Computing and Informatics, Vol. 27, 2008, 913–930

MEASURING CADET PERFORMANCE BY MEANS OF FITTEST_BENCH06 BENCHMARK CIRCUITS

Josef Strnadel, Tomáš Pečenka, Zdeněk Kotásek

Faculty of Information Technology
Brno University of Technology
Božetěchova 2
612 66 Brno, Czech Republic
e-mail: {`strnadel, pecenka, kotasek`}`@fit.vutbr.cz`

Manuscript received 21 February 2007; revised 13 July 2007
Communicated by Elena Gramatová

Abstract. Benchmark circuits provide a basis for both research institutions and industry to measure their methods and products against. This paper focuses on utilization of recently published FITTest_BENCH06 benchmarks for measuring quality of our novel academic design for testability tool called CADeT. The paper presents basic characteristics of benchmarks and CADeT tool, provides results and analysis of implementing individual testing techniques and their constraint-driven combination to particular benchmarks.

Keywords: Benchmark, design for testability, register-transfer level, test point insertion, scan design technique

1 INTRODUCTION

Over the years, many attempts existed to create and utilize *benchmarks* (note: specially designed circuits for evaluation of new software tools and algorithms) for evaluation and comparison of various tools, algorithms etc. Typically, a *benchmark set* (*suite*) is a set of benchmarks that (in the ideal case) is a representative for a certain type of circuit structures, or the types of structures designed by the particular *electronic design automation* (EDA) tool [1].

The type of description of a benchmark and its level of abstraction depend on the application. For example, the evaluation of high-level synthesis algorithms requires high-level behavioral circuit descriptions, while routing algorithms can only be tested with low-level physical descriptions. Actually, many initiatives dealing with benchmarks exist. For example, benchmark sets for following areas are available in [2]: gate-level test generation (ISCAS8x), high-level synthesis (HLSynth89, HLSynth9x), logic synthesis (LGSynth89, LGSynth9x), physical implementation (LayoutSynth9x, PDWorkshop9x etc.), circuit simulation (CircuitSim90), partitioning (Partitioning93) etc.

For the purposes of verifying our *CADeT* (Combined Automated Design for testability Tool [3]) operating over *register-transfer level* (RTL) digital circuits (note: RTL is a level of circuit description based on data transfer among registers), FITTest_BENCH06 synthetic benchmark set [4, 5] consisting of 31 synthetic RTL benchmarks of various complexities and parameters is utilized. Actually, this is the set of the most complex RTL benchmarks. As the major advantage of synthetic benchmarks [6], the fact they provide full control over important characteristics, such as size, topological, diagnostic or functional parameters of circuits can be seen.

The main goal of verifying CADeT by means of FITTest_BENCH06 benchmarks is to show our CADeT tool is able to find acceptable, highly testable solutions fulfilling user defined design constraints maximally by utilizing *design for testability* (DFT) techniques allowed for that purpose by a user (note: DFT is a design strategy for complex circuits aimed to improve their testing). As a side effect, we hope FITTest_BENCH06 benchmarks with low testability parameters (see Tables 2, 3) will occupy our CADeT tool more than those with better testability properties, so suitability of FITTest_BENCH06 set for benchmarking purposes will be evaluated in the paper too. As it is known to the authors, this paper is the first one presenting results over FITTest_BENCH06 set at all!

The structure of the article is as follows. FITTest_BENCH06 set is briefly presented first. Then, basic properties of our CADeT tool are summarized including information about DFT techniques actually utilized by CADeT, solution state space sizes and principle of evaluating solutions during search process. After that, detailed results of the most important experiments gained during verifying CADeT tool by means of FITTest_BENCH06 benchmarks are presented. Finally, experimental results are summarized into conclusion about CADeT suitability for exploration of design-constrained DFT state space.

2 FITTEST_BENCH06 BENCHMARK SET

During research activities of T. Pečenka et al. [5], benchmark set FITTest_BENCH06 consisting of 31 synthetic sequential circuits was created. The set was geneated by Cirgen [4] tool allowing to generate synthetic circuits reflecting user defined requirements posed on complexity and testability parameters. Fault coverage parameter was used as testability measure for the circuits from the set. Circuits in the set can be divided into the following two classes: variable complexity/variable diagnostic properties (first class, named FITTest_BENCH06(a)) and constant complexity/variable diagnostic properties (second class, named FITTest_BENCH06(b)).

2.1 FITTest_BENCH06(a) Class

This class consists of 20 circuits with 5 levels of complexity – approx. 2 K (2 000), 10 K, 28 K, 150 K and 300 K gates. For each level of complexity, circuits at four levels reflecting diagnostic properties exist (fault coverages related to the levels are approx. 0 %, 33 %, 66 % and 100 %). The complexity of developed circuits after synthesis into 0.35 μm *TSMC* (Taiwan Semiconductor Manufacturing Company) production technology is summarized in Table 1, which shows circuit name, the number of primary inputs/outputs, the number of flip-flops and the number of logic gates.

