

Hipex RC: 寄生容量抽出精度改善への取り組み事例

はじめに

Hipex RCは、ブロックレベルの小規模なデザインからフルチップレベルのデザインまで対応した、ルールベースの寄生素子抽出ツールですが、その精度は、抵抗、および、容量の抽出ともに、使用されるルールファイルの質に大きく依存します。そのため、ルールファイルの作成方法を工夫することで、抽出精度の向上が期待出来ます。

近年の微細プロセスにおいては、寄生素子の回路動作への影響が顕著なため、レイアウトから精度良く寄生素子抽出を行うことが重要です。

今回、容量抽出ルールファイルの作成において、いくつかの抽出精度向上のための施策を適用し、また、Hipex RCツール自体の改良も行われました。

この新しい容量抽出ルールファイルを使用し、効果の確認を行いましたので、その方法、および、結果をお伝えします。

1. Hipex RCの容量抽出ルールファイル

シルバコ PDKに含まれるHipex RCの容量抽出ルールファイルは、ルールに関する既存の情報が参照可能な場合は、その情報を用いて作成されますが、他の場合はExactを使用して作成します。プロセス情報さえあれば、Exactを用いて、あらゆるプロセス・テクノロジーの容量抽出ルールファイルを作成することが可能です。

Exactは、次の入力データを使用します。

- ・ レイアウト・モデル
- ・ 配線幅
- ・ 配線間隔のバラエティ
- ・ プロセスの縦構造情報（プロセス・ステップ、膜厚など）
- ・ 素材の比誘電率

そして、Exactは次の手順で容量抽出ルールファイルを生成します。

1. レイアウト・モデルに配線間隔のバラエティを適用して、様々な基本レイアウトを作成します。

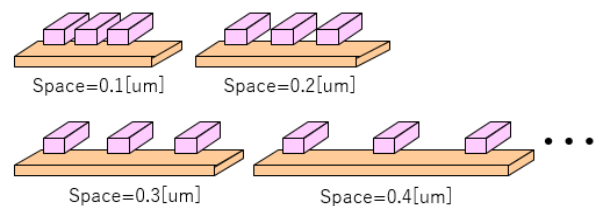


Figure 1. 生成された基本レイアウト

2. 各レイアウトに対して、2D/3Dフィールドソルバ寄生素子抽出ツール Cleverを実行して配線間の高精度な寄生容量抽出を行います。

| 配線間隔 [um] | 抽出された容量 [aF] |
|-----------|--------------|
| 0.1 | 160.0 |
| 0.2 | 81.0 |
| 0.3 | 47.0 |
| 0.4 | 32.0 |
| 0.5 | 23.0 |

3. 抽出された容量値を元に、配線間隔の関数（フィッティングカーブ）を作成します。

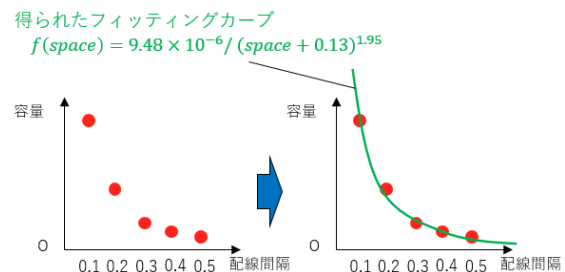
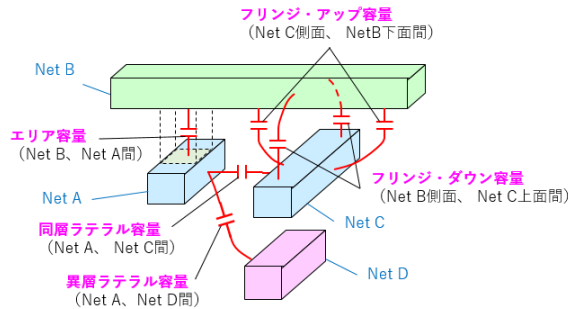


Figure 2. フィッティングカーブの生成

4. 得られたフィッティングカーブを、Hipex RCの容量抽出ルールファイルとして出力します。

一般的に、配線間の容量の種類として次のようなものがあり、Hipex RCの容量抽出ルールファイル内で定義することが可能です。



エリア容量、フリンジ・ダウン容量については、サブストレートとの容量も存在します。

Figure 3. 容量の種類

2. 従来の容量抽出ルールファイルと、その課題

Exactを使用して作成された従来の容量抽出ルールファイルには、次の種類の容量抽出ルールが含まれていました。

- エリア容量
- 同層ラテラル容量
- フリンジ・アップ容量
- フリンジ・ダウン容量

しかしながら、他の配線が上層や下層に存在している場合には、通常、配線間の容量は影響を受けて小さくなります。また、ラテラル容量や、フリンジ容量は、抽出対象の配線幅にも依存しますが、これらは考慮されていません。

