

寄生素子を含む回路のシミュレーション および RCL リダクション

はじめに

設計ジオメトリが縮小し、回路速度が高速化するにつれ、寄生容量および抵抗はますます重視すべき事項になりました。寄生素子により、回路の振る舞いが大幅に変化し、正常な動作の回路から全く機能しない回路へ変化する可能性があります。シンプルな 4 ビット・シフト・レジスタにより、どのように寄生素子が回路の出力に影響を与えるかを容易に実証できます。

バックグラウンド

これまで、寄生素子を追加すると、容易に素子の合計数が 1 桁増加するので、寄生素子を含む回路のシミュレーションは非常に時間がかかりました。下記のシンプルな入力デッキ例では、最初は素子数がわずか 104 個でしたが、素子抵抗と容量を追加すると、素子の合計数が 947 個まで増加しました。数多くの素子が追加されることで、一般的な抵抗および容量のシミュレーションでさえ、実行時間が大幅に増加します。しかし、シミュレーション精度を維持しながら、SmartSpice 内蔵 RCL リダクション・ツールを使用して、シミュレーション時間を大幅に短縮できます。

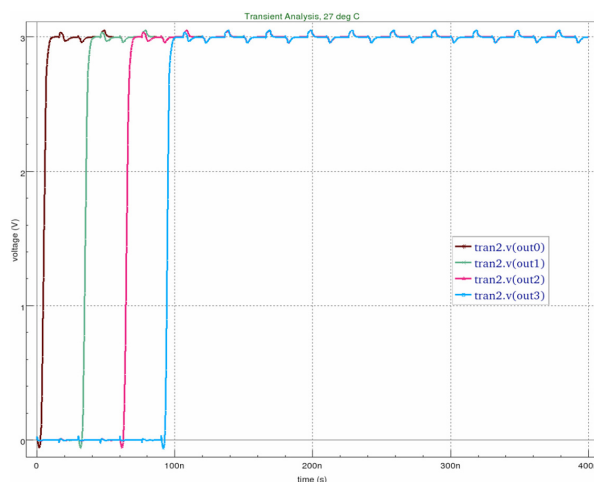


図 1. 寄生素子を含まない回路のシミュレーション

シミュレーション

Simucad の Logic Demo PDK に含まれているモデルを使用して、入力デッキを実行しました。各 DFF の出力時に電圧に対するシミュレーション結果を観察すると、DFF を通して信号が最終の出力ポート(out3)に伝搬していることがわかります。図 1 を参照してください。

物理設計の抽出およびバックアノテーション方法については、アプリケーションノートの『ポスト・レイアウト・シミュレーションのための寄生素子のバックアノテーション』を参照してください。レイアウトが完成した後、寄生素子が抽出され、バックアノテートされました。同じタイミングで回路を再実行した結果を図 2 に示します。

明らかに、レイアウト配線や相互接続から抵抗および容量を追加すると、回路が機能できない遅延が発生します。要求された仕様を満たすためにレイアウトを変更するか、物理設計にマッチするためにタイミングを調整する必要があります。入力信号の周期を倍にすると、図 3 に示すように DFF を通して再び伝搬するための十分な時間が信号に与えられます。

