User Guide for QuickHnery

SPICE (RL) compatible and INDUCTWISE (RL/RK) compatible

(Ver. 1.0)

Clement Luk, Tsung-Hao Chen and Charlie Chung-Ping Chen

VLSI-EDA Group (http://vlsi.ece.wisc.edu)

Electrical and Computer Engineering, University of Wisconsin-Madison

Aug 15, 2002

Contents

1 Introduction

1.1 Abstract

As semi-conductor industries embark to the giga hertz era, correct modeling and simulation of digital circuit that incorporate inductance becomes indispensable issue. Simple RC extraction and simulation can no longer cope with the ever-increasing operating frequencies. With the introduction of more metal layers, lowered operating voltage, thinner and longer interconnect wires for global signals and clock distribution seem to even aggravate the inductance effect. Inductance effect impacts various aspects in digital design in terms of propagation delay, degradation of signal integrity due to undershoots, overshoots and oscillations, and increasing power gird noise and worsening signal cross talk. However, due to the long-range effect of inductance, simulating the dense partial inductance matrix is difficult and impractical when applying simulation to entire chip. A recent proposed circuit element K [1], which defined as inverse of partial inductance matrix L, has properties reminiscent of capacitance extraction such as locality and stability. Based on K-method matrix sparsification technique, we developed a robust, efficient, and accurate extraction tool that could directly extract multi-conductors system into K circuit compliance circuit simulation tool, QuickHenry and can be simulated by INDUCTWISE [2]. The experimental results show that circuit extracted under K-method that simulate using INDUCTWISE has over 146X speedup compared to Tanner SPICE with less than 0.1% error.

1.2 Product Features

This tool is used for efficiently extract on-chip inductance for VLSI application. With the continuous shrinking feature size of transistor, efficient extraction for power-grid simulation, signal integrity verification for global interconnect, and clock becomes an imperative mission for today's VLSI design.

Full L extraction

Full SPICE compatible self-inductance and mutual inductance value that could be simulated in either INDUCTWISE or any other commercial SPICE simulator.

Novel technique K (inverse inductance) method

Using the recently proposed K (inverse inductance, a.k.a. reluctance) method, this tool could extract self and mutual reluctance extraction via windowing. This method could significantly sparsify the reluctance matrix but still capturing the inductance effect. More experimental result could be found in [2] experimental result section.

3-Dimensional K method

For a multi-layer conductor system, coupling across different layer is also a concern for the correct modeling for the system. This method is to taken care of coupling across different layer by assuming a virtual aggressor exists in the layer to be extracted.

2 How to run the program?

2.1 Input File Format

We have employed a very easy-to-understand format to clearly describe the geometry, width, layer information and orientation.

The general format is as follow:

P[name] [node1] [node2] [x1 cord] [y1 cord] [x2 cord] [y2 cord] [layer={M (1,2,..)}] [w={width}] [v|h]

P[name]
All conductor starts with 'P' and follow by the name.

[node 1]
Name of starting side of the conductor.

[node 2]
Name of ending side of the conductor.

[x1 cord]
X coordinate for node 1. For vertical (v) orientation, this is the mid-point of the conductor. This number should be the same as the x2 for vertical (v) orientation. This means that for vertical orientation x1, and x2 describe the spatial location and y1 and y2 describe the length. See figure 1 for pictorial description for vertical orientation. For horizontal (h) orientation, it is the starting point, and x2 is the ending point of the width. See figure 2 for pictorial description for horizontal orientation

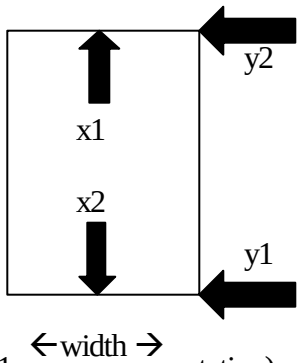

Figure 1. (Vertical orientation)

