

Contents

Samenvatting	1
Summary	5
Preface	9
1 An introduction to fault-tolerant computing	13
1.1 Introduction	13
1.2 Designing reliable systems	14
1.2.1 The various stages in the life of a system	14
1.2.2 Fault classification	15
1.2.3 Reliability criteria	16
1.3 Methods to improve the reliability of computer systems	18
1.3.1 Fault avoidance	18
1.3.2 Fault tolerance	19
1.4 Systems based on fault masking versus dynamic redundant systems	29
1.5 The Input Problem	32
1.6 Conclusion	37
2 Generalized Masking Redundancy	41
2.1 Introduction	41
2.2 The (N, K) -concept	43
2.3 On modeling the behaviour of fault-tolerant systems	48
2.3.1 The combinatorial model	48
2.3.2 The Moore model	49
2.3.3 Unfolding time into space	51
2.3.4 The meaning of behaviour	52

2.3.5	System decomposition	54
2.3.6	Specification, design and implementation	55
2.3.7	Correct and malicious behaviour	56
2.4	An abstract view on N modular redundancy	60
2.4.1	An N -modular redundant implementation of a combinatorial system	61
2.4.2	An N -modular redundant implementation of a sequential system without state voting	63
2.4.3	An N -modular redundant implementation of a sequential system with state voting	68
2.5	A generalization of masking redundancy	73
2.5.1	Introduction	73
2.5.2	$(\mathcal{X}, \mathcal{Y}, T)$ Fault-tolerant systems	77
2.5.3	Examples of $(\mathcal{X}, \mathcal{Y}, T)$ Fault-tolerant systems	78
2.5.4	$(\mathcal{X}, \mathcal{Y}, T)$ fault-tolerant systems based on authentication	84
2.5.5	$(\mathcal{X}, \mathcal{Y}, \mathcal{Z}, T)$ Fault-tolerant systems	85
2.5.6	Examples of $(\mathcal{X}, \mathcal{Y}, \mathcal{Z}, T)$ fault-tolerant systems	88
2.6	The $(4, 2)$ -concept	94
2.6.1	System description	94
2.6.2	Data transfer between processor and memory	95
2.6.3	Applicable symbol-error-correcting codes for the $(4, 2)$ -concept	96
2.7	Symbol- and bit-error-correcting codes	99
2.8	Decoding symbol- and bit-error-correcting codes	102
2.9	Decoder implementation	107
2.10	Some facts about the implementation of the $(4, 2)$ -concept	113
3	A class of algorithms for reaching interactive consistency based on voting and coding	115
3.1	Introduction to the Byzantine Generals Algorithms	116
3.1.1	The definition of the Byzantine Generals problem	116
3.1.2	The parameters relevant for interactive consistency algorithms	117
3.1.3	Results published	120
3.2	Introduction to the algorithms and their proof	124
3.2.1	A survey of the algorithms considered	124
3.2.2	The way in which the algorithms are described	125
3.3	The Dispersed Joined Communication Algorithms	128

3.3.1	Introduction	128
3.3.2	The construction of the Dispersed Joined Communication Algorithms	132
3.3.3	The existence of Dispersed Joined Communication Algorithms in the classes $\mathcal{A}(T, K, a, \mathbf{D}, \mathbf{Ns})$	139
3.3.4	Some behavioural properties of the Dispersed Joined Communication algorithms in the presence of at most T modules which behave maliciously	143
3.4	A class of algorithms for reaching interactive consistency based on voting and coding	147
3.5	Some remarks on the construction of Interactive Consistency Algorithms which are based on voting and coding	149
3.5.1	The general construction of Interactive Consistency Algorithms which are based on voting and coding	150
3.5.2	Two simple examples	155
3.5.3	The Minimal Voting algorithms and the Maximal Coding algorithms	162
3.5.4	The Subset Method	164
4	A comparison of the existing algorithms and the algorithms based on voting and coding	167
4.1	Introduction	167
4.2	The algorithms selected for comparison	168
4.3	The criteria	169
4.3.1	Introduction to the criteria	169
4.3.2	The number of messages in the algorithm based on voting and coding	169
4.3.3	The number of messages in the Subset Method	171
4.3.4	The minimum size of the original message in the source	175
4.3.5	The number of messages in the Dolev-algorithm	176
4.4	The algorithms compared	177
5	Interconnecting fault-tolerant systems	185
5.1	Introduction	185
5.2	Communication of a fault-tolerant system with its environment	187
5.2.1	The DJC Method applied to a single input device	189
5.2.2	The DJC Method applied to a fault-tolerant input device with post-observation	191

5.2.3	The DJC Method applied to a fault-tolerant input device with pre-observation	195
5.2.4	The DJC Method applied to an NMR input device with pre-coding and pre-observation	199
5.3	Some examples of the interconnection of fault-tolerant systems	203
5.3.1	An (N, K) -concept fault-tolerant system interconnected with external sources	203
5.3.2	Some simple examples of the interconnection of fault-tolerant systems	204
5.3.3	The architecture of a $(4, 2)$ module	208
5.3.4	Some concluding remarks	210
6	Conclusions	211
	Curriculum vitae	223