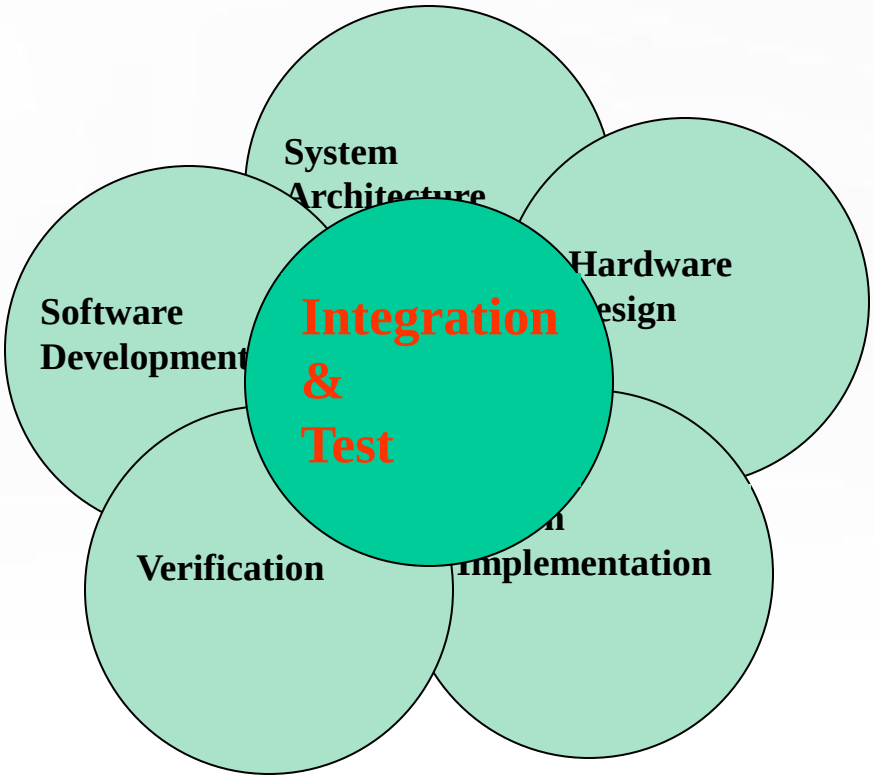
Virtual Platforms : Challenges and Opportunities

