Design of System-on-a-Chip Test Access Architectures using Integer Linear Programming[1]

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

Abstract

Test access is a major problem for system-on-a-chip (SOC) designs. Since embedded cores in an SOC are not directly accessible via chip I/Os, special access mechanisms are required to test them after system integration. An efficient test access architecture should reduce test cost and time-to-market by minimizing test application time. We address several issues related to the design of test access architectures. Even though these design problems are NP-complete, they can be solved exactly using integer linear programming (ILP). As a case study, the ILP models for two hypothetical but representative systems are solved using a public-domain ILP software package.

1 Introduction

Embedded cores are now being increasingly used in large system-on-a-chip (SOC) designs [14]. However, the manufacturing test and debug of such SOC designs remains a major challenge. Since embedded cores are not directly accessible via chip inputs and outputs, special access mechanisms are required to test them after system integration. The development of efficient test access architectures is of considerable interest to the SOC design and test community.

A test access architecture, also referred to as a test access mechanism (TAM), provides means for on-chip test data transport [14]. A number of test access architectures have been proposed in the literature [14]. These include macro test [2], core transparency [6], dedicated test bus [12], and multiplexed access [8], and a bus architecture based on the concept of a TESTRAIL [9]. A TESTRAIL provides a flexible and scalable test access mechanism; a single TESTRAIL can provide access to one or more cores, and an IC may contain one or more TESTRAILs of varying widths. The width of a TESTRAIL is referred to as the test data bandwidth since it determines the overall system testing time. Figure 1, derived from [9], illustrates one possible implementation of the TESTRAIL architecture. (The core wrapper and the bypass mechanism are not explicity shown in the figure.)

[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.

In order to reduce test cost and shorten short time-to-market, the testing time for an SOC should be minimized by adopting an appropriate test access architecture. While TESTRAIL allows the system designer to trade off testing time with area overhead by varying tests data bandwidths, the precise relationship between the testing time and the test access architecture has not been formally studied. Related prior work has either been limited to test scheduling for a given test access mechanism [4, 13], or to determine 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. 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].

The following TAM design problems are therefore of interest to the system integrator: (1) Given an SOC and maximum test data bandwidth, how should the bandwidth be distributed among the various test buses in order to minimize the testing time? (2) How should the cores in an SOC be assigned to the test buses? (3) For a given test access architecture, how much test data bandwidth is required to meet specified testing time objectives? To the best of our knowledge, this paper presents the first systematic solutions to these SOC design problems.

The main contributions of the paper are listed below.

- We formulate several design problems related to test access architectures, and show that all these problems are NP-complete.
- We develop an integer linear programming (ILP) model for optimally assigning cores to test buses when the bandwidths of the test buses are known. We refer to this as the "test bus assignment problem".
- We develop an ILP model for minimizing the testing time by combining optimal bandwidth distribution with optimal test bus assignment.
- Given a constraint on the maximum testing time, we develop an ILP model to determine the minimum test data bandwidth and an optimal assignment of cores to test buses.

0-7695-0613-5/00 $10.00 © 2000 I

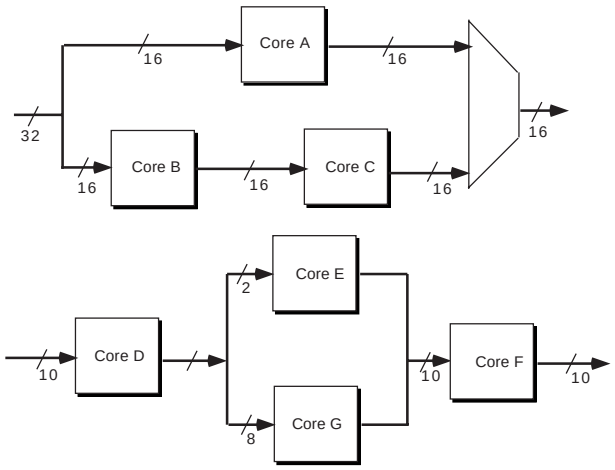

Figure 1. An example of the TESTRAIL architecture.

- The above ILP models make the simplifying assumption that a test bus cannot be subdivided into test buses of smaller bandwidth which subsequently merge before the test data is transported to the IC outputs. In order to account for this realistic scenario, we refine our ILP models to allow a test buses to fork into test buses of smaller bandwidth.
- 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.

The organization of the paper is as follows. In Section 2, we briefly review ILP and formulate the problem of optimal test bus assignment. In Section 3, we develop ILP models for minimizing the testing time by determining an optimal test bandwidth distribution. In Section 4, we present case studies for two example SOCs (described below). Finally, in Section 5, we extend our basic ILP models to handle cases where a test bus may fork into several test buses that subsequently merge before the test data is transported to the IC outputs. We present experimental results on optimal and near-optimal subdivision of the test buses.

