

차례

| | | |
|----------|------------------------------------|------------|
| 1 | 일반적인 기초지식 | 15 |
| 1.1 | 실시간 | 15 |
| 1.2 | 단계적인 개발 모델 — V-모델 | 15 |
| 1.3 | 빌드 과정 - 모델에서 실행 파일로 | 18 |
| 1.4 | 요약 | 34 |
| 2 | 마이크로프로세서 기술 기초지식 | 36 |
| 2.1 | 마이크로프로세서 설계 | 36 |
| 2.2 | 코드 실행 | 39 |
| 2.3 | 메모리 주소 지정과 주소 지정 모드 | 43 |
| 2.4 | 대기 상태와 버스트 접근 | 52 |
| 2.5 | 캐시 | 53 |
| 2.6 | 파이프라인 | 60 |
| 2.7 | 인터럽트 | 62 |
| 2.8 | 트랩/예외 | 63 |
| 2.9 | 데이터 일관성 | 64 |
| 2.10 | 데스크톱 프로세서와 임베디드 프로세서의 비교 | 67 |
| 2.11 | 요약 | 69 |
| 3 | 운영체제 | 70 |
| 3.1 | 운영체제 미사용 — 무한루프 더하기 인터럽트 | 71 |
| 3.2 | OSEK/VDX | 75 |
| 3.3 | 협력형 멀티태스킹과 선점형 멀티태스킹 | 83 |
| 3.4 | POSIX | 94 |
| 3.5 | 요약 | 102 |
| 4 | 타이밍 이론 | 103 |
| 4.1 | 타이밍 파라미터 | 104 |

| | | |
|----------|---|------------|
| 4.2 | 통계적인 측면 | 113 |
| 4.3 | CPU 부하 | 118 |
| 4.4 | 버스 부하 | 128 |
| 4.5 | 논리적인 수행 시간(LET) | 128 |
| 4.6 | 요약 | 130 |
| 5 | 타이밍 분석 기법 | 132 |
| 5.1 | 개괄, 계층적인 시각 | 132 |
| 5.2 | 용어 정의 | 137 |
| 5.3 | 정적 코드 분석 | 139 |
| 5.4 | 코드 시뮬레이션 | 153 |
| 5.5 | 시간 측정 | 165 |
| 5.6 | 하드웨어 기반 트레이싱 | 183 |
| 5.7 | 계측 기반 트레이싱 | 205 |
| 5.8 | 스케줄링 시뮬레이션 | 231 |
| 5.9 | 정적 스케줄링 분석 | 243 |
| 5.10 | 진화 알고리즘을 사용한 최적화 | 258 |
| 5.11 | V 모델에서 타이밍 분석 기법 | 261 |
| 6 | 타이밍 문제의 실제 예제 | 264 |
| 6.1 | 이 모든 인터럽트는 어디서 올까? | 264 |
| 6.2 | OSEK ECC — 최고의 선택은 아니다 | 266 |
| 6.3 | 리셋 후 17분에 발생하는 드문 충돌 | 270 |
| 6.4 | 빠지거나 중복된 센서 데이터 | 273 |
| 6.5 | 핸드 브레이크를 당긴 상태에서 경주 | 281 |
| 6.6 | 측정이 정적 코드 분석에서 나온 결과보다 더 큰 WCT를 내 놓다 | 282 |
| 6.7 | 네트워크 관리 메시지가 곧 나타난다 | 284 |
| 6.8 | 대량 양산 프로젝트에서 아주 매끄러운 타이밍 프로세스 | 286 |
| 6.9 | 타이밍 분석이 1,200만 유로를 절약하다 | 287 |

| | |
|---|------------|
| 6.10 요약 | 289 |
| 7 멀티코어, 매니코어, 그리고 멀티 ECU 타이밍 | 290 |
| 7.1 멀티코어 기초 | 290 |
| 7.2 병렬 수행의 다양한 유형 | 297 |
| 7.3 데이터 일관성, 스핀락 | 309 |
| 7.4 메모리 주소 복제 | 317 |
| 7.5 요약 | 321 |
| 8 타이밍 최적화 | 323 |
| 8.1 스케줄링 수준에서 타이밍 최적화 | 323 |
| 8.2 메모리 사용의 타이밍 최적화 | 330 |
| 8.3 코드 수준에서 타이밍 최적화 | 338 |
| 8.4 타이밍 최적화를 위한 요약과 지침 | 367 |
| 9 소프트웨어 개발 과정에서 사용하는 방법론 | 371 |
| 9.1 타이밍과 관련한 요구 사항 | 371 |
| 9.2 개발 과정에서 협업 | 383 |
| 9.3 타이밍 컨셉, 스케줄링 배치, 그리고 운영체제 구성 | 385 |
| 9.4 타이밍 디버깅 | 386 |
| 9.5 타이밍 최적화 | 387 |
| 9.6 타이밍 검증 | 387 |
| 9.7 향후 기능을 위한 초기 고려 사항 | 391 |
| 9.8 최종 제품에서 타이밍 관리 | 393 |
| 9.9 긍정적인 사례 - 비테스코 테크놀로지스의 CoReMa | 395 |
| 9.10 요약 | 397 |
| 10 AUTOSAR | 399 |
| 10.1 AUTOSAR 전통적인 플랫폼(CP) | 401 |
| 10.2 AUTOSAR 적응 플랫폼(AP) | 403 |
| 10.3 TIMEX(AUTOSAR 타이밍 확장) | 417 |

| | |
|--|------------|
| 10.4 ARTI (AUTOSAR/ASAM 실행 시간 인터페이스) | 421 |
| 10.5 기술 보고서 "타이밍 분석" | 426 |
| 10.6 요약 | 427 |
| 11 안전과 ISO 26262 | 428 |
| 11.1 기초 | 428 |
| 11.2 안전 표준, 타이밍, 그리고 타이밍 검증 | 431 |
| 11.3 타이밍 검증을 위한 도구 | 435 |
| 11.4 법적인 측면 | 435 |
| 11.5 요약 | 436 |
| 12 전망 | 437 |