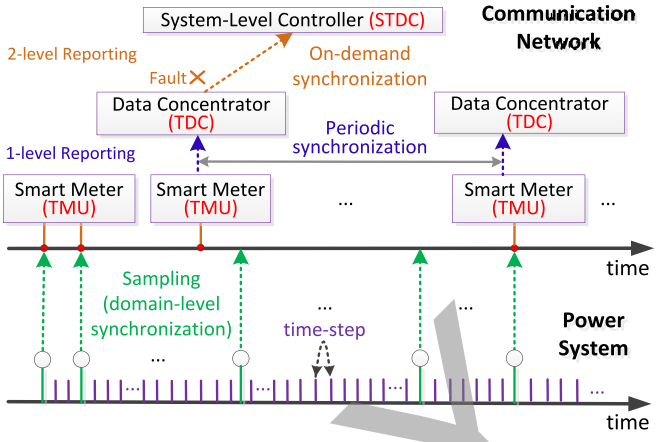


Fig. 1. Example of interaction between power system and communication network simulation.

electrical quantities, which is also called *domain-level synchronization*, since it is used to exchange the data of the two domain simulators. The domain-level synchronization does not involves the network packet encapsulation, which is usually performed periodically at a constant sampling rate for a stable measurement. The reporting operation refers to reporting the measured electrical values from a smart meter to a power area data concentrator, and from a data concentrator to the system controller. It is called the *application-level synchronization*, since it is based on the network data packet and is handled within the communication network domain for the application-level purpose. The application-level synchronization contains a two-stage reporting. The representative approaches for synchronization in each level are the periodic and on-demand methods: when applying the periodic synchronization strategy, the reporting only happens at each synchronization point; for the on-demand synchronization, the data-exchange is only initiated if an event of interest is detected, and under normal conditions the corresponding measurement values are regarded as unchanged. An example is shown in Fig. 1, it can be seen that the domain-level synchronization is performed at a constant sampling rate; while the two stages of the application-level synchronization are performed using the periodic and on-demand strategies respectively.

III. Proposed Real-Time Co-Emulation (RTCE) Framework

Based on the co-simulation background, this section describes the architecture and work principles of the proposed hardware-based real-time co-emulator.

A. Motivation

To conclude from the state of the art, most leading research in this area is either built on existing simulators or the development of a middleware to interface simulators from the two domains; thus their performance is dramatically limited by the software environment and achieving fast simulation for large-scale cyber-physical power systems becomes difficult. There comes an alternative strategy into mind that combines both programmability and high-speed computation: hardware-based

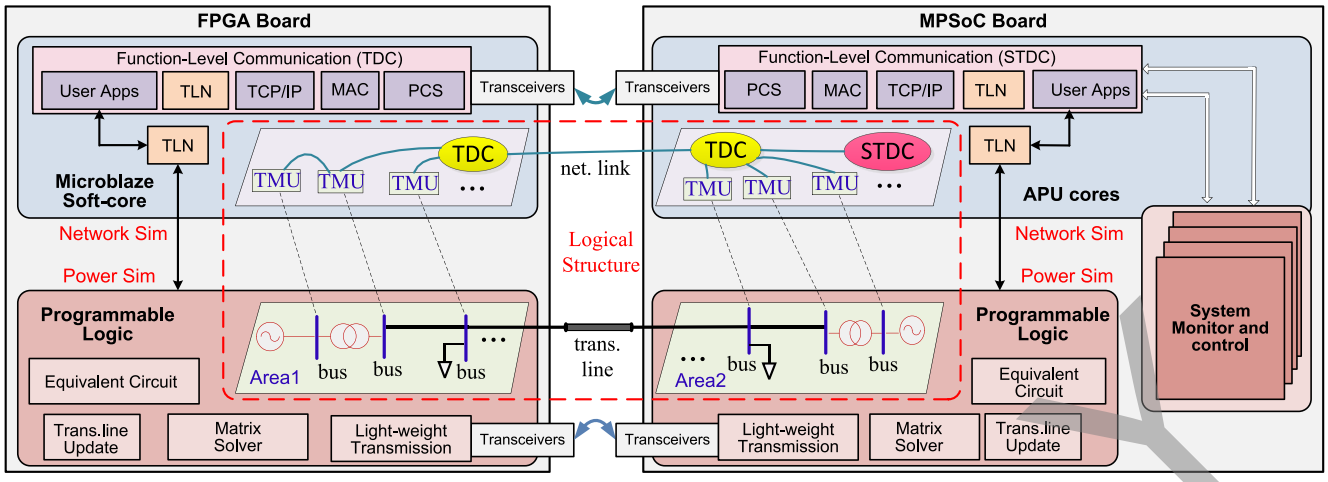


Fig. 2. Demonstration of real-time co-emulation (RTCE) architecture on FPGA-MPSoC platform.

co-emulation. Furthermore, real-time power system simulation has already been extensively carried on FPGA [12]–[14] and communication network functions such as Ethernet, TCP/IP are also available as vendor specific IP cores or soft-codes built on FPGA. The multiprocessor system-on-chip (MPSoC) [21] integrates the FPGA based programmable hardware logic and the ARM based multi-core processor systems within one platform, which enable complex embedded applications to be developed. So why not deploy FPGA-MPSoC platform for implementing power-communication co-simulation? Thus in this work, a real-time co-emulation (RTCE) framework based on an integrated FPGA-MPSoC hardware platform is proposed to take advantage of the above mentioned features.

