Design of System-on-a-Chip Test Access Architectures under Place-and-Route and Power Constraints[1]

Krishnendu Chakrabarty
Department of Electrical & Computer Engineering
Duke University
Durham, NC 27708, USA

Abstract

Test access is a difficult problem encountered in the testing of core-based system-on-a-chip (SOC) designs. Since embedded cores in an SOC are not directly accessible via chip inputs and outputs, special access mechanisms are required to test them at the system level. We propose test access architectures based on integer linear programming (ILP) that incorporate place-and-route constraints arising from the functional interconnections between cores, as well as system-level constraints on power consumption. As a case study, we apply the ILP models to two representative SOCs, and solve them using a public-domain ILP software package.

1 Introduction

Embedded cores are now widespread in large system-on-a-chip (SOC) designs [14]. Howevever, since embedded cores are not directly accessible via chip inputs and outputs, special access mechanisms are required to test them at the system level. The design of the test access architecture is especially important for the system designer/integrator since the IEEE P1500 standard, which is being developed for embedded core testing, leaves TAM design upto the system integrator [10].

A test access architecture, also referred to as a test access mechanism (TAM), provides means for on-chip test data transport [14]. It can be used to transport test patterns from a pattern source to a core-under-test, and to transport test responses from a core-under-test to a response monitor. A number of test access architectures have been proposed in the literature [2, 4, 6, 9, 12, 14]. In order to reduce test cost, the testing time for a core-based system should be minimized by adopting an appropriate test access architecture. In addition, the test access mechanism should reflect the place-and-route constraints imposed by the functional interconnections between the embedded cores. Furthermore, the test access architecture, which to a large extent determines the amount of test parallelism, should also take into account system-level constraints on power consumption.

Existing test access mechanisms are ad hoc and do not directly address the problem of minimizing testing time under place-and-route and power constraints. Related prior work has either been limited to test scheduling for a given test access mechanism [5, 11], or to determining the optimal number of internal scan chains in the cores [1]. The latter requires redesign of the scan chains for each customer and thereby affects core reuse. We are interested instead in the problem of minimizing the SOC testing time without any redesign of the embedded cores.

Recently, we addressed the following TAM design problems that are of interest to the system integrator [4]: (1) Given an SOC and maximum total test bus width, how should the bus width be distributed among the various test buses in order to minimize the testing time? (2) How should the embedded cores in the system be assigned to the test buses? (3) For a given test access architecture, how much test bus width is required to meet specified testing time objectives?

In this paper, we formulate the above optimization problems in the context of place-and-route constraints arising from the functional interconnections betwen cores. We also incorporate power dissipation constraints in the optimization framework.

The main contributions of the paper are listed below.

- We first review an integer linear programming (ILP) model for optimally distributing the total test bus width among the individual test buses [4]. If the assignment of cores to test buses is not pre-determined by placement and routing constraints, this model combines width distribution with the problem of assigning cores to test buses. We also present an ILP formulation based on a *reward model* that takes into account the system integrator's preferences in allocating cores to test buses.
- We extend the above ILP models by including limits on test parallelism imposed by system-level power consumption constraints.
- We evaluate the feasibility of the proposed ILP models by solving them using an ILP solver for two hypothetical, but non-trivial and representative SOCs.

In order to illustrate the proposed optimization methods, we use the core-based SOCs $\mathcal{S}_1$ and $\mathcal{S}_2$ shown in Figure 1 as examples throughout the paper. These hypothetical but non-trivial SOCs consist of ten ISCAS 85 and ISCAS 89 benchmark circuits each. We assume that the ISCAS 89 circuits contain internal scan chains. $\mathcal{S}_1$ contains seven combinational cores and three sequential cores, while $\mathcal{S}_2$ consists of two combinational cores and eight sequential cores. The complexity of the ILP models depends more on the number of cores in the SOC than on the sizes of the cores. For the sake of illustration, only two test buses are shown in Figure 1. Our ILP models can be easily used for any number of test buses.

The organization of the paper is as follows. In Section 2, we review ILP models for determining an optimal test width distribution. In Section 3, we present the reward model for taking into account designer's preferences when the assignment of cores to test

[1] This research was supported in part by the National Science Foundation under grant number CCR-9875324, and by an equipment grant from Sun Microsystems.