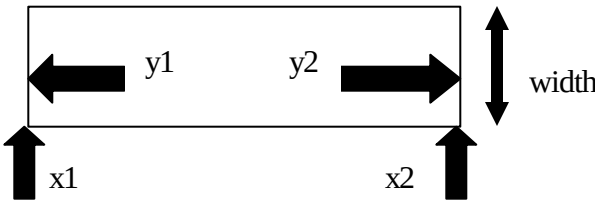

Figure 2. (Horizontal orientation)

[y1 cord]
Y coordinate for node 1. See figure 1 for pictorial description for vertical orientation. And see figure 2 for pictorial description for horizontal orientation.

[x2 cord]

X coordinate for node 2. See figure 1 for pictorial description for vertical orientation.
And see figure 2 for pictorial description for horizontal orientation.

[y2 cord]
Y coordinate for node 2. See figure 1 for pictorial description for vertical orientation.
And see figure 2 for pictorial description for horizontal orientation.

[layer={M (1,2,..)}]
Layer of conductor.

[w={width}]
Width of the conductor.

[v|h]
Orientation of the conductor: 'v' means vertical and 'h' means horizontal.

Comment
Also, anything starting with "*" is treated as a comment line.

A typical input file example:

Ptest1 a1 b1 x1=1e-6 y1=4e-6 x2=1e-006 y2=24e-6 layer=M6 w=0.6e-6 v

2.2 Instruction for running the program

fext <input file> <spice file name> {-t tech_file} {Switch}

---------- Switch ----------
[-N {number of conductor}]
[-L]
[-K {number of conductor} [optional: -R {% increse}]
[-3DK {number of conductor} [optional: -R {% increase}]
[-3DL]
[-LWIN {number of conductor} [optional: -R {% increase}]
[-INV_K {number of conductor} [optional: -R {% increase}]
------- switch description ----------
-L : Full L extraction
-N : K method next neighbor search for sorted dimension (x or y)
-K : K method nearest neighbor search
-3DK : 3D K method nearest neighbor search
-3DL : 3D L extraction
-LWIN : Full extraction using windowing algorithm
-INV_K: Use K method nearest neighbor search, then re-inverse it to get L value

-R : percentage to increase for search, e.g. 0.5 = 50%

There are four different methods for running the program:

1) –t technology_file

If technology file presents, the extractor will use the width specified by the technology file. User can override the value by explicitly specifies the width in the input file.
For example:
Ptest1 a1 b1 x1=1e-6 y1=4e-6 x2=1e-006 y2=24e-6 layer=M6 w=0.6e-6 v

Beside width, if technology file is given, the extractor will use the following value:

1) Conductivity for a particular layer
2) Thickness for a particular layer
3) VIA height (use only in 3D extraction)

Example:

