PAPER Special Section on Test and Verification of VLSI

A DFT Selection Method for Reducing Test Application Time of System-on-Chips

Masahide MIYAZAKI†, Toshinori HOSOKAWA†, Hiroshi DATE†, Michiaki MURAOKA†, *Members*, *and* Hideo FUJIWARA††, *Fellow*

SUMMARY This paper proposes an SoC test architecture generation framework. It contains a database, which stores the test cost information of several DFTs for every core, and a DFT selection part which performs DFT selection for minimizing the test application time using this database in the early phase of the design flow. Moreover, the DFT selection problem is formulated and the algorithm that solves this problem is proposed. Experimental results show that bottlenecks in test application time when using a single DFT method for all cores in an SoC is reduced by performing DFT selection from two types of DFTs. As a result, the whole test application time is drastically shortened.
key words: *test scheduling, test access mechanism, wrapper, design for test*

1. Introduction

With the progress of the semiconductor process technology, the gate count of SoCs is increasing as large as one hundred million gates through the use of 90 nm process design rule toward 2010. As the size of the SoC is getting larger, the reduction of the design productivity will be the most important issue. The technologies that solve this issue are the design reuse methodology and design automation at the high level design phase. Research and development of these technologies is the key to innovate the SoC design methodology.

In order to reduce design time, SoCs consist of a large number of reusable cores. To test such SoCs, a test pattern is prepared for each core, and the modular testing of embedded cores is carried out. Effective modular test requires efficient management of the test resources for core-based SoCs. This involves the design of core test wrappers and TAMs (Test Access Mechanisms), and the scheduling of core tests. In recent years, many research works relevant to these issues have been presented.

Core test wrapper design and TAM design are important since they have impact on hardware overhead and test application time. There are three main approaches to achieve accessibility of embedded cores. The first approach is based on test bus architectures, where the cores are isolated from each other in test mode using a dedicated bus [1]–[3] around the cores to propagate test data. The second approach uses boundary scan architectures [4], [5] to isolate the core during test. The third approach uses core bypass mode [6] or transparency [7]–[9]. Wrapper and TAM design include wrapper optimization, core assignment to TAM wires, sizing of the TAMs, and routing of TAM wires. So, wrapper and TAM co-optimization approach [10] is one of the important subjects in modular testing.

The objective of test scheduling [11]–[13] is to minimize test application time under one or more of the following constraints: maximum TAM width and maximum allowed power consumption. Furthermore, optimal wrapper width selection and test scheduling techniques have been proposed [14], [15].

Most of the above research work assume scan design as a DFT of each core, or do not mention about the DFT method. To cope with the testing of large and complex SoCs, the modular testing of embedded cores have to be reevaluated. For a core, which is reused in a high level design methodology, the designer has to determine the DFT method along with the required quality and cost. For this reason, the technique of determining an SoC test architecture including DFT of each core, taking test cost and test quality into consideration, during the early design phase is needed.

In this paper, we present the DFT selection method for reducing test application time under the following constraints: maximum TAM width, maximum allowed power consumption, total area size, and test data size. The DFT of each core is chosen from scan design or non-scan DFT [16], [17].

This paper is organized as follows. In Sect. 2, a framework of SoC test architecture generation is proposed. In Sect. 3, the selectable DFT method and the precondition of this research work are described. In Sect. 4, the formulation of the DFT selection problem and an algorithm are proposed. It is followed by experimental results in Sect. 5. Finally, Sect. 6 concludes this paper.

2. A Framework of SoC Test Architecture Generation

In a high level design methodology, the designer has to determine the DFT method of each core along with the required quality and cost. So we propose a new framework that includes DFT selection phase in the SoC early design phase. Figure 1 shows our SoC test architecture generation framework. The framework consists of the following stages.

In the first stage, the test cost information on each core is estimated, and the result is output to the test cost information database. In addition, for each core, the test cost information of two or more DFTs are estimated. Test

Manuscript received June 23, 2003.
Manuscript revised October 9, 2003.
†The authors are with STARC, Yokohama-shi, 222–0033 Japan.
††The author is with NAIST, Ikoma-shi, 630-0101 Japan.

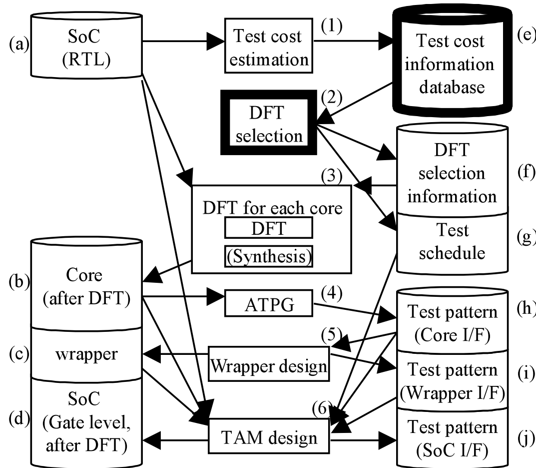

Fig. 1 Framework of SoC test architecture generation with consideration to DFT selection.

cost information includes the following information.

- Test application time
- TAM width
- Power consumption
- Area size
- Test data size

In an SoC design, a great portion of it is filled with reused IPs. Then, actual test cost information obtained from a past design can be used in a reused portion. Moreover, if possible, an estimation of the test cost information for a new design can be made based on the test cost information of the past design. For example, when there is an actual result value of the scan design of a certain core, it is possible to make an estimation of the test cost information in which the number of the scan chain was changed. If high accuracy is needed, it is necessary to carry out logic synthesis according to product specification, and to actually perform some scan design. However, if accuracy is not needed, some variations will be created only reflecting the change of the scan chain length when changing the number of scan chains by assuming that the area size, the number of test patterns and the number of flip-flops do not change. In this case, test application time, TAM width, and test data size are easily calculated. It is difficult to estimate the value of power consumption with high accuracy. However, there are conventional tools, which are able to estimate power consumption for an RTL description, and it is easy to perform relative comparison among two or more DFT(s). On the other hand, newly designed cores that have no past design information need to create the test cost information by actually applying DFT using RTL (a). However, accumulating the actual result value in the test cost information database (e) reduces the cost required to estimate test cost information, and it leads to an increase in accuracy.

