Goals

- Techniques for the automatic design of embedded systems:
  - starting from their specification throughout:
    - validation / verification
    - automatic synthesis
    - testing
- This lecture is focused on:
  - most important design languages
  - most evolved tools for their manipulation

Embedded Systems: Where?

- From computer (‘60-‘80):
  - General purpose systems for solution of general problems
- To digital control systems (‘80-‘90):
  - Systems dedicated to control and automation
- To distributed systems (‘90-‘00):
  - General purpose systems and/or dedicated systems cooperating through the network
- To embedded systems (‘00-):
  - Distributed systems integrated in non-computing objects and in the environment
- To cyber-physical systems (‘10-):
  - Embedded systems integrated with physical processes

ES: Historical perspective

- First computers in 1940’s were all Embedded Systems:
  - not showing the today characteristics, but devoted to the particular application of being programmable computers and embedded into a…
- The Apollo Guidance Computer is considered the world’s first modern Embedded System:
  - small size for a tremendous computational power devoted to guide Apollo
- Mass production of Embedded Systems:
  - 1961 with the Autonetics D-17

ES: History

- First comp Systems:
  - not show to the pan compute
- The Apolc the world’s
  - small siz devoted
- Mass prod
  - 1961 with
- No stop…

ES Market

- World Embedded Systems Revenue
  - 2004
  - 2009
  - 2010

- Embedded software
  - Embedded IC
  - Embedded board
  - Total Revenue

- Application

- Automotive
- Defense
- Industrial
- Others

- Enabling technologies

- MEMS
- RFID
- Others
Expected to increase from $92.0 billion in 2008 to $112.5 billion by the end of 2013:
- Compound annual growth rate (CAGR) of 4.1%
- Embedded hardware from $89.8 billion in 2008 to $109.6 billion in 2013
- Embedded software from $2.2 billion in 2008 to $2.9 billion in 2013, for a CAGR of 5.6%.

How Relevant (I)

• AngeL, Vertigo, Coconut, C4C, Complex, SMAC, Contrex
• 2 progetti europei in FP6
  - ANGEL (mobile gateway for sensors network)
  - VERTIGO (HW formal verification)
• 5 progetti europei in FP7
  - COCONUT (embedded systems design and verification)
  - C4C (control for coordination of distributed systems)
  - COMPLEX (platform-based design space exploration)
  - SMAC (smart systems design)
  - CONTREX (mixed-criticality systems)

How Relevant (II)

• & progetti europei completati e attivi:
  - AngeL, Vertigo, Coconut, C4C, Complex, SMAC, Contrex
• 2 progetti europei in FP6
  - ANGEL (mobile gateway for sensors network)
• 5 progetti europei in FP7
  - COCONUT (embedded systems design and verification)
• SW design skills
  - lots of languages continuously extending
• HW architecture alternatives
  - for a correct HW/SW trade-off
• SW design skills
  - lots of languages continuously extending
• HW/SW interaction mechanisms
  - O.S., MW, H/S for efficient SW development
• Network infrastructure
  - all ES are now networked embedded systems
• Computation effort estimation
  - Theory is important when used in practice
• Join 3C: computation, control & communication

ES Market: trend

ES: How to design?

- We cannot design embedded systems like general purpose systems
- Different design constraints, different goals
- Embedded design is about the system, not about the computer
- E.g.
  - In general purpose computing, design often focuses on building the fastest CPU
  - In embedded systems the CPU simply exists as a way to implement control algorithms communicating with sensors and actuators

ES: Designer knowledge

- SW design skills
  - lots of languages continuously extending
- HW/SW interaction mechanisms
  - O.S., MW, H/S for efficient SW development
- Network infrastructure
  - all ES are now networked embedded systems
- Computation effort estimation
  - Theory is important when used in practice
- Join 3C: computation, control & communication

ES: Design constraints

- Size and weight
  - Hand-held electronics
  - Weight costs money in transportation
  - Human body cannot eat desktops
- Power
  - Buttery power instead of AC
- Harsh environment
  - Power fluctuation, RF interferences, heat, vibration, water, ...
- Safety critical and real time operations
- Low costs

ES: How to design?
Course Structure

- 34 lectures:
  - 32 theory hours
  - 22 lectures
  - 24 practical hours
  - 12 lectures
- People:
  - Franco Fummi (theory)
  - Michele Lora (laboratory class)
  - … for practical elaborations

6 credits

Modalità di Esame (I)

- Teoria + lab. + opzioni:
  - teoria
    - scritto con votazione /30
  - relazione laboratorio
    - +3 punti max
    - (orale) +3
- Regole generali:
  - elaborato dura 1 anno accademico
  - consegna in date stabilite

Modalità di Esame (II)

- Alternative:
  - Elaborato personale
    - stage aziendale
    - tesi
  - Teoria
    - no way :-)
- Design&Reuse:
  - laboratorio di Informatica (ordinamento 509/99)
  - tesi
  - stage pre-tesi

Pre/post Condizioni

- Precedenze Indispensabili:
  - Architettura degli Elaboratori
  - Programmazione
  - Linguaggi ...
  - Sistemi (Metodi di specifica)
- Fondamentale per
  - Curriculum sistemi embedded (magistrale in Ingegneria)
  - Sistemi operativi avanzati, Architetture avanzate, Software per Sistemi Embedded, Sistemi Embedded Multimediali, Sistemi Embedded di Reti...

