NOTICE: This document contains references to Agilent Technologies. Agilent's former Test and Measurement business has become Keysight Technologies. For more information, go to **www.keysight.com.**





ME5

Building an Excel Table with Perl

Application Note

Copyright Notice and Proprietary Information

Copyright © Agilent Technologies, Inc. 2003, 2011. All rights reserved. This software and documentation contain confidential and proprietary information that is the property of Agilent Technologies, Inc. The software and documentation are furnished under a license agreement and may be used or copied only in accordance with the terms of the license agreement. No part of the software and documentation may be reproduced, transmitted, or translated, in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without prior written permission of Agilent Technologies, Inc., or as expressly provided by the license agreement.

Right to Copy Documentation

The license agreement with Agilent Technologies permits licensee to make copies of the documentation for its internal use only. Each copy shall include all copyrights, trademarks, service marks, and proprietary rights notices, if any. Licensee must assign sequential numbers to all copies.

Disclaimer

AGILENT TECHNOLOGIES, INC. AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Registered Trademarks (®)

Agilent, Model Builder Program, MBP, Model Quality Assurance, MQA, Advanced Model Analysis, AMA, Proximity Quality Assurance, PQA are registered trademarks of Agilent Technologies, Inc.

Building an Excel Table with Perl

Application Note

This application note describes how to build a user-defined Excel table with Perl utilizing a simple and fast method. **Note**: This document was originally released for MQA V2010.2.0 in April 2011.

Introduction

Calling SPICE and analyzing the resulting calculations by hand is always time-consuming. Sometimes shell scripts can be used to reduce the workload, but this is not an ideal solution for large projects.

Excel tables can provide a more efficient solution and are widely accepted in the industry by engineers. Model Quality Assurance's (MQA's) Perl-based rule provides a simple, flexible way to process SPICE results and write data to Excel using Perl's powerful functionality.

How to Build Excel Table

To better understand how an Excel table is built with Perl, consider the following test case. The example table includes two targets: Vthcon and Idsat/W. We want to make a comparison between the simulation and specification values. The error of Vth (dVth) is calculated as the absolute delta value, while the error of Idsat/W (dIdsat) is calculated as the relative percentage.

Prepare Excel Table

Licrosoft Excel - LyTable.xls											
▼ 12											
Image: Contraction of the second											
K											
h(mv)											
438.00											
497.00											
00.88											
391.00											
653.00											
9 Idsat/W											
sat (%)											
100.00)											
100.00)											
100.00)											
100.00)											
100.00)											
(1											

Figure 1. Excel table example

As shown in Figure 1, the simulation values in the yellow cells are first set to zero by default. The equation to calculate the error of Idsat/W (dIdsat) is defined in the Excel table (surrounded by the red

rectangle box).

The user can save this Excel file as "*MyTable.xls*" in an empty directory. It is recommended that the user also save a copy of this file as a backup since new results will be written into the Excel file directly.

Create Perl File to Process Excel Table

To create a PERL file, the user needs to create a new text file and save it as "*MyTable_update.pl*." Then, insert the lines shown in Table 1 into the Perl file.

Codes of Perl	Comments		
use acc_excel;	Import Agilent's scripts		
	for Excel.		
do "function.pl";	Import functions to		
	calculate Vth/Idsat.		
sacset("model","nch","-type","nmos","-file","Hb3v3.lib","-corner","SS");	Setup models that will		
	be used in simulation.		
sacset("model","nch","-type","nmos","-file","Hb3v3.lib","-corner","TT");			
sacset("model","nch","-type","nmos","-file","Hb3v3.lib","-corner","FF");			
openExcel("MyTable.xls");	Open Excel file.		
W = getcellvalue(4,1);	Get W, L with given		
	index of row, col.		
L = getcellvalue(4,2);			
	And, W, L can be given		
	with constants directly.		
<pre>setcellvalue(4,4,CalcValue("Vthcon","W=(\$W)L=(\$L)","SS"));</pre>	Call simulator and set		
	value to cell with given		
setcellvalue(4,7,CalcValue("Vthcon","W=(\$W)L=(\$L)","TT"));	row, col.		
setcellvalue(4,10,CalcValue("Vthcon","W=(\$W)L=(\$L)","FF"));			
closeExcel();	Save Excel file to disk.		

Table 1. Content of Perl File to Update Excel Table

Run Perl File

After saving the Excel file and coding in Perl scripts, the user can run Perl to update the Excel file with the results from SPICE: "*Perl MyTable_update.pl.*"

The resulting Excel table is shown in Figure 2.

X I6	M Licrosoft Excel - LyTable.xls												
:	文件(2) 编	辑(E) 视图	(V) 插入(E)	格式 (0)	工具(I) 数据	(11) 窗口(11)) 帮助(H)						
: D	📬 🛃 🖪 i	a 4 🖪	秒 🛍 🐰	🗈 🖺 • <	3 19 - (2 -	😣 Σ - 🖞	2 I 🕹 I 🛍 🛛	🦚 100% 🗸 🗸	② 🛃 🙀	5	• 12 ·		
D12 • 149.7214													
	A	В	С	D	E	F	G	Н	I	J	K		
1													
	<u>Vthcon</u>												
3	Ψ	L	SS Spec	SS	dVth(mv)	TT Spec	TT	dVth(mv)	FF Spec	FF	dVth(mv)		
4	1	1	0.65	0.648384	-1.62	0.54	0.541965	1.96	0.438	0.43856	0.56		
5	1	2	0.71	0.712826	2.83	0.6	0.602169	2.17	0.497	0.497814	0.81		
6	1	10	1.4	1.471729	71.73	1.235	1.235218	0.22	1.088	1.088968	0.97		
7	2	1	0.59	0.593401	3.40	0.49	0.491938	1.94	0.391	0.391019	0.02		
8	2	10	0.886	0.886224	0.22	0.759	0.759933	0.93	0.653	0.653333	0.33		
9	<u>Idsat/W</u>												
10	₩	L	SS Spec	SS	dIdsat (%)	TT Spec	TT	dIdsat(%)	FF Spec	FF	dIdsat (%)		
11	1	1	191.76	263.0946	37.20	239.59	326.3309	36.20	277.36	376.6272	35.79		
12	1	2	104.90	149.7214	42.72	130.80	185.1433	41.54	151.08	214.2265	41.79		
13	1	10	22.78	33.71969	48.03	28.43	41.69124	46.63	32.79	48.40279	47.60		
14	2	1	194.47	269.1363	38.40	234.57	322.1964	37.36	267.15	365.3023	36.74		
15	2	10	24.43	36.62526	49.90	29.41	43.57403	48.14	32.99	49.17348	49.05		
16													

Figure 2. Resulting Excel table

The values in the yellow cells are updated by the simulation results from SPICE. At the same time, the errors are automatically re-calculated by Excel.

Notes

The example case presented in this document was tested under Windows ® XP with Perl 5.10. The user can follow the steps mentioned above to customize different Excel tables. For more information go to <u>www.agilent.com/find/eesof</u> or contact your local Agilent office. The complete list is available at: <u>www.agilent.com/find/contactus</u>.

Windows and MS Windows are U.S. registered trademarks of Microsoft Corporation.