Multimedia SoC Design

with Specialization on Application Acceleration with High-Level-Synthesis

Jiin Lai 賴瑾

Shao-Yi Chien 簡韶逸

Fall 2020

Graduate Institute of Electronics Engineering
National Taiwan University

Motivation (1)

- System-on-a-Chip (SoC): system or chip?
- System know-how forms a high entry barrier for chip designers
  - Related algorithm (ex: Machine learning/DSP/image processing/video processing/computer vision/communication algorithms…)
  - Architecture design
  - Chip design flow
  - Verification method
  - Computer architecture
  - Embedded system design
  - Embedded software development
  - …

Motivation (2)

- Need new design methodology to conquer the high complexity:
  - System level design and verification
  - Hardware/software partition
  - Hardware/software co-design
  - Hardware/software co-simulation
- New methodologies: Electronics System Level (ESL), hardware/software co-verification, high-level synthesis (HLS)…
- Important next step of Taiwan's IC design industry

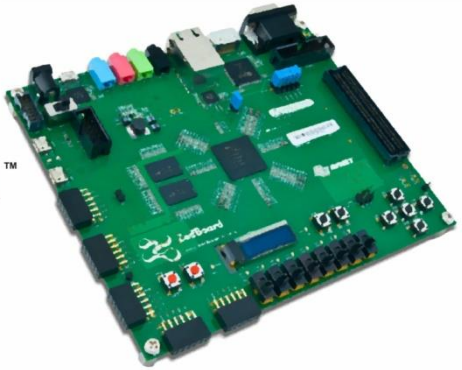
Introduction

- This course will introduce the concepts and design techniques of multimedia SoC design
- **The focus is on system design**, and we take multimedia system as an example
- The following topics will be covered
  - Architecture design
  - Embedded system design/SoC architecture design
  - Multimedia system design
  - HLS for FPGA
  - HLS for FPGA instance on cloud servers
- The course consists of
  - 1/2 Lecture
  - 1/2 Tutorial labs

Outline (1)

- System design concept
  - Introduction to SoC/ESL/Multimedia Systems
  - Platform-based design/Introduction to system design flow
  - Architecture design
  - Processors
  - Memory
  - Peripherals
  - Interfacing
  - Hardware/software partition
  - Multimedia hardware accelerators

Outline (2)



- High level synthesis
  - Platforms
    - Xilinx Vivado HLS
    - Pynq-Zedboard
    - AWS
  - Labs/tutorials/projects
    - Xilinx lab examples, application note, tutorials
    - Text book – pp4fpga
    - Rosetta benchmark program with open sources
    - Open source from various application area – finance, bioinformatics, ML, video, Database, Signal processing, Security …
  - Final projects
    - Real applications

Schedule

Week	Date	Lecture
1	9/18	Introduction/HLS Session 1
2	9/25	Introduction to SoC design/Architecture Design Basics
3	10/2	中秋節
4	10/9	國慶日連假
5	10/16	HLS Session 2
6	10/23	Processor
7	10/30	HLS Session 3
8	11/6	Bus/AMBA
9	11/13	HLS Session 4
10	11/20	Peripherals and Interfacing
11	11/27	HLS Session 5
12	12/4	Memory
13	12/11	HLS Session 6
14	12/18	Midterm
15	12/25	Hardware accelerator design
16	1/1	元旦
17	1/8	Hardware/software verification / Low-power design
18	1/15	No class
19	1/22	Final project presentation

HLS Sessions

session	Date	Suggest lecture title - self-paced	pdf	video	In-classs discussion topics	Assignment
1	18-Sep	Course Introduction	§	§		HLS flow
		Introduction PYNQ & Lab2	§	§		ug871 labs **ug871-[1:7]**
		Vitis OpenCL XRT and Lab3	§	§		Lab1: Tool Installation
		Introduction to FPGA				Lab2: PYNQ axi-m & stream
						Lab3: OpenCL/XRT
2	16-Oct	Kernal IO Interface	§	§	ug871 Labs ug871-[1:7]	Xlinx Training Lab **-xtrain-[1:13]**
		Introduction to High Level Synthesis	§	§	HLS, Vivado, Vitis usage experience	Xilinx HLS Coding Style
		FPGA - CLB	§	§	Lab1, Lab2, Lab3 sharing	Xilinx HLS Design - **xdesign-[1:15]**
		FPGA - Memory	§	§		
			§	§		
3	30-Oct	System Optimization - Host	§	§	Xlinx Training Lab -xtrain-[1:13]	Vitis Tutorial - **vitis-[1:7]**
		System Optimizatin - Kernel	§	§	Xilinx HLS Design - xdesign-[1:15]	UCSD Lab **ucsd-[1:5]**
		FPGA - DSP	§	§		Cornell - ECE5775 **cornell-[1:4]**
		FPGA - Interconnect	§	§		
4	13-Nov	Kernel Optimization - Area	§	§	Vitis Tutorial - vitis-[1:7]	**pp4fpga-[1:8]**
		Kernel Optimization - Latency	§	§	UCSD Lab ucsd-[1:5]	Xilinx HLx Examples **xhls-[1:18]**
		Kernel Optimization - Pipeline	§	§	Cornell - ECE5775 cornell-[1:4]	
5	27-Nov	Design Examples	§	§	pp4fpga-[1:8]	
		Application Cases	§	§	Xilinx HLx Examples xhls-[1:18]	Xilinx Application Notes **xapp-[1:13]**
6	11-Dec				All topics	Final Project
						Refer to project resource weight=10
	22-Jan	Final Project presentation				

