

Contents

Abstract	iii
Contents	ix
1 Introduction	1
1.1 Motivation	1
1.2 Objectives	6
1.3 Structure of the thesis	7
2 Dependability-aware Hardware Design Flow	9
2.1 Semicustom and FPGA-based design flow	10
2.1.1 Model-based design	10
2.1.2 SRAM-based FPGA as target implementation technology	13
2.1.3 Technology-specific libraries	15
2.2 Dependability assessment	21
2.3 Dependability benchmarking	26
2.4 Dependability-aware design space exploration	27
2.5 Conclusions	31

3	Fault Injection for Dependability Assessment of HW Designs	33
3.1	Introduction	34
3.2	Fault models	37
3.3	Simulation-based fault injection	40
3.3.1	SBFI techniques	40
3.3.2	Insufficiency of RT-level fault injection	42
3.3.3	Performance and accuracy challenges of implementation-level SBFI	44
3.3.4	SBFI tools	47
3.4	FPGA-based fault injection	48
3.4.1	FFI techniques	49
3.4.2	Locating the fault targets in the FPGA configuration memory	52
3.4.3	FFI tools	57
3.5	Existing strategies for improving fault injection performance	58
3.5.1	Optimizing the fault space through fault collapsing	58
3.5.2	Statistical fault injection	59
3.5.3	Speeding-up fault injection runs	62
3.6	Conclusions	64
4	Enabling Low-intrusive Simulation-based Fault Injection for Implementation-level Models	67
4.1	Introduction	68
4.2	Targeting VITAL-compliant models	69
4.2.1	Definition of generic operations to support fault injection	69
4.2.2	Stuck-at, pulse, and indetermination faults	71
4.2.3	Bit-flip faults in registers	72
4.2.4	Delay faults	73
4.2.5	Considering FPGA-specific components: bit-flips in configuration memory of LUTs	75
4.3	Targeting Verilog-based models	76
4.3.1	Bit-flip faults	77
4.3.2	Delay faults	79
4.4	Unified fault dictionary	80
4.5	Conclusions	83

5	Improving the Accuracy of FPGA-based Fault Injection	85
5.1	Introduction	86
5.2	Towards bit-accurate mapping of macrocells onto the configuration memory	87
5.2.1	Mapping of Look-Up tables	88
5.2.2	Mapping of Block RAMs	96
5.3	Optimized essential bits.	100
5.4	Exploiting optimized essential bits for the bit-accurate emulation of SEUs	104
5.5	Conclusions.	107
6	Contributions in Improvement of Fault Injection Performance	109
6.1	Introduction	110
6.2	Strategies to reduce the number of injection runs.	111
6.2.1	Filtering and prioritization of essential bits through the profiling of the target switching activity	111
6.2.2	Iterative statistical fault injection.	117
6.3	Strategies to speed-up the fault injection runs.	121
6.3.1	Mixed-level and multi-level fault injection	122
6.3.2	Simulation-based and FPGA-based checkpointing.	127
6.4	Discussion.	132
6.5	Conclusions.	133
7	Contributions in Dependability-aware Design Space Exploration	135
7.1	Introduction	136
7.2	DSE based on the design of experiments	138
7.2.1	Background on design of experiments and its statistical analysis.	138
7.2.2	Exploring regular design spaces by means of fractional factorial designs	140
7.2.3	Exploring irregular design spaces through iterative refinement of D-optimal designs.	144
7.3	Speeding-up the GA-based DSE by means of iterative selection	148
7.4	Conclusions.	153
8	DAVOS Toolkit	155
8.1	Introduction	155

- 8.2 DAVOS architecture 156
- 8.3 Fault injection tools for dependability assessment 160
 - 8.3.1 DAVOS-SBFI tool 161
 - 8.3.2 DAVOS-FFI tool 164
 - 8.3.3 Interactive reporting interface 168
- 8.4 Automated PPAD evaluation of parametrized designs. 168
 - 8.4.1 Implementation support tool 170
 - 8.4.2 PPAD evaluation engine 172
- 8.5 Decision support tool for selecting and optimizing HW designs 174
- 8.6 Conclusions 177

- 9 Experimental Evaluation 179
 - 9.1 Introduction 180
 - 9.2 Dependability benchmarking of soft-core processors 181
 - 9.2.1 Experimental procedure 181
 - 9.2.2 Fault injection results and dependability metrics 185
 - 9.2.3 Ranking of DUTs 191
 - 9.2.4 Experimental effort and speed-up. 193
 - 9.2.5 Discussion 200
 - 9.3 Dependability-aware design space exploration for optimal tuning of EDA parameters. 202
 - 9.3.1 Experimental procedure 203
 - 9.3.2 DSE results obtained by GA-based approach. 204
 - 9.3.3 DSE results obtained by DoE-based approach 208
 - 9.3.4 Discussion 215
 - 9.4 Dependability assessment and verification of fault-tolerant HW design 217
 - 9.4.1 Experimental procedure 217
 - 9.4.2 Experimental results. 220
 - 9.4.3 Discussion 222
 - 9.5 Conclusions 223

- 10 Conclusions and Future Work 227
 - 10.1 Conclusions 227

10.2 Summary of contributions an publications.	233
10.2.1 Contributions of the thesis	233
10.2.2 Publications	235
10.2.3 Research projects	236
10.3 International research stay	236
10.4 Future work.	237
Appendices	239
A Details of Bit-accurate FPGA-based Fault Injection Approach	241
A.1 Accessing the configuration memory of Xilinx FPGAs	241
A.2 Bit-accurate mapping of LUTs onto the configuration memory	245
A.3 Determining the state of unused LUT pins.	248
A.4 Extracting the macrocells descriptors from implementation-level netlist	251
B Case Study Details	253
B.1 Architecture of the DUTs	253
B.2 Convergence of GA/NSGA-based DSE	254
B.3 Regression models for PPAD attributes.	256
B.4 Comparison of experimentally obtained PPAD optimization results with the predicted ones.	259
Bibliography	261