In order to illustrate the proposed optimization methods, we use the core-based SOCs $\mathcal{S}_1$ and $\mathcal{S}_2$ shown in Figure 2 as examples throughout the paper. These hyopthetical but non-trivial SOCs consist of ten ISCAS 85 and ISCAS 89 benchmark circuits each. We assume that the three ISCAS 89 circuits contain internal scan chains. $\mathcal{S}_1$ contains seven combinational cores and seven sequential cores, while $\mathcal{S}_2$ consists of two combinational cores and eight sequential cores. The complexity of the ILP models depends on the number of cores in the SOC and not on the sizes of the cores. For the sake of illustration, only two test buses are shown in Figure 2. Our ILP models can be easily used for any number of test buses.

2 Optimal assignment of cores to test buses

We first briefly review ILP using matrix notation [11]. The goal of ILP is to minimize a linear objective function

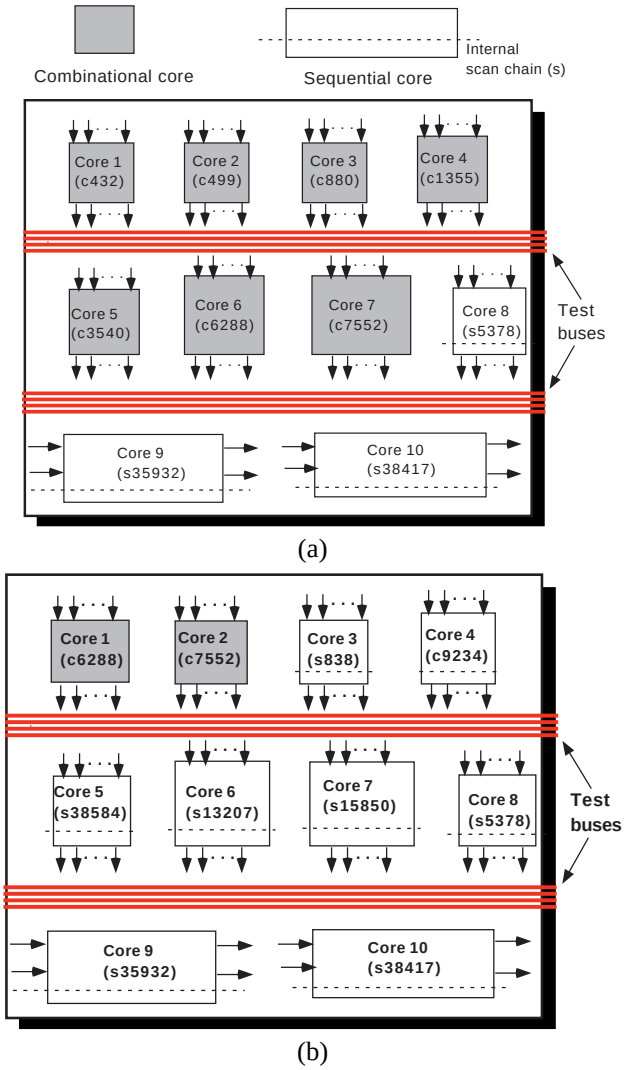

Figure 2. Two examples of SOCs: (a) $\mathcal{S}_1$ (b) $\mathcal{S}_2$.

on a set of integer variables, while satisfying a set of linear constraints. A typical ILP model is described as follows:

$$\text{minimize:} \quad \mathbf{A}\mathbf{x}$$
$$\text{subject to:} \quad \mathbf{B}\mathbf{x} \leq \mathbf{C}, \mathbf{x} \geq 0,$$

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. For our experiments (described in Sections 4 and 5), we used the *lpsolve* package from the Eindhoven University of Technology in the Netherlands [3].

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 system contains N_B test buses with bandwidths $w_1, w_2, \ldots, w_{N_B}$, respectively.

The problem that we address in this section is to minimize the system testing time by optimally assigning cores to test buses. It is formally stated as follows:

- $\mathcal{P}1$: Given N_C cores and N_B test buses of bandwidths $w_1, w_2, \ldots, w_{N_B}$, respectively, determine an assignment of cores to test buses such that the total testing time is minimized.

Note that $\mathcal{P}1$ is equivalent to the well-known multiprocessor scheduling problem[1], and is therefore NP-complete ([5], page 65). Nevertheless, we show that as in the case of many other NP-complete problems, $\mathcal{P}1$ can be solved exactly using ILP. We assume for now that a test bus does not fork (split) into multiple branches which may merge later. (This restriction will be removed in Section 5.) 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 issues.

We first note that 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 bandwidth of the test bus is insufficient for parallel testing, and serialization of the test data is necessary at 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 bandwidth 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 $\max_{j\in\{1,2,\ldots N_B\}} \sum_{i=1}^{N_C} T_{ij}x_{ij}$. We now formulate an ILP for minimizing the testing time.

Objective: Minimize $C = \max_j \sum_{i=1}^{N_C} T_{ij}x_{ij}$ subject to

[1] Test buses correspond to processors and test sets correspond to tasks.

