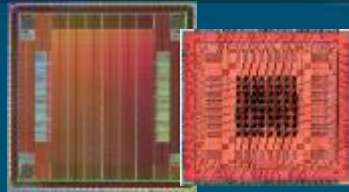
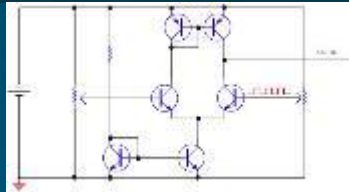


Latest Developments in the Xyce Large-Scale Analog Circuit Simulator



PRESENTED BY

Jason C. Verley



Why Develop an Analog Circuit Simulator at Sandia?

Why Develop an Analog Circuit Simulator at Sandia?



Commercial tools cannot simulate the high dose rate radiation environments that are Sandia's primary concern

A known, reliable code base, owned by Sandia

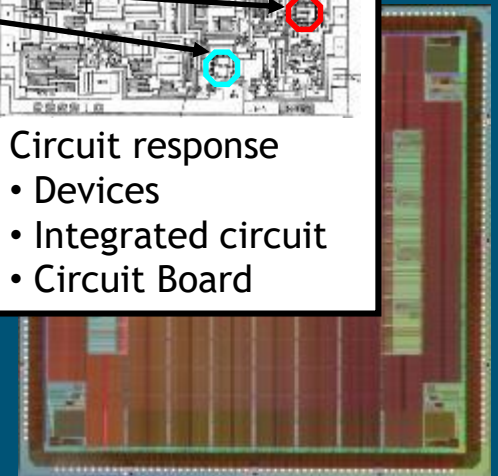
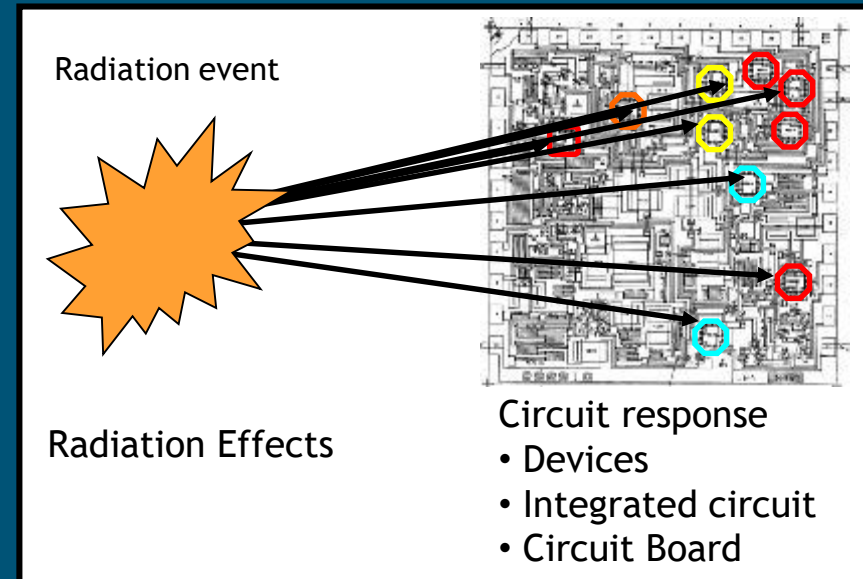
- Prompt response to internal needs
- Trusted code

Specialized compact models for radiation effects

- High-energy photons (X-rays and γ -rays)
- Neutrons

Simulation of entire systems

- Radiation effects are not isolated...
- Need massively parallel simulations



The Xyce Analog Circuit Simulator



SPICE-Compatible syntax (Berkeley 3f5)

Two versions, **Serial** and...