// assume the technology file is called, tech.txt
// assume we use full L method

fext test.txt output.sp –t tech.txt –L

If technology file is not given, the extractor will use the following value as specified by the default technology file that can be downloaded from the web site.

```
**********************************************************************
* M[Sub|Poly|1~8]      : Metal Layer
* V[00~67]            : Via between two layers
* t                   : Thickness
* w                   : Width
* l                   : Length
* r                   : Resistance
* d                   : Conductivity
**********************************************************************

MSub   t=1.96e-7    w=7.7e-8          d=1.0e+4   v
V00    t=3.90e-7    w=7.7e-8  l=7.7e-8   r=2.6 d=4.947e+7
MPoly  t=1.12e-7    w=7.7e-8          d=3.80e+7  h
V01    t=1.03e-7    w=1.2e-7  l=1.2e-7   r=2.6 d=4.947e+7
M1     t=2.80e-7    w=1e-6     d=4.947e+7 v
V12    t=2.0e-7     w=1.2e-7   l=1.2e-7   r=2.6 d=4.947e+7
M2     t=3.6e-7     w=2.25e-7  d=4.996e+7 h
V23    t=2.10e-7    w=1.2e-7   l=1.2e-7   r=2.6 d=4.947e+7
M3     t=3.6e-7     w=3.80e-7  d=4.996e+7 v
V34    t=2.10e-7    w=1.6e-7   l=1.2e-7   r=2.5 d=4.947e+7
M4     t=5.70e-7    w=3.80e-7          d=4.996e+7 h
```

```
V45    t=4.60e-7    w=2.0e-7  l=2.0e-7   r=1.6 d=4.947e+7
M5     t=9.0e-7     w=5.63e-7 d=5.8e+7   v
V56    t=5.30e-7    w=2.4e-7  l=2.4e-7   r=1.2 d=4.947e+7
M6     t=1.2e-6     w=6.0e-7    d=5.8e+7   v
V67    t=5.305e-7   w=3.6e-7 l=3.6e-7   r=0.6 d=4.947e+7
M7     t=1.2e-6     w=6.0e-7    d=5.8e+7   v
V7C    t=8.90e-5    w=3.0e-6  l=3.0e-6   r=0.06 d=5.8e+7
M8     t=8.90e-5    w=3.0e-6   d=4.990e+7 h
```

2) –L

This is to extract the full L self and mutual inductance.

Example:

```
// full L method
fext  test.txt output.sp –L
// if using technology file
fext test.txt output.sp –t tech.txt -L
```

3) –K

This is to use the K method to extraction via windowing. The number next to"-K" is the number of neighbors to be chosen in the search window.

Example:

```
// using K method with 3 neighbor search
fext test.txt output.sp –K 3

// using K method with 2 neighbor search and also with 50 % extension search range
fext test.txt output.sp –K 2 –R 0.5

// using K method with 2 neighbor search and with 50 % extension search range
// and also using technology file (tech.txt)

fext test.txt output.sp –t tech.txt –K 2 –R 0.5
```

4) –3DK

This is for multi-layer conductor system. It extracts K method for multiple layers. The number next to "-3DK" is the number of neighbors to be chosen in the search window.

```
// using K method with 3 neighbor search
Example:
// using 3D K method with 3 neighbor search in each layer
// we assume a virtual aggressor in different layer
fext test.txt output.sp –3DK 3

// using 3D K method with 2 neighbor search and also with 50 % extension search range
fext test.txt output.sp –3DK 2 –R 0.5
```

// using 3D K method with 2 neighbor search and with 50 % extension search range
// and also using technology file
fext test.txt output.sp –t tech.txt –3DK 2 –R 0.5

5) –N

This is a method to do K method extraction without using windowing. This method only picks out the nearby neighbor by the sorted dimension without using any windowing. In fact, this is just an experimental method that to confirmed the stability of using windowing.

Example:

// using next neighbor search (in sorted dimension) with 2 neighbor search
fext test.txt ouput.sp –N 2

// using next neighbor search (in sorted dimension) with 2 neighbor search
// and also using technology file (tech.txt)
fext test.txt output.sp –t tech.txt –N 2

6) –3DL

This is to extract the full L self and mutual inductance for multi-layer conductor system.

Example:

// using 3D L method with 3 neighbor search in each layer
// we assume a virtual aggressor in different layer
fext test.txt output.sp –3DL 3

// using 3D L method with 2 neighbor search and also with 50 % extension search range
fext test.txt output.sp –3DL 2 –R 0.5

// using 3D L method with 2 neighbor search and with 50 % extension search range
// and also using technology file
fext test.txt output.sp –t tech.txt –3DL 2 –R 0.5

7) –LWIN

This is the full L extraction that employs the same windowing algorithm as K method.