Minimize C subject to:
1) $C \geq \sum_{i=1}^{N_C} T_{ij}x_{ij}, 1 \leq j \leq N_B$
2) $\sum_{j=1}^{N_B} x_{ij} = 1, 1 \leq i \leq N_C$
3) $x_{ij} = 0$ or 1

Figure 3. ILP model for $\mathcal{P}1$.

1) $\sum_{j=1}^{N_B} x_{ij} = 1, 1 \leq i \leq N_C$
2) $x_{ij} = 0$ or 1

The above minmax nonlinear cost function can easily be linearized [11]. The resulting ILP model is shown in Figure 3.

As an example, 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 three 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].

Consider the SOC $\mathcal{S}_1$ with $N_B = 2$. Let the total test data bandwidth be 48 bits, i.e. $w_1 + w_2 = 48$. In addition, let $w_1 = 32$ and $w_2 = 16$. The optimal assignment of cores to these two test buses is given by the vector (1,1,1,1,1,1,1,2,2,1), where a 1 (2) in position i of the vector indicates that core i is assigned to bus 1 (2). This is shown in Figure 4. The optimal testing time for these values of w_1 and w_2 obtained using *lpsolve* is 411884 cycles. Note that this is not the minimum testing time that can be achieved with a total test bandwidth of 48 bits. For example, a testing time of 408077 cycles is achieved using $w_1 = 28$, $w_2 = 20$, and the test bus assignment vector (1,1,2,1,2,1,2,2,2,1). In the next section, we will examine the problem of determining an optimal distribution of the total test data bandwidth.

The following theorem presents a lower bound on the total testing time when the bandwidths of the test buses are known. (Most proofs are omitted due to lack of space.) This lower bound can indeed be achieved in practice—we illustrate this below using $\mathcal{S}_1$ as an example.

Theorem 1 *For an SOC with N_C cores and N_B test buses with bandwidths $w_1, w_2, \ldots w_{N_B}$, respectively, a lower bound on the total testing time $\mathcal{T}$ is given by $\mathcal{T} \geq \max_i\{\min_j\{(\phi_i - w_j + 1)t_i\}\}$, where ϕ_i is the test width of core i.*

For the system $\mathcal{S}_1$ with two test buses of 32 bits and 16 bits, respectively, Theorem 1 provides a lower bound on the testing time of 391190 cycles. This corresponds to a test bus assignment in which only core 10 is assigned to test bus 1. Such an assignment is indeed optimal and the lower bound

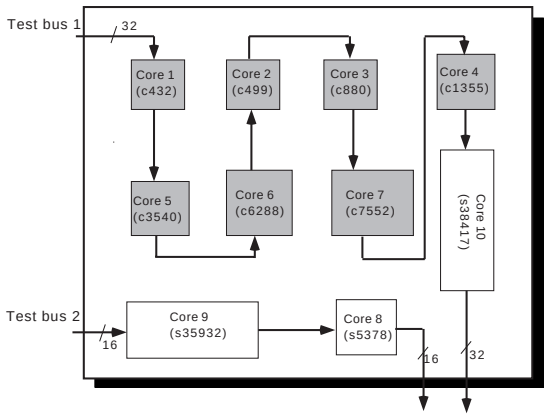

Figure 4. Optimal test bus assignment for $\mathcal{S}_1$ with two test buses of 32 bits and 16 bits, respectively.

of Theorem 1 is achieved if the bandwidth of the second test bus is increased, or if a third test bus is used.

3 Optimal test bus bandwidth

In this section, we examine the problem of minimizing SOC testing time by determining (i) optimal bandwidths for the test buses, and (ii) optimal assignment of cores to test buses. We assume that the total test bandwidth can be at most W. We also assume that the bandwidth of a test bus does not exceed the bandwidth required for any given core, i.e. $\max_j\{w_j\} \leq \min_i\{\phi_i\}$ for all values of i and j, and test data serialization is required for every core. This assumption is necessary to avoid complex non-linear models that are difficult to linearize. From a practical point of view, this implies that cores with very small test widths are assigned to test buses after the cores with larger test widths are optimally assigned. We will extend the ILP model and remove this restriction in Section 4.

We now formulate the problem of optimally allocating the total bandwidth among the N_B buses, as well as determining the optimal allocation of cores to these buses. The optimization problem is formally stated as follows:

- $\mathcal{P}2$: Given N_C cores and N_B test buses of total bandwidth W, determine the optimal bandwidth of the test buses, and an assignment of cores to test buses such that the total testing time is minimized.

$\mathcal{P}2$ can be shown to be NP-complete by restricting it to $\mathcal{P}_1$. Even though $\mathcal{P}2$ is NP-complete, it can be solved exactly for non-trivial SOCs. We now present an ILP for $\mathcal{P}2$ using the 0-1 variable x_{ij} defined in Section 2.

