UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

COURSE SYLLABUS

Embedded Systems Architectures and Design

2223-1-F9102Q012

Obiettivi

L'obiettivo di questo corso è fornire agli studenti una comprensione avanzata dei principali aspetti dei Sistemi Embedded e Cyber-fisici (ECPS) con funzionalità AI. Alla fine del corso gli studenti conosceranno i principali concetti relativi agli ECPS, i loro più rilevanti campi applicativi, i principali requisiti per gli ECPS nei diversi campi applicativi, le tecnologie attualmente disponibili per costruire ECPS con funzionalità AI. Gli studenti saranno in grado di progettare applicazioni embedded con funzionalità AI che soddisfino opportuni requisiti real-time, di consumo di memoria e di energia. Le conoscenze acquisite saranno applicate realizzando nel corso delle sessioni di laboratorio alcuni semplici ECPS di esempio.

Contenuti sintetici

I sistemi embedded e cyber-fisici (ECPS) nel loro contesto. La struttura degli ECPS: unità di elaborazione, memorie, sistemi di comunicazione, sensori, attuatori. Computazioni AI negli ECPS. Introduzione allo scheduling real-time. Progettare ECPS con vincoli real-time, di memoria ed energetici.

Programma esteso

I sistemi embedded e cyber-fisici (ECPS) nel loro contesto:

- Caratteristiche e tipologie di ECPS; Internet of Things (IoT), Wireless Sensor Networks (WSN); domini applicativi; valore di mercato e diffusione.
- Principali requisiti: temporali, di affidabilità, di efficienza; requisiti nei diversi domini applicativi; standards.

La struttura degli ECPS:

- Unità di elaborazione: CPU e microcontrollori, DSP e GPU, ASIC, logiche programmabili (FPGA). Acceleratori AI per sistemi embedded.
- Memorie. Sistemi di comunicazione: GPIO, bus seriali sincroni e asincroni.
- Modelli di sensori e attuatori: modelli affini, saturazione, distorsione armonica, range dinamico. Convertitori analogico-digitale e digitale-analogico.

Computazioni AI negli ECPS:

- Introduzione a Python e Tensor Flow.
- Cloud, edge ed endpoint AI. Introduzione a Tiny Machine Learning (TinyML).

Introduzione allo scheduling real-time:

- Definizione di sistema real-time e concetti fondamentali dello scheduling real-time: task e job, task periodici, aperiodici e sporadici, metriche, ottimalità.
- Scheduling per task aperiodici e periodici. Scheduling per insiemi di task eterogenei con task sporadici.
- Coordinazione tra tasks che condividono risorse: sezioni critiche e locking; inversione di priorità; protocolli di accesso alle risorse.

Progettare ECPS con vincoli real-time, di memoria ed energetici:

- Interazione con le periferiche in polling e guidata dagli interrupt.
- Architetture software: non preemptive (“superloop”) e preemptive (sistema operativo real-time).
- Progettare sistemi real-time con macchine a stati gerarchiche.
- Progetto software con vincoli di occupazione di memoria e di consumo energetico.

Prerequisiti

Conoscenze di base di programmazione con un linguaggio procedurale e uno orientato agli oggetti (un’introduzione di base ai linguaggi C e Python sarà fornita durante il corso); conoscenza di base dei principi dei sistemi operativi (processi, thread, scheudling) e dell’architettura degli elaboratori; conoscenza di base dei principi di analisi dei requisiti software e della progettazione del software con UML; conoscenza di base dei principi del machine learning, come insegnati nel corso Advanced Foundations of Artificial Intelligence.

Modalità didattica

Lezioni in aula e attività pratica di laboratorio consistente nello sviluppo di una serie di project works in piccoli gruppi. La presenza alle lezioni e ai laboratori è altamente raccomandata.

Materiale didattico

Gli argomenti che vengono insegnati nel corso non vengono discussi in maniera comprensiva in alcun libro: corrispondentemente, il corso non ha un libro di testo. Il materiale didattico verrà fornito durante il corso, e consisterà in una serie di dispense, slides, articoli di ricerca, e una selezione di capitoli tratti da libri tra i quali:

E. A. Lee, S.A. Seshia. Introduction to Embedded Systems: A Cyber-Physical Approach. Seconda edizione, MIT

Press, 2017.
G. C. Buttazzo. Hard Real-Time Computing Systems, Predictable Scheduling Algorithms and Applications. Terza edizione. Springer, 2011.
P. Warden, D. Situnayake. TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers. O'Reilly, 2019.

Periodo di erogazione dell'insegnamento

Secondo semestre.

Modalità di verifica del profitto e valutazione

Prova scritta con domande aperte, problemi ed esercizi; relazione scritta sull'attività di laboratorio.

Orario di ricevimento

Su appuntamento (inviare un email al docente per concordare una data e un orario per l'incontro).

Sustainable Development Goals