

Hardware-in-the-Loop Emulation of Linear Induction Motor Drive for MagLev Application

Behzad Jandaghi, *Student Member, IEEE*, and Venkata Dinavahi, *Senior Member, IEEE*

Abstract—Linear induction machines are widely used in transportation systems due to their many advantages. Design and prototyping of electric machines are an expensive and time-consuming process; hardware-in-the-loop simulation provides an efficient alternative. In this paper, a field-programmable gate array-based real-time digital emulation of single-sided linear induction motor with the drive system is proposed. Implementation of the model is performed in both fixed-point using Xilinx system generator and floating-point number representations using a handwritten VHDL Hardware Description Language code. Then, an evaluation in terms of real-time step-size and accuracy as well as hardware resource utilization is provided. The whole design was fully paralleled, which resulted in a considerable reduction of model execution time. The minimum time step of 2.3 and 0.8 μ s was achieved for floating-point and fixed-point implementations, respectively. The results of the real-time simulation are verified by the experimental results as well as a 2-D finite-element simulation in JMAG software.

Index Terms—Field-programmable gate arrays (FPGAs), fixed point, floating point, hardware-in-the-loop (HIL), linear induction motor (LIM), magnetically levitated (MagLev), real-time systems, rotor field-oriented control.

I. INTRODUCTION

LINEAR induction motors (LIMs) provide both the propulsion force in the horizontal direction and the levitation force in the vertical direction. As a distinct feature, LIMs levitation force is required in conveyance systems from small to large-scale applications [1], [2]. Two major large-scale applications include magnetically levitated (MagLev) vehicles [3], [4] and electromagnetic launch systems [5], [6]. LIMs are widely used in the transportation system, especially in urban areas, due to many advantages, including high speed and noiseless operation, as well as fast acceleration and deceleration as a result of controlling the levitation force.

An equivalent circuit model of the LIM to formulate the machine behavior was first proposed in [7], which was derived from the equivalent circuit of a rotary induction machine. Different approaches were proposed to derive the equivalent circuit parameters, including magnetic equations of the

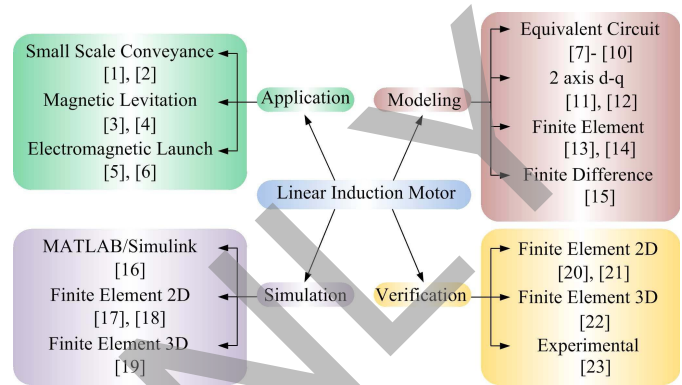


Fig. 1. Taxonomy of literature on LIM.

air gap [8], field analysis [9], and winding functions [10]. Based on the presented equivalent circuits, the two axis $d-q$ model has been derived in [11] and [12], which is a useful tool for control purposes. Other models presented for LIM include the Finite element model (FEM) in 2-D [13] and 3-D [14] and the finite difference time domain method [15]. Some other studies were conducted for simulation and validation of the LIM in [16]–[23]. A taxonomy of research performed on LIMs in different areas of application, modeling, simulation, and validation is shown in Fig. 1.

In the process of prototyping an electric machine drive system, the design of the controllers requires several cycles of testing, which have the drawbacks of expensive equipment, human resources, time-consuming construction and testing, destructive to actual equipment, power consumption, and physical space requirement. Conventionally, offline simulations were used to evaluate the behavior of an electric machine and its drive, which suffer from long simulation time as well as inaccuracy due to simplification of the model. Furthermore, offline simulations cannot emulate the behavior of an electric component interacting with other equipment under realistic conditions.

Hardware-in-the-loop (HIL) simulation is an efficient and cost-effective approach to predict the behavior of a system prior to manufacturing and commissioning. HIL simulation allows testing newly designed simulated component interaction with external devices in a nondestructive environment, minimizing the error of the testing setup by removing inaccurate modeling of the interacted devices [24]. This scenario is efficient when the simulated component can exchange data in real time. Real-time simulation is achieved when the system computations take less time than a specified time step.

Manuscript received February 8, 2015; revised October 6, 2015 and December 17, 2015; accepted February 17, 2016. Date of publication March 15, 2016; date of current version April 8, 2016. This work was supported by the Natural Sciences and Engineering Research Council of Canada. (Corresponding author: Behzad Jandaghi.)

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

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TPS.2016.2535460

0093-3813 © 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.

See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

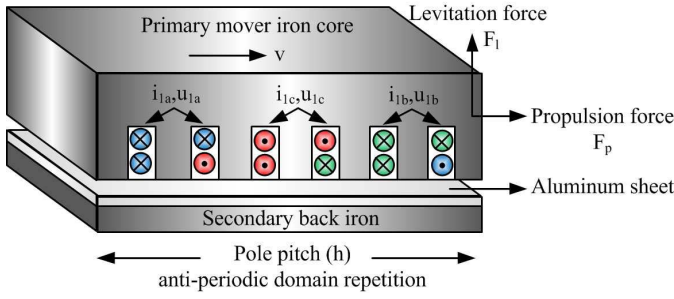


Fig. 2. Structure of LIM.

The time step should be specified small enough in such a way that the required accuracy is satisfied. Recently, field-programmable gate array (FPGAs) have gained a lot of attention in real-time modeling and emulation of electrical machines, power electronics, and large-scale power systems [25]–[32], due to their parallel hardwired architecture and reconfigurability.

There are two hardware implementation methods: the schematic block method of Xilinx system generator (XSG) with fixed-point number representation and the textual programming language using VHSIC Hardware Description Language (VHDL) with floating-point number representation. Up to now, no contribution in the area of the HIL simulation is available for LIM drive systems. This paper focuses on the real-time simulation of a complete LIM drive system for magnetic levitation application using both the fixed-point and floating-point number representations on FPGA.