Course Information (1)

- Classroom
  - EEII-143 ~~or PC room B (in EEII, for labs)~~
- Course Texts: Required
  - R. Kastner, J. Matai, and S. Neuendorffer, Parallel Programming for FPGAs, arXiv, 2018
  - Xilinx ug902
- Supplementary Materials: Optional (various references listed in lecture slides）
  - Reference Papers
  - Manual/Datasheets
  - Selected chapters from books

Course Information (2)

- Website:
  - http://media.ee.ntu.edu.tw/courses/msoc
  - https://cool.ntu.edu.tw/courses/3773
  - TA
    - 翁華揚 (BL-421) huayang@media.ee.ntu.edu.tw
    - 陳柏穎 (BL-421) byc@media.ee.ntu.edu.tw
    - Arthur Hsiao givenchy5168@gmail.com
- Prerequisite
  - Knowledge of C/C++
  - Basic concept of digital logic, computer architecture
  - Knowledge of basic algorithm, and data structures
  - Experiences with RTL design for ASIC or FPGA would be helpful, but not required

References

- 李昆忠、簡韶逸、鄺獻榮、邱瀝毅、郭致宏，電子系統層級設計教授，教育部顧問室「超大型積體電路與系統設計教育改進」計畫SLD聯盟。
- **ITRI/STC, Alan P. Su, *System Design with ESL*.**
- *Slides of System-Level Modeling for System-on-a-Chip Design Workshop*, http://lina.ee.ntu.edu.tw/SLD.htm
- **Website of Open SystemC Initiative (OSCI), http://www.accellera.org**
- CIC, CoWare *ConvergenSC training material*
- Documents of CoWare, **http://www.synopsys.com**
- ARM, http://www.arm.com
- **David C. Black and Jack Donovan, *SystemC: From the Ground Up*, 2nd Ed., Springer, 2009.**
- F. Ghenassia, *Transaction-Level Modeling with SystemC: TLM Concept and Applications for Embedded Systems*, Springer, 2005.
- J. Bhasker, *A SystemC Premier*, Star Galaxy Publishing, 2004.
- T. Grotker, S. Liao, G. Martin, and S. Swan, *System Design with SystemC*, Kluwar Academic Publishers, 2002.
- **B. Bailey, G. Martin, and A. Piziali, *ESL Design and Verification*, Morgan Kaufmann Publishers, 2007**
- W. Wolf, *Computers as Components: Principles of Embedded Computing System Design*, Morgan Kaufmann Publishers, 2001.
- F. Vahid and T. Givargis, *Embedded System Design: a Unified Hardware/Software Introduction*, John Wiley & Sons, 2002.
- P. Rashinkar, P. Paterson, and L. Singh, *System-on-a-Chip Verification: Methodology and Techniques*, Kluwar Academic Publishers, 2001.

Grading Policy

- High level synthesis 70%
  - There are three ways to earn credits
    - Turn-in Lab/Assignment
    - Give class Presentation
    - Work on Final project, presentation, and publication (github submit)
  - Final score is derived by normalizing the credits
- Mid-term exam 30%

Action Items

- Take this course with groups. 3 members in one group. Upper limit: 25 groups
- Please form your group and send the list to Prof. Chien
  - sychien@ntu.edu.tw, with the title “MSOC Group”
  - With name, email, affiliation (grade), and advisor
  - 3 members in each team
  - Send before 10:00AM 24 Sept.

Acceptance Priority

- GIEE, PhD
- GIEE, M2
- GIEE, M1
- Graduate students in related research groups
- Undergraduate students having projects with professors of GIEE
- Others