Contents

INTRODUCTION xiii

CHAPTER 1. FROM THE SYSTEM TO THE SOFTWARE 1

1.1. Introduction. 1
1.2. Command/control system 2
1.3. System. 6
1.4. Software application 8
1.4.1. What is software?. 8
1.4.2. Different types of software 9
1.4.3. The software application in its proper context 10
1.5. Conclusion 11

CHAPTER 2. RAILWAY STANDARDS 13

2.1. Introduction. 13
2.2. Generic standards 14
2.2.1. Introduction 14
2.2.2. Safety levels 15
2.3. History between CENELEC and the IEC 16
2.4. CENELEC referential framework. 17
2.4.1. Introduction 17
2.4.2. Description. 18
2.4.3. Implementation 21
2.4.4. Software safety 22
2.4.5. Safety versus availability 22
2.5. EN 50155 standard 23
2.6. CENELEC 50128 26
2.6.1. Introduction 26

2.6.2. SSIL management 26
2.6.3. Comparison of 2001 and 2011 versions. 28
2.7. Conclusion 30

CHAPTER 3. RISK AND SAFETY INTEGRITY LEVEL 31

3.1. Introduction. 31
3.2. Basic definitions 31
3.3. Safety enforcement 37
3.3.1. What is safety? 37
3.3.2. Safety management. 40
3.3.3. Safety integrity 47
3.3.4. Determination of the SIL 50
3.3.5. SIL table 55
3.3.6. Allocation of SILs 56
3.3.7. SIL management 57
3.3.8. Software SIL. 58
3.3.9. Iterative process. 59
3.3.10. Identification of safety requirements. 60
3.4. In IEC 61508 and IEC 61511 61
3.4.1. Risk graph 62
3.4.2. LOPA. 64
3.4.3. Overview. 66
3.5. Conclusion 66

CHAPTER 4. SOFTWARE ASSURANCE 67

4.1. Introduction. 67
4.2. Prerequisites 67
4.3. Quality assurance 68
4.3.1. Introduction 68
4.3.2. Quality assurance management. 69
4.3.3. Realization of a software application 73
4.3.4. Software quality assurance plan (SQAP) 75
4.4. Organization 78
4.4.1. Typical organization 78
4.4.2. Skill management. 80
4.5. Configuration management 82
4.6. Safety assurance management. 84
4.7. Verification and validation. 86
4.7.1. Introduction 86
4.7.2. Verification 87
4.7.3. Validation 103

4.8. Independent assessment 104
4.9. Tool qualification 104
4.10. Conclusion 105
4.11. Appendix A: list of quality documents to be produced 106
4.12. Appendix B: structure of a software quality assurance plan 106

CHAPTER 5. REQUIREMENTS MANAGEMENT 109

5.1. Introduction 109
5.2. Requirements acquisition phase 110
5.2.1. Introduction 110
5.2.2. Requirements elicitation 111
5.2.3. Process of analysis and documentation 119
5.2.4. Verification and validation of the requirements 126
5.3. Requirements specification 129
5.3.1. Requirements characterization 129
5.3.2. Characterization of requirements specification 135
5.3.3. Expression of requirements 135
5.3.4. Requirements validation 140
5.4. Requirements realization 140
5.4.1. Process 140
5.4.2. Verification 141
5.4.3. Traceability 143
5.4.4. Change management 146
5.5. Requirements management 150
5.5.1. Activities 150
5.5.2. Two approaches 151
5.5.3. Implementation of tools 152
5.6. Conclusion 154

CHAPTER 6. DATA PREPARATION 155

6.1. Introduction 155
6.2. Recap 156
6.3. Issue 156
6.4. Data-parameter-based system 158
6.4.1. Introduction 158
6.4.2. Characterization of data 161

