
Testi del Syllabus

Docente

Matricola:

Anno offerta: **2013/2014**

Insegnamento: **1002548 - SISTEMI EMBEDDED**

Corso di studio: **5015 - INGEGNERIA INFORMATICA**

Anno regolamento: **2013**

CFU: **6**

Settore: **ING-INF/05**

Tipo attività: **B - Caratterizzante**

Partizione studenti: **-**

Anno corso: **1**

Periodo: **II° semestre**

Sede: **SEDE DIDATTICA DI PARMA**

Tipo testo	Testo
Lingua insegnamento	Italiano
Contenuti	<ul style="list-style-type: none">- Introduzione alle caratteristiche di un sistema embedded- Architetture hardware per Sistemi Embedded- Architetture software per Sistemi Embedded- Analisi e ottimizzazione di architetture miste Hw/Sw- Casi di Studio
Testi di riferimento	<ul style="list-style-type: none">• Carlo Brandolese, William Fornaciari, Sistemi embedded, Editore: Pearson, Anno edizione: 2007, ISBN: 9788871923420
Obiettivi formativi	<p>L'obiettivo del corso è fornire allo studente le competenze e le metodologie necessarie per la progettazione di sistemi dedicati ad applicazioni specifiche. La presenza di tali sistemi, spesso "invisibili" rispetto ai sistemi general-purpose come i PC, è il costante aumento e copre all'incirca i due terzi del mercato dei sistemi basati su processore. Esempi di tali sistemi sono le apparecchiature biomedicali, centraline e sistemi per automotive, apparati per applicazioni telecom, sistemi di controllo industriale, set-top-box e buona parte dell'elettronica di consumo.</p> <p>Il progetto di tali sistemi richiede l'utilizzo coordinato di competenze e tecnologie spesso trasversali a quanto presentato nei vari corsi di laurea, al fine di identificare la soluzione migliore sotto un notevole insieme di aspetti quali la flessibilità, la standardizzazione, il costo, le dimensioni, la potenza dissipata e le prestazioni, gli aspetti metodologici legati al flusso della progettazione, ecc. .</p> <p>Il contenuto del corso considera tali esigenze sistematizzando dapprima le competenze inerenti le architetture hardware, di comunicazione e software di base utilizzabili per la realizzazione di sistemi embedded, con riferimento agli standard esistenti e alle peculiarità del sistema da realizzare. Successivamente viene mostrato come gestire un flusso di progettazione che porti ad una soluzione "ottimale" sia sotto il profilo dei vincoli tradizionali di progetto (come ad es. la potenza dissipata o le prestazioni), sia considerando l'impatto che le soluzioni tecnologiche adottate hanno sotto il profilo del flusso di progettazione (es. costo o tempo di sviluppo)</p> <p>Gli approfondimenti relativi allo sviluppo del software di sistema e real time per applicazioni embedded è l'oggetto principale del corso di Sistemi Operativi Real Time, che si sviluppa in parallelo e complementa il corso di Sistemi Embedded.</p> <p>Per aggiungere incisività alle materie trattate è prevista la trattazione di alcuni casi di studio rappresentativi dei vari ambiti applicativi.</p> <p>Durante il corso saranno proposti stage aziendali e temi da approfondire come tesi di laurea</p>
Metodi didattici	Lezioni frontali in aula con approfondimenti monografici e casi di studio. Agli studenti verrà proposto di sviluppare, sulla base dei propri interessi e competenze, una piccola ricerca monografica o un progetto sperimentale.
Modalità di verifica dell'apprendimento	Due prove scritte intermedie durante il corso oppure una prova scritta complessiva. Presentazione orale della ricerca svolta o del progetto sviluppato
Programma esteso	<ul style="list-style-type: none">* Introduzione- Caratteristiche di un sistema embedded- Requisiti e vincoli dei vari ambiti applicativi- I trend del mercato e della progettazione

Tipo testo

Testo

- Flussi di sviluppo Hw/Sw
- Problematiche di ottimizzazione del consumo energetico
- * Architetture hardware per Sistemi Embedded
 - Caratteristiche degli esecutori del software: processori General-Purpose, Processori dedicati, DSP, Network processors, Microcontrollori, Sistemi Multiprocessore e Clusters, Architetture multi-many core.
 - Architetture di comunicazione: i bus standard, Network-on-Chip (NoC), i bus di campo, interfacce wireless. Esempi.
 - Esecutori hardware: caratteristiche degli Application Specific IC e del loro flusso di progettazione, sistemi per il fast prototyping, sistemi riconfigurabili, esempi di componentistica off-the-shelf, wireless sensor networks.
- * Architetture software per Sistemi Embedded
 - Livelli di astrazione: assembler, sorgente, libreria, software di base, middleware
 - Strumenti ed ambienti di sviluppo del software per applicazioni embedded
 - Generalità sui sistemi operativi real-time: caratteristiche generali, lo scheduler, la configurazione, gestione dinamica delle risorse per
 - Analisi del software embedded: formalismi e strumenti di analisi/profiling del codice, ottimizzazione energetica
 - Esempi: estensioni RT di Linux, Android, sistemi operativi per reti di sensori
- * Analisi e ottimizzazione di architetture miste Hw/Sw
 - L'analisi dei vincoli di progetto e la scelta dell'architettura realizzativa.
 - Problematiche di stime delle prestazioni/potenza dissipata ai vari livelli di astrazione.
 - Problematiche di ottimizzazione delle prestazioni/potenza/costo ai vari livelli di astrazione.
 - Analisi e modellazione di problematiche thermal e di affidabilità con enfasi verso architetture multi-core con comunicazione basata su NoC
 - Gestione e pianificazione della progettazione di sistemi embedded: modelli per il flusso di sviluppo della realizzazione di sistemi, design-for-resuse e problematiche di stime per hw e sw, problemi di project management.
- * Casi di Studio
 - Esempi di analisi, scelte realizzative e conduzione delle varie fasi di un progetto. Si considerano esempi legati allo sviluppo di applicazioni con differente granularità, come le reti di sensori, i cluster, i sistemi multiprocessore e SoC, sistemi basati su RFID, ponendo sempre in evidenza i vari trade-off da affrontare.

