Contents

- Introduction
- System Modeling
- Hardware-Software Codesign
- SoC Verification
- SoC Testing
System-on-Chip Design Trend

System-on-Board (SoB)

System-on-Chip (SoC)
SoC Design Trend


Process Technology

- CPU
- DSP
- DRAM

0.35 μm
0.25 μm
0.18/0.15 μm
0.13 μm

(System Integration)

(DSP Integration)
SoC: System-On-Chip

- System
  A collection of all kinds of components and/or subsystems that are appropriately interconnected to perform the specified functions for end users.

- An SoC design is a “product creation process” which
  - Starts at identifying the end-user needs
  - Ends at delivering a product with enough functional satisfaction to overcome the payment from the end-user
SoC Applications

- Communication
  - Digital cellular phone
  - Networking
- Computer
  - PC/Workstation
  - Chipsets
- Consumer
  - Game box
  - Digital Camera
Benefits of Using SOC

- Reduced size
- Reduced overall system cost
- Lower power consumption
- Increased performance
Challenges in SoC Era (1/2)

- **Time-to-market**
  - Process roadmap acceleration
  - Consumerization of electronic devices

- **Silicon Complexity**
  - Heterogeneous processes
  - Billion Transistors, Deep submicron effects: crosstalk, wire delays, electromigration, mask costs
Challenges in SoC Era (2/2)

- Design Complexity
  - \( \mu \)Cs, DSPs, HW/SW, SW protocol stacks, RTOS’s, digital/analog IPs, On-chips buses
  - System-level architecture

- Time-in-market
  - Performance/Energy/Cost tradeoff
  - Scalable architecture with unified design environment
How to Conquer the Complexity?

- Use a known real entity
  - A pre-designed component (IP reuse)
  - A platform (architecture reuse)
- Partition
  - Based on functionality
  - Hardware and software
- Modeling
  - At different level
  - Consistent and accurate
A predefined, designed/verified, reusable building block for System-on-Chip

Software IP, Silicon IP (Soft IP, Hard IP, …)

IP types
- Foundation IP (cell library, gate array)
- Standard IP (MPEG2/4, JPEG, USB, IEEE 1394, PCI…)
- Star IP (ARM, MIPS, Rambus, …)

Ancillary characteristics
- Deliverable at certain level, software/hardware interfaces
- Modeling at different levels
- Customizable, Configurable, Parameterizable
IPs in SoC
Factors in Selecting IP’s

- Processor IP selection criteria
  - Power, performance, area, cost
  - Flexibility
  - Hardness (hard IP vs. soft IP)
  - Available system software
  - Development environment
  - Simulation model
  - Support library
  - Support OS
  - Inter-operability with other IP’s
Challenges for CAD Tools in IP-based SoC Design

- Designing at higher levels of abstraction
- Verification
  - Better and faster verification
- Timing & Power
  - Better physical design tools and tool integration, for instance 3D modeling
- Testing
  - Different testing schemes
- Capacity
  - To support high number of gate counts
Challenges for CAD Tools in IP-based SoC Design

- IP Integration
  - To support use of commercial IP
- Hard IP Transition
  - Better physical design tool
- IP Standards
  - To facilitate use of IP from multiple sources
- IP security
  - To support various business model
Platform

- A fully defined bus structure and a collection of IP blocks
- A design methodology to support the feature of “Plugging and Playing”
- The definition of a platform is the result of a trade-off process involving reusability (programmability and configurability), cost and performance optimization.
- Enhance the differentiation
Reference design

Derivative design

Added

Removed

Modified
SoC Design Flow

Specifications

- 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

Software Design

- RTOS
  - WinCE/VxWorks
- Device Driver
  - Driveway
- API
- Embedded Software
SOC Co-design Flow

**Design Specification**

- HW/SW Partitioning
- Estimators

**Architecture Description Language**

- Verification
  - Rapid design space exploration
  - Quality tool-kit generation
  - Design reuse

**IP Library**

- HW VHDL, Verilog
- SW C

**Synthesis**

- Compiler

**Co-simulation**

- On-Chip Memory
- Processor Core
- Synthesized HW
- Interface
- Off-Chip Memory
SOC Specification

