

High Performance Computing

0. Introduzione al corso

Andrea Marongiu
(andrea.marongiu@unimore.it)
AA 2018-2019

Credits

- Moreno Marzolla, Luca Benini - Università di Bologna, 2018
- Marwedel, Embedded System Design, Springer 2018,
- Wolf, Computers as Components 4th Ed. Morgan Kaufmann 2016
- Wolf, High-Performance Embedded Computing 2nd Ed. Morgan Kaufmann 2014
- Lee, Seshia: Introduction to Embedded Systems, A Cyber-Physical Systems Approach, 2nd Ed., MIT Press, 2017

Motivazioni per il corso

- Cos'è l'*high-performance computing*?

Con High Performance Computing (HPC)

*(in italiano calcolo ad elevate prestazioni), [...], ci si riferisce alle tecnologie utilizzate da **computer cluster** per creare dei sistemi di elaborazione in grado di fornire delle prestazioni molto elevate [...], ricorrendo tipicamente al **calcolo parallelo**.*

[fonte: Wikipedia]

Motivazioni per il corso

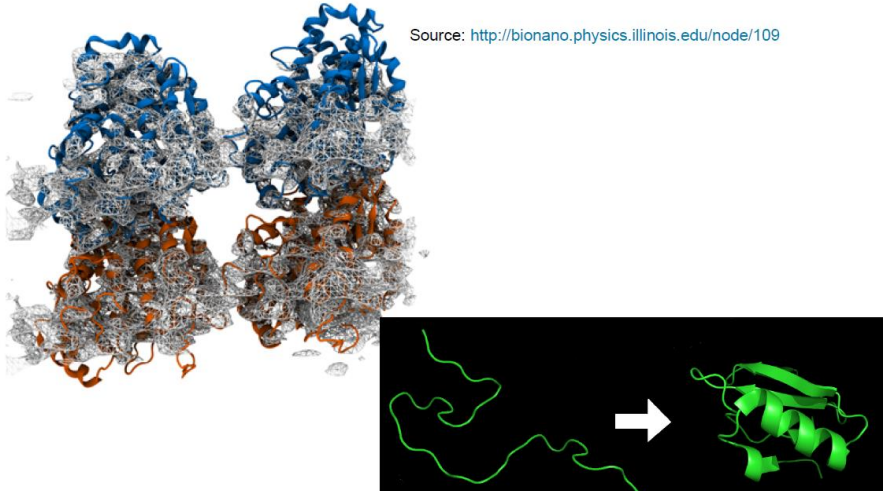
Storicamente...

High Performance Computing

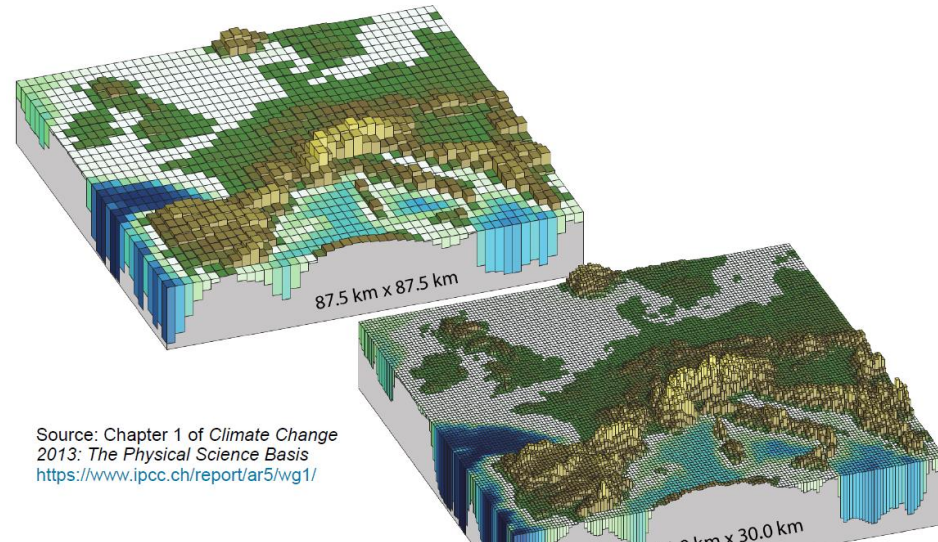
- Molte applicazioni hanno bisogno di elevata potenza di calcolo
 - Meteorologia, fisica, ingegneria, animazione 3D, finanza...
- Perché?
 - Per risolvere problemi più complessi
 - Per risolvere lo stesso problema in modo più accurato
 - Per risolvere lo stesso problema in minor tempo
 - Per sfruttare al meglio le risorse disponibili

