**Design Cycle for Microprocessors**

Matteo Monchiero, Raúl Martínez

Intel Barcelona Research Center

Aula Empresa, Facultat d’Informàtica de Barcelona

© Intel Corporation, 2011

---

**Agenda**

- Introduction
- General Phases
- Design Steps
- Validation
- Design Types and Intel Tick-Tock model
- Conclusions

---

**Introduction**

---

**General Phases**

- Market Analysis
- Target Market segments
- Key features
- New technologies
- Platform innovation & partitioning
- Explore Tools & Methodology

---

**General Phases**

- Product Requirements
- Technology identification
- Technology selection
- Technology development
- Design methodology & tool selection
- Team composition

---

**Design Steps**

- Design plan
- Architecture
- Microarchitecture
- Logic
- Circuits
- Layout
- Silicon debug
- Physical design
- Silicon ramp
- Validation
- Production
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 it going to be used the processor?
  - What 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?

Microprocessor Products and Market segments

<table>
<thead>
<tr>
<th>Market</th>
<th>Product</th>
<th>Priorities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server</td>
<td>High-end server</td>
<td>Performance, reliability, and</td>
</tr>
<tr>
<td></td>
<td>Farm/Blade server</td>
<td>multiprocessing</td>
</tr>
<tr>
<td>Desktop</td>
<td>High-end desktop</td>
<td>Performance</td>
</tr>
<tr>
<td></td>
<td>Mainstream desktop</td>
<td>Balanced performance and cost</td>
</tr>
<tr>
<td>Mobile</td>
<td>Value desktop</td>
<td>Lowest cost at required performance</td>
</tr>
<tr>
<td></td>
<td>Mobile desktop</td>
<td>Performance within power limits</td>
</tr>
<tr>
<td>Embedded</td>
<td>Mobile handheld</td>
<td>Ultralow power</td>
</tr>
<tr>
<td></td>
<td>Consumer electronics and appliances</td>
<td>Lowest cost at required performance</td>
</tr>
</tbody>
</table>

Architecture design

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

Instruction Set Architecture (ISA) Category

<table>
<thead>
<tr>
<th>Category</th>
<th>Architecture</th>
<th>Processor</th>
<th>Manufacturer</th>
</tr>
</thead>
<tbody>
<tr>
<td>CISC</td>
<td>VAX</td>
<td>MicroVAX 78032</td>
<td>DEC</td>
</tr>
<tr>
<td></td>
<td>X86</td>
<td>Pentium 4, Athlon XP</td>
<td>Intel, AMD</td>
</tr>
<tr>
<td>RISC</td>
<td>SPARC</td>
<td>UltraSPARC IV</td>
<td>Sun</td>
</tr>
<tr>
<td></td>
<td>PA-RISC</td>
<td>PA 8800</td>
<td>Hewlett Packard</td>
</tr>
<tr>
<td></td>
<td>PowerPC</td>
<td>PPC 970 (G5)</td>
<td>IBM</td>
</tr>
<tr>
<td>VLIW</td>
<td>EPIC</td>
<td>Itanium 2</td>
<td>Intel</td>
</tr>
</tbody>
</table>

Microarchitecture design

- The architecture defines the instructions the processor can execute.
- The microarchitecture defines 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.

Microarchitecture design

The microarchitecture defines the different functional units on the processor as well 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.

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.

HDL levels of abstraction

<table>
<thead>
<tr>
<th>Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Behavioral level</td>
<td>Includes all the important events but not specifies their exact timing.</td>
</tr>
<tr>
<td>Register transfer level (RTL)</td>
<td>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.</td>
</tr>
<tr>
<td>Structural level</td>
<td>Shows the detailed logic gates to be used within each cycle.</td>
</tr>
</tbody>
</table>
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.

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.
**Validation**

- Design plan
- Architecture design
- Pre-silicon validation
- Physical design
- Timing closure
- Tape-out
- Post silicon validation

**Pre-silicon validation**

- Verify functionality and basic performance of a module
- Designer
  - prepares test cases,
  - run test cases on a simulator,
  - compare outputs with specification
- Formal methods
  - FP ALU, protocols

**Post-silicon validation**

- Functional verification of “system” aspects
  - For example, power modes and memory controller
- 4/5 iterations or “steppings”
- Lasts 1-2 years

**Post-silicon validation methods**

- Random Instruction Testing (RIT)
  - “Volume”
- Real software testing
- Output checked against architectural simulators
- Simulators or emulators (FPGA) used to reproduce bugs

**DFT/DFD features**

- Design for Test/ Design for Debug
  - Specific HW features to ease testing/debug
- Scan logic
  - Logic to load and extract fine grain data
  - Allow reachability
  - Allow observability
**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.