Applying Computer Modeling to Post-Silicon Electrical Validation

Jennifer L. Vaccaro, Intel Corporation and Olin College of Engineering

April 16, 2017

Abstract

Because the post-silicon validation process requires a high amount of effort from engineers, we propose the use of a variety of modeling techniques to tune the silicon. Black-box models interpolate measured physical data (Kriging), while white-box models generate simplified simulation data with some correlation to the physical data. We demonstrate that these two modeling techniques, as well as a combined simulation-physical data model (co-Kriging), can be used to tune registers for the I2C interface. The use of our models decreases the effort necessary to successfully tune the silicon. More work is needed to apply our approach to high-speed interfaces.

1 Problem and Motivation

Post-silicon validation consumes a significant portion of semiconductor product development resources and time. One of its aspects, electrical validation (EV), aims at ensuring reliably correct operation in a full system environment. Because parts are subject to manufacturing variability and must operate in a variety of systems under different conditions, a major portion of post-silicon validation efforts focuses on tuning the part to ensure it meets specifications across the broad operational space. This system is not yet highly automated, and instead requires many measurements on a population of parts and systems, which is costly and time consuming. This work describes an approach to accelerate post-silicon validation by streamlining the tuning process using a combination of circuit models and statistical data gathering.

2 Background and Related Work

White-box models rely on a physical representation of the circuits, but need complex calibration in order to match the actual system in silicon. A popular framework to implement such models is Simulink [5] which allows a good trade-off between model details and level of abstraction. However, with advanced process nodes, circuit non-idealities make these models difficult to create and calibrate accurately.

Conversely, Design of Experiments (DOE) and associated Response Surface Models (RSM) have long been used for building black-box system representations [1]. These typically involve factorial-type designs and low-order polynomial models. However, highly nonlinear systems with many parameters and noisy measurements require a lot of data and complex models to achieve the desired accuracy[4]. These measurements are expensive to obtain. One particular RSM technique is Kriging [3]. It offers the ability to model very nonlinear surfaces, can tolerate some measurement errors, and lends itself to incorporating prior knowledge in the form of physical models using a technique called co-Kriging [3].

3 Approach and Uniqueness

Our goal is to demonstrate the use of computer models to efficiently discover the optimal recipe, i.e. the set register values that enable reliable operation of all parts across the operational space. We selected the Inter-IC (I2C) interface [2] as a benchmark, due to its ubiquity and criticality in enabling the validation of other interfaces. The I2C protocol is a low-speed bus which includes both a clock and data line. In order to ensure that the data is sampled correctly, the timing between the lines must meet specifications under a variety of environmental conditions and processing results. In this case, the recipe consists of a combination of the pull up resistor (R_{term}), the clock duty cycle (HCNT), and the delay between data and clock (SDA-delay).

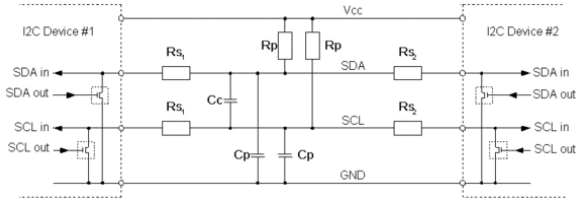

Figure 1: A simplified I2C circuit diagram, on which the physical model was based.

3.1 Design of Experiment

Our DOE includes a combination of recipe parameters, channel lengths, and temperature. This DOE contains two subsets: the larger was used to train our models (training DOE), and the smaller was used to confirm their validity (testing DOE). The training DOE uses Plackett-Burman, Axial, and Latin Hypercube Sampling designs for a total of 50 trials at each interface speed. The testing DOE uses Latin Hypercube design for a total of 20 trials at each speed. The two DOE subsets do not intersect, so we do not train and validate from the same recipe. We measure timing data that is most likely to fail specifications: high and low clock period, and the setup

and hold time. We define these timing measurements in accordance with Intel's I2C specifications.

3.2 Generating Models

Using the training data, we create and tune several models. We then use test data to compare their performance. With the exception of the physical model, we must create a new model for each speed.

The physical model was created with Simulink. Because it inherently models the physical interactions, it was not trained on the data. However, several model parameters were tuned using the training data to ensure a better fit. The physical model block diagram (Fig. 2) includes in both the Data and Clock lines an nMOS transistor, pull-up resistor, series resistor, and capacitive load to ground.