Ramesh Chandra

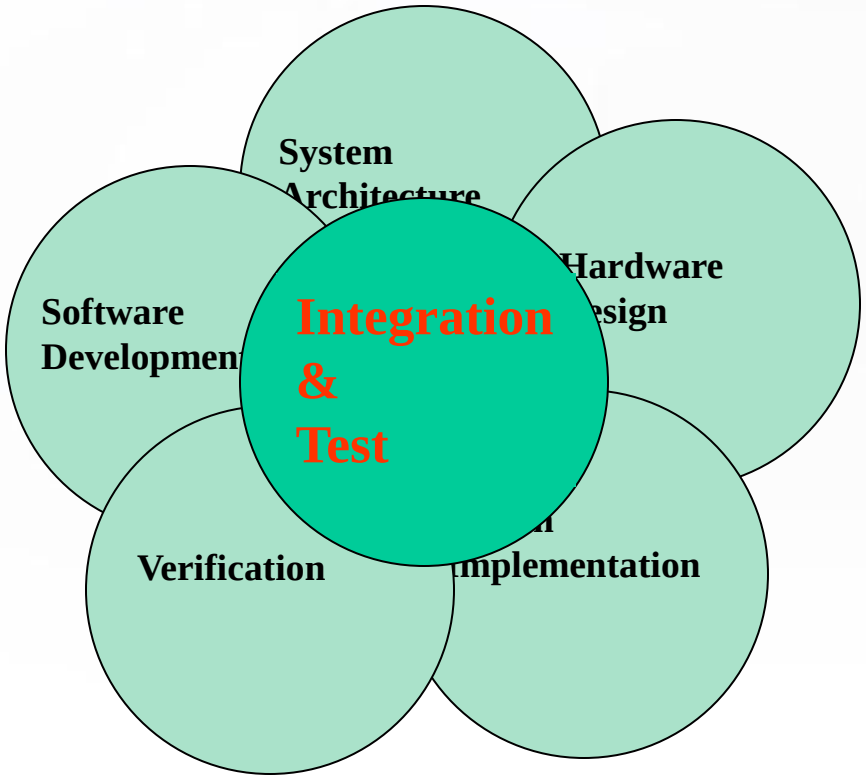
Topics Covered

- Scope
- Structure
- Opportunities
- Challenges
- Summary

SoC Design Framework



SoC Design Communication



- Documents
- Meetings
- Memos, reports

=> Need to use executable specification models

The Design World : Transitions

Algorithm Development

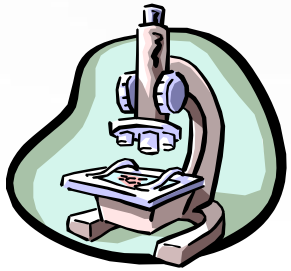
How To Do it ?

NO CLUE YET

Architecture Development

How to do it Effectively

PLATFORM METHODOLOGY



Micro-Architecture / Design Development

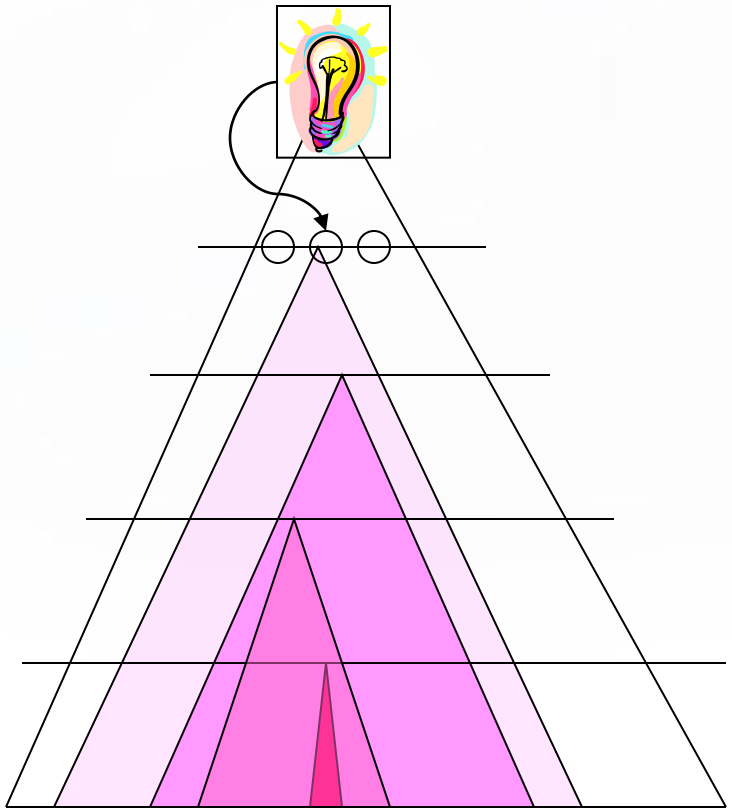
This is how we will do it

RTL2LAYOUT FLOW

Design Implementation

The Gory Details

Abstraction Layers & Interaction



Abstraction Layer

Functional

Architecture

RTL / u-Arch

Gate-Level

Silicon

Design Consistency

Top Down Design Methodologies

Bottom-Up Design Abstraction

Hw / Sw Co-design

- **Traditional Model**
  - HW => SW Development
  - SW Development used HW Prototyping
  - Little to None Pre-silicon Integrated validation

- **HW-SW Co-design**
  - HW || SW Development
  - HW Virtual Model for SW Development
  - Enables Integrated Validation

- **HW-SW Integrated Design**
  - HW && SW Development
  - Co-operative Design
  - Architectural Analysis
  - Platform Design

Virtual Platform Objectives

- **Time To Market**
  - Faster design Implementation
  - Early HW-SW Integration
- **Product Quality**
  - Better co-ordination among Design Teams
  - Improved Verification Coverage
  - Pre-silicon System Validation
  - Pre-silicon / Post silicon co-relation
- **Architecture / Design Reuse**
  - Arch To Implementation
  - HW-SW Arch Trade-offs
  - Product Requirement to Arch Mapping
  - Test Re-use

