

Test Scheduling Optimization Model for IEEE 1687 Multi-Power Domain Networks Using Boolean Satisfiability

Payam Habiby*

Sebastian Huhn*†

Rolf Drechsler*†

*University of Bremen, Germany
{habiby,huhn,drechsler}@uni-bremen.de

†Cyber-Physical Systems, DFKI GmbH
28359 Bremen, Germany

Abstract—The IEEE 1687 Std. provides an efficient access methodology for embedded instruments in complex system-on-a-chip designs by introducing reconfigurable scan networks. This flexibility enables the reduction of the overall test access time, which significantly decreases the test costs compared to the conventional daisy-chaining method. However, the new access methodology strictly requires new effective test schedulers that consider multi-power domains with individual constraints, as given on the chip-level. This work proposes a novel SAT-based optimization modeling scheme for complex IEEE 1687 networks with multi-power domains, which yields highly effective but power-safe tests.

I. INTRODUCTION

The steadily increasing number of cores in System-on-a-Chip (SoC) designs has been boosting the demand for highly efficient instrument access methodologies since the last years. The IEEE 1687 Std. [1] – also known as IJTAG – was introduced to integrate reconfigurable scan networks into SoCs. Any on-chip module like a built-in self-test engine that holds a client interface and is accessed/managed by an external controller is referred to as an instrument. The faster instruments’ access in IJTAG networks is realized by employing a new component called *Segment Insertion Bit* (SIB), which is programmed to include or exclude the segment of the network that is connected to the host interface on the chip-level. This principle aims at the scan chain’s shortening and, hence, the reduction of the access time overhead during test, debug, and monitoring. In every *Capture-Shift-Update* (CSU) cycle, all instruments of the current active chain receive the new data from *scan-in* port and at the same time shift their individual data to *scan-out* port of the chip. Due to the power, temperature, and further system limitations, not all of the SoC’s components can be simultaneously tested. Further power demand is introduced when the network components are considered in addition to the instruments’ scan and self-test components and, hence, the design is even more prone to issues like IR-drop and test failures [2], [3]. As the overall test time directly affects the test cost, determining an optimized access sequence to the cores massively improves the test procedure and reduces the resulting test costs.

Significant research has been carried out on such an optimization of the test scheduling [4]–[7]. The authors in [5] propose a session-less test scheduling method by sorting the instruments according to their resource conflict and their number of test patterns. In [6], a pre-emptive scheduling

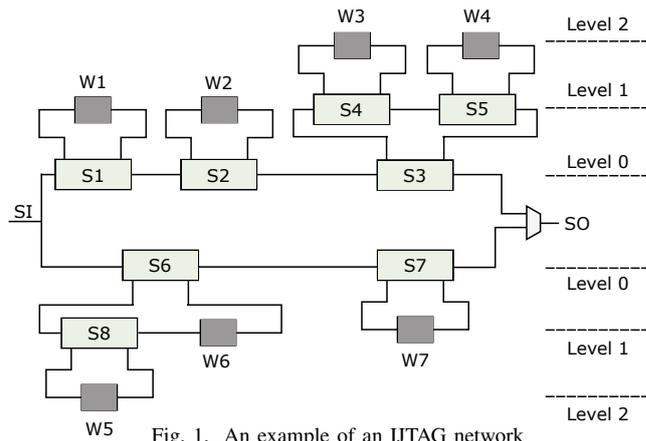


Fig. 1. An example of an IJTAG network

method is introduced, which invokes an exhaustive search over all possible concurrent instruments. The scan architecture of [8] uses a reconfigurable broadcast mechanism to reduce the time required for test data transfer. However, all these works do not consider either any power constraints or do not incorporate multi-power domains, which is strictly required for an effective test of state-of-the-art SoCs. In [7], a graph-based model of IJTAG networks is proposed for the first time, which allows a power-aware test access scheduling in multi-power domain networks. Even if the approach [7] delivers already quite promising results, it typically does not yield an *optimal* solution but remains in local optima. This is since the internal decision-making are, at least, partially done on a greedy-like algorithmic basis.