Motivazioni per il corso

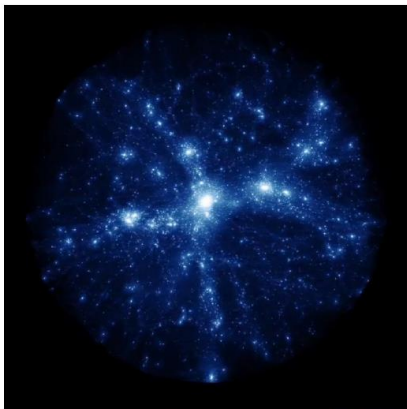
Molecular dynamics



Global Climate models



The Bolshoi Simulation



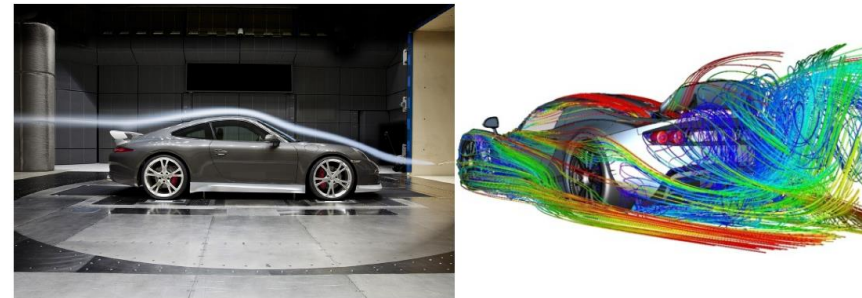
Bolshoi simulation <https://vimeo.com/29769051>

The **Bolshoi** Simulation recreates the large-scale structure of the universe; it required **6 million CPU hours** on NASA's Pleiades Supercomputer



Source : <https://www.nasa.gov/hecc/resources/pleiades.html>

Numerical Wind Tunnel



Source: <http://ecomodder.com/forum/showthread.php/random-wind-tunnel-smoke-pictures-thread-26678-12.html>

Motivazioni per il corso

Quando la programmazione parallela è necessaria

- Paradigma scientifico tradizionale
 - Sviluppa una teoria, e poi effettua esperimenti per confermare la teoria
- Paradigma ingegneristico tradizionale
 - Prima sviluppa un progetto, poi costruisci un prototipo
- Entrambi questi paradigmi stanno venendo rimpiazzati da **esperimenti e prototipazione numerica**
 - Certi fenomeni reali sono troppo complessi per essere modellati (es., previsioni del tempo)
 - Certi tipi di esperimenti sono troppo complessi, costosi, o pericolosi per essere condotti in laboratorio (es., simulazioni sismiche, gallerie del vento, progettazione di aerei, dinamica di ammassi stellari...)
- *Computational science*
 - Le simulazioni numeriche stanno diventando il nuovo modo per "fare scienza"

Motivazioni per il corso

La programmazione parallela in breve

- Fino a pochi anni fa, la programmazione parallela era un argomento di nicchia, dato che i calcolatori paralleli erano rari e costosi(ssimi)
- Oggi la programmazione parallela è essenziale dato che i processori paralleli sono ovunque



Apple iPhone 7
Quad-core 2.3 GHz
ARMv8-A



Huawei P9
Quad-core 2.5 GHz
Cortex-A72

Motivazioni per il corso

*I termini **high-performance computing** e **supercomputing** sono spesso utilizzati in maniera intercambiabile..*

