Lecture 1&2

Hardware/Software Codesign of Embedded Systems: Introduction and Course Organization

- Difficulties with the design of heterogeneous Hardware/Software systems
- Requirements of modern embedded systems
- Embedded system design flow
- Research issues
- Course topics

Readings

1. W. Wolf, “Hardware-Software Codesign of Embedded Systems”, Proceedings of the IEEE, V82, No7, 1994.
2. G. De Micheli, R.K. Gupta, “Hardware-Software Codesign”, Proceedings of the IEEE, V85, No3, 1997.
3. R. Ernst, “Codesign of Embedded Systems: Status and Trends”, IEEE Design&Test of Computers, V15, No2, 1998.

Lecture 3&4:

System Modelling. Models of Computation and System Specification Languages

- Models of Computation
  - Basic models, specific features, comparison
  - Multimodel specification
- Specification Languages
  - Specification Languages and their relation to models of computation
  - Multilanguage specification and Cosimulation
- Formal verification
  - Formal verification approaches
  - Model checking

Readings

1. Luciano Lavagno, Alberto Sangiovanni-Vincentelli, Ellen Sentovich, "Models of Computation for Embedded System Design", Ahmed A. Jerraya and Jean Mermet eds.: System Level Synthesis, Kluwer 1999.
2. Stephen Edwards, Luciano Lavagno, Edward Lee, Alberto Sangiovanni-Vincentelli, "Design of Embedded Systems: Formal Models, Validation, Synthesis", Proceedings of the IEEE, Vol85, No3, 1997.
3. Stephen Edwards, "Languages for Digital Embedded Systems", Chapters 11, 12, Kluwer 2000.
4. Ahmed A. Jerraya, M. Romdhani, et al., "Multilanguage Specification for System Design", Ahmed A. Jerraya and Jean Mermet eds.: System Level Synthesis, Kluwer 1999.
5. Carlos Delgado Kloos, Simon Pickin, et al., "High-level Specification Languages for Embedded System Design", Ahmed A. Jerraya and Jean Mermet eds.: System Level Synthesis, Kluwer 1999.
6. Joost Pieter Karoen, "System Validation"
7. J.R. Burch, E.M.Clarke, K.L. McMillan, "Sequential Circuit Verification Using Symbolic Model Checking", Proc. DAC, 1990.
8. R. Alur, C. Courcoubetis, D. Dill, "Model-Checking for Real-Time Systems", Proc 5th Symp. on Logic in Comp. Science, 1990.

Lecture 5&6

Processors and Architectures for Embedded Systems

- General Purpose vs. Application Specific Processors
  - Instruction set, Memory, Interconnect and control
  - DSPs, Microcontrollers, VLIW processors
  - Design Challenges
- Core (IP)-based design
  - Reusable components
  - Communication-based design
  - Platform-based design
- Reconfigurable Systems
  - Hardware/Software partitioning with reconfigurable processors
  - Dynamically reconfigurable systems

Readings

1. Rolf Ernst, "Embedded Systems Architecture", Ahmed A. Jerraya and Jean Mermet eds.: System Level Synthesis, Kluwer 1999.
2. Manfred Schlett, "Trends in Embedded Microprocessor Design", IEEE Computer, August '98.
3. M.F. Jacome, "Design Challenges for New Application-Specific Processors", IEEE Design & Test of Computers, April-June 2000.
4. J.A. Fisher, "Customized Instruction-Sets for Embedded Processors", Proc. DAC, 1999.

5. D. Gajski, et al., "IP-Centric Methodology and Design with the SpecC Language", Ahmed A. Jerraya and Jean Mermet eds.: System Level Synthesis, Kluwer 1999.
6. K. Keutzer, et al. "System level Design: Orthogonalization of Concerns and Platform-Based Design", IEEE Transactions on CAD, V19, N12, 2000.
7. J Rowson, A Sangiovanni-Vincentelli, "Interface-based Design", Proc. DAC, 1997.
8. A.M. Rincon et al.: "Core-Design and System-on-a-Chip Integration", IEEE Design & Test of Computers, October-December 1997.
9. D. Lyonnard, S. Yoo, et al., "Automatic generation of Application-Specific Architectures for Heterogeneous Multiprocessor System-on-Chip", Proc. DAC 2001.

10. Yanbing Li, et al., "Hardware-Software Codesign of Embedded Reconfigurable Architectures", Proc. DAC, 2000.
11. M. Kaul, R. Vemuri, "Temporal Partitioning Combined with Design Space Exploration for Latency Minimization of Run-Time Reconfigured Designs", Proc. DATE 1999.
12. S. Ganesan, R. Vemuri, "An Integrated Temporal Partitioning and Partial Reconfiguration Technique for Design Latency Improvement", Proc. DATE 2000.

Lecture 7&8

Code Generation and Retargetable Compilers

- Compiler Generators and Retargetable Compilers
  - Front end processing
  - Back end processing
  - Processor modeling
- Specific issues related to embedded processor architectures
  - DSP processors
  - SIMD instructions
  - VLIW processors

Readings

1. M. Imai, Y. Takeuchi, et al., “Compiler Generation Techniques for Embedded Processors and their Application to Hw/Sw Codesign”, Ahmed A. Jerraya and Jean Mermet eds.: System Level Synthesis, Kluwer 1999.
2. C. Liem, P. Paulin, “Compilation techniques and Tools for Embedded Processor Architectures”, Jorgen Staunstrup and Wayne Wolf eds.: Hardware/Software Co-Design: Principles and Practice, Kluwer 1997.
3. R. Leupers, “Code generation for Embedded Processors”, Proc. ISSS, 2000.
4. J. van Praet, et al., “Processor Modelling and Code Selection for Retargetable Compilation”, ACM Transactions on Design Automation of Electronic Systems, V6, N3, 2001.
5. R. Leupers, S. Bashford, “Graph-Based Code Selection techniques for Embedded Processors”, ACM Transactions on Design Automation of Electronic Systems, V5, N4, 2000.
6. R. Leupers, “Instruction Scheduling for Clustered VLIW DSPs”, Proc. Int. Conf. on Parallel Architectures and Compilation Techniques, 2000.