This work proposes a novel optimization model for the test scheduling in large IJTAG networks with multiple power domains using *Boolean Satisfiability* (SAT). By this, it can be taken advantage of a powerful formal solving engine to, finally, calculate a global optimal test schedule yielding highly effective but power-safe tests.

II. PROPOSED SAT-BASED OPTIMIZATION MODEL

This section presents the proposed model for describing a generic IJTAG network containing *instruments*, *SIBs*, *Scan-Muxes* and branching nodes. This model is then used to obtain an optimal test schedule, which covers all instruments within a minimum overall access time without exceeding the power

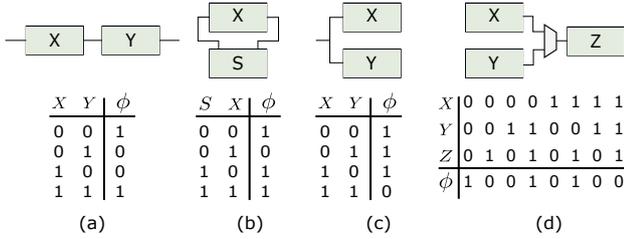


Fig. 2. The structures used in IJTAG network modeling

domains' limits. The IEEE 1687 Std. utilizes an ICL file to describe the network structure. Analogously to the graph approach of [7], this ICL file is used to extract possible concurrent groups of instruments. Fig. 1 shows an IJTAG network embracing eight *SIBs* and seven *instruments*, which are organized in two main branches. Each of these branches can access two levels of hierarchy by opening the related *SIBs*. A *ScanMux* sensitizes one of two branches to create a chain between *SI* and *SO*. If a chain is only composed of the elements of the same hierarchy, this chain is said to be plain. Components of such a chain have exactly the same situation on the network.

The proposed SAT-based optimization scheme involves four steps as follows:

- 1) Four **building blocks** are introduced allowing to create a highly reduced basic network structure using a (pure) Boolean representation.
- 2) The individual power domain limit is reflected by newly introduced **enumeration constraints**, which invalidate the parts of the search space that exceed the considered power limits.
- 3) Two **objective functions** are defined, which guide the optimization procedure, i.e., the minimization of the access latency and the overall power consumption.
- 4) To ensure a full coverage of all reachable instruments, the solving process is invoked in an incremental fashion. The recently covered instruments of run $i-1$, are then masked in the next i -th run.

A. Building Blocks

As stated in Fig. 2(a), the relation between the elements of a plain chain is defined by using bi-implication:

$$\phi = X \leftrightarrow Y = (X \vee \neg Y) \wedge (\neg X \vee Y)$$

Here, a logical value '1' means that the item is on an active chain and is accessible while a value '0' states that the element cannot be accessed. In Fig. 2(b), S represents a *SIB* and X can be either another *SIB* or an instrument. If S is not accessible, X is neither accessible. However, if S is accessible – depending on S being opened or closed – X will be either accessible or inaccessible. This is expressed by

$$\phi = S \vee \neg X$$

Fig. 2(c) shows a branching node and two elements that cannot be placed on a common chain. Considering that only one active chain in an IJTAG network can exist at a time, all states

including more than one chain are invalid and the following equation holds:

$$\phi = \neg X \vee \neg Y \quad (1)$$

Finally, Fig. 2(d) presents the state in which a *ScanMux* selects between two branches of different concurrency groups. The CNF for this structure is $\phi = (\neg X \vee \neg Y) \wedge (\neg Y \vee Z) \wedge (\neg X \vee Z) \wedge (X \vee Y \vee \neg Z)$. However, since the first clause is already covered in branch structure, it is ignored and the equation is reduced to

$$\phi = (\neg Y \vee Z) \wedge (\neg X \vee Z) \wedge (X \vee Y \vee \neg Z) \quad (2)$$

After these four building blocks are gradually applied to generate the network model, the test scheduling problem is modeled accordingly to Fig. 3. The horizontal axis is divided to test pattern units. Each strip is assigned to one test pattern, which is applied to the active chain over a *CSU* and every instrument occupies a number of strips based on its required number of accesses. As long as the access to none of the instruments on the current chain is completed, the path from *SI* to *SO* remains unchanged. w_1 and w_2 in Fig. 3 have this condition over the three first strips. Once the access to w_2 is completed, the new configuration is applied to exclude this instrument from the next scans. The *SIB* programming bits required for the next chain's configuration are appended to the test data of each unit. The gap between the two branches is due to the required *SIB* configuration for opening the first level of hierarchy, including w_5 and w_6 .