Minimize C subject to:

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

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) $y_{ij} - w_{max}x_{ij} \leq 0, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
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 5. ILP model for $\mathcal{P}2$.

5) $x_{ij} = 0$ 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 $\mathcal{P}2$ shown in Figure 5.

The ILP model for $\mathcal{P}_2$ is especially useful in determining the effect of increased test data bandwidth on the testing time. However, there is a limit to which the testing time can be decreased by simply increasing the system test bandwidth. The following theorem provides a lower bound on the testing time $\mathcal{T}$ for a core-based system. It is useful in determining the maximum test bandwidth beyond which the testing time cannot be decreased by simply increasing bandwidth.

Theorem 2 *For a core-based system with N_C cores, a lower bound on the total testing time $\mathcal{T}$ is given by $\mathcal{T} \geq \max_{i\in\{1,2,\ldots,N_C\}}\{(\phi_i - \min_i\{\phi_i\} + 1)t_i\}$.*

Proof: Let the system consist of N_B test buses with (undetermined) test widths $w_1, w_2, \ldots w_{N_B}$ such that $\min_i\{\phi_i \geq \max_j\{w_j\}$. We know from Theorem 1 that $\mathcal{T} \geq \max_i\{\min_j\{(\phi_i - w_j + 1)t_i\}\}$ Since $\min_j\{(\phi_i - w_j + 1)t_i\} = (\phi_i - \max_j\{w_j\} + 1)t_i$, and $\max_j\{w_j\} \leq \min_i\{\phi_i\}$, we have $\mathcal{T} \geq \max_i\{(\phi_i - \max_j\{w_j\} + 1)t_i\}$, which implies that $\mathcal{T} \geq \max_i\{(\phi_i - \min_i\{\phi_i\} + 1)t_i\}$. □

We next address the related optimization problem of determining the minimum test bandwidth required to meet a minimum testing time objective. In addition, we determine an optimal distribution of the bandwidth among the test buses, and an optimal test bus assignment. The optimization problem is formally stated as follows:

- $\mathcal{P}3$: Given N_C cores, N_B test buses, and a maximum testing time $\mathcal{T}$, determine the minimum total test bandwidth, an optimal distribution of the test bandwidth

Minimize W subject to:
1) $\sum_{j=1}^{N_B} w_j = W$
2) $\sum_{i=1}^{N_C}((\phi_i + 1)t_i x_{ij} - t_i y_{ij}) \leq \mathcal{T}, 1 \leq j \leq N_B$
3) $y_{ij} - w_{max} x_{ij} \leq 0, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
4) $-w_j + y_{ij} \leq 0, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
5) $w_j - y_{ij} + w_{max} x_{ij} \leq w_{max}, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
6) $\sum_{j=1}^{N_B} x_{ij} = 1, 1 \leq i \leq N_C$
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 6. ILP model for $\mathcal{P}3$.

among the test buses, and an optimal assignment of cores to test buses.

$\mathcal{P}3$ can also be shown to be NP-complete. However, as in the case of $\mathcal{P}2$, even though $\mathcal{P}3$ is NP-complete, it can be solved exactly using ILP. The ILP model for $\mathcal{P}3$ can be derived directly from $\mathcal{P}2$ and is shown in Figure 6. The following theorem relates the bandwidth of the widest test bus to the minimum testing time $\mathcal{T}$ and the test widths of the cores.

Theorem 3 *Let $\{w_1, w_2, \ldots, w_{N_B}\}$ be the optimal bandwidth distribution for a core-based system with N_C cores and maximum testing time $\mathcal{T}$. A lower bound on the bandwidth of the widest test bus is given by* $\max_j\{w_j\} \geq \max_i\{\phi(i) - \mathcal{T}/t_i + 1\}$.

As an example, consider $\mathcal{S}_1$ with two test buses as shown in Figure 2. If an upper bound $\mathcal{T} = 430000$ cycles is placed on the testing time, then Theorem 3 yields $\max_j\{w_j\} = 22$. As demonstrated in Table 2, this lower bound on the test bus bandwidth is achieved using the ILP model for $\mathcal{P}3$, hence Theorem 3 provides a tight lower bound.

4 Case studies

In this section, we present case studies using $\mathcal{S}_1$ and $\mathcal{S}_2$ for the optimization problems $\mathcal{P}2$ and $\mathcal{P}3$. (Solutions for the optimization problem $\mathcal{P}1$ were presented in Section 2.) We also remove some of the restrictions that were imposed in Sections 2 and 3 in order to simplify the ILP models. We solved the ILP models using the *lpsolve* software package on a Sun Ultra 10 workstation with a 333 MHz processor and 128 MB memory. The user time for $\mathcal{P}1$ was less than one minute in all cases, while the user time for $\mathcal{P}2$ and $\mathcal{P}3$ 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 bandwidth, optimal bandwidth 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 lower bound of 391190 cycles predicted by Theorem 3 is reached for $W = 56$ bits. Any further increase in the system test bandwidth 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 bandwidth and bandwidth distribution for $\mathcal{S}_1$ and $\mathcal{S}_2$ with two test buses for various values of the maximum testing time $\mathcal{T}$.

