SANDIA REPORT

SAND2014-2402
Unlimited Release
Printed March 2014

Xyce™ Parallel Electronic Simulator Release Notes Release 6.1

Eric R. Keiter, Ting Mei, Thomas V. Russo, Richard L. Schiek, Peter E. Sholander, Heidi K. Thornquist, Jason C. Verley, David G. Baur

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2014-2402
Unlimited Release
Printed March 2014

Xyce™ Parallel Electronic Simulator Release Notes Release 6.1

Eric R. Keiter, Ting Mei, Thomas V. Russo,
Richard L. Schiek, Peter E. Sholander, Heidi K. Thornquist, Jason C. Verley
Electrical Models and Simulation

Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-1177

David G. Baur
Raytheon
1300 Eubank Blvd
Albuquerque, NM 87123

Abstract

The highlights of the ***Xyce***™ 6.1 release are documented.

Contents

Scope/Product Definition

The **Xyce** Parallel Electronic Simulator has been written to support, in a rigorous manner, the simulation needs of the Sandia National Laboratories electrical designers. Specific requirements include, among others, the ability to solve extremely large circuit problems by supporting large-scale parallel computing platforms, improved numerical performance and object-oriented code design and implementation.

The **Xyce** release notes describe:

- Hardware and software requirements
- New features and enhancements
- Any defects fixed since the last release
- Current known defects and defect workarounds
- Incompatibilities With Other Circuit Simulators

For up-to-date information not available at the time these notes were produced, please visit the **Xyce** web page at http://xyce.sandia.gov/.

Hardware/Software

This section gives basic information on supported platforms and hardware and software requirements for running **Xyce** 6.1.

Supported Platforms (Certified Support)

Xyce 6.1 currently supports any of the following operating system platforms (all versions imply the earliest supported—**Xyce** generally works on later versions as well). These platforms are supported in the sense that they have been subject to certification testing for the **Xyce** version 6.1 release.

- Red Hat Enterprise Linux® 5, x86 (serial only) and x86-64 (serial and parallel)
- Red Hat Enterprise Linux® 6, x86-64 (serial and parallel)
- Microsoft Windows 7®, x86 (serial)
- Apple® OS X, x86-64 (serial and parallel)
- TLCC (serial and parallel)
- Red Sky (serial and parallel)

Build Supported Platforms (not Certified)

The platforms listed in this section are "not supported" in the sense that they are not subject to nightly regression testing, and they also were not subject to certification testing for the **Xyce** version 6.1 release. Despite this lack of testing rigor, **Xyce** has been known to run under these configurations.

- FreeBSD 9.x on Intel x86 and x86-64 architectures (serial and parallel)
- Distributions of Linux other than Red Hat Enterprise Linux
- Microsoft Windows under Cygwin and MinGW.

Please contact the Xyce development team for platform and configuration availability.

Hardware Requirements

The **Xyce** team has not determined a minimum memory or processor speed requirement. Any modern computer should have enough memory and processor power to run moderately sized circuits in serial with **Xyce**. Naturally, memory requirements grow with problem size.

Running **Xyce** in parallel will require a system with at least two processors. For problems of the size where parallel operation is beneficial (thousands of devices per processor), one can expect to need several gigabytes of memory per processor.

For the very largest problems that **Xyce** can run (millions of devices), one will require a cluster system with a sufficient number of nodes to distribute the problem efficiently, and sufficient memory per node to support the distributed problem.

Software Requirements

Several libraries are required to run **Xyce** or build **Xyce** from source on a platform. Serial versions of the **Xyce** binary have no run-time software requirements, as they ship with all the shared libraries they need. However, parallel versions require Open MPI (`http://www.open-mpi.org/`) (version 1.4 or higher) at run time. TLCC and Red Sky users can load the **xyce** module to properly set the environment.