Circuit name	**# of PIs**	**# of POs**	**# of FFs**	**# of gates**	**in-circuit modules**
e01	86	80	160	1 985	5x 16-bit adders (ADDs), 5x 16-bit subtractors (SUBs), 5x 16-bit multiplexers (MUXs), 10x 16-bit registers (REGs)
e02	86	80	144	1 657	
e03	86	80	160	2 046	
e04	86	80	160	2 221	
e05	186	160	792	10 011	25x 16-bit ADDs, 25x 16-bit SUBs, 25x 16-bit MUXs, 50x 16-bit REGs
e06	186	160	831	9 999	
e07	186	160	785	9 894	
e08	186	160	778	9 559	
e09	211	192	2 020	28 065	25x 8-bit + 25x 16-bit ADDs, 25x 8-bit + 25x 16-bit SUBs, 50x 8/16-bit MUXs, 25x 8/16 MULs, 125x 8/16-bit REGs
e10	179	208	1 979	27 853	
e11	211	200	2 058	28 231	
e12	203	208	2 106	28 438	
e13	1 669	1 904	6 304	155 046	50x 16-bit + 50x 32-bit ADDs, 25x 16-bit + 25x 32-bit SUBs, 50x 16/32-bit MUXs,50x 16/32-bit MULs, 250x 16/32-bit REGs
e14	1 621	1 904	6 368	155 380	
e15	1 701	1 840	6 368	155 207	
e16	1 589	1 744	6 368	155 045	
e17	3 833	4 272	12 672	310 122	100x 16-bit+100x 32-bit ADDs, 100x 16-bit+100x 32-bit SUBs, 100x 16/32-bit MUXs, 100x16/32-bit MULs, 500x16/32-bit REGs
e18	3 913	4 512	12 608	309 856	
e19	3 833	4 320	12 576	309 874	
e20	3 961	4 352	12 736	310 610	

Table 1. Complexity of FITTest_BENCH06(a) benchmarks

Results gained for FITTest_BENCH06(a) circuits are summarized in Table 2, which shows circuit name, the number of faults, fault coverage of distinguishable faults [7] and the number of test cycles.

Circuit name	# of faults	Fault coverage	# of cycles
e01	6 830	90.45 %	235
e02	5 716	60.69 %	206
e03	7 114	39.43 %	141
e04	7 900	0.00 %	23
e05	33 146	90.11 %	1 123
e06	33 028	43.90 %	4 402
e07	33 338	22.87 %	1 797
e08	31 358	0.00 %	31
e09	93 556	91.90 %	4 748
e10	93 132	64.22 %	33 514
e11	94 488	27.46 %	15 009
e12	93 948	0.00 %	390
e13	572 722	89.38 %	13 878
e14	572 890	64.46 %	35 509
e15	572 430	31.84 %	16 895
e16	572 025	12.50 %	11 532
e17	1 146 522	81.73 %	56 227
e18	1 147 050	56.72 %	55 902
e19	1 146 978	40.28 %	39 920
e20	1 148 130	23.13 %	20 518

Table 2. Diagnostic properties of FITTest_BENCH06(a) benchmarks

2.2 FITTest_BENCH06(b) Class

This class consists of 11 circuits with equal complexity but with various diagnostic properties (fault coverage is in the range from 0 % to 100 %), details are in Table 3.

In this section, the set of FITTest_BENCH06 benchmark circuits was described. In the following section, it will be demonstrated how the set will be used to verify the effectiveness of CADeT tool used for the design of testable circuits.

3 CADET TOOL

CADeT is an academic tool being developed by members of Diagnostics Research Group at the Faculty of Information Technology, Brno University of Technology (FIT BUT). The main goal of the tool is to experimentally verify diagnostic methods being developed by the group [8].

Before CADeT takes an RTL netlist containing information about original circuit structure as an input, the following actions are to be done by a user:

Circuit name	**# of PIs/POs**	**# of gates**	**# of FFs**	**# of faults**	**Fault coverage**	**in-circuit modules**
a00	820/850	108 627	4 384	399 806	1.28 %	35x 16-bit ADDs 35x 16-bit SUBs 35x 16-bit MUXs 35x 32-bit ADDs 35x 32-bit SUBs 35x 16/32-bit MULs 175x 16-bit REGs 175x 32-bit REGs
a01	839/880	108 748	4 448	399 786	10.11 %	
a02	775/816	108 532	4 416	398 012	23.81 %	
a03	775/912	108 876	4 448	400 166	31.60 %	
a04	791/912	108 551	4 416	399 334	40.38 %	
a05	775/848	108 740	4 448	399 534	50.19 %	
a06	775/896	108 607	4 416	399 494	65.56 %	
a07	743/848	108 811	4 448	399 708	66.37 %	
a08	839/992	108 650	4 448	399 566	74.86 %	
a09	743/944	108 345	4 384	398 888	86.54 %	
a10	775/832	108 652	4 448	399 112	94.29 %	

Table 3. Details related to FITTest_BENCH06(b) circuits

- selection of DFT techniques for testability enhancement of the original circuit,
- definition of design constraints in the form of area and pin overheads to be paid at the worst for the enhancement,
- setup of search parameters.

Then, CADeT explores state space of all possible solutions and tries to find solutions with an acceptable cost/quality trade-off between achieved testability enhancement and its price reflecting user defined design constraints. During the exploration, CADeT stores both

- a log-file containing information about all solutions identified during the exploration, and
- a log-file containing information about evolution of best solutions found during the exploration.

After exploration is finished, CADeT stores the best solution recognized, i.e. original circuit structure with inserted DFT structures.

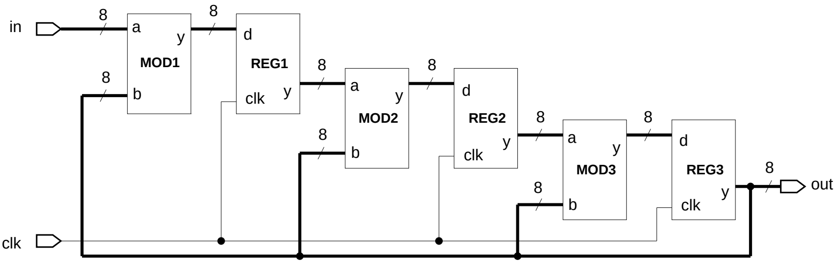


Fig. 1. Experimental circuit (NL) containing 3 modules and 3 registers in its structure

3.1 Bounds of the State Space Size

During our previous research activities, we have analyzed the size of the state space of solutions modified by means of *scan* [7] – Figure 2 illustrates application of this popular low-cost DFT technique.

