DesignCon 2013

Using Power Aware IBIS v5.0 Behavioral IO Models to Simulate Simultaneous Switching Noise

Romi Mayder, Xilinx, Inc.
RomiM@Xilinx.com (408) 879-6083

Chris Wyland, Xilinx, Inc.
ChrisW@Xilinx.com (408) 879-2774

Bradley Brim, Cadence, Inc.
BradB@Cadence.com (360) 446-0298

Yingxin Sun, Cadence, Inc.
Sunyx@Cadence.com (408) 944-7237

Abstract

Typically simultaneous switching noise (SSN) transient simulations require significant CPU and RAM resources. A prominent factor affecting both CPU and RAM resource requirements is the number of MOSFET models included in the post layout extracted IO netlists. By replacing the IO netlists with power aware IBIS v5.0 behavioral models, both the CPU and RAM resource requirements are dramatically reduced. A comparison of several SSN transient simulations whereby the aggressor frequency is sweep across a wide frequency range is shown. The resultant victim waveforms will clearly demonstrate that each SSN transient simulation using post layout extracted IO netlists requires days to run compared to just mere minutes using power aware IBIS v5.0 behavioral models. Most notably, there is no significant loss in accuracy. In fact, in many cases, there is an increase in accuracy due to convergence issues associated with post layout extracted IO netlists. The power aware IBIS v5.0 behavioral models offer both dramatically faster transient simulation times and lower memory requirements. Improvements to these two key metrics without sacrificing accuracy, allows for more aggressive and accurate signal and power integrity analysis than has previously been possible.

Author(s) Biography

Romi Mayder is a senior staff signal and power integrity engineer at Xilinx, Inc. Prior to joining Xilinx, Mr. Mayder worked as a consultant specializing in silicon die level signal and power integrity. He also consulted in the field of design and fabrication of advanced package technologies, including stacked silicon interconnect. Mr. Mayder has been employed by two companies in the Test and Measurement industry, Agilent Technologies and Anritsu (Wiltron) Company, where he specialized in microwave and millimeter wave microelectronic circuit design and fabrication. Mr. Mayder has over 10 years experience in semiconductor process technologies including photolithography, ion implantation, plasma enhanced chemical vapor deposition, dielectric sputtering, and chemical mechanical polishing of silicon wafers. Mr. Mayder received his Bachelor of Science degree in Electrical Engineering and Computer Science from the University of California at Berkeley in 1992. Mr. Mayder has published 25 patent applications in the fields of signal and power integrity as well as semiconductor process technologies.

Chris Wyland is a Senior Staff Signal Integrity Engineer in the System Signal Integrity Group at Xilinx, Inc. He has over 20 years experience in package electrical modeling and signal integrity, working at such companies as Xilinx, Philips Semiconductors, and Integrated Device Technology. He has published over ten papers in the area of package signal integrity and holds over 20 patents.

Bradley Brim is Senior Staff Product Engineer in the Silicon/Package/Board division for Cadence Design Systems, Inc. His present responsibilities focus on analysis and design flow automation for power integrity and signal integrity for boards, packages and chip/package/board systems. Throughout his career he has focused on electromagnetic and circuit modeling of components,

circuits and systems for SI, PI, EMC, RF, microwave and antennas. He has worked in the EDA industry for over 20 years; previously with Sigrity, Ansoft and Agilent. Brad has experience in various roles for software development, applications engineering and product marketing.

Yingxin Sun is a Senior Architect in Sigrity R&D US Group at Cadence Design System, Inc. Dr. Sun has over 12 years experience in EDA tools development for signal integrity including chip, package and board model extractions and simulations. He mainly worked on power-aware buffer modeling extraction in recent two years. He received his Ph. D from Tsinghua University, Beijing, China in Electromagnetic Theory and Microwave Technology in 1996

Introduction: Simultaneous Switching Noise

To ensure reliable performance of today's high speed electrical systems, signal and power integrity transient simulations have become a necessity for both pre-layout and post-layout electrical designs. Today's electrical systems operate at much high data rates than in previous generations. The electrical wavelengths associated with the frequencies of today's digital signal rise/fall times have become shorter than the electrical interconnects. Once the electrical wavelength becomes shorter than the electrical interconnects, high speed digital effects are created. These high speed digital effects create voltage noise and current demands at different physical locations within the electrical system. Understanding these high speed digital effects and designing the system appropriately is a necessity to assure reliable electrical performance.

Simultaneous switching noise (SSN) is a major component in this signal and power integrity analysis. Simultaneous switching noise is caused by a number of signals transitioning simultaneously. This causes an instantaneous current demand on the power distribution network as well as the signal's output pad. The most noticeable effect of this is typically voltage noise on the power supply at the die which can slow down or even negate function of the switching outputs. Due to this effect, it is critically important to understand the performance of the power distribution network and its effect on the signaling performance of the electrical system.

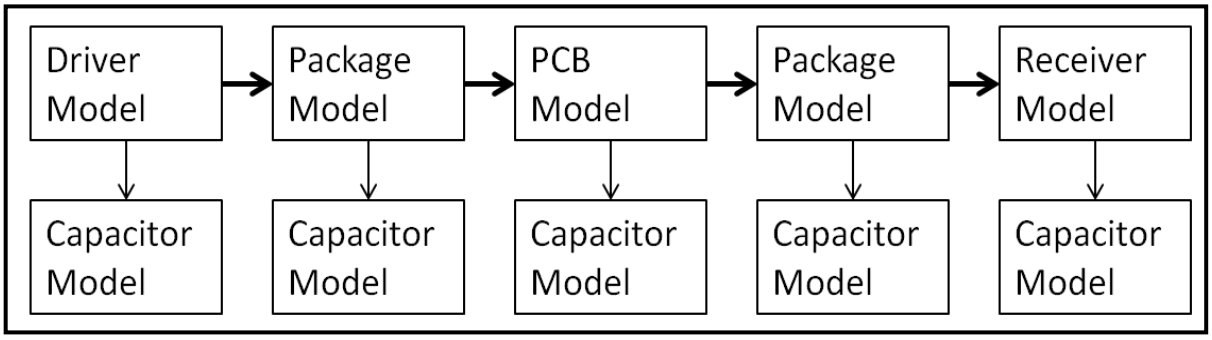
To simulate the effects of SSN at the system level, it would be convenient if we could simply simulate a subset of signals switching simultaneously powered by a real world power distribution network. Then, linearly scale the transient simulation results for a higher signal count electrical system. Unfortunately, to minimize cost, many electrical systems are extremely compact in nature and therefore, voltage noise and current demands do not scale well. The signal and power interconnects are interwoven too tightly that local coupling effects become significant. Thus, SSN simulations need to be performed with the true number of signals switching simultaneously to assess the system's electrical performance.

The SSN simulation schematic should include high bandwidth package and printed circuit board s-parameter models inclusive of not only the signal nets but also the power distribution network. Additionally, a model of the die level circuit should be included in the schematic. When all these

elements are combined in a SPICE netlist, this level of detail can lead to many simulation challenges including but not limited to convergence issues. By utilizing a behavioral model of the die level circuit inclusive of the power distribution network, such as power aware IBIS v 5.0 behavioral models, many of these challenges can be significantly mitigated.

SSN Simulations Using Post-Layout Extracted IO Netlists

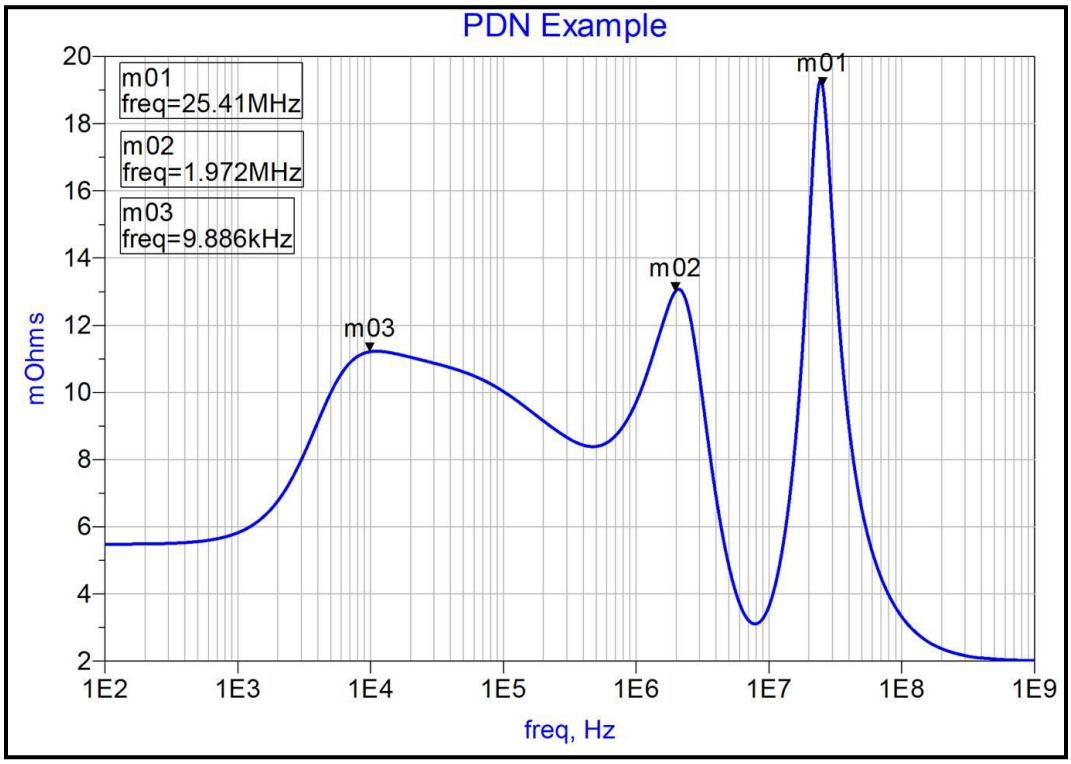
It is desirable to run SSN simulations under the worst case operating conditions of the electrical system. With regard to SSN, these worst case conditions are not always simple to determine. First and foremost, we must accurately determine the power distribution network (PDN) resonance and anti-resonance frequencies. These frequencies are typically determined by simulating the frequency domain self impedance profile of the PDN interconnects and associated capacitances over a wide frequency range. Often times, toggling signals near one of the anti-resonance frequencies will result in the largest magnitude of voltage noise. It is important to consider that signaling circuits have their own parasitic capacitances attached to the power supply which will change the PDN system resonant frequencies. It is important to consider the current demand of the circuits themselves and how they interact with the signal interconnects. These signal interconnects become part of the system and their impedances and cross coupling will affect the resonant frequencies of the system by affecting the current demand of the signaling circuits. Lastly, the worst case operating condition is dependent upon the signaling history. Voltage noise is the superposition of pre-existing noise waveforms with new noise waveforms caused by signal current demands of the system. For slower signaling speeds, the induced noise settles quickly and there is very little superposition of the voltage noise waveforms. However for signals operating near the system resonant frequencies, the voltage noise waveforms don't settle quickly and therefore the noise created by new signaling superimposes rapidly. Thus, it is important to sweep the signaling frequencies to identify the system worst case conditions. The transient simulation duration must be sufficient to ensure that the superposition of voltage noise has reached a maximum steady state value. Figure 1 shows a representative example circuit diagram for an SSN simulation.