Several libraries (all freely available from Sandia National Laboratories or other sites) are always required when building **Xyce** from source. These are:

- Trilinos Solver Library version 11.6.1 (`http://trilinos.org`). Trilinos is a suite of packages that includes solver and partitioning libraries in parallel and serial, in addition to many other utilites.

- UMFPACK version 4.1 and AMD version 1.0 (libumfpack.a, libamd.a) (http://www.cise.ufl.edu/research/sparse/umfpack/). This is also provided by the SuiteSparse package that is available on many systems.
- BLAS (Basic Linear Algebra Subprograms). This may be included with the compiler, or it may require separate package installation. One may also use ATLAS (http://math-atlas.sourceforge.net).
- LAPACK (Linear Algebra PACKage). This may be included with the compiler, or it may require separate package installation. One may also use ATLAS (http://math-atlas.sourceforge.net).

For parallel builds, the following libraries are additionally required:

- Open MPI (http://www.open-mpi.org) library for message passing (version 1.4 or higher). The version used to build Xyce must be the same that is used for building Trilinos.
- ParMETIS (http://glaros.dtc.umn.edu/gkhome/metis/parmetis/overview) library for graph partitioning (version 3.1 or higher). The MPI compiler used to compile ParMETIS must be the same that is used for Trilinos and Xyce.

Xyce Release 6.1 Documentation

The following **Xyce** documentation is available at the **Xyce** website in pdf form.

- **Xyce** Release Notes, Version 6.1 (this document)
- **Xyce** Users' Guide, Version 6.1
- **Xyce** Reference Guide, Version 6.1
- **Xyce** Mathematical Formulation
- Application Node: Using Open Source Schematic Capture Tools with **Xyce**

Also included at the **Xyce** website as web pages are the following.

- Building Guide (instructions for building Xyce from the source code)
- Running the Xyce Regression Test Suite

New Features and Enhancements

Highlights for **Xyce** Release 6.1 include:

- Simplified device code interface.
- Improved robustness of Harmonic Balance analysis.
- Improved voltage limiting handling for both the Trapezoid and Gear time integrators.
- Parallel AC, harmonic balance and MPDE analysis.
- Support for multi-input ($\geq$2) AND, NAND, OR and NOR gates and a new digital Buffer (BUF) model.
- Improved compatibility between Xyce and PSpice digital device models.

For details of each of these new features, see the **Xyce** Users' Guide, the **Xyce** Installation Guide, and the **Xyce** Reference Guide. A more complete listing of new features and improvements is given in the following sections.

Device Support

The device model package has been reworked to simplify the process of adding new devices to **Xyce**.

An alpha-release capability for using dynamic device plug-ins has been added. This feature is supported on Linux, Macintosh, FreeBSD, and other true Unix-like operating systems. Accessing this feature requires that the user build **Xyce** from source code; it is not available in the pre-compiled binaries we provide. In this initial implementation, the feature is not yet supported in Cygwin builds on Windows, nor is it supported on native Windows builds.

An example of using this capability is provided with the **Xyce** source code in the `user_plugin` directory. The example demonstrates how to use a simple model defined in Verilog to create a new model in Xyce, and to use this model directly as a plug-in without recompiling the rest of **Xyce**.

Table 1 contains a complete list of devices for **Xyce** Release 6.1.

Table 1: Devices Supported by Xyce

Device	Comments
Capacitor	Age-aware, semiconductor
Inductor	Nonlinear mutual inductor (see below)
Nonlinear Mutual Inductor	Sandia Core model (not fully PSpice compatible) Stability improvements
Resistor (Level 1)	Semiconductor
Resistor (Level 2)	Thermal Resistor
Diode (Level 1)	
Diode (Level 2)	Addition of PSpice enhancements