3. 抽出精度向上のための施策

従来の容量抽出ルールファイルの問題点を解決するため、今回、その作成過程において、次のような施策を行いました。

・ 上層、下層の配線有無に対する依存性を付与

Exact実行時に容量抽出を行うレイアウト・モデルとして、次のケースを再現したものを使用しました。

- 上層、及び、下層に配線が存在しない場合
- 上層のみに配線が存在する場合
- 下層のみに配線が存在する場合
- 上層、及び、下層の両方に配線が存在する場合

例えば、同層ラテラル容量抽出ルール作成の際には、次のような断面構造を持つレイアウト・モデルを使用しました。

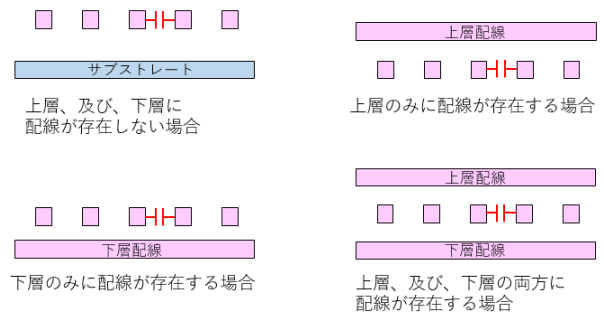


Figure 4. 同層ラテラル容量ルールのためのレイアウト・モデル

上層、下層を考慮したルールとしたことで、従来のHipex RCルールファイルでは1つのコマンドで記述されていたものが、場合分けされた複数のルールで記述されることとなります。

・ 異層ラテラル容量抽出ルールの追加

Exact実行時のレイアウト・モデルとして、次のような断面構造を持つものを追加しました。

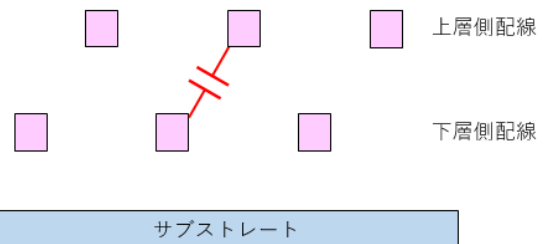


Figure 5. 異層ラテラル容量ルールのためのレイアウト・モデル

異層ラテラル容量抽出ルールについても上層、下層の配線有無に対する依存性を持たせるため、上層に配線がある場合、下層に配線がある場合、上下層ともに配線がある場合を再現したレイアウト・モデルも追加しました。

・ 各容量抽出ルールへの配線幅依存性の付与

ラテラル容量ルールについては、配線の側面だけでなく、上面、下面による容量への寄与も含むため、その値は配線幅に依存することになります。

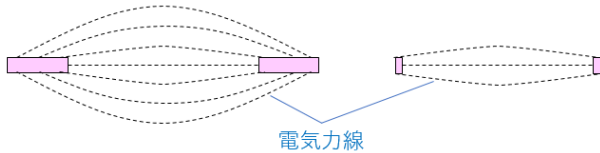


Figure 6. ラテラル容量の配線幅依存性

また、フリンジ容量ルールはエリア容量とのトレードオフの関係にあり、配線幅に応じて次のように容量値が変化することになります。

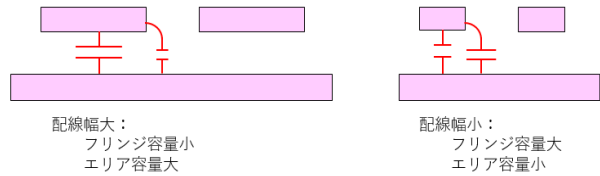


Figure 7. フリッジ容量とエリア容量のトレードオフ

これら容量抽出ルールにおける配線幅依存性の実現のため、Exact実行時に配線幅のバラエティを入力情報として与え、レイアウト・モデルから各配線幅を持つような基本モデルが生成されるようにしました。また、Hipex RCツール自体も、フリッジ容量抽出コマンドにて、レイアウト上の各配線幅が認識できるように改良が行われました。

・ コンタクト容量抽出ルールの追加

MOSTトランジスタの拡散層コンタクトは、ゲート上のPolyとの距離が近く、これらの間の容量は無視できないほどの大きさであることが予想されますが、コンタクト容量抽出ルールを実現する方法が確立されていなかったため、従来のHipex RCルールにはコンタクト容量抽出ルールは含まれていませんでした。

