EEM870 Embedded System and Experiment
Lecture 1: SoC Design Overview

Wen-Yen Lin, Ph.D.
Department of Electrical Engineering
Chang Gung University
Email: wylin@mail.cgu.edu.tw
Feb. 2013
Course Overview about SoC Part

You may find similar contents if you have taken **EEM830 Introduction to SoC Design**

Objectives

- Understand What SoC is and what the differences between SoC and Embedded System
- System Design Methodology in SoC
  - Design Flow
  - Design Tools
Course Overview about SoC Part

Teaching Methods

- Class Lectures with ppt slides
- Lab Exercise ~ Design of an ARM926EJS embedded system
  - Platform Architect

Exam

- Basic Concepts on the materials taught in the class
What is SoC?

SoC: System-on-a-Chip or System-on-Chip

System:

- A system could be a computer, PDA, mobile phone, MP3, Digital Camera or customized for specific purposes
- The system may contain digital, analog, mix-signal and often radio-frequency functions

SoC refers to integrating all components into a single integrated circuit (chip)
What is SoC?

Structure of a System
- Microcontroller, Microprocessor or DSP cores.
- Memory Blocks – ROM, RAM, EEPROM, Flash
- Clock sources – Local Oscillator, PLL
- Power Circuits – Voltage Regulator, DC-DC converters, etc.
- Peripherals – Counter-timers, real-time timers, etc.
- External Interfaces – USB, Ethernet, UART, RS-232, etc.
- Analog Interfaces – ADCs and DACs
- Blocks are connected by either a proprietary or industry-standard bus.
- Embedded Software
SoC Architecture
Trend of System Design

- **Discrete Components On PCB**
- **Chip Sets On PCB**
- **System On Chip**

- **Design Complexity**
- **System Integrity**
Moore’s Law

In 1965, Gordon Moore prophesized that number of transistors doubled every 18 months. In 1975, it was modified to every 24 months.

Almost every measure of the capabilities of digital electronic devices is linked to Moore’s Law: processing speed, memory capacity, and even the number and size of pixels in digital cameras.
Moore’s Law

Number of transistors doubling every 18 months.

Number of transistors doubling every 24 months.

Year

1971
1980
1990
2000
2004

Embedded System and Experiment, 101/2, EE/CGU, W.Y. Lin
Advances in Integration

If automobile speed had increased similarly over the same period, we could now drive from San Francisco to New York in about 13 seconds (Intel).
SoC Applications

Source: K.J. Lin, “Introduction to SoC Design"
ASIC Design Flow

Front end

Speciation Development
RTL Code Development
Functional Verification
Logic Synthesis

Back end

Timing Verification
Place and Route
Prototype Build and Test

Source: K.L. Tsai, Introduction to SoC Design
Wang, SoC Design Overview
SoC Design Flow

Source: C. W. Jen, NCTU, "SOC Design Methodology".

- Specification
- High Level Algorithm Model: C/C++/COSSAP/VCC/MATLAB
- Hardware/Software Partition: N2C/VCC
- Communication Refinement: N2C/Port-C/VCC
- System Level Design
- Front End
- Back End
- Hardware Design
- Chip
- Software Design
  - RTOS: WinCE/VxWorks
  - Device Driver: Driveway
  - API
  - Embedded Software
Silicon Intellectual Property

Emerging of Silicon IP

- Ideas similar to procedure or function codes sharing in software developments.
- Hardware Description Language (HDL), AHDL, VHDL, Verilog, etc.
- Microprocessors, SRAM, Ethernet controllers, UART's, etc.

Soft IP

- Design and distributed as HDL code.
- Synthesizable, extendable, revisable
- Process Independent

Hard IP

- After Synthesis
- Layout and timing information provided
- Performance Predictable
- Easy to integration
SoB v.s. SoC


Embedded System and Experiment, 101/2, EE/CGU, W.Y. Lin
IP Reuse

- **Personal Reuse**
  - Area Driven Design => Timing Driven Design

- **Intra-Company Reuse**
  - Block Based Design
  - Source code reuse or core reuse

- **Global Reuse**
  - Platform Based Design
  - Plug & Play IP
Why Do We Need SoC?

- Lower Cost
- Reusability
- Lower Power
- High Performance
- High Reliability
- Reduce Size
Challenges of SoC Design

- Productivity Gap
  - Chip Complexity v.s. Process Evolution speed

- Circuit complexity
  - Bigger circuit size but smaller device demand

- Heterogeneous Integration
  - DSPs, Memory, Microprocessors, RF analog circuits, MEMS may require different process

- Higher Density Integration
  - Deep submicron effects: cross-talk, wire delays, etc.

- Time-to-market

- Design Complexity
  - DSPs, HW/SW, RTOS, Digital/Analog IPs, On-Chip buses
Solutions to Design Complexity

 Partition
  ⇒ Well system partition into Hardware and Software based on the functionality

 Modeling
  ⇒ At higher level
  ⇒ Consistent and Accurate

 Use a know real entity
  ⇒ Uniform development platform (platform-based)
  ⇒ Pre-designed component (IP-based)
Hot Topics in SoC Design

- System level design methodology
  - Specification
  - Modeling
  - HW/SW partition and co-design
- Verification and Testing
- IP
- Platform
- DFx
  - Design for manufacturing
  - Design for testing
- Embedded Software
  - RTOS
  - Device Driver
- Mixed mode design
- Low power design
What is an Embedded System?

- A dedicated computer-based system for an application or product
- A computer should have hardware and software run on a programmable device (e.g. processor)
  - HW = processor + memory + I/O
- Either an standalone system or a part of a larger system
SoC and Embedded System

Rapid progress in silicon technology enables complete system to be integrated on single chip, making every single SoC design at the same time a complex real time embedded syste.

-- Filip Thoen, CTO, Virtio Corp