Testi in inglese

Tipo testo	Testo
Lingua insegnamento	Italian
Contenuti	<ul style="list-style-type: none">- Features of an embedded system- Hardware Architectures for Embedded Systems- Software Architecture for Embedded Systems- Analysis and optimization of mixed Hw / Sw architectures- Case Studies
Testi di riferimento	<ul style="list-style-type: none">• Carlo Brandolese, William Fornaciari, Sistemi embedded, Editore: Pearson, Anno edizione: 2007, ISBN: 9788871923420
Obiettivi formativi	<p>The aim of the course is to provide students with the skills and methodologies necessary for the design of systems dedicated to specific applications. The presence of such systems, often "invisible" compared to general-purpose systems such as PCs, is increasing and covers approximately two-thirds of the market of processor-based systems. Examples of such systems are the biomedical equipment units, the automotive systems, equipment for telecom, industrial control systems, set-top-box and a large part of consumer electronics.</p> <p>The design of such systems requires coordinated use of different skills and technologies in order to identify the best solution under a large variety of aspects such as flexibility, standardization, cost, size, power dissipation and performance, the methodological aspects related to the flow of the design, and so on.</p> <p>The content of the course considers these needs first systematizing the skills on hardware architectures, communication and basic software used for the realization of embedded systems, with reference to existing standards and to the peculiarities of the system to be designed. It is then shown how to run a design flow leading to an "optimal" solution both in terms of the constraints of traditional project (such as dissipated power or performance), and considering the impact that technological solutions have under the profile of the design flow (eg. cost or development time)</p> <p>The insights related to software development of system for real-time embedded applications is the main object of the course of Real Time Operating Systems, which develops in parallel and complement the course in Embedded Systems.</p> <p>The course will be offered job placements and topics to be addressed as thesis</p>
Metodi didattici	Lectures with in-depth studies and case studies. Students will be asked to develop, on the basis of their interests and skills, a little research or an experimental project.
Modalità di verifica dell'apprendimento	Two written exams during the course or a written overall test. Oral presentation of the research conducted or of the project developed.
Programma esteso	<ul style="list-style-type: none">* Introduction<ul style="list-style-type: none">- Features of an embedded system- Requirements and constraints of the various application fields- The trend of the market and of the design- Development workflows (Hw / Sw)- Problems of energy consumption optimization* Hardware Architectures for Embedded Systems<ul style="list-style-type: none">- Features of the softwareexecutors : general-purpose processors, dedicated processors, DSP, Network Processors, Microcontrollers,

Tipo testo

Testo

Multiprocessor Systems and Clusters, many multi-core architectures.

- Communication Architectures: the standard buses, the Network-on-Chip (NoC), field buses, wireless interfaces. Examples.
- Hardware: characteristics of Application Specific IC and their design flow, systems for fast prototyping, reconfigurable systems, examples of off-the-shelf components, wireless sensor networks.

* Software Architecture for Embedded Systems

- Levels of abstraction: assembler, source, library, basic software, middleware
- Tools and environments for software development for embedded applications
- General information on real-time operating systems: general characteristics, the scheduler, the configuration, dynamic resource management
- Analysis of embedded software: formalisms and analysis tools / code profiling, energy optimization
- Examples: RT extensions for Linux, Android, operating systems for sensor networks

* Analysis and optimization of mixed architectures Hw / Sw

- The analysis of the design constraints and the choice of the final architecture.
- Estimates of performance/power_loss at different levels of abstraction.
- Optimization of performance/power/cost at various levels of abstraction.
- Analysis and modeling of thermal and reliability issues with an emphasis towards multi-core architectures with communication-based NoC
- Management and planning of the design of embedded systems: models for the flow of development of the implementation of systems, design-for-reuse, problems of project management.

* Case Studies

- Examples of analysis and realization choices at the various stages of a project. We consider examples related to the development of applications with different granularity, such as sensor networks, clusters, multiprocessor systems and SoC systems based on RFID, always putting out the various trade-offs to be addressed.