Lecture 9

Software Performance Estimation by Static Analysis

- Program path analysis
- Microarchitecture modeling
    - Cache memory
    - Pipeline architecture
    - Branch prediction

Readings

1. Y.-T. S. Li, S. Malik, A. Wolfe, “Performance Estimation of Embedded Software with Instruction Cache Modeling”, ACM Transactions on Design Automation of Electronic Systems, V4, N3, 1999.
2. A. Colin, I. Puaut, “Worst Case Execution Time Analysis for a Processor with Branch Prediction”, Real-Time Systems, V18, N2/3, 2000.
3. A. Hergenham, W. Rosenstiel, “Static Timing Analysis of Embedded Software on Advanced Processor Architectures”, Proc. DATE 2000.
4. R. Ernst, W. Ye, “Embedded Program Timing Analysis Based on Path Clustering and Architecture Classification”, Proc. ICCAD 1997.

Lecture 10&11

System-Level Power/Energy Optimization

- Main issues in system-level power/energy optimization
  - System modeling
  - Hardware and software implementation
  - Dynamic power management
  - Computing, memory, communication
- Dynamic power management
  - Modeling issues
  - Predictive, adaptive, and stochastic techniques
- Power estimation and low power software generation
- Low power/energy scheduling for real-time systems
  - Variable voltage systems
  - Static and dynamic approaches
  - Energy efficient priority-based scheduling

Readings

1. G. De Micheli, L. Benini, A. Bogliolo, “Dynamic Power Management of Electronic Systems”, Ahmed A. Jerraya and Jean Mermet eds.: System Level Synthesis, Kluwer 1999.
2. Y.-H. Lu, G. De Micheli, “Comparing System-Level Power Management Policies”, IEEE Design & Test of Computers, March-April, 2001.
3. L. Benini, G. De Micheli, “System-Level Power Optimization: Techniques and Tools”, ACM Transactions on Design Automation of Electronic Systems, V5, N2, 2000.
4. W. Fornaciari, P Gubian, et al., “Power Estimation of Embedded Systems: A hardware/Software Codesign Approach”, IEEE Transactions on VLSI Systems, V6, N2, 1998.
5. M. T.-C. Lee, V. Tiwari, et al., “Power Analysis and Minimization Techniques for Embedded DSP Software”, IEEE Transactions on VLSI Systems, V5, N1, 1997.
6. T. Ishihara, H. Yasuura, “Voltage Scheduling Problem for Dynamically Variable Voltage Processors”, Proc. Int. Symp. on Low Power Electronics and Design, 1998
7. T. Okuma, T. Ishihara, H. Yasuura, “Real-Time Task Scheduling for a Variable Voltage Processor”, Proc. Int. Symp. on System Synthesis, 1998.
8. T. Okuma, T. Ishihara, H. Yasuura,, “Software Energy reduction Techniques for Variable-Voltage Processors”, IEEE Design & Test of Computers, March-April, 2001.
9. Y. Shin, K. Choi, “Power Conscious Fixed Priority Scheduling for Hard Real-Time Systems”, Proc. DAC 1999.
10. G. Quan, X. Hu, “Energy Efficient Fixed-Priority Scheduling for Real-Time Systems on Variable Voltage Processors”, Proc. DAC 2001.

Lecture 12&13

Hardware/Software Codesign Environments

- The Cosyma System
- The Cosmos Environment
- The SpecSyn Environment
- Synthesis of Distributed Embedded Systems
- The POLIS Environment
- The CoWare Environment

Readings

1. W. Wolfe, “Hardware/Software Co-synthesis Algorithms”, Ahmed A. Jerraya and Jean Mermet eds.: System Level Synthesis, Kluwer 1999.

2. A. Österling, T. Benner, R. Ernst, et al., “The Cosyma System”, Jorgen Staunstrup and Wayne Wolf eds.: Hardware/Software Co-Design: Principles and Practice, Kluwer 1997.
3. C.A. Valderrama, et al., “Cosmos: A Transformational Co-Design Tool for Multiprocessor Architectures”, Jorgen Staunstrup and Wayne Wolf eds.: Hardware/Software Co-Design: Principles and Practice, Kluwer 1997.
4. D. Gajski, et al.: “SpecSyn: An Environment Supporting the Specify-Explore-Refine Paradigm for Hardware/Software System Design”, IEEE Transactions on VLSI Systems, V6, N1, 1998.
5. W. Wolfe, “An Architectural Co-synthesis Algorithm for Distributed, Embedded Computing Systems”, IEEE Transactions on VLSI Systems, V5, N2, 1997.
6. M. Chiodo, et al., “A Case Study in Computer-Aided Co-design of Embedded Controllers”, Design Automation for Embedded Systems, V1, N1-2, 1996.
7. D. Verkest, et al.: “CoWare - A design environment for heterogeneous hardware/software systems”, Design Automation for Embedded Systems, V1, N4, 1996.