Table 1: Devices Supported by Xyce

Device	Comments
Independent Voltage Source (VSRC)	
Independent Current Source (ISRC)	
Voltage Controlled Voltage Source (VCVS)	
Voltage Controlled Current Source (VCCS)	
Current Controlled Voltage Source (CCVS)	
Current Controlled Current Source (CCCS)	
Nonlinear Dependent Source (B Source)	
Bipolar Junction Transistor (BJT) (Level 1)	
Bipolar Junction Transistor (BJT) (Level 10)	Vertical Bipolar Intercompany (VBIC) model, version 1.2
Bipolar Junction Transistor (BJT) (Level 23)	FBH (Ferdinand-Braun-Institut für Höchstfrequenztechnik) HBT model, version 2.1
Junction Field Effect Transistor (JFET) (Level 1)	SPICE-compatible JFET model
Junction Field Effect Transistor (JFET) (Level 2)	Shockley JFET model
MESFET	
MOSFET (Level 1)	
MOSFET (Level 2)	SPICE level 2 MOSFET
MOSFET (Level 3)	
MOSFET (Level 6)	SPICE level 6 MOSFET
MOSFET (Level 9)	BSIM3 model
MOSFET (Level 10)	BSIM SOI model
MOSFET (Level 14)	BSIM4 model
MOSFET (Level 18)	VDMOS general model
MOSFET (Level 103)	PSP model
MOSFET (Level 107)	BSIM-CMG version 107.0.0 **New!**
MOSFET (Level 301)	EKV model version 3.0.1
Transmission Line	Lossless
Transmission Line	Lossy

Table 1: Devices Supported by Xyce

Device	Comments
Controlled Switch (S,W) (VSWITCH/ISWITCH)	Voltage or current controlled
Generic Switch (SW)	Controlled by an expression
PDE Devices (Level 1)	one-dimensional
PDE Devices (Level 2)	two-dimensional
Digital (Level 1)	Behavioral Digital
ACC	Accelerated mass device, used for simulation of electromechanical and magnetically-driven machines

Robustness Improvements

- Solver defaults have been modified for increased robustness of parallel runs with a minimum of user-specified options.

New Devices

- BSIM-CMG version 107.0.0.
- Digital buffer model (BUF).
- Multi-input ($\geq$2) AND, NAND, OR and NOR gates.

Enhanced Solver Stability, Performance and Features

- The AC, harmonic balance, and MPDE analysis features have been improved, and are now available in parallel builds of **Xyce**.
- Xyce has been updated to use Trilinos 11.6, which has resulted in significant performance improvements.

Defects of Release 6.0.1 Fixed in this Release

Table 2: Fixed Defects. Note that we have two different Bugzilla systems for Sandia users. SON, which is on the open network, and SRN, which is on the restricted network.

Defect	Description
474-SON: Errors in HB time domain output and during HB solve	The time domain output is incorrectly shifted when a nonzero startupperiods is specified. During HB solve, the related vector and matrix evaluations and voltage limiting handling have errors. ***Solution***: These errors were fixed.
475-SON: Voltage limiting information is not assembled consistently for high orders in the Trap and Gear time integrators	The voltage limiting information obtained from device package has F and Q parts. They need to be assembled in the similar way to how the Jacobian is assembled. For the Trap and Gear time integrators, it is consistent for the first order. Higher order needs to be fixed. ***Solution***: This was fixed.
387-SON: Failure to find <file> in .lib <file> causes Xyce to emit confusing error message	If a `.LIB` statement is given a file argument, but the file cannot be found, an error in input processing causes Xyce to exit with an error stating that no analysis statement was specified instead of a more meaningful "cannot find file" error. ***Solution***: The code now emits a "Could not find include file . . . " when no filename is present.
401-SON: Frequency-domain-specific voltage drop specifiers produce incorrect output	The VR, VI, VM, VP and VDB output specifiers for printing real and imaginary parts, magnitude, phase, or magnitude in DB of voltages in frequency domain do not work if given two nodes, but Xyce will not emit any warning or error messages. ***Solution***: This was fixed in the parser.
412-SON: .measure ... when ... cross= does not seem to work as advertised	The code to process the .measure ... when... cross=... construct is not working correctly. ***Solution***: The code was only checking for zero crossings. This was corrected.
413-SON: Parser ignores instance parameters if user incorrectly places them before model name	If instance parameters (e.g. length or width of a MOSFET device) are placed before the model name on an instance line, Xyce simply ignores them without emitting a syntax error message. ***Solution***: An error message is now output when the above conditions occur.
440-SON: Measure functions should support lead currents and expressions.	The measure function code was only parsing out voltage nodes and current values for voltage sources. Better solution variable parsing was needed in the measure support code. ***Solution***: Parsing was improved in the measure code to support lead currents and expressions in measure statements.