In the second stage, the DFT of each core is selected to reduce the total test application time using test cost information (e), which was estimated in the first stage, and the selected DFT of each core (f) and test schedule (g) are

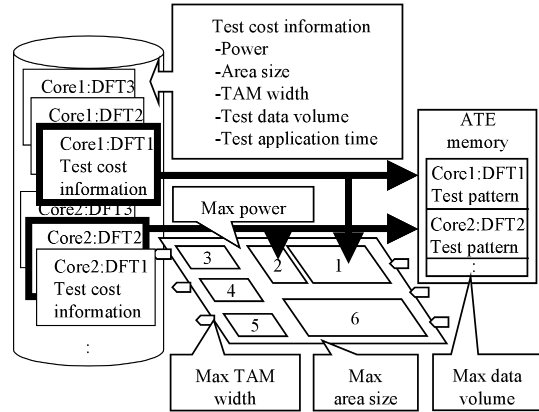

Fig. 2 DFT selection under constraints.

output. Test schedule means test start time and test end time of each core. According to the DFT selection information of each core, design for test of each core is performed in the third stage. The test pattern of the core (h) is created in the fourth stage using existing ATPG tools.

In the fifth stage, the test wrapper of each core is designed. Based on the test pattern of a core, which was created in the third stage, the bit width compression function is incorporated if needed. Moreover, the test pattern is modified to match the interface of the designed wrappers.

In the last stage, the cores tested simultaneously are divided into several TAMs according to the test schedule (g), which was created in the second stage, and SoC design that include TAMs is created. Moreover, the test pattern of each core is edited and output with the interface of SoC pin.

The DFT selection stage is especially important among the above mentioned framework stages. Figure 2 shows the work for which a designer is asked in this stage. The designer determines the test strategy of the SoC, including DFT selection of each core to reduce test application time under the following constraints: maximum TAM width, maximum allowed power consumption, total area size, test data size.

3. DFT of Each Core

3.1 Scan Design Method

Full scan design is one of the most popular DFT methods. The scan test application time depends on the maximum scan chain length. As a large TAM width can be taken, a scan chain can be divided and thus the maximum scan chain length and test application time can be shortened.

Figure 3 shows an example of the relationship between scan chain length and number of chains in the case in which the number of FFs is 100. The vertical axis shows the scan chain length in logarithm scale, and the horizontal axis shows the number of scan chains. Figure 3 shows that when the number of the scan chains is large, the scan chain length function becomes a stair function. Therefore even if the TAM width increases, the test application time is not always

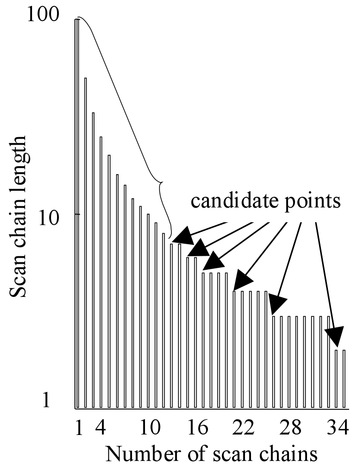

Fig. 3 Relationship between scan chain length and number of chains.

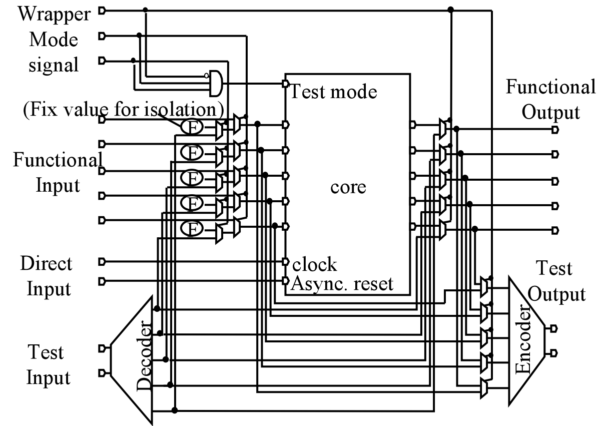

Fig. 4 Example of wrapper design for NS-DFT.

shortened [14]. In modular testing, the TAM width of each core should be selected to minimize the total test application time. However, in the stair function portion shown in Fig. 3, it is enough that only the pareto-optimal-points [14] are taken into consideration as the candidate points of the selection.

3.2 Non-scan DFT Method

Scan design methods have the following disadvantages concerning test cost and test quality:

– The additional test circuits for DFT cause the degradation of performance.
– The test length is very long.
– It is not suited for at-speed-testing.

In order to drastically improve the above-mentioned disadvantages while keeping complete fault efficiency, non-scan DFT methods [16], [17] for RTL design circuits were proposed. In this paper, the non-scan DFT method (NS-DFT) of reference [16] shall be chosen as another DFT of a core.

NS-DFT needs parallel access from LSI pins to all the inputs and all the outputs. Thus, if the sum of the number of inputs and the number of outputs is larger than the number of LSI pins, NS-DFT is inapplicable to the core.

Therefore, the core test wrapper with bit width compression function shall be prepared. Figure 4 shows the wrapper design. Wrapper mode signals are used to change among five modes: normal, test, isolation, input interconnect test, and output interconnect test. In normal mode, Functional inputs and Functional outputs are connected to the core. When another core is tested, the core inputs are fixed to isolate, if needed. In input interconnect test mode, the Functional Inputs are connected to the Encoder and observed at Test Outputs. In output interconnect test mode, the Functional Outputs are controlled from Test Inputs. In test mode, encoded test patterns are supplied to Test Inputs, and decoded patterns are provided to the inputs of the core. The outputs are encoded at Encoder, and observed at Test Outputs. The clock and Asynchronous signal are directly connected in all modes.