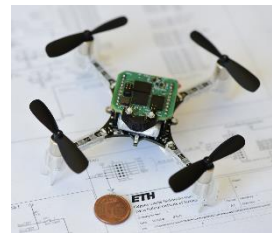


Ma questo **NON** è un corso sul **supercomputing** e sui **supercomputers**

Il focus è su architetture **embedded** ad alte prestazioni

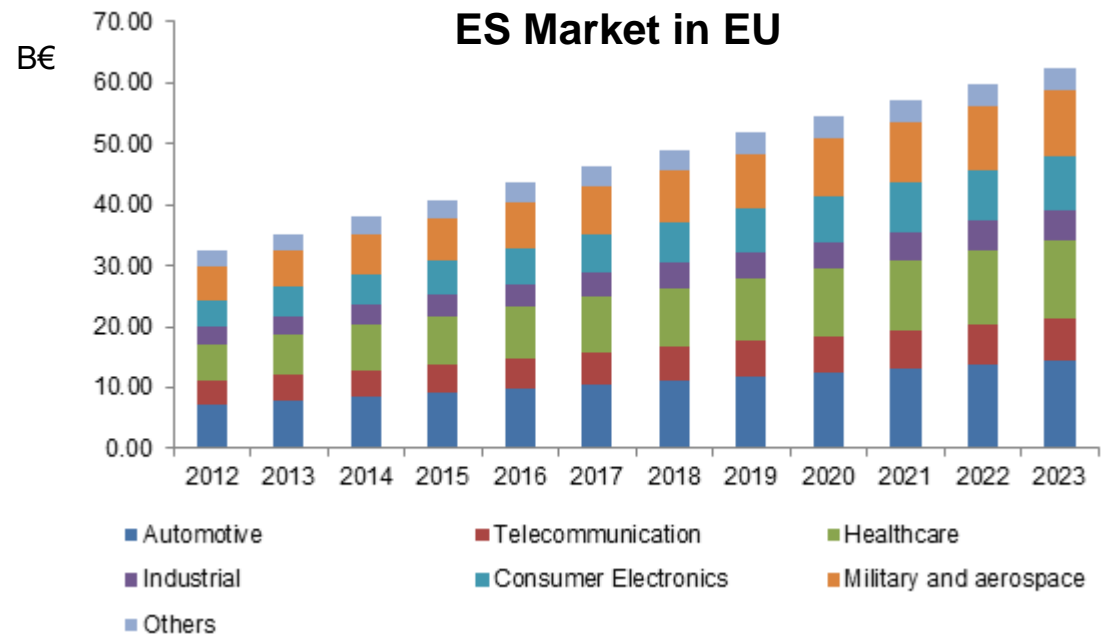
Motivazioni per il corso

- I computer sono dappertutto
 - Disappearing computer
 - Ubiquitous computing
 - Pervasive computing
 - Ambient intelligence
 - Post-PC era.
- Tecnologie abilitanti:
 - *Embedded Systems*
 - Tecnologie per la comunicazione



Definizione

- **Embedded computing system**: qualsiasi dispositivo che includa un computer programmabile, ma non è esso stesso un general-purpose computer.
- Sfrutta le caratteristiche dell'applicazione target per ottimizzare il suo design.



Esempi: ES per il trasporto

Gli ES tradizionali sono al giorno d'oggi ampiamente impiegati nei mezzi di trasporto:

- Automotive
- Avionica
- Treni



Elettronica di consumo

- Mobile
 - Phones
 - Tablets
 - Wearables



- Home
 - DTV (4K, OLED)
 - Game (PS4,
 - Smart white goods



Internet of Things (IoT)

Un mondo in cui gli oggetti sono raggiungibili in rete:

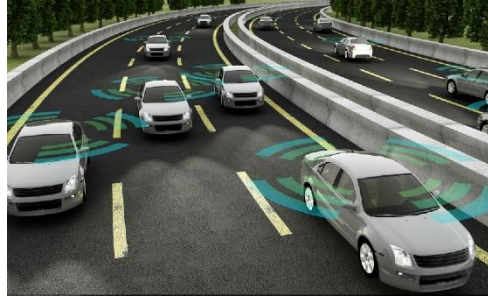
- Smart buildings and home appliances
 - washing machines, TVs, fridges, cookers, doors, chairs...
- Civil engineering structures
 - bridges, railways ...
- Wearable devices
 - smart watches, smart glasses, rings, clothes ...
- Medical devices
 - embedded pills
- possibly every THING...

