UNIVERSITÀ DI PISA

Master Program (Laurea Magistrale) in Computer Science and Networking

Academic Year 2012-13

High Performance Computing

Marco Vanneschi

Department of Computer Science, University of Pisa

Scuola Superiore Sant'Anna
di Studi Universitari e di Perfezionamento

Università di Pisa

Master Program (Laurea Magistrale) in Computer Science and Networking

High Performance Computing

Marco Vanneschi

Course Introduction

www.di.unipi.it/~vannesch section: Teaching

Course objectives

- Providing a solid framework of concepts and techniques in high-performance computing
  - Parallelization methodology and models
  - Support to parallel programming models and software development tools
  - Parallel Architectures
  - Performance evaluation (cost models)
- Methodology for studying existing and future systems
- Technology: state-of-the-art and trends
  - Parallel processors
  - Multiprocessors
  - Multicore / manycore / … / GPU
  - Shared *vs* distributed memory architectures
  - Programming models and their support

First course in HPC and enabling platforms

- Distributed Systems: Paradigms and Models (Prof. Danelutto), mandatory, 2nd year
  - **Preceded by HPC**

- Other courses in this area
  - Programming tools for parallel and distributed systems
  - Distributed enabling platforms
  - Parallel and distributed algorithms
  - …

A Computer Science approach

- Computer Science approach
  - Parallel computing: computational and programming models
  - Cost models of parallel computations
  - Computing architecture has its own concepts, principles, models, and techniques
  - Conceptual framework in common with the other disciplines of Computer Science:
    - Programming languages, algorithms, computability and complexity, …

Background and prerequisites

- An undergraduate-level course on *structured* computer architecture
  - Firmware level structuring
  - Assembler level, CPU architecture, compiling
  - Memory hierarchies and caching
  - Interrupt handling, exception handling
  - Process level, addressing space, low level scheduling, interprocess communication
  - Input/Output processing
- In Pisa: the course “Computer Architecture” (Bachelor Program in Computer Science) adopts such an approach

Lectures Notes in english on Background

1. System structuring: level, modules, cooperation models
2. Firmware level fundamentals
3. Assembler level
4. Memory hierachies and caching
5. Process level fundamentals and run-time support

Written ad-hoc for HPC course

Available in my page

First Part (Background) of Lecture Notes

Other reference books on Background

- D.A. Patterson, J.H. Hennessy, "*Computer Organization and Design: the Hardware/Software Interface*", Morgan Kaufman Publishers Inc.
- A. Tanenbaum, "*Structured Computer Organization*", Prentice-Hall.

- (in Italian: M. Vanneschi, "*Architettura degli Elaboratori*", PLUS, Pisa.)

Working approach

- As in any other course, it is fundamental to acquire skills and capabilities in concepts and principles, besides knowing the technologies.
- **Critical aptitude** must be properly developed.

- Interaction with the teacher is strongly recommended
  - Questions during the lectures
  - Presentation and discussion of exercises and problems
  - **HOME WORK**
  - **Question time** (“**orario di ricevimento**”) (in Italian for Italians)
    - **Monday 14:00 – 16:00, Tuesday 16:00 – 18:00**

Course Program

Part 0: Background – Structured Computer Architecture

Part 1: Structuring and Design Methodologies for Parallel Applications

Part 2: Parallel Architectures

fully covered by the Lecture Notes.

Course Program

Part 0: Background - Structured Computer Architecture

Structuring by levels and processing modules

The firmware level

The assembler machine and its basic interpreter

Processes and virtual memory

Memory hierarchies and cache architecture

Interprocess communication mechanisms and their run-time support

- The student must be aware of the way in which this Part has to be utilized.
- Because we cannot replicate an entire Computer Architecture basic course,
- during the initial lectures we'll review only some major concepts and techniques, which are fundamental to fully understand the issues studied in Part 1 and 2.
- A detailed treatment will not be provided during the initial lectures.
- The student is invited to use the Sections of the Part 0 in order to be (to become) able of applying the concepts and techniques needed in Part 1 and Part 2,
- and to fill any possible gap whenever it is necessary or for personal culture.