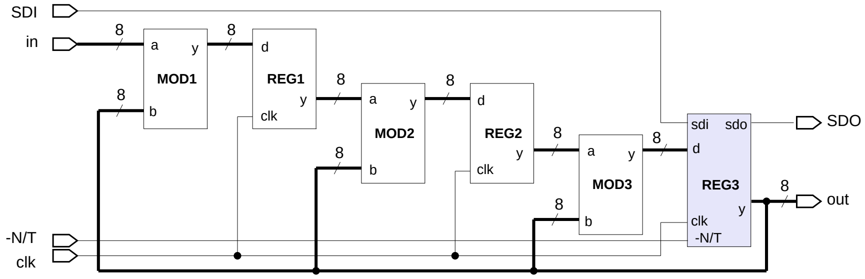


Fig. 2. NL circuit with REG3 modified to scan

We have concluded that the size is a function of the Lah [9] number (if ordering of registers within scan chains is important) or the Bell [9] number (if ordering of registers within scan chains is not important). Note: the Bell number is the number of ways a set of n elements can be partitioned into nonempty subsets, and the Lah number is the number of ways a set of n elements can be partitioned into $1 \leq k \leq n$ nonempty subsets in such a way that all elements of the same subset are linearly ordered; n is equal to the total number of in-circuit registers.

In both cases, the size grows combinatorially with the number of registers within the circuit structure, but it holds that the Bell number << the Lah number for given number of registers. Much research was done (e.g., [10, 11, 12, 13]) demonstrating that dealing with placement of registers into multiple scan chains and with their ordering make sense. Dealing only with selection of registers into scan, but ommiting their final ordering within particular scan chains, state space can be reduced substantially. The final ordering will be searched lastly, after selection is finished. The other way how to approach the problem is to solve both selection and ordering concurrently, utilizing information about circuit testability to reduce the set of in-circuit registers for scan purposes. This approach is utilized in CADeT.

The second DFT technique CADeT utilizes is a *test point insertion* technique (note: the technique is based on inserting extra logic into a circuit structure, see, e.g., [7, 14]). Application of the technique is illustrated in Figure 3. In CADeT, only the insertion of multiplexers enhancing controllability of in-circuit nodes is implemented. The upper bound of the state space size of all possible solutions created by means of this technique is equal to 2^n, where n is the number of in-circuit ports.

If the combination of DFT techniques is utilized, then the size of merged state space is equal to the product of state space sizes of particular techniques.

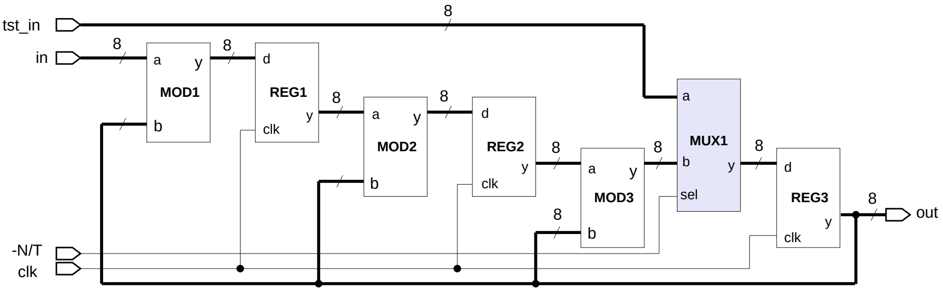


Fig. 3. NL circuit with multiplexer inserted between MOD3 and REG3

3.2 Evaluating Solutions

Evaluation of solutions gained during state space exploration by CADeT is based on the evaluation of the price of cost/quality trade-off achieved by application of DFT techniques in the particular solution. The higher the price of particular solution is, the lower is the fitness value of the solution and vice versa. The principle of fitness-evaluating formula is presented in the following text. Before the formula will be presented, few symbols need to be introduced first. Let us suppose

- maximal area overhead and pin overhead a user is willing to pay for testability enhancement of the original circuit structure are denoted by ao_{constr} and po_{constr} parameters, and
- ao_{act}, po_{act} and tst_{act} denote area overhead, pin overhead (both in %/100) and testability (in $< 0.0; 1.0 >$ interval) values of particular solution from the state space. Testability parameters are evaluated by means of the method published in [15]. Suitability of utilizing the evaluation is shown in [16], where close correlation between testability value gained by our *ADFT* (Automated DFT) tool and fault coverage parameter gained by commercial ATPG tools is shown.

During the exploration of the solution state space, fitness value fit_{act} is assigned to particular solution according to the following formula:

$$fit_{act} = \frac{tst_{act}}{(1 + |ao_{act} - ao_{constr}|) \times (1 + |po_{act} - po_{constr}|)} \tag{1}$$

According to the values assigned to ao_{constr} and po_{constr} parameters, the following special cases of applying formula 1 can be distinguished:

- $(ao_{constr} > 0)$ and $(po_{constr} > 0)$: Solutions having high testability value and ao_{act} and po_{act} parameters close to ao_{constr} and po_{constr} constraints are assigned high fit_{act} value. CADeT is looking for the best testable solution satisfying ao_{constr} and po_{constr} constraints maximally;

- ($ao_{constr} = 0$) and ($po_{constr} = 0$): Solutions having high testability value and minimal values of ao_{act} and po_{act} parameters are assigned high fit_{act} value. CADeT is looking for the best testable solution at minimal DFT costs;
- constraints are placed on only one of ao_{constr}, po_{constr} parameters and the other is minimized by setting it to 0;
- ($ao_{constr} \to \infty$) or ($po_{constr} \to \infty$): In this case, ao_{constr} or po_{constr} parameter is ignored – i.e. area or pin overhead is not important for a user.