In this paper, first, the rotor field-oriented control scheme is applied to LIM to facilitate decoupled control of the levitation and propulsion forces. Second, the derived machine model accompanied with the drive system model is used to obtain parallelized architecture for hardware representation using the VHDL codes with both the fixed-point and floating-point number representations. Then, the models are implemented on the Xilinx Virtex-7 (XC7VX485T-2FFG1761C) FPGA in order to emulate the machine behavior and control system characteristics in real time. Finally, the real-time results are verified with the experimental results as well as the finite-element simulation using JMAG software, and the discussions are provided.

II. HARDWARE-IN-THE-LOOP EMULATION OF LIM DRIVE

A. Hardware Realization of LIM

1) *Model Formulation*: The structure of LIM consists of the primary and secondary parts is shown in Fig. 2. The primary mover includes the iron core and a three-phase winding supplied by an external source to create the magnetic field. The secondary includes an aluminum sheet to conduct the induced current providing a secondary magnetic field and the back iron to provide a low reluctance flux path. The interaction between the two magnetic fields provides the propulsion and levitation forces in the horizontal and vertical directions, respectively.

The LIM model can be extracted from the conventional model of a rotary induction machine, considering the primary current and the secondary flux as the state variables

as follows [33]:

$$\frac{di_{1q}}{dt} = -\frac{\pi}{h}v_e i_{1d} - \left(\frac{R_1}{\sigma L_1} + \frac{1-\sigma}{\sigma T_2}\right)i_{1q} - \frac{L_m \pi}{\sigma L_1 L_2 h}v \lambda_{2d} + \frac{L_m}{\sigma L_1 L_2 T_2}\lambda_{2q} + \frac{1}{\sigma L_1}u_{1q} \quad (1)$$

$$\frac{di_{1d}}{dt} = -\left(\frac{R_1}{\sigma L_1} + \frac{1-\sigma}{\sigma T_2}\right)i_{1d} + \frac{\pi}{h}v_e i_{1q} + \frac{L_m}{\sigma L_1 L_2 T_2}\lambda_{2d} + \frac{L_m \pi}{\sigma L_1 L_2 h}v \lambda_{2q} + \frac{1}{\sigma L_1}u_{1d} \quad (2)$$

$$\frac{d\lambda_{2q}}{dt} = \frac{L_m}{T_2}i_{1q} - \left(\frac{\pi}{h}v_e - \frac{\pi}{h}v\right)\lambda_{2d} - \frac{1}{T_2}\lambda_{2q} \quad (3)$$

$$\frac{d\lambda_{2d}}{dt} = \frac{L_m}{T_2}i_{1d} - \frac{1}{T_2}\lambda_{2d} + \left(\frac{\pi}{h}v_e - \frac{\pi}{h}v\right)\lambda_{2q} \quad (4)$$

where u_1 , i_1 , R_1 , and L_1 are the voltage, the current, the per phase resistance, and the self-inductance of the primary mover, respectively, L_m is the mutual inductance between the primary and the secondary, L_2 and R_2 are the secondary inductance and resistance, $T_2 = (L_2/R_2)$ is the secondary time constant and $\sigma = 1 - (L_m^2/L_1 L_2)$ is the leakage factor, λ_{2d} and λ_{2q} are the secondary flux in the d and q axes, h is the pole pitch, v_e is the synchronous linear speed, and v is the linear speed of the primary mover.

The equation of motion for the LIM is as follows:

$$F_p = M\dot{v} + Dv + F_L \quad (5)$$

where F_p and F_L are the LIM propulsion and load forces, respectively. M is the mover mass and D is the viscous friction and iron loss factor.

The propulsion and levitation force equations can be presented in terms of state variables, i.e., the d - q primary currents and secondary fluxes as follows:

$$F_p = \frac{3\pi L_m}{2hL_2}(\lambda_{2d}i_{1q} - \lambda_{2q}i_{1d}) \quad (6)$$

$$F_l = \frac{3L_m}{4gL_2}(\lambda_{2d}i_{1d} + \lambda_{2q}i_{1q}) \quad (7)$$

where F_l is the levitation force and g is the air gap.

The state-space system of equations of (1)–(4) discretized using forward Euler can be presented as follows:

$$\begin{cases} \mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \Delta t[\mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t)] \\ \mathbf{y}(t) = \mathbf{C}(t)\mathbf{x}(t) + \mathbf{D}(t)\mathbf{u}(t) \end{cases} \quad (8)$$

where $\mathbf{x} \in \mathbb{R}^{4 \times 1}$ is the state variable vector, i.e., $\mathbf{x}^T = [i_{1q}, i_{1d}, \lambda_{2q}, \lambda_{2d}]$, $\mathbf{u} \in \mathbb{R}^{2 \times 1}$ is the input primary mover voltage vector, and Δt is the time step. $\mathbf{y}(t)$ is the output primary mover vector, i.e., the propulsion and levitation forces. \mathbf{A} , \mathbf{B} , \mathbf{C} , and \mathbf{D} are the coefficient matrices depending on machine parameters, as presented in (1)–(7).

2) *Hardware Emulation*: In a hardware design, compromising between the hardware resource utilization, the simulation speed, and the accuracy is challenging and depends on the case study. Massively parallel implementation of algebraic calculations (single data per logic resource per time step) implies more resource utilization with the advantage of reaching less time step and simplicity. However, fully pipelined implementation (multiple data per logic resource per time step)

