

VLSI テストデータ自動処理システムの開発

立石 昭光, 新田 進

㈱東芝 超L S I 研究所

本報告は、東芝標準テストデータインターフェース言語 (TSTL2 : Toshiba Standard Test data interface Language) ・テストデータベース・ TSTL2コンパイラ、TSTL2 逆コンパイラ、論理シミュレータ/機能シミュレータインターフェース、テストインターフェース等 V L S I テストデータ自動処理システムの開発について述べたものである。本システムではテストデータベース (TIF:Test data Interface File)を中心に構築し、東芝標準テストデータインターフェース言語によって、論理シミュレーション/機能シミュレーションやテスト用のデータ作成、テストの制約事項に対するエラーチェック等を自動的に行うことが可能となり、テストデータの移植性が向上し、L S I テスタのテストプログラムの開発期間が短縮された。

WGPA 39-4

“ A Test Data Generation System For VLSI ”

(in Japanese)

by Akimitsu TATEISHI and Susumu NITTA

(VLSI Research Center , TOSHIBA Corporation ,
1 Komukai-Toshiba-cho , Saiwai-ku , Kawasaki , Kanagawa , 211 , JAPAN)

This paper describes a test data generation system for VLSI. This system consists of Toshiba Standard Test data interface language (TSTL2) , TSTL2 compiler , TSTL2 reverse-compiler , a logic/function simulator interface , a tester interface and Test data Interface File (TIF) at the core . The same test description in TSTL2 may be compiled to run against logic/function simulations and testers. It has proved to be an effective tool for cutting testing time.

1. はじめに

近年、集積回路技術の進歩によりLSIの高集積化が著しい。しかし、LSIの大規模化は機能向上、装置の軽薄短小化の利点をもたらすのとは反対にテストを非常に困難なものにしている。

このため、各テストメーカーからも精度・機能面で非常に優れたATE (Automatic Test Equipment) が開発されているが、これらのテストはハードウェアのアーキテクチャー・機能・性能がそれぞれ異なるため様々な問題を生み出している。

例えば異なるテスト間でテストプログラムの交換を行おうとすると、非常に困難な作業が伴ない、個々にデータ交換のためのインターフェースプログラムを開発する場合には、機種異なるテスト毎に各テスト間のインター

フェースプログラムを作る必要がある。さらにテストプログラム自体の開発も高集積化・複雑化によって開発期間が長期化する傾向にある。

これらの問題点を解決するために、東芝では従来使用されていた東芝標準テストデータインターフェース言語 (TSTL : Toshiba Standard Test data interface Language) に基づいて 'TSTL2' を開発し、これをもとにテストデータベースの仕様を決定し、このテストデータベースを介して処理を行うための各種インターフェース (TSTL2コンパイラ、TSTL2 逆コンパイラ、テストインターフェース、TEPASインターフェース、論理シミュレータ/機能シミュレータインターフェース、テスト制約事項チェック、MTインターフェース) を開発した。

