Quick learning guide

Dedicated systems

Teacher: Giuseppe Scollo

University of Catania
Department of Mathematics and Computer Science
Graduate Course in Computer Science, 2019-20

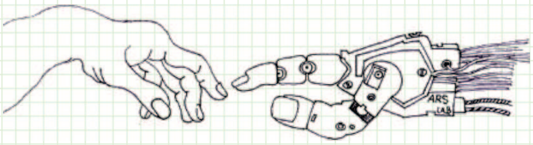

Table of Contents

Educational goals

To acquire and develop the following capabilities:

- model, design and optimal implementation of application-specific information processing systems
- use of harware and software tools, such as development platforms for embedded systems, to design and to implement embedded systems for high-performance applications

Course organization

The course is organized in two modules: 24-hour lecture and 24-hour codesign lab tutorials (lecture room 24, on mondays and wednesdays, 3-5 pm)

Acquisition of the subject concepts and methods is supported by:

- attendance to lectures and problem solving
- participation to lab experiences
- study of reference readings
- consultation of supplementary readings
- experience with software tools and development platforms for embedded systems
- working out solutions to proposed problems and exercises, see Test area
- writing reports on lab experiences, in the same area
- interaction with teacher, weekly schedule:

  G. Scollo: room 325 (I block, II floor) [phone]ext.: [095738]3007
  on mondays and wednesdays, 1-3 pm
- collaborative interaction with colleagues and teacher

Educational activities

Lectures: the study of the recommended readings sets the methodological grounds for effective application of a technology-transverse, result-unitary design approach:

> versatile design of systems with optimal figure of merit that are dedicated to perform highly specialized tasks

Exercises: starting from a specification of the abstract functionalities of the system, the first problem which is often faced with is to select an architecture wherein to map them out, in order to further proceed to the synthesis of all components: hardware, software, communication interfaces. The proposed exercises deal with the different parts of this process.

Codesign lab: it is envisaged the use of development boards and platforms to implement embedded applications, ranging from on-board configuration to FPGA-based synthesis of components, up to System-on-chip (SoC) implementation. Reports on lab experiences may be outcomes of collaborative group work.

Seminars: as an experimental feature, some lectures (max. 1/5 of the total) take the form of seminars that are prepared and delivered by students; planning of the seminars is carried out as part of a lab tutorial.

Exams and evaluation

Oral exam, project (optional)

- evaluation objectives
- oral exam:
  assessment of the achievement of educational goals
- optional project:
  assessment of conceptual and scientific maturity exhibited in the practice of the subject
- oral colloquies
- assessment of critical learning of the subject
- evaluation of individual contribution to lab experience reports
- (optional) colloquy on individual use of subject concepts and methods within an original lab project, previously agreed with the teacher, that may be the outcome of a collaborative group effort
- exam schedule

Exam success yields the acquisition of 6 credits.

Lecture Program

legenda: r = reference readings, s = supplementary readings, rn.# = reference lecture note, sn.# = supplementary lecture note #

1. Course goals and organization.   Introduction to dedicated systems codesign
   L01: 16/10/2019, r: S.01(1.1.4-1.4,1.1.6); s: VG.01(1.1-1.4), BF.01
2. Architectures and design process of dedicated systems
   L02: 22-23/10/2019, r: S.01(1.5,1.7), Z.01(1.1); s: VG.01(1.5-1.6)
3. Dataflow models, control flow
   L03: 28/10/2019, r: S.02; s: LS.06(6.3), M.02(2.5), sn.3, sn.4
4. Software implementations of dataflow models
   L04: 04-06/11/2019, r: S.03(3.1); s: S.04
5. Synchronous systems as finite state machines with datapath (FSMD)
   L05: 11/11/2019, r: S.05(5.3-5.4.3,5.6); s: S.05(5.7)
6. Microprogramming: architectures, control, microprogrammed interpreters
   L06: 18/11/2019, r: S.06(6.1-6.4); s: S.06(6.6-6.8)
7. Program design and analysis for dedicated systems
   L07: 25/11/2019, r: S.07(7.1,7.3); s: S.07(7.2,7.5), sn.7
8. System-on-Chip (SoC) design
   L08: 02/12/2019, r: S.08(8.1-8.3); s: S.08(8.4), R.01
9. HW/SW communication, on-chip bus systems
   L09: 09/12/2019, r: S.09(9.1-9.4), S.10(10.1); s: sn.11, S.10(10.2-10.4)
10. Microprocessor interfaces (student seminar)
    L10: 10/01/2020, r: S.11(11.1.1-11.1.5,11.2.0,11.3.0-11.3.1,11.3.3); s: S11(11.1.6,11.2.1-11.2.2,11.3.2,11.3.4)
11. Hardware interfaces
    L11: 13/01/2020, r: S.12(12.1-12.3.1,12.4); s: S.12(12.3.2)
12. Design of a memory-mapped multicore coprocessor
    L12: 20/01/2019, lab project

Lab Tutorial Program

legenda: r = reference readings, s = supplementary readings, rn.# = reference lecture note, sn.# = supplementary lecture note #

