Xyce™ Parallel Electronic Simulator Version 7.1 Release Notes

Sandia National Laboratories

May 27, 2020

The Xyce™ Parallel Electronic Simulator has been written to support the simulation needs of Sandia National Laboratories' electrical designers. Xyce™ is a SPICE-compatible simulator with the ability to solve extremely large circuit problems on large-scale parallel computing platforms, but also includes support for most popular parallel and serial computers.

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.

Contents

New Features and Enhancements

XDM

- XDM works with Python 3.x
- Translation of HSPICE .DATA command has been added.
- Better translation of HSPICE .MEASURE commands when compatible with Xyce.
- The translation of P-elements from HSPICE to Xyce is more complete.

New Devices and Device Model Improvements

- The BSIM SOI version 4.6.1 has been added as the level 70 MOSFET. At this time the implementation is limited to the four-node version without exposed body nodes. It has been observed that the gate capacitor model may have compatibility issues with PDKs that are expecting versions 4.5 or earlier. For this reason the older BSIM-SOI 4.5.0 has also been added as the level 70450 MOSFET.
- The BSIM6 model now supports a multiplicity factor, M, on its instance line.
- Models derived from Verilog-A sources (PSP, MEXTRAM, BSIM6, BSIM-CMG, BSIM SOI 4.6.1, HICUM, MVS, and JUNCAP200) now support output variables such as transconductances, capacitances, and conductances. A new table of available output variables and how to access them has been added to the Reference Guide section for each of these models.
- The L_UTSOI MOSFET model version 102.4.0 has been added as the level 10240 MOSFET. Due to licensing restrictions, this model is not available in open source builds or in source form.

Interface Improvements

- The AVG, DERIV, FIND-WHEN, INTEG, RMS and WHEN measure types are now supported for .AC and .DC analyses.
- The ERR, ERR1, ERR2 and FIND-AT measure types are now supported for all three measure modes (AC, DC and TRAN).
- For the Mixed Signal interface, methods were added to allow the user to query for the names of all of the devices in the netlist, and for the parameter values of any device in the netlist.

Important Announcements

- The Xyce project is no longer providing binaries for RHEL6.
- The model interpolation technique described in the Xyce Reference Guide in section 2.1.18 has been marked as deprecated, and will be removed in a future release of Xyce.
- It has been determined that some distributions of Linux have broken builds of OpenMPI in their package repositories. Building Xyce from source code in parallel with these OpenMPI installs will result in a version of Xyce that may crash on some problems. This is not a bug in Xyce, but a packaging error of the OpenMPI package on those operating systems. Please see commentary in the "Known Defects" section of these release notes under bug number "967-SON".

- Xyce has deprecated the default conversion of quoted-string file names to a table of x,y pairs of data. The old convention of `PARAMETER="file.dat"` which worked in some model statements and in behavioral sources will now generate a warning in the Xyce output. The correct way to specify a file of data for a parameter is to use the new `tablefile` keyword as in `PARAMETER=tablefile("file.dat")`. While this release of Xyce will accept both the old and new syntax, the double quote technique will be removed in a future release, after which Xyce will only accept the syntax of `PARAMETER=tablefile("file.dat")`. Additionally, a new syntax of `PARAMETER=string("string value")` has been introduced to specify parameters that are pure strings. This will be deprecated in a future release and the simpler syntax of `PARAMETER="string value"` will be used to specify string valued parameters.

- The "Xygra" device, which was written as a special-purpose coupling mechanism to ALEGRA but which has occasionally been used for other coupling problems, has been marked as deprecated. The new, more flexible "General External" device was created to take its place, and has supplanted the use of Xygra in ALEGRA. The Xygra device and the API that enables it may be removed without notice in a future release of Xyce. If your code has been using the Xygra capability to couple to Xyce, you must replace your usage with the new capability. The "General External" coupling mechanism is documented thoroughly in an application note available on the Xyce web site.