本報告では、そのシステム構成と概要について報告する。

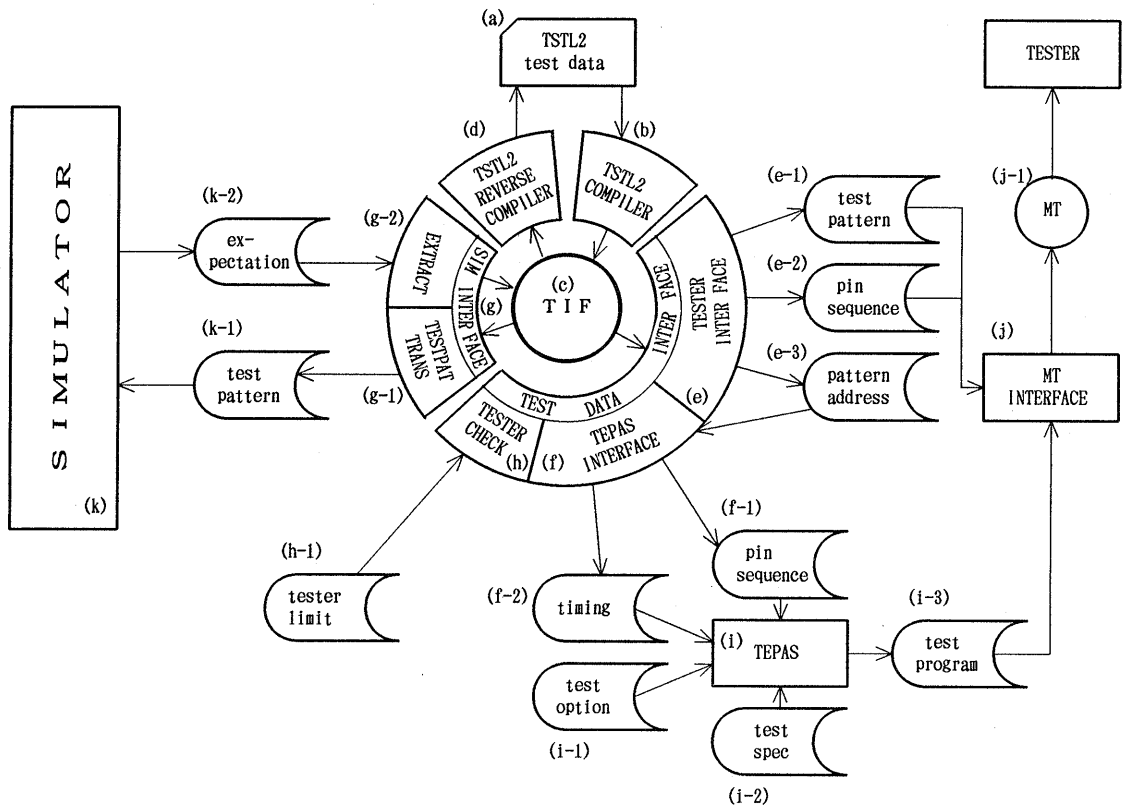


図 1

2. 概要

本システムは東芝標準テストインターフェース言語から論理シミュレータ・機能シミュレータ・故障シミュレータ用のデータとして入力パターン・期待値パターン・テスト用データを作成する機能を持つ。

各インターフェースプログラムは TIF (Test data Interface File) と呼ばれるテストデータベースを介して処理が行われる。したがってシミュレータとテスト間のデータ交換がスムーズに行える形態になっている。

図 1 は本システムのシステム構成図である。

TSTL2 で記述されたテストデータ (a) は、TSTL2 コンパイラ (b) によって TIF (c) に格納されテストインターフェース (e) ・ TEPAS インターフェース (f) ・シミュレータインターフェース (g) ・テスト制約事項チェック (h) の入力になる。

また、TIF (c) から TSTL2 逆コンパイラ (d) を通して、TSTL2 記述のテストデータ (a) を再生することも可能である。

テストインターフェース (e) からは、テスト用のテストパターンファイル (e-1) ・端子情報ファイル (e-2) や TEPAS インターフェース (f) にパターンアドレスの受け渡しをするためのアドレスファイル (e-3) が出力され、TEPAS インターフェース (f) からは、既に当社半導体事業本部で開発済みの TEPAS (Test Program Auto generation System) (i) の入力となる端子情報ファイル (f-1) 及びタイミングファイル (f-2) が出力される。

TEPAS (i) では、端子情報ファイル (f-1) ・タイミングファイル (f-2) の他に、人手によって作成された TEPAS のオプションを記述したオプションファイル (i-1) ・電源値などを記述したスペックファイル (i-2) を入力として、テスト用のテストプログラムファイル (i-3) を出力する。

出力されたテストパターンファイル (e-1) ・テストプログラムファイル (i-3) etc. は MT インターフェース (j) を通して各種テストのフォーマットに合った MT (j-1) を出力し、テストにかけられる。

シミュレータインターフェース (g) は、TIF のデータからシミュレーション用のデータを作成するパターン変換プログラム (g-1) と、シミュレーション結果から期待値を抽出する期待値抽出プログラム (g-2) からなっており TIF のデータからシミュレータ (k) のフォーマットに変換したデータ (k-1) を出力すると共に、シミュレーシ

ョン結果の期待値 (k-2) を TIF に戻す役割を果たしている。

テスト制約事項チェック (h) ではテスト制約事項が記述されたファイル (h-1) を参照してテストデータが各種テストで使用可能かどうかのチェックを行う。

3. 東芝標準テストインターフェース言語 (TSTL2)

TSTL の特徴としては、

- (1) 論理シミュレーション用のデータとして使用されると共に、デバイステストに使用可能である。
- (2) 記述は、入出力端子のタイミング記述とテストパターンの記述からなっている。

点であるが、実際にはシミュレーション中心であるため今回 TSTL2 としてテスト中心にテストとシミュレーション両方の入力に使用することを考え、TSTL2 として次の部分を拡張した。