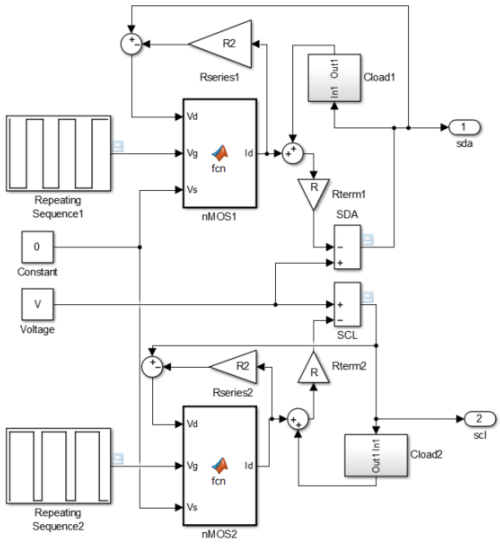

Figure 2: The physical model block diagram, generated in Simulink.

The Kriging model was created directly from the training DOE and measurement data. Because of its interpolative nature, it fits measured values exactly, and predicts the most statistically likely timing measurements for unmeasured values.

The co-Kriging model was generated using both the physical model and a subset of the I2C measured data. In this approach, the physical model provides the a priori knowledge of the circuit, while the black-box portion of model accounts for any systematic error between the physical model predictions and the measured data.

3.3 Recipe Tuning Optimization

In order to select the optimal recipe, we prioritized the measurements based on customer need. For example, high and low clock period failures rarely limit I2C performance. However, setup and hold time are often not met. Because specifications are defined for each speed separately, we created a different recipe for each speed.

After calculating each model's prediction for the full suite of recipes and conditions, we selected the optimal recipe for each speed. First, we removed recipes that did not meet the basic specifications on every channel and at every temperature. Then we selected the recipe which maximized the product of setup and hold time, because they were the highest priority, often fail to meet specifications, and are by definition negatively correlated.

4 Results

In order to demonstrate how this approach can simplify EV work, we tested the correlation of our models to silicon at two training data speeds (100 kHz and 1MHz) and at a different speed than the training data (400kHz). We then used the successful models to find the optimal recipe. Even though all 3 modeling techniques proved successful when tuned to a specific data set, only two of our models correlated successfully at a new link speed (400kHz). On each plot, (Fig. 3-5) the x-axis represents silicon data and the y axis represents the model prediction. The solid black lines represent the specs for the measurement, and the red dashed line is the ideal 1:1 ratio between silicon and model prediction.

4.1 Model prediction at 100kHz and 1MHz

The Kriging model was able to predict the measurements most accurately at both 100kHz and 1MHz. The results are shown in Fig. 3 and Fig. 4, and the Kriging model follows the ideal ratio better than the physical model, especially for the low clock period and the hold time.

4.2 Model prediction at 400kHz

In order to determine the relative success of each modeling technique, we compared their predictions for I2C performance at an untrained speed. The Kriging model alone was not accurate at 400kHz. Observing the correlation plots in Fig. 5, we are able to see a clear linear trend between the co-Kriging model and the physical model, respectively, to the data.

4.3 Optimizing Recipes

We were able to find an optimal recipe at 100kHz, 400kHz, and 1MHz and demonstrate specification compliance with acceptable margins. In addition, the Kriging (at 100kHz and 1MHz) and co-Kriging model calculate the responses for the full recipe suite quasi-instantaneously, while the physical model runs at approximately 400 ms per trial. The latter runtime would increase dramatically for more complex interfaces, particularly with training and adaptive equalization.

4.4 Conclusions

This work demonstrates that correlated models are useful for EV work and can reduce the effort required for spec compliance, margin predictions, and recipe tuning.

Because I2C is a simple and slow protocol, once tuned, the physical model was able to accurately predict the timing measurements at each speed. However, physical models become considerably more difficult to implement and tune for

higher speed channels and with more complex circuitry. Furthermore, due to the increasingly large number of interfaces per SOC it may not be practical to rely solely on physical models, due to the prohibitive cost of writing, maintaining and tuning them. This work has demonstrated the ability to use approximate physical models with co-Kriging to achieve the accuracy needed for EV work without the overhead associated with either developing an accurate physical model or the large number of measurements needed for accurately fitting a black box model.

While more work is needed to evaluate the proposed approach for high-speed interfaces with adaptive equalization, we have shown that it can form the basis for validating slower links with high confidence and reduced overhead.

[2]

5 Acknowledgments

This project was conducted under advisement of Dr. Nagib Hakim at Intel Corporation. He proposed that modeling techniques could be used to speed up post-silicon validation, and was able to connect this work to the wider context of the tuning process. This work would not have been possible without his guidance and mentorship.

