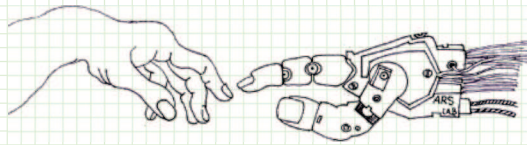


Guida rapida all'insegnamento

Sistemi dedicati

Docente: Giuseppe Scollo

Università di Catania
Dipartimento di Matematica e Informatica
Corso di Laurea Magistrale in Informatica, AA 2016-17



Indice

1. Guida rapida all'insegnamento
2. Obiettivi formativi
3. Organizzazione dell'insegnamento
4. Attività formative
5. Modalità di valutazione
6. Programma delle lezioni
7. Programma delle esercitazioni
8. Testi consigliati
9. Testi per consultazione
10. Laboratorio
11. Interazioni

Acquisizione e sviluppo della capacità di:

- modellazione, progettazione e realizzazione ottimale di sistemi di elaborazione dedicati a una specifica applicazione
- uso di strumenti hardware e software, quali piattaforme di sviluppo di sistemi dedicati, per progettare e realizzare sistemi dedicati ad applicazioni con alte prestazioni

Organizzazione dell'insegnamento

L'organizzazione dell'insegnamento prevede 24 ore di lezioni e 24 ore di esercitazioni di laboratorio di codesign (aula 24, martedì e giovedì, h. 09:00-11:00)

L'acquisizione di concetti e metodi nella disciplina è sostenuta da:

- frequenza delle lezioni ed esercitazioni
- partecipazione alle esperienze di laboratorio
- studio di testi di riferimento
- consultazione di altri testi e materiali didattici
- sperimentazione con strumenti software e piattaforme di sviluppo di sistemi dedicati
- elaborazione di soluzioni a problemi ed esercizi proposti: v. area **Esercizi**
- stesura di relazioni sulle esperienze di laboratorio: nella stessa area
- interazione con il docente, ricevimento settimanale:
G. Scillo: studio 325 (I blocco, II piano) [tel./cit.: [095738]3007
martedì e giovedì, h. 14:30-16:30
- interazione collaborativa con colleghi e docente

Attività formative

Lezioni: lo studio dei testi di riferimento pone le basi metodologiche per applicare efficacemente un approccio trasversale nelle tecnologie e unitario nel risultato:

progettare in modo versatile sistemi con cifra di merito ottimale dedicati a svolgere compiti ad alto grado di specializzazione

Esercizi: a partire dalla specifica delle funzionalità astratte del sistema, il primo problema che spesso si pone è quello di selezionare l'architettura in cui tradurle, per poi procedere alla sintesi di tutti i componenti: hardware, software e interfacce di comunicazione. Gli esercizi proposti affrontano le diverse parti di questo processo.

Laboratorio di codesign: è previsto l'uso di sistemi di sviluppo per la realizzazione di applicazioni embedded, spaziando dalla configurazione su board alla sintesi di componenti con FPGA, fino alla realizzazione dell'intero sistema su un singolo chip. La stesura di relazioni sulle esperienze di laboratorio può essere prodotta da collaborazioni di gruppo.

Seminari: a titolo sperimentale, alcune lezioni (circa 1/4 del totale) sono preparate e prodotte in forma seminariale da studenti; una esercitazione è dedicata alla pianificazione dei seminari. Nel periodo di esami è prevista una prova scritta di autovalutazione delle attività didattiche dell'insegnamento.

Modalità di valutazione

Esame orale, progetto (opzionale)

- obiettivi della valutazione
- esame orale:
 - valutazione del conseguimento degli obiettivi formativi
- progetto opzionale:
 - valutazione della maturità concettuale e scientifica nella pratica della disciplina
- colloqui orali
- verifica dell'apprendimento critico della disciplina
- valutazione del contributo a relazioni sulle esperienze di laboratorio
- eventuale colloquio (opzionale) individuale sull'uso di concetti e metodi della disciplina in un progetto originale di laboratorio, concordato con il docente, che può essere realizzato in collaborazione con altri studenti
- calendario di esami
 - prima sessione: 7 febbraio, 23 febbraio
 - seconda sessione: 22 giugno, 11 luglio
 - terza sessione: 12 settembre, 28 settembre
 - sessione straordinaria: 11 aprile, 13 dicembre

Il superamento dell'esame porta all'acquisizione di 6 crediti.

