Course Description

Embedded Parallel Computing, 7.5 credits

Parallelldatorer I Inbyggda System, 7,5 hp

Course Start: Monday January 16 2012, room R3149. *Introductory Lecture.*

Lecturer: Assistant Professor Tomas Nordström

Document date: 2012-01-15

Welcome to the world of embedded parallel computing!

Embedded electronic systems are finding increased applications in our daily life. In order to meet the application demands in embedded systems, parallel computing is used. The focus of the course is how parallel computing can be used to enhance performance and improve energy efficiency of embedded systems. Further, it is intended to give a general insight into current research and development in regard to parallel architectures and computation models. Since the course is an advanced level course, the students are expected to have a basic knowledge about the fundamentals of computer architecture and their common programming methodologies.

Administrative information

Syllabus	Follow link on <http://www.hh.se/DO8003>
Study period:	Period 3 in academic year 2011-2012 (January - March)
Course code:	DO8003
Programme:	The course is included as an elective course in the Master Programme in Embedded and Intelligent Systems 120 credits, the Master Programme in Information Technology 120 credits, and the Master Programme in Computer Systems Engineering 60 credits.
Course responsible:	Tomas Nordström Tomas.Nordstrom@hh.se; Room E313; Tel: +46 35 16 7334 School of Information Science, Computer and Electrical Engineering, Halmstad University, Box 823, SE-30118, Halmstad
Other teachers:	Zain-ul-Abdin <Zain-Ul-Abdin@hh.se>
Examiner:	Tomas Nordström <Tomas.Nordstrom@hh.se>
Supervision time slots	Tomas Nordström will normally be available for supervision Mondays 13:15-14:30, and Thursdays 10:15-11:30

Course Objectives and Learning Outcomes

The course is intended to develop the student's knowledge and understanding of how parallel computing can be used, as a way to meet application demands in embedded systems, such as performance and power efficiency. Further, it is intended to give a general insight into current research and development in regard to parallel architectures and computation models. Parallelism of various types exists in all modern computer architectures, and knowledge about how to apply parallelism is necessary, in particular, when designing embedded computer systems.

Upon completion of the course, the student shall be able to:

Knowledge and understanding

- describe and explain the most important parallel architecture models, as well as parallel programming models, and discuss their respective pros, cons, and application opportunities
    - That is, the student shall be able to identify where current research and development is heading and be able to compare different proposals that are relevant for future design of parallel embedded systems.

Skills and abilities

- program parallel computer systems intended for embedded applications
    - That is, the student shall be able to apply the knowledge of the computer architecture and its programming techniques, by programming some basic signal processing applications intended for embedded systems.
- describe, evaluate, and discuss how the choice of programming model and method influences, e.g., execution time and required resources
    - That is, the student shall be to explain how various programming models and methodologies can be used to achieve better performance and energy efficiency.
- read and understand scientific articles in the area, to review and discuss them and to make summaries and presentations; as well as identify relevant research publications and research groups in the area
    - That is, the student shall be able to read and understand scientific articles in journals and conference publications in the area of parallel embedded systems and be able to follow the developments and trends in the field of parallel computing. In order to do that the student shall be able to identify relevant research publications and research groups in the area.

Judgement and approach

- relate the state of the art in the area to the current research and development, in particular such research and development that is important for the design of embedded systems
    - That is, the student shall be able to identify where current research and development is heading and be able to compare different proposals that are relevant for future design of parallel embedded systems.

Primary Content

The course is divided into a lecture part, a laboratory part including a small project, and a seminar part.

The lecture part initially gives a motivation for parallelism, based on demands on embedded computing (such as performance and power efficiency) and applications that require parallelism. Then it presents the fundamentals of parallel architectures (forms of parallelism, SIMD, MIMD, dataflow, reconfigurable architectures, interconnection networks, etc.) and parallel programming models (shared memory, message passing, stream programming, communicating sequential processes, process networks, etc.). Example architectures and programming techniques are presented and discussed. This part will also discuss various limitations, such as power and energy limitations or chip technology, which will influence future architecture developments.

The laboratory part provides hands-on experience of embedded parallel computing, primarily based on many-core processors on a chip and their available programming tools.

In the seminar part of the course, course participants make detailed studies of various sub-areas or specific architectures and lead seminars in these. The university's research projects are included in these special studies.

Teaching Formats

Instruction is in the form of lectures, laboratory sessions, project tutoring and seminars. The latter are given by the students based on literature studies and discussions on subjects determined in consultation with the course instructor. All participants in a seminar prepare themselves by reading an introductory text for the topic, while those responsible for the seminar search additional information and make presentation of this as a background for further discussion. The project and the seminars shall be documented in short reports. The laboratory sessions are mandatory, and also leading at least one seminar.