Representative SSN Block Diagram per IO

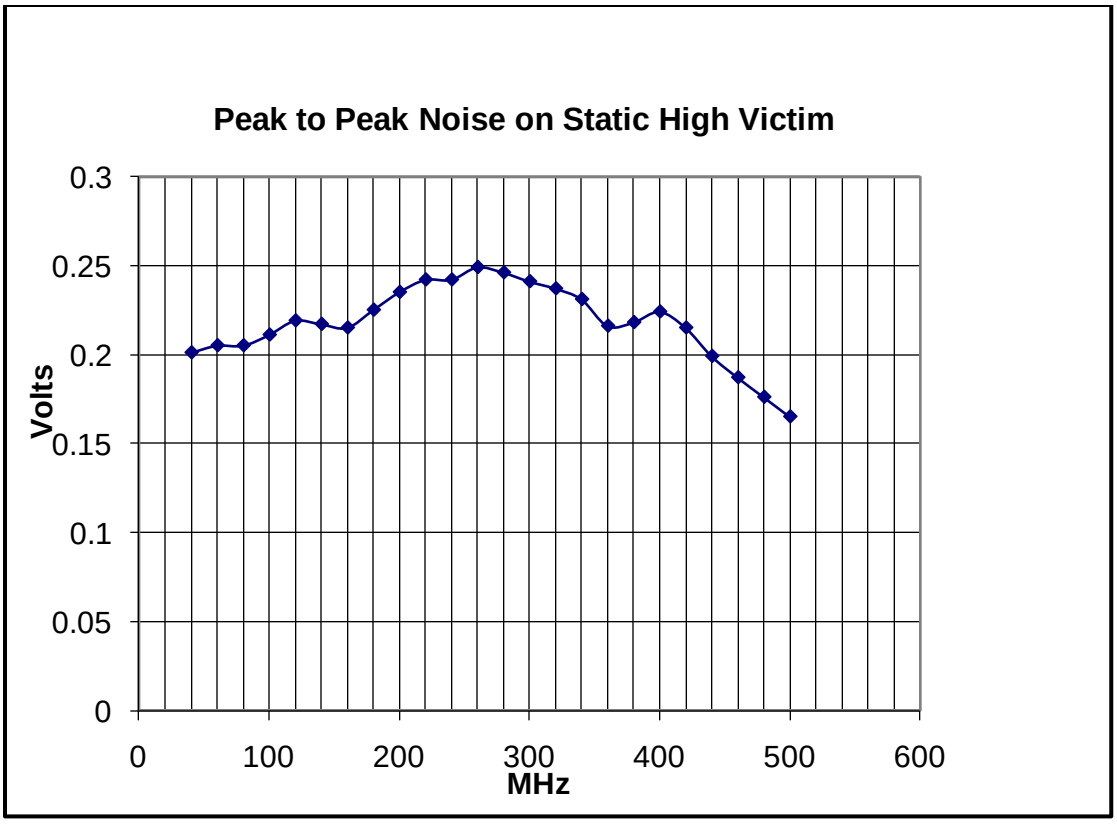
Figure1

Figure 2 shows a representative frequency domain self-impedance profile of a power distribution network. Marker M01 represents the anti-resonance frequency of the silicon die capacitors and package inductance. Marker M02 represents the anti-resonance frequency of the package capacitors and PCB inductance. Marker M03 represents the anti-resonance frequency of the PCB capacitors and the VRM choke coil. This example is related to a power distribution network of a Xilinx XC7K325T-FFG900 device/package combination.



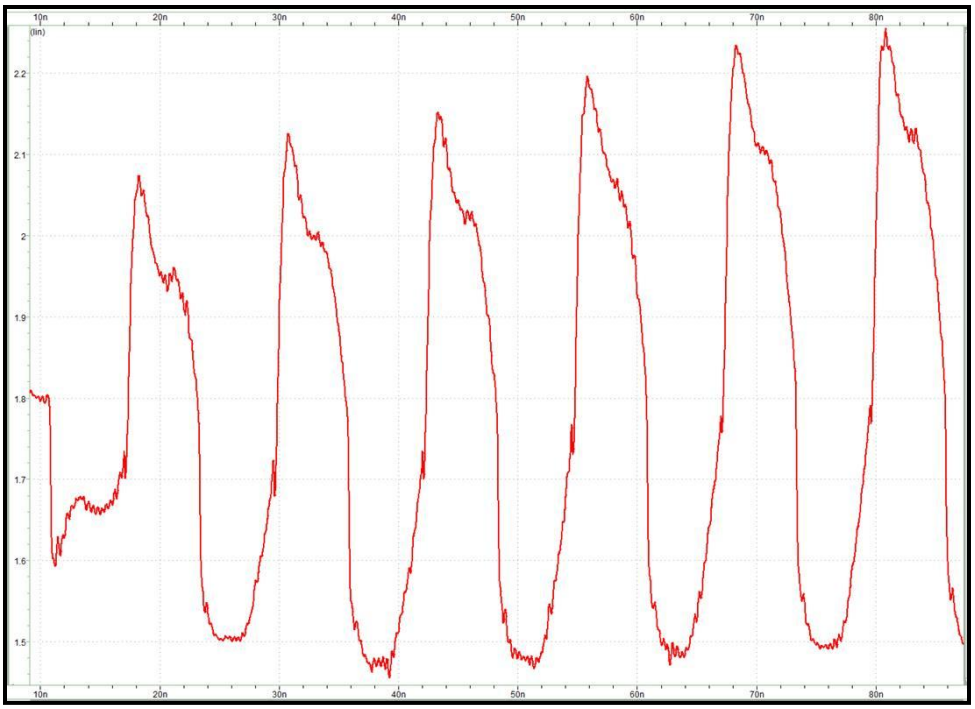
A representative frequency domain self-impedance
Profile of a Power Distribution Network of a Xilinx FPGA
Figure 2

Figure 3 represents the peak-peak voltage noise on a static high signal with 49 aggressors running a clocking pattern for 10 consecutive cycles. As can be seen, the worse case noise occurs at roughly 260 MHz. This is the PDN anti-resonant frequency of the silicon die capacitance and package inductance. The data shown was simulated using the Xilinx XC7K325T-FFG900 device using HSTL_I_18_F_HP IO standard. The 49 aggressor signals are terminated with 50 ohms to VTT on the printed circuit board while the victim is terminated with 50 ohms to VCCO.



SSN Voltage Noise
on Static High Victim vs Frequency (49 Aggressors)
Figure 3

Figure 4 shows how the voltage noise on VCCO and the static high victim signal buildups with consecutive pulses of the aggressors. Typically, Xilinx uses a minimum of 10 consecutive aggressor pulses at each frequency to ensure that the voltage noise has reached maximum amplitude. The data shown corresponds to 49 aggressor signals switching at the same frequency. The SSN SPICE netlist includes S-parameter models for the package substrate, package capacitors, PCB substrate, and PCB capacitors.



Die Level Voltage Noise Buildup
on Power Distribution Network
Figure 4

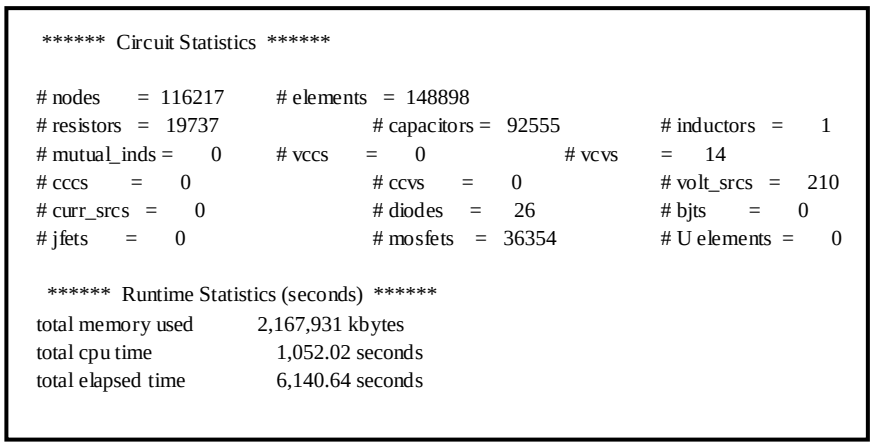
Due to the many factors which determine the worst case operating condition for SSN, it is desirable to use the most accurate models available for SSN simulation. These models need to adequately represent performance across the bandwidth of the system. For high frequency systems, this usually means S parameter models of the package, capacitors and printed circuit board (PCB). It also usually means a fully extracted signal driver SPICE model. It is also important to use a worst case signaling pattern to simulate the noise build up.

