

ASICサインオフシミュレータemslの高速化

7L-1

下出 協子 奥田 亮輔 小栗 澄男
三菱電機(株) システムLSI開発研究所

1.はじめに

ASICサインオフシミュレータとして実用しているemslは、インタプリタ方式の内製論理シミュレータである。emslの高速化を行なった結果、従来比約8倍、一般に最も高速と言われているEDAベンダ製のインタプリタ方式論理シミュレータの約2倍の実効速度を達成した。

2.ASICサインオフシミュレーション

ASICサインオフシミュレーションとは、チップ製造直前に実行される最終動作確認である。製造されるチップがこのシミュレーション通りの動作をすることを保証しなければならないため、次に述べるような精密な条件で実行される。

(1)ゲートレベルシミュレーション

回路は、andゲートやFF(フリップフロップ)等のセルの接続で記述されている。一部、RAM、ROMなど機能ブロックのHDL記述を含む場合もある。

(2)レイアウト設計後の詳細遅延モデル

実配線の寄生容量を考慮して、(入力ピン>出力ピン)のすべての組合せに対して、“立上り遅延値”と“立下り遅延値”が割り当てられる(pin_to_pin遅延モデル)。例えば図1の素子では、(A->Y)、(B->Y)、(C->Y)それぞれのピンの組合せに対して、立上り遅延値と立下り遅延値が与えられる。入力ピンAが変化した場合で出力ピンYが立上り変化する場合、(A->Y)に対して割り当てられた立上り遅延値を用いる。emslでは、さらに配線抵抗を考慮してネットに遅延値を付加する場合もある。

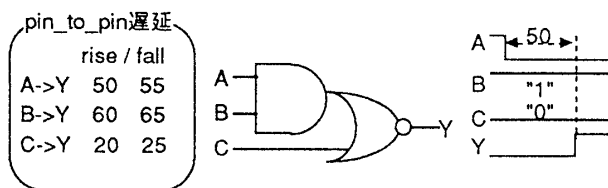


図1 pin_to_pin遅延モデル

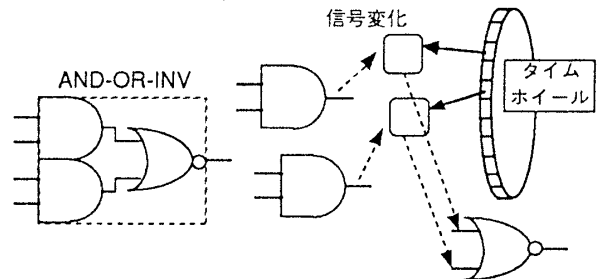


図2 EDAベンダ製シミュレータでのセル内の信号変化の伝搬

(3)タイミング検証

FFが誤動作を起こさないよう、セットアップ時間やホールド時間が十分にとられているか等のタイミング検証を行なう。emslでは、LSIテストのピン間スキューまでを考慮した厳密なタイミング検証[1]を行なっている。

3.ASICベンダ内製シミュレータの優位点

EDAベンダ製のシミュレータにはAND・OR等の基本的なプリミティブしか用意されておらず、サインオフシミュレータとして使用するにはそれらを組み合わせてセルを定義するという方式を取らざるを得ない[2]。例えば、図2のAND-OR-INVセルの場合は2個のANDと1個のNORで実現される。ところが、ANDの出力からNORの入力への信号変化の伝播は、汎用スケジューリング機構(タイムホイールなど)を介して行なわれるので、処理時間のオーバーヘッドが大きい。

一方、ASICベンダ内製シミュレータでは、当該ベンダのセルだけをサポートすればよい。emslでは、200種類以上ある全てのセルをそれぞれ1つのプリミティブとしてC言語で最適にコーディングしてある。図2のセルの模擬では、2回のAND演算の後、(汎用スケジューラを介さず)ただちにNOR演算が実行される。このように各セルをプリミティブとしているASICベンダ内製シミュレータの方が、EDAベンダ製のシミュレータに比べ処理速度の点で有利である。我々は、多くのプリミティブを効率よく開発できる環境を整えている[3]。

4.高速化改良のポイント

今回、我々は主にソースコードの再コーディングを行なうことによりemslの高速化を達成した。この際工夫した点は、無駄な処理を省くこと、特にメモリアクセス回数を削減すること、キャッシュメモリのヒット率を向上させることである。さらに、関数(サブルーチン)呼出し回数を削減したり、関数の処理量を最適にする工夫も行った。以下、具体例を説明する。