In all cases, the solution CADeT is looking for is the solution with the highest fitness value found during the search process. The following steps are repeated until the solution is not found during a given number of iterations during the process:

1. [*DFT implementation*] new solution with built-in DFT techniques is generated
   a) [*Selection of DFT modification*] particular configuration based on user-selected DFT techniques is generated randomly (special mutation of previously gained solutions is utilized for the purpose)
   b) [*Implementation of DFT modification*] configuration from point 1 a) is built into the original circuit structure
2. [*Evaluation*] fit_{act} value is assigned to the implemented solution
3. [*Detection*] if $fit_{act} > fit_{best}$, the (so far) best solution is found
4. [*DFT removal*] DFT techniques are removed from the circuit.

In this section, the principles utilized by the CADeT tool were described. They were implemented and the software was experimentally verified on FITTest_BENCH06 benchmarks. The results are discussed and summarized in the following section.

4 EXPERIMENTAL RESULTS

The main goal of the results is to show that CADeT is able to find acceptable, highly testable solutions fulfilling user-given design constraints maximally by utilizing DFT techniques allowed for that purpose by a user. As a partial goal, we want to check the following statement: FITTest_BENCH06 benchmarks with low testability require more computational time of CADeT than those with high testability (Tables 2, 3).

In the following subsections, the results of selected experiments are presented and summarized in sequence according to the above-mentioned goals.

4.1 Fitness Evaluation Quality

The objective of the first experiment was to check whether solutions with lower ao_{act}, po_{act} values and higher tst_{act} value are assigned higher fit_{act} value than solutions with higher values of overhead parameters and lower value of testability parameter. Because conclusions derived from the results are the same for all FITTest_BENCH06 benchmarks, only results for *a00* benchmark are presented in the following text.

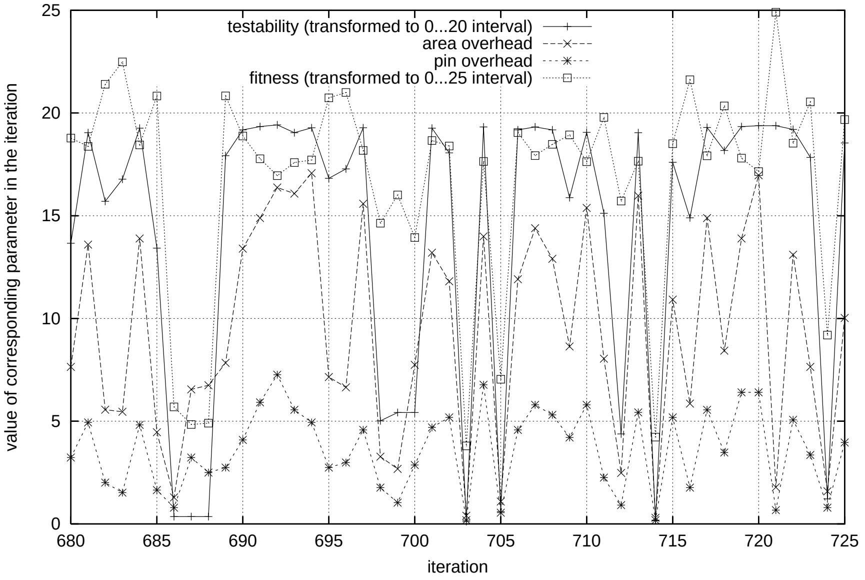


Fig. 4. Search process (cutout) for a00 benchmark

In Figure 4, it can be seen that solutions evaluated by higher fit_{act} values are those with good cost/quality trade-off between costs of DFT application and testability enhancement achieved by the application. As an example of such solutions, see values the for iterations 683, 716 or 721. As an "opposite" example, see the values for iterations 687, 688 or 703.

Also, it can be seen there is a great diversity of quality (measured by means of fit_{act} values) of solutions generated during the search process. It means that CADeT is able to search through solutions differing from each other by appied DFT technique significantly. This fact is important for CADeT's ability to find solutions with high cost/quality trade-off.

4.2 Impact of FITTest_BENCH06 Benchmarks to Search Process

In this subsection, results of experiments dealing with the complexity of exploring state spaces for particular FITTest_BENCH06 benchmarks are presented. The main objective of these experiments was to check whether FITTest_BENCH06 benchmarks presented as difficult to be tested, complex circuits will increase CADeT's computational time more than the less complex ones.

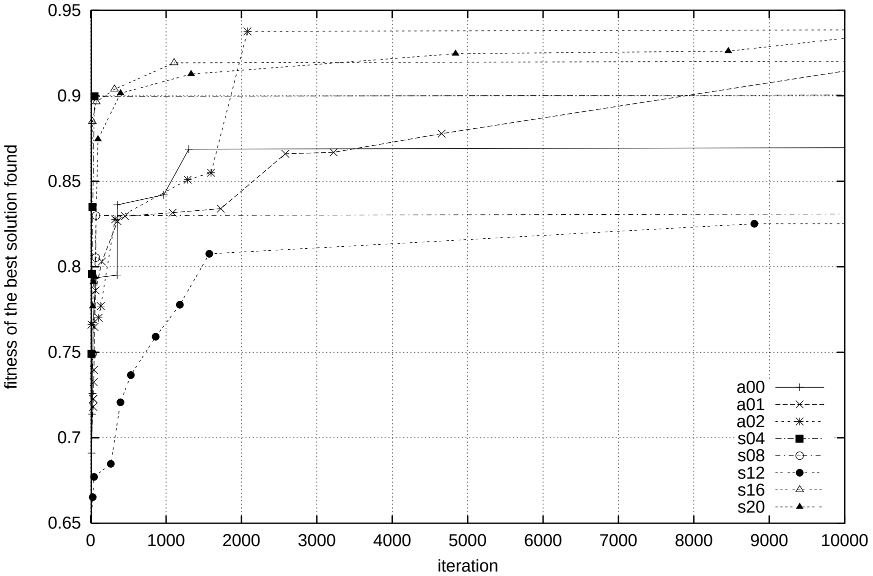


Fig. 5. Evolution of best merged solutions

In Figure 5, evolution of best merged solutions (i.e. those with more than one DFT technique implemented) found during the first 10^4 iterations of the search-process is presented for selected benchmarks. It can be seen the time CADeT needs to process particular benchmark from FITTest_BENCH06 set is proportional to the complexity of the benchmark.