The need for S-parameter models of the package and PCB interconnects stems from the bandwidth requirement of the signals. If the aggressor signals have relatively slow rise and fall times such that the highest frequency component wavelengths are greater than 10 times the electrical length of the interconnects, then RLC models will suffice. If the interconnects are constant impedance and spaced relatively far apart to minimize crosstalk, then maybe transmission lines (T elements or W element) models can be utilized. Unfortunately, it is quite common to have very high speed signaling in systems with long traces crowded close together weaving through a variety of layers. This requires the need for a more accurate model which can represent the performance across the frequencies used in the system.

Some of these S-parameter models can be quite complex with varying reference impedances. For signal interconnects, 50 Ohm reference impedance S parameters work well since the characteristic impedance is relatively close to 50 Ohms. Power supply nets typically have lower normalized

impedances, such as 0.1 Ohms due to their characteristic impedance being similarly low. Thus, in some package or PCB S-parameter models, there can be multiple interconnects with varying reference impedances to aid in the circuit simulation accuracy. Touchstone V2.0 is an S-parameter file format which supports multiple reference impedances and is, thus, very useful for SI/PI S-parameter models.

For best accuracy, it is desirable to use post layout extracted IO netlists in the SSN simulation. For larger circuit designs, this can mean including tens of thousands of transistors and capacitors per signal. If the system contains many signals, this overhead can lead to very long simulation times and/or convergence issues. For the XC7K325T-FF900, the number of elements used to simulate one pair of IO standards is shown in Figure 5.



****** Circuit Statistics ******

# nodes = 116217	# elements = 148898	
# resistors = 19737	# capacitors = 92555	# inductors = 1
# mutual_inds = 0	# vccs = 0	# vcvs = 14
# cccs = 0	# ccvs = 0	# volt_srcs = 210
# curr_srcs = 0	# diodes = 26	# bjts = 0
# jfets = 0	# mosfets = 36354	# U elements = 0

****** Runtime Statistics (seconds) ******

total memory used	2,167,931 kbytes
total cpu time	1,052.02 seconds
total elapsed time	6,140.64 seconds

Figure 5

Due to the 36,354 MOSFETs included in the SSN simulation per IO pair, the transient simulation time and memory requirements may be large. In order to have more reasonable simulation resource requirements, it would be advantageous to use a power aware behavioral IO driver model. The IBIS v5.0 model meets this desire while resulting in much more reasonable simulation resource requirements while maintaining the same or better accuracy.

Power Aware IBISv5.0 Behavioral Models

The IBIS open forum committee ratified IBIS v5.0 in August 2008. There are two BIRDs related to the power awareness of the IBIS v5.0 models. Additionally, there is one BIRD ratified as part of IBIS v4.2 which is equally important to achieving accurate SSN simulations.

The first power aware BIRD is 95.6. The title of BIRD 95.6 is “Power Integrity Analysis using IBIS”. This BIRD introduces the keyword [Composite Current]. The [Composite Current] keyword describes the waveform shape of the rising and falling edge currents originating from the power supply terminal of the buffer. The data contained in the *.ibs file consists of a table of power supply current vs. time (I-T). These (I-T) tables use the same test fixture load conditions as the (V-T) data associated with the keywords [rising waveform] and [falling waveform].

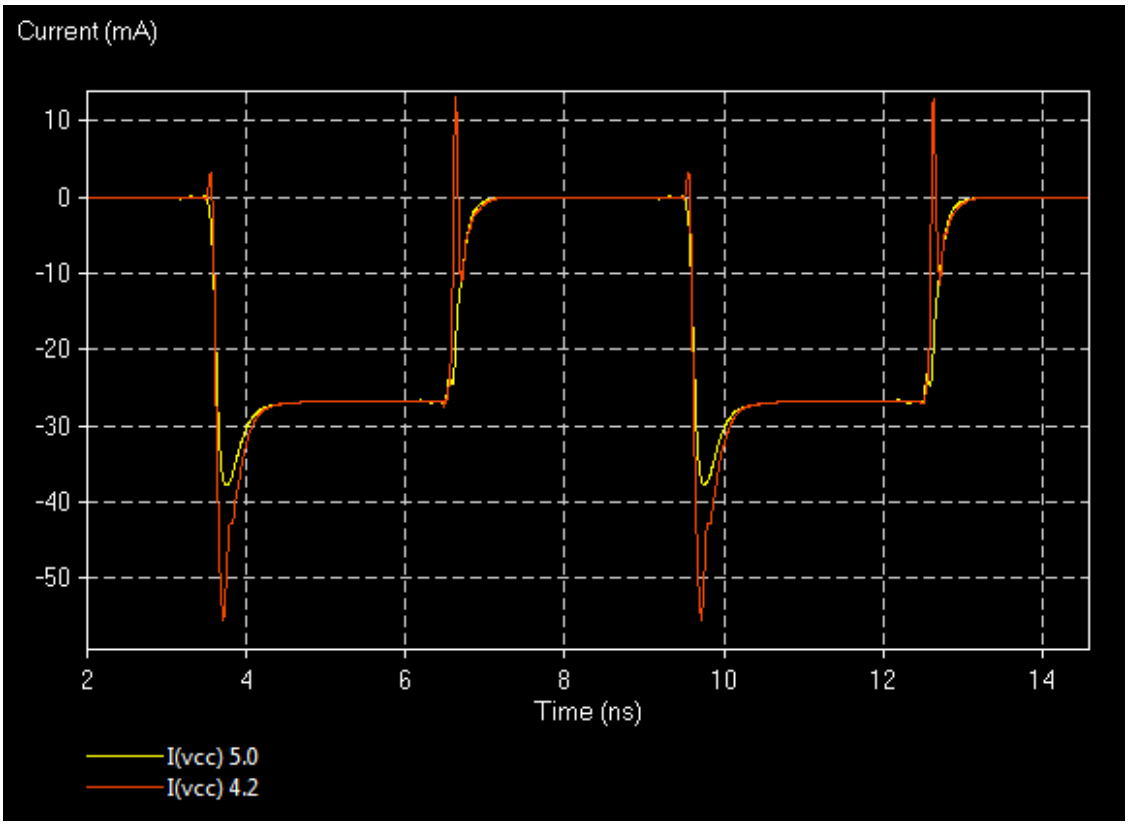
Additionally, the time stamps associated with the (I-T) data must identical to the time stamps of the (V-T) data. The second power aware BIRD is 98.3. The title of BIRD 98.3 is "Gate Modulation Effect". BIRD 98.3 adds to the *.ibs file two new keywords [ISSO PU] and [ISSO PD]. These keywords provide tables of the effective saturation current with respect to the voltage variation of the reference supplies.

Lastly, BIRD 76.1 was ratified as part of IBIS v4.2. The title of BIRD 76.1 is "Additional Information Related to C_comp Refinements". IBIS models can have up to four I-V tables, each of which can have their own supply connection (node). BIRD 76.1 seeks to allow a unique C_comp value for each of these four I-V curves. These four C_comp values allow the splitting of the total pad capacitance between each of these supply nodes. This can be important for power integrity simulations to couple the signal on the IO pad to the appropriate supply node. C_comp_pullup is in parallel with the PU I-V curve. C_comp_power_clamp is in parallel with the PC I-V curve. C_comp_pulldown is in parallel with the PD I-V curve. C_comp_gnd_clamp is in parallel with the GC I-V curve. The individual 4 I-V curves can be separately turned on/off by switching the buffer to low/high/tri-state. However, the C_comp_* are always on.

Creating IBIS v5.0 Behavioral Models using EDA Tools

The IBIS v5.0 models used for the SSN simulations in this paper were created using Cadence's T2B EDA Tool. To generate the typical/minimum/maximum corner data usually requires more than 50 HSPICE simulations for a typical IO model. When using T2B, all the netlist creating, data collection, and post processing are automatically done. Additionally, the I-V, V-T and I-T tables are automatically extracted in T2B.

One key different between IBIS v4.2 and IBIS v5.0 models is the accuracy of the power supply current. In IBIS v4.2 models, the VCCO currents cannot be correctly simulated due to the lack of (1) the pre-driver current, (2) on-die decap bypass current and (3) the crow-bar. The different power supply current profiles of IBIS v5.0 and IBIS v4.2 are shown in Figure 6.



Difference in Power Supply Current Profile
for IBIS v4.2 and IBIS v5.0 IO Models

Figure 6

SSN Simulation Example using IBIS v5.0 Behavioral Models

The SSN schematic is that shown in Figure 1. Additionally, the package S-parameter 103-port model was extracted using Cadence Power SI and uses the Touchstone v2.0 file format supporting multiple normalization values. This model represents 50 IOs (from silicon die to BGA ball), VCCO power at the die, VCCO power at the BGA balls with all the VCCO balls lumped together into one port, and the multiple reverse geometry package capacitors lumped into one port. The PCB is represented as T elements using Hspice syntax. To ensure good correlation over frequency, we have chosen to sweep the aggressor frequency from 50MHz to 500MHz in 50MHz frequency increments and use a clocking pattern for the aggressors. There are 16 aggressors and 1 victim. Each of the 16 aggressors is terminated to VCCO/2 with a 50 ohm resistor. The victim line is driving a static high and is terminated to VCCO with a 50 ohm resistor. We have chosen to show SSN simulations with a victim driving a static high because the driver is a CMOS circuit. Thus, the static high output will be connected to the VCCO power rail by way of a PMOS transistor. This will demonstrate the behavior of the power distribution network and its effect on SSN.

