Raven: A Tool for Automatic Generation of Analog Behavioral Models from Schematics

Chandramouli V. Kashyap and Chirayu S. Amin
Intel Corp., Hillsboro, OR

Abstract
Analog content in SOCs is increasing every generation as these systems due to increasing support for consumer electronics and increasing IO and graphics bandwidth. Since these circuits are part of a larger digital system, the analog-digital interactions must be verified, typically via suitable RTL level abstractions of the analog blocks. Due to the lack of any formal equivalence checking of analog circuits with the higher level RTL models, mixed-signal validation (MSV) is used extensively to verify correctness of the digital and analog sub-systems working together.

Figure 1. A comparison of all digital validation flow with a mixed-signal validation flow. The analog models in an MSV flow typically come in three flavors: spice-level schematic netlists, Verilog-AMS (analog mixed-signal), and Verilog with real numbers (abbreviated as Verilog-Real in rest of this abstract) – with the latter two referred to as behavioral models or BMODs in short. The schematic models are the slowest to simulate but are most accurate and the Verilog-Real models are the fastest to simulate with some loss in accuracy. At least an order of magnitude speedup across each level is typical as we move from schematic to Verilog-AMS to Verilog-Real models.

To alleviate these problems, we have created a tool called Raven that automates BMOD creation from schematics.
have plans to extend it to output Verilog-AMS if necessary. Raven follows the following steps to generate the BMOD of a given circuit:

1. Based on the input specification, sample the input space and run circuit simulations for all the samples.
2. Automatically extract important output parameters from each of the simulation: period, pulse width, phase, steady state dc value, delays, etc.
3. Build a piecewise linear (PWL) look-up table based functional model of the output parameters as a function of input parameters.
4. Encode the PWL look-up table (i.e. functionality) in Verilog format.

Figure 2. Automated, meaningful behavioral model generation using Raven

The input specification file captures the designer intent for the circuit. Its three main characteristics are input specification, output specification, and analysis environment specification. The input specification contains names of the input pins, their types - LOGIC or ANALOG, current or voltage, as well as the range of values. The output specification also contains names of output pins which are desired as interface nodes in the generated Verilog BMOD and their type, and whether current or voltage. The analysis specification describes paths to work area, settings for circuit simulation, and other CAD environment items.

Based on the specification, Raven samples the input signals and creates a circuit simulation deck for each of the sample. Raven then simulates all the samples using a circuit simulator and captures the waveforms at all the outputs of interest. Raven makes extensive use of distributed computing resources to spawn multiple simulations of the samples in parallel to reduce BMOD generation runtime. Based on the observed waveforms, appropriate parameters such as final value, delay, pulse width, period etc. are extracted. Since the temperature, voltage and process skew can be varied as part of simulations, Raven generated models can be made PVT aware.

Raven then builds a function of the extracted output parameters as a piecewise-linear function of the inputs based on the sampled space. This is a key step that allows Raven to capture the core functionality of the circuit using the appropriate functional relationship that makes sense. This is done automatically based on the the output parameters that are changing as input parameters change. Finally, Raven codes both the look-up table and the linear interpolation method in Verilog. Raven can generate both SystemVerilog with Reals or Verilog-AMS. While it may seem that a large number of samples are necessary when the circuit has several operating modes with many changing inputs. In reality the different operating modes are usually mutually exclusive and hence a full factorial sampling of the input space is unnecessary. Moreover, since Raven uses interpolation to compute output values from input values not stored in the table, this allows the user to trade-off the number of samples (runtime) with accuracy. Often, for functional validation, sufficient accuracy can be achieved with only a few sample points. Finally, we note that Raven generated Verilog code has built-in assertions that check to see if the input values are within range during BMOD simulation in MSV flows allowing the validators to check if the circuit does not meet the specs.

We have used Raven to successfully generate models for phase-interpolator, a VCO, a thermal sensor, a signal-delta ADC and a transmitter with pre-emphasis. Some of these models have been used in industrial MSV flows resulting in an order of magnitude improvement in runtime and validation productivity.