Table 2: Fixed Defects. Note that we have two different Bugzilla systems for Sandia Users. SON, which is on the open network, and SRN, which is on the restricted network.

Defect	Description
460-SON: Support equaiton evaluation measure.	This was a feature request to support the general evaluation of an equation within the same context as the measure functions. ***Solution***: The measure function "EQN" has been added to support the evaluation of expressions during the simulation.
461-SON: A *Version* string should not be part of the RAW file format header.	***Solution***: *Version* has been removed from Xyce's default RAW header. The *Version* string can be re-enabled by setting the following option in the netlist: `.options output outputversioninrawfile=true`
481-SON: Measure functions should have a settable default.	The default value for a measure function was zero, but if the measure function was unable to calculate a value it was unclear to the user if this was the case. ***Solution***: The measure functions now have a default value of -1. This is settable in the measure function definition with the paramter `default_val` as in `default_val=-100`.
772-SRN: Infinite-slope transitions in B-sources causes "time step too small" errors	The nonlinear dependent source ("B-source") allows the user to specify expressions that could have infinite-slope transitions, such as `Bcrtl OUTA 0 V={ IF( (V(IN) > 3.5), 5, 0 ) }` This can lead to "timestep too small" errors when **Xyce** reaches the transition point. Infinite-slope transitions in expressions dependent only on the `time` variable are a special case, because **Xyce** can detect that they are going to happen in the future and set a "breakpoint" to capture them. ***Solution***: Infinite-slope transitions depending on other solution variables cannot be predicted in advance, and cause the time integrator to scale back the timestep repeatedly in an attempt to capture the feature until the timestep is too small to continue. Since this kind of situation is non-physical, and any fix would likely be not robust, this bug was closed as a WONTFIX.
855-SRN: AC and HB analysis do not work in parallel	Analysis types that require use of block linear algebra classes in Trilinos/Epetra do not work in parallel. The analysis types that use block linear algebra classes are AC and HB. ***Solution***: This was fixed with recent enhancements to the solvers.
1918-SRN: The FREQ measure function should support the From/To time qualifiers.	***Solution***: The FREQ measure function now supports the From/To qualifiers to limit the time window where the frequency is measured.

Known Defects and Workarounds

Table 3: Known Defects and Workarounds.

Defect	Description
27-SON: Fix handling of .options parameters	When specifying .options for a particular package, what gets applied as the non-specified default options might change.
37-SON: Connectivity checking is broken for devices with more than 10 leads	The diagnostic code used by the **Xyce** setup that checks circuit topology for basic errors such as a node having no DC path to ground or a node being connected to only one device has a bug in it that causes the code to emit a cryptic error message, after which the code will exit. This error has so far only been seen when a user has attempted to connect a large number of inductors together using multiple mutual inductor lines. The maximum number of non-ground leads that can be used without confusing this piece of code is 10. If your circuit has that type of large, highly-connected mutual inductor and the code exits with an error message, this bug may be the source of the problem. The error message now includes a recommendation to use the workaround below. ***Workaround***: Disable connectivity checking by adding the line `.OPTIONS TOPOLOGY CHECK_CONNECTIVITY=0` to your netlist. This will disable the check for the basic errors such as floating nodes and improperly connected devices, but will allow the netlist to run with a highly-connected mutual inductor.
195-SON: Restart files are not produced if initial_interval is not specified	The users' guide states that if `.OPTIONS RESTART` is specified without an `INITIAL_INTERVAL` that the restart file will be saved at every time step. This is not working, and if no initial interval is specified, no restart is saved at all. ***Workaround***: Always specify a suitable initial interval when requesting restart.
244-SON: The defaults in Xyce bsimsoi do not match defaults in spice3 bsimsoi	Some of the default parameters used in the BSIM3 SOI (level 10) MOSFET do not match SPICE3F5's defaults, leading to observed differences in behavior if a model card does not specify these parameters. ***Workaround***: The parameters at issue appear all to be parameters that have *computed* rather than hard-coded defaults. These include `CGDO`, `CGSO`, `K1`, `K2`, `XJ`. It is best to use extracted values for these parameters rather than relying on the defaults until this bug is fixed.
247-SON: Expressions don't work on .options lines	Expressions enclosed in braces ({ }) are handled specially throughout Xyce, and may only be used in certain contexts such as in device model or instance parameters or on `.PRINT` lines.