W	(w_1, w_2)	Optimum testing time	Test bus assignment 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)

* Lower bound on the system testing time (Theorem 3)

(a)

W	(w_1, w_2)	Optimum testing time	Test bus assignment 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 bandwidth distribution for: (a) $\mathcal{S}_1$ (b) $\mathcal{S}_2$.

We now describe how the ILP models can be extended to remove the restriction $\max_j\{w_j\} \geq \min_i\{\phi_i\}$. This is necessary to decrease the testing time below the limit of Theorem 3 if greater test bandwidth is available. Let δ_{ij} be an "indicator" 0-1 variable defined as follows:

$$\delta_{ij} = \begin{cases} 1, \text{ if } w_j > \phi_i \\ 0, \text{ otherwise} \end{cases}$$

The testing time T_{ij} for core i assigned to test bus j can now be expressed as:

$$T_{ij} = \delta_{ij} x_{ij} t_i + (1 - \delta_{ij})(\phi_i - w_j + 1) t_i x_{ij}$$

with the constraint that $\delta_{ij}(w_j - \phi_i) + (1 - \delta_{ij})(\phi_i - w_j) \geq 0$. The non-linear terms in this formulation can be linearized as in Section 3, and the resulting ILP model can be easily solved to obtain optimal bandwith allocation and test bus

Maximum testing time $\mathcal{T}$	W	(w_1, w_2)	Test bus assignment vector
400000	52	(21,31)	(2,2,2,2,2,2,1,1,1,2)
410000	48	(27,21)	(2,2,2,2,2,2,2,2,2,1)
420000	43	(25,18)	(2,2,2,2,1,1,2,2,2,1)
430000	39	(22,17)	(2,2,2,2,2,2,2,2,2,1)
440000	34	(19,15)	(2,2,2,2,2,2,2,2,2,1)
450000	30	(16,14)	(2,2,2,2,2,2,2,2,2,1)
460000	25	(14,11)	(2,2,2,1,2,2,2,2,2,1)
470000	21	(11,10)	(2,2,2,2,2,2,2,2,2,1)
480000	16	(8,8)	(2,2,2,2,2,2,2,2,2,1)

Table 2. Total bandwidth and bandwidth distribution for $\mathcal{S}_1$ for a given maximum testing time.

W	(w_1, w_2)	Optimum testing time	Test bus assignment vector
48	(40,8)	2020973	(2,2,2,1,2,1,1,1,2,1)
56	(42,14)	1967215	(2,2,2,1,2,1,1,1,2,1)
60	(43,17)	1940336	(2,2,2,1,2,1,1,1,2,1)
64	(53,11)	1937253	(2,2,2,2,2,1,1,1,1,1)
80	(48,32)	1867442	(2,1,2,2,2,1,1,1,2,1)

Table 3. Results for $\mathcal{S}_2$ with no explicit bound on the width of the individual test buses.

assignment. We solved the ILP model for $\mathcal{S}_2$, and the results shown in Table 3 indicate that significant reductions in testing time are achieved, especially for higher test bandwidths.

The ILP formulation also allows us to decrease the test bandwidth of the cores in the system, i.e. the number of lines that connect the cores to their respective test buses, without increasing the overall testing time. For example, if core i is connected to test bus j of width w_j then even though w_j lines are available for propagating test data to and from core i, it is not always necessary to use all these lines. In such cases, $b_i < w_j$ lines connect core i to test bus j. We refer to b_i as the test bandwidth of core i. The motivation for using smaller core test bandwidth lies in the reduction of routing and interconnect area for SOC testing.

We allow b_i to be less than w_j in our ILP model by introducing the constraint $x_{ij}b_i \leq w_j$, which implies that the test bandwidth for core i may be less than the bandwidth of test bus j if core i is assigned to bus j. We also replace w_j by b_i in the expression for T_{ij}. The product term $x_{ij}b_i$ does not pose a problem since it is linearized for both $\mathcal{P}2$ and $\mathcal{P}3$. Table 4 shows how the test bandwidth cores in $\mathcal{S}_2$ can be reduced without increasing the system testing time.

5 Optimal subdivision of test buses

In this section, we allow the bandwidth of the test buses to be distributed among several buses with smaller bandwidths. This allows the w_j bits of test bus j to be divided into several parts, each of which can test one or more cores

W	w_1	w_2	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9	b_{10}
20	1	19	**4**	**3**	1	**17**	1	**19**	1	**18**	**17**	**18**
32	3	29	**1**	**12**	29	**26**	3	29	29	**28**	29	3
36	4	32	**1**	**16**	32	32	4	32	**31**	4	4	**30**
40	9	31	**7**	**6**	31	**28**	9	31	31	**30**	31	9

Table 4. Reducing the test bandwidth for cores in $\mathcal{S}_2$.

in parallel. We first make the simplifying assumption that the bandwidth w_j for each test bus j is known. Later we will extend our model to the case where the bandwidths are not known and optimal bandwidths have to be determined. The optimization problem being considered here is stated formally below.