DAC 2000, Los Angeles, California
(c) 2000 ACM 1-58113-188-7/00/0006..$5.00

buses is not pre-determined. In Section 4, we extend the ILP models of the previous sections by incorporating constraints on power consumption. In each case, we present case studies for the two example SOCs. We solve the various ILP models for these systems using the *lpsolve* software package from Eindhoven University of Technology in the Netherlands [3].

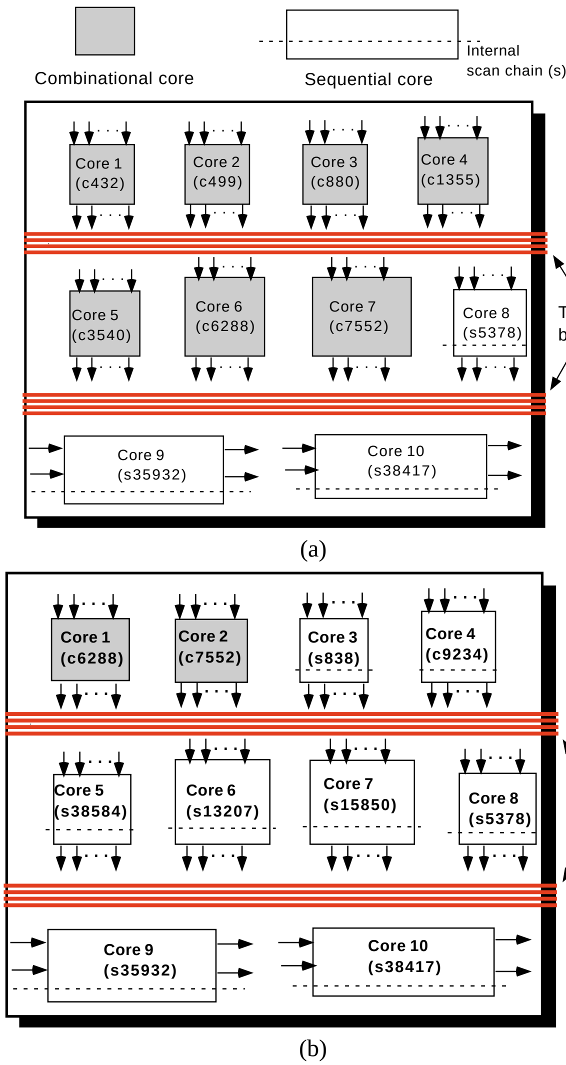

Figure 1. Two examples of SOCs: (a) $\mathcal{S}_1$ (b) $\mathcal{S}_2$.

2 Optimal test bus width distribution

We first briefly review ILP using matrix notation [13]. The goal of ILP is to minimize a linear objective function on a set of integer variables, while satisfying a set of linear constraints. A typical ILP model is described as follows:

$$\begin{array}{ll} \text{minimize:} & \mathbf{Ax} \\ \text{subject to:} & \mathbf{Bx} \leq \mathbf{C}, \mathbf{x} \geq 0, \end{array}$$

where $\mathbf{A}$ is a cost vector, $\mathbf{B}$ is a constraint matrix, $\mathbf{C}$ is a column vector of constants, and $\mathbf{x}$ is a vector of integer variables. Efficient ILP solvers are now readily available, both commercially and in the public domain.

Let the SOC design consist of N_C cores, and let core i, $1 \leq i \leq N_C$, have n_i inputs and m_i outputs. We assume that the n_i inputs of core i include data inputs and scan inputs. Similarly, the m_i outputs of core i include data outputs and scan outputs. Each full or partial scan core may have one have one or more internal scan chains. (A combinational or non-scan legacy core has no scan inputs and outputs.)

The amount of test data serialization necessary at the inputs and outputs of core i is determined by its *test width* $\phi_i = \max\{n_i, m_i\}$. This influences the testing time for core i. We assume that core i requires t_i (scan) cycles for testing. Finally, we assume that the SOC contains N_B test buses with widths $w_1, w_2, \ldots, w_{N_B}$, respectively.