As expected, analysis of the most complex *s20* benchmark circuit was most time-consuming. The processing of *s20* benchmark by CADeT took cca 0.8 s, which can be seen as short time for circuits of such complexity level (Figure 6). On the basis of the figure, the following conclusions can be accepted:

- for all tested benchmarks, CADeT is able to find "average" solutions very quickly because fitness values of solutions found during the first cca 2 K iterations grow quickly. But, it takes much more iterations to find optimum-close solutions;
- growth gradient of fitness values of best merged solutions found during search process is proportional to testability and complexity values of original benchmark structures. This fact can be seen as the proof of suitability of FITTest_BENCH06 set for benchmarking purposes.

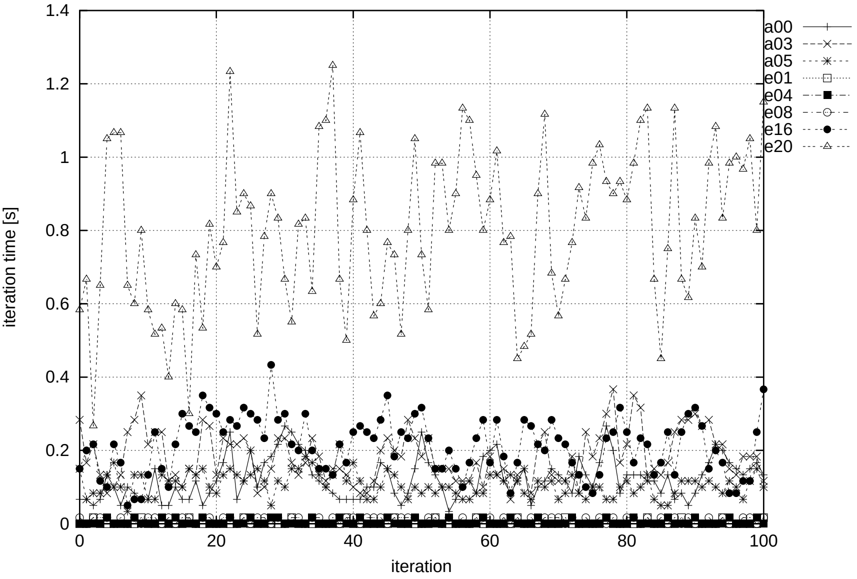


Fig. 6. Sample (100 iterations) of the search process

4.3 Results of Exploring Solution Space

In this subsection, acceptable-quality (but not the best!) results found during searching state spaces for various DFT techniques allowed by a user are presented. The reason why the best results are not presented is as follows. The state space sizes for processed benchmarks and DFT techniques are so comprehensive that it is impossible to search through complete state spaces. Thus, CADeT is searching for an acceptable solution with the presumption it will be as close to the best solution as possible under the conditions given for the search process. Results are presented for benchmarks from FITTest_BENCH06 set with low testability parameters (*a00–a05* and *e04*, *a08*, *a12*, *a16*, *a20* benchmarks).

4.3.1 Scan Technique Results

Below, the results of exploring state space for scan DFT technique are presented. To receive a result for particular benchmark, it took 2 hours in the average. Figure 7 presents testability values of solutions with an aceptable cost/quality trade-off ratio between testability value and testability costs of the solutions found by CADeT for selected benchmarks. In the figure, it can be seen that testability values for most of presented benchmarks are very close to 1, which corresponds (in terms of testability analysis utilized) to solutions with high testability parameters. Because the costs needed for achieving such testability parameters are (in the average) about 1.5 %

in terms of area and pin overheads, it can be said that the costs are low, which means CADeT is able to find high-quality solutions based on application of scan DFT technique.

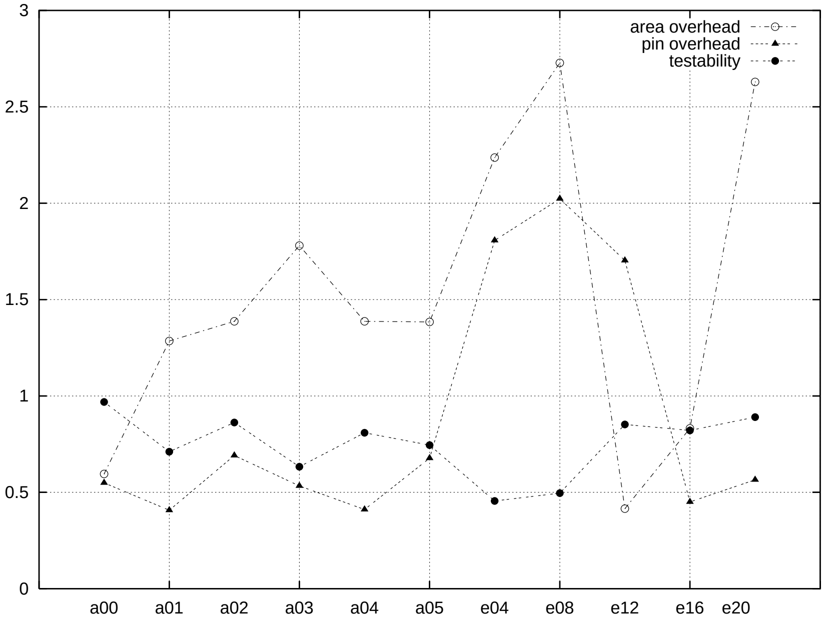


Fig. 7. Testability values and application costs of acceptable scan solutions

In Table 4, more detailed results of acceptable quality scan solutions found and their scan application costs are presented. For each benchmark, the following data is stored in its row: the number of registers in original benchmark structure, the suggested number and portion of registers selected for scan by CADeT in the solution, the suggested number of scan chains for including scan registers, the costs of the solution in terms of area and pin overheads and finally, the portion of controllable (observable) nodes for the solution as a result of scan application suggested by CADeT.