(1)セルの模擬におけるメモリアクセス回数の削減

前の時刻の入出力ピンの状態などセルの模擬に必要なメモリ上の情報を、数ワードのビット列にバックした。移植性の問題はあるが、数回のメモリアクセスで全ての情報をCPUレジスタ上にロードすることができるようになった。

(2)データ構造改良によるキャッシュヒット率向上

キャッシュメモリへのアクセス時にミス(データがキャッシュに無いこと)が発生すると、当該データの前後数ワード(SPARCstation2では8ワード)のデータがまとめてメインメモリからキャッシュに転送される。よって、連続してアクセスするデータのアドレスが近い程、キャッシュのヒット率が向上して高速に動作できる。

我々は、構造データのメンバの順序を最適に並べ替える工夫の他に、図3(a)のリスト構造を図3(b)のように変更することも行った。図3(a)のように次の"ノード"(リストの要素)を指すポインタがノード自身に存在するリスト構造よりも、図3(b)のようにノードへのポインタを配列にしたリスト構造の方が、アクセスするデータが固まって配置されるので高速にアクセスできる。

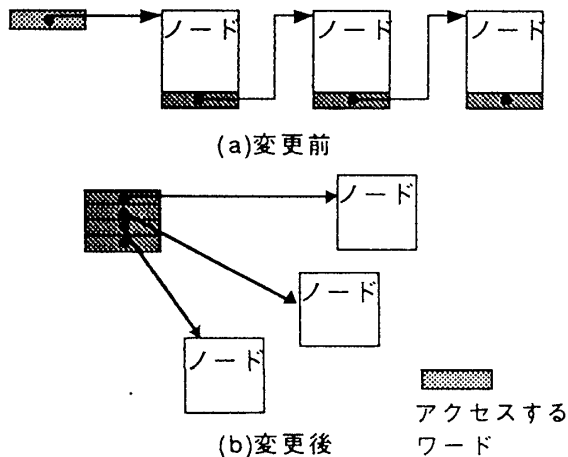


図3 リスト構造の変更

表1 サインオフシミュレーション実行時間(秒)

チップ (gate数)	emsl		EDAベンダ製 シミュレータ
	改良後	改良前	
1(250K)	15375	-----	27730(1.8)
2 (51K)	276	2366(8.5)	1003(3.0)
3 (18K)	1125	13042(11.6)	1819(1.6)
4 (54K)	919	9760(10.6)	1500(1.6)
5 (10K)	103	712(7.0)	-----
6 (4K)	186	1251(6.7)	-----
7 (7K)	118	874(7.4)	-----
8 (8K)	170	1037(6.1)	-----
9 (10K)	522	3453(6.6)	-----
10(44K)	1362	11325.4(8.3)	-----
平均		8.1倍	2.0倍

(括弧内は、改良後のemsl実行時間との比)

(3)関数の呼出し回数や処理量の調節

呼び出しレベルが最下位の関数の処理を上位の関数に移すことで関数を減らし、関数の呼び出し回数を削減した。また、全関数において全ローカル変数がCPUレジスタに割り当てられるよう、コンパイラが出力するアセンブラコードを参考にして関数の処理量を調節した。これは、関数の処理量が大きすぎてローカル変数がメモリ上に割り当てられると、処理速度が遅くなるからである。

5.適用結果

emslのサインオフシミュレーション実行時間を実チップ10品種について表1に示す。従来比で平均約8倍の高速化を達成した。これは、インタプリタ方式で最も高速と言われているEDAベンダ製ゲートレベル論理シミュレータと比較しても、平均2倍程度高速である。

6. おわりに

emslでは、ASICベンダ内製のシミュレータである利点を生かし、ライブラリ内の全てのセルの模擬処理を最適にコーディングしてある。今回、高速化を行った結果、従来比約8倍、EDAベンダ製論理シミュレータの約2倍の高速化を達成した。

謝辞

常に有益なご助言を頂く当社佐藤興二博士に感謝します。

参考文献

- [1]S.Oguri, et al., EDAC93, pp.460-464,1993.
- [2]Verilog-XL Reference Manual Version 1.6.
- [3]奥本他,「ASICサインオフシミュレータemslのプリミティブテスト手法」,情処第49回全国大会(1994).