1. Introduction to the combined use of Gezel with a VHDL simulator
   E01: 22/10/2019, r: S.01(1.1.1), rn.1, rn.2; s: S.A(A.1)
2. Introduction to design of hardware systems using FPGA
   E02: 23-28/10/2019, r: Z.01(1.3), rn.3; s: rn.4
3. Hardware description languages: Gezel, VHDL, Verilog, SystemC
   E03: 30/10/2019, r: S.05(5.1-5.2), Z.03(3.1-3.7); s: BF.aB, Z.aB, M.2(2.7), sn.1, sn.2
4. Combinational network examples in VHDL
   E04: 04/11/2019, r: Z.04(4.1-4.6); s: rn.6(App.A)
5. Sequential network examples in VHDL, hardware implementation of dataflow models
   E05: 06-11/11/2019, r: Z.06(6.1-6.5.1), S.03(3.2); s: S.03(3.3)
6. FSMD examples in Gezel and in VHDL. Planning of student seminars
   E06: 13/11/2019, r: S.05(5.4.4-5.5); s: Z.07(7.1-7.2), sn.5, sn.6
7. Microprocessor design example in Gezel and VHDL
   E07: 22/11/2019, r: S.06(6.5); s: Z.07(7.3-7.5)
8. Program analysis tools and examples
   E08: 25/11/2019, r: S.07(7.4); s: S.07(7.5,7.6), sn.7, sn.8, sn.9, sn.10
9. Component development on FPGA
   E09: 06/12/2019, , r: rn.5, rn.6
10. SoC development on FPGA with application profiling
    E10: 13/12/2019, r: rn.7; s: sn.12
11. FPGA implementation of a memory-mapped coprocessor
    E11: 17/01/2020, r: lab project
12. FPGA implementation of a memory-mapped multicore coprocessor
    E12: 24/01/2020, lab project

Recommended readings

Reference textbooks

N.B. The recommended parts are displayed in the lecture program, for each lecture, in the short form *A.C(S)*, where: *A* = initials of textbook author(s), *C* = chapter, *S* = section(s)

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

with textbook corrections made starting from the 2013-2014 course edition

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

Reference lecture notes

1. *Quartus Prime Introduction Using VHDL Designs - For Quartus Prime 16.1*, Intel Corp. - FPGA University Program, November 2016
2. *Using TimeQuest Timing Analyzer - For Quartus Prime 16.1*, Intel Corp. - FPGA University Program, November 2016
3. *Quartus Prime Introduction Using Schematic Designs - For Quartus Prime 16.1*, Intel Corp. - FPGA University Program, November 2016
4. *Using Library Modules in VHDL Designs - For Quartus Prime 16.1*, Intel Corp. - FPGA University Program, November 2016
5. *Introduction to the Qsys System Integration Tool - For Quartus Prime 16.1*, Intel Corp. - FPGA University Program, November 2016
6. *Making Qsys Components - For Quartus Prime 16.1*, Intel Corp. - FPGA University Program, November 2016
7. *Performance Counter Unit Core*, Ch. 36 in: *Embedded Peripherals IP User Guide*, Intel Corp., UG-01085 | 2019.12.16

Supplementary readings

Textbooks

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., MIT Press (2017)

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

C. Rowen: *iEngineering 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)

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

Supplementary lecture notes

1. F. Vahid (2006): Digital Design, Chapter 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. C. Brandolese, *Introduzione al linguaggio VHDL*, Politecnico di Milano
6. *Note sul VHDL*, Corso di Architettura dei Sistemi Integrati
7. *Introduction to the ARM® Processor Using Intel FPGA Toolchain - For Quartus Prime 16.1*, Intel Corp. - FPGA University Program, November 2016
8. *Introduction to the Intel Nios II Soft Processor - For Quartus Prime 16.1*, Intel Corp. - FPGA University Program, November 2016
9. *Intel FPGA Monitor Program Tutorial for ARM - For Quartus Prime 16.1*, Intel Corp. - FPGA University Program, November 2016
10. *Intel FPGA Monitor Program Tutorial for Nios II - For Quartus Prime 16.1*, Intel Corp. - FPGA University Program, November 2016
11. *Avalon® Interface Specifications*, MNL-AVABUSREF, Intel Corp., 2019.10.08
12. *Profiling Nios II Systems*, AN-391-3.0, Altera Corp., July 2011

Laboratory

Lab activities consist of a series of experiences with the following topics:

- codesign and cosimulation in Gezel and VHDL
- program optimization in C and assembly languages
- FPGA programming in VHDL
- codesign, simulation and performance evaluation on SoC development boards with FPGA

Interaction

Forum, Moodle, Galileo: what goes where?

- Forum: discussions about
  - course organization, news, FAQ
  - problems with use of on-line services, software tools etc.
  - proposals of *ideas* of dedicated system projects
- Moodle (restricted access services):
  - access to educational support materials
  - development of proposed problems and exercises
  - discussions about topics relating to lectures, lab experiences and learning materials
  - group collaboration, delivery of lab experience reports
  - discussions about project proposals and their development
  - discovery and discussion of errors in educational materials (this may award *bonus points* !)
- Galileo:
  - development of hardware/software projects with documentation and dissemination of results in the public domain