

Contents

1	Introduction	1
Part I Preliminaries and Previous Work		
2	Circuits and Testing	11
2.1	Post-Production Test	11
2.2	Circuits	14
2.2.1	Scan-Based Testing	15
2.3	Fault Models	16
2.3.1	Stuck-at	17
2.3.2	Delay	20
2.4	Classical ATPG Algorithms	25
2.4.1	ATPG for Stuck-at Faults	25
2.4.2	ATPG for Delay Faults	32
2.5	Industrial Test Environment	36
3	Boolean Satisfiability	41
3.1	Boolean Algebra	41
3.2	SAT Solver	42
3.2.1	DLL-Algorithm	43
3.3	Advanced SAT Techniques	44
3.3.1	Fast Boolean Constraint Propagation	45
3.3.2	Conflict Analysis	46
3.3.3	Conflict-Driven Heuristics	49
3.3.4	Incremental SAT	51
3.3.5	Restarts	52
3.4	Circuit-to-CNF Transformation	52
3.5	Circuit-Oriented SAT and Observability Don't Cares	54
3.5.1	Exploitation of Observability Don't Cares	56

4	ATPG Based on Boolean Satisfiability	59
4.1	SAT-Based ATPG for Boolean Circuits	60
4.1.1	SAT Formulation: Stuck-at Fault Model	60
4.1.2	SAT-Based ATPG Techniques	63
4.2	SAT-Based ATPG for Industrial Circuits	66
4.2.1	Multiple-Valued Logic	66
4.2.2	Hybrid Logic	68
4.2.3	Improving Compactness	68
4.3	Combination with Structural Algorithm	69
 Part II New SAT Techniques and their Application in ATPG		
5	Dynamic Clause Activation	73
5.1	Overall Framework for Dynamic Clause Activation	74
5.2	Efficient Activation Methodology	77
5.2.1	Consistent SAT Instance	78
5.2.2	Structural Watch List	79
5.3	Literal-Based Activation	82
5.4	Implicit Observability Don't Cares	84
5.4.1	Clause-Based J-Frontier	85
5.4.2	SCOAP-Based Decision Heuristic	88
5.5	Classical SAT Solver Emulation	89
5.6	SAT Formulation for Test Generation Using DCA	91
5.6.1	Dynamic Clause Activation and Multiple-Valued Logic	93
5.7	Experimental Results	94
5.7.1	Results for the Stuck-at Fault Model	95
5.7.2	Results for the Path Delay Fault Model	98
5.7.3	Results for Industrial Circuits	100
5.8	Summary	105
6	Circuit-Based Dynamic Learning	107
6.1	Integration of Dynamic Learning	108
6.1.1	Pervasive Conflict Clause Identification	108
6.1.2	Variable-Based Activation	110
6.1.3	Combination with Dynamic Clause Activation	111
6.2	Post-Classification Phase	112
6.3	Learning Strategies	113
6.4	Improved SAT Solving Engine	114
6.5	Experimental Results	115
6.5.1	Dynamic Learning for Stuck-at Faults	116
6.5.2	Dynamic Learning for Path Delay Faults	118
6.5.3	Dynamic Learning for Industrial Circuits	119
6.5.4	Combination of Structural and SAT-Based Algorithms	122

6.6	Summary	123
Part III High Quality Delay Test Generation		
7	High Quality ATPG for Transition Faults	127
7.1	Transition Fault Model: SAT Formulation	128
7.1.1	Iterative Logic Array	128
7.1.2	Injection of Stuck-at Faults	129
7.1.3	Experimental Results	131
7.2	Long Propagation Paths	134
7.2.1	Incremental Instance Generation	135
7.2.2	Output Ordering	136
7.2.3	Experimental Results	137
7.3	Timing-Aware ATPG	140
7.3.1	Motivational Example	140
7.3.2	Pseudo-Boolean Optimization	142
7.3.3	PBO Formulation: Timing-Aware ATPG	143
7.3.4	Using Structural Information	149
7.3.5	Considering Transition-Dependent Delays	150
7.3.6	Experimental Results	151
7.4	Summary	152
8	Path Delay Fault Model	155
8.1	Related Work	156
8.1.1	Robust Tests in Combinational Circuits	157
8.1.2	Incremental SAT and Learning	157
8.1.3	PDF Test Generation Using CSAT	158
8.2	Non-Robust Test Pattern Generation	158
8.3	Robust Test Pattern Generation	160
8.4	SAT Instance Generation Flow	162
8.5	As-Robust-As-Possible Test Generation	163
8.5.1	Test Generation for ARAP Tests	166
8.5.2	Incremental SAT Formulation for Static Value Justification	168
8.5.3	Considering the Presence of Small Delay Defects	170
8.6	Experimental Results	172
8.6.1	Comparison with Competitive Approach	173
8.6.2	SAT Instance Generation Flow	173
8.6.3	MONSOON Using DynamicSAT	175
8.6.4	ARAP Test Generation	177
8.7	Summary	179
9	Summary and Outlook	181
References		183
Index		191

High Quality Test Pattern Generation and Boolean
Satisfiability

Eggersgluß, S.; Drechsler, R.

2012, XVIII, 193 p., Hardcover

ISBN: 978-1-4419-9975-7