Each simulation consists of 250nS transient simulation to ensure that a minimum of 10 consecutive pulses of the aggressor is achieved. The only difference between the two cases of SSN simulations

is that the first case uses a Post-Layout extracted IO netlist model and the second case uses power aware IBIS v5.0 behavioral models. All simulations are run with Hspice VER: G-2012.06-SP1 on Windows 7 operating system. The option –mt 4 was used for all SSN simulations. Figure 7 through figure 16 show the comparison of the victim and aggressors for both case (1) and case (2). Figure 17 provides a table summary of the transient simulation results.

```
   ******  Runtime Statistics (seconds)  ******

   analysis            time    # points   tot. iter  conv.iter
   op point            0.01           1           4
   transient         622.73        2501       63587      22803
rev=     10159
   readin              0.02
   errchk            478.32
   setup               0.03
   output              0.00


           peak memory used        381.23 megabytes
           total cpu time         1101.16 seconds
           total elapsed time     1102.91 seconds
           job started at     10:45:27 11/07/2012
           job ended   at     11:03:50 11/07/2012
```

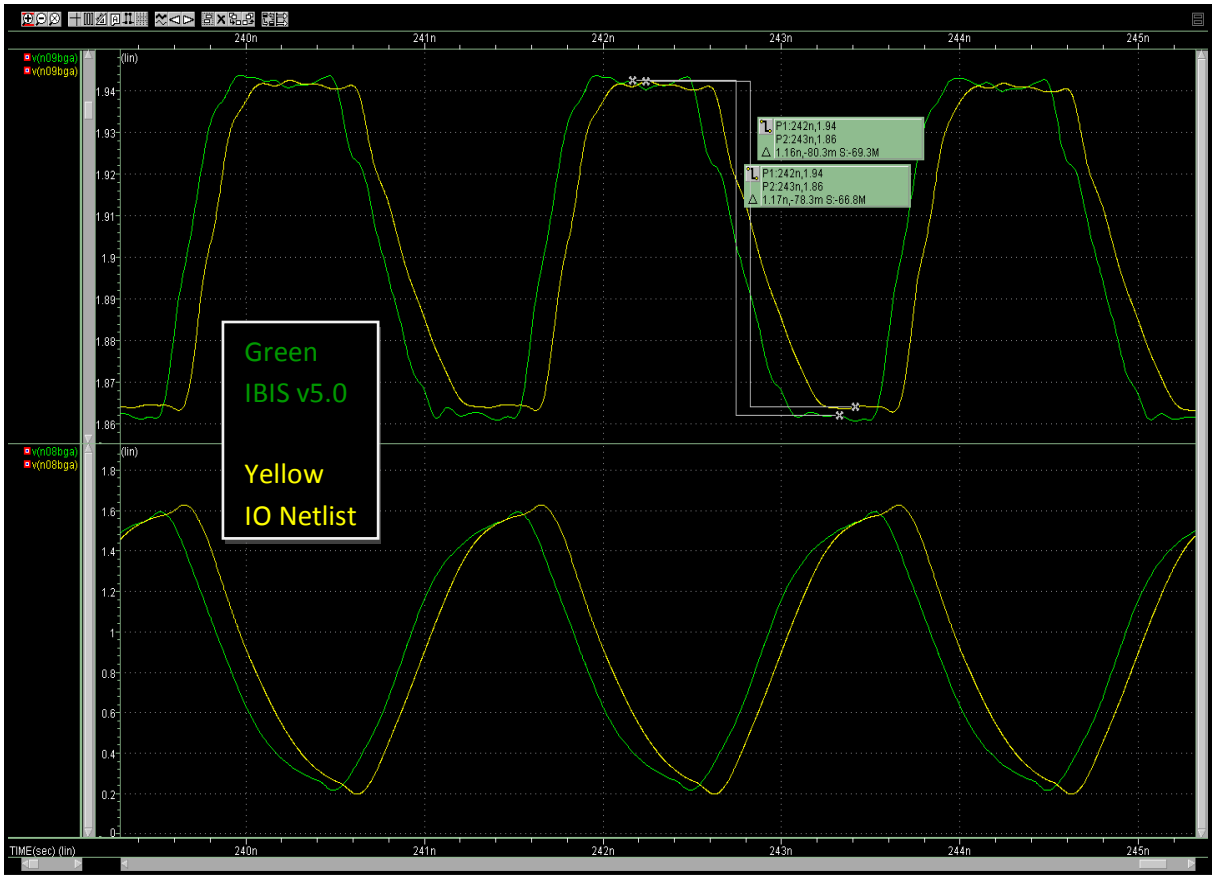
IBIS v5.0

```
   ******  Runtime Statistics (seconds)  ******

   analysis            time    # points   tot. iter  conv.iter
   op point           89.72           1          27
   transient      520858.95        2501      284815      96361
rev=      8605
   readin             92.35
   errchk           2223.93
   setup              14.15
   output              0.00


           peak memory used      17587.93 megabytes
           total cpu time       523809.36 seconds
           total elapsed time   523788.30 seconds
           job started at     12:04:16 11/01/2012
           job ended   at     12:34:27 11/07/2012
```

IO Netlist



500MHz Aggressors

Figure 7

```
******  Runtime Statistics (seconds)  ******

analysis            time     # points   tot. iter  conv.iter
op point            0.05            1           4
transient         625.43         2501       64932      22941
rev=      10326
readin              0.02
errchk            479.05
setup               0.01
output              0.00


          peak memory used        380.43 megabytes
          total cpu time         1104.58 seconds
          total elapsed time     1106.42 seconds
          job started at     10:45:50 11/07/2012
          job ended   at     11:04:17 11/07/2012
```

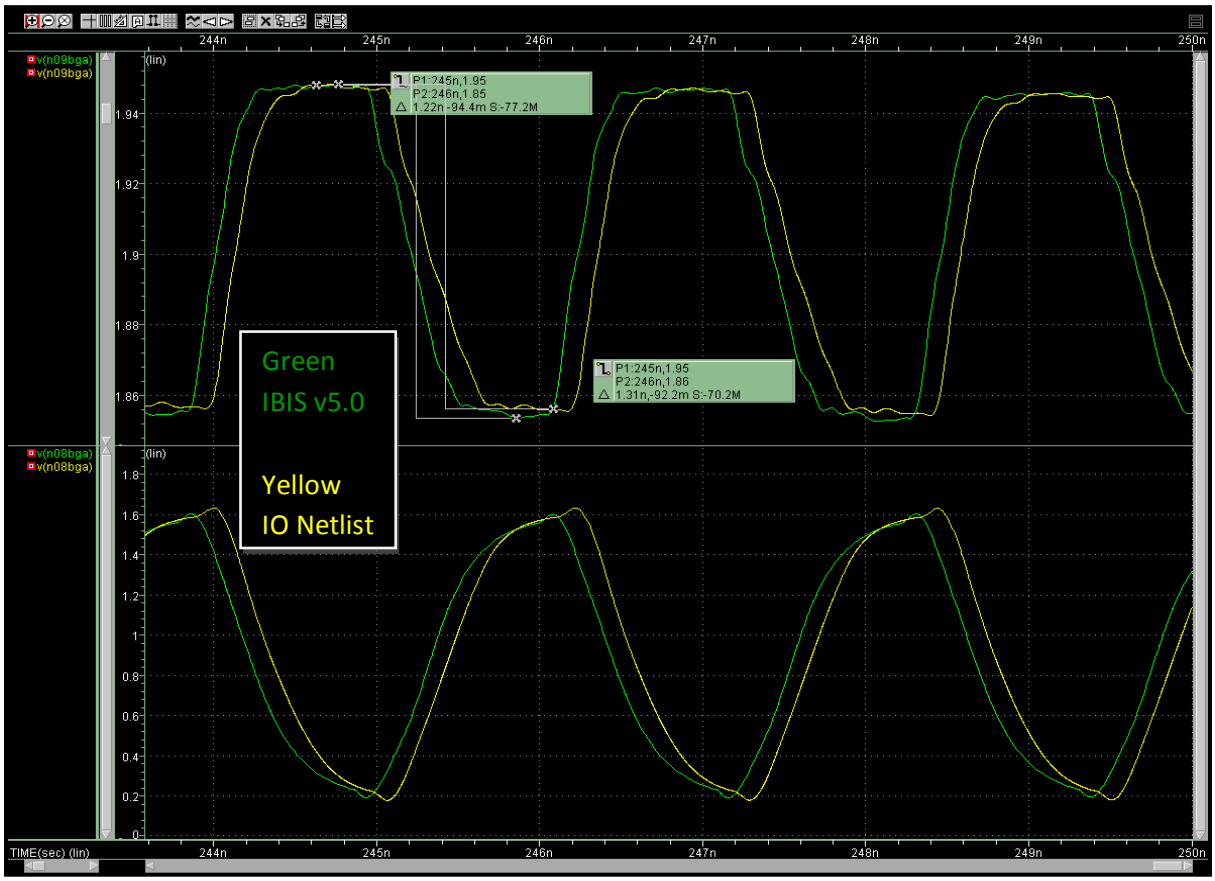
IBIS v5.0

```
******  Runtime Statistics (seconds)  ******

analysis            time     # points   tot. iter  conv.iter
op point           75.08            1          27
transient      470598.17         2501      284689      96336
rev=      8419
readin             92.18
errchk           2012.51
setup              13.91
output              0.02


          peak memory used      17588.93 megabytes
          total cpu time       473315.67 seconds
          total elapsed time   473296.89 seconds
          job started at     12:02:53 11/01/2012
          job ended   at     22:31:31 11/06/2012
```

IO Netlist



450MHz Aggressors

Figure 8

```
  ******  Runtime Statistics (seconds)  ******

  analysis           time    # points   tot. iter  conv.iter
  op point           0.01           1           4
  transient        632.82        2501       60322       20957
rev=      9819
  readin             0.02
  errchk           486.27
  setup              0.03
  output             0.00


          peak memory used        380.70 megabytes
          total cpu time         1119.20 seconds
          total elapsed time     1120.94 seconds
          job started at     09:56:19 11/07/2012
          job ended   at     10:15:00 11/07/2012
```

