Design Cycle for Microprocessors

Qiong Cai, Josep M Codina, Antonio González, Pedro Marcuello, **Raúl Martínez**, Kyriakos Stavrou, Xavier Vera, Polychronis Xekalakis

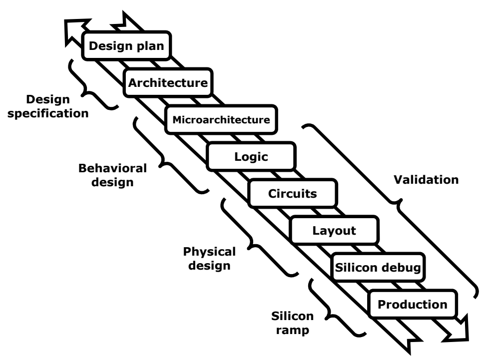
Intel Barcelona Research Center

Aula Empresa, Facultat d'Informàtica de Barcelona, February 2010

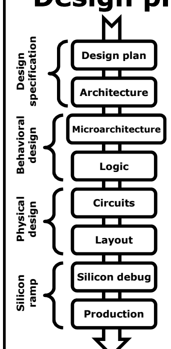
Agenda

- Introduction
- Design plan
- Architecture design
- Microarchitecture design
- Logic design
- Physical design
- Silicon ramp
- Design Types and Intel Tick-Tock model
- Conclusions

Introduction



Design plan

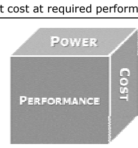


- Design plan must consider the entire design flow from start to finish and answer several important questions:
  - For which product and market segment is going to be used the processor?
  - Which are the requirements in terms of performance, power, and cost?
  - What previous design (if any) will be used as a starting point and how much will be reused?
  - How long will the design take and how many designers are needed?

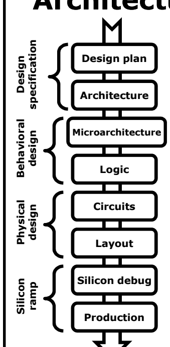
Design plan

Market	Product	Priorities
Server	High-end server	Performance, reliability, and multiprocessing
	Farm/Blade server	Performance, reliability, and multiprocessing within power limit
Desktop	High-end desktop	Performance
	Mainstream desktop	Balanced performance and cost
	Value desktop	Lowest cost at required performance
Mobile	Mobile desktop replacement	Performance within power limits
	Mobile battery optimized	Power and performance
Embedded	Mobile handheld	Ultralow power
	Consumer electronics and appliances	Lowest cost at required performance

Microprocessor Products and Market segments



Architecture design



- A processor architecture is all the features of a processor that are visible to the programmer (Operating System and Applications).

Design decision	Possible choices
Operand types for computation	Pure register, register/memory, pure memory
Data format supported	Integer, floating point, SIMD, …
Data Address Modes	Absolute, register indirect, displacement, indexed, …
Virtual memory implementation	Virtual address size, allowed page sizes, page properties
Instruction encoding	Fixed or variable, number registers, size of immediates, …

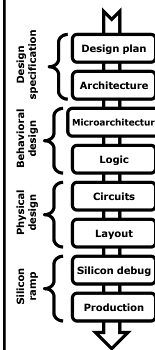
⇨ **Instruction Set Architecture (ISA)**

Architecture design

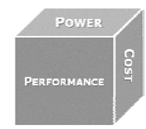
Instruction Set Architecture (ISA) Category		
CISC	Complex Instruction Set Computers	Complex but compact instructions
RISC	Reduce Instruction Set Computers	Simple instructions
VLIW	Very Long Instruction Word	An instruction is a set of operations grouped together by the compiler

Category	Architecture	Processor	Manufacturer
CISC	VAX	MicroVax 78032	DEC
	X86	Pentium 4, Athlon XP	Intel, AMD
RISC	SPARC	UltraSPARC IV	Sun
	PA-RISC	PA 8800	Hewlett Packard
	PowerPC	PPC 970 (G5)	IBM
VLIW	EPIC	Itanium 2	Intel

Microarchitecture design

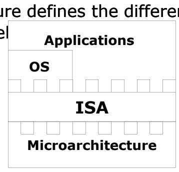


- The architecture defines the instructions the processor can execute.
- The microarchitecture define determines the way in which those instructions are executed.
- Microarchitecture decisions have the greatest impact on the processor's performance, power, and area (cost).



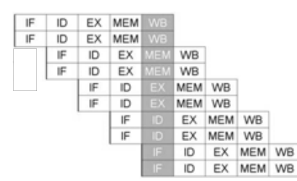
Microarchitecture design

- Microarchitecture changes are not visible to the programmer and can improve performance without software changes.
- Because microarchitectural changes maintain software compatibility, processor microarchitecture have changed much more quickly than architectures.
- Today's higher integration capacity allows more complex techniques to be implemented.
- The microarchitecture defines the different functional units on the processor as wel[l as the interactions] and division of work between them.



Microarchitecture design

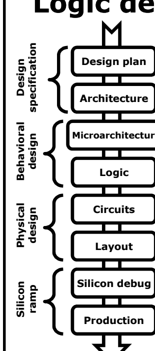
- Designing a processor microarchitecture involves trade-offs of IPC, frequency, die area, power, and design complexity.
  - Number of stages of the pipeline.
  - Instruction issue width.