Table 3: Known Defects and Workarounds.

Defect	Description
250-SON: NODESET in xyce is not equivalent to NODESET in SPICE	As currently implemented, .NODESET applies the initial conditions given throughout a full nonlinear solve for the operating point, then uses the result as an initial guess for a second nonlinear solve with no constraints. This is not the same as SPICE, which merely applies the given initial conditions to a single nonlinear solve for the first two iterations, then lets the problem converge with no further constraints. This can lead to Xyce's .NODESET failing where the same netlist in SPICE might not, if the initial conditions are such that a full nonlinear solve cannot converge with those constraints in place. There is no workaround.
365-SON: Missing Default Instance Length Parameter in Resistor Model Parameters	The resistor allows a semiconductor resistor formulation given a sheet resistance, length, and width, but no reasonable default is given for the length. A default width parameter is provided through the model card, and is used if no width is specified on the instance line. ***Workaround***: Always specify a length on the instance line for any resistor for which the semiconductor resistor syntax is used (see the **Xyce** Reference Guide for usage).
384-SON: Errors compiling with Bison 3.0	Bison 3.0's C++ parser has changed significantly and thus Bison version 3.0 cannot be used to build Xyce's reaction parser. ***Workaround***: Use only Bison versions between 2.3 and 2.7 if reaction parsing is desired. Disable building of the reaction parser by configuring with `--disable-reaction_parser` if unable to provide a suitable version of Bison.
427-SON: Certain subcircuit nodes cannot be accessed in expressions	Xyce allows nodes within a subcircuit to be referenced in expressions and on .print lines using the syntax "<subcircuit instance name>:<nodename>". On .print lines this works even if the node named is one that is on the subcircuit declaration line (i.e., dummy argument nodes, which are just aliases for names in the calling circuit context). This does not work inside expressions. ***Workaround***: Use the name of the node at the higher level of the circuit instead of the dummy argument names.
429-SON: Support AC print-line accessors for magnitude, phase, etc. of branch currents	SPICE is capable of outputting the magnitude, phase, real and imaginary parts, and magnitude in dB for currents. Xyce currently supports these accessors only for voltage nodes.
438-SON: .step output for HB in Standard (.prn) format does not reset Index for each step or otherwise mark new step data	It is difficult to separate the results, in an .HB.FD.prn file, for each step because the Index column is monotonic instead of resetting to zero at the beginning of each step. In transient or DC analysis runs, .step (wrapping the other analysis) causes the Index column to reset to zero for every step.

Table 3: Known Defects and Workarounds.

Defect	Description

439-SON: Interpolation does not work well with TRAP and GEAR	The problem occurs in certain situations when the output is interpolated, e.g., in a circuit with
	`.tran 50us 15ms`
	`.options output initial_interval=150us`
	`.print tran i(V2)`
	as part of the output directions could show the problem. The problem does go away when using BDF.
	Workaround: Use BDF, or do not specify an `initial_interval`.