Defects Fixed in this Release

Table 1: 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
807-SON: BSIM4 convergence problems with non-zero rgatemod value	A bug in the BSIM4 led to convergence problems (e.g., the Xyce simulation fails part way through and says that the "time step is too small") when the `rgatemod` parameter is non-zero.
1299-SON: Xyce incorrectly outputs multiple unwanted solution results when 'tstart' given on tran line and .options output also given	When a non-zero third parameter ("TSTART") was given to a `.TRAN` analysis statement and a `.OPTIONS OUTPUT` line was also used to specify detailed printing intervals, Xyce would output unwanted information for times prior to `TSTART`.
203-SON: Add BSIM 4 SOI model to Xyce	The BSIM SOI version 4.6.1 has been added as the level 70 MOSFET.
1296-SON: Add multiplicity factor to BSIM 6	The BSIM 6 in Xyce prior to this release did not support a multiplicity factor ("M" instance parameter). Support for multiplicity in the BSIM 6 has been added in this release.
788-SON: Support output variables from Verilog-A models	The Verilog-A Language Reference Manual (LRM) states that module scoped variables that have "desc" or "units" attributes should be considered output variables, and users should have access to these values during simulation. Xyce now supports this usage, and Xyce/ADMS will generate appropriate code so that these output variables may be printed with the "N()" notation on `.PRINT` lines.
410-SON: Implement ddx() in Xyce/ADMS Verilog-A compiler	Xyce/ADMS did not support use of the Verilog-A construct "ddx()", which provides symbolic differentiation capabilities. ddx() is now supported with minor limitations that impact no published Verilog-A models we have encountered. See the Xyce/ADMS Users' Guide for more details.

1298-SON: Bug in handling FREQ special variable for AC measures	There was an error in handling the `FREQ` special variable in `.MEASURE` statements that can be illustrated with this simple netlist fragment:
	
	`.ac dec 5 100Hz 1e6`
	`.print AC FREQ EQNFREQ vm(b)`
	`.MEASURE AC EQNFREQ EQN {FREQ}`
	
	The value of `FREQ`, generated by the `.PRINT AC` line, in the output file `<netlistName>.FD.prn` would be correct. However, the value reported for the AC measure `EQNFREQ`, in that output file, would be the value from the previous `AC` sweep value. So, in this example, the final value reported for `EQNFREQ` in the file `<netlistName>.ma0` would be 6.31e+05 rather than 1e+06.

Table 1: 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
1284-SON: Rewrite Xyce/ADMS to generate code that does not use Sacado at all	With release 6.11 of Xyce, Xyce/ADMS was rewritten to require Sacado only for the functions that are used during sensitivity analysis, resulting in a dramatic performance improvement for all analyses other than sensitivity. This effort has concluded in release 7.1 by implementing sensitivity code without Sacado as well. This change has little impact on performance of sensitivity analysis, but does significantly reduce memory requirements at compile time.
1283-SON: Make RMS measure results more compliant with the HSPICE method	The Xyce method for calculating the results of `RMS` measures, for `AC`, `DC` and `TRAN` measure modes, now matches the HSPICE approach of using the "the square root of the area under the out_var curve, divided by the period of interest".
1275-SON: Bug with handling SIMPLE and PAUSE breakpoints that occur at the same time	Xyce supports two types of "breakpoints" – namely SIMPLE and PAUSE breakpoints. The SIMPLE breakpoints are, for example, set based on the times of the slope discontinuities in any PULSE and PWL sources in the netlist. The PAUSE breakpoints are set, for example, by the simulation end time and any `simulateUntil(requested_time)` calls used by the Mixed Signal Interface. In previous versions of Xyce, if a SIMPLE and PAUSE breakpoint occurred at the same time then the PAUSE breakpoint would be ignored and the `simulateUntil()` calls would work incorrectly.

1273-SON: Bug fixes for DERIV, FIND-WHEN and WHEN measures for all analysis modes	Several issues with the `DERIV`, `FIND-WHEN` and `WHEN` measures, for `AC`, `DC` and `TRAN` measure modes, were addressed for this release. First, the right-hand side of the equality in the `WHEN` clause can now be an expression. An example is:
	
	`.MEASURE DC whenExample1 when v(2)={v(1)+1}`
	
	The second issue was that the interpolation algorithm for determining the time (or frequency or DC sweep value) at which the `WHEN` clause was satisifed could be inaccurate if the `WHEN` clause used varying quantities. An example is:
	
	`.MEASURE AC whenExample2 when vr(1)=vi(1)`
	
	The third issue was that, especially for `TRAN` measures, the `WHEN` clause could miss some of the crossings if that clause used varying quantities. All of the previous discussion applies to `DERIV` and `FIND-WHEN` measures.