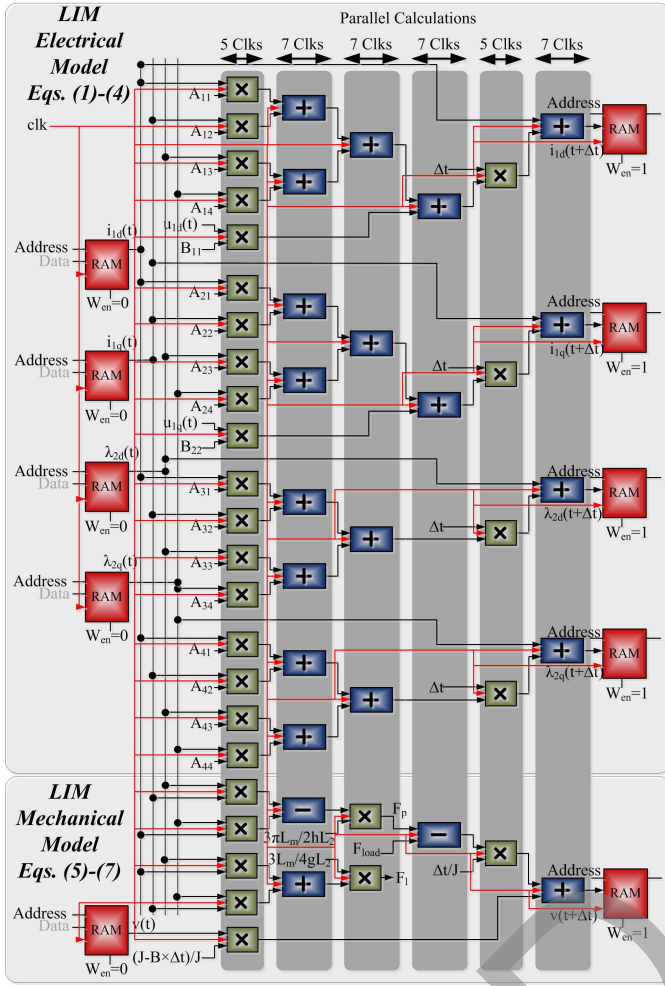


Fig. 3. Massively parallel state-space hardware implementation of LIM model.

means the use of an IP core several times in each time step, which implies utilizing less hardware resources while increasing complexity as well as having larger time step. In most implementations for the optimal hardware design, both the parallel and pipelined architectures should be used. For the case of vector control of electric machines, due to small amount of calculations, the hardware resource utilization might not be a problem. Therefore, parallelism is more suitable to achieve minimum latency.

The details of both electrical and mechanical machine model implementations based on (1)–(8) using the floating-point IP cores are shown in Fig. 3 in a massively parallel architecture by performing the independent multiplications and additions in parallel. The blocks can also be implemented using the fixed-point schematic block of XSG in the desired architecture. The latency of seven and five clocks is considered for additions and multiplications, respectively, and the LIM hardware emulation is reached within 47 clocks latency. The state variables of the next time step are written to 32-b RAMs with the depth of 16 in order to be used as the input of the next time step.

B. Hardware Realization of the Drive System

1) *Model Formulation:* Decoupled control of the propulsion and levitation forces of LIMs can be extracted from decoupled

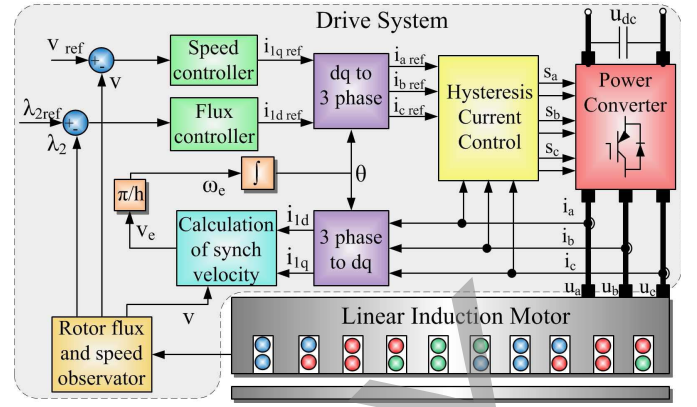


Fig. 4. LIM rotor field-oriented control scheme.

control of rotor flux and torque of conventional rotary induction machines, respectively.

Ideally, for decoupled control, the secondary flux is controlled to be aligned with the d -axis, that is

$$\lambda_{2q} = 0. \quad (9)$$

As a result, the propulsion and levitation forces can be controlled independently through the q and d components of the primary current, respectively. Fig. 4 shows the rotor field-oriented control schematic.

Conventional PI controllers are used for the primary speed and secondary flux control through i_{1qref} and i_{1dref} , respectively. The discretized speed controller based on the first-order forward Euler method can be obtained as follows:

$$i_{1qref}(t + \Delta t) = i_{1qref}(t) + k_p[2v_{err}(t) - v_{err}(t - \Delta t)] + [-k_p + k_i \Delta t]v_{err}(t) \quad (10)$$

where

$$v_{err}(t) = v_{ref}(t) - v(t). \quad (11)$$

Similarly, for the flux control unit

$$i_{1dref}(t + \Delta t) = i_{1dref}(t) + k_{pf}[2\lambda_{2derr}(t) - \lambda_{2derr}(t - \Delta t)] + [-k_{pf} + k_{if} \Delta t]\lambda_{2derr}(t) \quad (12)$$

where

$$\lambda_{2derr}(t) = \lambda_{2dref}(t) - \lambda_{2d}(t). \quad (13)$$

The next time step position can be calculated as follows:

$$\beta(t + \Delta t) = \beta(t) + \frac{\pi}{h}v_e(t)\Delta t \quad (14)$$

where

$$v_e(t) = v(t) + \frac{h}{\pi} \frac{i_{1q}(t)L_m R_2}{\text{int}(t)L_2} \quad (15)$$

in which int is an intermediate signal that can be calculated as follows:

$$\text{int}(t) = \frac{L_2 - R_2 dt}{L_2} \text{int}(t - \Delta t) + \frac{L_m R_2 dt}{L_2} i_{1d}(t - \Delta t). \quad (16)$$

The switching model of an inverter is used to emulate a two-level three-phase converter as a voltage source supplying the LIM.

