1. Record Nr. UNINA9910138962803321

| | |
|---|---|
| Autore | Rochange Christine |
| Titolo | Time-predictable architectures / / Christine Rochange, Sascha Uhrig, Pascal Sainrat |
| Pubbl/distr/stampa | London, England ; ; Hoboken, New Jersey : , : ISTE Ltd : , : John Wiley & Sons, , 2014<br>©2014 |
| ISBN | 1-118-79026-X<br>1-118-79022-7<br>1-118-79013-8 |
| Descrizione fisica | 1 online resource (192 p.) |
| Collana | Focus Computer Engineering Series |
| Altri autori (Persone) | UhrigSascha<br>SainratPascal |
| Disciplina | 004.33 |
| Soggetti | Real-time data processing<br>Computer architecture |
| Lingua di pubblicazione | Inglese |
| Formato | Materiale a stampa |
| Livello bibliografico | Monografia |
| Note generali | Description based upon print version of record. |
| Nota di bibliografia | Includes bibliographical references and index. |
| Nota di contenuto | Cover; Title Page; Contents; Preface; CHAPTER 1. REAL-TIME SYSTEMS AND TIME PREDICTABILITY; 1.1. Real-time systems; 1.1.1. Introduction; 1.1.2. Soft, firm and hard real-time systems; 1.1.3. Safety standards; 1.1.4. Examples; 1.2. Time predictability; 1.3. Book outline; CHAPTER 2. TIMING ANALYSIS OF REAL-TIME SYSTEMS; 2.1. Real-time task scheduling; 2.1.1. Task model; 2.1.2. Objectives of task scheduling algorithms; 2.1.3. Mono-processor scheduling for periodic tasks; 2.1.4. Scheduling sporadic and aperiodic tasks; 2.1.5. Multiprocessor scheduling for periodic tasks; 2.2. Task-level analysis<br>2.2.1. Flow analysis: identifying possible paths2.2.2. Low-level analysis: determining partial execution times; 2.2.3. WCET computation; 2.2.4. WCET analysis tools; 2.2.5. Alternative approaches to WCET analysis; 2.2.6. Time composability; CHAPTER 3. CURRENT PROCESSOR ARCHITECTURES; 3.1. Pipelining; 3.1.1. Pipeline effects; 3.1.2. Modeling for timing analysis; 3.1.3. Recommendations for predictability; 3.2. Superscalar architectures; 3.2.1. In-order execution; 3.2.2. Out-of-order execution; 3.2.3. Modeling for timing analysis; |

3.2.4. Recommendations for predictability; 3.3. Multithreading
3.3.1. Time-predictability issues raised by multithreading3.3.2. Time-predictable example architectures; 3.4. Branch prediction; 3.4.1. State-of-the-art branch prediction; 3.4.2. Branch prediction in real-time systems; 3.4.3. Approaches to branch prediction modeling; CHAPTER 4. MEMORY HIERARCHY; 4.1. Caches; 4.1.1. Organization of cache memories; 4.1.2. Static analysis of the behavior of caches; 4.1.3. Recommendations for timing predictability; 4.2. Scratchpad memories; 4.2.1. Scratchpad RAM; 4.2.2. Data scratchpad; 4.2.3. Instruction scratchpad; 4.3. External memories; 4.3.1. Static RAM
4.3.2. Dynamic RAM4.3.3. Flash memory; CHAPTER 5. MULTICORES; 5.1. Impact of resource sharing on time predictability; 5.2. Timing analysis for multicores; 5.2.1. Analysis of temporal/bandwidth sharing; 5.2.2. Analysis of spatial sharing; 5.3. Local caches; 5.3.1. Coherence techniques; 5.3.2. Discussion on timing analyzability; 5.4. Conclusion; 5.5. Time-predictable architectures; 5.5.1. Uncached accesses to shared data; 5.5.2. On-demand coherent cache; CHAPTER 6. EXAMPLE ARCHITECTURES; 6.1. The multithreaded processor Komodo; 6.1.1. The Komodo architecture; 6.1.2. Integrated thread scheduling
6.1.3. Guaranteed percentage scheduling6.1.4. The jamuth IP core; 6.1.5. Conclusion; 6.2. The JOP architecture; 6.2.1. Conclusion; 6.3. The PRET architecture; 6.3.1. PRET pipeline architecture; 6.3.2. Instruction set extension; 6.3.3. DDR2 memory controller; 6.3.4. Conclusion; 6.4. The multi-issue CarCore processor; 6.4.1. The CarCore architecture; 6.4.2. Layered thread scheduling; 6.4.3. CarCore thread scheduling algorithms; 6.4.4. Conclusion; 6.5. The MERASA multicore processor; 6.5.1. The MERASA architecture; 6.5.2. The MERASA processor core; 6.5.3. Interconnection bus
6.5.4. Memory hierarchy

Sommario/riassunto

Building computers that can be used to design embedded real-time systems is the subject of this title. Real-time embedded software requires increasingly higher performances. The authors therefore consider processors that implement advanced mechanisms such as pipelining, out-of-order execution, branch prediction, cache memories, multi-threading, multicorearchitectures, etc. The authors of this book investigate the timepredictability of such schemes.