Major Components of IoT



Embedded Artificial Intelligence

Autonomous Cars



Unmanned Aerial Vehicles



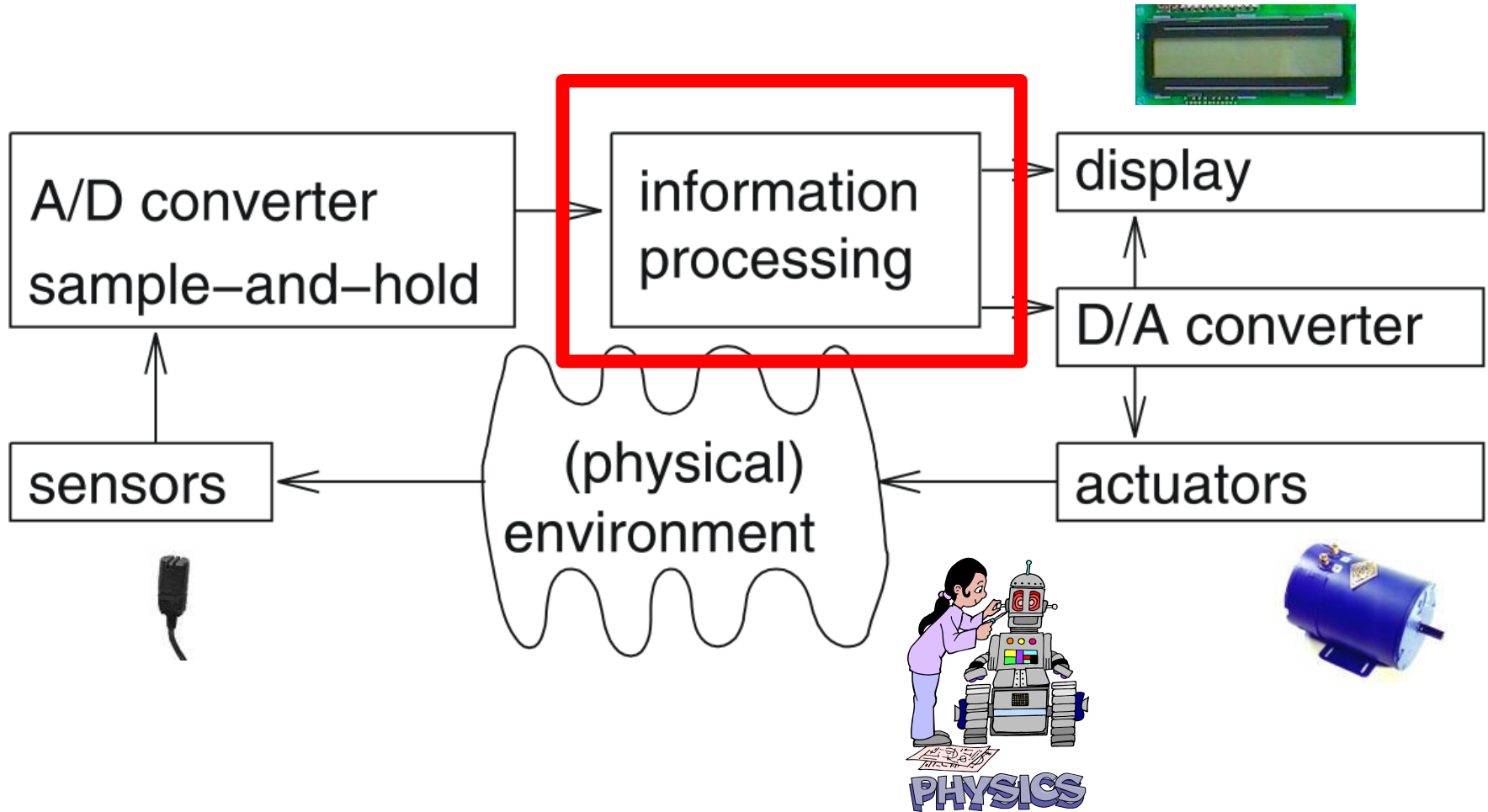
Humanoid Robots



Il presente e il futuro
degli ES

Embedded Systems

- The cyber-physical loop

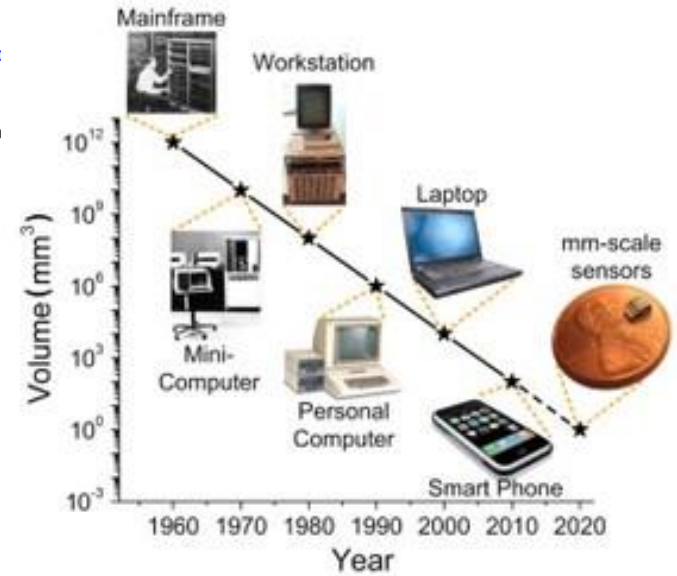
