

スーパースカラ計算機における検証カバレッジ解析の一手法

2E-3

*木戸 弘二 宮原 真次 嶋田 浩巳

富士通株式会社

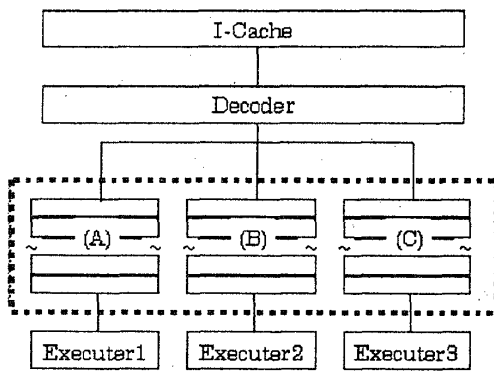
1 はじめに

近年、コンピュータシステムの高速度/大規模化に伴い、ハードウェア内部の制御は複雑化する傾向にある。また、CPUの処理能力を向上させるためにスーパースカラ機構が導入され並列実行の制御論理が増大したことにより、CPU制御論理の正当性を確認するための検証期間の長期化が懸念される。最近、大型高速エミュレータが開発され、論理設計段階で大規模な回路の設計検証が可能となってきたが、実マシンの1/1000程度のスピードしかなく検証プログラムに対する検証精度の向上が重要な課題となっている。

そこで、ハードウェア論理設計の正当性を保証する検証プログラムの検証精度を評価する一手法として、ハードウェアの内部トレース情報を基に検証カバレッジを解析する手法を研究・試作した。

2 対象機構

検証対象としたスーパースカラ機構の概略を図1に示す。



(A)ReservationStation-1 (B)ReservationStation-2
(C)ReservationStation-3

図1: スーパースカラ機構概略図

点線部の ReservationStation を検証対象とした。
 -ReservationStation を FULL にし負荷を与える試験
 -ReservationStation に溜まったコマンドキューを例外等でキャンセルを発生させ負荷を与える試験

One method of the inspection coverage analysis in the Superscalar computer.
 Kouji Kido, Shinji Miyahara, Hiromi Shimada

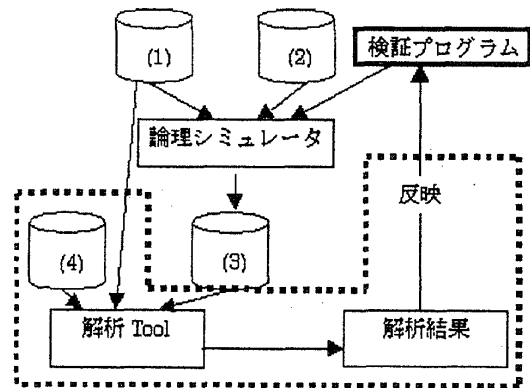
3 カバレッジ解析の手法

● 解析の目的

解析目的は以下の2つである。一つは、「検証プログラムが対象としている事象発生の確認」である。これは、スーパースカラ機構に対する検証プログラムの意図した事象が発生しハードウェアに対する検証がされているかの確認を行い、検証プログラムに対してフィードバックを行う。もう一つは、「解析結果を基に冗長要素を省いた検証項目の生成」である。これは、検証プログラムの中で最も有効な部分(ユニット)を抽出し、それらの有効な検証ユニットを集約することで検証密度を高める。

● 解析システム構成

今回、採用した解析システムについての概略を図2に示す。点線部が今回開発した手法



(1) 論理回路ファイル (2) シミュレーション制御ファイル (3) トレースログファイル (4) 解析用ハードウェア信号ファイル

図2: 解析システム構成図

● 解析手法

[入力情報]

(a) ハードウェアシミュレータでスタンプされたハード信号遷移(0↔1)の時刻。(図2の(3))
 時刻:1Clock を更に細分割した時間

(b) 解析対象となるハードウェア信号。(図2の(4))
単一事象と組合せ事象の指定が可能である。

[出力情報]

入力情報を基に解析ツールにより以下の出力を得る。

- (a) 対象事象が発生した時刻 (Clock)
- (b) 対象事象の発生回数
- (c) 対象事象の連続発生回数
- (d) 対象事象の発生箇所 (Program Counter)