2014/2015 news

- Smart devices:
  - The Open Source Test Case (SMAC project)
- Laboratorio Ciberfisico:
  - Secondo piano CV2

Detailed Program

<table>
<thead>
<tr>
<th>Date</th>
<th>NO</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-Oct</td>
<td></td>
<td>Course introduction; Embedded systems modeling</td>
</tr>
<tr>
<td>8-Oct</td>
<td>2</td>
<td>Embedded systems modeling II; SystemC-based design</td>
</tr>
<tr>
<td>10-Oct</td>
<td></td>
<td>NO</td>
</tr>
<tr>
<td>15-Oct</td>
<td>2</td>
<td>SystemC-based design II; SystemC-based design III</td>
</tr>
<tr>
<td>17-Oct</td>
<td>3</td>
<td>Platform-based design; Transactional-based design; TLM 2.0 standard</td>
</tr>
<tr>
<td>22-Oct</td>
<td>2</td>
<td>TLM 2.0 standard II; SystemC/AMS support</td>
</tr>
<tr>
<td>24-Oct</td>
<td>2</td>
<td>SystemC modeling at RTL</td>
</tr>
<tr>
<td>29-Oct</td>
<td>2</td>
<td>SystemC compilation/execution/debugging</td>
</tr>
<tr>
<td>31-Oct</td>
<td>2</td>
<td>SystemC timing evolution</td>
</tr>
<tr>
<td>5-Nov</td>
<td>2</td>
<td>SystemC modeling at TLM</td>
</tr>
<tr>
<td>7-Nov</td>
<td>3</td>
<td>High-level synthesis (HLS): scheduling; High-level synthesis: allocation</td>
</tr>
<tr>
<td>12-Nov</td>
<td>2</td>
<td>Automatic synthesis from TLM</td>
</tr>
<tr>
<td>14-Nov</td>
<td>3</td>
<td>Software embedded synthesis; Model-based design (MBD) of embedded software; HMI design</td>
</tr>
<tr>
<td>19-Nov</td>
<td>2</td>
<td>SystemC / AMS</td>
</tr>
<tr>
<td>21-Nov</td>
<td></td>
<td>intermediate exam</td>
</tr>
<tr>
<td>26-Nov</td>
<td>2</td>
<td>Mixed RTL/TLM/AMS SystemC</td>
</tr>
<tr>
<td>28-Nov</td>
<td>3</td>
<td>VHDL introduction; VHDL syntax</td>
</tr>
<tr>
<td>3-Dec</td>
<td>2</td>
<td>Platform, testbench and device driver (OSTC)</td>
</tr>
<tr>
<td>5-Dec</td>
<td>3</td>
<td>VHDL modeling; VHDL timing simulation</td>
</tr>
<tr>
<td>10-Dec</td>
<td>2</td>
<td>Embedded software: radCASE</td>
</tr>
<tr>
<td>12-Dec</td>
<td>3</td>
<td>VHDL timing simulation II; VHDL synthesis</td>
</tr>
<tr>
<td>17-Dec</td>
<td>2</td>
<td>VHDL modeling at RTL</td>
</tr>
<tr>
<td>19-Dec</td>
<td></td>
<td>Cyber-physical systems: bioaspects + interfaces</td>
</tr>
<tr>
<td>7-Jan</td>
<td>2</td>
<td>VHDL timing simulation</td>
</tr>
<tr>
<td>9-Jan</td>
<td>3</td>
<td>Networked embedded systems (NES); Middleware for embedded systems</td>
</tr>
<tr>
<td>14-Jan</td>
<td>2</td>
<td>Automatic synthesis from RTL VHDL</td>
</tr>
<tr>
<td>16-Jan</td>
<td>2</td>
<td>Introduction to embedded systems verification; Introduction to embedded systems testing</td>
</tr>
<tr>
<td>21-Jan</td>
<td></td>
<td>NO</td>
</tr>
<tr>
<td>30-Jan</td>
<td>2</td>
<td>final exam</td>
</tr>
</tbody>
</table>

Credits

6,0, 4,0, 2,0
Topics (theory)

• Specification:
  – Embedded systems modeling
  – SystemC-based design
  – TLM design introduction
  – TLM 2.0 standard
  – VHDL modeling
  – VHDL syntax
  – Networked embedded systems (NES)

• HW synthesis:
  – Introduction to TLM design
  – High-level simulation
  – Automatic VHDL synthesis

• SW synthesis:
  – Embedded software generation
  – Automatic device driver generation
  – Middleware for embedded systems
  – Model-based design
  – HW design

• Verification & testing:
  – Introduction to verification
  – Introduction to testing
  – VHDL timing simulation
  – RTL-TLM mixed simulation
  – Embedded software verification

Topics (lab.)

• Specification:
  – Compiling / executing /debugging SystemC
  – Modeling
  – SystemC
  – TLM design introduction
  – VHDL modeling
  – VHDL syntax
  – Networked embedded systems (NES)

• HW synthesis:
  – Automatic synthesis from TLM
  – VHDL modeling at RT
  – Automatic synthesis from RTL

• Software synthesis:
  – Testbench and device driver
  – Model based design
  – HMI design

Teaching supports (I)

• Course web page
  – Detailed program
  – Complete program

• E-learning web page
  – Slides
  – Laboratory instructions
  – Questions/answers

• Seminars
  – Indications during the course

Teaching supports (II)

• Theory slides:
  – Course Introduction
  – Embedded Systems Modeling
  – SystemC-Based Design Flow
  – Platform Based Design
  – Smart device SystemC/AMS
  – High Level Synthesis
  – Embedded Software
  – Model Based Design

• Theory slides:
  – VHDL Design Introduction
  – VHDL Syntax
  – VHDL Specification
  – VHDL Simulation
  – VHDL Synthesis
  – NES Design
  – Embedded HW
  – Verification And Testing

More information

http://www.di.univr.it/~fummi

For the stronger ...

Tuesday 8:30 – 10:30

In the corridors... running
For the strongest…

michele.lora@univr.it

Monday
10.00 – 11.00

On the e-learning

7048