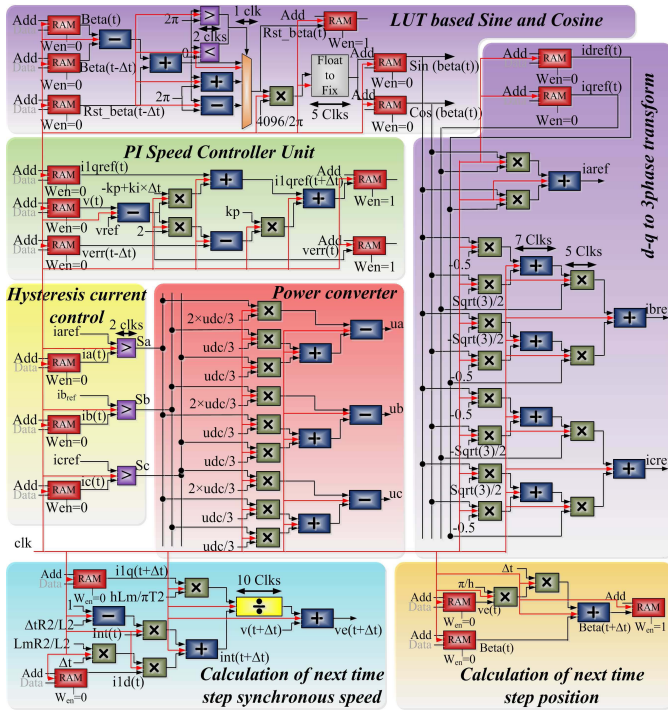


Fig. 5. Massively parallel hardware implementation of rotor field-oriented control units.

2) *Hardware Emulation*: The details of hardware architecture of the control system units, including PI and hysteresis controllers, switching model of the power converter, speed and position calculations, and sine and cosine functions for d - q to three-phase conversion, are shown in Fig. 5 with a synchronous clock signal. The design is fully paralleled in order to decrease the time step. The sine and cosine functions required for electric machine modeling are generated by lookup table (LUT). To do so, the sine and cosine values corresponding to each angle need to be stored in RAMs. The accuracy of the sine and cosine generators using this method depends on the depth of the RAMs, which is 4096 for 2π in this paper. In order to use the angle as the input address of the RAM, the periodicity of sine and cosine functions is used to find the corresponding angle between $[0, 2\pi]$. It can be achieved by adding the corresponding angle of machine primary mover position increment to the previous time step angle between $[0, 2\pi]$. Then, the angle was mapped to the address of the RAM by the gain of depth/ 2π , as shown in Fig. 5, where the width of each block corresponding to the latency of each IP core is shown.

C. FPGA Implementation

Two types of implementation exist depending on the number representation the using schematic blocks of XSG with fixed-point and VHDL textual programming language with floating-point number representation.

For the fixed-point evaluation of the drive system, XSG can be used as a tool for the implementation of the system in a relatively simple manner in comparison with HDL coding. XSG provides the schematic blocks called visual block

programming language for different operations in MATLAB/Simulink environment. Then, the VHDL code based on the fixed-point number representation can be generated and synthesized using ISE or Vivado softwares to be implemented on FPGA through the JTAG cable for real-time emulation. In the fixed-point number representation, a specific number of bits should be allocated for integer and binary part of the number. In order to capture the best accuracy with a specified number of bits, the information about the value of each signal should be known, which is a disadvantage of the fixed-point number representation.

With allocation of the same number of bits, a floating-point number representation format allows covering a wider range of numbers, which cause having more accurate results without requiring any information about the value of the data. According to the IEEE standard 754, numbers have two floating-point formats, i.e., single-precision and double-precision formats with 32 and 64 b wide, respectively.

The entire system signals for a data path is shown in Fig. 6. States 0–State 5 form the algorithm to calculate all signals for an LIM drive system in each time step. In the design architecture, a massively parallel approach for the algebraic calculations is used in order to achieve the minimum time step. Most calculations that are independent can be performed in parallel at the beginning of each time step (State 0) using the values calculated in the previous time step, when the forward Euler method for discretization is utilized.

The process begins in State 0 with the calculation of the sine and cosine of the corresponding rotary angle of the LIM primary mover position, required for d - q to three-phase conversion matrix in State 1, in parallel with the calculation of the reference d - q current of the next time step as well as the propulsion and levitation forces of the current time step. Using the mechanical equation (5), with the calculation of the propulsion force, the next time step speed is calculated in State 1. In State 1, Park's transformation is implemented in order to calculate three-phase reference currents from the reference d - q currents. With the comparison of the three-phase reference currents with measured currents, the switching signals using hysteresis current control can be achieved, which results in the determination of three-phase voltages in State 2. State 3 represents the LIM equations in state space in order to calculate state variables of the next time step using the state of the system at the current time step. Then, the measured three-phase current as well as the electrical speed of the next time step is calculated in State 4. At the end of each time step (State 5), RAMs should be updated to be used as the input for the next time step. Fig. 7 shows the FPGA-based real-time emulator setup.

III. RESULTS AND DISCUSSION

A. Real-Time Simulation Results

In the real-time closed-loop LIM drive system, the machine speed, the secondary flux, and the propulsion load references are considered as the inputs. The machine current is controlled in a way that the frequency depends on the machine speed and the amplitude depends on the propulsion and levitation forces.