Additional thanks go to Candace Zeng, Govardhani Edpuganti, and Ryan Brim for their technical expertise in the I2C interface, and their generous support collecting data in the lab.

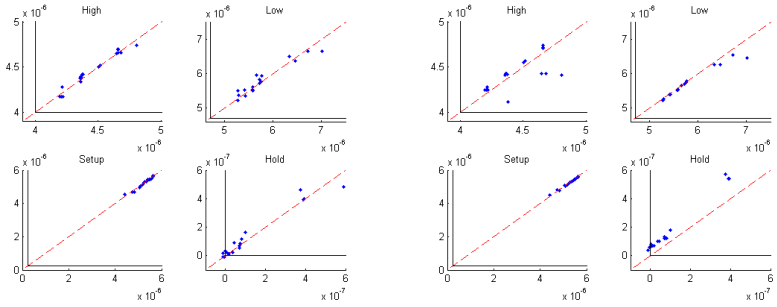

Figure 3: Correlation of a. the Kriging model and b. the physical model to silicon at 100kHz. On each plot, the x-axis represents silicon data and the y axis represents the model prediction. The solid black lines represent the specs for the measurement, and the red dashed line is the ideal 1:1 ratio between silicon and model prediction.

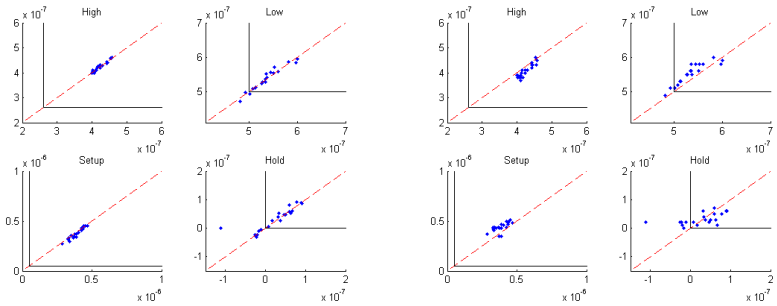

Figure 4: Correlation of a. the Kriging model and b. the physical model to silicon at 1MHz.

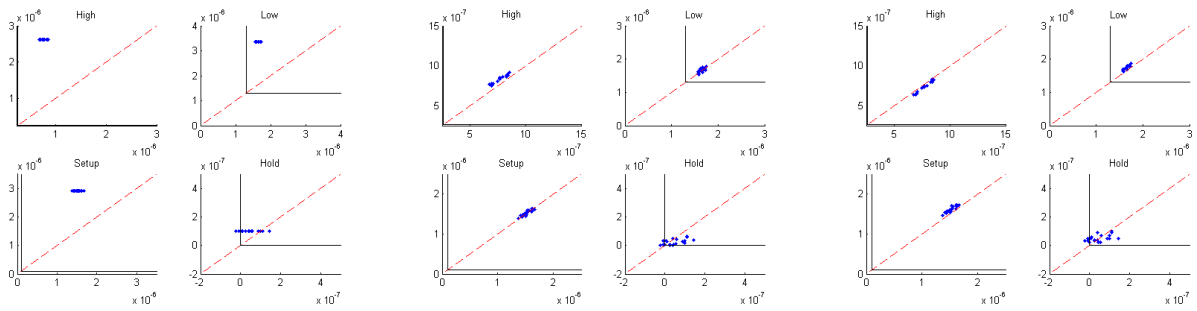

Figure 5: Correlation of a. the Kriging model, b. the co-Kriging model, and c. the physical model to silicon at 400kHz.

References

[1] J. Sacks, W. J. Welch, T. J. Mitchell, and H.P. Wynn, *Design and analysis of computer experiments.* Statistical science, 4(4):409-435, 1989.

[2] NXP: telos Tracii *I2C Primer* Source: http://www.i2c-bus.org/i2c-primer/ Accessed: Aug. 12 2016.

[3] I. Couckuyt, T. Dhaene, P. Demeester *ooDACE toolbox: a flexible object-oriented Kriging implementation.* J Mach Learn Res 15:3183-3186, 2014.

[4] R. Gray, *Post-Silicon Validation Experience: History, Trends, and Challenges* GSRC Wokshop on Post-Si Validation, Anaheim, June 9, 2008

[5] Mathworks. *MatLab & Simulink: Simulink Reference R2013b (PDF)* Retrieved 28 Sep. 2015.