- $\mathcal{P}4$: Given N_C cores, N_B test buses with known bandwidths $w_1, w_2, \ldots, w_{N_B}$, respectively, and an upper limit j_{max} on the number of subdivisions allowed for test bus j, $1 \leq j \leq N_B$, determine (i) an optimal subdivision of test bus bandwidths, and (ii) an optimal assignment of cores to test buses such that the total testing time is minimized.

Note that $\mathcal{P}4$ can also be shown to be NP-complete using the method of restriction. Let x_{ij} be a 0-1 variable as defined in Section 2. Test bus j can be divided into a maximum of j_{max} parts, each part serving as a test bus for a subset of cores in the system. Suppose that these parts have bandwidths $w_{j1}, w_{j2}, \ldots w_{jj_{max}}$, respectively, such that $\sum_{k=1}^{j_{max}} w_{jk} = w_j$, $1 \leq j \leq N_B$.

Let y_{ijk} be a 0-1 variable defined as follows:

$$y_{ijk} = \begin{cases} 1, \text{ if core } i \text{ is assigned to the } k\text{th part of bus } j \\ 0, \text{ otherwise} \end{cases}$$

The following constraint follows directly from the definitions of the 0-1 variables. It denotes the fact that a core is either assigned to a test bus with its complete bandwidth or to a portion of a test bus (with reduced bandwidth).

$$\sum_{j=1}^{N_B} \sum_{k=1}^{j_{max}} y_{ijk} + \sum_{j=1}^{N_B} x_{ij} = 1, 1 \leq i \leq N_C$$

If core i is assigned the entire bandwidth of bus j then its testing time is $(\phi_i - w_j + 1)t_i$. (We assume as before that $\max_j\{w_j\} \leq \min_i\{\phi_i\}$.) On the other hand, if it is assigned to the kth test bus derived from bus j, then its testing time is $(\phi_i - w_{jk} + 1)t_i$.

The cost function (testing time) C can now be expressed in terms of the above parameters.

$$C = \max_j\{\max_k \sum_{i=1}^{N_C} (\phi_i - w_{jk} + 1)t_i y_{ijk} + \sum_{i=1}^{N_C} (\phi_i - w_j + 1)t_i x_{ij}\}$$

The right hand side of the above equation is non-linear, but it can be linearized as before by a sequence of transformations. Let $C_{1j} = \max_k \sum_{i=1}^{N_C} (\phi_i - w_{jk} + 1)t_i y_{ijk}$ and $C_{2j} = \sum_{i=1}^{N_C} (\phi_i - w_j + 1)t_i x_{ij}$. The cost function can then be expressed as $C = \max_j (C_{1j} + C_{2j})$ and the optimization problem can be formulated as:

Minimize C subject to:

Minimize C subject to:
1) $C \geq (C_{1j} + C_{2j}), 1 \leq j \leq N_B$
2) $C_{1j} \geq \sum_{i=1}^{N_C}((\phi_i + 1)t_i y_{ijk} - t_i r_{ijk}), 1 \leq j \leq N_B,$
$1 \leq k \leq j_{max}$
3) $r_{ijk} - W y_{ijk}, 1 \leq i \leq N_C, 1 \leq j \leq N_B, 1 \leq k \leq j_{max},$
W is an upper bound on w_{jk}
4) $-w_{jk} + r_{ijk} \geq 0, 1 \leq i \leq N_C, 1 \leq j \leq N_B,$
$1 \leq k \leq j_{max}$
5) $w_{jk} - r_{ijk} + W y_{ijk} \leq W, 1 \leq i \leq N_C, 1 \leq j \leq N_B,$
$1 \leq k \leq j_{max}$
6) $C_{2j} \geq \sum_{i=1}^{N_C}(\phi_i - w_j + 1)t_i x_{ij}, 1 \leq j \leq N_B$
7) $\sum_{j=1}^{N_B} x_{ij} = 1, 1 \leq i \leq N_C$
8) $\sum_{k=1}^{j_{max}} w_{jk} = w_j, 1 \leq j \leq N_B$
9) $w_j \leq \phi_i, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
10) $x_{ij} = 0$ or 1, $1 \leq i \leq N_C, 1 \leq j \leq N_B$
11) $y_{ijk} = 0$ or 1, $1 \leq i \leq N_C, 1 \leq j \leq N_B, 1 \leq k \leq j_{max}$

Figure 7. ILP model for $\mathcal{P}4$.