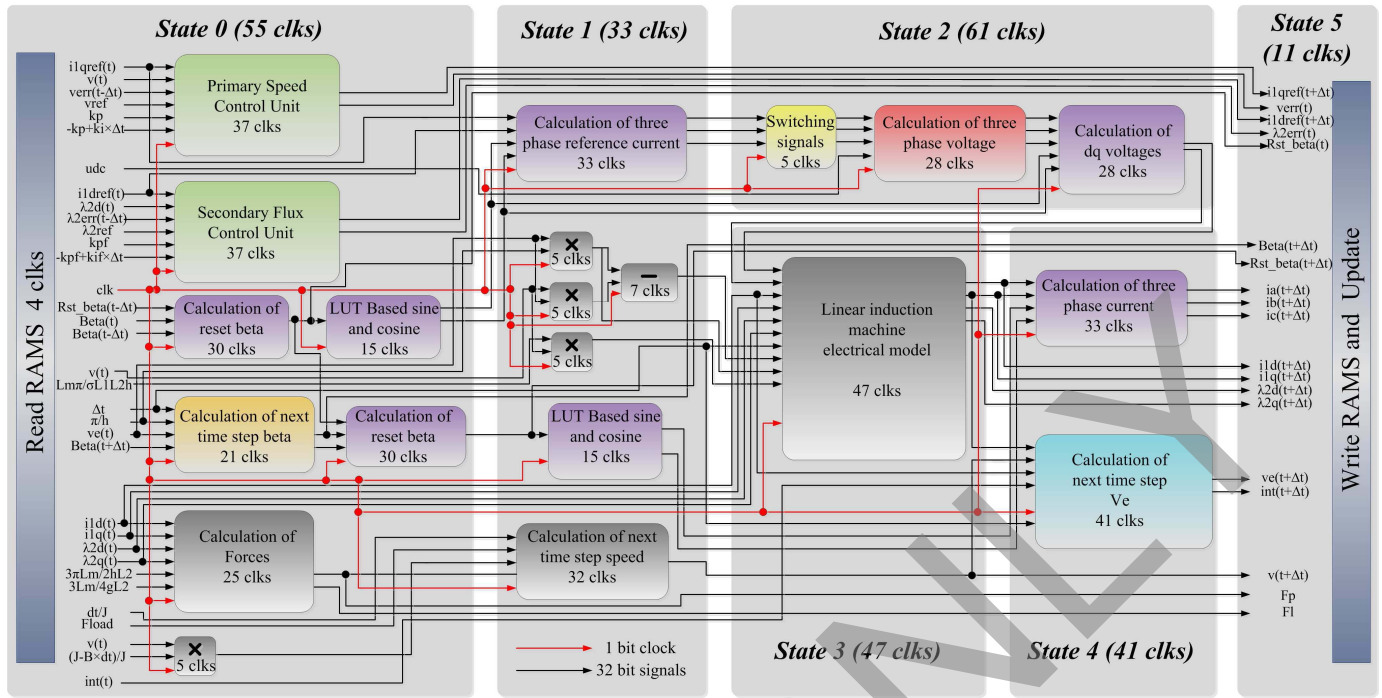


Fig. 6. Signal integration in the hardware emulation architecture for the LIM drive system (modules scaled by latency).

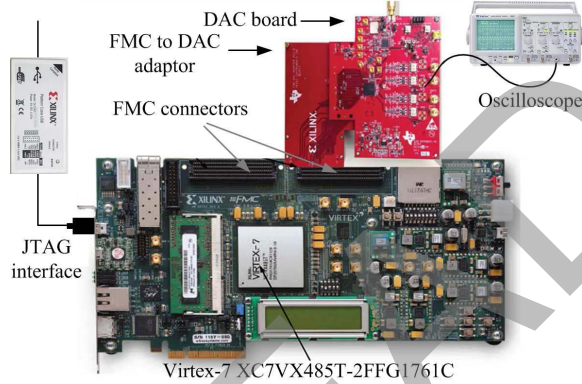


Fig. 7. Real-time emulator setup for LIM drive system.

A practical sequence of references according to time intervals based on a real transportation system is applied to emulate the LIM behavior as follows.

- 1) 0–1 s: For departure, at stand still, the secondary flux is ramped up to 0.5 Wb to provide the levitation force, which creates a fixed air gap for contactless operation.
- 2) 1–2 s: With the constant flux, the machine is accelerated and the speed and the propulsion force are ramped up to 20 m/s and 17 500 N, respectively.
- 3) 2.8–3.2 s: A disturbance in load force is considered as the result of wind speed or rail curve.
- 4) 4–5 s: The machine is decelerated and the speed is ramped down to stop in the station as a result of reduction in propulsion force and viscous friction.
- 5) 5–6 s: In the station, at stand still, more passengers are getting on and additional flux build up is made to keep the air gap fixed for appropriate contactless operation.

- 6) 6–7 s: The speed and the additional amount of propulsion force are ramped up to 20 m/s and 20 000 N due to the additional passengers.
- 7) 8–8.3 s: Emergency electromagnetic braking for an unexpected event is applied by negative propulsion force that results in sharp speed ramp down.

The real-time oscilloscope results are shown in Fig. 8.

B. Floating-Point Versus Fixed-Point Comparison

1) *Accuracy Evaluation:* In the LIM control system, online control of the secondary flux is performed by the estimation of the reference secondary flux to adjust the levitation force in order to keep the air gap fixed. Therefore, an accurate calculation of the secondary flux is crucial. Fig. 9 shows the accuracy of flux calculation by means of both the fixed-point and floating-point number representations, when a 0.5-Wb step change on the secondary flux is applied. The minimum achievable time step for each design is utilized, and the forward Euler method is used as the discretization method. In order to have a fair comparison, 32-b single-precision floating-point format (IEEE Standard 754) and 32-b fixed-point format have been chosen for number representations. The binary point in the fixed-point is located to achieve the maximum accuracy, e.g., for small values, more number of bits are allocated for the binary part, and for large values, more number of bits are allocated for the integer part. Considering the MATLAB m-file double-precision number calculation as the reference, it can be seen that the calculation of the secondary flux with the floating-point number representation using VHDL code is more accurate than the fixed-point using XSG, since, with the same number of bits, the floating-point format can represent a wider range of numbers.