The input compression technique shall use the coding technique using EOR network [18], [19]. The output compression technique shall use EOR tree. These bit width compression techniques do not change test length, but the area size of the wrapper is different with the compression ratio. If the compression rate is high, the area size of the decoder and the encoder increase. Moreover, if the area size increases, power consumption increases under the same frequency. Since the reduction of TAM width increases the possibility that cores can carry out a simultaneous test, the total test application time may be shortened. Thus, there is a trade-off between the total test application time and area, and the total test application time and power consumption.

4. DFT Selection Problem Formulation and Algorithm

4.1 DFT Selection Problem Formulation

To select the DFT of each core in order to reduce the test application time under constraints, we formulate the DFT selection problem as follows.

Inputs:

(1) Test cost information of each core:
$\boldsymbol{D} = D_{ij}(\mathrm{DFT}_{ij}, w_{ij}, p_{ij}, v_{ij}, a_{ij}, t_{ij})$
Here, DFT_{ij}, w_{ij}, p_{ij}, v_{ij}, a_{ij}, and t_{ij} are, respectively:
DFT_{ij}: jth DFT of core i.
w_{ij}: TAM width of core i to which DFT_{ij} is applied.
p_{ij}: maximum power consumption of testing core i to which DFT_{ij} is applied.
a_{ij}: Area of core i to which DFT_{ij} is applied.
v_{ij}: The amount of test data of core i to which DFT_{ij} is applied.
t_{ij}: Test application time of core i to which DFT_{ij} is applied.
(2) Maximum TAM width of the SoC: W
(3) Maximum available peak power of the SoC: P

(4) Maximum amount of the total test data size: V
(5) Maximum area size of the SoC: A

Outputs:

(1) Selected DFT of each core
(2) Test schedule

Objective:
The test application time of the SoC is minimum, and constraints (Input: (2), (3), (4), (5)) are satisfied.

To solve this problem, an algorithm is proposed in the next paragraph.

4.2 Algorithm

The overview of this algorithm is as follows.

When the test of a certain core is the bottleneck, the total test application time may be improved by changing the DFT method or the TAM width of the core. Therefore, by repeating the following three steps, DFT assignment is changed so that test application time is shortened.

(Step 1) The test application time under an initial DFT assignment is calculated by executing test scheduling.

(Step 2) In order to find the test bottleneck, the following process is performed iteratively.
For each core in the SoC, replace the initial DFT with another DFT in the database. The test application time under the replaced DFT assignment is calculated by executing test scheduling.

(Step 3) The initial DFT assignment is updated by the iteratively determined DFT assignment whose test application time improved most.

If any change of DFT assignment in Step 2 caused an increase the test application time, it means that there is no bottleneck and, therefore, the algorithm ends.

The algorithm is shown in Fig. 5. The following variables are used in this algorithm.

(1) ***C***: Variable to store test cost information of each core: $\boldsymbol{C} = C_i(\mathrm{CDFT_i}, \mathrm{w_i}, \mathrm{p_i}, \mathrm{v_i}, \mathrm{a_i}, \mathrm{t_i})$
$\mathrm{CDFT_i}$, $\mathrm{w_i}$, $\mathrm{p_i}$, $\mathrm{v_i}$, $\mathrm{a_i}$, $\mathrm{t_i}$ are, respectively:
$\mathrm{CDFT_i}$: selected DFT of core i.
$\mathrm{w_i}$: TAM width of core i.
$\mathrm{p_i}$: Power consumption of core i.
$\mathrm{a_i}$: Area of core i.
$\mathrm{v_i}$: Test data volume of core i.
$\mathrm{t_i}$: Test application time of core i.
(2) ***Cinit***: Test cost information under the initial DFT selection.
(3) ***Ccurrent***: Variable to store the test cost information under the present DFT selection.
(4) ***Cbest***: Variable to store the test cost information under DFT selection of the minimum test application time.
(5) current_tat: Variable to store the test application time.
(6) best_tat: Variable to store the minimum test application time.
(7) trial_tat: Variable to store the test application time.

```
Procedure DFT_selection(D,W,P,A,V)
1  Define initial DFT assignment Cinit;
2  C=Cinit;
3  current_tat= sum of the test application time of each core;
4  best_tat=rectangle_packaging(C,W,P,A,V);
5  do{
6      current_tat= best_tat;
7      Ccurrent = C;
8      for( i = 1; i <= number of cores; i++ ){
9          for( j = 1; j <= number of DFTs of core i; j++){
10             if( CDFT_i != DFT_ij ){
11                 C_i = D_ij   /* change DFT of core i */
12                   trial_tat= rectangle_packaging(C,W,P,A,V);
13                 if( trial_tat < best_tat ){
14                     best_tat = trial_tat;
15                     Cbest = C;
16                 }
17             }
18         }
19         C =Ccurrent;
20     }
21     if( best_tat <current_tat ){
22         restore C = Cbest;
23     }
24 }while(best_tat<current_tat);
```

Fig. 5 DFT selection algorithm.

In the first, the initial DFT selection and the test cost information ***Cinit*** are created (line1). Initialize the variable that stores the test cost information current_tat with the sum of the test application time of each core of the initial DFT selection (line2). Then, test scheduling aiming at minimizing test application time is performed with the rectangle_packaging algorithm [14], [15]. The return value of the above-mentioned algorithm is the test application time of the whole SoC. This result is stored in the variable best_tat (line4).

Hereafter, while best_tat is updated, change the DFT selection and test scheduling repeatedly (line5–line24). The loop iteration is as follows.

The variable current_tat is updated with the value of best_tat (line6). Test cost information ***C*** under the current DFT selection is held as ***Ccurrent*** (line7). The following procedure is performed on all DFTs of each core (line8–line20).

If the current DFT of core i is not $\mathrm{DFT_{ij}}$ then copy D_{ij} to C_{i} (line10–11). Then, test scheduling is performed (line12). Consequently, if the obtained test application time is shorter than best_tat (line13), best_tat will be updated (line14) and the test cost information ***C*** will be stored as ***Cbest*** (line15). After all DFTs of the concerned core are tried, ***C*** is written back to ***Ccurrent*** (line18). Then, after trial all core trial, if best_tat was updated after trying all cores and all DFTs (line21), ***Cbest*** is copied to ***C*** (line22), and go back to the line 6 (line24), otherwise the algorithm ends. DFT of each core of ***Cbest*** obtained at the end is the solution of DFT selection algorithm.