B. RTCE Hardware Architecture

The top-level architecture of RTCE framework is shown in Fig. 2, wherein the power system EMT emulation and the communication network emulation are integrated on each board. In power system domain, each board emulates a subsystem of the entire system, and emulation results are exchanged with connected boards in each time-step; and in communication network domain, each board emulates the information networking between the buses located in its emulated area. Different from the commonly used PMU and PDC concepts, in this work, the concepts of transient measurement unit (TMU) and transient data concentrator (TDC) are applied, since the measurement data is obtained from the instantaneous transient level EMT power system simulation. The TDC is responsible to monitor the emulated subsystem and collect the measurements from the TMUs; TDC does not have a global view and control applications of the whole power system, thus the super TDC (STDC) is also modeled to receive messages from TDCs and compute the centralized control strategies when detecting abnormal conditions. The corresponding TMU, TDC, and STDC components are marked in Fig. 1.

Power system EMT emulation: For large power systems, it is difficult to emulate the whole circuit topology in one board due to the requirements of huge amount of hardware logic resources and memory. Thus, a common approach is to decompose the system into several subsystems using the inherent latency of the widely distributed transmission lines, and these decomposed subsystems can be allocated to different boards for parallel processing. After the results are obtained in each time-step, they should be exchanged among adjacent subsystems using fast data exchange path, as shown in Fig. 2. It is enough to use light-weight communication such as the Xilinx Aurora protocol for fast data exchange. In EMT emulation, each subsystem can be abstracted into an equivalent circuit along with a matrix equation to be solved, and transmission line data should also be calculated and exchanged between the adjacent subsystems. Thus the power subsystem emulation can divided be into three parts: equivalent circuit parameters updating, matrix equation solution and transmission line data updating, as illustrated in Fig. 2.

Communication network emulation: Since each FPGA/MPSoC board has limited I/O resources and soft-processors, it can only operate as one network device. However, the network nodes (TMUs) in a smart grid can even have the same number as the buses, so it is impossible to emulate all of the network nodes with bit-level details in hardware and obtain the correct transmission parameters between each TMU and STDC. Thus in this work, RTCE emulates the communication network with the two-level model: transmission-level networking and function-level networking. The transmission-level networking is used to model the end-to-end packet transmission characteristics such as delay and loss, while the function-level networking is used to emulate the TDCs and model the specific network functions or protocols such as Ethernet or TCP/IP. The transmission-level emulation is inspired by work [23], which simply models individual network components such as routers, hosts, links, and protocols with specific abstractions rather than completely implementing all the details to avoid complex hardware-based networking executions that are not concerned in power system analysis. In this work, the transmission-level networking models all of the transmission characteristics between TMUs and TDCs and between TDCs and STDC using parameters such as the link losses and delays, which can be obtained through software simulators

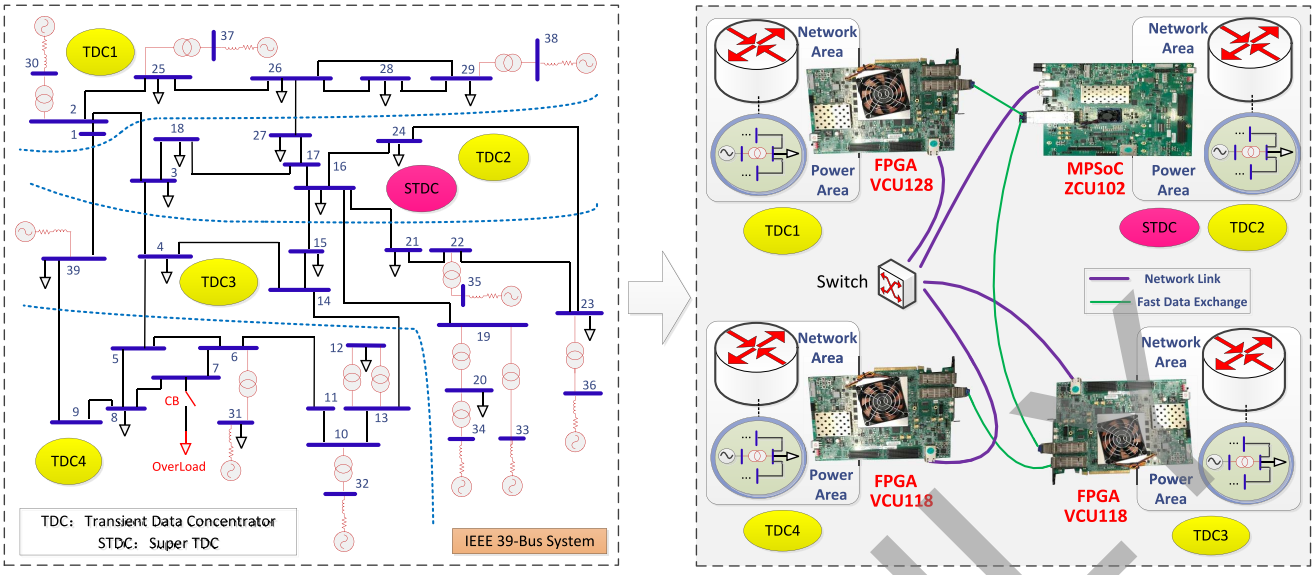


Fig. 3. Example of co-emulating a cyber-physical system on multi-board hardware platform.

such as NS-2/3. The function-level networking is carried out directly on the I/O resources of FPGA/MPSoC boards, because the physical coding sublayer (PCS) and media access controller (MAC) functions can be implemented using hardware logic resources, and the Internet protocol (IP) or other upper network layer functions (TCP, UDP, etc.) can run in the soft-processors (such as Xilinx MicroBlaze) to handle the complex control tasks such as routing, connection establishment and flow control.