- Document-based specification
- Executable specification
  - Precise behavior description
  - No communication overhead
  - No standard yet
  - Stable methodology
  - CAD tool support required
Specification Languages

- HDL-based specification language
  - VHDL, Verilog
  - Benefit from existing design flow
  - Good for hardware description

- HLL-based specification language
  - SystemC, SpecC
  - Typically based on C/C++
  - Good for software/system description

- Mixed form
  - Superlog, CoWareC
Chip Modeling Language Trends

System

RTL

Gate

Schematic Entry

Verilog
VHDL

C/C++
SystemC
SpecC

1980s 1990s 2000s
Core Technologies

- IP Development
- System Architecture
- SoC Verification
- Embedded Software
- High Speed/Low Power Design
From Requirement to Deliverables
System Architecture Design

- Specification, Requirement, Functionalities ➔ Architecture
- C-level design, SystemC description and simulation
- Advantages
  - Broader design space
    - performance, power, cost tradeoff
  - scalability, good for time-in-market
  - Early verification
    - module well-defined, partition, refinement
  - necessary for time-to-market
A Typical SOC Architecture

Processor
On-chip bus

CPU
Co-processor
Cache

Core
Core
CPU
Bridge
System
On-chip bus

Arbiter

IP’s with high bandwidth
Core
Core

OCB
Bridge
Peripheral
On-chip bus

IP’s with low bandwidth
Interfacing IPs

- PCB (=Processor + Peripheral) shrinks to SOC
- Interface between HW and HW
  - FIFO-based interface: I/O
  - On-chip bus is required
- On-chip bus: System Bus & Peripheral Bus
Candidates for Standard On-Chip Bus

- ARM (www.arm.com)
  - AMBA (Advanced Microcontroller Bus Architecture)
- IBM (www.chips.ibm.com)
  - CoreConnect (PLB/OPB/DCR)
- PALM Chip (www.palmchip.com)
  - M Bus/Palm Bus
- Mentor Graphics (www.inventra.com)
  - FISP Bus
- OMI (www.omimo.be)
  - PI (peripheral Interconnect) Bus
- Fujitsu (www.fujitsu.com)
  - Spcl Bus
Scalable Design

Application Software in C/C++

Profiling

Software code  Instruction Extension  Function Unit Extension  ASIC block

Power

Performance

Efficiency

Software code

1st product release

IE

Software code

2nd product release

FUE

IE

Software code

3rd product release

ASIC

FUE

IE

Software code

Final product release
SoC Verification

- System-level verification
  - concurrent, early software hardware co-simulation
  - testbench setup
  - behavior modeling: instruction set simulator, bus functional model, memory behavior model, Verilog or SystemC hardware model

- A dedicated testbench for every IP
  - Register access test to verify bus
  - Test for checking of blocks interconnected functionality and block external interfaces
  - Emulation
What is ISV?

- ISV = In-System Verification

When is ISV required?

- Design refinement down along the hierarchy
  - Comparison between design levels
- In-system operation: confirm correct behavior in system environment
  - Simulation (Chip, I/F)
  - All-software (Software, Software)
  - Emulation (HW[FPGA], HW)
  - Virtual chip (Software, Hardware)
## Embedded Software Architecture for SoC Design

<table>
<thead>
<tr>
<th>Host Application</th>
<th>MMI/GUI</th>
<th>Error Handling</th>
<th>Memory Allocation</th>
<th>Diagnostics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message Manager</td>
<td>Task controller</td>
<td>State Machine</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Application Program Interface

<table>
<thead>
<tr>
<th>Stack Protocol</th>
<th>Display Services</th>
<th>Alarm Services</th>
<th>Event Manager</th>
<th>Kernel Services</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>File Manager</td>
<td>Library</td>
<td>Data I/O</td>
<td></td>
</tr>
</tbody>
</table>

### Device Drivers

### Hardware
High Speed / Low Power Design

- Deep submicron effect
- High speed circuit design
- Low power / low voltage design
- Tradeoffs:
  - Cost v/s Functionality
  - Cost v/s Speed
  - Power v/s Speed
Conclusion

- SoC產業就是知識經濟
  知識： 客戶需求
  功能實現
  設計方法
  整合化
  加值化
  “SoC產業”

- 建立Infrastructure、掌握Core Technologies是我們努力的目標