This algorithm performs exhaustive search in the worst case. The complexity of the scheduling algorithm is $O(n \log n)$. The search space size of the DFT selection is

given by ($\prod_{i=1}^{n} M_i$), where M_i is the number of selectable DFTs of the core i. Therefore the complexity of this algorithm is $O((n \log n) \times \prod_{i=1}^{n} M_i)$. In other words, if the number of selectable DFTs of each core is at most m, the total complexity is $O(m^n n \log n)$. Because the search space becomes large rapidly to the increase in the number of cores and the number of DFTs, a search space reduction heuristic is needed for practical use.

Generally, in order to reduce the search space, and in order to estimate the quality of a solution, it is useful to know the lower bound of the cost function. In the DFT selection problem, it is difficult to find the optimum solution. However, it is possible to calculate some lower bounds of the test application time.

One of the optimum cases is shown in Fig. 6. The numbered rectangles represent the test application time and the TAM width of each core under the selected DFT. The vertical length of each rectangle represents the TAM width, and the horizontal length represents test application time. The dotted line shows the total rectangle, in which the vertical length represents the total TAM width and the horizontal length represents the total test application time. If (1) the DFT of each core is selected so that the size of each core's rectangle is minimum, and (2) the core's rectangles fill the total rectangles, then the total test application time is the true minimum. Thus a lower bound of test application time lb1 can be calculated as follows.

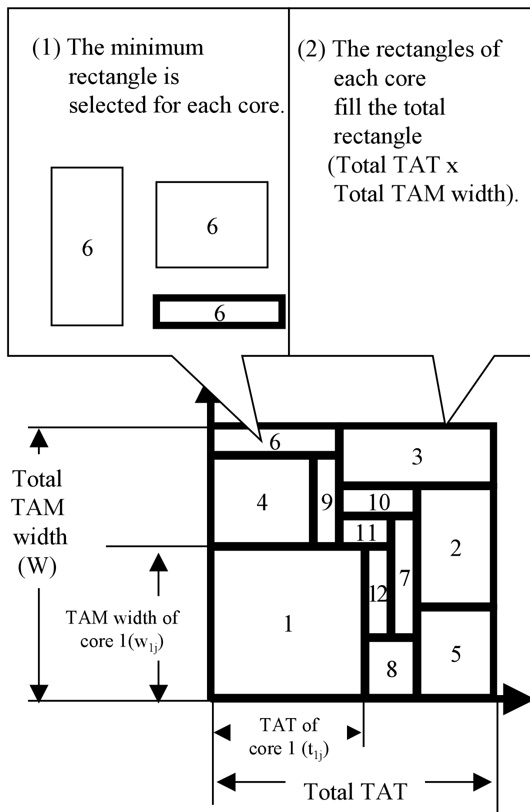

Fig. 6 Lower bound case of test application time.

$$\mathrm{lb1} = \sum_{i=1}^{n} \mathrm{min_j}(\mathrm{t_{ij}} \times \mathrm{w_{ij}})/\mathrm{W}$$

Furthermore, the total test application time cannot be shorter than the test application time of a core. Therefore another lower bound lb2 can be calculated as follows.

$$\mathrm{lb2} = \mathrm{max_i}(\mathrm{min_j}(\mathrm{t_{ij}}))$$

Let the largest of these two values lb1 and lb2 be the lower bound LB.

$$\mathrm{LB} = \max(\mathrm{lb1}, \mathrm{lb2})$$

If some choices can be deleted so that LB is not changed, it may be used effectively in a heuristics to prune the search space in the DFT selection algorithm. The other usage of LB is that the quality of a solution can be pessimistically estimated by comparing with LB.

5. Experimental Results

5.1 Experimental Environment

The experimental environment is as follows.

(1) The experiment had been held on the Sun ultra80 workstation, (Sun OS 5.6), 400 MHz, 2 Gbyte memory.
(2) The proposed algorithm was implemented in C.
(3) Nine RTL designs were used as experimental circuit.
(4) To prepare the test cost information, we used the following EDA tools:

– power consumption estimation: Wattme/Artgraphics
– logic synthesis: DesignCompiler/Synopsys
– Scan path synthesis: DFT Compiler/Synopsys
– ATPG: Tetra MAX/Synopsys
– NS-DFT: an in-house tool.

5.2 Circuits

Table 1 shows the characteristics of the circuits that used in this experiment without DFT. These nine circuits were used as cores. The 1st column shows the core number. The 2nd column shows the name of the core. The 3rd column shows the number of inputs. The 4th column shows the number of outputs. The 5th column denotes the number of memory elements. The 6th column shows the area in terms of gate counts after logic synthesis. The 7th column shows the system clock frequency. In addition, the frequency of the scan clock is assumed to be 1/5 of the system clock of normal operation. This assumption is based on the survey of some product data. Also, the frequency of the test clock of NS-DFT is the same as the system clock at normal operation. The 8th column shows the estimated power consumption. The values are relative to the normal operation of core No.1 without DFT. The 9th column denotes the minimum number and maximum number of the scan chains, which were added to prepare test cost information of scan DFT. The values inside the parenthesis show the

Table 1 Information of the cores.

No.	PI (bit)	PO (bit)	# of FF	Area	frequency (MHz)	power *1	Scan DFT # of chain min - max		NS-DFT TAM width min - max	
1	32	16	51	1094	100	1.00	1 - 17	(11)	16	(1)
2	20	16	192	2525	100	0.93	1 - 20	(16)	16	(1)
3	80	80	228	4494	100	2.16	1 - 80	(29)	32 - 64	(2)
4	32	32	83	1647	100	1.04	1 - 32	(16)	32	(1)
5	32	32	115	4713	100	1.48	1 - 33	(18)	32	(1)
6	32	98	1284	40528	100	141.31	1 - 92	(58)	32 - 98	(3)
7	58	128	1942	36560	100	116.15	1 - 122	(75)	32 - 128	(3)
8	129	260	357	22650	100	19.40	1 - 179	(37)	34 - 260	(5)
9	96	224	838	58287	100	54.77	1 - 210	(54)	32 - 224	(4)

*1 Values relative to the normal operation of core No.1

Table 2 Test application time (Select only from Scan).