Programma delle lezioni

legenda: r = letture di riferimento, s = letture supplementari, rn.# = nota integrativa di riferimento, sn.# = nota integrativa supplementare #

1. Finalità e organizzazione dello studio. Introduzione al codesign di sistemi dedicati
L01: 11/10/2016, r: S.01(1.1.4-1.4.1.6); s: VG.01(1.1-1.4)
2. Architetture e progettazione di sistemi dedicati
L02: 18/10/2016, r: S.01(1.1.2-1.1.3.1.5.1.7); s: VG.01(1.5-1.6), BF.01
3. Modelli dataflow, flusso del controllo
L03: 08/11/2016, r: S.02; s: L.S.06(6.3), M.02(2.5), sn.3, sn.4
4. Realizzazioni software di modelli dataflow
L04: 15/11/2016, r: S.03(3.1), s: S.04
5. Sistemi sincroni come macchine a stati finiti con datapath (FSMD)
L05: 22/11/2016, r: S.05(5.3-5.4.3.5.6); s: S.05(5.7)
6. Microprogrammazione: architetture e controllo
L06: 29/11/2016, r: S.06(6.1-6.4); s: S.06(6.6-6.8)
7. Progetto e analisi di programmi per sistemi dedicati
L07: 06/12/2016, r: S.07(7.1.7.3); s: S.07(7.2.7.5), sn.6
8. Progettazione di System-on-Chip (SoC)
L08: 13/12/2016, r: S.08(8.1-8.3); s: S.08(8.4), R.01
9. Principi di comunicazione HW/SW
L09: 20/12/2016, r: S.09(9.1-9.4)
10. Interfacce di microprocessore
L10: 10/01/2017, r: S.11(11.1.1-11.1.5.11.2.0.11.3.0-11.3.1.11.3.3); s: S.11(11.1.6.11.2.1-11.2.2.11.3.2.11.3.4)
11. Interfacce hardware
L11: 17/01/2017, r: S.12(12.1-12.3.1.12.4); s: S.12(12.3.2)
12. Machine learning e FPGA
L12: 24/01/2017, seminario studenti

Programma delle esercitazioni

legenda: r = letture di riferimento, s = letture supplementari, rn.# = nota integrativa di riferimento, sn.# = nota integrativa supplementare #

1. Introduzione all'uso combinato di Gezel con un simulatore VHDL
E01: 13/10/2016, r: S.01(1.1.1), S.A(A.1), rn.1, rn.2; s: S.A(A.2)
2. Linguaggi di descrizione dell'hardware: Gezel, VHDL, Verilog, SystemC
E02: 20/10/2016, r: S.05(5.1-5.2), W.01, W.03-04; s: M.2(2.7), BF.aB, sn.1, sn.2
3. Introduzione al progetto di sistemi hardware con FPGA
E03: 03/11/2016, r: W.02, W.05-06, rn.3, rn.4
4. Esempi di reti combinatorie e sequenziali in VHDL
E04: 10/11/2016, r: W.20-21, W.24-25, W.27; s: Z.4(4.1-5), Z.6(6.1-5), rn.6(App.A)
5. Esempi di reti dataflow in Gezel e in VHDL
E05: 17/11/2016, r: S.03(3.2)
6. Esempi di FSMD in Gezel e in VHDL
E06: 24/11/2016, r: S.03(3.3), S.05(5.4.4-5.5), W.22
7. Esempio di progetto di un microprocessore in VHDL
E07: 01/12/2016, r: S.06(6.5); s: W.08(8.1-8.3), Z.7(7.3-7.5)
8. Esempi di analisi di programmi e uso di simulatori di processori
E08: 15/12/2016, r: S.07(7.4.7.6.2); s: S.07(7.6.1.7.6.3), sn.5, sn.6, sn.7, sn.8
9. Pianificazione di seminari degli studenti
E09: 20/12/2016
10. Sistemi di bus su chip, sviluppo di SoC con FPGA
E10: 12/01/2017, r: S.10(10.1), rn.5, rn.6; s: S.10(10.2-10.4)
11. Progetto di coprocessori e ASIP su sistema di sviluppo di SoC con FPGA
E11: 19/01/2017, r: rn.7, rn.8
12. Progetto su FPGA di coprocessori per machine learning
E12: 26/01/2017, progetto di laboratorio

Testi consigliati

Testi di riferimento

N.B. Le parti dei testi raccomandate sono indicate nel programma, per ciascuna lezione, nella forma abbreviata A.C(S), dove: A = iniziali degli autori del testo, C = capitolo, S = sezione/i