Course Program

Part 1: Structuring and Design Methodology for Parallel Applications

Structured parallelism at applications and process levels

Cost models

Impact of communications

Parallel computations as queueing systems / queueing networks

Parallel paradigms: Pipeline, Data-flow, Farm, Function partitioning, Data parallel

Parallel systems at the firmware level: Instruction level parallelism: Pipeline, superscalar, multithreaded CPUs; SIMD architectures and GPUs

Part 2: Parallel Architectures

Shared memory multiprocessors: SMP and NUMA architectures

Distributed memory multicomputers: Clusters and MPP architectures

Run-time support to interprocess communication

Interconnection networks

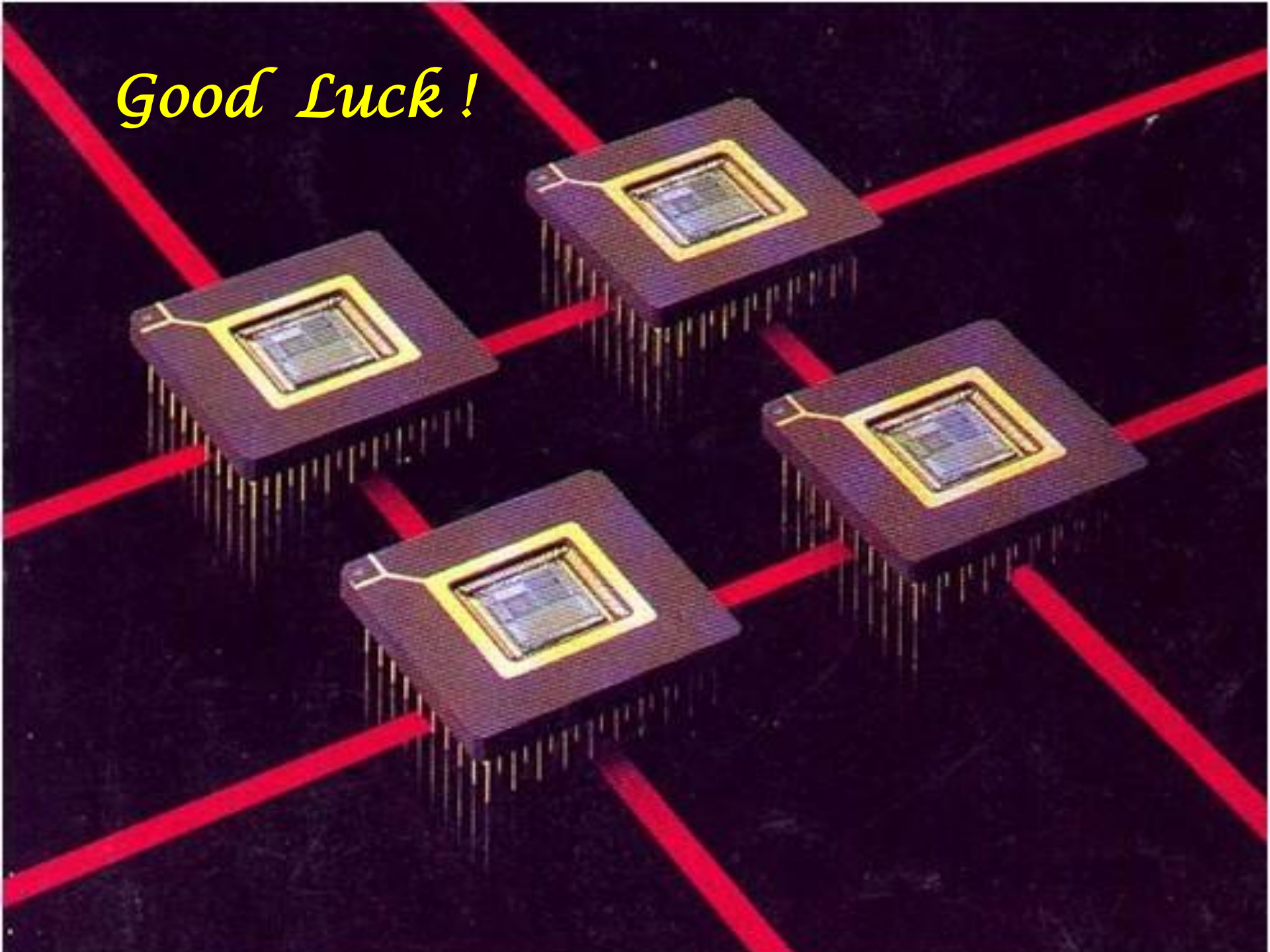
Performance evaluation

Multicore architectures

Exams

- **Written + oral exam**
- **Midterms**: see regulations
  - 5 '**APPELLI**': January - February (recommended !), June, July, September

- ***Registration*** to the exam on the Official Site of Corso di Laurea:
  - http://compass2.di.unipi.it/didattica, section Laurea Magistrale in Informatica e Netwoking, subsection "calendar"


Good Luck !