Name	TAM width	No.	TAM width	Power *1	Area (gates)	TAT (10^{-6}s)	Schedule Start	Schedule End	Max Power	Total Area	Total TAT
SoC1	512	1	9	0.04	1301	50	0	50	15.40	192885	304
		2	11	0.04	3296	34	0	34			
		3	15	0.10	5409	56	0	56			
		4	11	0.05	1982	27	0	27			
		5	12	0.06	5176	43	0	43			
		6	96	6.36	45667	304	0	304			
		7	76	5.62	44331	195	0	195			
		8	55	0.82	24081	217	0	217			
		9	74	2.31	61642	244	0	244			
SoC2	256	1	9	0.04	1301	50	0	50	15.40	192885	466
		2	11	0.04	3296	34	0	34			
		3	15	0.10	5409	56	0	56			
		4	11	0.05	1982	27	0	27			
		5	12	0.06	5176	43	0	43			
		6	63	6.36	45667	466	0	466			
		7	34	5.62	44331	461	0	461			
		8	27	0.82	24081	462	0	462			
		9	74	2.31	61642	244	0	244			
SoC3	128	1	9	0.04	1301	50	0	50	15.32	192885	916
		2	11	0.04	3296	34	56	90			
		3	15	0.10	5409	56	0	56			
		4	11	0.05	1982	27	56	83			
		5	12	0.06	5176	43	0	43			
		6	34	6.36	45667	893	0	893			
		7	19	5.62	44331	916	0	916			
		8	17	0.82	24081	788	0	788			
		9	22	2.31	61642	902	0	902			
SoC4	64	1	10	0.04	1301	41	1330	1371	8.76	192885	1373
		2	11	0.04	3296	34	1330	1364			
		3	19	0.10	5409	43	1330	1373			
		4	11	0.05	1982	27	1330	1357			
		5	12	0.06	5176	43	1330	1373			
		6	63	6.36	45667	466	864	1330			
		7	25	5.62	44331	657	0	657			
		8	16	0.82	24081	843	0	843			
		9	23	2.31	61642	864	0	864			

*1 Values relative to the normal operation of core No.1 without DFT

number of choices in DFT selection. For example, we prepared 11 cases of the test cost information about scan DFT for core No.1. Scan DFT needs 4 more inputs (clock, reset, test mode, scan enable) beside the scan chains. Therefore, the sum of the extra 4 inputs and the number of chains corresponds to the TAM width. The 10th column denotes

Table 3 Test application time (select only from NS-DFT).

Name	TAM width	No.	TAM width	Power *1	Area (gates)	TAT (10^{-6}s)	Schedule		Max Power	Total Area	Total TAT
							Start	End			
SoC1	512	1	16	1.53	1680	4	0	4	461.19	254980	835
		2	16	1.19	3229	6	0	6			
		3	64	2.92	6087	3	0	3			
		4	32	1.53	2428	3	0	3			
		5	32	1.76	5635	13	0	13			
		6	32	163.75	46966	51	0	51			
		7	32	161.06	50702	835	0	835			
		8	64	25.09	29300	15	0	15			
		9	64	102.36	108953	228	0	228			
SoC2	256	1	16	1.53	1680	4	0	4	456.74	254980	835
		2	16	1.19	3229	6	0	6			
		3	64	2.92	6087	3	228	231			
		4	32	1.53	2428	3	228	231			
		5	32	1.76	5635	13	0	13			
		6	32	163.75	46966	51	0	51			
		7	32	161.06	50702	835	0	835			
		8	64	25.09	29300	15	0	15			
		9	64	102.36	108953	228	0	228			
SoC3	128	1	16	1.53	1680	4	228	232	429.04	255172	850
		2	16	1.19	3229	6	835	841			
		3	64	2.92	6087	3	232	235			
		4	32	1.53	2428	3	228	231			
		5	32	1.76	5635	13	0	13			
		6	32	163.75	46966	51	0	51			
		7	32	161.06	50702	835	0	835			
		8	34	25.17	29396	15	835	850			
		9	32	102.46	109049	228	0	228			
SoC4	64	1	16	1.53	1680	4	228	232	263.53	255172	904
		2	16	1.19	3229	6	886	892			
		3	64	2.92	6087	3	901	904			
		4	32	1.53	2428	3	232	235			
		5	32	1.76	5635	13	835	848			
		6	32	163.75	46966	51	835	886			
		7	32	161.06	50702	835	0	835			
		8	34	25.17	29396	15	886	901			
		9	32	102.46	109049	228	0	228			

*1 Values relative to the normal operation of core No.1 without DFT

the minimum TAM width and the maximum TAM width of NS-DFT. The values inside the parenthesis show the number of choices in DFT selection.

In this experiment, two types of DFTs are applied to each circuit: one is Scan DFT, and the other is NS-DFT. NS-DFT is a technique still under research and, for this reason, there are a few constraints when using it, such as controller and data-path must be designed separately. The selected experimental circuits satisfy these constraints.

Throughout this experiment, the constraints of the test scheduling were set as follows.

- Maximum allowed power consumption:
  (Sum of the power consumption of each core)
  $\times$ 1.5 = 507.34
- Total area size:
  (Sum of the area size of each core) $\times$ 1.5 = 258747
- Total TAM width: 512, 256, 128, 64

We made experiments for the following three cases: (1) Scan only (number of scan chain selection), (2) NS-DFT only (TAM width selection), (3) Scan and NS-DFT (DFT method and number of scan chain, TAM width selection).

In this experiment, the constraint values had to be increased in order to obtain a solution of each case, so that we were able to compare the result.

