Validation of Frequency- and Time-domain Fidelity of an Ultra-low Latency Hardware-in-the-Loop (HIL) Emulator

Elaina Chai, Ivan Celanovic
Institute for Soldier Nanotechnologies
Massachusetts Institute of Technology
Cambridge, MA 02139
echai@mit.edu

Jason Poon
Department of EECS
University of California, Berkeley
Berkeley, CA 94720
jason@berkeley.edu

Abstract—This paper presents frequency- and time-domain validation of an ultra-low latency real-time hardware-in-the-loop (HIL) emulator. We investigate the frequency domain fidelity of the small-signal transfer functions, the input impedance, and the output impedance of a buck converter emulated on HIL. We show that the frequency response of the hardware-in-the-loop emulation closely matches the analytical results for input impedance, output impedance, line-to-output, and control-to-output transfer functions. Additionally, we present time domain comparisons between the real-time emulation and a reference hardware design under steady state and transient conditions. Our results demonstrate the ultra-high fidelity of the hardware-in-the-loop emulator and also present its capability as a design optimization and verification tool for advanced power electronics controller development.

Keywords—Hardware-in-the-loop, real-time emulation, controller development

I. INTRODUCTION

As the applications and complexity of power electronics systems continue to grow, there is an increasing demand for fast and accurate tools for improving power electronics controls design, test, and verification processes. Hardware-in-the-loop (HIL) has been receiving increased attention as an indispensable tool for designing and validating complex controls systems. Hardware-in-the-loop enables control engineers to test real controller systems by directly interfacing them with a real-time emulation of a converter power stage. Because engineers are able to quickly test controllers under a wide range of operating and fault conditions without the need for a high-power lab, the amount of time and resources required for design, testing, and validation of these systems can be drastically reduced. In addition, the HIL configuration enables repeatable and formalized testing over a spectrum of operating conditions (including faults) that are often impossible or too expensive to test with real hardware.

Until recently however, the computational and latency requirements for real-time simulation of fast, non-linear switching power electronics have limited the speed, fidelity, and applicability of HIL emulators in power electronics. For instance, in [1], the authors have designed and tested a real-time HIL simulator of an H-bridge inverter. However, because the HIL simulation relied on a time averaged model with a minimum time step of 250 µs, the authors were unable to characterize the model beyond 100 Hz, which limits the fidelity of the model. In another study, authors have presented fast real-time HIL simulators with a minimum time step of 10 µs [2]. While they have validated the results of the system under steady state conditions, the fidelity of the system for fast transient conditions and in the frequency domain have not been presented. Lastly, in [3], the authors have presented results under steady state and transient conditions for a real-time HIL simulator with a PWM sampling resolution of 12.5 ns and 10 µs simulation time step. However, the fidelity of this system in the frequency domain has not been validated, and it appears that the flexibility of the proposed modeling approach is limited.

In this paper, we demonstrate the concept and utility of a network analyzer in a high-fidelity HIL design environment. In addition, we present a frequency domain analysis of a HIL system by means of comparisons with small-signal analytical models and a hardware reference converter. The HIL emulator platform that is used to implement the frequency-domain analysis is based on a novel application specific processor in FPGA that computes piecewise linear state space models in hard real-time with a 1 µs time step [4]. We validate the ultra-low latency HIL emulator in both the time-domain (large signal) and in the frequency domain (small signal analysis). We experimentally validate several frequency domain transfer functions for a buck converter, including input impedance, output impedance, line-to-output, and control-to-output transfer functions using a HIL-based spectrum analyzer. To the best of our knowledge, extensive frequency domain fidelity validation has not been reported for any ultra-low latency HIL emulators. In addition, we believe that the “network analyzer” utility in HIL can be an extremely useful tool for control engineers to quickly tune and optimize performance of a closed loop control system.

This paper has been organized as follows. Section II will develop the approach used by the ultra-low latency HIL emulator to emulate power electronics systems, and the piecewise linear model used to emulate a buck converter. Section III develops the implementation of the “network analyzer” for frequency response analysis in the HIL system. Section IV describes the hardware reference design used to
experimentially validate the HIL emulator. Section V presents comparisons between hardware reference design and the HIL based emulation. Section VI concludes the paper.

II. REAL-TIME HARDWARE-IN-THE-LOOP IMPLEMENTATION

The real-time HIL emulator uses a hybrid system approach for modeling power electronics systems. It combines the flexibility and scalability of processor based simulations with the ultra-low latency performance of modern FPGAs. The architecture of the HIL platform consists of a schematic editor used to describe the circuit to be simulated, and a software tool chain comprising a piecewise linear state space equation solver on the back-end to analyze and compile the circuit. The circuit is compiled offline into a piecewise linear state space representation with a fixed deterministic computation time step that maps all reachable topologies. To achieve ultra-low latencies, the state space representation and the topology selector are directly loaded into the data and program memory of the FPGA before runtime.

