Hipex RC: Accuracy Improvement in Parasitic Capacitance Extraction

Introduction

Hipex RC is a rule-based parasitic extraction tool. It is used not only for block designs, but also for full-chip designs. Its accuracy in both resistance and capacitance extractions largely depends on the quality of the extraction rule file used. Therefore, to improve accuracy, the extraction rule file must be created in a better way.

With the recent deep-submicron process, accurate parasitic extraction from layout designs is critical because the parasitics affect the circuit behavior significantly.

In this application note, we apply several ideas for accuracy improvement to creating the rule file.

This application note walks you through creating this new rule file in Hipex and confirming the accuracy improvement.

1. Hipex Capacitance Extraction Rule File

The Hipex capacitance extraction rule files provided by Silvaco PDK are often created by existing information about rules provided by FABs, if available. In other cases, we use a tool named Exact to create a parasitic extraction rule file. If process information is available, Exact can create the rule files for any process technology.

Exact uses the following input data:

- Layout model
- Wire width
- Wire spacing variety
- Cross-sectional information (process steps, layer thickness, etc.)
- Permittivity of each material

Exact creates the parasitic extraction rule file through the following steps.

1. Apply wire spacing values to the layout models and generate basic layout for each spacing value applied.

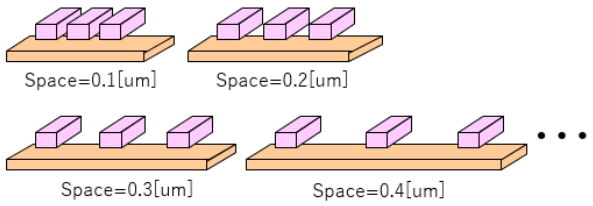


Figure 1. Generated basic layouts.

2. Run the 2D/3D field solver-based parasitic extraction tool Clever for each generated basic layout, then extract highly accurate capacitances between nets.

Space [um]	Capacitance[aF]
0.1	160.0
0.2	81.0
0.3	47.0
0.4	32.0
0.5	23.0

3. Create capacitance functions in terms of wire spacing from extracted capacitance values (called “fitting curve”).

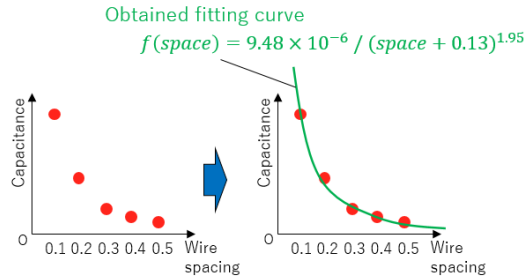
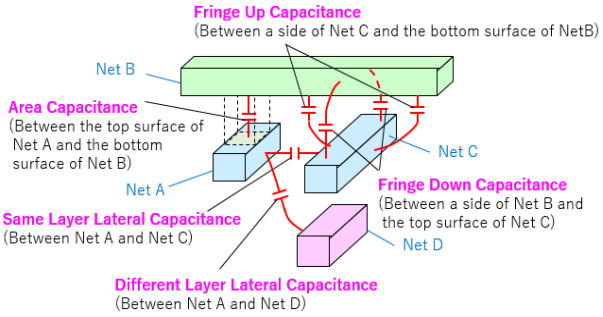


Figure 2. Getting a fitting curve.

4. Output the obtained fitting curves as a Hipex capacitance extraction rule file.

In general, the following kinds of capacitance between wires exist. All of them can be defined within Hipex capacitance extraction rule file.



Capacitances to substrate also exist for Area Capacitance and Fringe Down Capacitance.

Figure 3. Kinds of capacitances.

2. Conventional Parasitic Extraction Rule File and its Issues

Conventional parasitic extraction rule files created by Exact include the following variety of rules.

- Area Capacitance
- Same Layer Lateral Capacitance
- Fringe Up Capacitance
- Fringe Down Capacitance

This rule set appears good enough, since all of the capacitances shown in the previous section (except for Different Layer Lateral Capacitance) are included. However, the capacitance between two nets should be smaller when other wire exists on the upper or lower layer. Also, lateral capacitance and fringe capacitance should depend on the wire width of nets.