P.R. Schaumont: *A Practical Introduction to Hardware/Software Codesign*
2nd Edition. Springer (2012)

con correzioni al testo prodotte a partire dall'edizione 2013-2014 dell'insegnamento

P. Wilson *Design Recipes for FPGAs: Using Verilog and VHDL*
2nd Edition. Newnes, Elsevier (2015)

Note integrative di riferimento

1. *Quartus II Introduction Using VHDL Designs - For Quartus II 13.1*, Altera University Program, April 2014
2. *Using TimeQuest Timing Analyzer - For Quartus II 13.1*, Altera University Program, 2014
3. *Quartus II Introduction Using Schematic Designs - For Quartus II 13.1*, Altera University Program, 2014
4. *Using Library Modules in VHDL Designs - For Quartus II 13.1* Altera University Program, October 2013
5. *Introduction to the Altera Qsys System Integration Tool - For Quartus Prime 16.0*, Altera University Program, May 2016
6. *Making Qsys Components - For Quartus Prime 16.0*, Altera University Program, May 2016
7. *Using the SDRAM on Altera's DE1-SOC Board with VHDL Designs - For Quartus Prime 16.0*, Altera University Program, May 2016
8. *Nios II Custom Instruction User Guide*, UG-N2CSTNST, Altera Corp. (2015-11-02)

Testi per consultazione

Libri di testo

C. Brandolese, W. Fornaciari: *Sistemi embedded: sviluppo hardware e software per sistemi dedicati*
Pearson, Milano (2007)

E.A. Lee & S.A. Seshia: *Introduction to Embedded Systems - A Cyber-Physical Systems Approach*
2nd Ed., Version 2.0 (2015)

P. Marwedel: *Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems*
2nd Edition. Springer (2011)

C. Rowen: *Engineering the Complex SOC - Fast, Flexible Design with Configurable Processors*
Prentice-Hall (2004)

F. Vahid & T. Givargis: *Embedded System Design: A Unified Hardware/Software Introduction*, Wiley (2002)

F. Vahid, T. Givargis & B. Miller: *Programming Embedded Systems: An Introduction to Time-Oriented Programming*
Version 4.0. Uniworld (2012)

M. Wolf: *Computers as components: Principles of embedded computing system design*
3rd Edition, Morgan Kaufmann (2012)

M. Zwolinski: *Digital System Design With VHDL*, 2nd Edition, Pearson (2004)

Note integrative supplementari

1. F. Vahid (2006): *Digital Design*, Ch. 9, Hardware Description Languages (PDF slides)
2. D.J. Smith (1996): *VHDL & Verilog Compared & Contrasted*
3. Ghamarian et al. (2007): *Latency minimization for Synchronous Data Flow Graphs*
4. Stuijk et al. (2006): *Exploring trade-offs in buffer requirements and throughput constraints for Synchronous Dataflow Graphs*
5. *Introduction to the Altera Nios II Soft Processor*, Altera University Program, May 2016
6. *Introduction to the ARM® Processor Using Altera Toolchain*, Altera University Program, May 2016
7. *Altera Monitor Program Tutorial for Nios II*, Altera University Program, May 2016
8. *Altera Monitor Program Tutorial for ARM*, Altera University Program, May 2016

Le attività di laboratorio consistono di una serie di esperienze sui seguenti argomenti:

- codesign e cosimulazione in Gezel
- ottimizzazione di programmi nei linguaggi C e assembly
- programmazione FPGA in VHDL
- codesign, simulazione e valutazione delle prestazioni su sistemi di sviluppo SoC con FPGA

Forum, Moodle, Galileo: cosa va dove?

- Forum: discussioni di
 - organizzazione dell'insegnamento, avvisi, FAQ
 - problemi con l'uso di servizi on-line, strumenti software ecc.
 - discussioni di idee di progetti di sistemi dedicati
- Moodle (servizi ad accesso riservato):
 - accesso a materiali didattici di supporto
 - sviluppo di problemi ed esercizi proposti
 - discussioni di tematiche pertinenti a lezioni, esperienze di laboratorio e materiali didattici
 - collaborazione di gruppo, consegna di relazioni sulle esperienze di laboratorio
 - segnalazione e discussione di eventuali errori nei materiali didattici (possono valere punti bonus!)
- Galileo:
 - sviluppo di progetti di sistemi dedicati
 - documentazione e distribuzione dei risultati nel pubblico dominio