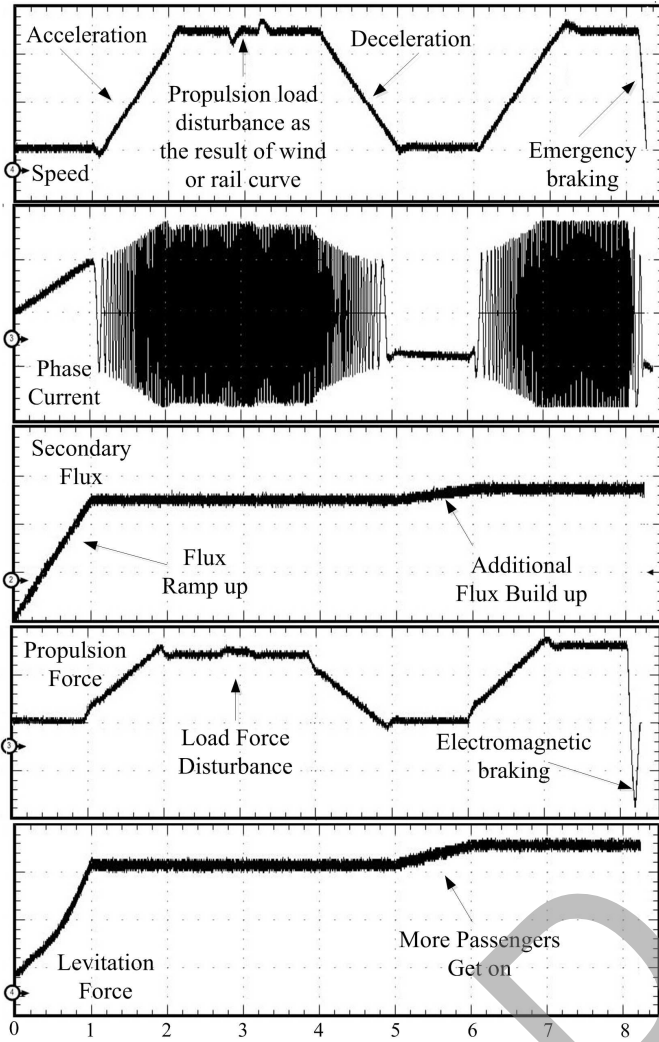


Fig. 8. Floating-point real-time results of LIM (time: 1 s/div, speed: 8 m/s/div, current: 1440 A/div, secondary flux: 0.2 Wb/div, propulsion force: 12000 N/div, and levitation force: 16000 N/div).

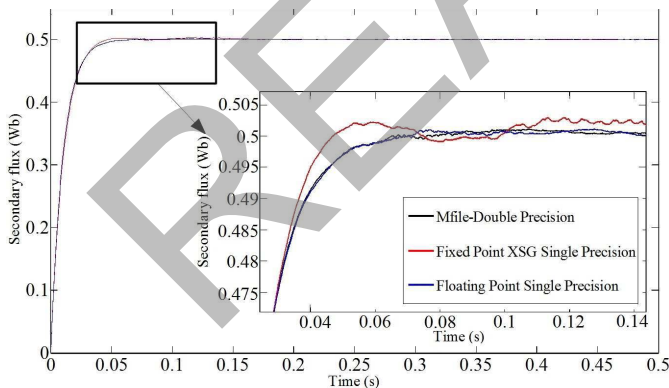


Fig. 9. Fixed-point and floating-point accuracy comparison.

2) *Timing Analysis*: The time step of a hardware design is based on the signal path through the circuit from input to output with the longest delay called the critical path of the system, which generally contains the largest number of logic gates. The real-time emulator time step size is important, since currently available FPGAs-based HIL setups are not

TABLE I
HARDWARE RESOURCE UTILIZATION

Resources	Implementation Method	
	Floating-Point	Fixed-Point
Slice Registers	89,114 (14%)	52,632 (8%)
Slice LUTs	101688 (33%)	33,842 (11%)
Occupied Slices	26,452 (34%)	12,758 (16%)
Memory	13,483 (10%)	0 (0%)
DSP48	0 (0%)	168 (6%)
Bonded IOBs	83 (11%)	83 (11%)

able to accommodate hardware design time steps more than a few microseconds due to practical limitations on data transfer between the setup and actual external devices.

Assigning appropriate delay for each algebraic calculation is arbitrary, while at the time of implementation, the FPGA clock would be affected. Therefore, seven clock delay is considered for addition or subtraction and five clock delay for multiplication. With the specified delays and hardware implementation architecture, the floating-point implementation has 248 clock latency with the FPGA clock of 9.61 ns, which resulted in a time step size of 2.3 μ s, while the fixed-point implementation has 11 clock latency with the FPGA clock of 74.19 ns, which resulted in a time step size of 816 ns. Thus, with fixed-point implementation, a smaller time step size can be achieved.

3) *Hardware Resource Utilization*: The designed architectures are implemented on Xilinx Virtex-7 XC7VX485T FPGA with available 607 200 slice registers, 303 600 slice LUTs, 75 900 slices, 130 800 memory units, 2800 DSP48, and 700 bonded Input/Output Blocks. The hardware resource utilization in terms of the number of logic resources and the occupation percentage with both the fixed-point using XSG and the floating-point using the VHDL code is shown in Table I. It can be seen that the floating-point approach requires more hardware resources, since the algorithms of floating-point computations require more hardware with the advantage of having more accuracy in comparison with fixed-point computations.

C. Result Verification

1) *Finite-Element Simulation*: The real-time emulation results were validated by the Finite element model (FEM) simulation of the full structure LIM through JMAG designer software on a PC with Intel Core i7-2600 CPU at 3.4 GHz. The simulation time for 0.16 s with the time step of 1.25 ms with 194 743 of elements and 107 685 nodes is performed in 8 h and 25 min, utilizing the Newton-Raphson method as the nonlinear solver and incomplete Cholesky-conjugate gradient method for solving the system of linear equation. With supplying the primary mover, the flux passes through the secondary aluminum sheet and induces eddy currents. The steady-state flux and eddy current distributions in the LIM are shown in Fig. 10.

2) *Experimental Setup*: The case study used is an industrial sample of a single-sided short primary LIM with the length of 3.33 m, the width of 26.67 cm, and the weight

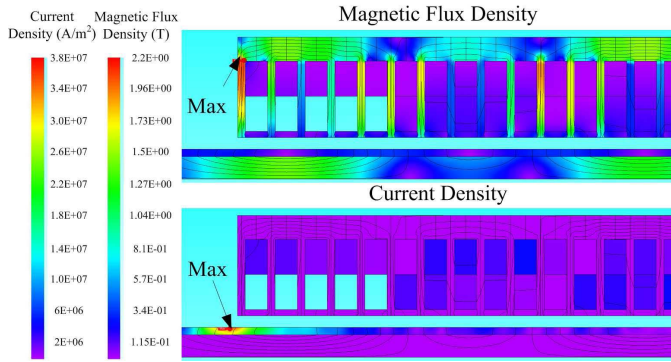


Fig. 10. Finite element result of flux and current distribution in JMAG.