3. Measures for Accuracy Improvement

To solve the issues in the conventional capacitance extraction rule file, we apply the following measures.

- **Add the Dependency on Upper/Lower Layer Wire Existence**

Use Exact layout models to take the following cases into account.

- Wire exists neither on upper nor lower layer
- Wire exists only on upper layer
- Wire exists only on lower layer
- Wires exist both on upper and lower layers

For example, we use layout models that have the following cross-sectional views for Same Layer Lateral Capacitance extraction rule creation:

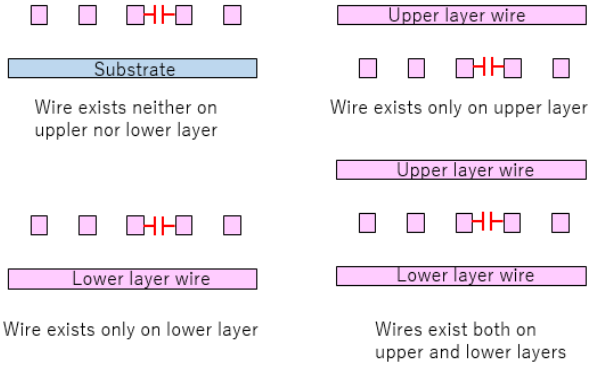


Figure 4. Layout models for Same Layer Lateral Capacitance rule creation.

By doing this, a rule with multiple case-divided commands is created. Those commands are described by only one command in the conventional rule file.

- **Add Different Layer Lateral Capacitance Extraction Rules**

Add an Exact layout model with the following cross-sectional view:

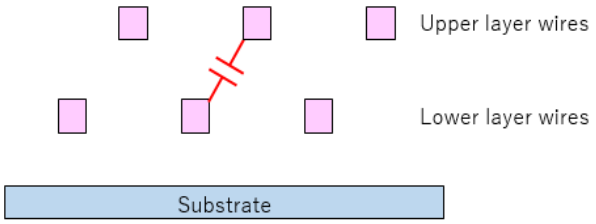


Figure 5. Layout model for Different Layer Lateral Capacitance rule creation.

In addition to the one above, layout models for the following cases are added to make the rules have dependency on upper/lower layer wire existence:

- Wire exists only on upper layer
- Wire exists only on lower layer
- Wires exist both on upper and lower layers

- **Add Dependency on Wire Width**

Because lateral capacitance is calculated not only between sides of wires but also top and bottom surfaces, its extraction rule has dependency on wire width of nets.

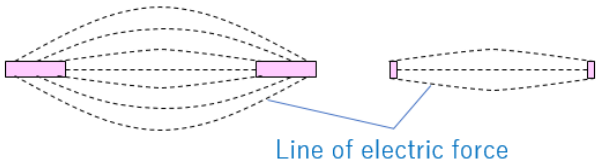


Figure 6. Wire width dependency of lateral capacitance.

In addition to that, fringe capacitance has a trade-off relationship with area capacitance. Its capacitance varies depending on the wire width as follows:

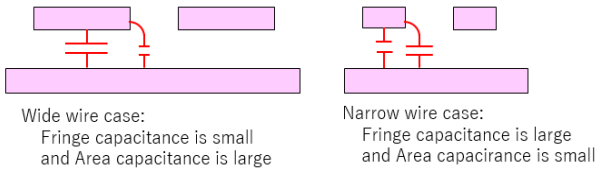


Figure 7. Trade-off between fringe and area capacitances.

To realize the wire width dependency for these rules, we modify the script for Exact to take a list of wire width variety, so that Exact generates basic layouts having wires of such widths. The Hipex tool itself was enhanced to recognize each wire width at fringe capacitance extraction command.