As shown in Fig. 2, each TMU receives the measurement from power system emulation as the sampling operation, and then sends the data packet to the corresponding function-level TDC module via the transmission-level networking (TLN) module as the reporting operation. Then the TDCs send data packets to the function-level STDC module via the real-world communication link if detecting abnormal conditions, but still need to pass the TLN module in the function-level networking to emulate the transmission process between the TDC and STDC since TDCs/STDC are not directly connected in practical network. Using this method, the specific network functionalities can be emulated to observe the interaction between power and communication domains, while the detailed transmission processes between TMUs, TDCs and the STDC are simplified and abstracted as crucial and practical transmission parameters.

Co-emulation: In each board, power system EMT emulation and communication network emulation run concurrently. Thus, the data exchange and synchronization between the two domains are simplified: the results of power system emulation in each bus and each time-step can be output to the network emulation module directly without any delays to emulate the sampling process of the TMU in a single process bus. That means, at every hardware clock the results of the two domains can be exchanged within the FPGA logic. If the sampling rate is assigned practically, this data measuring process can also be done for every specific time space. The TMU in each bus is connected with the TDC via the TLN module to send measurement data packets periodically, and TDCs check the received measurement data to find if there exists abnormal conditions; if some transients are detected, the TDC will send messages to STDC for system control command. This on-demand TDC-to-STDC reporting strategy could reduce the amount of generated data packets and thus reduce the congestion of the communication network. The control command from the STDC that may change the circuit topology or other circuit parameters can also be sent to the controllable power devices in the same way so that the power system emulation can respond to the abnormal conditions with low emulation delay. TDCs can run in the soft-processors of FPGA platform, but the system-level monitor and control of STDC should run in the soft-processors of MPSoC platform because the soft-processors in the MPSoC board have more computation resources and a higher clock frequency.

IV. Hardware Implementation of RTCE

To test and verify the advantages of the proposed RTCE framework, the IEEE 39-bus system [24] was selected as the test system, as shown in Fig. 3(a). The test system is partitioned into 4 areas, and TDCs are deployed in each area (at Bus 25, 24, 4, 9 respectively) to accumulate the measurements from each bus and inform the STDC deployed in area 2 (at Bus 24) if detecting abnormal conditions. The hardware RTCE emulator is developed to co-emulate the test system in real-time. As shown in Fig. 3 (b), there are two Xilinx VCU118 FPGA boards, one Xilinx VCU128 FPGA board and one Xilinx ZCU102 MPSoC board used in this emulator, which emulate the partitioned four areas concurrently. The communication network links between RJ-45 ports of each board are connected via a switch, while the fast data exchange links for power system emulation are connected on the SFP/SFP+ ports

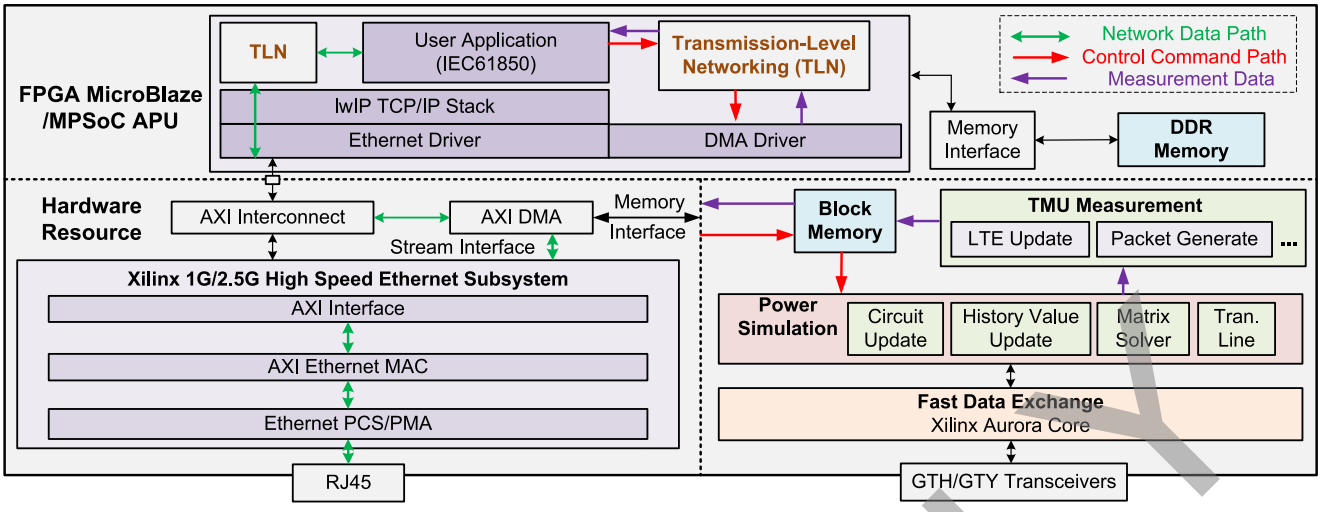


Fig. 4. Illustration of detailed block design on a single FPGA/MPSoC board.

via the 10G optical fiber. The detailed hardware block design in each board is shown in Fig. 4.