6.4.3. Service inhibition 162
6.4.4. Overview 164
6.5. From the system to the software 165
6.5.1. Need 165
6.5.2. What the CENELEC framework does not say 167
6.6. Data preparation process 169
6.6.1. Context 169
6.6.2. Presentation of section 8 of the CENELEC 50128:2011 standard 170
6.7. Data preparation process 174
6.7.1. Management of the data preparation process 174
6.7.2. Verification 182
6.7.3. Specification phase 182
6.7.4. Architecture phase 186
6.7.5. Data production 190
6.7.6. Integration of the application and acceptance of the tests 196
6.7.7. Validation and evaluation of the application 197
6.7.8. Procedure and tools for preparation of the application 197
6.7.9. Development of generic software 198
6.8. Conclusion 199
6.9. Appendix: documentation to be produced 199

CHAPTER 7. GENERIC APPLICATION 201

7.1. Introduction 201
7.2. Software application realization process 201
7.3. Realization of a generic application 203
7.3.1. Specification phase 203
7.3.2. Architecture and component design phase 213
7.3.3. Component design phase 236
7.3.4. Coding phase 242
7.3.5. Execution of component tests 243
7.3.6. Software integration phase 246
7.3.7. Overall software testing phase 247
7.4. Some feedback on past experience 249
7.5. Conclusion 250
7.6. Appendix A: the programming language “Ada” 251
7.7. Appendix B: the programming language “C” 253
7.7.1. Introduction 253

7.7.2. The difficulty with C 253
7.7.3. MISRA-C 254
7.7.4. Example of a rule 255
7.8. Appendix C: introduction to object-oriented languages 255
7.9. Appendix D: documentation needing to be produced. 258

CHAPTER 8. MODELING AND FORMALIZATION 261

8.1. Introduction. 261
8.2. Modeling 261
8.2.1. Objectives 261
8.2.2. Different types of modeling 263
8.2.3. Model 264
8.3. Use of formal techniques and formal methods 265
8.3.1. Definitions 265
8.3.2. UML 268
8.4. Brief introduction to formal methods. 269
8.4.1. Recap 269
8.4.2. Usage in the railway domain 270
8.4.3. Summary 276
8.5. Implementation of formal methods 279
8.5.1. Conventional processes 279
8.5.2. Process including formal methods 280
8.5.3. Issues 282
8.6. Maintenance of the software application. 284
8.7. Conclusion 285

CHAPTER 9. TOOL QUALIFICATION 287

9.1. Introduction. 287
9.2. Concept of qualification 288
9.2.1. Issue. 288
9.2.2. CENELEC 50128:2001 288
9.2.3. DO-178 291
9.2.4. IEC 61508 292
9.2.5. ISO 26262 293
9.3. CENELEC 50128:2011. 293
9.3.1. Introduction 293
9.3.2. Qualification file 294

9.3.3. Qualification process 295
9.3.4. Implementation of the qualification process 297
9.4. Fitness for purpose 305
9.4.1. Design method 305
9.4.2. In case of incompatibility 305
9.4.3. Code generation 306
9.5. Version management 306
9.5.1. Identification of versions 306
9.5.2. Bug/defect analysis 307
9.5.3. Changing versions 307
9.6. Qualification process 307
9.6.1. Qualification file 307
9.6.2. Ultimately 308
9.6.3. Qualification of non-commercial tools 308
9.7. Conclusion 308

CHAPTER 10. MAINTENANCE AND DEPLOYMENT 309

10.1. Introduction 309
10.2. Requirements 309
10.2.1. Fault management 309
10.2.2. Managing changes 310
10.3. Deployment 312
10.3.1. Issue 312
10.3.2. Implementation 313
10.3.3. In reality 314
10.4. Software maintenance 315
10.4.1. Issue 315
10.4.2. Implementation 315
10.5. Product line 316
10.6. Conclusion 318
10.7. Appendix: documentation needing to be produced 319

CHAPTER 11. ASSESSMENT AND CERTIFICATION 321

11.1. Introduction 321
11.2. Evaluation 321
11.2.1. Principles 321
11.2.2. CENELEC 50128:2011 324
11.3. Cross-acceptance 325
11.4. Certification 326
11.4.1. Product certification 326

11.4.2. Software certification 327
11.4.3. Evolution management 327
11.5. Conclusion 328
11.6. Appendix: documentation needing to be produced 328

CONCLUSION 329

BIBLIOGRAPHY 331

GLOSSARY 343

INDEX 351