For this paper, we have chosen to model the buck converter. This is because the buck converter is a simple, well understood example of a switching converter in both the time and frequency domain. Because the HIL emulator uses the switching model of the converter, unlike the time averaged model, validation of the fidelity of this system in the time and frequency domain can be easily extended to more complex switching converter systems.

III. FREQUENCY DOMAIN ANALYSIS IN HIL

In order to validate the fidelity of the HIL emulator in the frequency domain, we have created a test setup where we measure the following frequency domain transfer functions: input impedance, output impedance, line-to-output, and control-to-output. We will briefly describe the test setup used to obtain the frequency domain measurements, the four experiments used to measure the transfer functions, and the analytical models used to compare the results.

We have chosen the converter under test to be the buck converter, due to its analytical small-signal model being very well understood. However, the HIL system can be easily expanded to model more complex systems such as a two-level inverter.

A. Test Setup

The procedure for the tests was as follows: the buck converter was created inside the schematic editor and assigned the values shown in Table 2. For each experiment, an external source is used to inject a disturbance into the HIL emulator. The frequency of the disturbance is swept and the resulting analog waveforms from the emulator are captured on an oscilloscope for post processing. Post processing is carried out using Matlab. A script is used to extract the Discrete Fourier Transform of the swept frequencies using the built-in Goetzel algorithm. The DFT of the output is compared to the disturbance, and plotted against the analytical model in the frequency domain.

The open loop control algorithm for the buck converter is implemented on a separate DSP based Texas Instruments TMS320F2808 Control Card.

The signal generator used to generate the disturbance is an Agilent 33220A and is injected through an interface board between the TMS320F2808 Control Card and the HIL emulator. For the input impedance, output impedance, and line-to-output tests, the disturbance is routed directly to the HIL emulator and is used to drive the simulated disturbance source. For the control-to-output test, the disturbance is routed directly to the internal ADC module of the TMS320F2808, where it is used to perturb the control signal, as seen in Figure 1. The specifications for the HIL computational platform can be seen in Table 1.

a) Input Impedance Experiment

In the first experiment in which we measure the input impedance, we injected a disturbance into the input voltage and measured the effect this disturbance had on the input current. The disturbance is modeled as a small sinusoidal voltage source in series with the DC input voltage. Indeed here we can see a benefit of using the HIL emulator system, in which there is no need for an isolation transformer or impedance matching in order to successfully inject our disturbance into the power electronics system. We compare this to the following analytical model [5]:

\[
Z_{in} = \frac{1}{D^2} \left( \frac{s^2 R LC + sL + R}{sRC + 1} \right)
\]  

(1)

The result of this comparison can be seen in Figure 2.

b) Output Impedance Experiment

In the second experiment, in which we measure the output impedance, we inject a current disturbance into the output load of the system. We measure both the resulting current and voltage across the load to calculate the frequency response and compare it to the following analytical model [5]:

![Figure 1: Model of a buck converter simulated in real-time, as shown interfacing with an open-loop embedded controller to measure the control-to-output transfer function.](image-url)
\[ Z_{out} = \frac{sLR}{s^2RLC + SL + R} \] (2)

The result of this measurement can be seen in Figure 3.

c) Line-to-Output Experiment

In the third experiment, in which we measure the line-to-output transfer function, we use the same setup as the input impedance experiment, but instead of measuring the input current, we measure the resulting output voltage to get the line-to-output transfer function. This result is compared to the following analytical model [5]:

\[ \frac{V_{out}}{V_{in}} = D \left( \frac{1}{1 + \frac{sl}{R} + s^2LC} \right) \] (3)

The result of this measurement can be seen in Figure 4.

d) Control-to-Output Experiment

For our fifth experiment, we inject our disturbance directly into the controller. The ADC module inside the TMS320F2808 requires unipolar signals, so we inject this with an offset. Inside the controller, we remove this offset and appropriately scale it before adding it to the value of the duty cycle inside the controller. An interrupt service routine runs at 8 kHz to read the ADC and update the PWM generator module inside the controller. The PWM module then uses this updated duty cycle value to determine the duty cycle of the PWM used to drive the converter. The corresponding analytical model is as follows [5]:

\[ \frac{\dot{\theta}}{d} = V \left( \frac{1}{1 + \frac{sl}{R} + s^2LC} \right) \] (4)

However, the analytical model for the control-to-output transfer function has to be modified to take into account the delay due to the PWM implementation of the controller. This is unlike the previous frequency response measurements, in which comparisons were made directly to the analytical models described in their respective sections. Thus, the updated control-to-output transfer function is as follows:

\[ \frac{\dot{\theta}}{d} = V \left( \frac{1}{1 + \frac{sl}{R} + s^2LC} \right) e^{-sr} \] (5)

The result of this measurement can be seen in Figure 5.