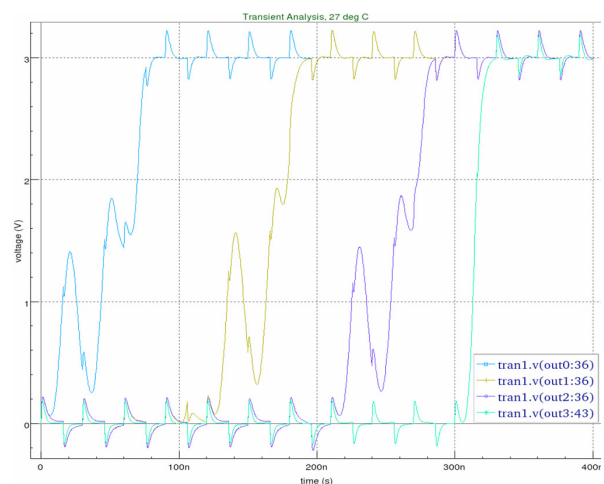


図 2. 寄生素子を含む回路のシミュレーション

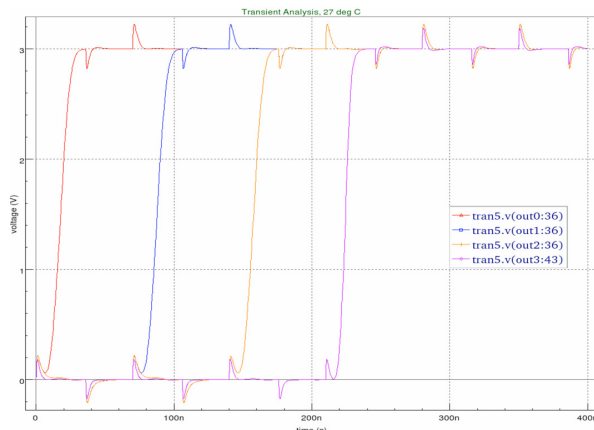


図3. 寄生素子と遅い入力信号を持つ回路のシミュレーション

信号

レイアウト前の回路とレイアウト後の回路のシミュレーション時間の差異は、レイアウト後の回路シミュレーションがおよそ2.5倍長かかります。しかし、SmartSpice 内蔵 RCL リダクション・ツールを使用して、シミュレーション時間を大幅に短縮できます。レイアウト後の回路は、コマンドライン・スイッチ `-rclevel 1` を使用して再実行しました。このコマンドラインにより、サブサーキットの RCL リダクションやポスト・フィルタリングのオプションが有効になります。これらのオプションを設定してシミュレーションを実行すると、過渡解析時間が 30% 短縮され、シミュレーション結果はわずか 0.007% 変化するだけです。2 セットのレイアウト後のシミュレーションを同じグラフにプロットすると、違いがほとんどないことがわかります。より大きい規模の回路またはより長いシミュレーションで、シミュレーション時間をさらに短縮できます。

まとめ

レイアウト後の回路シミュレーションは、設計プロセスの非常に重要なステップです。シミュレーション結果からわかるように、寄生容量と抵抗は著しい影響を与えます。これまで、レイアウト後のシミュレーションには時間がかかりましたが、SmartSpice RCL リダクション・ツールを用いることで高精度を保持しながらシミュレーション時間を大幅に短縮できます。

`-rclevel` には、さまざまなオプションが数多くあります。`-rclevel` スイッチを用いたシミュレーション実行の詳細については、『SmartSpice ユーザーズ・マニュアル Vol.1』の「第13章」を参照してください。

Simulated schematic input deck

```
* Schematic name: shift_register
*
R5 OUT3 GND 10Meg
R6 NET4 GND 10Meg
R7 NET5 GND 10Meg
R8 NET3 GND 10Meg
R9 NET1 GND 10Meg
V1 VDD GND DC 3
V2 C GND PULSE(0 3 0 0.01ns 0.01ns 15.98ns 30ns)
V3 NET7 GND DC 3
X1 C NET7 OUT0 NET1 VDD GND DFF
X2 C OUT0 OUT1 NET3 VDD GND DFF
X3 C OUT1 OUT2 NET5 VDD GND DFF
X4 C OUT2 OUT3 NET4 VDD GND DFF
*
* Schematic name: DFF
*
.SUBCKT DFF C D Q Q_bar VDD VSS
*
X1 NET13 Q_bar Q VDD VSS NAND2
X2 Q NET10 Q_bar VDD VSS NAND2
X3 NET10 D NET8 VDD VSS NAND2
X4 NET12 C NET13 VDD VSS NAND2
X5 NET8 NET13 NET12 VDD VSS NAND2
X10 NET13 C NET8 NET10 VDD VSS NAND3
*
.ENDS DFF
*
* Schematic name: NAND2
*
.SUBCKT NAND2 IN1 IN2 OUT VDD VSS
*
M1 OUT IN1 NET2 VSS CMOSN L=2U W=5U
+ AD=27.5P AS=27.5P PD=21U PS=21U M=1
M2 NET2 IN2 VSS VSS CMOSN L=2U W=5U
+ AD=27.5P AS=27.5P PD=21U PS=21U M=1
M3 OUT IN1 VDD VDD CMOSN L=2U W=10U
+ AD=55P PD=31U AS=55P PS=31U M=1
```

```
M4 OUT IN2 VDD VDD CMOS L=2U W=10U
+ AD=55P PD=31U AS=55P PS=31U M=1
*
.ENDS NAND2
*
* Schematic name: NAND3
*
.SUBCKT NAND3 IN1 IN2 IN3 OUT VDD VSS
*
M1 OUT IN1 NET2 VSS CMOSN L=2U W=5U
+ AD=27.5P AS=27.5P PD=21U PS=21U M=1
M2 NET2 IN2 NET1 VSS CMOSN L=2U W=5U
+ AD=27.5P AS=27.5P PD=21U PS=21U M=1
M3 OUT IN1 VDD VDD CMOS L=2U W=10U
+ AD=55P PD=31U AS=55P PS=31U M=1
M4 OUT IN2 VDD VDD CMOS L=2U W=10U
+ AD=55P PD=31U AS=55P PS=31U M=1
M5 NET1 IN3 VSS VSS CMOSN L=2U W=5U
+ AD=27.5P AS=27.5P PD=21U PS=21U M=1
M6 OUT IN3 VDD VDD CMOS L=2U W=10U
+ AD=55P PD=31U AS=55P PS=31U M=1
*
.ENDS NAND3
*
* End of the netlist
*
* Markers to save
*
.INC `../models/SBCD.lib'
.IC V(OUT0)=0
.IC V(OUT1)=0
.IC V(OUT2)=0
.IC V(OUT3)=0
.TRAN 0.1ns 400n
.SAVE ALL(I) ALL(V)

.END
```