// using L windowing method with 3 neighbor search
fext test.txt output.sp –LWIN 3

// using L windowing method with 2 neighbor search
//and also with 50 % extension search range
fext test.txt output.sp –LWIN 2 –R 0.5

// using L windowing method with 2 neighbor search
//and with 50 % extension search range
// and also using technology file (tech.txt)

fext test.txt output.sp –t tech.txt –LWIN 2 –R 0.5

8) –INVK
This is essential K method but instead of outputting it as K self and mutual inductance value, we re-invert the matrix first to get the full L matrix value.

Example:
// using inverse K method with 3 neighbor search
fext test.txt output.sp –K 3

// using inverse K method with 2 neighbor search
//and also with 50 % extension search range
fext test.txt output.sp –K 2 –R 0.5

// using inverse K method with 2 neighbor search
// and with 50 % extension search range
// and also using technology file (tech.txt)

fext test.txt output.sp –t tech.txt –INVK 2 –R 0.5

9) Extension factor ("-R")
Example: –K 2–R 0.5

In order to fully capture the inverse inductance effect, we have included an extension factor in window searching. What "-R 0.5" means is that it extends the current conductor by 50% so as to capture those that still fall within the extended window. From the experiment we found out that this using the extension factor could significantly improve the accuracy of K method. In figure 3, the extension factor is indicated as "xL".

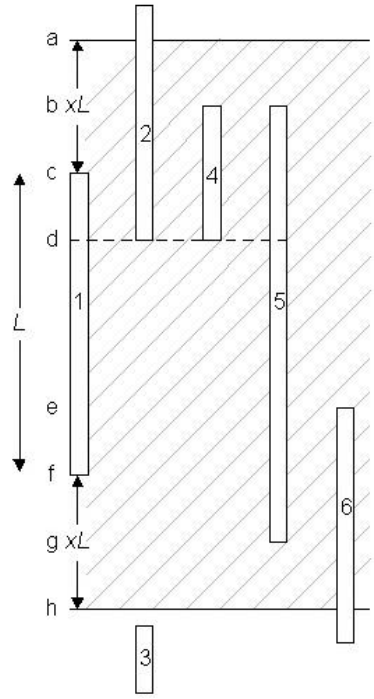

Figure 3

2.3 Output file format:

Generally, it is the exact format as in the any commercial available SPICE format.

Self-inductance:
Ltest9 a9_mid b9 1.27261E-011

Note: All self-inductance begins with keyword “L”.

// inductor --- name
// a_9_mid --- one end of the node
// b9 --- another end of the node
// 1.27261E-011 inductance value

Mutual inductance:
Ktest0_test9 Ltest0 Ltest9 1.90628E-001
// mutual inductance between Ltest0 and Ltest9

Note: All mutual inductance begins with keyword “K”.

Self-reluctance:
Ytest0 a0_mid b0 1.734504E+010

Note: All self-reluctance begins with keyword “Y”.

Mutual reluctance:
Wtest0_test1 Ytest0 Ytest1 -4.267909E-001
// mutual reluctance between Ytest0 and Ytest1
Note: All mutual reluctance begins with keyword “W”.

Instruction for inductance extraction:

Format:

fext <input file> <spice file name> {-t tech_file} {Switch}

---------- Switch ----------
[-N {number of conductor}]
[-L]
[-K {number of conductor} [optional: -R {% increase}]
[-3DK {number of conductor} [optional: -R {% increase}]
[-3DL]
[-LWIN {number of conductor} [optional: -R {% increase}]
[-INV_K {number of conductor} [optional: -R {% increase}]
------- switch description ----------

-L : Full L extraction
-N : K method next neighbor search for sorted dimension (x or y)
-K : K method nearest neighbor search
-3DK : 3D K method nearest neighbor search
-3DL : 3D L extraction
-LWIN : Full extraction using windowing algorithm
-INV_K: Use K method nearest neighbor search, then re-inverse it to get L value
-R : percentage to increase for search, e.g 0.5 = 50%
Example:

1)
To extract full L self and mutual inductance:

fext p500_test.txt p500_test.sp –L

2)
To extract K method self and mutual reluctance with 2 neighbor:

fext p500_test.txt p500_test.sp –K 2

3)
To extract K method self and mutual reluctance with 2 neighbors and with extension factor 50 %: (estimated time 6 secs)

fext p500_test.txt p500_test.sp –K 2 –R 0.5

2.4 A 2D small example

INPUT FILE

* A SMALL EXAMPLE WITH 2 CONDUCTORS

Ptest12 a12 b12 x1=11.6e-006 y1=0 x2=11.6e-006 y2=90e-6 layer=M6 w=0.6e-6 v
Ptest40 a40 b40 x1=51.3e-6 y1=0 x2=51.3e-6 y2=70e-6 layer=M6 w=0.6e-6

COMMAND:
(using the full L method)
fext testing.txt testing.sp –L

OUTPUT (using the full L method)

*the spice file

Ltest0 a0_mid b0 1.04265E-010
Ltest1 a1_mid b1 3.50630E-011
Ktest0_test1 Ltest0 Ltest1 5.29182E-001

Rtest0 a0 a0_mid 2.39464E+000
Rtest1 a1 a1_mid 9.72082E-001

COMMAND:
(using the K method with 2 neighbor for windowing)

fext testing.txt testing.sp –K 2

OUTPUT (using the K method with 2 neighbor for windowing)

***This is the spice file
Ytest0 a0_mid b0 1.332139E+010
Ytest1 a1_mid b1 3.961306E+010

Wtest0_test1 Ytest0 Ytest1 -5.291818E-001

Rtest0 a0 a0_mid 2.39464E+000
Rtest1 a1 a1_mid 9.72082E-001

2.5 A 3D small example

INPUT FILE

*A 3D EXAMPLE WITH 6 layers of metal

*in layer M2 (vertical orientation)
Ptest1 a1 b1 x1=25e-006 y1=0 x2=25e-006 y2=90e-6 layer=M2 w=1e-6 v
Ptest2 a2 b2 x1=35e-006 y1=0 x2=35e-006 y2=90e-6 layer=M2 w=1e-6 v

*in layer M3 (horizontal orientation)
Ptest3 a3 b3 x1=0 y1=5e-6 x2=200e-6 y2=5e-6 layer=M3 w=1e-6 h
Ptest4 a4 b4 x1=0 y1=15e-6 x2=200e-6 y2=15e-6 layer=M3 w=1e-6 h
Ptest5 a5 b5 x1=0 y1=25e-6 x2=200e-6 y2=25e-6 layer=M3 w=1e-6 h

*in layer M4 (vertical orientation)
Ptest6 a6 b6 x1=20e-006 y1=0 x2=20e-006 y2=90e-6 layer=M4 w=1e-6 v
Ptest7 a7 b7 x1=30e-006 y1=0 x2=30e-006 y2=90e-6 layer=M4 w=1e-6 v
Ptest8 a8 b8 x1=40e-6 y1=0 x2=40e-6 y2=70e-6 layer=M4 w=1e-6

* in layer M5 (horizontal orientation)
Ptest9 a9 b9 x1=0 y1=10e-6 x2=200e-6 y2=10e-6 layer=M5 w=16e-6 h
Ptest9 a9 b9 x1=0 y1=20e-6 x2=200e-6 y2=20e-6 layer=M5 w=1e-6 h
Ptest10 a10 b10 x1=0 y1=30e-6 x2=200e-6 y2=30e-6 layer=M5 w=1e-6 h

* in layer M6 (vertical orientation)
Ptest11 a11 b11 x1=11.6e-006 y1=0 x2=11.6e-006 y2=90e-6 layer=M6 w=1e-6 v
Ptest12 a12 b12 x1=51.3e-6 y1=0 x2=51.3e-6 y2=70e-6 layer=M6 w=1e-6