We assume that a test bus does not fork (split) into multiple branches which may merge later. This restriction can be easily removed by extending the basic ILP model [4]. We also assume that all cores on any given test bus are tested sequentially. Two or more test buses can be used simultaneously for delivering test data to cores and for propagating test responses. We assume that the number of test buses (and thereby the amount of test parallelism) is determined by the core user (system integrator) after a careful consideration of system-level I/O, area, and power dissipation issues.

If core i is assigned to bus j, then the testing time for core i is given by

$$T_{ij} = \begin{cases} t_i, & \text{if } \phi_i \leq w_j \\ (\phi_i - w_j + 1)t_i, & \text{if } \phi_i > w_j \end{cases}$$

If $\phi_i > w_j$ then the width of the test bus is insufficient for parallel testing, and serialization of the test data is necessary at the wrapper on the inputs and/or outputs of core i. In order to calculate the test time due to serialization, we assume the interconnection strategy suggested in [9] for connecting core I/Os to the test bus, namely, provide direct (parallel) connection to core I/Os that transport more test data. If the width of bus j is adequate, i.e. $\phi_i \leq w_j$, then no serialization is necessary and core i can be tested in exactly t_i cycles.

Let x_{ij} be a 0-1 variable defined as follows:

$$x_{ij} = \begin{cases} 1, \text{ if core } i \text{ is assigned to bus } j \\ 0, \text{ otherwise} \end{cases}$$

The time needed to test all cores on bus j is therefore $\sum_{i=1}^{N_C} T_{ij}x_{ij}$. Since all the test buses can be used simultaneously for testing, the system testing time equals $\max_j \sum_{i=1}^{N_C} T_{ij}x_{ij}$.

As examples, we consider the SOCs $\mathcal{S}_1$ and $\mathcal{S}_2$ introduced in Section 1. We assume that s838 contains one internal scan chain, and s5378 and s9234 contain 4 internal scan chain each. We also assume that s35392 and s38417 contain 32 internal scan chains each, and s13207 and s15850 contain 16 scan chains each. For the combinational cores, $1 \leq i \leq 7$, the number of test cycles t_i is equal to the number of test patterns p_i. However, for the remaining cores with internal scan, $t_i = (p_i + 1)\lceil f_i/N_i \rceil + p_i$, where core i contains f_i flip-flops and N_i internal scan chains [1]. The test patterns for these circuits were obtained from [7].

We now investigate the problem of minimizing system testing time by determining (i) optimal widths for the test buses, and (ii) optimal assignment of cores to test buses. A special case of this problem is encountered when the assignment of cores to test buses is pre-determined by place-and-route constraints arising from functional interconnections between cores. We assume that the total system test bus width can be at most W. We also assume that the width of a test bus does not exceed the width required for

Minimize C subject to:

1. $C \geq \sum_{i=1}^{N_C}((\phi_i + 1)t_i x_{ij} - t_i y_{ij}), 1 \leq j \leq N_B$
2. $y_{ij} - w_{max} x_{ij} \leq 0$, $1 \leq i \leq N_C$, $1 \leq j \leq N_B$, where w_{max} is an upper bound on the w_j's.
3. $-w_j + y_{ij} \leq 0, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
4. $w_j - y_{ij} + w_{max} x_{ij} \leq w_{max}, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
5. $\sum_{j=1}^{N_B} x_{ij} = 1, 1 \leq i \leq N_C$
6. $\sum_{j=1}^{N_B} w_j = W$
7. $w_j \leq \phi_i, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
8. $x_{ij} = 0$ or 1

Figure 2. Integer linear programming model for $\mathcal{P}1$.

any given core, i.e. $\max_j\{w_j\} \leq \min_i\{\phi_i\}$ for all values of i and j. This assumption is necessary to avoid complex non-linear models that detract from the main ideas behind the ILP model. From a practical point of view, this assumption implies that cores with very small test widths are assigned to test buses after the cores with larger test widths are optimally assigned. An extension of this basic ILP model was presented in [4] and is therefore not described here.

We now formulate the problem of optimally allocating the total test bus width among the N_B buses, as well as determining the optimal allocation of cores to these buses.