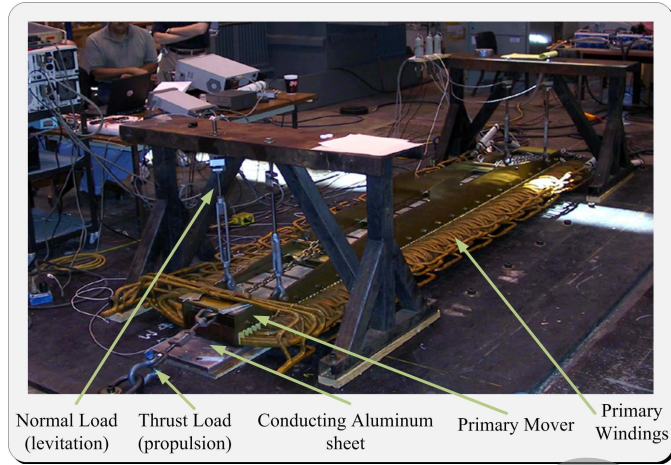


Fig. 11. LIM experimental setup, courtesy of American Maglev Technology and Prof. K. Davey, used in [34] at Center of Electromechanics at the University of Texas at Austin.

of 1136 kg constructed at the Center of Electromechanics, University of Texas at Austin, where the experimental results were obtained. The experimental setup of the LIM is shown in Fig. 11.

Fig. 12 shows the propulsion and levitation forces versus speed change, indexed with slip frequency, with three different methods, i.e., experimental measurements, 2-D FEM, and the real-time emulation with the speed of 26.82 m/s and the primary current of 1543 A. The difference between the FEM simulation and the experimental result comes from the inaccurate measurement of the propulsion and levitation forces due to practical circumstances. However, the difference between the real-time emulation and the FEM arises from the model used. The real-time simulation is based on the lumped $d-q$ model, which does not consider the distributed nature, the nonlinearity of the iron core, and the end effects. Considering these reasons, the results are in acceptable agreement.

IV. CONCLUSION

In this paper, a complete LIM drive system is emulated on FPGA for real-time simulation with the advantage of HIL testing, interacting with other actual external devices. The implementation is performed through an evaluation of the fixed-point using XSG and floating-point number representations using the handwritten VHDL code. It is shown that with the floating-point number representation, more accurate results

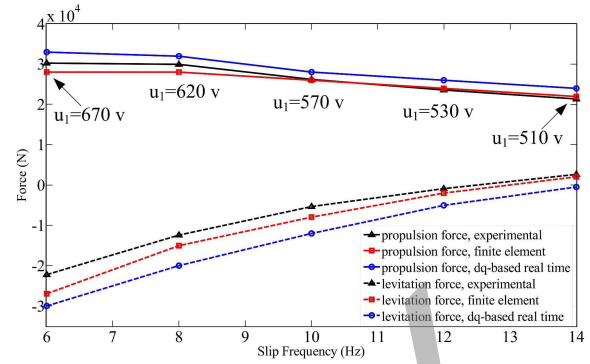


Fig. 12. Torque versus machine speed with experimental, real-time, and finite element analysis.

can be achieved with the cost of requiring more hardware resources and larger time step, when both the fixed-point and floating-point implementations are performed in a massively parallel approach. The compatibility of the real-time results with offline simulations and verification by experiments as well as finite-element simulation shows the accuracy and effectiveness of the proposed scheme in high-speed execution. For the large-scale systems, hardware emulation using FPGAs has the limitation of hardware resource as well as requiring large time step that can affect the accuracy. The real-time emulation of electric machines considering the nonlinearity of the iron core through $d-q$, magnetic equivalent circuit, and finite-element models in order to obtain more accurate results is interesting to be investigated for future works.

ACKNOWLEDGMENT

The authors would like to thank T. Morris from American Maglev Technology, and Prof. K. Davey from the University of Texas at Austin for providing the data of the industrial sample linear induction machine required for this paper.

REFERENCES

- [1] G. V. Sadler and A. W. Davey, "Applications of linear induction motors in industry," *Proc. Inst. Elect. Eng.*, vol. 118, no. 6, pp. 765–776, Jun. 1971.
- [2] J. S. Choi and Y. S. Baek, "Magnetically-levitated steel-plate conveyance system using electromagnets and a linear induction motor," *IEEE Trans. Magn.*, vol. 44, no. 11, pp. 4171–4174, Nov. 2008.
- [3] Y. Cho and J. Lee, "An investigation on the characteristics of a single-sided linear induction motor at standstill for maglev vehicle," *IEEE Trans. Magn.*, vol. 33, no. 2, pp. 2073–2076, Mar. 1997.
- [4] U. Hasirci, A. Balıkcı, Z. Zabar, and L. Birenbaum, "A novel magnetic-levitation system: Design, implementation, and nonlinear control," *IEEE Trans. Plasma Sci.*, vol. 39, no. 1, pp. 492–497, Jan. 2010.
- [5] M. R. Doyle, D. J. Samuel, T. Conway, and R. R. Klimowski, "Electromagnetic aircraft launch system-EMALS," *IEEE Trans. Magn.*, vol. 31, no. 1, pp. 528–533, Jan. 1995.
- [6] H. Chen and Q. Wang, "Modeling of switched reluctance linear launcher," *IEEE Trans. Plasma Sci.*, vol. 41, no. 5, pp. 1123–1130, May 2013.
- [7] J. Duncan, "Linear induction motor-equivalent-circuit model," *IEE Proc. B, Electr. Power Appl.*, vol. 130, no. 1, pp. 51–57, Jan. 1983.
- [8] W. Xu *et al.*, "Equivalent circuits for single-sided linear induction motors," *IEEE Trans. Ind. Appl.*, vol. 46, no. 6, pp. 2410–2423, Nov./Dec. 2010.
- [9] J. Y. Lu and W. Ma, "Research on end effect of linear induction machine for high-speed industrial transportation," *IEEE Trans. Plasma Sci.*, vol. 39, no. 1, pp. 116–120, Jan. 2011.