* in layer M7 (horizontal orientation)
Ptest13 a13 b13 x1=0 y1=12e-6 x2=200e-6 y2=12e-6 layer=M7 w=1e-6 h
Ptest14 a14 b14 x1=0 y1=22e-6 x2=200e-6 y2=22e-6 layer=M7 w=1e-6 h

COMMAND: (3D full L method)
fext 3D_test.txt 3D_test.sp –3DL

OUTPUT: (3D full L method)

Ltest11 a11_mid b11 8.83634E-011
Ltest6 a6_mid b6 9.44010E-011
Ltest1 a1_mid b1 9.69712E-011
Ltest7 a7_mid b7 9.44010E-011
Ltest2 a2_mid b2 9.69712E-011
Ltest8 a8_mid b8 6.99231E-011
Ltest12 a12_mid b12 6.52341E-011
Ltest9 a9_mid b9 1.47566E-010

Ltest3 a3_mid b3 2.47340E-010
Ltest13 a13_mid b13 2.28163E-010
Ltest4 a4_mid b4 2.47340E-010
Ltest9 a9_mid b9 2.34012E-010
Ltest14 a14_mid b14 2.28163E-010
Ltest5 a5_mid b5 2.47340E-010
Ltest10 a10_mid b10 2.34012E-010
Ktest11_test6 Ltest11 Ltest6 4.15647E-001
Ktest11_test1 Ltest11 Ltest1 3.31378E-001
Ktest11_test7 Ltest11 Ltest7 2.88834E-001
Ktest11_test2 Ltest11 Ltest2 2.47400E-001
Ktest11_test8 Ltest11 Ltest8 2.06776E-001
Ktest11_test12 Ltest11 Ltest12 1.69941E-001
Ktest6_test1 Ltest6 Ltest1 4.91413E-001
Ktest6_test7 Ltest6 Ltest7 3.81159E-001
Ktest6_test2 Ltest6 Ltest2 3.08919E-001
Ktest6_test8 Ltest6 Ltest8 2.48781E-001
Ktest6_test12 Ltest6 Ltest12 1.93959E-001
Ktest1_test7 Ltest1 Ltest7 4.93054E-001
Ktest1_test2 Ltest1 Ltest2 3.71084E-001
Ktest1_test8 Ltest1 Ltest8 2.86868E-001
Ktest1_test12 Ltest1 Ltest12 2.13589E-001
Ktest7_test2 Ltest7 Ltest2 4.91413E-001
Ktest7_test8 Ltest7 Ltest8 3.54241E-001
Ktest7_test12 Ltest7 Ltest12 2.47012E-001
Ktest2_test8 Ltest2 Ltest8 4.56125E-001
Ktest2_test12 Ltest2 Ltest12 2.80251E-001
Ktest8_test12 Ltest8 Ltest12 3.43454E-001
Ktest9_test3 Ltest9 Ltest3 7.31015E-001
Ktest9_test13 Ltest9 Ltest13 7.29721E-001
Ktest9_test4 Ltest9 Ltest4 7.31015E-001
Ktest9_test9 Ltest9 Ltest9 6.18851E-001
Ktest9_test14 Ltest9 Ltest14 5.65441E-001
Ktest9_test5 Ltest9 Ltest5 5.03308E-001
Ktest9_test10 Ltest9 Ltest10 4.56571E-001
Ktest3_test13 Ltest3 Ltest13 4.83308E-001
Ktest3_test4 Ltest3 Ltest4 4.42950E-001
Ktest3_test9 Ltest3 Ltest9 3.90725E-001
Ktest3_test14 Ltest3 Ltest14 3.70570E-001
Ktest3_test5 Ltest3 Ltest5 3.38549E-001
Ktest3_test10 Ltest3 Ltest10 3.14452E-001
Ktest13_test4 Ltest13 Ltest4 5.64794E-001
Ktest13_test9 Ltest13 Ltest9 4.99941E-001
Ktest13_test14 Ltest13 Ltest14 4.79997E-001
Ktest13_test5 Ltest13 Ltest5 4.11035E-001
Ktest13_test10 Ltest13 Ltest10 3.76764E-001