A. Multi-Board EMT Emulation

In this work, the applied power equipment models (synchronous machine, power transformer, transmission line and loads, etc.) are just the same as those of PSCAD/EMTDC [25] for verification. For real-time multi-board EMT emulation, the power system is usually partitioned by the long transmission lines. Then the fast data exchange path is required to exchange the transmission line model data among adjacent subsystems. In this work, the lightweight communication module is used: Xilinx 64-bit Aurora core. Instead of using the complex communication functionalities such as the Ethernet protocol, the simple high-speed Aurora communication core enables direct data transfer through different types of GTH/GTY transceivers on FPGA/MPSoC boards with up to 10 Gb/s speed. Aurora enables AXI-4-stream based framing and flow control, which helps the users conveniently generate and receive data without the need of handling transmission states. Based on the on-board test, the latency of Aurora communication is 0.95 μs for data transmission of fifteen 64 bit single floating-point data, which includes the transmission latency and the latency of writing and reading data to/from the storage. This means that it costs less than 1 μs to exchange data between boards within every time-step.

To interact with the communication networking emulation, the abnormal condition detection is the prerequisite to determine the data packets generation. State estimator based on phasor values is usually used to predict the next state variable values and be compared with the real value to determine if the system is running in control. For transient-level analysis of this work, the local truncation error (LTE) is used to estimate the perturbation, and the abnormal condition is found once the LTE or the measured current and voltage exceed the predetermined threshold. After the main conductance matrix is solved in each time-step, the state variables of linear equipment are obtained, and then the LTE of the n^{th} time-step is computed for the linear equipment, given by [26]:

$$LTE(t_n) \approx C_{p+1}\Delta t_n^{p+1}(p+1)!\boldsymbol{g}\left[t_n, \ldots, t_{n-1-p}\right] \tag{1}$$

where C_{p+1} is the error constant of a specific discretization method, p is the order, and $\boldsymbol{g}[t_{n-1}, \ldots, t_{n-1-k}]$ can be calculated as:

$$\boldsymbol{g}(t_{n-1}) = \boldsymbol{x}_{n-1} \tag{2}$$

$$\boldsymbol{g}\left[t_{n-1}, \ldots, t_{n-k}\right] = \frac{\boldsymbol{g}\left[t_{n-1}, \ldots, t_{n-k+1}\right] - \boldsymbol{g}\left[t_{n-2}, \ldots, t_{n-k}\right]}{t_{n-1} - t_{n-k}} \tag{3}$$

For the subsystems where nonlinear equipment exists, the iterative approach is involved. The standard method is to first use an explicit method or interpolation polynomial (called the predictor) to compute a candidate value of the state variable to be solved, and then use it as the initial solution to apply Newton's method for the implicit integrator (called the corrector) until convergence is achieved. Then the LTE can be estimated by comparing the initial solution $\boldsymbol{x}_n^0$ and final solution $\boldsymbol{x}_n$ [26]:

$$LTE(t_n) \approx \frac{C_{p+1}}{1 - C_{p+1}}\left(\boldsymbol{x}_n - \boldsymbol{x}_n^0\right). \tag{4}$$

B. Communication Protocol and Implementation

In order to practically emulate the communication network, the standardized communication protocol, IEC 61850 - Communication Networks and Systems in Substations [27], is applied in this work. IEC 61850 is used to manage a large number of communication devices. The existing communication protocols such as the Ethernet, TCP/IP typically define how data bytes are transmitted on the wire, however, they do not specify the application layer data organization. The IEC 61850 model defines how the application data bits should be arranged and how the created data items/objects and services are mapped to any other existing under layer protocols that can meet the service requirements. In this work, all the communication packets are encapsulated based on the IEC 61850 standard.

In the communication network emulation, each bus is aligned with a TMU, which is able to digitize the base measurement quantities at the source and transmit the resulting sample values to the TDC. The measurement data includes voltages, currents, and some status information (LTE). As introduced in Section III, in RTCE architecture the data is transferred from power emulation to communication networking emulation in each time-step to do real-time EMT analysis, which means the sampling rate of TMU is related to the actually applied time-step size. Different from the TMU measurement gathering that does not involve network packet generation, the TDC collects data from the TMUs in its area based on a specific reporting rate. In this work, the reporting rate from TMU to TDC is set at 60 Hz. As mentioned before, each TDC is required to compare the LTE, current or voltage with the specific threshold; once the measurement exceeds the threshold, the abnormal condition is detected and then it is responsible to create a measurement data packet to be sent to the STDC.