- (3) function テスト、DC テスト、AC テストに対するブロック記述を導入した。(例 1)
- (4) テスタ名の記述が可能となった。(例 2)
- (5) テスタピン番号やパッケージピン番号の記述が可能となった。(例 3)
- (6) タイミングの切り換えが可能となった。(例 4)
- (7) テストパターンの圧縮、結合用に各種記述を用意した。

① SEQUENCE 文 (例 5)

この記述は部分的にパターンを作成した後で結合させたり既成のパターンを再利用する際に使用する。

② CONST 文 (例 6)

状態の変化しない固定信号値の端子の状態値をパターン別に指定する。

③ 16 進記述 (例 7)

パターンを短縮して記述できる。

④ MEASURE 文 (例 8)

DC テストのとき、どのパターンで、どの測定項目を、どの端子について行うかを指定する。

⑤ SUBPATT 文 (例 9)

なんども同じパターンを使用したり、既成のパターンを再利用する際に、サブルーチンパターン記述として使用する。

⑥ DO 文 (例 10)

繰返しパターンを記述する際に使用する。

⑦ SHIFT文 (例1 1)

パターンが規則的なシフトパターンの場合に使用する。

⑧ COMPATT文 (例1 2)

パターンが連続して出現している場合に使用する。

(8) スキャンデザイン用テストパターン記述を用意した。

```

FUNCTEST FC1 ;
}
ENDFUNC ;

DCTEST DC1 ;
}
ENDDC ;

ACTEST AC1 ;
}
ENDAC ;
    
```

例 1

```

TESTER TOS101 ;
    
```

例 3

```

PROBE A.B.C.D.E.,F.G ;
PACKAGE C.B.A.,F.G.,D.E ;
    
```

例 4

```

PROBE : テスターピン番号の並び
PACKAGE : パッケージピン番号の並び
    
```

```

TIMING TS1 ;
CYCLE 400 ;
}
ENDTIM ;
TIMING TS2 ;
CYCLE 300 ;
}
ENDTIM ;
ENABLE TS1 ;
}
ENABLE TS2 ; ←タイミング切換
}
    
```

例 2

```

ASSIGN A.B.C.D ;
CONST E-I.F-L ;
TESTPATT PAT1 ;
ENABLE TS1 ;
0011 ;
1100 ;
ENDTEST ;
    
```

例 6

```

TESTPATT PAT1 ;
ENABLE TS1 ;
$0F ;
$0E ;
$3F ;
ENDTEST ;
    
```

→ 等価

```

TESTPATT PAT1 ;
ENABLE TS1 ;
00001111 ;
00001110 ;
00111111 ;
ENDTEST ;
    
```

例 7

```

}
0011 ;
MEASURE(I) A.B ;
1011 ;
}
    
```

→ 後続のパターンで端子A,B
について11H.11Lの測定
を行う指定

例 8

```

SEQUENCE PAT1.(PAT2,PAT3) ;
ASSIGN A.B.C.D ;
TESTPATT PAT1 ;
ENABLE TS1 ;
0 1 0 1 ;
0 0 0 0 ;
ENDTEST ;
ASSIGN A.B ;
TESTPATT PAT2 ;
ENABLE TS1 ;
1 0 ;
1 1 ;
ENDTEST ;
ASSIGN C.D ;
TESTPATT PAT3 ;
ENABLE TS1 ;
1 1 ;
1 0 ;
ENDTEST ;
    
```

例 5

```

ASSIGN A.B.C.D ;
TESTPATT XXXX ;
ENABLE TS1 ;
0 1 0 1 ;
0 0 0 0 ;
1 0 1 1 ;
1 1 1 0 ;
ENDTEST ;
    
```

```

SUBPATT SUB1(JJJ,KKK,MMMM) ;
JJJ KKK 000 MMMM ;
ENDSPATT ;
000 010 111 01010 ;
REFER SUB1($7.101,LLHHH) ;
}
}
}
    
```

↓ 等価

```

000 010 111 01010 ;
111 101 000 LLHHH ;
}
    
```

例 9

4. TSTL2 コンパイラ

図1(b)のTSTL2コンパイラではTSTL2をコンパイルしTIFを作成する。TIFの構成としては、

- (1) 端子情報 (端子名, 種類, テスタ/パッケージ端子番号, 各種属性)
- (2) タイミング情報 (タイミング番号, 波形, デレイ, 波形幅)
- (3) パターン情報 (タイミングコール, サブルーチン情報, DC端子情報)
- (4) 繰り返し情報 (REPEAT)
- (5) スキャン記述情報