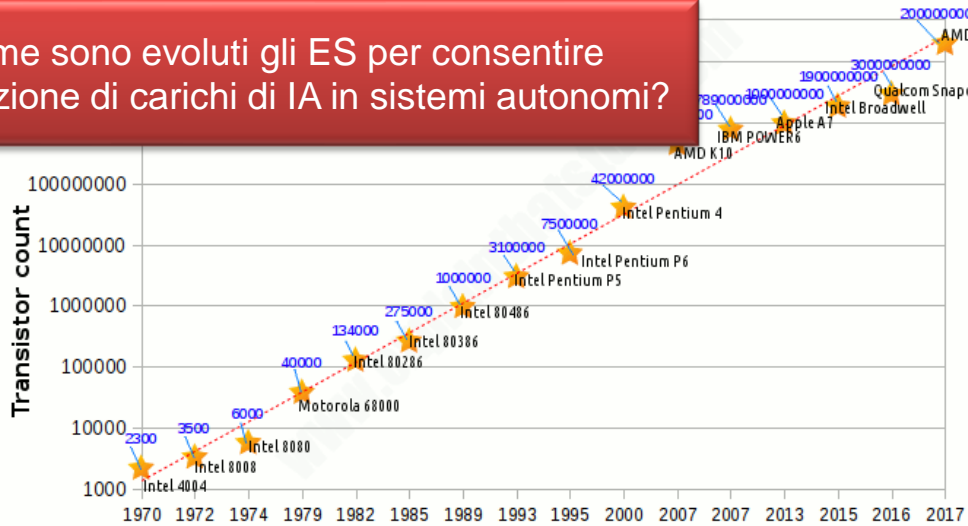


Motivazione per il corso

50 Years of Moore's law

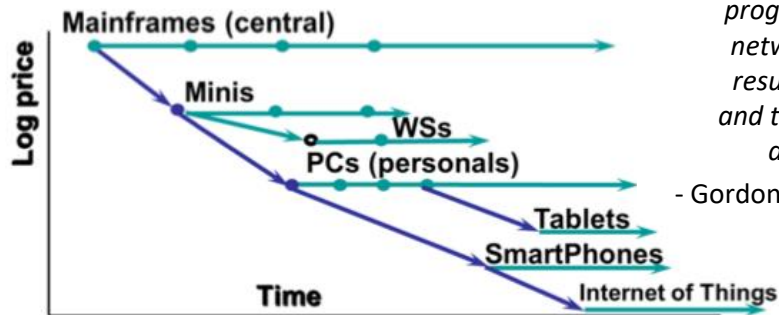
www.explainthatstuff.com

Come sono evoluti gli ES per consentire l'esecuzione di carichi di IA in sistemi autonomi?