The three forms of teaching and instructions used in this course:

Lecture part (L1, L2, etc. in the course planning). Overview lectures will be given as a preparation for the students' own reading of the material and problem solving; Discussion sessions will be provided in the following lecture after the students' own reading and problem solving.

Seminar part (S1, S2, etc. in the course planning). Seminars prepared by the students: Presentations of current "hot topics" followed by discussions.

Project part (P1, P2, etc. in the course planning). Hands-on parallel programming of multiprocessor on a chip, in laboratory sessions followed by a small project.

All teaching will be conducted in English.

Detailed Course Description

Lectures

Lecture 1: Introduction to Advanced Computer Architecture and Parallel Processing

Theoretical basis of parallel computing, and speedup. The course starts with an introductory lecture about the course and provides an overview of the course literature and the topics covered in the course. After that some theoretical concepts of parallel computing such as fine-grained vs coarse-grained and shared memory vs distributed memory are discussed. This lecture describes the two general classes of parallel processing architectures i.e., single instruction multiple data (SIMD) and multiple instructions multiple data (MIMD). The differentiating features of SIMD/MIMD architectures are presented and their corresponding merits and demerits are discussed. Students will also learn about how to calculate speedups to compare different approaches of parallel computing.

Reading Material: [ACAPP] (Overview is given for Chapter 1&3);

*Lecture 2: Parallelism in microprocessor*s

Key concepts: RISC, Superscalar, VLIW micro-architecture, Single-instruction multiple-data streams (SIMD) parallel architectures, and vectoring. The basic architectural differences of reduced instruction set computers (RISC), superscalar and very long instruction word (VLIW) computers are discussed. The parallelism is identified and contrasted to explicit SIMD and vectoring architectures. The lecture also presents the salient features of these micro-architectures and highlights their pros and cons with respect to architectural complexity and performance.

Reading Material: Wikipedia. [ACAPP] Chapter 1.3

Lecture 3: Multiprocessors Interconnection Networks

In this lecture, both classical multiprocessor interconnection network technologies as well as interconnection networks with a Network-on-Chip (NoC) will be presented. the fundamental characteristics of interconnection networks such as network topology, flow control, routing, and quality of service are discussed. The lecture will also describe the basic components of NoC i.e., network adapter, routing node, and links.

Reading Material: [ACAPP] (Overview is given for Chapter 2&3.3); Introduction article on network on chip (NoC)

Lecture 4: Multiple-instruction multiple-data streams (MIMD) parallel architectures

The lecture will discuss the difference between the two main MIMD architectural concept: Shared Memory Architecture and Message Passing Architecture

Reading Material: [ACAPP] (Overview is given for Chapter 4&5);

Lecture 5: The anatomy of a modern multiprocessor, the multi-core processors

The lecture will discuss the anatomy of one of the modern multi-core processors that we can find in any modern stationary computer.

Reading Material: Chip manufacturer data sheets and other on Internet available information, HotChips conference articles. Article: Parallelism via Multithreaded and Multicore CPUs (pdf, 1 MB)

Lecture 6: Fundamentals of embedded many-core architectures

In this lecture, information is provided about the basic architecture of individual processing elements and the kind of interconnection networks adopted by the many-core architectures. Some emerging examples of many-core architectures are also presented.

Reading Material: Chip manufacturer data sheets and other on Internet available information, HotChips conference articles

Lecture 7: Programming models and methodologies for parallel embedded processors

The basic definition of computation models and their corresponding programming models is presented in this lecture. Afterwards some emerging computation models such as stream processing model, communicating sequential processes (CSP), and Kahn's process networks are discussed and the example languages based on these models of computations are described.

Reading Material: [ACAPP] (Overview is given for Chapter 8&9);

Lecture 8: Energy efficiency in modern embedded parallel processors

Energy consumption and energy scalability is becoming the main driver for new (parallel) computer architectures. In this lecture we look into how this focus on energy consumption have influenced current architectures and how on-going research suggests the future embedded parallel processor will look like.

Reading Material: Key article(s) describing how energy efficiency is driving current architecture research.

In addition to above lectures there will be two additional lectures, described below, given in connections to the laboratory exercises.

Seminars
Students will be provided a list of different topic studied during the course along with an article associated with the topic and students choose one of these topics to present. Each student is given a quiz based on the article. Presentations are performed in groups of two or three student. All the students are expected to have read the article and prepare questions or points to be raised for discussion after the presentation.

Introductory Texts for Seminars
NOTE: Please respect that these texts are copyright protected!