【結果の反映】

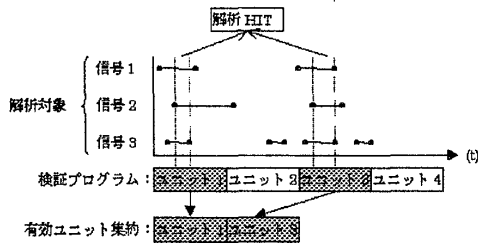
事象未発生

⇒ 検証プログラムにフィードバック

事象発生

⇒ 発生させた検証ユニットを抽出し、それらを集約して検証精度の高いプログラムを生成する。

一例として、解析結果からの有効ユニット抽出論理を図3に示す。



網掛け部が有効ユニットである

図3: 有効ユニット抽出論理図

- 今回対象とした解析
 - 一般的なスーパースカラ機構負荷 (Reservation Station FULL) に対する事象が発生しているかの解析。
 - 上記負荷状態に命令の実行 Cancel を付加した事象が発生しているかの解析。

4 評価 (有効性)

本手法で5本の検証プログラムに対するカバレッジ解析を行った結果を表1と表2に示す。解析事象は、各 ReservationStation(以下 RS) の FULL の組合せと RS-FULL かつ命令実行 CANCEL の組合せ事象である。

表1: RS FULL の組合せ

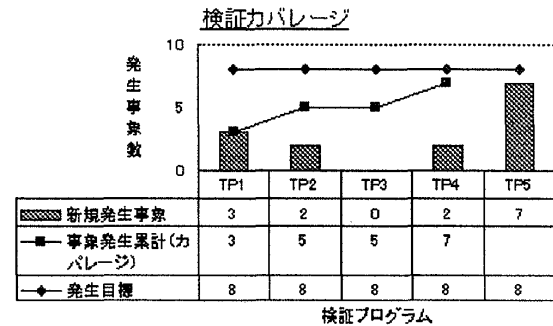
検証プログラム	解析事象			
	RS1/2	RS1/3	RS2/3	RS1/2/3
TP1	1(3)	-	2(4,5)	-
TP2	-	5(5,3,13,4,3)	-	-
TP3	2(5,11)	-	-	-
TP4	3(3,13,20)	-	-	1(2)
TP5	2(3,13)	2(13,4)	1(5)	1(2)

表2: RS FULL と CANCEL の組合せ

検証プログラム	解析事象			
	RS1/2+C	RS1/3+C	RS2/3+C	RS1/2/3+C
TP1	-	-	1(1)	-
TP2	-	2(1,1)	-	-
TP3	-	-	-	-
TP4	1(1)	-	-	-
TP5	1(1)	1(1)	1(1)	-

TP1、TP2、TP3は、上表に示した機構を狙った検証プログラムであったが、意図した事象が発生していない部分があった。これに対し、TP4というフィードバックプログラムを作成し未発生事象部分の検証が可能となった。また、各検証プログラムの有効ユニットを集めたTP5は、他TPより検証密度が高いことが実証できた。

以下に検証カバレッジ変化をグラフに示す。



今回対象とした検証プログラムはスーパースカラ機構の特殊条件を狙って開発したものである。RS1/2/3+Cの条件は、実行タイミング等が絡み、ハンドコードでは条件の作成が困難であった。この条件はRANDOM要素を含んだRUNNING系の検証プログラムで発生しえると考えており、今後評価を行っていく。

(参考: 解析実行環境)

マシン S-4/20H M125(Hyper SPARC 125MHz)
 搭載メモリ 340MB
 トレースログ 34MB
 解析時間 3分30秒

5 結び

今回開発したカバレッジ解析手法は、ハードウェア信号の組合せで検証カバレッジの解析を行うことを特徴とする手法である。最初の課題で述べた検証精度の向上という観点から評価すると、TP4とTP5の検証プログラムを生成できたことによって本手法の有効性の確認ができた。

更に、本手法を応用することによってハードウェアの論理がありえない動作をした場合のAssertion Checkの一部として適用できる。Assertion Checkは一般的にハードウェア論理に組み込まれていることが多いが、矛盾するハードウェア信号の組み合わせを定義し解析対象とすることにより論理矛盾チェックが可能と考える。