しかしながら、Polyと拡散層コンタクトは、Polyと同じ高さの範囲で側面が向い合わせになっているため、側面間の容量が支配的となり、Polyの同層ラテラルルールと同じもので近似できることが予想されました。

そのため、今回Polyの同層ラテラルルールを、Polyとコンタクトの間の異層ラテラルルールとして複製して、これをコンタクト容量抽出ルールとしました。

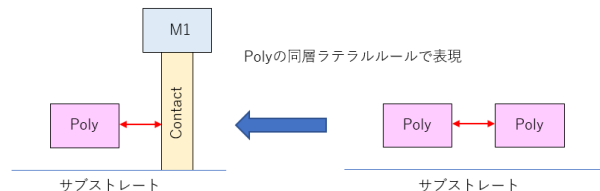


Figure 8. コンタクト容量抽出ルールの定義

4. 精度向上の確認方法

サンプルプロセス、および、サンプルデザインを使用して、各ネット間の抽出容量をCleverとHipex RCで比較しました。

サンプルプロセス

プロセスノード: 0.18um

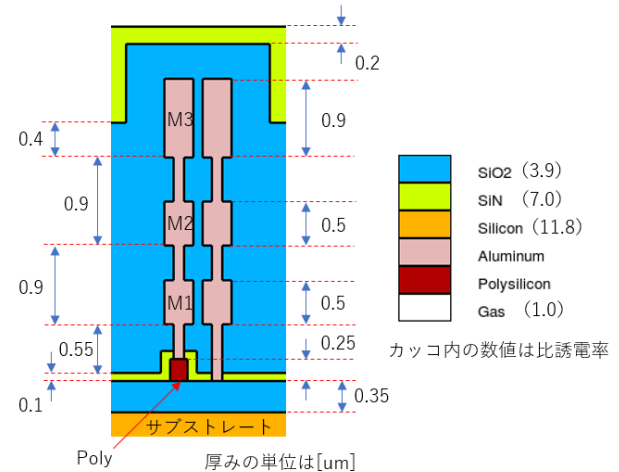


Figure 9. サンプルプロセスの断面構造

サンプルレイアウト

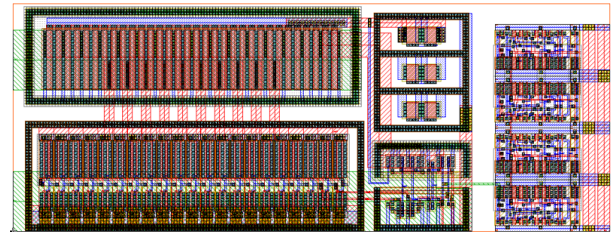


Figure 10. サンプルレイアウト

| 項目 | 内容 |
|---------|------------------------------------|
| 回路 | VCO(Voltage-Controlled Oscillator) |
| サイズ | 幅 100[um], 高さ 38[um] |
| トランジスタ数 | 256 |
| ネット数 | 102 |

Table 1. サンプルレイアウトの仕様

Cleverの実行方法

P型サブストレート、Nウェル、拡散領域は、それぞれ電源/グラウンドに接続されるため、導体として扱いました。そして、これらのショートを防ぐために、これらの間に細い隙間を作り、その間を比誘電率の非常に小さい材料で埋めました。

低誘電率材料の比誘電率は 1×10^{-20} としており、この材料で覆われた導体の部分に関わる容量は、0[F]に近い値となります。

また、通常、ゲート上のPolyと拡散層との間の容量は、各デバイスのシミュレーションモデルにてモデル化されています。そのため、容量の二重カウントを防ぐ目的で、次の手順でCleverを実行しました。

手順1 : Type_A、Type_B、Type_CをそれぞれCleverにて実行します。

手順2 : 各ネット間の容量について、

$$Cap_ref = Cap_A - Cap_B + Cap_C$$

を計算し、これをCleverによる参照容量とします。

但し

- Cap_ref : Cleverによる参照容量
- Cap_A : Type_Aにおいて得られた容量
- Cap_B : Type_Bにおいて得られた容量
- Cap_C : Type_Cにおいて得られた容量