Transmission-level networking: The data packet networking is achieved in two modes: transmission-level networking and function-level networking. In the transmission-level emulation, the transmission process between TMUs and the TDC and between TDCs and the STDC is modeled as bandwidth, communication delay and loss rate. The transmission-level networking was implemented in the soft-processor of FPGA/MPSoC boards, because the transmission parameters can be flexibly configured in software without modification of the hardware design. As illustrated in Fig. 4, for example, if the TMU in Bus 6 generates a packet to be sent to the TDC at Bus 9, then the data packet is generated in the hardware module and then is sent to the soft-processor and passes the TLN function; and if the TDC (run as the function-level networking module) at Bus 9 detects an abnormal condition and send a data packet to the STDC at Bus 24, the packet needs to first pass the TLN function in the application layer of TCP/IP stack and then be sent to the lower TCP/IP and MAC layer. The implementation of the TLN function is quite straightforward: based on the Xilinx timer() function, the TLN function detects the source and destination of the input packet and searches for the corresponding end-to-end transmission delay; then, after the delay outputs the input data packet. The real communication delay is related to the distance and hops between the source and destination, while the loss rate refers to the possibility of an unsuccessful transmission process. In this implementation, the values chosen for the delay and loss are determined based on the testing results from the real network simulator NS-3 [28].

Function-level networking: For function-level communication network emulation, the high-level view is shown in Fig. 4. The Xilinx 1G/2.5G High Speed Ethernet Subsystem core is used to implement the Media Access Controller (MAC) with a Physical Coding Sublayer (PCS) based on the hardware logic and I/O resources. The upper network layer functions are implemented in the Lightweight IP (lwIP) stack [29], which is an open source TCP/IP networking stack for embedded systems and is available in the Xilinx programming tools. The Xilinx Software Development Kit (SDK) provides lwIP software customized to run on various embedded systems that can be MicroBlaze soft-processor in FPGA chip or the ARM-based SoC devices. Through the lwIP application programming interface, users can add networking capability to an embedded system. Since the echo server application that can create a TCP connection and sets up a callback on a connection being accepted is already provided by Xilinx SDK, the IEC 61850 protocol and data encapsulation are implemented based on the existing echo server code to achieve the specific communication patterns of smart grid.

The function-level networking data path is marked as green arrows in Fig. 4, which is achieved by the stream interface between DMA module and Ethernet module; the control command path that is used to send control commands from the controller in the soft-processor to the power emulation module is marked as red arrows in Fig. 4, which is achieved though the memory interface between DMA module and the block memory; the measurement data path that is used to send measurement data from each TMU to the TDC is marked as the purple arrows, which is transmitted via the transmission-level networking module. Note that the measurement data in the block memory on FPGA can only be passively read from the DMA driver function; that means the reporting operation from TMU to TDC is achieved by the application on the lwIP stack to use the driver function to read the values from the block memory periodically. As mentioned before, this reporting rate is set at 60 Hz.

V. Real-Time Emulation Results and Verification

Based on the implemented RTCE hardware emulator, the test system was emulated to evaluate how the communication failure affected the power system, and how the power system transient impacted the communication network.

A. Processing Delay and Hardware Resource Cost

The advantages of the proposed FPGA-MPSoC based co-emulation framework should be evaluated by comparing with the existing approaches, however, since the existing co-simulators are not fully open-sourced and are implemented on disparate platforms, it is extremely difficult to get the results of the test power system using the existing co-simulators for comparison. In fact, the greatest advantage of the proposed co-emulator is that it utilizes the hardware computing and networking resources to conduct fast and practical co-emulation of the EMT-based power system and communication network, and this advantage can be revealed in the real-time performance. Thus in this section, the processing latencies of the power system emulation, the communication network emulation parameters, and the data transmission performance of the interaction between the two domains are first described to show how the RTCE framework can achieve the real-time co-emulation performance.

For the hardware emulation, the FPGA boards (including the Microblaze softprocessor) run at the clock frequency of 100 MHz. According to the system partition and RTCE implementation described above, the hardware resource consumption and maximum processing latency for one time-step

TABLE I
HARDWARE RESOURCE CONSUMPTION OF THE CASE STUDY

Resource	VCU128	ZCU102	VCU118-1	VCU118-2
LUT	40.3%	95.2%	57.1%	55.2%
FF	36.1%	70.1%	50.3%	48.9%
BRAM	73.3%	86.9%	77.4%	75.5%
DSP	35.6%	87.2%	64.1%	62.8%
Latency	10.1μs	8.7μs	16.2μs	13.1μs

are presented in Table I. The VCU118-1/2 boards emulate the power areas 3 and 4, while the VCU128 and ZCU102 boards emulate the areas 1 and 2 respectively. The processing latency of one time-step includes the latency of power equipment model computation, matrix solution, transmission line updating, history item exchange (including board-to-board fast data exchange), and LTE computation, which indicates the minimum time-step size that can be applied for real-time EMT emulation. Since the two ends of a transmission line can be computed concurrently based on the traveling wave line model, the main circuit matrix actually can be decomposed into small matrices. The maximum size of matrix to be solved in the four power areas is: 6*6, 3*3, 12*12, and 9*9, respectively. In this work the power areas in the four boards are emulated using the same time-step, 20 μs, which is larger than the latencies of each subsystem and is sufficient for EMT-level analysis.