4.3.2 Test Point Insertion Technique

In the following text, the results of exploring state space for test point insertion DFT technique are presented – see Figure 8 for the visualization. It took 1 hour in the average to find a result for particular benchmark. In the figure, it can be seen that testability values of solutions gained for presented benchmarks are lower than in the case of previously presented scan DFT technique. Testability values presented in the figure correspond to solutions with good testability parameters. The costs of the solutions are worse than in the case of scan technique solutions, because test point

Benchmark							
	Original structure	**Solution (acceptable-quality found DFT-modification)**					
Name	**total # of registers**	**# (%) of scan registers**	**# of scan chains**	**area over. [%]**	**pin over. [%]**	**con. nodes [%]**	**obs. nodes [%]**
a00	172	6 (3.5)	4	0.595	0.545	100.0	98.3
a01	174	23 (13.2)	5	2.275	0.64	96.9	97.6
a02	173	14 (8.0)	5	1.387	0.691	95.5	97.6
a03	174	11 (6.3)	6	1.087	0.77	100.0	98.4
a04	173	14 (8.1)	3	1.387	0.411	98.8	98.6
a05	174	14 (8.0)	5	1.385	0.678	93.4	97.0
e04	10	1 (10)	1	2.2	1.8	100.0	98.1
e08	48	6 (12.5)	3	2.7	2.0	100.0	98.1
e12	125	3 (2.4)	3	0.4	1.7	100.0	97.2
e16	247	12 (4.9)	7	0.8	0.4	100.0	98.9
e20	498	61 (12.2)	23	2.1	0.5	100.0	98.4

Table 4. Selected solutions with high testability/low cost parameters identified in scan state space

insertion technique is based on inserting extra logic (multiplexers in our case) into benchmark structures, which is more expensive than including original registers into scan chains. More detailed results of achieved solutions are presented in Table 5.

4.3.3 Results of Combining Scan and Test Point Insertion Techniques

Lastly, the results of exploring state space for combination of above-mentioned DFT techniques are presented in Figure 9 and Table 6. To receive a result for particular benchmark was more complicated process; it required 2.5 hours of computational time in the average. In the figure, it can be seen that quality and cost of merged solutions costs (in the average) more than scan solutions and less than test point insertion solutions. This result is in contrast with our original assumption, which expected that solutions consisting of proper combination of more DFT techniques will produce better cost/quality trade-off than the solutions gained through one DFT technique only.

In our opinion, such a "bad" result can be probably caused by the following facts:

- state space size of merged solutions is of higher order than state space sizes for scan or test point insertion solutions, so it takes much more time to find merged solutions of similar or better quality (supposing they exist at all) compared to pure scan or test point insertion solutions,
- in FITTest_BENCH06 benchmarks, there is a high number of registers densely covering all feedback loops. It means that pure scan solutions can be of better

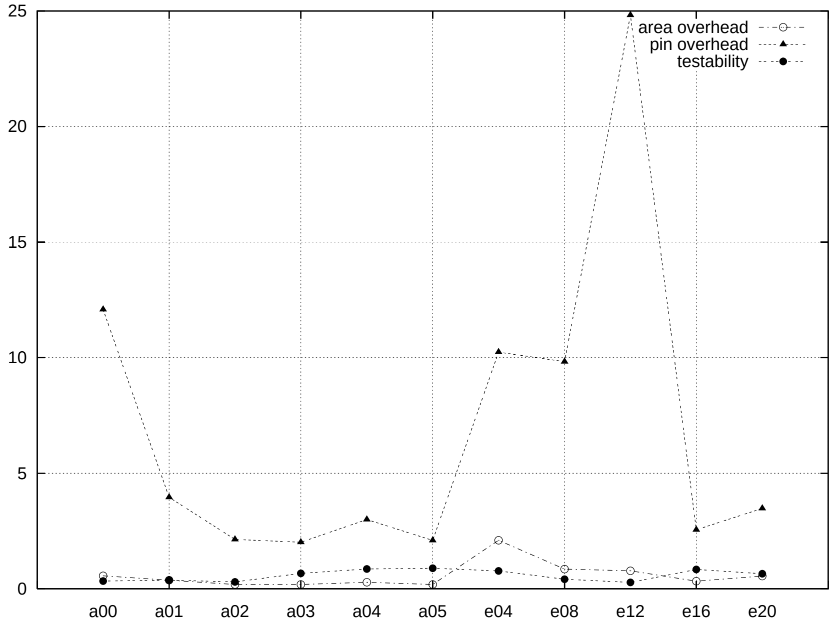


Fig. 8. Testability values and application costs of acceptable test point insertion solutions