加えていますので、結果としてはゲート上Polyと拡散領域の間の容量だけを除いた容量が、Cleverによる参照容量として得られていることとなります。

なお、低誘電率材料で覆うことにより、他の部分の電界には影響を与えないため、このような容量の加減算を行うことができます。

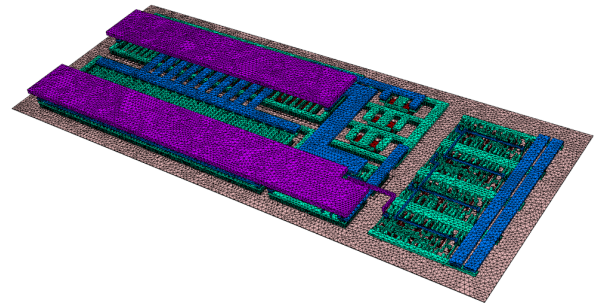


Figure 12. Cleverにより生成された3D構造

Hipex RCの実行方法

従来の容量抽出ルールファイル、および、抽出精度向上のための施策を行って作成した容量抽出ルールファイルのそれぞれでHipex Cを実行しました。

5. 結果

比較結果を以下に示します。

| 実行ツール (使用ルールファイル) | 実行時間 | 使用メモリ |
|------------------------|-----------|--------|
| Clever | 5h 18m | 119.7G |
| Hipex RC (従来抽出ルールファイル) | 0h 0m 8s | 81M |
| Hipex RC (新抽出ルールファイル) | 0h 0m 19s | 106M |

Cleverは10コアによる並列処理実行
Hipex RCはシングルコア実行

| 低誘電率材料での被覆対象部分/実施事項 | Type_A | Type_B | Type_C |
|--------------------------|--------|--------|--------|
| ゲート上Polyの下面 | ○ | ○ | ○ |
| ゲート上Polyの側面 | × | × | × |
| 上記以外のPolyの表面 | × | ○ | ○ |
| P型サブストレート上面 | × | ○ | ○ |
| Nウェル上面 | × | ○ | ○ |
| 拡散領域上面 | × | × | ○ |
| チャンネル領域上面 | × | × | ○ |
| コンタクト側面 | × | ○ | ○ |
| Metal1下面 | × | ○ | ○ |
| Metal1以上の絶縁体を低誘電率材料に置き換え | 実施せず | 実施 | 実施 |

○ : 覆った
× : 覆わなかった

Table 2. 低誘電率材料での被覆を行った箇所

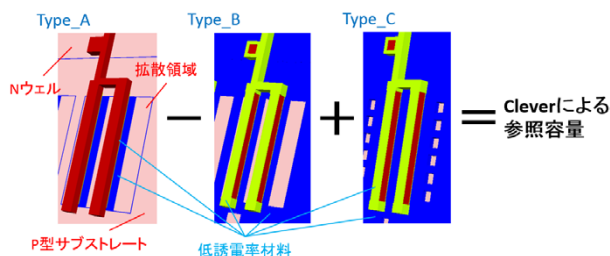


Figure 11. 参照容量の算出

Type_Bでは、上述のように各部分を低誘電率材料で覆っているため、ゲート上Polyと拡散領域の間の容量、および、ゲート上Polyどうしの間の容量が抽出されていることとなります。

これを、Type_Aの結果から差し引いていますので、ゲート上Polyどうしの間の容量を引き過ぎてしまっていますが、Type_Cではゲート上Polyどうしの間の容量のみを抽出しており、この容量を

Table 3. 実行時間と最大メモリ使用量

<実行環境>

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

メモリ : 141.9GB

OS : Red Hat Enterprise Linux Workstation 6.5

ルールベース寄生抽出ツールのため、Cleverに比べてHipex RCの方が使用メモリが少なく、短い時間で実行可能であることが分かります。新抽出ルールファイルでは、上層、下層の状況に応じたルールが増えており、また、Hipex RC自体がフリンジ容量抽出コマンドで配線幅を検出するように改良されていること、および、コンタクト図形の検出が行われるため、実行時間は従来抽出ルールファイル使用時に比べて増えています。

統計量

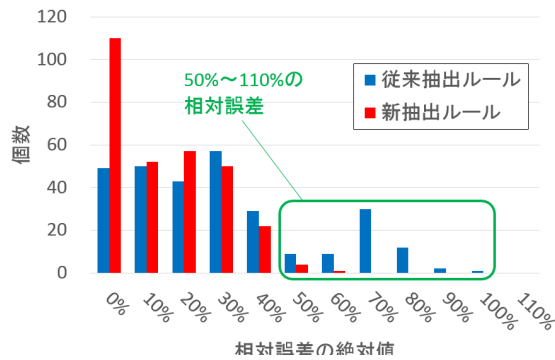


Figure 13. 相対誤差（絶対値）のヒストグラム

- 相対誤差は、次のように定義されます。

$$\text{Rel}_e (\%) = (\text{Cap}_h - \text{Cap}_{\text{ref}}) / \text{Cap}_{\text{ref}} \times 100$$
 但し、
 Rel_e : 相対誤差
 Cap_h : Hipex RCにより抽出された値
 Cap_{ref} : Cleverにより抽出された参照容量を示しています。
- ヒストグラムでは、相対誤差120%未満のみを表示しています。
 相対誤差（絶対値）が120%以上の容量の数は、それぞれ次の通りです。
 従来抽出ルール： 5
 新抽出ルール： 0
- ヒストグラムでは、Cleverでの抽出値が 1×10^{-17} [F]以上のものをカウントしています。