1) $C \geq (C_{1j} + C_{2j}), 1 \leq j \leq N_B$
2) $\sum_{k=1}^{j_{max}} w_{jk} = w_j, 1 \leq j \leq N_B$
3) $w_j \leq \phi_i, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
4) $x_{ij} = 0$ or 1, $1 \leq i \leq N_C, 1 \leq j \leq N_B$
5) $y_{ijk} = 0$ or 1, $1 \leq i \leq N_C, 1 \leq j \leq N_B, 1 \leq k \leq j_{max}$

We next linearize constraint 1). Note that C_{1j} can be linearized by adding a nonbinary, integer variable r_{ijk} for each i, j, k, and adding three constraints as in the case of $\mathcal{P}2$ in Section 3. This yields the ILP model for $\mathcal{P}4$ shown in Figure 7.

We next generalize $\mathcal{P}4$ to the case where the bandwidths of the test buses also need to be optimally determined. The formal statement of this problem is given below:

- $\mathcal{P}5$: Given N_C cores, N_B test buses with total bandwidth W, and an upper limit j_{max} on the number of subdivisions allowed for test bus j, $1 \leq j \leq N_B$, determine (i) an optimal bandwidth for each test bus, the optimal subdivision of the bandwidth of every test bus, and (ii) an assignment of cores to test buses such that the total testing time is minimized.

The problem $\mathcal{P}5$ can also be shown to be NP-complete by restricting it to $\mathcal{P}2$. Its ILP model, shown in Figure 8, is obtained by combining the ILP models for $\mathcal{P}2$ and $\mathcal{P}4$. Integer variables s_{ij} are introduced for linearization.

Finally, we present experimental results on solving optimization problems $\mathcal{P}4$ and $\mathcal{P}5$. We considered $\mathcal{S}_1$ and $\mathcal{S}_2$ with two test buses (1 and 2), and we modeled the situation where the first test bus can fork into at most two branches (1a and 1b). The objective of this set of experiments was twofold: (i) demonstrate that $\mathcal{P}4$ ($\mathcal{P}5$) provides lower testing time than $\mathcal{P}1$ ($\mathcal{P}2$), and (ii) show that even non-optimal solutions for $\mathcal{P}5$ provide lower testing time than $\mathcal{P}2$.

Minimize C subject to:
1) $C \geq (C_{1j} + C_{2j}), 1 \leq j \leq N_B$
2) $C_{1j} \geq \sum_{i=1}^{N_C}((\phi_i + 1)t_i y_{ijk} - t_i r_{ijk}), 1 \leq j \leq N_B,$
$1 \leq k \leq j_{max}$
3) $r_{ijk} - W y_{ijk}, 1 \leq i \leq N_C, 1 \leq j \leq N_B, 1 \leq k \leq j_{max},$
W is an upper bound on w_{jk}
4) $-w_{jk} + r_{ijk} \geq 0, 1 \leq i \leq N_C, 1 \leq j \leq N_B,$
$1 \leq k \leq j_{max}$
5) $w_{jk} - r_{ijk} + W y_{ijk} \leq W, 1 \leq i \leq N_C, 1 \leq j \leq N_B,$
$1 \leq k \leq j_{max}$
6) $C_{2j} \geq \sum_{i=1}^{N_C}((\phi_i + 1)t_i x_{ij} - t_i s_{ij}), 1 \leq j \leq N_B$
7) $s_{ij} - w_{max} x_{ij} \leq 0, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
8) $-w_j + s_{ij} \leq 0, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
9) $w_j - s_{ij} + w_{max} x_{ij} \leq w_{max}, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
10) $\sum_{j=1}^{N_B} x_{ij} = 1, 1 \leq i \leq N_C$
11) $\sum_{k=1}^{j_{max}} w_{jk} = w_j, 1 \leq j \leq N_B$
12) $w_j \leq \phi_i, 1 \leq i \leq N_C, 1 \leq j \leq N_B$
13) $x_{ij} = 0$ or 1, $1 \leq i \leq N_C, 1 \leq j \leq N_B$
14) $y_{ijk} = 0$ or 1, $1 \leq i \leq N_C, 1 \leq j \leq N_B, 1 \leq k \leq j_{max}$

Figure 8. ILP model for $\mathcal{P}5$.

We first return to the example based on $\mathcal{S}_1$ which we presented in Section 2 to illustrate $\mathcal{P}1$. For this example, $w_1 = 32$, $w_2 = 16$, and an optimal testing time of 411884 cycles was obtained using $\mathcal{P}1$. By allowing the 32-bit bus to fork into two branches of 27 bits (1a) and 5 bits (1b) each, we achieve a reduced, but non-optimal, testing time of 409472 cycles using the test bus assignment (2,2,2,2,2,2,2,2,1b,1a).