Benchmark						
	Original structure	**Solution (acceptable-quality found DFT-modification)**				
Name	**total # of nodes**	**# (%) of nodes enhanced by multiplexers**	**area over. [%]**	**pin over. [%]**	**con. nodes [%]**	**obs. nodes [%]**
a00	1149	6 (0.52)	0.6	12.0	72.5	74.1
a01	1153	4 (0.34)	0.4	4.0	76.2	80.4
a02	1151	2 (0.17)	0.2	2.1	62.3	69.5
a03	1153	2 (0.17)	0.2	2.0	93.0	93.8
a04	1151	3 (0.26)	0.3	3.0	99.7	98.8
a05	1153	2 (0.17)	0.2	2.0	99.7	98.8
e04	75	1 (1.3)	2.0	10.2	100.0	98.4
e08	341	2 (0.58)	0.9	9.8	79.5	90.1
e12	805	6 (0.75)	0.8	24.8	72.9	72.7
e16	1694	5 (0.29)	0.3	2.6	99.6	98.9
e20	3456	16 (0.46)	0.6	3.5	87.4	89.3

Table 5. Selected solutions with high testability/low cost parameters identified in test point insertion state space

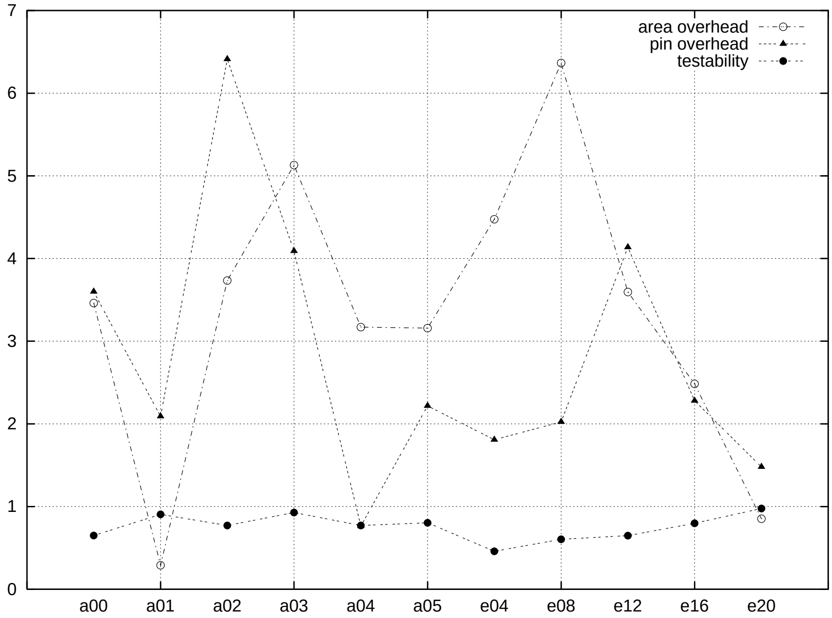


Fig. 9. Testability properties and application costs of acceptable merged solutions

Benchmark					
	Solution (acceptable-quality found DFT-modification)				
Name	**# (%) of nodes/registers enhanced by/modified to multiplexers/scan**	**area over. [%]**	**pin over. [%]**	**con. nodes [%]**	**obs. nodes [%]**
a00	2 (0.17) / 33 (19.18)	3.5	3.6	97.2	98.0
a01	1 (0.09) / 2 (1.15)	0.29	2.09	97.7	97.2
a02	5 (0.43) / 33 (19.07)	3.7	6.4	94.9	96.7
a03	2 (0.17) / 50 (28.74)	5.1	4.0	100.0	98.7
a04	0 (0) / 32 (18.5)	3.1	0.7	96.7	97.5
a05	1 (0.087) / 31 (17.81)	3.1	2.2	94.8	96.1
e04	0 (0) / 2 (20)	4.4	1.8	100.0	98.0
e08	0 (0) / 14 (29.1)	6.3	2.0	97.7	94.5
e12	0 (0) / 26 (20.8)	3.6	4.1	99.5	97.4
e16	3 (0.18) / 33 (13.36)	2.5	2.3	97.3	97.8
e20	6 (0.17) / 19 (3.82)	0.9	1.5	99.8	98.6

Table 6. Selected solutions with high testability/low cost parameters identified in merged state space

cost/quality trade-off between testability achieved and its price than merged solutions, which costs more because of inserting extra multiplexers into benchmark structures,

- CADeT parameters working properly when searching in one-DFT solution state spaces need not to be the best for searching merged solutions.

The above-mentioned features are good stimuli for our future long-term research activities, which will be primarily focused, but not limited to, the following topics:

- acceleration of merged state space search process by means of bringing a knowledge into the search algorithm (application of general legality after its discovery, heuristic rules etc. for state space restriction),
- merged search applied to circuits containing less registers in their structures than are available in FITTest_BENCH06 structures (however, no such circuits of complexity similar to FITTest_BENCH06 are available at present),
- analyzing impact of CADeT parameters to the search process, especially of probability constants related to mutation operation.

5 CONCLUSIONS

The paper has focused on utilization of FITTest_BENCH06 benchmarks for measuring quality of the CADeT tool. In the paper, basic characteristics of the benchmarks were presented as well as experimental results of implementing selected DFT techniques and their combination by CADeT to particular benchmarks.

From experimental results presented at the end of the paper, it can be seen that CADeT is able to find acceptable, highly testable solutions fulfilling user-given design constraints maximally by utilizing DFT techniques allowed for that purpose by a user. However, at the end of experimental section it was concluded that there is still a lot of further research needed to enhance the search process, especially when combination of DFT techniques is allowed by a user. Another activity can be seen in comparing results presented in the paper with results gained by other methods. However, no such results exist at present.

On the basis of the above-mentioned experimental results, it can be stated that FITTest_BENCH06 benchmarks with low testability parameters required longer computational time than those with better testability properties. Because of rarity of optimum-close solutions in DFT state space for particular benchmarks, it was difficult for CADeT to find the solutions and thus FITTest_BENCH06 set has been found as suitable for benchmarking of various diagnostic tools and algorithms.