Distributed Memory Parallel (MPI-based)

Unique solver algorithms

Industry standard models

Non-traditional models

- Memristor
- Neuron/synapse
- TCAD (PDE-based)

<http://xyce.sandia.gov>

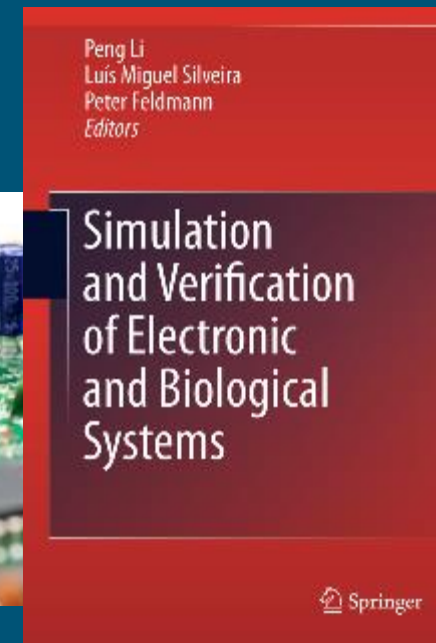
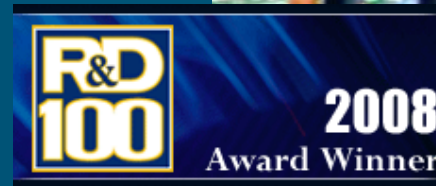
<http://github.com/Xyce>

Open Source, GPLv3

- Since September of 2013 (Xyce 6.0)

Xyce Release 6.12

- October, 2019; 25th major release
- >5,100 external downloads since 6.0



Keiter, et al.,
“Parallel
Transistor-Level
Circuit Simulation”





Xyce defaults are conservative

- The industry standard is, “The simulator must never fail to provide an answer...”
...even if it’s wrong.
- The Xyce philosophy: provide a numerically accurate answer, and fail if asked to do something “wrong.”

Simulations in Serial vs. MPI Parallel

- Distributed parallelism can take more tuning than shared memory approaches...
especially device distribution, and direct vs. iterative linear solvers.
- Very large parallel simulations are challenging (need to find the “right” linear solver)
- Leverages Sandia’s Trilinos High-Performance Computing (HPC) solver framework

Xyce is the simulator (like HSPICE, SmartSpice, Spectre,...)

- There is no Schematic Design/Capture Front End (like Virtuoso, Tanner AMS, Gateway,... i.e., a GUI)
...for now?

Others:

- Xyce, *its e lf*, is not 100% compatible with any other simulator (at the moment)
- There may be an expected feature that we don’t (yet) support; e.g., .OP functionality is limited



Typical

DC, Transient, AC, Noise

- .DC, .TRAN, .NOISE, .AC (and .STEP)

Post Processing:

- Fourier transform of transient output (.FOUR)
- Post-simulation calculation of simulation metrics (.MEASURE)

Output (.PRINT)

- Text Files (tab or comma delimited)
- Probe (PSPICE)
- Gnuplot, TecPlot, RAW (SPICE 3f5)

Analog Behavioral Modeling

Expressions, functions, parameterizations...

Others

Harmonic Balance Analysis (.HB)

- Steady state solution of nonlinear circuits in the frequency domain

Random Sampling Analysis

- Executes the primary analysis (.DC, .AC, .TRAN, etc.) inside a loop over randomly distributed parameters

Sensitivities

- Computes sensitivities for a user-specified objective function with respect to a user-specified list of circuit parameters ($\partial O / \partial p \dots$)
- DC or Transient
- E.g., an output voltage's dependence on a capacitance

S-Parameter Analysis

- Touchstone file output



Latest Xyce Developments



Addition of .LIN (Linear Analysis)

- Extract linear transfer parameters (S-, Y- or Z-parameters) for a general multiport network
- Output in the Touchstone standard format (1 or 2)
- Input of Touchstone files is continuing now

Improvements to the mixed-signal interface via VPI (Verilog Procedural Interface)

Up to 8× Improvement in Verilog-A-based device evaluation speeds (next slide)

- ~ 5× improvement in certain test circuit run times

20% reduction in memory use

2× improvement in certain circuit evaluations due to linear solver optimizations

Bug fixes, output enhancements, .MEASURE with .AC, etc. (see the Release Notes for full details)



The original Xyce/ADMS backend used the Sacado (Trilinos) automatic differentiation library.