B. Enumeration Constraints

The overall power limit $P_{max,j}$ of each power domain j is modeled as one enumeration constraint C_j , which is added to the overall instance. This follows the basic idea that the accumulated power consumption of (active) instruments should not exceed the domain's power limit during a *CSU*:

$$\sum_{w_i \in CSU_n} p_i \leq P_{max} \quad (3)$$

More precisely, a Pseudo-Boolean co-factor – the weight w – is introduced to the Boolean variable v , which reflects the instrument's activation status while w represents the weighted (classified) instrument's power consumption p . Consequently, a valid solution must fulfill all enumeration constraints:

$$\bigwedge_{C_{1..j}} \left(\sum_{i \in \mathcal{I}_j} (w_i * v_i) \leq P_{max,j} \right) \quad (4)$$

where \mathcal{I}_j is the set of reachable instruments of power domain j .

C. Objective Functions

The blocks have to be distributed over the strips such that the minimum number of *CSUs* are required. The optimization criteria are defined as follows: Let $I_i(a_i, p_i, h_i)$ be the i -th instrument with a_i number of accesses and power consumption p_i , which is located in the hierarchical level h_i and scheduled for the n -th *CSU*. If h_n is the maximum accessible hierarchy in the n -th strip, then the constraint for accessing instruments in different hierarchies is $h_{n+1} - h_n \leq 1$. In other words, given $h_2 > h_1$, for accessing an instrument located in hierarchy

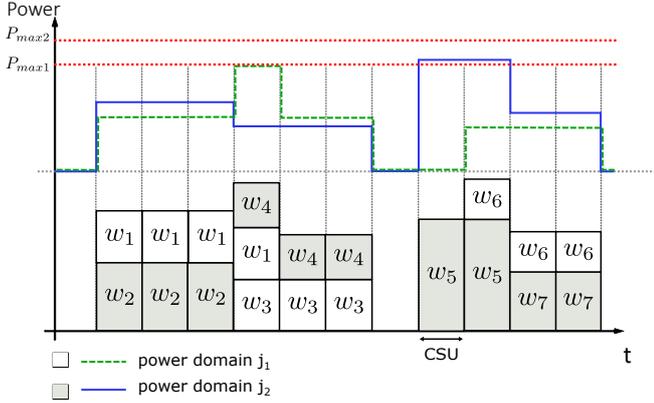


Fig. 3. The scheduling optimization problem

h_2 from the current hierarchy h_1 at least, $h_2 - h_1$ CSUs are required. These criteria yield both a primary Pseudo-Boolean objective function \mathcal{F}_1 and the secondary one \mathcal{F}_2 . \mathcal{F}_1 addresses the minimization of the active hierarchies.

To implement \mathcal{F}_1 , a weight w_i is added to every of the N Boolean variable v_i representing a network component that opens/closes a hierarchical level. The weight is accumulated if the hierarchy is opened, whereby the weight is determined by the actual hierarchical level h_i . \mathcal{F}_1 is defined as follows:

$$\mathcal{F}_1(v_1, \dots, v_N) = \sum_{i=1}^N (w_i \cdot v_i) \quad \text{with} \quad w_i = h_i \quad (5)$$

The second optimization stage (using \mathcal{F}_2) is then applied to reduce the power consumption further, analogously to the enumeration constraints.

D. Incremental Invocation

The blocks of Fig. 3 have to be distributed over the strips such that the minimum number of CSUs is required. This is obtained by allocating the maximum number of instruments within each strip. Since no strip initially exists, 1 is assumed for the SI port's corresponding variable. This yields a path to SO including the maximum number of instruments such that the active instrument does not violate the power limit, as determined by the SAT solver. The set of instruments that have been activated by the SAT solver are considered as the first strip. The next strip is created when the access to at least one of the instruments (of the current set) has been completed. Accordingly, the corresponding literals are assumed as 1 to deactivate the already covered instruments for the solver's next invocation.