- **Add Contact Capacitance Extraction Rule**

Since the distance between diffusion contact and poly on a gate of MOS transistor is relatively short, it is predicted that the capacitance between them is too large to ignore. However, conventional Hipex extraction rules do not contain a rule for contact because there are no established ways to realize it. If both sides of poly and diffusion contact are facing each other at the same height range, it is likely that the capacitance between these sides is dominant. Therefore, the Same Layer Lateral capacitance rule for poly is suited for it. Copy the Same Layer Lateral Capacitance rule for poly and use it as the Different Layer Lateral Capacitance rule between poly and contact.

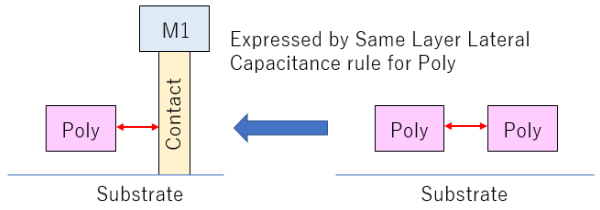


Figure 8. Contact rule definition.

4. Measure the Accuracy Improvement

Compare all capacitances extracted from a sample layout by Clever with those of Hipex.

Sample process

Process node : 0.18um

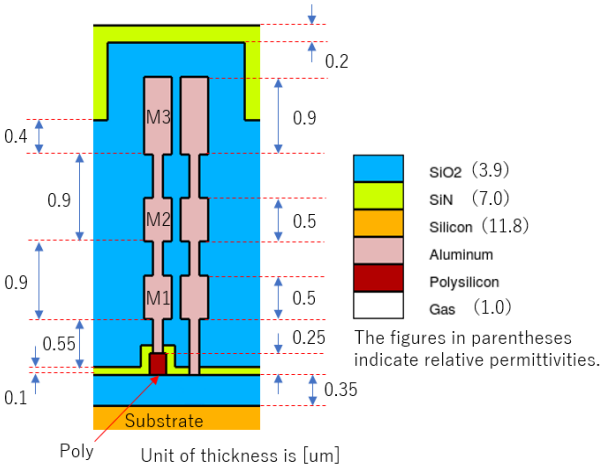


Figure 9. Cross-section of sample process.

Sample layout

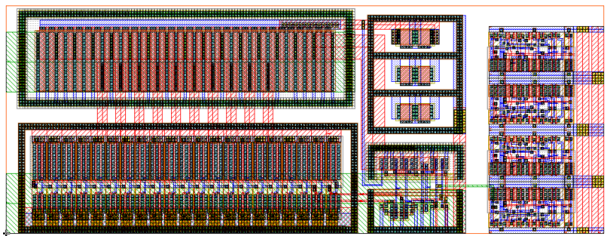


Figure 10. Sample layout.

Specification	Values
Circuit	VCO(Voltage-Controlled Oscillator)
Size	Width 100[um], Height 38[um]
Number of Transistors	256
Number of Nets	102

Table 1. Specifications of the sample layout.

How to run Clever

Treat P-substrate, N-well, and Diffusion areas as conductors because they are connected to Power/Ground. To avoid electrical short among them, make short gaps and fill them with a material of very low permittivity. Because the permittivity is extremely low (relative permittivity = $1x10^{-20}$), the capacitance related to parts of conductor covered with such material will be almost 0[F].

We need to be careful to extract capacitances of only required parts of the layout. The capacitance between poly on a gate and the diffusion area is normally included in the SPICE simulation model of each device. Therefore, Clever needs to be executed with the following steps to avoid double counting of capacitances.

Step 1. Do Type_A, Type_B, and Type_C with Clever.

Step 2. Get reference capacitances from Clever for each pair of nets with the following calculation.

Cap_ref = Cap_A – Cap_B + Cap_C

where, Cap_ref is the reference capacitance by Clever, Cap_A is the capacitance extracted at Type_A execution, Cap_B is the capacitance extracted at Type_B execution, and Cap_C is the capacitance extracted at Type_C execution.