1271-SON: Incorrect error handling for .PRINT AC when .LIN is used with .STEP	An attempt to print out an S-,Y- or Z-parameter value via a .PRINT AC line would cause a parsing error when .LIN was used with .STEP.
1292-SON, Gitlab Issue #6: Inductor coupling (mutual inductor) code does not respect inductor initial condition specification	Linear and non-linear mutual inductors were ignoring the initial conditions set on component inductors. This was a code deficiency as the initial conditions were not passed into the mutual inductor device.

Table 1: 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
1287-SON, Github Issue #6: Ill-formed .IC line leads to segfaults on some Linuxen	Empty .IC/.DCVOLT/.NODESET lines were causing Xyce to seg fault on certain platforms. A warning is now emitted for empty .IC/.DCVOLT/.NODESET lines and the line is effectively ignored.
1264-SON: .dc specification including a temperature sweep can cause many unnecessary processParams calls	A flaw in Xyce design resulted in a lot of superfluous parameter handling, which caused Xyce to be inefficient in some use cases.

Interface Changes in this Release

Table 2: Changes to netlist specification since the last release.

Change	Detail
Mixed Signal Interface behavior when a "backwards or null time step" is requested.	Previously, for the Mixed Signal Interface, if the `requestedUntilTime` in a `simulateUntil()` call was less than the current simulation time then the Xyce simulation would run to completion. If the `requestedUntilTime` was equal to the current simulation time then the simulation would get stuck in a "time-step too small" error loop. The interface change is that both of these cases are now simulation errors that will cause the Xyce simulation to abort.
Removed support for RISE, CROSS and FALL qualifiers for DUTY measure	This change makes all of the duty-cycle calculation measures (`DUTY`, `FREQ`, `OFF_TIME` and `ON_TIME`) have similar behavior. The `FROM`, `TO` and `TD` qualifiers are still supported for all four measure types.

Handling of measures with duplicate names	In previous versions of Xyce, measures with duplicate names would both be evaluated correctly. However, this could cause ambiguity if that name was also used as part of an expression such as in an `EQN` measure. An example is this netlist fragment:
	
	`.MEASURE TRAN M1 MIN V(1)`
	`.MEASURE TRAN M1 MAX V(1)`
	`.MEASURE TRAN EQNM1 EQN M1`
	
	To remove this ambiguity, Xyce will now use the last `M1` measure definition, which is the `MAX` measure in this example, and issue a warning message about (and discard) any previous `M1` definitions. This change applies to all measure modes (`AC`, `DC` and `TRAN`).

Known Defects and Workarounds

Table 3: Known Defects and Workarounds.

Defect	Description

1262-SON: Duplicate L device definitions are not a parsing error when one of the duplicate L devices is part of a K device	As an example, this netlist will not produce a parsing error. Instead, the first L1 definition will be used in the K1 device definition.
	
	`* parsing fails to detect duplicate L1 devices`
	`V1 1 0 SIN(0 1 1KHz)`
	`L1 1 2 1e-3`
	`R1 2 0 1`
	`C1 2 0 1e-9`
	`* mutual inductor definition, with duplicate L1 device`
	`L1 4 0 1e-6`
	`L2 3 0 1mH`
	`K1 L1 L2 0.75`
	`.TRAN 0 1ms`
	`.PRINT TRAN V(1) v(2)`
	`.END`
	
	Workaround: There is none.

1241-SON: Expression library parsing bottleneck on large expressions	It has been determined that the expression library in Xyce can be the source of a severe parsing bottleneck when expressions are large and complex. Expressions of this sort show up most often when parsing large PDKs with complex use of the `.FUNC` feature, and when using the "tablefile" feature to import a large file of time/voltage pairs for use in a `B` source. ***Workaround***: There is currently no workaround for the issue of complex PDK function use, and the team is working on fixing this issue by redesigning the way Xyce handles expressions with user defined functions. For the "tablefile" issue, one should avoid using `B` sources with "tablefile" to read in large tables, and instead use the "PWL FILE" option of the `V` source, which does not have this parsing issue.
1085-SON: Expression library mishandles `.FUNC` definitions of functions that begin with "I" and are two characters long	Xyce's expression library assumes that all terms of the form "`Ix(<arguments>)`" are lead current expressions, where "x" is either a lead designator such as "D", "G" , or "S" for a MOSFET or "C", "B", "E" for a BJT, or a digit indicating the pin number of the device associated with the lead. This assumption makes it impossible for users to define a function with a two-character name starts with "I". Unfortunately, the parser does not warn of this problem should a user define such a function, and the first indication of something being wrong is an unhelpful error message about an "undefined parameter or function" where the problematic function is used. ***Workaround***: Do not use function names of two character length that begin with the letter "I". If you are making use of a vendor-supplied library that includes definitions of functions such as "I0", you will have to modify the library to change the function name and all the instances of its use.