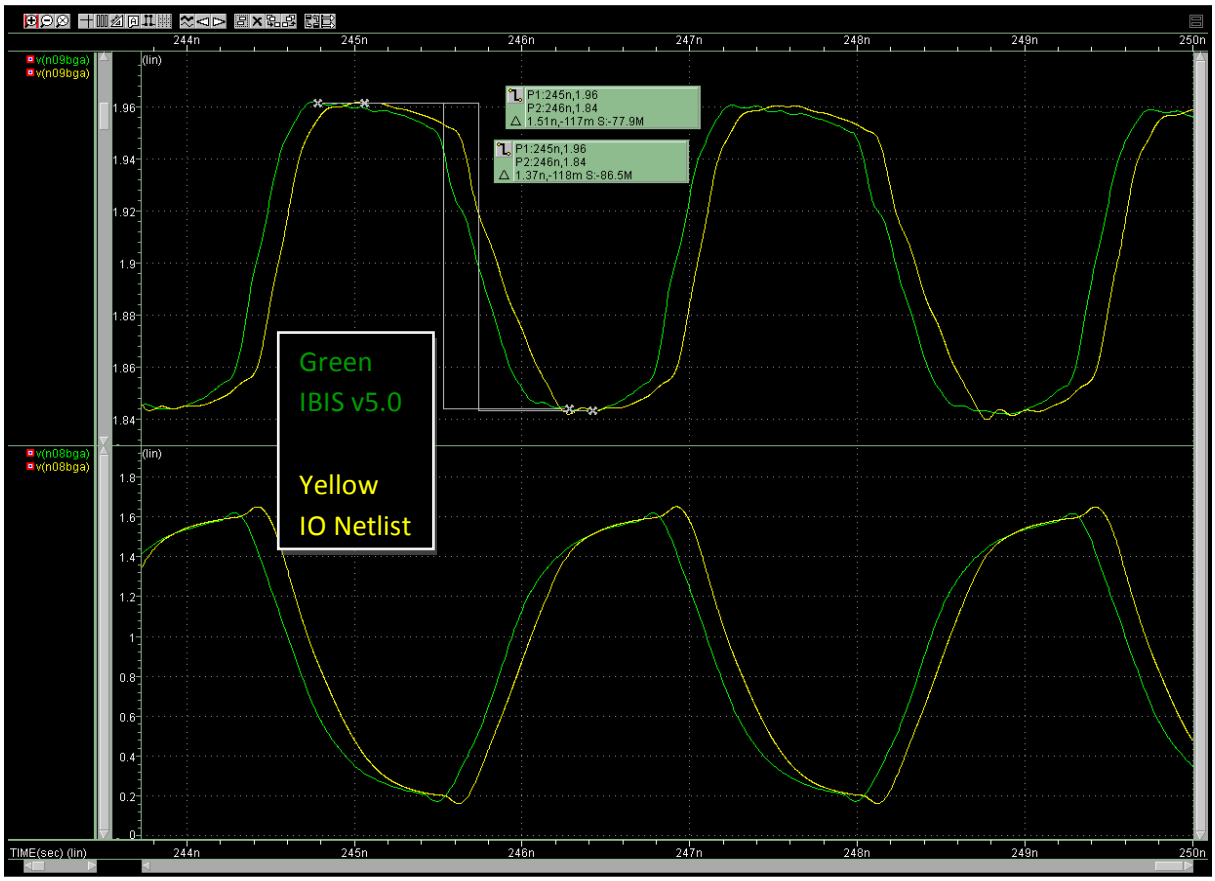
IBIS v5.0

```
  ******  Runtime Statistics (seconds)  ******

  analysis           time    # points   tot. iter  conv.iter
  op point          76.52           1          27
  transient     476529.10        2501      288106       97956
rev=      8970
  readin            92.00
  errchk          2044.21
  setup             13.68
  output             0.02


          peak memory used      17589.76 megabytes
          total cpu time       479279.42 seconds
          total elapsed time   479260.49 seconds
          job started at     12:02:29 11/01/2012
          job ended   at     00:10:30 11/07/2012
```

IO Netlist



400MHz Aggressors

Figure 9

```
******  Runtime Statistics (seconds)  ******

analysis            time     # points   tot. iter  conv.iter
op point            0.06            1           4
transient         602.87         2501       56394       19712
rev=      9190
readin              0.03
errchk            483.99
setup               0.03
output              0.00


          peak memory used        380.47 megabytes
          total cpu time         1086.98 seconds
          total elapsed time     1088.77 seconds
          job started at     09:54:58 11/07/2012
          job ended   at     10:13:06 11/07/2012
```

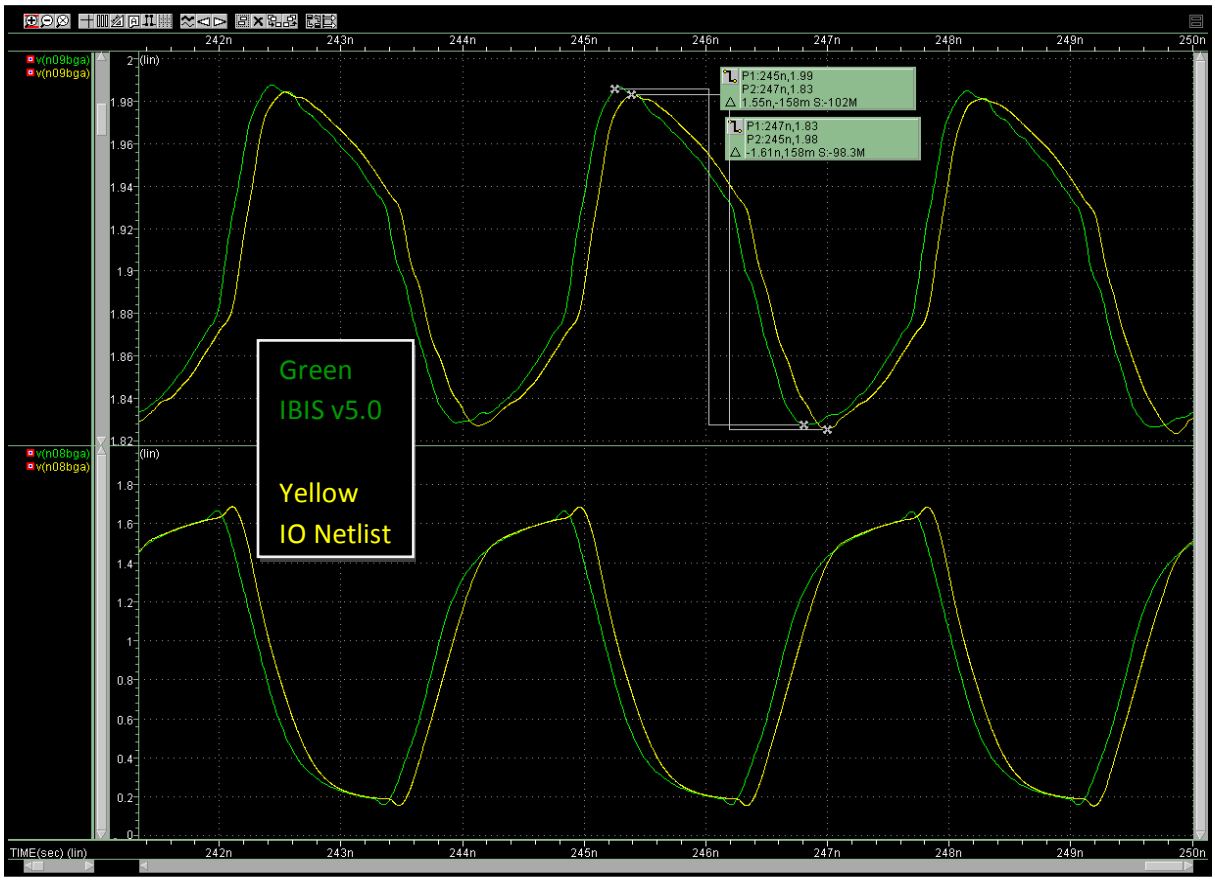
IBIS v5.0

```
******  Runtime Statistics (seconds)  ******

analysis            time     # points   tot. iter  conv.iter
op point           76.27            1          27
transient      443357.90         2501      267572       91125
rev=      8654
readin             98.45
errchk           2057.07
setup              13.74
output              0.01


          peak memory used      17589.05 megabytes
          total cpu time       446125.44 seconds
          total elapsed time   446107.89 seconds
          job started at     12:02:06 11/01/2012
          job ended   at     14:57:33 11/06/2012
```

IO Netlist



350MHz Aggressors

Figure 10

```
******  Runtime Statistics (seconds)  ******

  analysis            time    # points   tot. iter  conv.iter
  op point            0.03           1           4
  transient         220.09        2501       24176       9997
rev=       814
  readin              0.02
  errchk            496.37
  setup               0.03
  output              0.00


          peak memory used        379.86 megabytes
          total cpu time          716.57 seconds
          total elapsed time      718.30 seconds
          job started at     15:03:17 10/26/2012
          job ended   at     15:15:16 10/26/2012
```