For the communication network emulation that runs concurrently, since the transmission-level communication behaviors are parameterized by real emulator results and the function-level emulation is just executed on the hardware transceivers for real world networking, the communication network emulation also runs in real-time. The real-time networking performance is actually confined by the transmission-level parameters and function-level hardware capabilities. In addition, the TCP/IP protocol execution and system-level control performance are limited by the computational capabilities of soft-processors in FPGA and APU cores on the MPSoC board. In this work, the tested function-level networking performance is 984 Mbit/s based on the lwIP performance testing program; the transmission delay of each communication link in the NS-3 simulator is set at 1 ms/200 km, which is a classical value used in communication simulation; and the forwarding rate of each network device that impacts the delay of packet processing within a device is set at 100 Mbit/s, which is also a commonly used value.

For the interaction between the two domains, the delay actually refers to the delay of the reporting operation between the FPGA MicroBlaze/MPSoC APUs soft-processor and the block memory on FPGA, since the sampling operation is performed directly on the programmable logic with zero delay. On the FPGA board, the tested throughput of the reading data operation of the driver function is 186 Mbit/s; on the MPSoC board, the test throughput is 214 Mbit/s. Note that the driver read function does not influence the power system simulation, which means, the data exchange delay between the two domains is sufficiently small for real-time interaction.

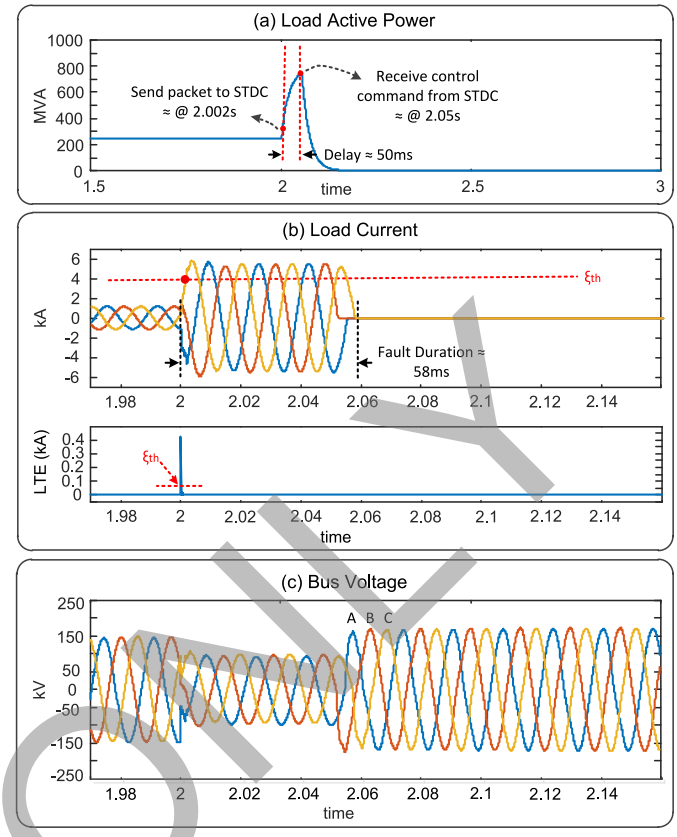


Fig. 5. Overcurrent fault case study: (a) active power of load at Bus 7; (b) total load current at Bus 7; (c) voltage of Bus 7.

B. Case Study 1: Over-Current Fault

To study how the power system fault conditions affects the entire cyber-physical system, a sudden load increase case is applied on Bus 7 at emulation time 2 s, which results in a over-current fault. Upon detecting this condition, the STDC will perform the centralized algorithm for protection. The emulation results are shown in Fig. 5.

As shown in the results, after the fault is initiated, the corresponding message (including the bus voltage, load current, and LTE) is sent to STDC due to the sudden transient-level increase of LTE. Subsequent messages are generated because the load current exceeds the threshold value. And about 50 ms after the fault, the controllable circuit breaker at Bus 7 receives the control command from STDC to open the load circuit for protection. The time delay between the fault and the response message mainly includes the transmission time and computation latency for control command. In this work, the control program is just developed for this case study thus it consumes a very low latency. The transmission time consists of the in-board communication (referred to TLN delay between Bus 7 and TDC4 and between TDC4 and STDC) and board-to-board communication latency. Since the board-to-board communication delay is extremely low, it can be omitted compared to the TLN delay that is measured through the software network simulator. The real-time emulation results indicate that the interaction between the two domains is quite fast. Besides, as shown in Fig. 5(b), (c), when co-emulating the two domains,

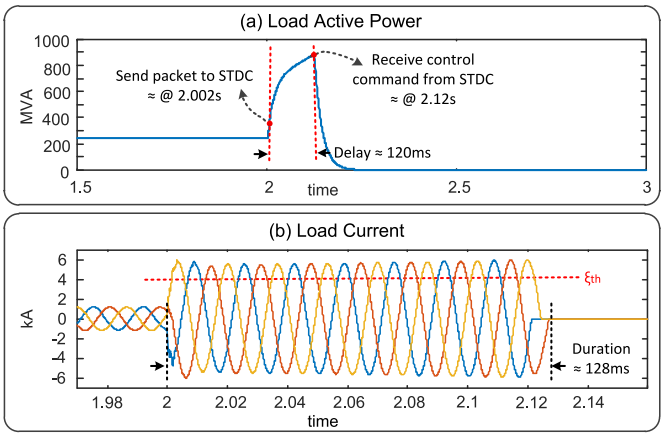


Fig. 6. Communication link failure case study: (a) active power of load at Bus 7; (b) total load current at Bus 7.

the transient-level waveforms can also be captured and output in real-time to observe the details of the value change due to the small time-step applied.