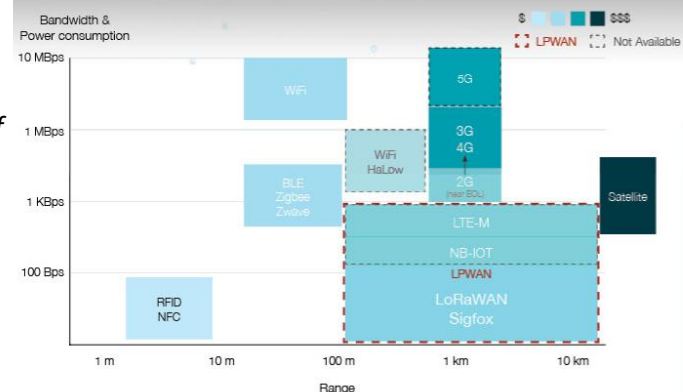


"Roughly every decade a new, lower priced computer class forms based on a new programming platform, network, and interface resulting in new usage and the establishment of a new industry."

- Gordon Bell [1972,2008]



LPWAN COMPLEMENTS EXISTING CONNECTIVITY TECHNOLOGIES



MAIN IOT USES

- RFID & NFC
Industrial and retail short range applications
- WiFi & BLE
Consumer applications
- 3G & 4G
Data intensive and expensive equipment (vehicles, cameras,...)
- Satellite
Remote applications

Figure 1 illustrates Bell's Law and how less hardware evolves every decade to create new kinds or classes of computers every decade at a new price level resulting in new industries with peripherals and applications.

Motivazione per il corso

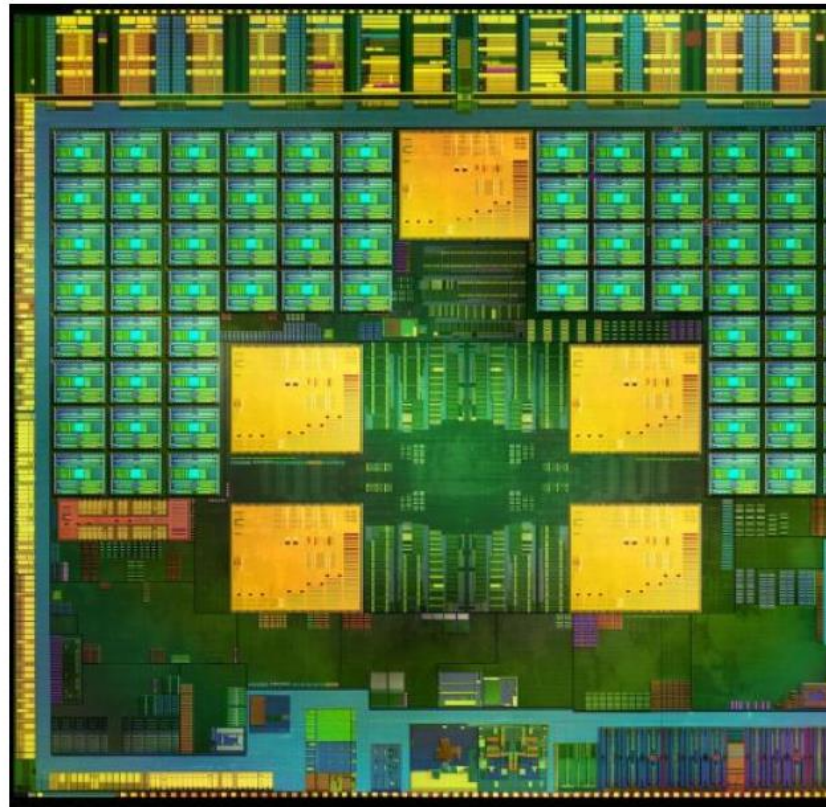
- **In maniera analoga a quanto avvenuto ai sistemi HPC tradizionali**
 - Elevato parallelismo
 - Eterogeneità
 - Compute clusters
 - ...ma su un singolo chip

(embedded)
high performance computing

Motivazione per il corso

Che succede oggi?

