連　載

第8回（最終回）

スイッチ・レベルのシミュレーション

渡辺　玲

Verilog-HDL によるコンピュータ設計

CD-ROM 設計データ

前回の番外編で話題を一つ積み残してしまいました．Verilog-HDLによるスイッチ・レベル・シミュレーションの話です．

そこで，今回は番外編パート2をお送りします．

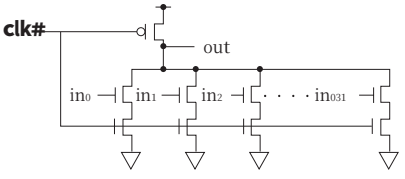
スイッチ・レベル・シミュレーション

スイッチ・レベルのシミュレーションとは一口に言うと，トランジスタをもっと単純なスイッチに見立てて，論理回路の動作を確認する手法です．

●時代の流れ

連載本編（第1回から第6回まで）でプロセッサを設計する流れ（設計フロー）に沿って，その各ステップでVerilog-HDLという便利な道具がどのように使われるのかという話をしてきました．そのとき，最後にスイッチ・レベル・シミュレーションの話をしようかどうか迷いました．そして結局，やめました．

それと言うのも，スイッチ・レベル・シミュレーションは最近ではあまり使われなくなった手法だからです．実際筆者が現在，設計しているプロセッサの設計フローにもスイッチ・レベルのシミュレーションは出てきません．



〔図1〕 トランジスタ・ベース設計の例

昔（十数年前），筆者がこの世界に入った頃は，チップはフルカスタムで作るのが普通でした．一部では自動配置配線ツールが試験的に使われていましたし，HDLによる回路の記述も始められていましたが，論理合成ツールはまだ研究室で評価されている段階でした．

当時，ASICの世界ではゲート・アレイが主流でしたが，スタンダード・セルが台頭する兆しも見えていました．しかし，筆者は汎用LSIを設計していたので，ASICの世界の動きとはまったく無縁な世界に住んでいました．

その頃の筆者にとってスタンダード・セルとは，あくまで論理ゲートのたんなる「見本」でした．いわゆる雛形です．設計時に，セル内のトランジスタのサイズを細かく変え，最適化を図る（チューニングする）のはサイジングと言って回路設計の重要な仕事の一つでした．

またレイアウト時には，マスク設計者の人と相談し，セルのレイアウトの形を崩して，いかに高密度にシリコン・ダイにトランジスタを詰め込むかといったこともよく検討しました．その当時のマスク設計者にとっても，セルはたんなるレイアウトの「見本」だったのです．

●トランジスタ・ベース設計

ところが，それだけ自由度があっても，求める性能やサイズに収まらないときがありました．その場合はどうするかというと，トランジスタ・レベルで論理回路を設計するのです．論理ゲートで設計するよりトランジスタで設計するほうが自由度が高く，より小さくて，またより高速に動作する論理回路が作れました（**図1**）．とくにデータパスのALUなどは設計者の腕の見せどころでした．

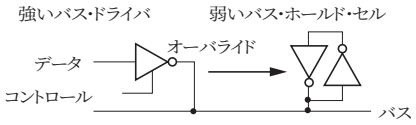
Verilog-HDLはそんな時代に産声を上げたので，ビヘイビア・レベルやゲート・レベルのシミュレーションだけでなく，トランジスタ・レベルのシミュレーションもしっかりとサポートしなければなりませんでした．

今は，ASICだけでなく，汎用LSIもスタンダード・セルで作る時代です．論理合成ツールや自動配置配線ツールが進歩し，タイミングもセル・ベースで解析するほうがはるかに便利なので，この手法が好まれます．トランジスタ・ベースの設計はすっかりすたれました．

老兵はただ去りゆくのみなのでしょうか？しかし，老兵の昔話に耳を傾けるのもたまには良いものだと思います．知識のすそ野が広がります．今回は，そんな話です．

●トランジスタ・レベルとの違い

トランジスタ・ベースで論理回路を設計するのなら，トランジスタ・レベルのシミュレーションが必要…といきたいところですが，ちょっと待ってください．



〔図2〕 バス・ホールド・セル

トランジスタの動作を正確に表現するシミュレータにはSpiceがあります．これはアナログ的なシミュレータです．ものすごく正確に電圧や電流や抵抗をシミュレーションしますが，ものすごく遅いのが欠点です．

普通，トランジスタ・ベースで設計した論理回路の場合，そのクリティカル・パスの解析はSpiceでやります．しかし，その論理回路全体の動作の試験には，Spiceは遅くてとても使えません．