Ktest4_test9 Ltest4 Ltest9 5.56028E-001
Ktest4_test14 Ltest4 Ltest14 4.83308E-001
Ktest4_test5 Ltest4 Ltest5 4.42950E-001
Ktest4_test10 Ltest4 Ltest10 3.90725E-001
Ktest9_test14 Ltest9 Ltest14 6.33316E-001
Ktest9_test5 Ltest9 Ltest5 5.61137E-001
Ktest9_test10 Ltest9 Ltest10 4.68085E-001
Ktest14_test5 Ltest14 Ltest5 5.64794E-001
Ktest14_test10 Ltest14 Ltest10 4.99941E-001
Ktest5_test10 Ltest5 Ltest10 5.56028E-001

Rtest11 a11 a11_mid 1.29310E+000
Rtest6 a6 a6_mid 3.16042E+000
Rtest1 a1 a1_mid 5.00400E+000
Rtest7 a7 a7_mid 3.16042E+000
Rtest2 a2 a2_mid 5.00400E+000
Rtest8 a8 a8_mid 2.45811E+000
Rtest12 a12 a12_mid 1.00575E+000

Rtest9 a9 a9_mid 2.39464E-001
Rtest3 a3 a3_mid 1.11200E+001
Rtest13 a13 a13_mid 2.87356E+000
Rtest4 a4 a4_mid 1.11200E+001
Rtest9 a9 a9_mid 3.83142E+000
Rtest14 a14 a14_mid 2.87356E+000
Rtest5 a5 a5_mid 1.11200E+001

COMMAND: (3D K method with 2 neighbor search)
fext 3D_test.txt 3D_test.sp –3DK 2
// to attach technology file, use –t <file name>
fext 3D_test.txt 3D_test.sp –t tech.txt –3DK 2
// to increase search range by a extension factor of 50% to –R 0.5
fext 3D_test.txt 3D_test.sp –t tech.txt –3DK 2 –R 0.5

OUPUT: (3D K method with 2 neighbor search)
Ytest11 a11_mid b11 1.435304E+010
Ytest6 a6_mid b6 1.594788E+010
Ytest1 a1_mid b1 1.637523E+010
Ytest7 a7_mid b7 1.703440E+010
Ytest2 a2_mid b2 1.598372E+010
Ytest8 a8_mid b8 1.978590E+010
Ytest12 a12_mid b12 1.804725E+010

Wtest11_test6 Ytest11 Ytest6 -2.807786E-001
Wtest11_test1 Ytest11 Ytest1 -9.994011E-002
Wtest11_test7 Ytest11 Ytest7 -6.486580E-002
Wtest11_test2 Ytest11 Ytest2 -4.944108E-002
Wtest11_test12 Ytest11 Ytest12 -4.894918E-002
Wtest6_test1 Ytest6 Ytest1 -3.124134E-001
Wtest6_test7 Ytest6 Ytest7 -1.013220E-001
Wtest6_test2 Ytest6 Ytest2 -5.036202E-002
Wtest6_test8 Ytest6 Ytest8 -3.916992E-002
Wtest6_test12 Ytest6 Ytest12 -3.430985E-002
Wtest1_test7 Ytest1 Ytest7 -2.826645E-001
Wtest1_test2 Ytest1 Ytest2 -8.819343E-002
Wtest1_test8 Ytest1 Ytest8 -4.668042E-002
Wtest1_test12 Ytest1 Ytest12 -3.545606E-002
Wtest7_test2 Ytest7 Ytest2 -2.963210E-001
Wtest7_test8 Ytest7 Ytest8 -9.717074E-002
Wtest7_test12 Ytest7 Ytest12 -5.436836E-002
Wtest2_test8 Ytest2 Ytest8 -2.941556E-001
Wtest2_test12 Ytest2 Ytest12 -8.670772E-002
Wtest8_test12 Ytest8 Ytest12 -2.244084E-001