Unfortunately, *lpsolve* did not run to completion for all cases when we attempted to solve $\mathcal{P}5$. Nevertheless, we allowed it to run for upto 2 hours, after which we tabulated the best solution obtained. These results (optimal and non-optimal) for $\mathcal{P}5$ are presented in Tables 5 and 6. The experimental results show that the added flexibility of allowing test buses to be subdivided can reduce the testing time significantly, especially for an SOC such as $\mathcal{S}_2$. Note also that subdivision also provides the same minimum testing time with 36-bit bandwidth as with a 44-bit-bandwidth for $\mathcal{S}_1$. Figure 9 illustrates an optimal test access architecture based on $\mathcal{P}5$ for $\mathcal{S}_1$ when the total bandwidth of 36 bits and at most one subdivision of the first test bus is allowed.

6 Conclusions

We have presented a formal methodology for designing optimal test access architectures for testing SOC designs. 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, distribution of

W	(w_1, w_2)	Distribution of w_1	Testing time	Test bus assignment vector
20	(19,1)	(1,18)	441404*	(1b,1a,1a,1a,1a,1a,2,2,1a,1b)
24	(23,1)	(1,22)	426564*	(1b,1a,1a,1a,1a,1a,2,2,1b,1b)
28	(27,1)	(26,1)	412427*	(1a,1b,1b,1b,1b,1b,2,2,1b,1a)
32	(19,13)	(1,18)	441404	(1b,1a,1a,1a,2,1a,2,2,2,1b)
36	(32,4)	(31,1)	394012*	(1a,1b,1b,1b,1b,1b,2,2,1a,1a)
44	(32,12)	(31,1)	394012*	(1a,1b,1b,1b,1b,1b,2,2,1b,1a)
52	(22,20)	(1,21)	397748	(1b,1a,1a,1a,2,1a,1,1,1a,2)

(a)

W	(w_1, w_2)	Distribution of w_1	Testing time	Test bus assignment vector
24	(23,1)	(12,11)	1677735*	(1a,1a,1a,1a,1b,1a,2,2,2,2)
36	(28,8)	(16,12)	1672265	(1a,1,1a,1a,1b,1a,2,2,2,2)
40	(30,10)	(18,12)	1672119	(1a,1,1a,1a,1b,1a,2,2,2,2)
44	(32,12)	(17,15)	1633600	(1a,1,1a,1a,1b,1a,2,2,2,2)

(b)

* Optimum testing time (*lpsolve* ran to completion)

Table 5. Results on $\mathcal{P}5$ for: (a) $\mathcal{S}_1$ (b) $\mathcal{S}_2$.

W	Percentage improvement over $\mathcal{P}2$ (for $\mathcal{S}_1$)	Percentage improvement over $\mathcal{P}2$ (for $\mathcal{S}_2$)
20	6.16	—
24	7.53	23.36
28	8.91	—
32	0.50	—
36	9.43	23.10
40	—	22.22
44	5.53	23.19
52	0.39	—

Table 6. Improvement in testing time obtained by using $\mathcal{P}5$ over $\mathcal{P}2$.

a given test data bandwidth among multiple test buses, and determining the amount of test data bandwidth required to satisfy an upper bound on the testing time. We have shown that even though these design problems are NP-complete, they can be efficiently modeled using integer linear programming. We have applied these models to two non-trivial SOCs, and solved them using a standard software package available in the public domain. We are currently extending the ILP models to incorporate routing and additional power constraints.

References

[1] J. Aerts and E. J. Marinissen. Scan chain design for test time reduction in core-based ICs. *Proc. Int. Test Conf.*, 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.

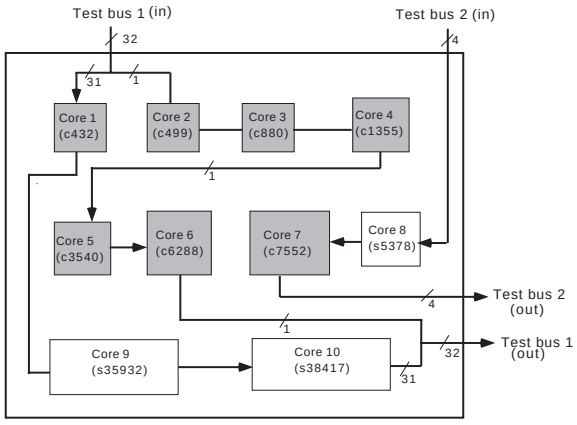

Figure 9. An optimal test bus architecture for $\mathcal{S}_1$ with total bandwidth of 36 bits, and only one subdivision allowed for the first test bus.

[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. Test scheduling for core-based systems. *Proc. Int. Conf. CAD*, pp. 391–394, November 1999.

[5] M. S. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, 1979.

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

[7] I. Hamzaoglu and J. H. Patel. Test set compaction algorithms for combinational circuits. *Proc. Int. Conf. CAD*, pp. 283-289, 1998

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

[9] E. J. Marinissen, R. Arendson, G. Bos, H. Dingemanse, M. Lousberg and C. Wouters. A structured and scalable mechanism for test access to embedded reusable cores. *Proc. Int. Test Conf.*, 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. Int. Test Conf.*, pp. 616–627, 1999.

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

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

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

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