Topic 1: Processor Arrays
Text: The RAW Microprocessor (pdf, 420 kB)

Topic 2: Energy Efficient Computing for Mobile Applications
Text: Mobile Supercomputing for the Next-Generation Cell Phone (pdf, 1 MB)

Topic 3: Parallel Processing with Graphics Processors
Text: Parallel Processing with CUDA (pdf, 1 MB)

Topic 4: Future Computing Architectures
Text: From Microprocessors to Nanostore (pdf, 2 MB)s

Topic 5: Security in Embedded Systems
Text: Researchers Fight to Keep Implanted Medical Devices Safe from Hacker (pdf, 2 MB)s

Topic 6: Future of Processor Design
Text: Rethinking Digital Design: Why Design Must Change (pdf, 1 MB)

Topic 7: Heterogeneous Parallel Processors
Text: (pdf, NN MB)

Programming Labs

Two lab sessions are planned in the course. The students are supposed to work in groups of two. In the first sessions, students will use MPI library to program some simple tasks and calculate speedup by executing their application on a parallel machine. In the second session, students get the opportunity to implement some basic signal processing functions in Ambric structured programming model and simulate their results in the eclipse integrated development environment provided by Ambric. Labs Material: MPI_Lab1.pdf, MPI_Lab2.pdf

Lab introduction P1: Introduction of MPI library and parallel hardware used in the lab. The intention for this lecture is to prepare students for the accompanying lab sessions. In this regard, an introduction is provided to the MPI library routines used to implement communication and a description about the hardware infrastructure used in the lab is also provided to get them familiarize.

Reading Material: Handouts of lecture slides, MPI_docs.pdf

Lab introduction P3: Introduction to the Ambric processor architecture and its structured programming model. A description about the Ambric processor architecture is provided in the lecture, highlighting its bric architecture and globally asynchronous locally synchronous (GALS) principal based clocking technique. The structured object programming model of Ambric is also introduced in the lecture.

Reading Material: Handouts of lecture slides, aDesignersUserGuide.pdf, aDesignersRefGuide.pdf

Programming project

In the project part, students are expected to program a specified task in either MPI or Ambric programming model according to their choice. After implementing the task, they are expected to report their results in the form of a report. Labs Material: MPI_Project.pdf

Examination

The laboratory exercises and programming project are obligatory parts of the course. The student also needs to make at least one seminar presentation to pass the course. The programming project is examined based on the reports submitted by the students. For the lectures part, the students will be assessed in a written examination at the end of the course. Bonus points for the written exam may be earned through participation in the seminars and providing correct answers on the written quizzes. The overall grade given is Fail, 3, 4, or 5 based on the performance of the students in course examination, seminar quizzes, and their performance in presenting their own seminar.

The written exam will take place in week 11, 2012. First opportunity for re-examination will be week NN, 2012

Course Evaluation

Course evaluation is part of the course. This evaluation should offer guidance in the future development and planning of the course. Course evaluations will be documented and made available to the students. As this is the first time Tomas Nordström is responsible for this course he is eager to receive feedback both on his own performance as well as the course format.

Course Literature

The main textbook to be used for the lecture part of the course is:
[ACAPP] H. El-Rewini & M. Abd-el-Barr, "Advanced Computer Architecture and Parallel Processing", John Wiley & Sons, 2005, ISBN 0-471-46740-5. The book is electronically available in full-text form through the database ebrary, available through the University's library.

Other suitable books, recommended for reading, are

Kornaros, G., Multi-core Embedded Systems (Embedded Multi-core Systems), 2010.

Wolf, W., High-performance Embedded Computing: Architectures, Applications, and Methodologies, Morgan Kaufmann, 2007.

Hennessy, J.L. and D.A. Patterson, Computer Architecture: A Quantitative Approach, Fourth Edition, Morgan Kaufmann, 2006.

Almasi, G.S. and A. Gottlieb, Highly Parallel Computing, 2nd Edition, Benjamin/Cummings, 1994.

Zomaya, A.Y.H. (ed), Parallel and Distributed Computing Handbook, McGraw-Hill, 1996.

For the lectures, seminars and labs part, additional literature will be provided electronically. In addition, students are encouraged to use scientific publications databases that can be accessed from Halmstad University library.

Detailed Planning of Embedded Parallel Computing (DO 8003), spring 2011

This is the preliminary plan for the course. It will be updated during the course of the course. Tomas Nordström will hold the Lectures L*n* and Seminars S*n*, while the Laboratory and Project lectures P*n* will be held by Zain-ul-Abdin.