Microarchitecture design

- Designing a processor microarchitecture involves trade-offs of IPC, frequency, die area, power, and design complexity.
  - Number of stages of the pipeline.
  - Instruction issue width.
  - Methods to resolve control dependencies.
  - Methods to resolve data dependencies.
  - Memory hierarchy.
  - In-order / out-of-order execution
  - Multi threading
  - Branch prediction
  - Number and type of functional units

Logic design



- Typically, microarchitecture design produces diagrams showing the interaction of the different units of the processor and a written specification describing the different algorithms.
- The logical design goal is to obtain a much more detailed and formal description of the logical behavior of all the units and the signals that communicates them.
- The microarchitectural specification is turned into a logical model that can be tested for correctness.

Logic design

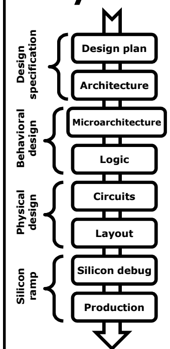
- In order to obtain this model, a Hardware Description Language (HDL) is used to describe the processor.
- HDL languages as Verilog and VHDL, are high-level programming languages created specifically to describe and simulate hardware designs.

Detail

HDL levels of abstraction	
Behavioral level	Includes all the important events but not specifies their exact timing.
Register transfer level (RTL)	Models the processor clock and the events/signals that happen at each cycle. An RTL model should be an accurate simulation of the state of the processor at each cycle boundary.
Structural level	Shows the detailed logic gates to be used within each cycle.

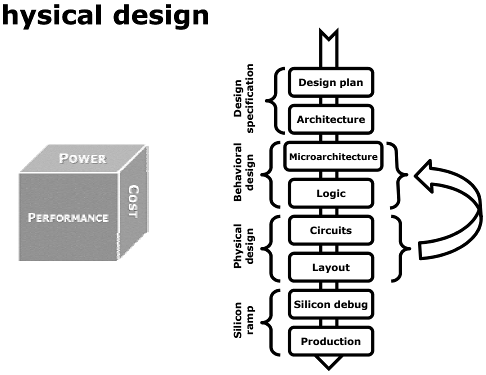
Abstraction

Physical design

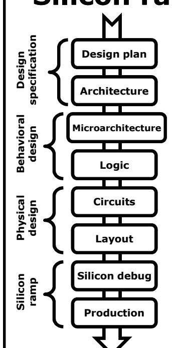


- Circuit design creates a transistor level specification of the logic modeled with HDL.
- The layout determines the position of the transistors and wires on the different layers of material that make up of the circuit design.
- An important result of this step is obtaining accurate estimates on the clock frequency, the power, and the area of the design.
- This is the first step where the real world behavior of transistors must be considered as well as how that behavior changes with each fabrication generation.

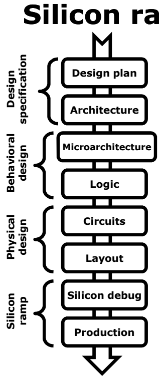
Physical design



Silicon ramp

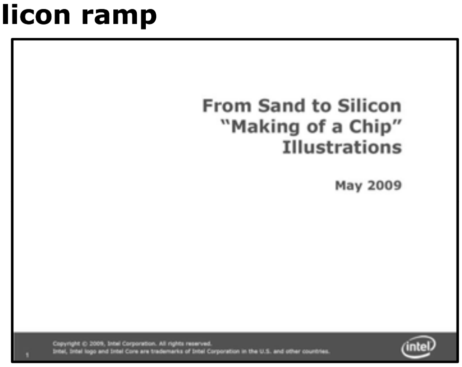


Silicon ramp



- Silicon debug is the process of identifying bugs in prototype chips.
- Design changes are made to correct any problem as well as improving performance as new prototypes are created.
- This continues until the design is fit to be sold and the product is released into the market starting the production phase.

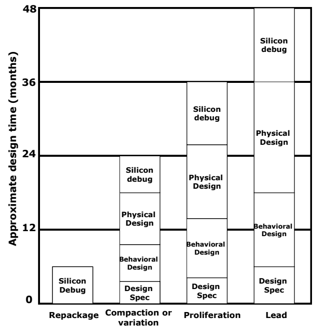
Silicon ramp



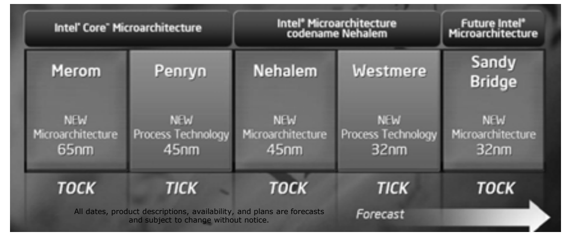
http://www.intel.com/pressroom/kits/chipmaking/index.htm

Design Types and Intel Tick-Tock model

Design Type	Reuse
Lead	Little to no reuse
Proliferation	Significant logic changes and new manufacturing process
Compaction	Little or no logic changes, but new manufacturing process
Variation	Some logic changes on same manufacturing process
Repackage	Identical die in different package



Design Types and Intel Tick-Tock model



Year 1: First the "Tick"
Intel delivers new silicon process technology, dramatically increasing transistor density while enhancing performance and energy efficiency within a smaller, more refined version of our existing microarchitecture.

Year 2: Then the "Tock"
Intel delivers entirely new processor microarchitecture to optimize the value of the increased number of transistors and technology updates now available.

Conclusions

- Moore's Law predicts the increase in transistor density.
- Transistor scaling and growing transistor budgets have allowed microprocessors performance to increase at a dramatic pace, but they have also increased the effort of microprocessor design.
- The production of new fabrication generations is inevitably more complex than previous generations.
- This implies a higher effort in validation at all the design levels.
- There is a need for new and better methodologies and tools to help in the different tasks.
- A sustained research at all the steps but specially at the fields of microarchitecture and process technology is required.