- $\mathcal{P}1$: Given N_C cores and N_B test buses of total width W, determine the optimal width of the test buses, and an assignment of cores to test buses such that the total testing time is minimized.

An ILP model for $\mathcal{P}1$ can be developed as follows:

Minimize C subject to:

1) $C \geq \sum_{i=1}^{N_C}(\phi_i - w_j + 1)t_i x_{ij}, 1 \leq j \leq N_B$

3) $\sum_{j=1}^{N_B} x_{ij} = 1, 1 \leq i \leq N_C$

4) $\sum_{j=1}^{N_B} w_j = W, 1 \leq j \leq N_B$

5) $w_j \leq \phi_i, 1 \leq i \leq N_C, 1 \leq j \leq N_B$

6) $x_{ij} = 0$ or 1

Note that constraint 1) above is non-linear since it contains a product term. We linearize it by replacing the product term $w_j x_{ij}$ with a new integer variable y_{ij}, and adding the following three constraints for every such product term:

1. $y_{ij} - w_{max} x_{ij} \leq 0$, where $w_{max} = W$ is an upper bound on the widths of the test buses.
2. $-w_j + y_{ij} \leq 0$
3. $w_j - y_{ij} + w_{max} x_{ij} \leq w_{max}$

This leads us to the (linearized) ILP model for $\mathcal{P}1$ shown in Figure 2. If the assignment of cores to test buses is pre-determined, then the $x_{i,j}$s are no longer treated as variables in the simplified ILP model.

A related optimization problem is one of determining the minimum system test width required to meet a minimum testing time objective. This problem was formally addressed in [4] and is therefore not described here any further.

We now present experimental results using $\mathcal{S}_1$ and $\mathcal{S}_2$ for $\mathcal{P}1$. We solved the ILP models using *lpsolve* on a Sun Ultra 10 workstation with a 333 MHz processor and 128 MB memory. We were unable to obtain actual CPU times from *lpsolve*; however, the user time was less than one hour in all cases—in fact, in most cases, the CPU time was only a few minutes.

Table 1 presents the optimal test data width, optimal width distribution, and test bus assignment vector when two test buses are considered for $\mathcal{S}_1$ and $\mathcal{S}_2$. For $\mathcal{S}_1$, the lowest testing time of 391190 is reached for $W = 56$ bits. Any further increase in the system test width W does not decrease testing time since the widest test bus can be at most $\min_i\{\phi_i\} = 32$ bits.

Table 2 shows the optimal width distribution and optimum testing time for $\mathcal{S}_1$ and $\mathcal{S}_2$ if the test bus assignment is pre-determined as (1,1,1,1,2,2,2,2,2,1) in each case due to place-and-route constraints. The results show that while there is no significant increase in the testing time for $\mathcal{S}_1$, the testing time grows considerably for $\mathcal{S}_2$

		Optimum	Test bus assignment
W	(w_1, w_2)	testing time	vector
8	(4,4)	497200	(2,2,2,1,2,1,2,2,2,1)
12	(6,6)	487940	(2,1,2,1,1,1,1,1,1,2)
16	(8,8)	478936	(2,2,2,2,2,2,2,2,2,1)
20	(11,9)	470380	(2,1,1,2,2,2,2,2,2,1)
24	(11,13)	461277	(2,1,1,1,1,1,1,1,1,2)
28	(16,12)	452781	(1,2,2,1,2,1,2,2,2,1)
32	(18,14)	443620	(2,1,2,2,2,2,2,2,2,1)
36	(21,15)	435042	(1,1,2,1,2,1,2,2,2,1)
40	(17,23)	426043	(2,2,2,1,1,1,2,1,1,2)
44	(25,19)	417057	(2,2,2,2,2,1,2,2,2,1)
48	(28,20)	408077	(1,1,2,1,2,1,2,2,2,1)
52	(22,30)	399290	(2,2,2,2,2,2,2,2,2,1)
56	(32,24)	391190	(2,2,2,2,2,2,2,2,2,1)
60	(32,28)	391190	(2,2,2,2,2,2,2,2,2,1)
64	(32,32)	391190	(2,2,2,2,2,2,2,2,2,1)

(a)