Part of layout to be covered by low permittivity material / Things to be done	Type_A	Type_B	Type_C
Bottom surface of Poly on a gate	Covered	Covered	Covered
Side surfaces of Poly on a gate	-	-	-
Surface of Poly other than above	-	Covered	Covered
Top surface of P-substrate	-	Covered	Covered
Top surface of N-well	-	Covered	Covered
Top surface of Diffusion area	-	-	Covered
Top surface of channel area	-	-	Covered
Side surface of contact	-	Covered	Covered
Bottom surface of Metal1	-	Covered	Covered
Replace dielectrics of Metal1 or above with low permittivity one	No	Yes	Yes

"-" : Not-covered

Table 2. Parts of layout covered with low permittivity material.

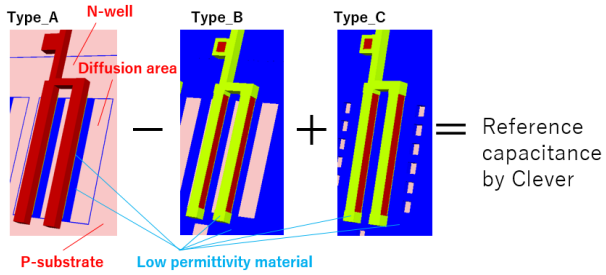


Figure 11. Reference capacitance calculation.

At Type_B, only two types of capacitances are extracted since most of conductor surfaces are covered with low permittivity material:

i) The one between poly on a gate and Diffusion

ii) The one between polys on gates

Subtracting Type_B from Type_A leads to over subtraction of ii), but only ii) is extracted at Type_C and added to it, it comes to the consequence of removing only i). The subtraction and addition are valid because, if the low permittivity material is thin enough covering part of a conductor does not affect electric field of other parts.

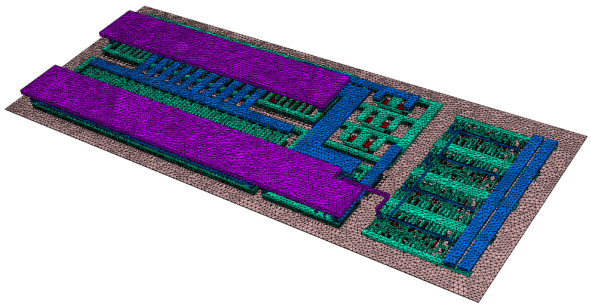


Figure 12. 3D structure created by Clever.

How to Run Hipex

Run Hipex C with both capacitance extraction rule files applied (the conventional extraction rule file and the one with all measures described in section 3).

5. Results

The results are as follows.

Tool names (Rule file used)	Run time	Maximum memory used
Clever	5h 18m	119.7G
Hipex (Conventional rule file)	0h 0m 8s	81M
Hipex (New rule file)	0h 0m 19s	106M

Used 10 cores for parallel processing at Clever.
Used single core for Hipex.

Table 3. Run time and Maximum memory used.

<Execution environment>

CPU : Intel Xeon X5690 @ 3.47GHz (12 cores)

Memory : 141.9GB

OS : Red Hat Enterprise Linux Workstation 6.5

Since it's rule-based, Hipex uses less memory and has shorter run times than Clever.

The run time of a new rule file is larger than conventional one because the new one:

- Has additional rules for upper/lower layers
- Has fringe capacitance extraction commands looking at wire widths (enabled by enhancement of Hipex)
- Has contact rule

Statistics

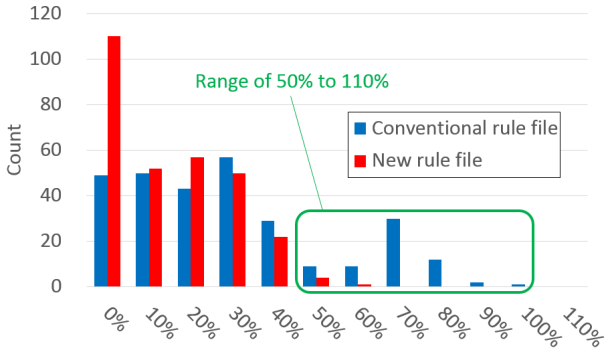


Figure 13. Histogram of relative errors (absolute values) of extracted capacitances.