For a thorough evaluation of the proposed algorithm, more experiments under suitable constraints are needed as future work.

5.3 Experimental Results

(1) case 1: select only from Scan

In case 1, the selectable DFT of each core was limited to scan design. The number of scan chains in each core is selected to reduce the total test application time. Table 2 shows the results of case 1. The 1st column shows the name

Table 4 Test application time (select from either Scan or NS-DFT).

Name	TAM width	No.	selected DFT	TAM width	Power *1	Area (gates)	TAT (10^{-6}s)	Schedule		Max Power	Total Area	Total TAT
								Start	End			
SoC1	512	1	NS-DFT	16	1.53	1680	4	0	4	200.92	201154	121
		2	NS-DFT	16	1.19	3229	6	0	6			
		3	NS-DFT	80	2.89	6039	3	118	121			
		4	NS-DFT	32	1.53	2428	3	118	121			
		5	NS-DFT	32	1.76	5635	13	0	13			
		6	NS-DFT	64	163.41	46870	51	0	51			
		7	ScanDFT	126	5.62	44331	118	0	118			
		8	NS-DFT	64	25.09	29300	15	0	15			
		9	ScanDFT	172	2.31	61642	112	0	112			
SoC2	256	1	NS-DFT	16	1.53	1680	4	169	173	171.69	201250	184
		2	NS-DFT	16	1.19	3229	6	169	175			
		3	NS-DFT	80	2.89	6039	3	169	172			
		4	NS-DFT	32	1.53	2428	3	169	172			
		5	NS-DFT	32	1.76	5635	13	169	182			
		6	NS-DFT	32	163.75	46966	51	0	51			
		7	ScanDFT	112	5.62	44331	132	0	132			
		8	NS-DFT	64	25.09	29300	15	169	184			
		9	ScanDFT	109	2.31	61642	169	0	169			
SoC3	128	1	NS-DFT	16	1.53	1680	4	237	241	271.84	248657	255
		2	NS-DFT	16	1.19	3229	6	237	243			
		3	NS-DFT	80	2.89	6039	3	252	255			
		4	NS-DFT	32	1.53	2428	3	252	255			
		5	NS-DFT	32	1.76	5635	13	237	250			
		6	NS-DFT	32	163.75	46966	51	0	51			
		7	ScanDFT	63	5.62	44331	237	0	237			
		8	NS-DFT	64	25.09	29300	15	237	252			
		9	NS-DFT	32	102.46	109049	228	0	228			
SoC4	64	1	NS-DFT	16	1.53	1680	4	480	484	266.21	248705	499
		2	NS-DFT	16	1.19	3229	6	480	486			
		3	NS-DFT	64	2.92	6087	3	493	496			
		4	NS-DFT	32	1.53	2428	3	496	499			
		5	NS-DFT	32	1.76	5635	13	480	493			
		6	NS-DFT	32	163.75	46966	51	237	288			
		7	ScanDFT	63	5.62	44331	237	0	237			
		8	NS-DFT	64	25.09	29300	15	465	480			
		9	NS-DFT	32	102.46	109049	228	237	465			

*1 Values relative to the normal operation of core No.1 without DFT

of the SoC. The 2nd column shows the maximum TAM width of each SoC. The 3rd column denotes the core number denoted in Table 1. The 4th column shows the selected TAM width of each core. The 5th column shows the power consumption. The power consumption results are shown relative to the normal operation figures of the original core No.1. The 6th column shows the area size of each core. The 7th column shows the test application time of each core. The 8th column and the 9th column show test schedule of each core. "Start" denotes the test start time, and "end" denotes test end time. The 10th column shows maximum power consumption of each SoC under this test schedule. The 11th column shows the total area size of each SoC. The 12th column shows the total test application time of each SoC, either.

In SoC1, a solution for testing all cores simultaneously was obtained. The DFT of core No.6 has the largest test application time among all cores of SoC1, even though the DFT with the smallest test application time was chosen for core No.6. Under this condition, as long as there is no other choice of DFT which improves the test application time of core No.6, the total test application time does not improve.

In SoC4, the found solution splits the test of core No.6 from the test of all other cores, i.e., core No.6 is tested independently.

(2) case 2: select only from NS-DFT

In case 2, the selectable DFT of each core was limited to NS-DFT. Table 3 shows the results of case 2. The meaning of each column is the same as that of Table 2.

We obtained a test schedule such that SoC1 and SoC2 have the same test application time. There were two factors that contributed to obtaining this result. First, the test length for any cores with NS-DFT does not depend on its TAM width, and initial DFT selection was not changed. Although the influence of the difference of TAM width appeared in the power consumption and the area after DFT, it did not appear on these results. Second, the test application time

of core No.7 was very large, and as long as there was no choice of DFT which improves its test application time, the test application time of the whole SoC was not shortened either. In addition, in the solution for SoC3 and SoC4, DFTs with the minimum TAM width were already chosen by the initial DFT selection except for core No.3, and there was no improvement from the initial DFT selection.

(3) case 3: Scan or NS-DFT

In case 3, the DFT of each core was selected from NS-DFT or scan design. Table 4 shows the result of case 3. The meaning of the 1st–3rd columns is the same as that of Table 2 and Table 3. The 4th column shows the selected DFT. The meaning of the 5–13th columns is the same as that of the 4–12th columns of Table 2 and Table 3.

As for the test application time of the whole SoC, we obtained the shortest times compared to cases 1 and 2. In case 1, the test application time was longer because of the DFT selected for core No.6. However, in case 3 we could choose another DFT for core No.6 such that test application time improved. In case 2, the test application time was longer because of the DFT selected for core No.7. However, in case 3 we could choose another DFT for core No.7 such that test application time improved.

Table 5 shows the comparison of test application time. The 1st column shows SoC name. The 2nd column shows total TAM width. The 3rd–5th columns show the total test application time of cases 1–3, respectively. The 6th column shows the ratio of test application time in case 3 to the test application time in case 1. The 7th column shows the ratio of test application time in case 3 to the test application time in case 2. In case 3, the total test application times were shortened from 60% to 72% compared with case 1, and from 45% to 86% compared with case 2.