		Optimum	Test bus assignment
W	(w_1, w_2)	testing time	vector
16	(15,1)	2423712	(2,2,2,1,2,1,2,1,1,1)
20	(1,19)	2363126	(2,2,1,2,1,2,1,2,2,2)
24	(23,1)	2278443	(2,1,1,1,2,1,2,1,1,1)
32	(3,29)	2202286	(2,2,2,2,1,2,2,2,2,1)
36	(4,32)	2174501	(2,2,2,2,1,2,2,1,1,2)
40	(9,31)	2149720	(2,2,2,2,1,2,2,2,2,1)
44	(12,32)	2123437	(2,2,2,2,1,2,2,2,2,1)
48	(32,16)	2099390	(2,1,1,1,2,1,1,1,1,2)
52	(32,20)	2086542	(2,2,1,1,2,1,1,1,1,2)
56	(25,31)	2069738	(2,2,2,2,1,2,1,2,2,2)
60	(28,32)	2044346	(2,2,2,2,1,2,1,2,2,2)
64	(32,32)	2029753	(2,2,1,2,2,1,1,1,1,2)

(b)

Table 1. Optimum testing time and optimal width distribution for: (a) $\mathcal{S}_1$ (b) $\mathcal{S}_2$.

		Optimum	Increase in testing
W	(w_1, w_2)	testing time	time (percent)
32	(19,13)	445300	0.38
36	(21,15)	435309	0.06
40	(23,17)	427639	0.37
44	(25,19)	418355	0.31

(a)

			Increase in testing
W	(w_1, w_2)	testing time	time (percent)
32	(1,31)	3585675	62.82
36	(4,32)	3562356	63.82
40	(8,32)	3562356	65.71
44	(12,32)	3562356	67.77

(b)

Table 2. Optimum testing time and optimal width distribution with two test buses for (a) $\mathcal{S}_1$ and (b) $\mathcal{S}_2$, when the assignment of cores to test buses is pre-determined as (1,1,1,1,2,2,2,2,2,1).

3 Reward model for place-and-route constraints

In this section, we present a *reward model* that allows the system integrator to incorporate *preferences* arising from place-and-route constraints. As discussed in Section 2, place-and-route constraints may be hard constraints that require the assignment of cores to test buses to be pre-determined. However, the system integrator may often express these constraints as *preferences* which the ILP model should *attempt to* satisfy while determining an optimal test access architecture. We consider a scenario in which preferences are stated for pairs of cores that should (preferably) be assigned to the same bus. For example, it may be desirable that cores i and j be assigned to the same bus. We denote this preference using the notation $r_{ij} = 1$ where r_{ij} is a 0-1 constant used in the ILP model. The *reward* R for a test access architecture is measured by $R = \sum_{k=1}^{N_B} \sum_{i=j+1}^{N_C} \sum_{j=1}^{N_C} x_{ik} x_{jk} r_{ij}$.

The formal problem statement is as follows:

- $\mathcal{P}2$: Given N_C cores, N_B test buses, maximum test data width W, reward R, and a *preferred* assignment of cores to test buses, determine the optimum testing time, an optimal distribution of the test width among the test buses, and an optimal assignment of cores to test buses, such that the *reward* exceeds R.

The ILP model for $\mathcal{P}2$ is derived from $\mathcal{P}1$ by adding the constraint $\sum_{k=1}^{N_B} \sum_{i=j+1}^{N_C} \sum_{j=1}^{N_C} x_{ik} x_{jk} r_{ij} \geq R$. This non-linear constraint can be linearized by replacing the product $x_{ik} x_{jk}$ by a new binary variable u_{ijk} and adding the following two inequalities for each such substitution:

1) $x_{ik} + x_{jk} - u_{ijk} \leq 1$

2) $-x_{ik} - x_{jk} + 2u_{ijk} \leq 0$

We applied the reward model to $\mathcal{S}_1$ and $\mathcal{S}_2$ for various values of R—the results are shown in Table 3. We assumed that $r_{12}, r_{13}, r_{14}, r_{110}, r_{23}, r_{24}, r_{210}, r_{34}, r_{310}, r_{410} = 1$, and $r_{56}, r_{57}, r_{58}, r_{5,9}, r_{6,7}, r_{6,8}, r_{6,9}, r_{7,8}, r_{7,9}, r_{8,9} = 1$. A higher value of R implies that a larger number of preferences must be satisfied by any solution to $\mathcal{P}2$. In our case, $R = 20$ implies that the preferences are hard constraints and that the test bus assignment is pre-determined. On the other hand, $R = 0$ implies that no preferences are outlined and the results of Table 2 are obtained. Note that for $\mathcal{S}_2$, for values of R upto 16, the preferences can be satisfiedt without increasing the testing time significantly. However, for higher preference values, a considerable penalty in testing time is incurred.