C. Case Study 2: Communication Link Failure

To investigate the effect of network link failure, the link between Bus 8 and Bus 9 that is essential for the communication between TMUs and TDC breaks down at emulation time 2 s, while the over-current fault is also applied at the same time. Since the on-demand synchronization strategy is applied, during normal conditions the influence is small, because the power system is regarded as working normally if there is no message sent from TDCs. However, when the overcurrent fault happens, this impact is essential to the system response. The system controller in STDC is required to respond to this situation according to the messages received from TDCs, however, since the link broke down, re-computing a new route costs a considerable time, and the generated new route has a much longer distance between the TMU at Bus 7 and TDC at Bus 9, which also results in an increased communication delay.

As shown in Fig. 6, the system is supposed to respond to the over-current fault caused by the sudden load change within a short delay, however, due to the transmission link failure, the system response becomes relatively slow. Since the link failure impacts the transmission-level communication delay, the increased latency is actually obtained from the NS-3 simulator. From Fig. 6(b) it can be observed that the total duration of overcurrent fault raises to 128 ms, which greatly increases the duration of damage.

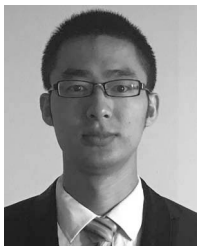
VI. Conclusion

For fast co-simulation of cyber-physical systems, this work proposes to use the FPGA-MPSoC based hardware platform instead of the software-based architecture to conduct the emulation. In the proposed real-time co-emulation (RTCE) framework, the power system EMT emulation is carried out on programmable hardware units so that transients can be captured in real-time; and the communication network emulation is modeled in abstraction for transmission-level characteristics and directly conducted on real-world physical ports to form the specific networking functions, which can emulate the communication networking in real-time. The power system and communication network emulation modules are embedded within each FPGA/MPSoC board so that the data exchange and synchronization between the two domains are sufficiently fast for real-time interaction. The IEEE 39-bus system test system is implemented and emulated on the FPGA-MPSoC platform. The hardware resource costs and processing latency are evaluated, and the real-time emulation results of power system faults and communication link faults cases show the effectiveness of the RTCE emulator. The proposed RTCE framework can be utilized for the study of emerging cyber-physical systems as well as fast emulation of test systems. More practical smart grids including power electronics equipment [30] and AC/DC transmission will be considered on the RTCE emulator in future work.

References

[1] K. Mets, J. A. Ojea, and C. Develder, "Combining power and communication network simulation for cost-effective smart grid analysis," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1771–1796, 3rd Quart., 2014.
[2] S. C. Müller *et al.*, "Interfacing power system and ICT simulators: Challenges, state-of-the-art, and case studies," *IEEE Trans. Smart Grid*, vol. 9, no. 1, pp. 14–24, Jan. 2018.
[3] M. Baran, R. Sreenath, and N. Mahajan, "Extending EMTP for simulating agent based distributed applications," *IEEE Power Eng. Rev.*, vol. 22, no. 12, pp. 52–54, Dec. 2002.
[4] J. Bergmann, C. Glomb, J. Götz, J. Heuer, R. Kuntschke, and M. Winter, "Scalability of smart grid protocols: Protocols and their simulative evaluation for massively distributed DERs," in *Proc. 1st IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, Oct. 2010, pp. 131–136.
[5] T. Godfrey, S. Mullen, R. C. Dugan, C. Rodine, D. W. Griffith, and N. Golmie, "Modeling smart grid applications with co-simulation," in *Proc. 1st IEEE Int. Conf. Smart Grid Commun. (SmartGrid-Comm)*, 2010, pp. 291–296.
[6] V. Liberatore and A. Al-Hammouri, "Smart grid communication and co-simulation," in *Proc. IEEE EnergyTech*, May 2011, pp. 1–5.
[7] K. Hopkinson, X. Wang, R. Giovanini, J. Thorp, K. Birman, and D. Coury, "EPOCHS: A platform for agent-based electric power and communication simulation built from commercial off-the-shelf components," *IEEE Trans. Power Syst.*, vol. 21, no. 2, pp. 548–558, May 2006.
[8] H. Lin, S. S. Veda, S. S. Shukla, L. Mili, and J. Thorp, "GECO: Global event-driven co-simulation framework for interconnected power system and communication network," *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1444–1456, Sep. 2012.
[9] H. Georg, S. C. Müller, C. Rehtanz, and C. Wietfeld, "Analyzing cyber-physical energy systems: The INSPIRE cosimulation of power and ICT systems using HLA," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2364–2373, Nov. 2014.
[10] Y. Cao, X. Shi, Y. Li, Y. Tan, M. Shahidehpour, and S. Shi, "A simplified co-simulation model for investigating impacts of cyber-contingency on power system operations," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 4893–4905, Sep. 2018.
[11] A. A. Jahromi, A. Kemmeugne, D. Kundur, and A. Haddadi, "Cyber-physical attacks targeting communication-assisted protection schemes," *IEEE Trans. Power Syst.*, vol. 35, no. 1, pp. 440–450, Jan. 2020.
[12] Y. Chen and V. Dinavahi, "FPGA-based real-time EMTP," *IEEE Trans. Power Del.*, vol. 24, no. 2, pp. 892–902, Apr. 2009.
[13] Y. Chen and V. Dinavahi, "Hardware emulation building blocks for real-time simulation of large-scale power grids," *IEEE Trans. Ind. Informat.*, vol. 10, no. 1, pp. 373–381, Feb. 2014.
[14] J. Marguč, M. Truntič, M. Rodič, and M. Milanovič, "FPGA based real-time emulation system for power electronics converters," *Energies*, vol. 12, no. 6, p. 969, 2019.
[15] K. Ma and Y. Song, "Power-electronic-based electric machine emulator using direct impedance regulation," *IEEE Trans. Power Electron.*, vol. 35, no. 10, pp. 10673–10680, Oct. 2020.