- [10] W. Xu, G. Sun, G. Wen, Z. Wu, and P. K. Chu, "Equivalent circuit derivation and performance analysis of a single-sided linear induction motor based on the winding function theory," *IEEE Trans. Veh. Technol.*, vol. 61, no. 4, pp. 515–525, May 2012.
- [11] G. Kang and K. Nam, "Field-oriented control scheme for linear induction motor with the end effect," *IEEE Proc.-Electr. Power Appl.*, vol. 152, no. 6, pp. 1565–1572, Nov. 2005.
- [12] D. Hall, J. Kapinski, M. Krefta, and O. Christianson, "Transient electro-mechanical modeling for short secondary linear induction machines," *IEEE Trans. Energy Convers.*, vol. 23, no. 3, pp. 789–795, Sep. 2008.
- [13] B.-I. Kwon, K.-I. Woo, and S. Kim, "Finite element analysis of direct thrust-controlled linear induction motor," *IEEE Trans. Magn.*, vol. 35, no. 3, pp. 1306–1309, May 1999.
- [14] T. Yamaguchi, Y. Kawase, M. Yoshida, Y. Saito, and Y. Ohdachi, "3-D finite element analysis of a linear induction motor," *IEEE Trans. Magn.*, vol. 37, no. 5, pp. 3668–3671, Sep. 2001.
- [15] M. Poloujadoff and H. El Khashab, "A finite difference model of linear induction motors, taking into account the finite length of iron," *IEEE Trans. Power App. Syst.*, vol. PAS-101, no. 8, pp. 2966–2974, Aug. 1982.
- [16] S.-M. Jang, Y.-S. Park, S.-Y. Sung, K.-B. Lee, H.-W. Cho, and D.-J. You, "Dynamic characteristics of a linear induction motor for predicting operating performance of magnetic levitation vehicles based on electromagnetic field theory," *IEEE Trans. Magn.*, vol. 47, no. 10, pp. 3673–3676, Oct. 2011.
- [17] B.-J. Lee, D.-H. Koo, and Y.-H. Cho, "Investigation of linear induction motor according to secondary conductor structure," *IEEE Trans. Magn.*, vol. 45, no. 6, pp. 2839–2842, Jun. 2009.
- [18] Q. Lu, Y. Li, Y. Ye, and Z. Q. Zhu, "Investigation of forces in linear induction motor under different slip frequency for low-speed maglev application," *IEEE Trans. Energy Convers.*, vol. 28, no. 1, pp. 145–153, Mar. 2013.
- [19] Y. Zhang, W. Ma, J. Lu, Z. Sun, J. Xu, and W. Li, "A new approach to research the transverse edge effect in linear induction motor considering the edge fringing flux," *IEEE Trans. Magn.*, vol. 47, no. 11, pp. 4660–4668, Nov. 2011.
- [20] A. Z. Bazghaleh, M. R. Naghashan, and M. R. Meshkatoddini, "Optimum design of single-sided linear induction motors for improved motor performance," *IEEE Trans. Magn.*, vol. 46, no. 11, pp. 3939–3947, Nov. 2010.
- [21] T. Yang, L. Zhou, and L. Li, "Influence of design parameters on end effect in long primary double-sided linear induction motor," *IEEE Trans. Plasma Sci.*, vol. 39, no. 1, pp. 192–197, Jan. 2011.
- [22] A. Shiri and A. Shoulaie, "Design optimization and analysis of single-sided linear induction motor, considering all phenomena," *IEEE Trans. Energy Convers.*, vol. 27, no. 2, pp. 516–525, Jun. 2012.
- [23] S. G. Lee, H.-W. Lee, S.-H. Ham, C.-S. Jin, H.-J. Park, and J. Lee, "Influence of the construction of secondary reaction plate on the transverse edge effect in linear induction motor," *IEEE Trans. Magn.*, vol. 45, no. 6, pp. 2815–2818, Jun. 2009.
- [24] M. O. Omar Faruque and V. Dinavahi, "Hardware-in-the-loop simulation of power electronic systems using adaptive discretization," *IEEE Trans. Ind. Electron.*, vol. 57, no. 4, pp. 1146–1158, Apr. 2010.
- [25] N. R. Tavana and V. Dinavahi, "A general framework for FPGA-based real-time emulation of electrical machines for HIL applications," *IEEE Trans. Ind. Electron.*, vol. 62, no. 4, pp. 2041–2053, Apr. 2015.
- [26] N. R. Tavana and V. Dinavahi, "Real-time FPGA-based analytical space harmonic model of permanent magnet machines for hardware-in-the-loop simulation," *IEEE Trans. Magn.*, vol. 51, no. 8, Aug. 2015, Art. no. 8106609.
- [27] J. Liu and V. Dinavahi, "Detailed magnetic equivalent circuit based real-time nonlinear power transformer model on FPGA for electro-magnetic transient studies," *IEEE Trans. Ind. Electron.*, vol. 63, no. 2, pp. 1191–1202, Feb. 2016.
- [28] W. Wang, Z. Shen, and V. Dinavahi, "Physics-based device-level power electronic circuit hardware emulation on FPGA," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2166–2179, Nov. 2014.
- [29] Z. Shen and V. Dinavahi, "Real-time device-level transient electro-thermal model of the modular multilevel converter on FPGA," *IEEE Trans. Power Electron.*, vol. PP, no. 99, pp. 1–14, Nov. 2015.
- [30] Y. Chen and V. Dinavahi, "An iterative real-time nonlinear electromagnetic transient solver on FPGA," *IEEE Trans. Ind. Electron.*, vol. 58, no. 6, pp. 2547–2555, Jun. 2011.
- [31] Y. Chen and V. Dinavahi, "Multi-FPGA digital hardware design for detailed large-scale real-time electromagnetic transient simulation of power systems," *IET Generat., Transmiss., Distrib.*, vol. 7, no. 5, pp. 451–463, May 2013.
- [32] 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.
- [33] I. Boldea and S. A. Nasar, *Linear Electric Actuators and Generators*. Cambridge, U.K.: Cambridge Univ. Press 1997.
- [34] K. R. Davey, "The equivalent T circuit of the induction motor: Its nonuniqueness and use to the magnetic field analyst," *IEEE Trans. Magn.*, vol. 43, no. 4, pp. 1745–1748, Apr. 2007.



Behzad Jandaghi (S'15) received the B.Sc. degree in electrical engineering from the Power and Water University of Technology, Tehran, Iran, in 2009, and the M.Sc. degree in electrical engineering from the Sharif University of Technology, Tehran, in 2011. He is currently pursuing the Ph.D. degree with the University of Alberta, Edmonton, AB, Canada.

His current research interests include real-time digital simulation of electrical machines, drives, and power electronics.



Venkata Dinavahi (S'94–M'00–SM'08) received the Ph.D. degree 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 current research interests include real-time simulation of power systems and power electronic systems, large-scale system simulation, and parallel and distributed computing.