	Reward		Optimum
W	R	(w_1, w_2)	testing time
32	0	(18,14)	443620
32	12	(18,14)	443719
32	16	(18,14)	443798
32	20	(19,13)	445300
36	0	(21,15)	435042
36	14	(21,15)	435089
36	16	(21,15)	435089
36	20	(21,15)	435309
40	0	(17,23)	426043
40	12	(23,17)	426043
40	16	(23,17)	426043
40	20	(23,17)	427639
44	0	(25,19)	417057
44	12	(26,18)	417402
44	16	(26,18)	418175
44	20	(25,19)	418355

(a)

	Reward		Optimum
W	R	(w_1, w_2)	testing time
32	0	(3,29)	2202286
32	12	(18,14)	2205479
32	16	(18,14)	2314969
32	18	(1,31)	3585675
32	20	(1,31)	3585675
36	0	(4,32)	2174501
36	14	(21,15)	2268083
36	16	(21,15)	2278541
36	18	(4,32)	3562356
36	20	(4,32)	3562356
40	0	(9,31)	2149720
40	12	(13,27)	2193692
40	16	(21,19)	2242113
40	18	(8,32)	3562356
40	20	(8,32)	3562356
44	0	(12,32)	2123437
44	12	(22,22)	2201049
44	16	(23,21)	2205685
44	18	(12,32)	3562356
44	20	(12,32)	3562356

(b)

Table 3. Optimum testing time and optimal width distribution with two test buses for (a) $\mathcal{S}_1$ and (b) $\mathcal{S}_2$, with the reward model.

4 Power consumption constraints

Power consumption is an important consideration in the testing of SOCs. If several embedded cores are tested in parallel, the resulting power consumption may exceed the power rating of the SOC. The system integrator can partially address this issue by limiting the number of test buses, which automatically limits the amount of test parallelism available at the system level. Nevertheless, special care must also be taken in the design of the test access architecture—for example, if two cores assigned to two different test buses are tested concurrently, the resulting power consumption may exceed the system power rating.

If core i with test width ϕ_i is assigned to a test bus with width w_j, the power consumed by it during testing depends on the relative values of ϕ_i and w_j. As a result of test data serialization, it takes $(\phi_i - w_j + 1)$ cycles to apply each test pattern to core i. This implies that the patterns are applied to the core at the rate of $1/(\phi_i - w_j + 1)$ per cycle. If we assume that each pattern for core i consumes G_i units of energy, then the power consumed during testing is given by $P_i = G_i/(\phi_i - w_j + 1)$. Thus, while higher test data width leads to less testing time, it also leads to higher power consumption due to greater activity in the core under test.

We now incorporate system-level power constraints in the ILP model of Figure 2. Without loss of generality, we assume that the energy consumed by each test pattern for a core is proportional to the number of gates in the core. We also assume that the total power budget is divided equally among the N_B test buses, i.e. if the total power budget is P_{max}, the power allocated to each test bus is P_{max}/N_B. This allows the system designer to schedule tests without considering power constraints once the test access architecture has been determined. Alternative strategies for allocating the power budget can easily be incorporated into the ILP model.

The formal problem statement is as follows:

- $\mathcal{P}3$: Given N_C cores, N_B test buses, maximum test data width W, maximum power budget P_j for each test bus, $1 \leq j \leq N_B$, determine the optimum testing time, an optimal distribution of the test width among the test buses, and an optimal assignment of cores to test buses.