[16] Q. Wang, W. Tai, Y. Tang, Y. Liang, L. Huang, and D. Wang, "Architecture and application of real-time co-simulation platform for cyber-physical power system," in *Proc. 7th IEEE Int. Conf. Cyber Technol. Autom. Control Intell. Syst.*, Honolulu, HI, USA, Jul./Aug. 2017, pp. 81–85.
[17] G. Cao *et al.*, "Real-time cyber-physical system co-simulation testbed for microgrids control," *IET Cyber-Phys. Syst., Theory Appl.*, vol. 4, no. 1, pp. 38–45, Mar. 2019.
[18] B. Chen, K. L. Butler-Purry, A. Goulart, and D. Kundur, "Implementing a real-time cyber-physical system test bed in RTDS and OPNET," in *Proc. North Amer. Power Symp. (NAPS)*, Pullman, WA, USA, Sep. 2014, pp. 1–6.
[19] H. Tong, M. Ni, L. Zhao, and M. Li, "Flexible hardware-in-the-loop testbed for cyber physical power system simulation," *IET Cyber-Phys. Syst., Theory Appl.*, vol. 4, no. 4, pp. 374–381, Dec. 2019.
[20] T. Duan, Z. Shen, and V. Dinavahi, "Multi-rate mixed-solver for real-time nonlinear electromagnetic transient emulation of AC/DC networks on FPGA-MPSoC architecture," *IEEE Power Energy Technol. Syst. J.*, vol. 6, no. 4, pp. 183–194, Dec. 2019.
[21] *Zynq UltraScale+ MPSoC Processing System V3.3—LogiCORE IP Product Guide*, document PG-201, Xilinx, San Jose, CA, USA, Jan. 2020.
[22] *IEEE Standard for Synchrophasor Measurements for Power Systems*, IEEE Standard C37.118.1, 2011.
[23] E. Moradi-Pari, N. Nasiriani, Y. P. Fallah, P. Famouri, S. Bossart, and K. Dodrill, "Design, modeling, and simulation of on-demand communication mechanisms for cyber-physical energy systems," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2330–2339, Nov. 2014.
[24] *PSCAD™ IEEE 39 Bus System, Reversion 1*, Manitoba HVDC Res. Centre, Winnipeg, MB, Canada, 2010.
[25] *EMTDC® User's Guide: A Comprehensive Resource for EMTDC, Version 4.7*, Manitoba HVDC Res. Centre, Winnipeg, MB, Canada, 2010.
[26] F. N. Najm, *Circuit Simulation.* Hoboken, NJ, USA: Wiley, 2010, pp. 241–252.
[27] "Communication networks and systems for power utility automation—Part 1: Introduction and overview," Int. Electrotech. Commission, Geneva, Switzerland, Rep. IEC TR 61850-1:2013, 2013.
[28] F. Aalamifar, A. Schlogl, D. Harris, and L. Lampe, "Modelling power line communication using network simulator-3," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Atlanta, GA, USA, Dec. 2013, pp. 2969–2974. [Online]. Available: http://www.ece.ubc.ca/faribaa/paper.pdf
[29] *LightWeight IP Application Examples*, document XAPP1026 (v5.1), Xilinx, San Jose, CA, USA, Nov. 2014.
[30] R. Burgos and J. Sun, "The future of control and communication: Power electronics-enabled power grids," *IEEE Power Electron. Mag.*, vol. 7, no. 2, pp. 34–36, Jun. 2020.



Tong Duan (Graduate Student Member, IEEE) received the B.Eng. degree in electronic engineering from Tsinghua University, Beijing, China, in 2013. He is currently pursuing the Ph.D. degree in electrical and computer engineering with the University of Alberta, Edmonton, AB, Canada. His research interests include real-time simulation of power systems, communication networking, parallel computing, and field-programmable gate arrays.

Zhen Huang (Member, IEEE) received the B.S. degree in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 2012, the M.S. degree in electrical engineering from Tsinghua University, Beijing, China, in 2015, and the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Canada, in 2020. He also has two years work experience in power electronic industry. His research interests include high-performance simulation and control of power electronics and power systems.

Venkata Dinavahi (Fellow, IEEE) received the B.Eng. degree in electrical engineering from the Visveswaraya National Institute of Technology, Nagpur, India, in 1993, the M.Tech. degree in electrical engineering from the Indian Institute of Technology Kanpur, India, in 1996, and the Ph.D. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, in 2000. He is currently a Professor with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada. His research interests include real-time simulation of power systems and power electronic systems, electromagnetic transients, device-level modeling, large-scale systems, and parallel and distributed computing.