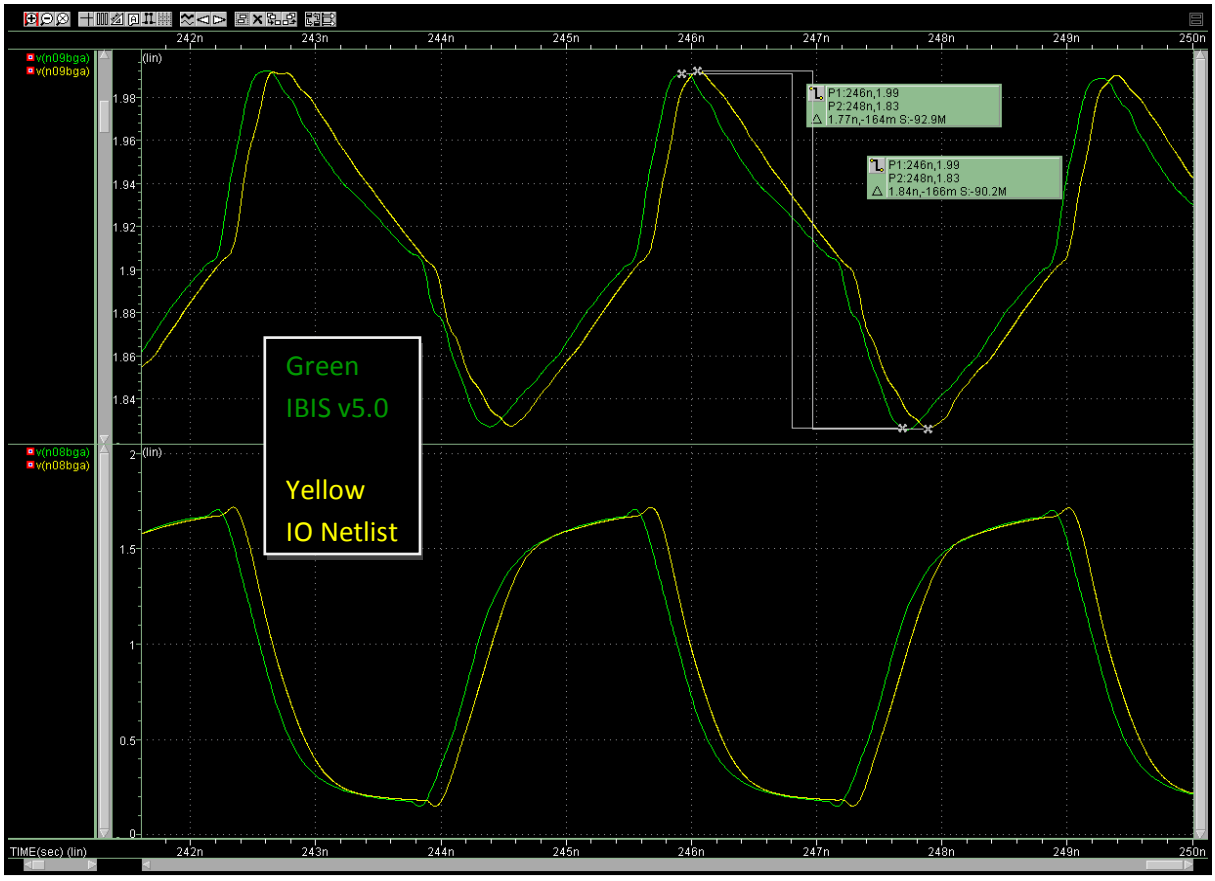
IBIS v5.0

```
******  Runtime Statistics (seconds)  ******

  analysis            time    # points   tot. iter  conv.iter
  op point           73.81           1          27
  transient      419634.29        2501      252878      86324
rev=      8469
  readin             90.31
  errchk           1987.83
  setup              13.00
  output              0.02


          peak memory used      17590.52 megabytes
          total cpu time       422314.92 seconds
          total elapsed time   422298.18 seconds
          job started at     13:32:30 10/26/2012
          job ended   at     10:51:07 10/31/2012
```

IO Netlist



300MHz Aggressors

Figure 11

```
  ******  Runtime Statistics (seconds)  ******

  analysis           time    # points   tot. iter  conv.iter
  op point           0.02           1           4
  transient        215.45        2501       23375        9604
rev=       797
  readin             0.01
  errchk           501.12
  setup              0.03
  output             0.00


          peak memory used        381.05 megabytes
          total cpu time          716.69 seconds
          total elapsed time      718.53 seconds
          job started at     15:06:48 10/26/2012
          job ended   at     15:18:46 10/26/2012
```

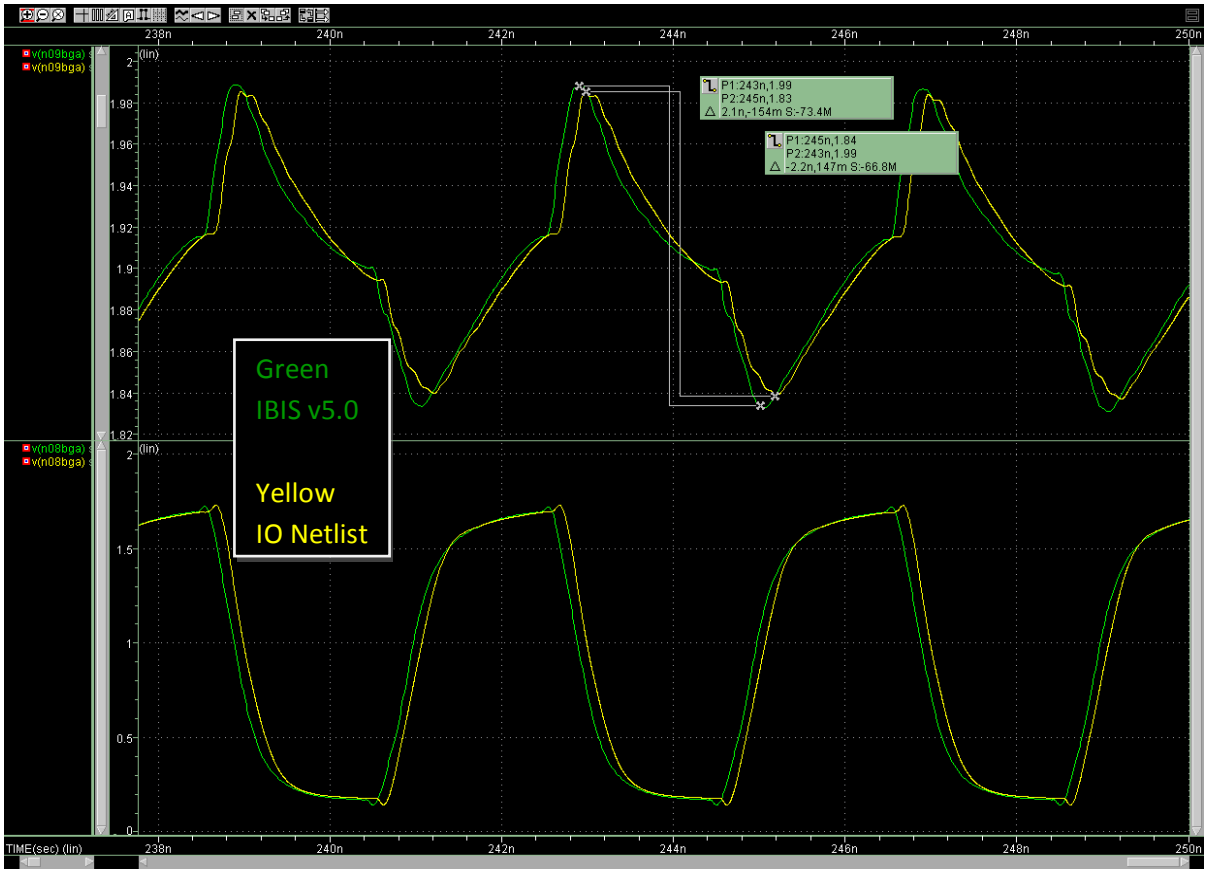
IBIS v5.0

```
  ******  Runtime Statistics (seconds)  ******

  analysis           time    # points   tot. iter  conv.iter
  op point          81.92           1          27
  transient     402337.99        2501      236983       81248
rev=      8063
  readin            90.42
  errchk          2001.91
  setup             15.24
  output             0.00


          peak memory used      17589.12 megabytes
          total cpu time       405082.77 seconds
          total elapsed time   405066.89 seconds
          job started at     14:01:48 10/26/2012
          job ended   at     06:33:13 10/31/2012
```

IO Netlist



250MHz Aggressors

Figure 12

```
******  Runtime Statistics (seconds)  ******

analysis           time     # points   tot. iter  conv.iter
op point           0.02            1           4
transient        165.95         2001       17714       7193
rev=       543
readin             0.02
errchk           504.23
setup              0.03
output             0.00


          peak memory used        380.36 megabytes
          total cpu time          670.28 seconds
          total elapsed time      672.23 seconds
          job started at     15:08:43 10/26/2012
          job ended   at     15:19:55 10/26/2012
```

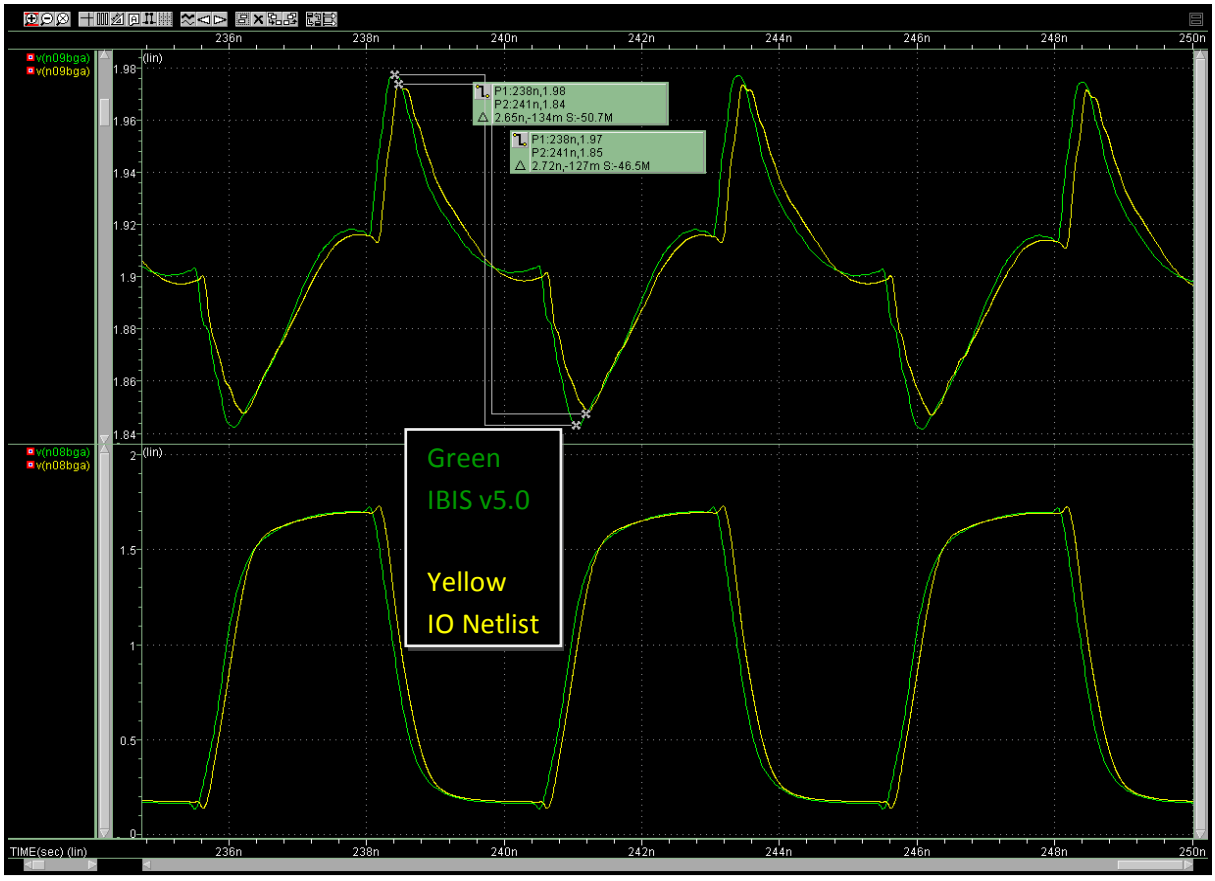
IBIS v5.0

```
******  Runtime Statistics (seconds)  ******

analysis           time     # points   tot. iter  conv.iter
op point          77.55            1          27
transient     335426.41         2501      200299      68830
rev=      6929
readin            92.20
errchk          2106.45
setup             14.96
output             0.00


          peak memory used      17591.40 megabytes
          total cpu time       338253.48 seconds
          total elapsed time   338240.47 seconds
          job started at     14:18:31 10/26/2012
          job ended   at     12:16:06 10/30/2012
```