| 比較対象 | Hipex RC (従来抽出ルールファイル) | Hipex RC (新抽出ルールファイル) |
|------------------------------|---------------------------|--------------------------|
| 抽出された寄生容量数 (Cleverでは 367) | 311 | 320 |
| 相対誤差（絶対値）の最大（※1） | 394.9% | 63.3% |
| 相対誤差（絶対値）の平均（※1） | 37.7% | 19.1% |
| 相対誤差（絶対値）の標準偏差（※1） | 37.4% | 14.2% |

※1： Cleverによる参照容量が 1×10^{-17} [F]以上の容量のみを計上
 Table 4. 統計量

ヒストグラムより、主に50%~110%の相対誤差を示すものが、明らかに減っていることが分かります。また、相対誤差（絶対値）の平均、標準偏差ともに、大幅な改善となりました。

Belledonneによる散布図

Belledonneを使用することで、2つのネットリストに含まれる寄生素子の値を散布図に表示して、値のズレ具合を視覚的に確認することが出来ます。

Cleverによる参照容量が閾値Cthよりも小さいものは、対応するHipex RCによる容量値の絶対誤差（Cleverによる参照容量値との差分の絶対値）が 1×10^{-17} 以下（今回、この値を使用しましたが、変更可能です）の場合、BelledonneによりGOODと判定されます。

また、Cleverによる参照容量が閾値Cth以上のものは、Hipex RCによる容量値の相対誤差（絶対値）が30%以下（今回、この値を使用しましたが、変更可能です）の場合、BelledonneによりGOODと判定されます。

閾値Cthは、今回の場合 $1 \times 10^{-17} / 0.3$ で定義されます。

Belledonneでは、GOODと判定された容量は青色の点でプロットされ、BADと判定された容量は赤色でプロットされます。容量が片方のネットリストにしか存在しない場合は、もう片方のネットリストにて、 1×10^{-19} [F]としてプロットされます。

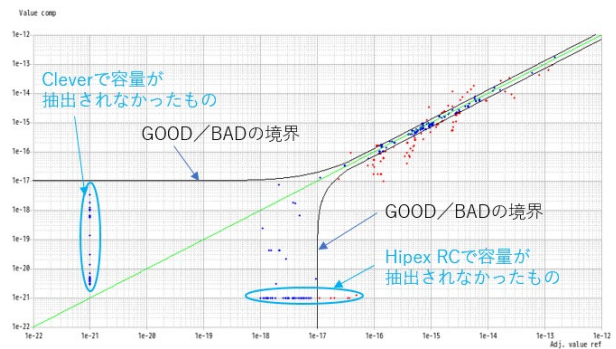


Figure 14. Clever(縦軸) vs Hipex RC従来抽出ルール (横軸)

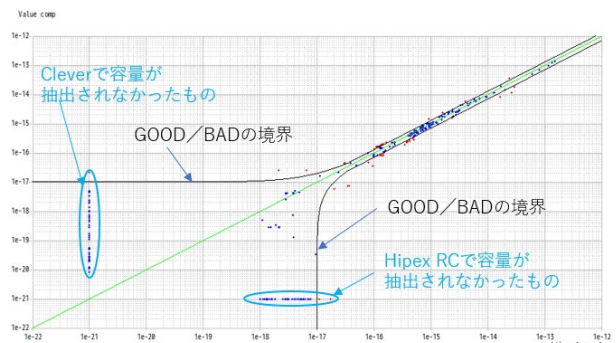


Figure 15. Clever(縦軸) vs Hipex RC新抽出ルール (横軸)

新抽出ルールファイルを使用した方が、Cleverとの差が小さくなっていることが一目で分かります。

6. 結論

Hipex RCの容量抽出ルールファイルへの精度向上のための施策適用、および、Hipex RC自体の改良により、従来に比べて3Dフィールドソルバ(Clever)による高精度な抽出結果に近づきました。

大規模なレイアウトで3Dフィールドソルバを実行することは、残念ながら使用メモリ、実行時間の面から現実的ではありませんが、Hipex RCには大規模デザインから現実的な実行時間、メモリ使用量で、十分な精度の寄生素子抽出を行うポテンシャルがあります。