These experimental results show that the differences in the selection scope of DFT drastically changes the test cost of an SoC. Therefore, the usefulness of having a database with the test cost information of several DFTs, and using this database to optimize DFT selection to reduce test cost in early stages of design flow was shown.

(4) The quality of the solution

Table 6 shows the test application time, which is calculated in cases 1 to 3, and the lower bounds of optimum solution, which is denoted in Sect. 4. The 1st column shows the case number. The 2nd column shows the name of the SoC. The 3rd column shows the total TAM width. The 4th column shows total test application time, which is calculated in this experiment. The 5th column shows the lower bound of test application time, which is described in Sect. 4. The 6th column shows the ratio of experimental results relative to the lower bounds. The last column shows the CPU time of the experiments. In cases 1, 2, and 3, the numbers of combinations of DFT selection are 1.28 × 1013, 360, 2.37 × 1013, respectively. In the worst case, the proposed algorithm performs exhaustive search, and the execution time is possibly so long that the experiment cannot be completed. However, in this experiment, the execution time is less than 100 seconds. Although the algorithm may have ended without carrying out sufficient search, the ratio of experimental results to the lower bounds shows that the experimental results are less than 1.6 times larger than the optimum solutions.

Table 5 Comparison of test application time.

Name	TAM width	case1 [A]	case2 [B]	case3 [C]	C/A	C/B
SoC1	512	304	835	121	0.40	0.14
SoC2	256	466	835	184	0.39	0.22
SoC3	128	916	850	255	0.28	0.30
SoC4	64	1373	904	499	0.36	0.55

Table 6 Test application times and lower bounds of optimum solution.

	Name	TAM width	Total TAT	LB	TAT/LB	CPU (sec)
case1	SoC1	512	304	304	1.00	75.8
	SoC2	256	466	304	1.53	55.4
	SoC3	128	916	596	1.54	19.5
	SoC4	64	1373	1191	1.15	32.6
case2	SoC1	512	835	835	1.00	0.6
	SoC2	256	835	835	1.00	0.6
	SoC3	128	850	835	1.02	0.6
	SoC4	64	904	835	1.08	0.5
case3	SoC1	512	121	118	1.03	33.7
	SoC2	256	184	118	1.56	55.1
	SoC3	128	255	198	1.29	26.4
	SoC4	64	499	396	1.26	28.5

6. Conclusions

The framework of an SoC test architecture generation framework was proposed. The framework contains a database, which stores the test cost information on several DFTs for every core, and DFT selection part. In the framework, the DFT of each core is selected for reducing the test application time using test cost information database in the early phase of the design flow. Moreover, the DFT selection problem was formulated and the algorithm, which solves this was proposed. Experimental results show that bottlenecks in test application time when using the single DFT method for all cores in an SoC is reduced by performing DFT selection from two types of DFTs. As a result, the whole test application time is shortened.

Our future work includes the following issues.

(1) Experiments under various constraints
(2) Optimization including the TAM design
(3) Addition of Scan BIST and Non-Scan BIST to the selectable DFTs
(4) Improvement of the algorithm to reduce the search space

Acknowledgments

This work was sponsored by NEDO (New Energy and Industrial Technology Development Organization) as VCDS Project (SoC advanced design technology development

project). The authors would like to thank Professor Michiko Inoue and Professor Tomokazu Yoneda of Nara Institute of Science and Technology, Professor Tomoo Inoue and Hideyuki Ichihara of Hiroshima City University, and Dr. Rafael K. Morizawa of STARC for their valuable discussion and comments.

References

[1] T. Ono, K. Wakui, H. Hikima, Y. Nakamura, and M. Yoshida, "Integrated and automated design-for-testability implementation for cell-based ICs," Proc. 6th Asian Test Symposium, pp.122–125, Nov. 1997.

[2] E. Marinissen, R. Arendsen, G. Bos, H. Dingemanse, M. Lousberg, and C. Wouters, "A structured and scalable mechanism for test access to embedded reusable cores," Proc. International Test Conf., pp.284–293, Oct. 1998.

[3] P. Vama and S. Bhatia, "A structured test re-use methodology for core-based system chips," Proc. International Test Conf., pp.294–302, Oct. 1998.

[4] N.A. Touba and B. Pouya, "Testing embedded cores using partial isolation rings," Proc. VLSI Test Symposium, pp.10–16, May 1997.

[5] L. Whetsel, "An IEEE 1149.1 based test access architecture for ICs with embedded cores," Proc. International Test Conf., pp.69–78, Nov. 1997.

[6] M. Nourani and C.A. Papachristou, "Structural fault testing of embedded cores using pipelining," J. Electron. Test., Theory Appl., vol.15, pp.129–144, 1999.

[7] I. Ghosh, S. Dey, and N.K. Jha, "A fast and low cost testing technique for core-based system-on-chip," Proc. 35th Design Automation Conf., pp.542–547, 1998.

[8] I. Ghosh, S. Dey, and N.K. Jha, "A low overhead design for testability and test generation technique for core-based system-on-a-chip," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.18, no.11, pp.1661–1676, Nov. 1999.

[9] T. Yoneda and H. Fujiwara, "A DFT method for core-based systems-on-a-chip based on consecutive testability," Proc. 10th Asian Test Symposium, pp.193–198, Nov. 2001.

[10] V. Iyengar, K. Chakrabarty, and E.J. Marinissen, "Test wrapper and test access mechanism co-optimization for system-on-chip," J. Electron. Test., Theory Appl., vol.18, pp.213–230, April 2002.

[11] Y. Huang, W.T. Cheng, C.C. Tsai, and N. Mukherjee, "Resource allocation and test scheduling for concurrent test of core-based SOC design," Proc. 10th Asian Test Symposium, pp.265–270, Nov. 2001.

[12] E. Larsson, K. Arvidsson, H. Fujiwara, and Z. Peng, "Integrated test scheduling, test parallelization and TAM design," Proc. 11th Asian Test Symposium, pp.397–404, Nov. 2002.