442-SON: lead currents are not correctly output for AC and HB analysis	The lead currents were not calculated in HB and AC analyses. The output will be all zeros for the lead currents in HB and AC analyses.

466-SON: .RESULT output doesn't recognize .param parameters	As an example, consider:
	`.param par1=1.0`
	`.param par2=2.0`
	`.result {par1/par2}`
	The code will run for a while, and then when the first .RESULT output is processed (at the end of the first .STEP iteration) it will exit with an error, claiming that par1 and par2 are not recognized variables.
	This .print line does work though:
	`.print tran {par1/par2}`

468-SON: It should be legal to have two model cards with the same model name, but different model types.	SPICE3F5 and ngspice only require that model cards of the same type have unique model names. They accept model cards of different types with the same name. Xyce requires that all model card names be unique.
469-SON: Belos memory consumption on FreeBSD and excessive CPU on other platforms	
477-SON: Internal variables are not accessible inside expressions	An internal variable, accessed via n(), is not properly resolved in an expression. As an example, this fails parsing. `.print tran {n(y%neuron%neuron1_V1)}` ***Workaround***: If the braces are removed then the netlist runs without problems.

Table 3: Known Defects and Workarounds.

Defect	Description

483-SON: ASCII rawfile output does not work with .dc without the -r flag	As an example, if the netlist:
	
	`VAB 1 0 5`
	`.dc VAB 0 5 1`
	`.print dc format=raw v(1)`
	`.end`
	
	is run using "Xyce -a <filename>", the resulting file is in the binary format, not ASCII.
	Workaround: Using both the "-r <rawfile>" and "-a" flags will result in <rawfile> being in the ASCII format.

484-SON: putting resistor parameter on .PRINT line, with homotopy simulation causes segfault	For example,
	`.DC RC:R 2K 2K 2K`
	`.PRINT DC format=tecplot RC:R V(4)`
	`+ I(VMON1) I(VMON2) V(1) V(2)`
	causes a segfault.

1595-SRN: Xyce won't allow access to inductors within subcircuits for mutual inductors external to subcircuits	It is not possible to have a mutual inductor outside of a subcircuit couple to inductors in a subcircuit. ***Workaround***: Put all inductors and mutual inductance lines that couple to them together at the same level of circuit hierarchy.
1922-SRN: Xyce nonlinear core model defaults seriously underestimate hysteresis effects for small signals	The magnetic core model has some parameter defaults that effectively turn off magnetic effects when small-amplitude driving signals are applied. This can show up when attempting to generate B-H loops for magnetic cores using standard test harnesses. A robust solution to this issue is being sought. ***Workaround***: Set the model parameter CONSTDELVSCALING to false. This will activate dynamic voltage scaling withing the mutual inductor to better capture the magentic effects for small signals. If that failes, then one can also revert to the default behavior (i.e. CONSTDELVSCALING=true) and adjust the model paramter DELVSCALING manually to work with a given circuit. Make DELVSCALING as large as possile while still getting circuit convergence (e.g. 1e2 to 1e6 may work).

Incompatibilities With Other Circuit Simulators

Table 4: Incompatibilities with other circuit simulators.