IO Netlist



200MHz Aggressors

Figure 13

```
******  Runtime Statistics (seconds)  ******

analysis           time    # points   tot. iter  conv.iter
op point           0.06           1           4
transient        236.81        2501       21441        8713
rev=       619
readin             0.02
errchk           526.51
setup              0.03
output             0.00


          peak memory used        380.34 megabytes
          total cpu time          763.48 seconds
          total elapsed time      765.26 seconds
          job started at     15:36:18 10/26/2012
          job ended   at     15:49:04 10/26/2012
```

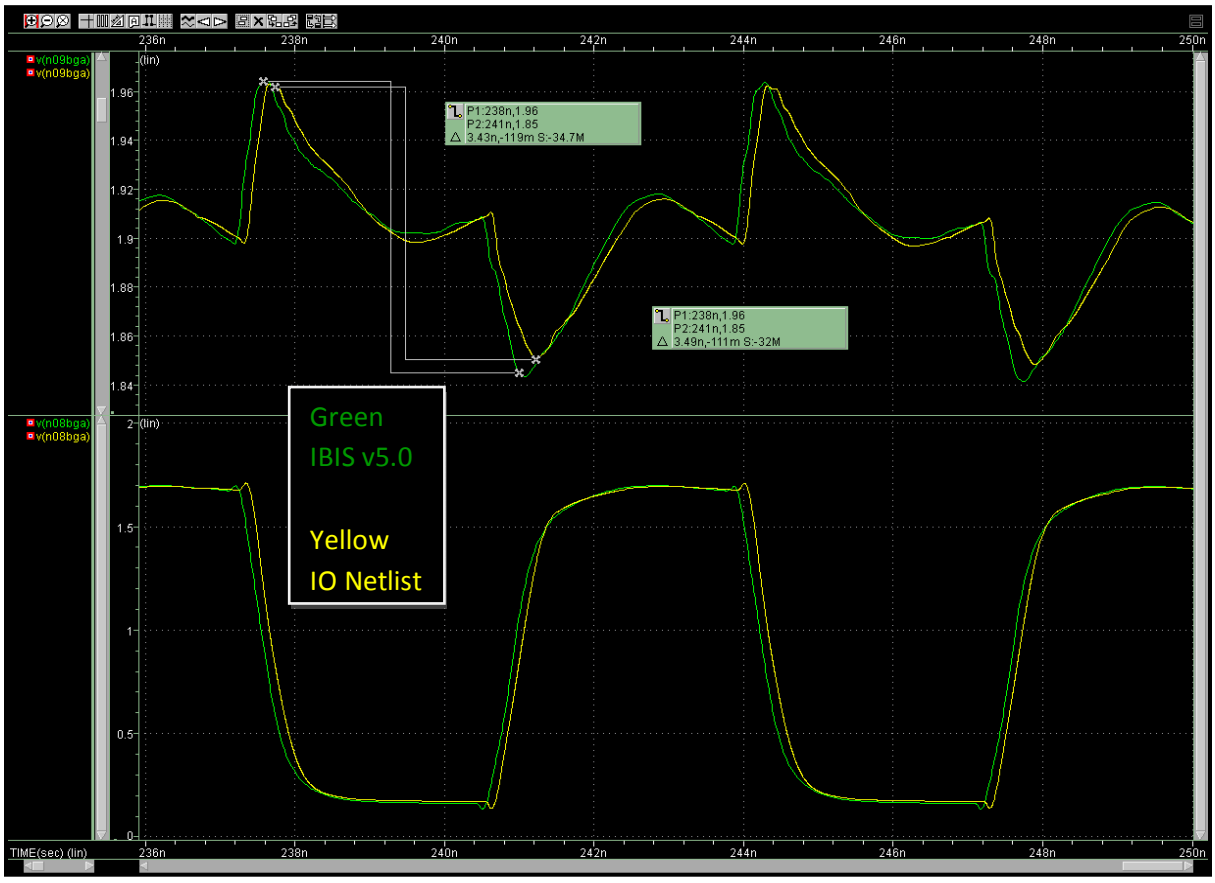
IBIS v5.0

```
******  Runtime Statistics (seconds)  ******

analysis           time    # points   tot. iter  conv.iter
op point          83.28           1          27
transient     284351.30        2501      164910       56786
rev=      5809
readin            94.76
errchk          2119.87
setup             14.77
output             0.00


          peak memory used      17589.78 megabytes
          total cpu time       287203.92 seconds
          total elapsed time   287193.02 seconds
          job started at     14:21:06 10/26/2012
          job ended   at     22:07:52 10/29/2012
```

IO Netlist



150MHz Aggressors

Figure 14

```
   ******  Runtime Statistics (seconds)  ******

   analysis           time    # points   tot. iter  conv.iter
   op point           0.02           1           4
   transient        195.05        2501       20700       8497
rev=       532
   readin             0.01
   errchk           493.78
   setup              0.01
   output             0.00


           peak memory used        380.50 megabytes
           total cpu time          688.92 seconds
           total elapsed time      690.69 seconds
           job started at     15:36:12 10/26/2012
           job ended   at     15:47:43 10/26/2012
```

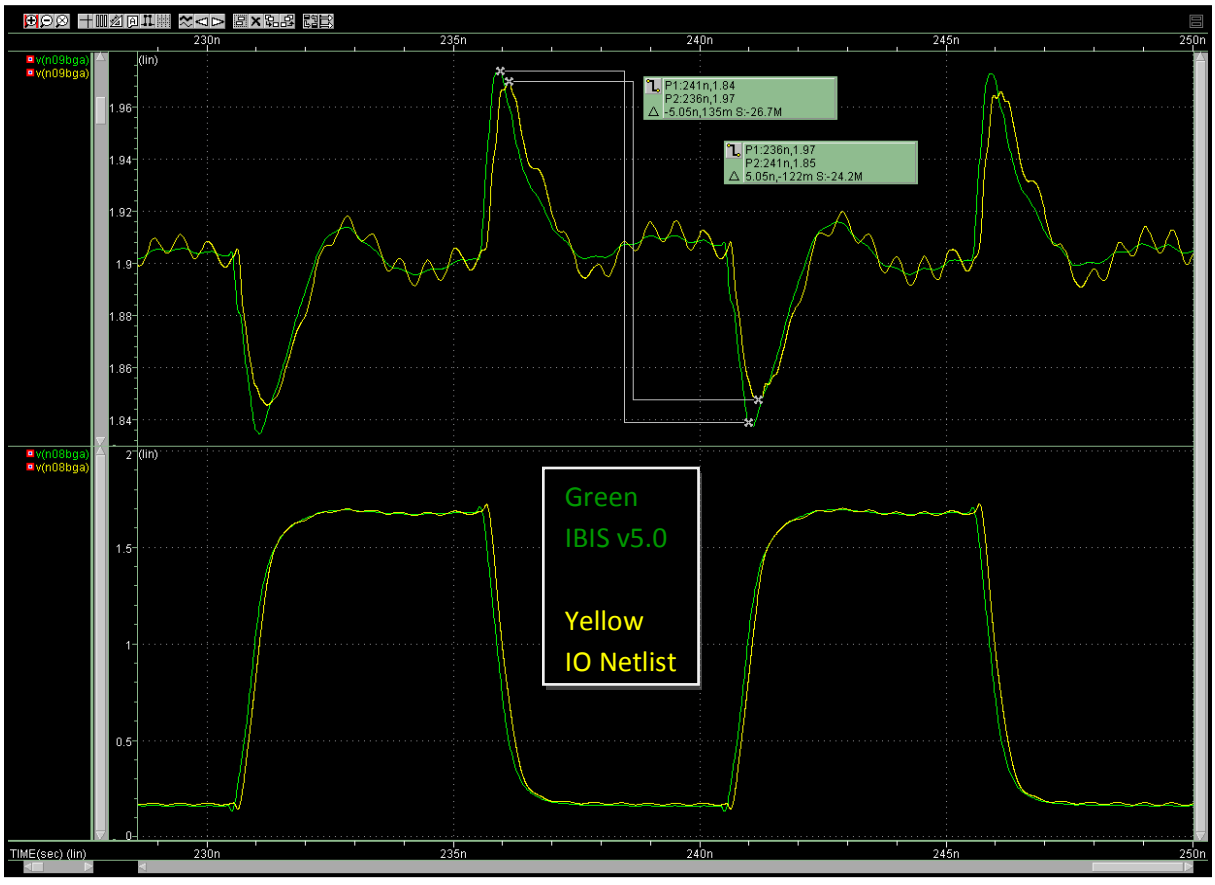
IBIS v5.0

```
   ******  Runtime Statistics (seconds)  ******

   analysis           time    # points   tot. iter  conv.iter
   op point          82.79           1          27
   transient     208975.16        2501      118908      40914
rev=      4071
   readin            93.48
   errchk          2001.57
   setup             13.63
   output             0.02


           peak memory used      17589.10 megabytes
           total cpu time       211708.69 seconds
           total elapsed time   211701.31 seconds
           job started at     14:22:46 10/26/2012
           job ended   at     01:11:16 10/29/2012
```