Table 3: Known Defects and Workarounds.

Defect	Description
1037-SON: The use of non-constant values in .PARAM statements may lead to unexpected results	This netlist line (`.PARAM PA = {TEMP}`) is forbidden in Xyce since the special variable `TEMP` is not constant. However, that netlist line will not produce a Xyce parsing error, and the value of `PA` in the simulation may then be set to zero in some contexts. ***Workaround***: Non-constant values should only be used in `.GLOBAL PARAM` statements in Xyce. This restriction may be different than in other Spice-like simulators.
1031-SON: .OP output is incomplete in parallel	When Xyce is run in parallel, the `.OP` output may be incomplete. ***Workaround***: One workaround is to run the netlist in serial. Another one is to use these command line options: `-per-processor -l output`. In that case, the per-processor log files will have the `.OP` information for the devices that were instantiated on each processor.
1009-SON: Transient adjoint sensitivities don't work with `.STEP`	Transient adjoint sensitivities require backward integrations to be performed after the primary transient forward integration. Doing this properly requires information to be stored during the forward solve, and for certain bookkeeping to be performed. Currently, these extra operations to support transient adjoints are not properly set up for `.STEP` analysis. ***Workaround***: None
1006-SON: SDT (expression library time integration) derivatives are not supported, so SDT can't be used for sensitivity analysis objective functions	SDT is a function supported by the Xyce expression library to compute numerical time integration. When this function is used, the expression library does not produce correct derivatives. This impacts Jacobian matrix entries, when SDT is used with a Bsrc, and it also impacts sensitivity analysis, when SDT is used in an objective function. For the former case, this can result in a lack of robustness for circuits that contain SDT-Bsrc devices. For the latter case, the objective function will simply be incorrect. ***Workaround***: None

1004-SON: Ill-defined .STEP behavior for "default parameters" for transient sources (SIN, EXP, PWL, PULSE and SFFM)	If, for example, these netlist lines are used in a transient (`.TRAN`) simulation:
	`V1 1 0 SIN(0 1 1)`
	`.STEP V1 1 2 1`
	then Xyce will run the simulation without warnings or errors, but no instance parameter of source `V1` will be stepped.
	Workaround: Explicitly use the desired stepped parameter (e.g., `V0`) on the `.STEP` line. For example, `.STEP V1:V0 1 2 1` would work correctly.

991-SON: Non-physical BH Loops in non-linear mutual inductor	Nonlinear mutual inductors that have high coupling coefficients (i.e. model parameter `ALPHA` over 1.0e-4) and low loss characteristics (i.e. zero `GAP`) can produce B-H loops with nonphysical hysteresis. ***Workaround***: Lower `ALPHA` values or larger `GAP` values can ameliorate this issue, but the root cause is still under investigation.

Table 3: Known Defects and Workarounds.

Defect	Description
800-SON: Use of global parameters in expressions on .MEASURE lines will yield incorrect results	The use of global parameters in expressions on .MEASURE lines is not allowed, as documented in the Xyce Reference Guide. However, instead of producing a parsing error the measure statement will be evaluated with the specified qualifier value (e.g., `FROM`) being left at its default value. ***Workaround***: None, other than not doing this.
970-SON: Some devices do not work in frequency-domain analysis	Devices that may be expected to work in AC or HB analysis do not at this time. For AC this includes, but is not limited to, the lossy transmission line (LTRA) and lossless transmission line (TRA). For HB, the transmission lines do work but the nonlinear dependent sources (B source and nonlinear E, F, G, or H source) do not. ***Workaround***: The LTRA and TRA models will need to be replaced with lumped transmission line models (YTRANSLINE) for AC analysis. There is not yet a workaround for the B source in harmonic balance.

Table 3: Known Defects and Workarounds.