- Simplified the development of Xyce-specific XML templates for the Jacobian element calculation
- Resulted in a different approach than that taken for SPICE (and others, like ngspice or QUCS)
- Slower models vs. using analytic derivatives
- Unable to implement **ddx**, since Sacado calculates derivatives at runtime

In Xyce 6.12, the Xyce/ADMS compiler was modified to use analytic derivatives

- Based on the homegrown Xyce XML templates, so it eased implementation
- Sacado is still used for sensitivity calculations, since the derivative combinations are not known until runtime

Future improvements

- Near-term: full support for **ddx** (necessary for the BSIM-SOI and HiSIM models)
- Long-term: development of a non-ADMS-based model compiler

Recent Compact Model Developments



Newly-released FOSS EKV 2.6 model implemented in Xyce

- Sandia has a lack of internal model cards for testing, so an EKV 2.6 test suite would be nice...

Xyce Support for the 18-month exclusivity of CMC-supported models

- Initiating a new (not open-source) copyright assertion
- Distribution of binary-only versions of Xyce that include the latest models
RHEL 7, Mac OS X, Windows

The Open Source (GPLv3) version of Xyce will continue to include the publicly available models



What is a PDK ...to an analog circuit simulator?

- Subcircuits
- .MODEL definitions
- Functions and parameters (expressions)
- Verilog-A
- ...basically a big netlist spread across hundreds of files.

PDK compatibility comes down to

- Netlist language compatibility (HSPICE or Spectre)
- Feature compatibility (HSPICE or Spectre)

So...



HSPICE netlist language compatibility

- XDM Translation Tool – a 90% solution

Feature compatibility

- Global Foundries 55/65 nm
 - Good (so far)
- Global Foundries 12 nm
 - Xyce bug regarding deeply nested .FUNC calls, hopefully to be resolved by the 7.0 release

Some known deficiencies:

- Spectre compatibility
- Multipliers ("M=") with subcircuits
- Full support of AGAUSS, GAUSS and RAND
- TNOM bug
- TMI/OMI (required for full TSMC PDK support)
- ...other, smaller issues

Translation Procedure

- Pre-process the HSPICE PDK, using XDM
- Fix any remaining issues by hand (most, if not all, are known)
- Translate the netlist for the specific simulation, using XDM
 - Again, fix any remaining issues by hand
 - Simulation-specific directives are often not translatable

When will 7.0 be released?

- January 2020

Planned Developments



Continue to Improve Compatibility with Commercial Simulators

- Integration of the XDM translator into Xyce, itself
- Improving HSPICE feature compatibility
- Develop compatibility with Spectre (both netlist and feature)

PDK Support

- Continue to test PDKs from Global Foundries, and other foundries (TSMC?), as available

Performance Improvements

- Begin to incorporate FastSpice methods
- Continued improvements to the linear solver

Others

- Development of a device, which has a response defined by a Touchstone 1 or 2 file
- Continued improvement for mixed-signal simulation (Verilog via VPI, VHDL via VHPI)
- Coupling to the gds2para code for improved simulation of IC parasitics
- Move build system to CMake

Xyce Team Acknowledgements



Eric R. Keiter

Thomas V. Russo

Richard L. Schiek

Heidi K. Thornquist

Ting Mei

Jason C. Verley

Peter E. Sholander

Karthik V. Aadithya

Gary J. Templet, Jr. (XDM)

Garrick Ng (XDM)

...and many others

Contact:

<http://xyce.sandia.gov>

<http://github.com/Xyce>

xyce@sandia.gov

Google Group Forum:

<https://groups.google.com/forum/#!forum/xyce-users>