[13] H.S. Hsu, J.R. Hung, K.L. Cheng, C.W. Wang, C.T. Huang, and C.W. Wu, "Test scheduling and test access architecture optimization for system-on-chip," Proc. 11th Asian Test Symposium, pp.411–416, Nov. 2002.

[14] V. Iyengar, K. Chakrabarty, and E.J. Marinissen, "On using rectangle packaging for SOC wrapper/TAM co-optimization," Proc. VLSI Test Symposium, pp.253–258, May 2002.

[15] Y. Huang, N. Mukherjee, S. Reddy, C. Tsai, W. Cheng, O. Samman, P. Reuter, and Y. Zaidan, "Optimal core wrapper width selection and SOC test scheduling based on 3-dimensional bin packing algorithm," Proc. International Test Conf., pp.74–82, Oct. 2002.

[16] H. Date, T. Hosokawa, and M. Muraoka, "A SoC test strategy based on a non-scan DFT method," Proc. 11th Asian Test Symposium, pp.305–310, Nov. 2002.

[17] S. Nagai, S. Ohotake, and H. Fujiwara, "A DFT method for RTL data paths based on strong testability to reduce test application time," IEICE Technical Report, DC2002-84, Feb. 2003.

[18] B. Koneman, "LFSR-coded test patterns for scan designs," Proc. European Test Conf., pp.237–242, March 1993.

[19] I. Bayraktarouglu and A. Orailoglu, "Test volume and application time reduction through scan chain concealment," Proc. 38th Design Automation Conf., pp.151–155, June 2001.

Masahide Miyazaki received the B.E. and M.S. degrees in Space Science from Nagoya University, Nagoya, Japan, in 1990 and 1992, respectively. In 1992, he joined Hitachi Ltd. He has worked on sequential redundancy, and design for testability. In 2002, he went to Semiconductor Technology Academic Research Center (STARC) as a Hitachi assignee, where is currently working on testing for system on a chip. He entered the graduate program of the Graduate School of Information Science, Nara Institute of Science and Technology in the Autumn of 2003.

Toshinori Hosokawa received the B.E. degree in Electronics and Communication Engineering from Meiji University, Kawasaki, Japan, in 1987. He also received the D.E. degree from Meiji University in 2001. In 1987, he joined Matsushita Electric Industrial Co., Ltd. He has worked on logic simulation engine, automatic test pattern generation, fault simulation, design for testability and high level testing. From 2000 to 2003, he was temporarily transferred to Semiconductor Technology Academic Research Center (STARC) and worked on testing for system on a chip and hardware/software co-verification. He was also a lecturer at Meiji University in 2001 and 2002. Since 2003, he has been an associate professor of Nihon University. He is a member of IEEE (Institute of Electrical & Electronics Engineers) and IPSJ (Information Processing Society of Japan).

Hiroshi Date received the B.S. and M.S. degrees in Science of Mathematics from Kyushu University, Fukuoka, Japan, in 1985 and 1987, respectively. He also received the D.E. degree from Kyushu University in 1993. From 1987 to 1996, he was with Hitachi Research Laboratory, Hitachi Ltd. and was engaged in VLSI-CAD and parallel processing. From 1990 to 1993, he joined Institute for New Generation Computer Technology (ICOT) and worked on VLSI-CAD using parallel processing. From 1996 to 2001, he was with Institute of Systems & Information Technologies/KYUSHU. From 1999 to 2001, he was an associate professor of Kyushu University. From 2001 to 2002, he was with Abel Systems Inc. Since 2000, he has joined Semiconductor Technology Academic Research Center. Since 2002, he has been a chairman & CEO of System J D Co., Ltd. His research interests include design & test methodology and CAD technologies for system LSI, and information security. He is a member of IPSJ (Information Processing Society of Japan), TTTC (Test Technology Technical Council), ACM (Association for Computing Machinery), and IEEE (Institute of Electrical & Electronics Engineers).

Michiaki Muraoka received the B.E. degree in Electrical Engineering from Keio University, Tokyo, Japan in 1974. He worked on the research and development of system and semiconductor EDA system, especially in the logic and RTL design and silicon compilation at OKI Electric from 1974 to 1988. He joined Matsushita Electric Industrial Company where he worked on the research and development of EDA technology and IP based design methodology as the technical leader from 1988 to 2000. He was temporarily assigned to STARC in 2000 as a senior manager of VCDS Development Group. His current research interests are in the areas of the next generation design methodology and EDA technology for the giga scale SoC (System on Chip).

Hideo Fujiwara received the B.E., M.E., and Ph.D. degrees in electronic engineering from Osaka University, Osaka, Japan, in 1969, 1971, and 1974, respectively. He was with Osaka University from 1974 to 1985 and Meiji University from 1985 to 1993, and joined Nara Institute of Science and Technology in 1993. In 1981 he was a Visiting Research Assistant Professor at the University of Waterloo, and in 1984 he was a Visiting Associate Professor at McGill University, Canada. Presently he is a Professor at the Graduate School of Information Science, Nara Institute of Science and Technology, Nara, Japan. His research interests are logic design, digital systems design and test, VLSI CAD and fault tolerant computing, including high-level/logic synthesis for testability, test synthesis, design for testability, built-in self-test, test pattern generation, parallel processing, and computational complexity. He is the author of Logic Testing and Design for Testability (MIT Press, 1985). He received the IECE Young Engineer Award in 1977, IEEE Computer Society Certificate of Appreciation Award in 1991, 2000 and 2001, Okawa Prize for Publication in 1994, IEEE Computer Society Meritorious Service Award in 1996, and IEEE Computer Society Outstanding Contribution Award in 2001. He is an advisory member of IEICE Trans. on Information and Systems and an editor of IEEE Trans. on Computers, J. Electronic Testing, J. Circuits, Systems and Computers, J. VLSI Design and others. Dr. Fujiwara is a fellow of the IEEE, a Golden Core member of the IEEE Computer Society, and a member of the Information Processing Society of Japan.