Defect	Description
967-SON: Zoltan segmentation fault with OpenMPI 2.1.x and 3.0.0 on some systems	It has been observed that when Xyce and Trilinos are built with OpenMPI 2.1.x or 3.0.0 on certain unsupported operating systems, a small number of test cases in the regression suite crash with a segmentation fault inside the Zoltan library. The Xyce team has determined that this is not a bug in either Xyce or Zoltan, but is instead due to some pre-packaged OpenMPI binaries on some operating systems having been built with an inappropriate option. This option, "–enable-heterogeneous" is explicitly documented in OpenMPI documentation as broken and unusable since 2013, but some package managers have OpenMPI binaries built with this option explicitly enabled. Turning on this option causes the resulting OpenMPI build to perform certain communication operations in a way that does not adhere to the MPI standard. There is nothing that can be done in Xyce or Zoltan to fix this issue — it is entirely a bug in the OpenMPI library as built on that system. A new test case has been added to the Xyce test suite in order to detect this problem. The test is "MPI_Test/bug_967", and it will be run whenever the test suite is invoked with the "+parallel" tag as described in the documentation for the test suite at `https://xyce.sandia.gov/documentation/RunningTheTests.html`. If this test fails, your system has a broken OpenMPI build that cannot be used with Xyce. At the time of this writing, this issue is present in Ubuntu Linux versions 17.10 and later, and there is an open bug report for it at `https://bugs.launchpad.net/ubuntu/+source/openmpi/+bug/1731938`. The issue may be present in other distros of Linux that are derived from Debian (as is Ubuntu), but we cannot confirm this. ***Workaround***: The only workaround for this problem is to build OpenMPI from source yourself, and not to include "–enable-heterogeneous" in its configure options. You should also post a bug report in your operating system's issue tracker requesting that they rebuild their OpenMPI binaries without the "–enable-heterogeneous" option. If you are using Ubuntu, you should register with that issue tracking system and add yourself to the list of people it affects in the existing bug report (doing so increases the "heat" of the bug, which may increase the likelihood of it being fixed).
964-SON: Compatibility of .PRINT TRANADJOINT with .STEP	The use of `.PRINT TRANADJOINT` is not compatible with `.STEP`. The resultant Xyce output will not be correct. ***Workaround***: There is none.

Table 3: Known Defects and Workarounds.

Defect	Description
932-SON: Analysis lines do not support expressions for their operating parameters	The Xyce parser and analysis handlers do not yet support the use of expressions on netlist analysis lines such as `.TRAN`. The parameters of these analysis lines (such as stop time for `.TRAN` or fundamental frequency for `.HB`) may only be expressed as literal numbers. ***Workaround***: There is no workaround internal to Xyce. Use of an external netlist preprocessor would be required.
883-SON .PREPROCESS REPLACEGROUND does not work on nodes referenced in expressions	The `.PREPROCESS REPLACEGROUND` feature does not replace ground synonyms if they appear in B source expressions. ***Workaround***: Do not use ground synonyms (`GND`, `GROUND`, etc.) in expressions. Use a literal "0" when referring to the ground node in expressions.
812-SON: Undocumented limitations on, and bugs with, parameter and global parameter names	Based on external customer input and pre-release testing, there are some bugs and undocumented limitations on parameter and global parameter names in Xyce. Parameters and global parameters should start with a letter, rather than with a number or "special" character like #. In addition, the use of a single character V as a global parameter name can result in either netlist parsing failures or incorrect results from `.PRINT` lines.
794-SON: Bug in TABLE Form of Xyce Controlled Sources	In some case, a Xyce netlist that contains a controlled source that uses the TABLE form will get the correct answer at first. However, it may then "stall" (e.g, keep taking really small time-steps) and never complete the simulation run. ***Workaround***: In some cases, the TABLE specification for the controlled source can be replaced with a Piecewise Linear (PWL) source that uses nested IF statements.
783-SON: Use of ddt in a B-Source definition may produce incorrect results	The `DDT()` function from the Xyce expression package, which implements a time derivative, may not function correctly in a B-Source definition. ***Workaround***: None.
727-SON: Xyce parallel builds hang randomly on OS X	During Sandia's internal nightly testing of the OSX parallel builds, we see that Xyce "hangs on exit" with an estimated frequency of less than 1-in-5000 simulation runs. We have not seen this issue with parallel builds for either RHEL6 or BSD. The hang is on exit, whether on a successful exit or on an error exit. The hang occurs after all of the Xyce output has occurred though. So, the user will get their sim results, but might have trouble if the individual Xyce runs are part of a larger script. ***Workaround***: None.
661-SON Lead currents and power accessors (I(), P() and W()) do not work properly in .RESULT Statements	There are two issues. First, `.RESULT` statements will fail netlist parsing if the requested lead current is omitted from the `.PRINT TRAN` line. As an example, this statement (`.RESULT I(R1)`) requires either `I(R1)`, `P(R1)` or `W(R1)` to be on the `.PRINT TRAN` line. Second, the output value, in the `.res` file, for the lead current or power calculation will always be zero.

