
Table of Contents

1	Introduction	1
2	Deep-Submicron CMOS Fault Models	5
2.1	Introduction	5
2.2	Transient Faults	7
2.3	Permanent Faults	9
2.4	Conclusions	12
3	FPGA-based Fault Injection	13
3.1	Introduction	13
3.2	Basic FPGA architecture and design flow	17
3.3	Fault emulation	21
3.3.1	Compile-time reconfiguration	22
3.3.2	Run-time reconfiguration	26
3.3.3	Discussion	28
3.4	Generic FPGA architecture	31
3.5	Conclusions	38
4	New Approaches for Transient and Permanent Faults Emulation	39
4.1	Introduction	39
4.2	Emulation of <i>bit-flips</i>	41
4.2.1	Injecting <i>bit-flips</i> into memory blocks	42
4.2.2	Injecting <i>bit-flips</i> into FFs	43
4.2.2.1	Using the Global Set Reset (GSRin) line	43
4.2.2.2	Using the Local Set Reset (LSRin) line	45
4.2.2.3	Discussion	48
4.2.3	Summary	49
4.3	Emulation of <i>stuck-ats</i>	50
4.3.1	Injecting <i>stuck-ats</i> into combinational logic	50
4.3.2	Injecting <i>stuck-ats</i> into sequential logic	53

4.3.2.1	Using the unused Local Set/Reset (LSRin) line	53
4.3.2.2	Using the unused LUT associated to the FF . . .	55
4.3.2.3	Using the clock signal (CLKin) of the FF . . .	57
4.3.2.4	Discussion	58
4.3.3	Summary	59
4.4	Emulation of pulses	59
4.4.1	Injecting <i>pulses</i> into combinational logic implemented as a LUT	61
4.4.2	Injecting <i>pulses</i> by using inverter multiplexers	62
4.4.3	Discussion	63
4.4.4	Summary	64
4.5	Emulation of stuck-opens	64
4.5.1	Injecting <i>stuck-opens</i> into combinational logic	65
4.5.2	Injecting <i>stuck-opens</i> into sequential logic	67
4.5.2.1	Using the unused Local Set/Reset (LSRin) line	67
4.5.2.2	Using the unused LUT associated to the FF . .	68
4.5.2.3	Using the clock signal (CLKin) of the FF . . .	70
4.5.2.4	Discussion	71
4.5.3	Summary	72
4.6	Emulation of indeterminations	72
4.6.1	Injecting <i>indeterminations</i> into combinational logic . . .	73
4.6.2	Injecting <i>indeterminations</i> into sequential logic	75
4.6.2.1	Using the unused Local Set/Reset (LSRin) line	75
4.6.2.2	Using the unused LUT associated to the FF . .	76
4.6.2.3	Using the clock signal (CLKin) of the FF . . .	79
4.6.2.4	Discussion	81
4.6.3	Summary	82
4.7	Emulation of delays	82
4.7.1	Increasing the delay of a line by adding pass transistors	84
4.7.1.1	Increasing the line's length	84
4.7.1.2	Increasing the line's fan-out	86
4.7.1.3	Discussion	87
4.7.2	Increasing the delay of a line by adding a LUT	88
4.7.3	Increasing the delay of a line by adding a FF	91
4.7.4	Discussion	94
4.7.5	Summary	95
4.8	Emulation of shorts	95
4.9	Emulation of open-lines	98
4.10	Emulation of bridgings	100
4.11	Conclusions	102

5	FADES: a Tool Supporting FPGA-Based Fault Injection	107
5.1	Introduction	107
5.2	Architecture of FADES	109
5.3	Experiments definition	113
5.3.1	Model implementation and basic information	113
5.3.2	Injection into sequential logic	115
5.3.3	Injection into combinational logic	117
5.3.4	Monitoring	118
5.3.5	Summary	120
5.4	Experiment execution	120
5.4.1	Initialisation	120
5.4.2	Workload execution	122
5.4.3	Observation points	123
5.4.4	Fault injection	124
5.4.5	Fault deletion	127
5.4.6	Reset to initial state	128
5.5	Analysis of results	128
5.6	Conclusions	130
6	Experimental Validation and Case Study	133
6.1	Introduction	133
6.2	Correctness of experiments' results	134
6.2.1	Experimental set-up	134
6.2.2	Analysis of results	135
6.3	Speeding-up experiments' execution	138
6.4	Methodology and tool features	141
6.4.1	Execution time and scalability	141
6.4.2	Spatial intrusion	145
6.4.3	Portability	145
6.4.4	Independence from model description	146
6.4.5	Controllability and accessibility	146
6.4.6	Observability	147
6.5	Case Study	148
6.5.1	Cores under study	149
6.5.2	Workload	150
6.5.3	Faultload	150
6.5.4	Measures	151
6.5.5	Analysis of results	152
6.5.5.1	Impact of transient faults on the sequential logic of the system	154

6.5.5.2	Impact of permanent faults on the sequential logic of the system	155
6.5.5.3	Impact of transient faults on the combinational logic of the system	156
6.5.5.4	Impact of permanent faults on the combinational logic of the system	157
6.5.5.5	Impact of faults targeting the combinational logic of the system versus faults targeting its sequential logic	158
6.5.5.6	Impact of transient versus permanent faults . .	159
6.5.5.7	Summary	160
6.6	Conclusions	161
7	Conclusions	165
A	Detailed description of files and commands used by FADES	169
A.1	Bitstream file	169
A.2	Logic Allocation file	171
A.3	User Constraints file	172
A.4	Workload file	173
A.5	Readback command	173
A.6	Trace file	174
A.7	Results file	175
	Bibliography	176