- Relative error is defined as follows:

Rel_e (%) = (Cap_h - Cap_ref) / Cap_ref × 100

where

Rel_e is relative error

Cap_h is a capacitance extracted by Hipex

Cap_ref is the reference capacitance by Clever

- Showing below 120%.

The number of capacitances that have relative error (absolute value) of 120% or higher:

Conventional rule file: 5

New rule file : 0

- Counted only the capacitances whose reference capacitance by Clever is $1x10^{-17}$[F] or higher.

Comparison items	Hipex (Conventional rule file)	Hipex (New rule file)
Number of Capacitances extracted (It is 367 at Clever.)	311	320
Maximum of relative errors (absolute values) *1	394.9%	63.3%
Average of relative errors (absolute values) *1	37.7%	19.1%
Standard deviation of relative errors (absolute values) *1	37.4%	14.2%

*1: Counted only the capacitances whose reference capacitance by Clever is $1x10^{-17}$ or higher

Table 4. Statistics.

The number of capacitances appearing at the range of 50% to 110% obviously decreased with Hipex (new rule file). Average and standard deviation of relative errors are also greatly improved.

Scatter diagrams by Belledonne

Belledonne enables us to see the differences between reference netlist and another one in parasitic values by plotting them to a scatter diagram.

For the capacitances extracted by Clever having values smaller than Cth, the capacitance in another netlist is taken as good by Belledonne if the absolute error (absolute value of subtraction) is $1x10^{-17}$ or smaller. (We used this value this time. It is modifiable).

For the capacitances extracted by Clever having values of Cth or larger, the capacitance is taken as good by Belledonne if its relative error(absolute value) is 30% or smaller (We used this value this time. It is modifiable). Cth is defined as $1x10^{-17}$ / 0.3 in this case.

In Belledonne, the capacitances that are considered good are plotted as blue points. Ones considered bad are plotted as red points. If a capacitance exists only in one netlist, it will be plotted as $1x10^{-19}$[F] in another netlist.

You can see the difference in capacitance is much smaller when using the new Hipex rule.

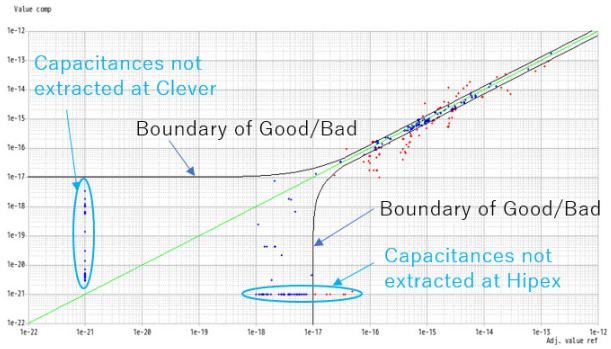


Figure 14. Clever (horizontal axis) vs Hipex conventional rule (vertical axis).

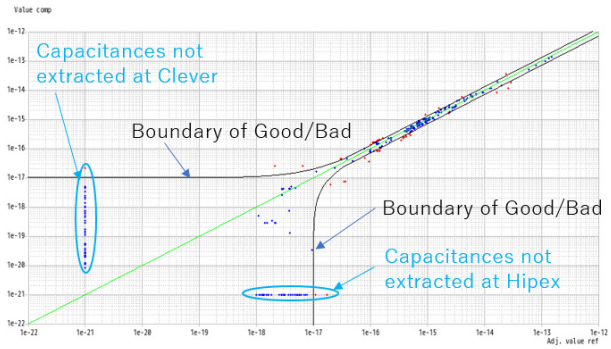


Figure 15. Clever (horizontal axis) vs Hipex new rule (vertical axis).

6. Conclusion

By applying measures for accuracy improvement to capacitance extraction rules, and with the improvement of Hipex itself, the extracted capacitances of Hipex are closer to those of the 3D field solver-based parasitic extractor Clever.

Using 3D field solver tools with large layout is unfortunately unrealistic in terms of memory usage and run time, but Hipex has a potential to do it with accuracy, using a reasonable amount of memory and run time.