IV. REFERENCE HARDWARE DESIGN

To comprehensively validate the fidelity of the real time HIL emulation, we created a validation test setup that runs the real-time HIL emulation in parallel with an identical power stage implementation. The test setup can be seen in Figure 1 and the specifications of the test bed can be seen in Table 1 and Table 3.

<table>
<thead>
<tr>
<th>Component</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPGA device</td>
<td>Xilinx Virtex-5 ML506</td>
</tr>
<tr>
<td>Clock Speed</td>
<td>100 MHz</td>
</tr>
<tr>
<td>ADC Sampling Rate</td>
<td>1 MSPS</td>
</tr>
<tr>
<td>DAC Sampling Rate</td>
<td>1 MSPS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Component Value for buck converter HIL setup.</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
</tr>
<tr>
<td>L</td>
</tr>
<tr>
<td>C</td>
</tr>
<tr>
<td>V</td>
</tr>
<tr>
<td>PWM switching frequency</td>
</tr>
<tr>
<td>Dead time</td>
</tr>
</tbody>
</table>

The hardware reference design consists of a custom-designed back-to-back two-level three-phase inverter drive rated at 20A and 600 V. The inverter modules are IRAMY20UP60B Power Hybrid ICs by International Rectifier. The input voltage is driven by an external DC voltage source. The inverter is capable of measuring phase currents, line-to-line voltage, DC voltage, and shunt currents, and has integrated temperature and over current protection. The reference design features contactors on all power outputs, enabling fault injection on any of the phases of either inverter modules. The hardware reference also features a fast CPLD to manage digital signal rerouting, fault injection and fast protection against over current, over and under voltage, and over temperature conditions.

<table>
<thead>
<tr>
<th>Component Value for reference hardware design.</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
</tr>
<tr>
<td>L</td>
</tr>
<tr>
<td>C</td>
</tr>
<tr>
<td>V</td>
</tr>
<tr>
<td>PWM switching frequency</td>
</tr>
<tr>
<td>Inverter module ratings</td>
</tr>
<tr>
<td>Dead time</td>
</tr>
</tbody>
</table>

V. FIDELITY COMPARISON: HIL VS. REFERENCE HARDWARE DESIGN

In this section, we investigate the fidelity of the real-time HIL emulator by comparing the results of the large-signal time domain and the small signal frequency domain of the HIL emulator with the real hardware setup shown in Figure 6. In each of the following experiments, we run both systems in parallel with the same PWM signals from the controller.
1) Time Domain
In the first experiment, we compare the steady state output voltage ripple of the HIL emulator with the real hardware reference. Figure 7 shows the comparison between the emulator and the real hardware reference on the time scale of the switching frequency.

In the second experiment, we compare the steady state current ripple through the inductor of the HIL emulator with that of the real hardware reference. Figure 8 shows the comparison between the two results.

As the figures show, the emulation of the buck converter matches very closely with the real hardware reference on the time scale of the switching frequency.

2) Frequency Domain
In the third experiment, we redo the small signal frequency domain experiment to measure the control-to-output transfer function, but this time we compare the results to the real hardware reference. Figure 9 shows the comparison in the frequency domain. As the figure shows, the emulation of the buck converter closely matches the frequency response of the real hardware converter.

VI. CONCLUSION

In this paper, we have presented results that compare the small signal frequency domain behavior of a HIL emulator with its theoretical analytical model. To further validate the fidelity of the emulator, we have compared the emulator to a reference hardware design, both in time domain (large signal) and the frequency domain (small signal AC analysis) for the control-to-output transfer function. All results show good agreement between the real-time emulation, analytical models, and the reference hardware design. In addition, we briefly demonstrated the utility of a “network analyzer” concept in the HIL real-time emulator for rapid control design tuning and optimization.
Figure 6: Photograph of the experimental buck converter HIL emulation validation setup.

Figure 7: Comparison of steady state output voltage of the HIL emulator and the reference hardware design.

Figure 8: Comparison of steady state inductor current of the HIL emulator and the reference hardware design.

Figure 9: Comparison of control-to-output measurements between the HIL emulator and the reference hardware design.

A key advantage of this system is the combination of its flexibility, high-fidelity, and robustness due to the hybrid modeling approach employed by the emulator. High-fidelity simulation, with a computation time-step of 1 µs, lends itself to frequency domain analysis and control design and optimization. By validating the fidelity of the HIL emulator in both time and frequency domains, we show the HIL emulator to be an invaluable verification, design, and optimization tool for power electronics control engineers.

REFERENCES