III. EVALUATION AND CONCLUSION

For the evaluation of the proposed modeling scheme, a framework has been written in C++, which utilizes the *clasp 3.1.4* Pseudo-Boolean optimization solver [9]. This framework allows for generating a Pseudo-Boolean instance by following the proposed scheme of this work. The networks from the *ITC'16* benchmark set [10] have been considered as a reference.

TABLE I
BENCHMARK IJTAG NETWORKS

benchmark	network	#(nodes,edges)	#instruments	#variables	#clauses
B	Mingle	(27,39)	8	35	62
	BasicSCB	(33,42)	5	39	101
	TreeFlat	(38,61)	11	38	107
C	q12710	(52,78)	23	75	102
	a586710	(94,124)	22	116	303
	t512505	(289,447)	128	417	576
	p22810	(520,789)	242	762	1038
A	N17D3	(52,66)	27	79	144
	N32D6	(79,101)	44	123	216
	N73D14	(155,200)	90	245	426
	N132D4	(293,371)	172	465	825

Table I presents rightmost the number of variables and clauses, which have been obtained while processing the individual ICL network. The benchmark networks have been classified into three groups of complexity, namely basic, classic and advanced. The values – being presented as nodes and edges – represent the total number of elements and their connections in the corresponding graphs. These metrics are meant to provide a general overview of the network scale. The experiments have shown that this model yields Pseudo-Boolean instances, which can be processed in a reasonable run-time.

This paper proposed a novel SAT-based test scheduling optimization model for complex IJTAG test networks with multi-power domains. The proposed model allows determining highly effective but power-safe tests and, by this, contributes to reduced test costs when testing complex and power-critical SoC designs.

IV. ACKNOWLEDGMENT

This work was supported by the AI initiative of the Free Hanseatic City of Bremen. The authors gratefully thank Stephan Eggersglüß from Mentor Graphics - A Siemens Business (Germany) for the inspiring research discussions and thoughtful feedback.

REFERENCES

- [1] "IEEE standard for access and control of instrumentation embedded within a semiconductor device," *IEEE Std. 1687-2014*, pp. 1–283, 2014.
- [2] H. Dhotre, S. Eggersglüß, and R. Drechsler, "Cluster-based localization of ir-drop in test application considering parasitic elements," in *IEEE Latin American Test Symposium*, 2019, pp. 1–4.
- [3] M. H. Tehranipoor and N. Ahmed, *Nanometer technology designs: High-quality delay tests*. Springer Science & Business Media, 2008, vol. 38.
- [4] V. M. V. Mureşan, X. Wang and M. Vlăduţiu, "Greedy tree growing heuristics on block-test scheduling under power constraints," *Journal of Electronic Testing*, vol. 20, pp. 61–78, 2004.
- [5] F. G. Zadehan, U. Ingelsson, G. Asani, G. Carlsson, and E. Larsson, "Test scheduling in an IEEE p1687 environment with resource and power constraints," in *IEEE Asian Test Symposium*, 2011, pp. 525–531.
- [6] S. Keshavarz, A. Nekooei, and Z. Navabi, "Preemptive multi-bit IJTAG testing with reconfigurable infrastructure," in *IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems*, 2014, pp. 293–298.
- [7] P. Habiby, S. Huhn, and R. Drechsler, "Power-aware test scheduling for IEEE 1687 networks with multiple power domains," in *IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems*, 2020, pp. 1–6.
- [8] S. Gupta, J. Wu, and J. Dworak, "Efficient parallel testing: A configurable and scalable broadcast network design using IJTAG," in *IEEE VLSI Test Symposium*, 2018, pp. 1–6.
- [9] M. Gebser, B. Kaufmann, A. Neumann, and T. Schaub, "Conflict-driven answer set solving," in *International Joint Conference on AI*, vol. 7, 2007, pp. 386–392.
- [10] "IEEE Std. 1687 benchmark networks," 2016. [Online]. Available: <https://gitlab.com/IJTAG/benchmarks>