- I progettisti HW creano processori con un numero elevato di core
- Risultato:
 - l'hardware parallelo è ovunque
 - il software parallelo è raro
- La sfida
 - Il software parallelo deve diventare diffuso come l'hardware parallelo



NVidia Tegra 4 SoC

Motivazione per il corso

- I sistemi di calcolo sono pervasivi nella nostra vita quotidiana
 - Le competenze in questo campo sono strategiche in un mercato dominato da questi sistemi
- I sistemi embedded sono evoluti, come le loro controparti HPC tradizionali, sistemi eterogenei e massivamente paralleli.
- La sfida ultima è sullo sviluppatore di applicazioni, che deve sapere come estrarre il massimo dall'hardware.

Struttura del corso

Cosa imparerete in questo corso?

1. Architetture

- CPU parallele e organizzazione della memoria
- Acceleratori programmabili (iGPU)
- Logica programmabile (FPGA)

Struttura del corso

2. Programmazione

- *Teoria*: code profiling, partitioning, performance
- *Practica*:

Multicore
Parallel SW in OpenMP

Performance mediante
parallelism e acceleratori

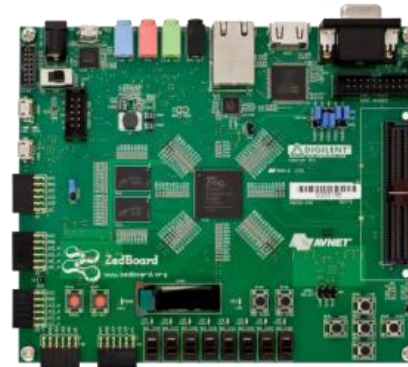
GPU Parallel SW in CUDA

FPGA Systems
HLS-based design

Performance mediante
accelerazione hardware



NVIDIA
Tegra



Xilinx Zynq

Struttura del corso

3. Uno sguardo da vicino alla compilazione e ottimizzazione del codice per i sistemi High-Performance Embedded

- Flusso di compilazione
- Struttura di un compilatore
- Ottimizzazione del codice
- Problematiche di performance



Organizzazione

- **Web:**

<http://algo.eng.unimo.it/people/andrea/Didattica/HPC/index.html>

- **Docenti:**

- Prof. **Andrea Marongiu** andrea.marongiu@unimore.it - Concetti teorici su architetture e parallelismo. Introduzione a tutti i moduli. Compilazione for sistemi High-Performance Embedded.
- Dr. **Paolo Burgio** paolo.burgio@unimore.it - OpenMP e CUDA, teoria e pratica.
- Dr. **Gianluca Brilli** gianluca.brilli@unimore.it – Esercitazioni HLS.

- **Lezioni:**

- Frontali con slides – teoria ed esempi
- Esercitazioni e esercizi d'esame risolti

Organizzazione

- **Esame**
 - Prova scritta (tre sezioni di domande a risposta multipla/singola/aperta, una per ogni modulo del corso)
 - Prova orale (successiva allo scritto, voto finale media delle prove)
 - Progetto (sostituisce la prova orale)
- **Prerequisiti (!!)**
 - Architettura dei calcolatori
 - Programmazione C
 - Concetti di base di Sistemi Operativi

Progetti e Tesi 😊



We are always looking for **motivated, hard-working and skilled students** for exciting:

- **projects** for course examination
 - Replace oral exam, grade is averaged with written exam
 - Will be presented later in the course
- **master theses**
 - + preparatory activity
 - follow up to a project
- **master theses** in collaboration with foreign institutions
 - depending on availability