Lecture	Date	Time & room	Content
L1	2012-01-16	13:15-15:00, R3149	*Lecture 1: Introduction to Advanced Computer Architecture and Parallel Processing* Course introduction. Scan through Ch 1+3 in [ACAPP]. Problems to solve: Ch1: 4, 6, 10. Ch3: 5, 6, 8.
L2	2012-01-18	10:15-12:00, D215	*Lecture 2: Parallelism in microprocessors* Discuss content of Lecture 1. Overview of parallelism found in today's microprocessors including SIMD extensions. In reading, you may pay less attention to Sections 3.3 and 3.5,
L3	2012-01-26	10:15-12:00, R3149	*Lecture 3: Multiprocessors Interconnection Networks* Discuss content of Lecture 2. Scan through Ch2+Ch3.3 in [ACAPP]. Problems to solve: Ch 2: 2, 4, 10-14. Ch 3: 1
L4	2012-01-27	10:15-12:00, R3149	*Lecture 4: Multiple-instruction multiple-data streams (MIMD) parallel architectures* Discuss content of Lecture 3. Scan through Ch4 + Ch5 in [ACAPP]. In reading, you may pay less attention to Sections 3.5, 4.4.3 - 4.4.5, 4.5.1 and 4.5.2.
L5	2012-01-31	10:15-12:00, R3149	*Lecture 5: The anatomy of a modern multiprocessor, the multi-core processors* Discuss content of Lecture 4. Scan through [Modern multi-core arch]
L6	2012-01-02	10:15-12:00, R3149	*Lecture 6: Fundamentals of embedded many-core architectures* Discuss content of Lecture 5. Scan through [single-chip many-core arch] Introduction to Ambric massively parallel processor on a chip [Ambric]
P1	2012-02-06	13:15-15:00, R3149	Lab and project preparation lecture 1 [Ambric] and [Ambric lecture 1]
P2	2012-02-09	10:15-12:00, D408	Lab 1 [Ambric Lab 1]. Note: MANDATORY!
L7	2012-02-15	13:15-15:00, R3149	*Lecture 7: Programming models and methodologies for parallel embedded processors* Discuss content of Lecture 6 and Programming Lab. Scan through Ch8 + Ch9 in [ACAPP] and OpenCL documentation Introduction to seminar topics.
L8	2012-02-17	13:15-15:00, R3149	*Lecture 8a: Energy Efficiency and Parallel Embedded Processors* Introduction to energy efficiency in modern embedded parallel processors *Lecture 8b: Group discussions of seminar topics*
P3	2012-02-17	13:15-15:00, R3149	Lab and project preparation lecture 2 [Ambric lecture 2]
S1	2012-02-20	13:15-15:00, R3149	Seminar Group Meetings (Group 1-5)
P4	2012-02-22	13:15-15:00, D408	Lab 2 [Ambric Lab 2]. Note: MANDATORY!
P5	2012-02-24	10:15-12:00, D408	Project introduction [Ambric Project].
	~~2012-02-27~~	~~13:15-15:00, R3149~~	
S2	2012-03-01	10:15-12:00, R3149	Seminar Group "Topic N". ALL: Read Introductory Text!!
S3	2012-03-02	10:15-12:00, R3149	Seminar Group "Topic N". ALL: Read Introductory Text!!
S4	2012-03-06	8:15-10:00, R3149	Seminar Group "Topic N". ALL: Read Introductory Text!!
S5	2012-03-08	10:15-12:00, R3149	Seminar Group spare

Evaluation matrix

Evaluation matrix for the course Embedded Parallel Computing, 7.5 hp

Teaching goals	Examination form	Grading
Knowledge and understanding		
describe and explain the most important parallel architecture models, as well as parallel programming models, and discuss their respective pros, cons, and application opportunities	Written examination	Grading for the written examination will be Fail, 3, 4, 5
Skills and abilities		
program parallel computer systems intended for embedded applications	Laboratory exercises and programming project completion (WE*)	Obligatory participation in the parallel programming laboratory exercises
describe, evaluate, and discuss how the choice of programming model and method influences, e.g., execution time and required resources	Programming project report (WE*)	Obligatory participation in the parallel programming project
read and understand scientific articles in the area, to review and discuss them and to make summaries and presentations; as well as identify relevant research publications and research groups in the area	Student presentations and discussions during seminars (WE*)	One own seminar is obligatory. Bonus points are given towards the written examination for active participation, quality of presentation, and answers to quizzes
Judgement and approach		
relate the state of the art in the area to the current research and development, in particular such research and development that is important for the design of embedded systems	Student presentations and discussions during seminars (WE*)	Bonus points are given towards the written examination for active participation, quality of presentation, and answers to quizzes

(WE*) Even if these course goal are examined as describe above the student can additionally be examined during the written exam on the same goal. That is, there might be questions covering parallel programming as well as the seminar topics on the written exam.

Grading for the written examination and the course will be Fail, 3, 4, and 5 (where 5 is the highest grade). At least 50% of the maximally points possible on the written exam must be correct to pass. To be given grade 4 at least 75% must be correct, and for grade 5 the student needs to have 90% correct. The maximum bonus points given for seminar participation and quizzes will be in total 25%.