からなっている。

5. TSTL2 逆コンパイラ

図1(d)のTSTL2逆コンパイラではTIFからTSTL2記述のテストデータを復元することが可能である。TSTL2逆コンパイラの使用法としては、

- (1) テストデータのダンプをとる。
- (2) シミュレータインターフェースから出力された期待値を含むTIFから、TSTL2テストデータを作成する。
- (3) TSTL2テストデータを誤って消した場合等に、TIFからTSTL2テストデータを再生する。

際に有効である。

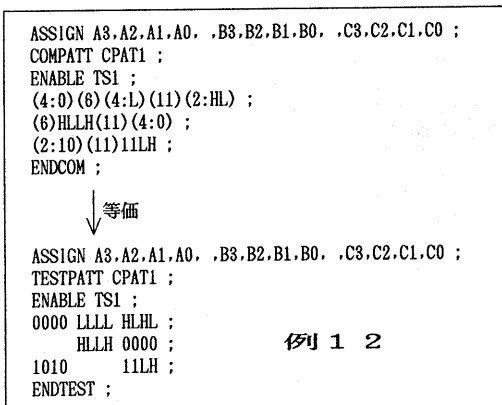
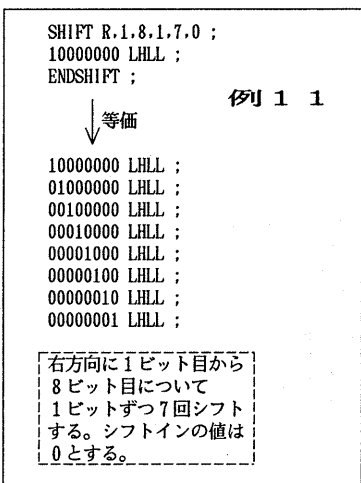
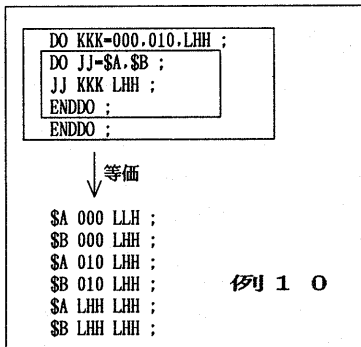
6. テスタインターフェース

近年、集積回路技術の進歩により、各テストメーカーからも精度・機能面で非常に優れたテストが開発されている。また一方ではこれらのテストが高価であるのに対して、量産用に、精度・機能面では劣るものの価格的に非常に廉価なテストが使用されている。

これらのテストはハードウェアのアーキテクチャー・機能・性能がそれぞれ異なるため様々な問題を生み出している。

例えば異なるテスト間でテストプログラムの交換を行おうとすると、非常に困難な作業が伴ない、個々にデータ交換のためのインターフェースプログラムを開発する場合には、機種の異なるテスト毎に各テスト間のインターフェースプログラムを作る必要がある。

これを解決する手段として、TIFを介して各種のテス



タにインターフェースする、テストインターフェースが開発された。

図1(e)のテストインターフェースはTSTL2コンパイラからの出力であるTIFから、テスト用のテストパターンファイル(e-1)・端子情報ファイル(e-2)やTEPASインターフェース(f)にパターンアドレスの受け渡しをするためのアドレスファイル(e-3)を出力する。

テストによってはスキャンデザイン用テストパターン記述ができるものがあるため、その対応も行っている。

テストパターンの各種テストへの対応はテスト毎に用意されたインターフェースプログラムによって行われる。

図2はテストインターフェースより出力されたテスト用テストパターンの一例である。

7. TEPASインターフェース

集積回路技術の進歩により、テストパターン開発と同様にテストプログラム自体の開発も高集積化・複雑化によって、開発期間が長期化する傾向にある。

TEPASインターフェース(f)からは、既に社内で開発済みのTEPAS(Test Program Auto generation System)(i)の入力となる端子情報ファイル(f-1)及びタイミングファイル(f-2)が出力される。