そこで考えられたのが，トランジスタをスイッチと見なしてシミュレーションを行う手法です．論理回路は連続値（つまりアナログ）では表現されないので，トランジスタをアンプとして扱う必要はありません．基本的にはON/OFFの2値（つまりディジタル）で表現されるので，トランジスタをたんにスイッチとして扱えばいいのです．

●Verilog-HDLの登場前後

Verilog-HDLの普及以前は，スイッチ・レベルの専用シミュレータが全盛でした．筆者はMOSSIMを使っていました．トランジスタ・ベースで作った論理回路を回路図エディタに手作業で入力し，できた回路図からネットリストを抽出し，それをMOSSIMでシミュレーションして論理回路の動作を確認していました．

RTLモデルの検証用には社内製のRTLシミュレータがありました．しかし，MOSSIMを走らせる前に，まずこのRTLシミュレータでMOSSIMが使う入出力ベクタを抽出しなければならず，とても厄介でした．

ところがVerilog-HDLが普及して，この面倒な状況は一変しました．Verilog-HDLシミュレータはRTLモデルでもトランジスタを含んだネットリストでも平気でシミュレーションすることができます．Verilog-HDLシミュレータによるRTLモデルとネットリストのミックス・レベル・シミュレーションはとても便利で強力でした．

〔表1〕 信号の強さ

強さ	種類
7	Supply Drive
6	Strong Drive
5	Pull Drive
4	Large Capacitor
3	Weak Drive
2	Medium Capacitor
1	Small Capacitor
0	High Impedance

2．スイッチのモデリング

スイッチ・レベルのシミュレーションではまずトランジスタをスイッチとしてモデリングする必要があります．

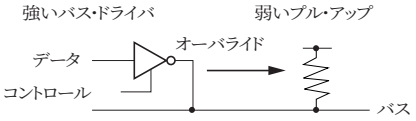
●トランジスタの強さとは

スイッチをモデリングするのに，ON/OFFで表すのは，一番簡単な方法です．しかし，実際のトランジスタにはチャネル幅（W）とチャネル長（L）があり，W/Lのサイズ比が，ON抵抗の違いになります．トランジスタはON時に抵抗がゼロで，OFF時に抵抗が無限大の理想スイッチではありません．

ON抵抗といっても，I_D対V_Dのトランジスタ動作曲線を見てもわかるように，実際はリニアな抵抗よりもっと複雑な曲線（ダイオード特性）になります．しかし，見積もりをするときなどは，これを直線で近似して，たんなる抵抗と見なします．

このON抵抗を，別の言い方で「トランジスタの強さ」と表現します．ON抵抗が小さいほど，トランジスタは強くなります．つまり，「W/Lが大きい（チャネル幅が大きくてチャネル長が小さい）→ON抵抗が小さい→トランジスタは強い」という関係になります．

プロセス技術の進化の歴史は，言い換えればこのチャネル長（L）が短くなる歴史で，「1.2 μm→0.8 μm→0.5 μm→0.35 μm→0.25 μm」と各世代ごとに「2の−0.5乗」（0.707）にほぼ近い割合で縮小しています．



〔図3〕 プル・アップ

もしトランジスタの強さを変えないと仮定すると，W/L比は一定に保たれることになります．よってチャネル幅（W）のほうもチャネル長（L）と同じ割合で縮小されます．この効果を，面積比に換算すると各世代ごとに面積がほぼ半分（$2^{-0.5^2}$は0.5）になる計算になります．

●信号の強さ

トランジスタをスイッチでモデリングするときに，この強さの違いをうまく表せないと，トランジスタの強さに依存して動く回路を正確にシミュレーションできなくなります．

たとえば，バスをホールドするセル（いわゆるジャム・ラッチ）があります．弱いトランジスタでバスをホールドしておいて，強いトランジスタでバスの値を書き換えるわけです（**図2**）．オーバライドするわけですが，論理的にではなく，回路的に，すなわちトランジスタの力と力の勝負でオーバライドされます．

モデリングのやり方はいろいろあって，シミュレータごとに違うのですが，Verilog-HDLでは，信号の強さという概念を導入して，この問題に対処しています（**表1**）．

Verilog-HDLでは8種類の信号の強さが定義されています．たとえば，Strong Drive(7)の“H”の信号とWeak Drive(3)の“L”の信号がぶつかった場合は，Strong Driveの勝ちで値は“H”になります．先のバス・ホールド・セルはこれでちゃんと表現されます．

●キャパシタンスやプルアップ

信号に強さを付けることのメリットはトランジスタの強さだけではなく，それで容量（キャパシタンス）の大小やプルア