IO Netlist



100MHz Aggressors

Figure 15

```
******  Runtime Statistics (seconds)  ******

analysis            time    # points   tot. iter  conv.iter
op point            0.03           1           4
transient         188.84        2501       19940        8237
rev=        483
readin              0.02
errchk            494.30
setup               0.03
output              0.00


          peak memory used        380.09 megabytes
          total cpu time          683.23 seconds
          total elapsed time      685.00 seconds
          job started at     15:36:08 10/26/2012
          job ended   at     15:47:33 10/26/2012
```

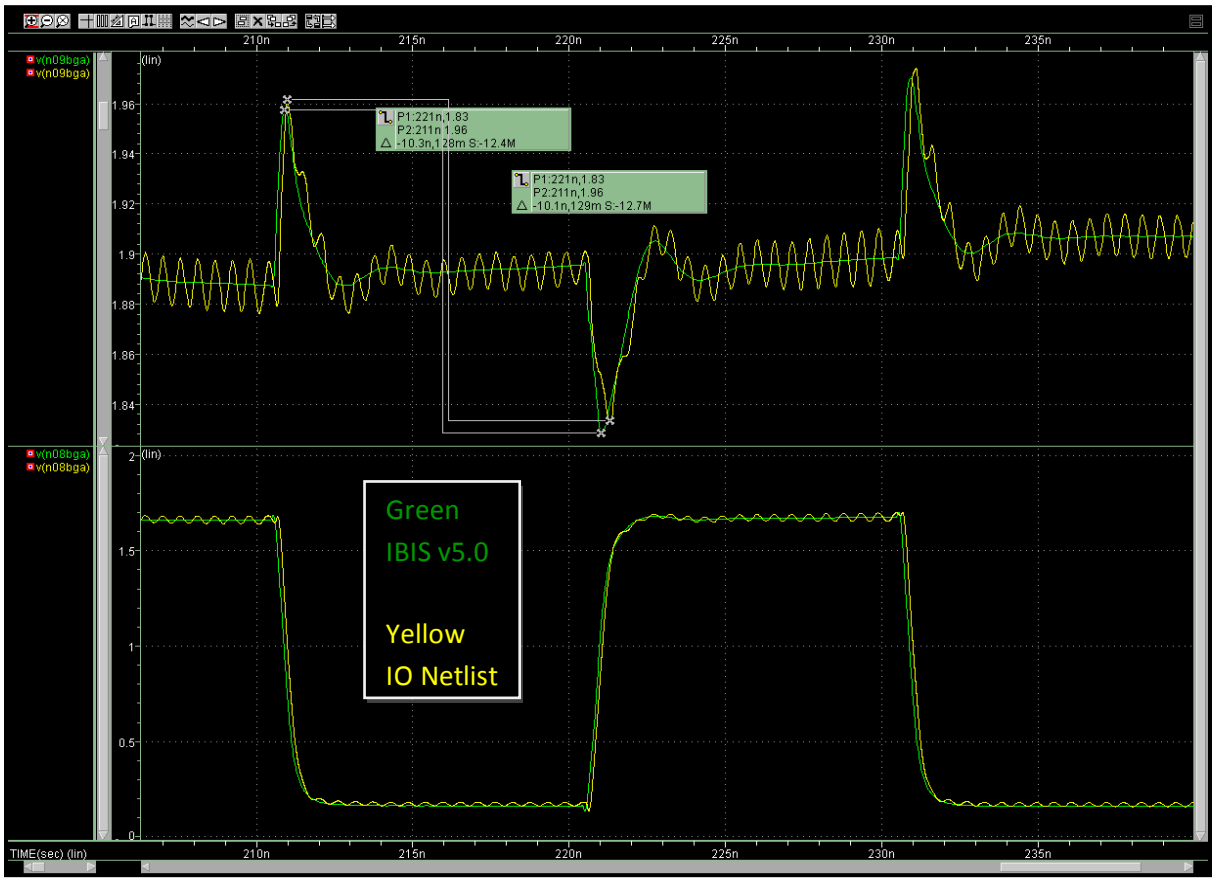
IBIS v5.0

```
******  Runtime Statistics (seconds)  ******

analysis            time    # points   tot. iter  conv.iter
op point           92.91           1          27
transient      141534.92        2501       79277       27365
rev=       2762
readin             94.91
errchk           2148.96
setup              14.65
output              0.00


          peak memory used      17603.46 megabytes
          total cpu time       144439.36 seconds
          total elapsed time   144434.98 seconds
          job started at     14:23:53 10/26/2012
          job ended   at     06:31:14 10/28/2012
```

IO Netlist



50MHz Aggressors

Figure 16

Comparing Simulation Results

	Post Layout Extracted IO Netlist Model			IBIS v5.0 Behavioral Model		
Freq (MHz)	P-P Noise (mV)	RAM Usage (Mb)	CPU Time	P-P Noise (mV)	RAM Usage (Mb)	CPU Time
50	128	17603.46	1d 16h 07m 16s	129	380.09	11m 23s
100	122	17589.10	2d 10h 48m 29s	135	380.50	11m 29s
150	111	17589.78	3d 07h 46m 43s	119	380.34	12m 43s
200	127	17591.40	3d 21h 57m 33s	134	380.36	11m 10s
250	147	17589.12	4d 16h 31m 23s	154	381.05	11m 57s
300	164	175905.5	4d 21h 18m 35s	166	379.86	11m 57s
350	158	17589.05	5d 03h 55m 25s	158	380.47	18m 07s
400	117	17589.76	5d 32h 49m 58s	118	380.70	18m 39s
450	91	17588.93	5d 11h 28m 36s	94	380.43	18m 24s
500	78	17587.93	6d 01h 30m 09s	80	381.23	18m 21s

Figure 17

The .option runlvl=6 was set when using IBIS v5.0 models. Setting runlvl=6 results in smaller time steps (more Newton-Raphson iterations) to meet stricter error tolerances, and higher simulation accuracy. Using runlvl=6 results in longer simulation than using runlvl=5. However, since using the IBISv5.0 behavioral models results in simulation times on the order of 10 minutes, we can afford to use the higher accuracy option of runlvl=6. For the case using the post layout extracted IO netlists, the runlvl was set to 5. When setting the runlvl to 6, the simulation time for the 50MHz aggressor frequency case was greater than 1 week before the *.tr0 file was created.

In Hspice transient analysis, there are several options to solve the circuit's differential algebraic equations. We have used .option METHOD=BDF for the IBIS v5.0 SSN simulations. The BDF method is a high-order integration method that uses backward differentiation formulae. The default method is the METHOD=TRAP. The trapezoidal method is the preferred algorithm for low simulation time. However, since using the IBISv5.0 models results in simulation times on the order of 10 minutes, we can afford to use the higher accuracy method of BDF. For the SSN using the post layout extracted IO netlists, the method was set to TRAP. Otherwise, the simulation was prohibitively long. As can be seen on the 50 MHz and 100MHz aggressor frequency using the Post-Layout IO netlists, there is a noticeably oscillation at 2GHz. This oscillation can be removed by setting the runlvl to 6 and the method to BDF but the simulation time will exceed 1 week.

Conclusion

Currently, high accuracy SSN simulations take days when using Post-Layout extracted IO netlist models, but take only minutes when using power aware IBIS v5.0 behavioral models. For similar circuits, SSN simulations using HSPICE IO models require gigabytes of RAM while power aware IBIS v5.0 behavioral models only use megabytes of RAM. Post-Layout extracted IO netlist models also may require lower accuracy simulation options compared to power aware IBIS v5.0 behavioral models due to the long simulation times. Unfortunately, waveform artifacts such as the ringing as seen in the 50MHz SSN simulation could be included in the results. With the improved simulation times, reduced memory requirements and consistent accuracy, Power Aware IBIS v5.0 behavioral models are an excellent improvement for SSN simulations allowing more aggressive SI/PI analysis than was previously possible.

References

[1] Madhavan Swaminathan, Ege Engin, "Power Integrity Modeling and Design for Semiconductors and Systems", 2007.

[2] Synopsys , "HSPICE user guide - 2012.06"

[3] IBIS Open Forum Committee, "I/O Buffer Information Specification (IBIS) Version 5.0", August 29, 2008, http://eda.org/ibis/ver5.0/ver5_0.txt

[4] Simon Ramo, John R. Whinnery, Theodore Van Duzer, "Fields and Waves in Communication Electronics", Second Edition, 1984.

[5] Xilinx Inc., "Simulating FPGA Power Integrity with S-Parameter Models", http://www.xilinx.com/support/documentation/white_papers/wp411/Sim_Power_Integrity.pdf

[6] Xilinx Inc., "Xilinx Virtex-6/Spartan-6 FPGA DDR3 Signal Integrity Analysis and PCB Layout Guidelines", http://www.xilinx.com/support/documentation/white_papers/wp420-DDR3-SI-PCB.pdf