Topics Covered

- Scope
- **Structure**
- Opportunities
- Challenges
- Summary

Real v/s Virtual

- **Real HW Platform**
  - Starts with RTL Design
  - RTL Simulation
  - Prototyping HW Platform
- **Real SW**
  - Real Drivers
  - Real OS
  - Full Application
- **Virtual HW Platform**
  - Pre-RTL Design
  - Abstract Design
  - C Modeling
- **Virtual SW**
  - Functional APIs
  - Integration Validation

Virtual Platform Components

- **Processor Models**
  - ISS Models
  - Limited Cycle Accuracy
  - High Performance
  - SW Debug
- **Interconnect /Memory Models**
  - Abstract Models
  - Successive Refinement
  - Protocol Dependency
  - Traffic Analysis
- **IP Core Models**
  - Functional Models
  - Performance Models
  - Cycle Accuracy
  - Interface Modeling
  - Interoperability using TLM

The Models

Algorithm / Mathematical Models — Research

Functional Models

Timed Functional Models — Architecture

Cycle Accurate Behavioral Models

RTL Models — Design

TLM Modeling

- Abstraction of Communication
- Separation of computation from communication
- Better Simulation performance
- Standard interfaces for interoperability of models
- Useful for all abstraction levels leading up to RTL
- SystemC TLM2.0 Standard Model Interface

Topics Covered

- Scope
- Structure
- **Opportunities**
- Challenges
- Summary

Opportunities : Productivity

- **SW Development**
  - Early HW Virtual Platforms
    - Phased Definitions of Virtual Platforms
  - Unambiguous HW-SW Interface
- **Architecture Definition**
  - Feature Concept Review and Validation
  - Performance Validation
  - HW Feature Finalization
  - SW Interface Validation
- **HW Development**
  - Unambiguous HW Specifications
  - Reference Model for Development
  - Models for Standard IPs & Debug Tools
  - VP for RTL Verification
- **Early Customer Access**
  - Platform for Customer Application development
  - Product Feature and Performance validation

Opportunities : Quality

- **HW Validation**
  - Early Test development
    - Validate Tests before design is available
  - Reference Verification Model for complex IPs
- **HW-SW Integrated Validation**
  - SW Driver Validation
  - OS and Application Bring-up
- **Improve Coverage**
  - HW Validation using SW use cases
  - Full Validation of SW APIs accessing
  - SW Interface Functional Coverage
  - Performance Simulations and Analysis
  - Concurrency and Application Based Verification

Topics Covered

- Scope
- Structure
- Opportunities
- **Challenges**
- Summary

Challenges : Productivity

- **Tool / Language Selection**
  - **Models Development Faster than RTL**
    - Needs to be well before Design to make an impact
  - **Design at Higher abstraction**
    - Complex communication & synchronization concepts
    - Lack of education and expertise
  - **Too complex modeling Lang/Tool**
    - Designed to solve universal problem?
    - Is it good or fast enough for me
    - Am I good enough to develop Models with it?

- **Model Development**
  - **Planning and resources**
    - Design is planned and staffed, models are required but not always planned
  - **Internal development v/s outsourcing**
    - Standard IPs can be outsourced but core IP need to be developed
  - **Specification and Expertise Needed**
    - Architecture /IP designer best suited to develop IP models
    - Tool/Language Training
  - **Legacy IP Models**
    - Invest time in Models for Legacy IP?
    - Platform dependence on Legacy cores.

Challenges : Quality

- **Model Specification**
  - Model specification and Scope
    - Models developed for one scenarios and expected to work everywhere
  - Design Specifications Keep changing
  - Models stop tracking design changes
- **Model validation**
  - Models are developed and used
    - Used as reference models for design verification
  - Coverage based Model verification
  - Model verification compromised for Platform delivery schedules
- **Platform Integration**
  - Different flavors of models
  - Adaptors and wrappers for Platform Integration
  - Functional v/s cycle accuracy
  - Platform Validation

Topics Covered

- Scope
- Structure
- Opportunities
- Challenges
- **Summary**

Summary

- **Virtual Platforms Have Great Potential to Make an Impact**
  - It takes some effort and planning to realize though
- **Tools/Vendors focus on Productivity**
  - Standard IP Models
  - Debug tools
  - Monitor/Checkers
  - Analysis and Profiling
- **Virtual Platform Development**
  - Plan and prioritize Model/Platform development
  - Platform requirements and Tool /Language selection
  - Scope and spec Models
    - Long Term re-use
    - Multiple platform re-use
  - Its real so start owning and benefitting from it