While new design methodologies are supported by modern design tools, our activities have focused on non-hierarchical digital RTL designs so far. By means of the method implemented in CADeT, our further research will be dedicated especially to testability analysis and DFT of hierarchical and *system-on-a-chip* (SOC) digital and mixed-signal designs, which belong to the most popular approaches at present.

Also, it is planned to extend CADeT towards other DFT techniques and constraints posed on power consumption and test application time of solutions from DFT state spaces. Afterwards, the results gained by CADeT can be seriously compared with those gained by commercial tools for automated DFT design.

Acknowledgement

The work related to the paper has been financially supported by the Grant Agency of the Czech Republic (GACR) under post-doctoral contract number GP102/05/P193 "Optimizing Methods in Digital Systems Diagnosis" and by the Research Plan No. MSM 0021630528 – Security-Oriented Research in Information Technology.

REFERENCES

[1] Harlow, J. E.: Overview of Popular Benchmark Sets. IEEE Design and Test of Computers, Vol. 17, No. 3, 2000, pp. 15–17.

[2] The Benchmark Archives at CBL [2007]. Available on: `http://www.cbl.ncsu.edu/benchmarks/`.

[3] Strnadel, J.: Educational Tools for Digital Circuit Diagnosis. 2006. Available at `http://www.fit.vutbr.cz/~strnadel/diag/`.

[4] Pečenka, T.: FITTest_BENCHxx Benchmarks & Cirgen, 2005. Available on: `http://www.fit.vutbr.cz/~pecenka/cirgen/`.

[5] Pečenka, T.—Kotásek, Z.—Sekanina, L.: FITTest_BENCH06: A New Set of Benchmark Circuits Reflecting Testability Properties. In: Proceedings of 2006 IEEE Design and Diagnostics of Electronic Circuits and Systems Workshop, Prague, IEEE Computer Society, 2006, pp. 285–289.

[6] Verplaetse, P.—Campenhout, J. V.—Stroobandt, D.: On Synthetic Benchmark Generation Methods. In: Proc. of IEEE International Symposium on Circuits and Systems, Vol. IV, Geneva, IEEE Circuits & Systems Society, 2000, pp. 213–216.

[7] Bushnell, M. L.—Agrawal, V. D.: Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits. Kluwer, Boston, 2000, 690 pp.

[8] Diagnostics Research Group at FIT BUT, 2000. Available on: `http://www.fit.vutbr.cz/research/groups/diag/index.php.en`.

[9] Convay, J. H.—Guy, R. K.: The Book of Numbers. Springer-Verlag, New York, 1996, 320 p.

[10] Ghosh, S.—Basu, S.—Touba, N. A.: Joint Minimization of Power and Area in Scan Testing by Scan Cell Reordering. In: Proceedings of the IEEE Computer Society Annual Symposium on VLSI, IEEE Computer Society, 2003, pp. 246–249.

[11] Gupta, P.—Kahng, A. B.—Mantik, S.: Routing-Aware Scan Chain Ordering. In: ACM Transact. on Design Automation of Electronic Systems, 2005, pp. 546–560.

[12] Hsu, L. C.—Chen, H. M.: On Optimizing Scan Testing Power and Routing Cost in Scan Chain Design. In: 7th International Symposium on Quality Electronic Design, 2006, pp. 451–456.

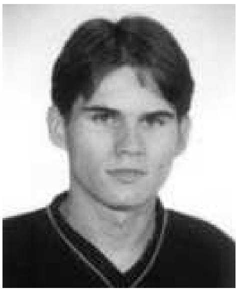
[13] Reddy, S. M.—Miyase, K.—Kajihara, S.—Pomeranz, I.: On Test Data Volume Reduction for Multiple Scan Chain Designs. In: ACM Transactions on Design Automation of Electronic Systems, 2003, pp. 460–469.

[14] Geuzebroek, J.: Test Point Insertion to Improve BIST Performance, and to Reduce ATPG Test Time & Data Volume. Delft University Press, 2003, 215 pp.

[15] Strnadel, J.: Testability Analysis and Improvements of Register-Transfer Level Digital Circuits. Computing and Informatics, Vol. 25, 2006, No. 5, ISSN 1335-9150.

[16] Pečenka, T.—Strnadel, J.—Kotásek, Z.—Sekanina, L.: Testability Estimation Based on Controllability and Observability Parameters. In: Proceedings of the 9th EUROMICRO Conference on Digital System Design, Cavtat, IEEE Computer Society, 2006, pp. 504–514.

Josef Strnadel received the M. Sc. degree in computer science and electrical engineering at the Faculty of Electrical Engineering and Computer Science (FEE), Brno University of Technology (BUT), Czech Republic, in 2000 and the Ph. D. degree in information technology at the Faculty of Information Technology (FIT) of BUT, in 2004. Now he works as an Assistant Professor at FIT BUT. His main research interests are related to embedded and real-time systems and to testability of digital systems. IEEE member (2004–).



Tomáš Pečenka received the M. Sc. degree in computer engineering at the FIT BUT, Czech Republic. Actually, he is a Ph. D. student at the Department of Computer Systems (DCSY) at FIT BUT. His main research interests are related to design and testability analysis of digital circuits and to automated generation of synthetic benchmark circuits.

Zdeněk Kotásek received his M. Sc. and Ph. D. degrees (in 1969 and 1991) from BUT, both in computer science. Between 1969 and 2001, he worked at the Department of Computer Science of FEE BUT, since 2002 at the DCSY, FIT BUT. He is Associate Professor at FIT BUT since 2000 and the head of the DCSY (2005–). His research interests include digital circuit diagnostics and testing, testability analysis and design and synthesis for testability. He is an IEEE member (2003–).