Issue	Comment
`.DC` sweep output.	The `.DC` sweep calculation does not automatically output the sweep variable. Only variables explicitly listed on the `.PRINT` line are output.
Pulsed source rise time of zero.	A requested pulsed source rise/fall time of zero really is zero in **Xyce**. In other simulators, requesting a zero rise/fall time causes them to use the printing interval found on the `.TRAN` line.
Mutual Inductor Model.	Not the same as PSpice. This is a Jiles-Atherton non-linear model developed at Sandia. However, it is compatible with the PSpice parameter set.
`.PRINT` line shorthand.	Output variables have to be specified as V(node) or I(source). Specifying the node alone will not work.
MOSFET levels.	In **Xyce** the MOSFET levels are not the same. A BSIM3 is MOSEFET level 9. Other simulators have different levels for the BSIM3.
BSIM SOI v3.2 level.	In **Xyce** the BSIM SOI (v3.2) is MOSFET level 10. Other simulators have different levels for the BSIM SOI.
BSIM4 level.	In **Xyce** the BSIM4 is MOSFET levels 14 and 54. Other simulators have different levels for the BSIM4.
The '%' symbol is not a valid in a device or node name.	The '%' symbol has a special meaning in **Xyce**, and therefore cannot be used in other cases.
Interactive mode.	**Xyce** does not have an interactive mode.
Syntax for `.STEP` is different.	The manner of specifying a model parameter to be swept is slightly different than in some other simulators. See the **Xyce** Users' and Reference Guides for details.
Switch is not the same as SPICE.	The **Xyce** switches are not compatible with the simple switch implementation in SPICE3F5. The switch in **Xyce** smoothly transitions between the ON and OFF resistances over a small range between the ON and OFF values of the control signal (voltage, current, or control expression). See the **Xyce** Reference Guide for the precise equations that are used to compute the switch resistance from the control signal values. The SPICE3F5 switch has a single switching threshold voltage or current, and RON is used above threshold while ROFF is used below threshold. **Xyce**'s switch is considerably less likely to cause transient simulation failures. Results similar to SPICE3F5 can be obtained by setting VON and VOFF to the same threshold value, but this is not a recommended practice.

Table 4: Incompatibilities with other circuit simulators.

Issue	Comment
U digital device syntax differs from PSpice U Device.	**Xyce** 6.1 contains a new U device with the goal of improved compatibility between the instance lines of **Xyce** and PSpice digital devices. Known incompatibilities are that PSpice uses separate model cards for IO and timing characteristics, while **Xyce** combines the IO and timing characteristics in one model card. The PSpice and **Xyce** model cards also have different parameters.
U device is different from SPICE3F5 U device.	The **Xyce** U devices are digital behavioral models, which are intended to be similar to the PSpice U digital devices. The SPICE3F5 U device is a uniform RC transmission line model.

Important Changes to **Xyce** Usage Since the Release 6.0.1.

Table 5 lists some usage changes for **Xyce**.

Table 5: Changes to netlist specification since the last release.

Issue	Comment
Added U device	Digital behavioral models. See the **Xyce** Reference Guide for more details on command syntax.
Deprecated Y Digital Devices	Syntax for Y digital devices is unchanged but the Y digital devices will be deprecated in a future **Xyce** release. See the **Xyce** Reference Guide for details on converting Y digital devices to the new U device syntax. The other Y devices (e.g., YPDE) are unaffected by this change.

Trademarks

The information herein is subject to change without notice.

Xyce™ Electronic Simulator and ***Xyce***™ are trademarks of Sandia Corporation.

Orcad, Orcad Capture, PSpice and Probe are registered trademarks of Cadence Design Systems, Inc.

Microsoft, Windows and Windows 7 are registered trademarks of Microsoft Corporation.

Medici, DaVinci and Taurus are registered trademarks of Synopsys Corporation.

Amtec and TecPlot are trademarks of Amtec Engineering, Inc.

Xyce's expression library is based on that inside Spice 3F5 developed by the EECS Department at the University of California.

The EKV3 MOSFET model was developed by the EKV Team of the Electronics Laboratory-TUC of the Technical University of Crete.

All other trademarks are property of their respective owners.

Contacts

Bug Reports (Sandia only)	`http://joseki.sandia.gov/bugzilla` `http://charleston.sandia.gov/bugzilla`
World Wide Web	`http://xyce.sandia.gov` `http://charleston.sandia.gov/xyce` (Sandia only)
Email	`xyce@sandia.gov` (outside Sandia) `xyce-sandia@sandia.gov` (Sandia only)

DISTRIBUTION:

1 MS 0899 Technical Library, 9536 (electronic copy)


Sandia National Laboratories