The ILP model for $\mathcal{P}_3$ is derived from $\mathcal{P}_1$ by adding the following (linear) constraint:

$$x_{ij} \frac{G_i}{\phi_i - w_j + 1} \leq P_j, 1 \leq i \leq N_C, 1 \leq j \leq N_B.$$

This implies that $x_{ij}G_i + w_j P_j \leq P_j(1 + \phi_i)$, $1 \leq i \leq N_C$, $1 \leq j \leq N_B$. G_i can be obtained from power models for core i. Experimental results for power-constrained test access architecture design for $\mathcal{S}_1$ and $\mathcal{S}_2$ are shown in Table 4. For our experiments, we approximated G_i by the number of gates in core i. As expected, the additional power constraints increase the testing time for both $\mathcal{S}_1$ and $\mathcal{S}_2$. However, we note that for small values of the total test width W, power constraints have less impact on the optimum testing time. For example, power constraints have no impact on $\mathcal{S}_1$ for $W = 20$ and $W = 24$. On the other hand, for higher values of W, the testing time is affected substantially. For example, in Table 4(a), for $W \geq 24$ and power budget of 300 units, the testing time does not decrease with an increase in W due to power constraints. In some cases, the ILP problem may even be infeasible for higher test widths, e.g. in Table 4(b) with $W = 48$ and power budget of 300 units for $\mathcal{S}_2$. Comparing with Table 2, we note that the width distribution is also significantly different due to power constraints.

Another striking observation is that under power constraints, the testing time may increase with higher test bus width. For example, for $\mathcal{S}_1$, with $P_1 = P_2 = 300$, the testing time for $W = 20$ is 471624 cycles. This is achieved using a width distribution of (10,10) and test bus assignment (2,2,2,2,2,2,2,2,2,1). However, as seen from Table 4. for test width $W = 24$, the test bus assignment has to be changed to meet power constraints, and the minimum testing time increases to 471900 cycles.

Finally, we investigate the combined impact of place-and route and power constraints on the width distribution and testing time. The experimental results for $\mathcal{S}_1$ and $\mathcal{S}_2$ are shown in Table 5. As expected, the testing time increases more if both these constraints are included in the ILP model. However, this increase is significant only if the total test width W is high. For smaller values of W, these constraints have little impact on the total testing time.

5 Conclusions

We have presented a formal methodology for designing optimal test access architectures for SOCs under place-and-route and power constraints. In doing so, we have attempted to provide a formal basis for comparing the several ad hoc test access architectures that have been proposed in the literature. The proposed methodology allows designers to explore design options and make appropriate choices. We have examined several problems related to the design of optimal test architectures. These include the assignment of cores to test buses and the distribution of a given test data width among multiple test buses. We have introduced a new reward model that allows the system integrator to specify place-and-route preferences. System-level power constraints can be easily incoporated into the ILP models. We have successfully applied these models to two non-trivial core-based systems, and solved them using a standard software package available in the public domain.

References

[1] J. Aerts and E. J. Marinissen. Scan chain design for test time reduction in core-based ICs. *Proc. International Test Conference*, pp. 448–457, 1998.

[2] F. Beenker, B. Bennetts and L. Tijssen. *Testability Concepts for Digital ICs—the Macro Test Approach*, vol. 3 of *Frontiers in Electronic Testing*. Kluwer Academic Publishers, Boston, 1995.

[3] M. Berkelaar. *lpsolve*, version 2.0. Eindhoven University of Technology, Design Automation Section, Eindhoven, The Netherlands. E-mail: michel@es.ele.tue.nl.

[4] K. Chakrabarty. Design of system-on-a-chip test access architectures using integer linear programming. Accepted for publication in *Proc. 2000 IEEE VLSI Test Symposium*.

[5] K. Chakrabarty. Test scheduling for core-based systems. *Proc. International Conference on Computer-Aided Design*, pp. 391–394, November 1999.

[6] I. Ghosh, N. K. Jha and S. Dey. A low overhead design for testability and test generation technique for core-based systems. *Proc. International Test Conference*, pp. 50–59, 1997.