Table 3: Known Defects and Workarounds.

Defect	Description
583-SON: Switch with RON=0 leads to convergence failure.	The switch device does not prevent a user from specifying `RON=0` in its model, but then takes the inverse of this value to get the "on" conductance. The resulting invalid division will either lead to a division by zero error on platforms that throw such errors, or produce a conductance with "Not A Number" or "Infinity" as value. This will lead to a convergence failure. ***Workaround***: Do not specify an identically zero resistance for the switch's "on" value. A small value of resistance such as 1e-15 or smaller will generally work well as a substitute.
469-SON: Belos memory consumption on FreeBSD and excessive CPU on other platforms	Memory or thread bloat can result when using multithreaded dense linear algebra libraries, which are employed by Belos. If this situation is observed, either build Xyce with a serial dense linear algebra library or use environment variables to control the number of spawned threads in a multithreaded library.
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.
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 a Xyce `.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.
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.
49-SON Xyce BSIM models recognize the model TNOM, but not the instance TNOM	Some simulators allow the model parameter TNOM of BSIM devices to be specified on the instance line, overriding the model parameter TNOM. Xyce does not support this.
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.

Table 3: Known Defects and Workarounds.

Defect	Description

2119-SRN: Voltages from interface nodes for subcircuits do not work in expressions used in device instance parameters	This bug can be illustrated with this netlist fragment:
	
	`X1 1 2 MySub`
	`.SUBCKT MYSUB a c`
	`R1  a b 0.5`
	`R2  b c 0.5`
	`.ENDS`
	`B1 3 0 V={V(X1:a)}`
	
	This fragment will produce the netlist parsing error `Directory node not found: X1:A`. The workaround is to use `V={V(1)}` in the B-source expression instead. This bug also affects the solution-dependent capacitor.

1923-SRN: LC lines run out of memory, even if equivalent (larger) RLC lines do not.	In some cases, circuits that run fine using an RLC approximation for a transmission line, exit with an out-of-memory error if the (supposedly smaller) LC approximation is used.
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.

Supported Platforms

Certified Support

The following platforms have been subject to certification testing for the Xyce version 7.1 release.

- Red Hat Enterprise Linux® 7, x86-64 (serial and parallel)
- Microsoft Windows 10®, x86-64 (serial)
- Apple® OS X Sierra, x86-64 (serial and parallel)

Build Support

Though not certified platforms, Xyce has been known to run on the following systems.

- FreeBSD 11.x on Intel x86-64 and AMD64 architectures (serial and parallel)
- Distributions of Linux other than Red Hat Enterprise Linux 7
- Microsoft Windows under Cygwin and MinGW.

Xyce Release 7.1 Documentation

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

- Xyce Version 7.1 Release Notes (this document)
- Xyce Users' Guide, Version 7.1
- Xyce Reference Guide, Version 7.1
- Xyce Mathematical Formulation
- Power Grid Modeling with Xyce
- Application Note: Coupled Simulation with the Xyce General External Interface
- Application Note: Mixed Signal Simulation with Xyce 7.1

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

- Frequently Asked Questions
- Building Guide (instructions for building Xyce from the source code)
- Running the Xyce Regression Test Suite
- Xyce/ADMS Users' Guide
- Tutorial: Adding a new compact model to Xyce

External User Resources

- Website: http://xyce.sandia.gov
- Google Groups discussion forum: https://groups.google.com/forum/#!forum/xyce-users
- Email support: xyce@sandia.gov
- Address:

Electrical Models and Simulation Dept.
Sandia National Laboratories
P.O. Box 5800, M.S. 1177
Albuquerque, NM 87185-1177

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.