Ytest9 a9_mid b9 3.803007E+010
Ytest3 a3_mid b3 9.168947E+009
Ytest13 a13_mid b13 9.634170E+009
Ytest4 a4_mid b4 9.370538E+009
Ytest9 a9_mid b9 9.337285E+009
Ytest14 a14_mid b14 8.923593E+009
Ytest5 a5_mid b5 7.550352E+009
Ytest10 a10_mid b10 6.884967E+009

Wtest9_test3 Ytest9 Ytest3 -6.129623E-001
Wtest9_test13 Ytest9 Ytest13 -4.426790E-001
Wtest9_test4 Ytest9 Ytest4 -4.670804E-001
Wtest9_test9 Ytest9 Ytest9 -1.715084E-001
Wtest9_test14 Ytest9 Ytest14 -9.070907E-002
Wtest9_test10 Ytest9 Ytest10 -5.376574E-002
Wtest3_test13 Ytest3 Ytest13 8.955547E-002
Wtest3_test4 Ytest3 Ytest4 1.745948E-001
Wtest3_test9 Ytest3 Ytest9 6.234923E-002
Wtest3_test14 Ytest3 Ytest14 1.740182E-002
Wtest3_test5 Ytest3 Ytest5 -1.882169E-002
Wtest3_test10 Ytest3 Ytest10 -1.206630E-002
Wtest13_test4 Ytest13 Ytest4 -2.924767E-002
Wtest13_test9 Ytest13 Ytest9 -2.226256E-002
Wtest13_test14 Ytest13 Ytest14 -7.261615E-002

Wtest13_test5 Ytest13 Ytest5 -1.934131E-002
Wtest13_test10 Ytest13 Ytest10 -2.361437E-002
Wtest4_test9 Ytest4 Ytest9 -1.235583E-001
Wtest4_test14 Ytest4 Ytest14 -1.868394E-002
Wtest4_test5 Ytest4 Ytest5 -5.165075E-002
Wtest4_test10 Ytest4 Ytest10 -2.052873E-002
Wtest9_test14 Ytest9 Ytest14 -3.144460E-001
Wtest9_test5 Ytest9 Ytest5 -1.975728E-001
Wtest9_test10 Ytest9 Ytest10 -6.493982E-002
Wtest14_test5 Ytest14 Ytest5 -2.077313E-001
Wtest14_test10 Ytest14 Ytest10 -1.593550E-001
Wtest5_test10 Ytest5 Ytest10 -3.175773E-001

Rtest11 a11 a11_mid 1.29310E+000
Rtest6 a6 a6_mid 3.16042E+000
Rtest1 a1 a1_mid 5.00400E+000
Rtest7 a7 a7_mid 3.16042E+000
Rtest2 a2 a2_mid 5.00400E+000
Rtest8 a8 a8_mid 2.45811E+000
Rtest12 a12 a12_mid 1.00575E+000

Rtest9 a9 a9_mid 2.39464E-001
Rtest3 a3 a3_mid 1.11200E+001
Rtest13 a13 a13_mid 2.87356E+000
Rtest4 a4 a4_mid 1.11200E+001
Rtest9 a9 a9_mid 3.83142E+000
Rtest14 a14 a14_mid 2.87356E+000
Rtest5 a5 a5_mid 1.11200E+001
Rtest10 a10 a10_mid 3.83142E+000

3 References:

[1] Hao Ji, Anirudh Devgan, and Wayne Dai, Ksim: A stable and efficient RKC simulator for capturing on-chip inductance effect

[2] Tsung-Hao Chen, Clement Luk, and Charlie Chung-Ping Chen, INDUCTWISE: Inductance-wise interconnect Simulator and Extractor