図1(f)のTEPASインターフェースは、TSTL2コンパイラからの出力であるTIFから、既存のテスト用テストプログラム発生システムTEPAS(Test Program Auto generation Sytem)の入力となる端子情報ファイル(f-1)及びタイミングファイル(f-2)を出力する。

```

# PATTERN # LP2 LP1 TESTER #
                                REM FC1
                                REM TESTER PIN-PINNAME
                                REM ( 0) 11 .12 .13 .14 .15
                                REM ( 5) 01 .02 .B1 .B2
                                REM PACKAGE PIN-PINNAME
                                REM ( 0) 15 .14 .13 .12 .11
                                REM ( 5) 02 .01 .B2 .B1
                                REM DATAHEAD:/FNCHEAD:FC1/SEQHEAD:FC1/PPHEAD:FC1 ;
0 LSET MA 000001110 ;
1 LSET MB 000001111 ;
2 LSET DA 111110000 ;
  REM SUBROUTINE ;
3 LSET I 000110000 ;
4 LSET RZ 111110000 ;
5 LSET ST 000000000 ;
6 LSET XOR 000000000 ;
  ENABLE TS1 ;
  ENABLE DA.MA ;
24 1 7 SET F 000000000 ;
25 2 8 SET F 000111101 ;
25 3 9 SET F 000011101 ;
  ENABLE MB ;
  REM TESTER # 10
26 4 10 SET F 010111100 ;
28 5 11 SET FC NORMAL 5 010011011 ;
12 LSET MB 000000000 ;
  ENABLE MB ;
30 6 13 SET F 000001111 ;
  REM SUBROUTINE S1 FROM PATTERN # 2 TO PATTERN # 3 ;
  REM REPEAT 5 TIMES FROM PATTERN # 5 TO PATTERN # 5 ;
  REM DATAHEAD:/DCCHEAD:DC1/SEQHEAD:DC1/PPHEAD:FC1 ;
0 LSET MA 000001110 ;
1 LSET MB 000001111 ;
2 LSET DA 111110000 ;
3 LSET I 000110000 ;
4 LSET RZ 111110000 ;
5 LSET ST 000000000 ;
6 LSET XOR 000000000 ;
  ENABLE TS1 ;
  ENABLE DA.MA ;
47 1 7 SET F 000000001 ;
54 2 8 SET F 000001100 ;
END ;

```

図 2

8. シミュレータインターフェース

図1 (g) のシミュレータインターフェースは、TIF のデータからシミュレーション用のデータを作成するパターン変換プログラム(g-1) と、シミュレーション結果から期待値を抽出する期待値抽出プログラム(g-2) からなっている。期待値抽出プログラムはシミュレーション結果から期待値を抽出すると共に、期待値(k-2) からTIFを再作成する。

シミュレータインターフェースでは、テストパターンのチェックとしてスパイクチェック・バスコンフリクトチェックも行っている。

9. テスタ制約事項チェック

テストの多様化に伴ないテストの制約事項も多種多様になりテストによってある機能・ない機能が入り組んできている。

図1 (h) のテスタ制約事項チェックはテスタ制約事項が記述されたファイル(h-1) を参照して TSTL2コンパイラからの出力であるTIF のデータが各種テストの制約事項(最大端子数, 最大パターン数etc.) を満足しているかどうかのチェックを行う。

図3は制限事項チェックテスタインターフェースより出力された出力リストの一例である。

10. まとめ

本システムは、論理シミュレータ等のCADプログラムとテスタ間のインターフェース及び自動テストプログラム生成が効率良く行え、簡易テスト等、異なるテストや、異なるシミュレータに対するテストデータのフォーマット変換をスムーズに行うことを目的として開発された。

その結果、テストデータベースを中心に置くことにより各インターフェースを有機的に結合し、拡張性のあるシステムとなり、また、シミュレータとLSIテスタ間のインターフェース・異機種テスタ間のインターフェースを確立することにより、テストプログラムの開発がスピードアップされ、さらに、テストの制約事項チェックも可能になった。

今後、さらにテストに対してより密接なCAT (Computer Aided Test) システムを開発するため、本システムをミニコンやEWS (Engineering Work Station) で稼働させる予定である。

//// TESTER NAME : XXXXXX ////

---- TEST PATTERN DATA ----

TEST NAME	PINCNT	PATCNT	PALCNT	PASCNT
FC1	0	0	0	0
DC1	0	0	0	0

MASK	DRIVER	ZCHECK	DCUMAX
0	0	-	0
0	0	-	0

TCAMAX	TSTMIN		
0	-		
0	-		

---- TIMING DATA ----

TIMING NAME	TMIN	DELTS	1TMMAX	RZMAX
TS1	0	X	0	0
TS2	0	0	0	0

NRZMAX	OTMMAX	TWMIN	SWMIN
0	0	0	X
0	0	0	0

IDEL	WDEL	ODEL	DELTC
0	0	0	0
0	0	0	0

DELTSW			
0			
0			

☒ 3

11. 参考文献

設計マニュアル テストデータインターフェース解説書
(TC17G005 SERIES) 朝東芝 半導体事業本部発行