Total	$P_1 = P_2 = 400$			$P_1 = P_2 = 300$		
test	Optimal width	Optimum	Test bus	Optimal width	Optimum	Test bus
width	distribution	testing	assignment	distribution	testing	assignment
W	(w_1, w_2)	time	vector	(w_1, w_2)	time	vector
20	(11,9)	470380*	(2,1,1,2,2,2,2,2,2,1)	(10,10)	471624	(2,2,2,2,2,2,2,2,2,1)
24	(11,13)	461277*	(2,1,1,1,1,1,1,1,1,2)	(10,14)	471900	(1,2,2,2,2,2,2,2,2,1)
28	(16,12)	452781	(1,1,1,1,1,2,1,2,2,1)	(10,18)	471900	(2,2,2,2,2,1,2,2,2,1)
32	(18,14)	443624	(1,1,1,1,1,2,1,2,2,1)	(10,22)	471900	(1,2,2,2,2,2,2,2,2,1)
36	(20,16)	435064	(2,2,2,2,2,2,2,2,2,1)	(10,26)	471900	(1,2,2,2,2,2,2,2,2,1)

(a)

Total	$P_1 = P_2 = 400$			$P_1 = P_2 = 300$		
test	Optimal width	Optimum	Test bus	Optimal width	Optimum	Test bus
width	distribution	testing	assignment	distribution	testing	assignment
W	(w_1, w_2)	time	vector	(w_1, w_2)	time	vector
28	(27,1)	2298179	(2,2,1,2,2,1,1,1,2,1)	(17,11)	2377579	((2,2,2,2,1,2,1,2,1,2)
32	(28,4)	2273109	(2,2,2,2,2,1,1,1,2,1)	*lpsolve* did not run to completion		
36	(29,7)	2233640	(2,2,2,2,2,1,1,1,2,1)	(26,10)	2322329	(2,2,1,2,1,2,1,2,1,2)
40	(32,8)	2179864	(2,2,1,2,2,1,1,1,2,1)	(30,10)	2300256	(2,1,1,2,1,2,1,2,1,2)
48	(23,16)	2167329	(2,2,2,2,2,1,1,1,2,1)	Infeasible		

(b)

*Equals testing time without power constraints

Table 4. Experimental results on optimal test access architecture design under power constraints: (a) $\mathcal{S}_1$ (b) $\mathcal{S}_2$.

Total test width W	Optimum testing time	Optimal width distribution (w_1, w_2)	Test bus assignment vector	Percent testing time increase (I)	Percent testing time increase (II)	Percent testing time increase (III)
32	2314969	(17,15)	(1,1,1,1,1,2,2,2,1)	5.1	0	1.8
36	2298428	(17,19)	(1,1,1,1,1,2,2,2,2,1)	5.7	0.9	2.9
40	2298428	(17,23)	(1,1,1,1,1,2,2,2,2,1)	6.9	2.5	5.4

I: over no place-and-route or power constraints, II: over no place-and-route constraints, III: over no power constraints

Table 5. Experimental results on optimal test access architecture design for $\mathcal{S}_2$ under power and reward constraints, $P_1 = P_2 = 400$, $R = 16$.

[7] I. Hamzaoglu and J. H. Patel. Test set compaction algorithms for combinational circuits. *Proc. International Conf. on Computer Aided Design*, pp. 283-289, 1998

[8] V. Immaneni and S. Raman. Direct access test scheme–design of block and core cells for embedded ASICs. *Proc. International Test Conference*, pp. 488–492, 1990.

[9] E. J. 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 Conference*, pp. 284–293, 1998.

[10] E. J. Marinissen, Y. Zorian, R. Kapur, T. Taylor and L. Whetsel. Towards a standard for embedded core test: an example. *Proc. International Test Conference*, pp. 616–627, 1999.

[11] M. Sugihara, H. Date and H. Yasuura. A novel test methodology for core-based system LSIs and a testing time minimization problem. *Proc. International Test Conference*, pp. 465–472, 1998.

[12] P. Varma and S. Bhatia. A structured test re-use methodology core-based system chips. *Proc. International Test Conference*, pp. 294–302, 1998.

[13] H. P. Williams. *Model Building in Mathematical Programming*, 2nd ed., John Wiley, New York, 1985.

[14] Y. Zorian, E. J. Marinissen and S. Dey. Testing embedded-core based system chips. *Proc. International Test Conference*, pp. 130–143, 1998.