Principles of VLSI Design

Instructor

Chintan Patel

(Contact using email: cpatel2@cs.umbc.edu).

Text

CMOS VLSI Design: A Circuits and Systems Perspective, Third Edition.

by Neil H.E. Weste and David Harris.

ISBN: 0-321-14901-7, Addison Wesley.

Supplementary texts

Digital Integrated Circuit Design

by Ken Martin, Oxford University Press (2000).

Digital Integrated Circuits, A Design Perspective, Second Edition

by J. Rabaey, A. Chandrakasan and B. Nikolic, Prentice Hall (2003).

Further Info

http://www.cs.umbc.edu/~cpatel2

Purpose of the Course

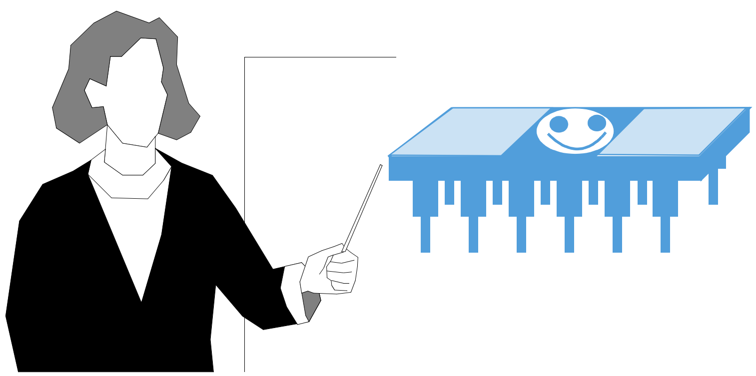
- To introduce the concepts and techniques of modern integrated circuit design (CMOS VLSI).

- To provide experience designing integrated circuits using Commercial Computer Aided Design (CAD) Tools (CADENCE).

The VLSI Design Process

The Design Process: An iterative process that refines an ***idea*** to a manufacturable device through at least five levels of design abstraction.

Top level: The ***idea*** refined into a set of requirements called → *Specification:*



- What does the chip do?
- How fast does it need to operate in order to be competitive?
- How much power will it consume?
- How big will it be?

Design Constraints:

- Speed, power and area.

The VLSI Design Process

Abstraction: A very effective means of dealing with design complexity. Creating a model at a higher level of abstraction involves replacing detail at the lower level with simplifications.

Simulation: The functional behavior of the design (or a parameter such as power) is determined by applying a set of excitation vectors to a circuit model.

Levels of abstraction:

(1) Functional (architecture)

(2) Register Transfer Level (microarchitecture, block)

(3) Logic Design

(4) Circuit Design

(5) Physical Design

The VLSI Design Process

Specification

↓

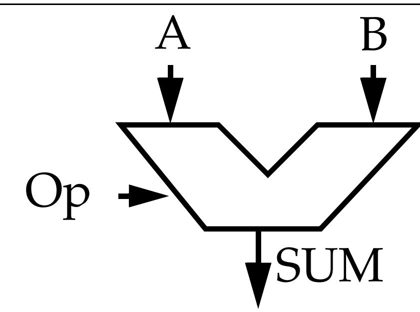
Functional Design

Behavioral Simulation

Hardware Description Languages

Verilog, VHDL etc.

VHDL Example: 32 bit adder



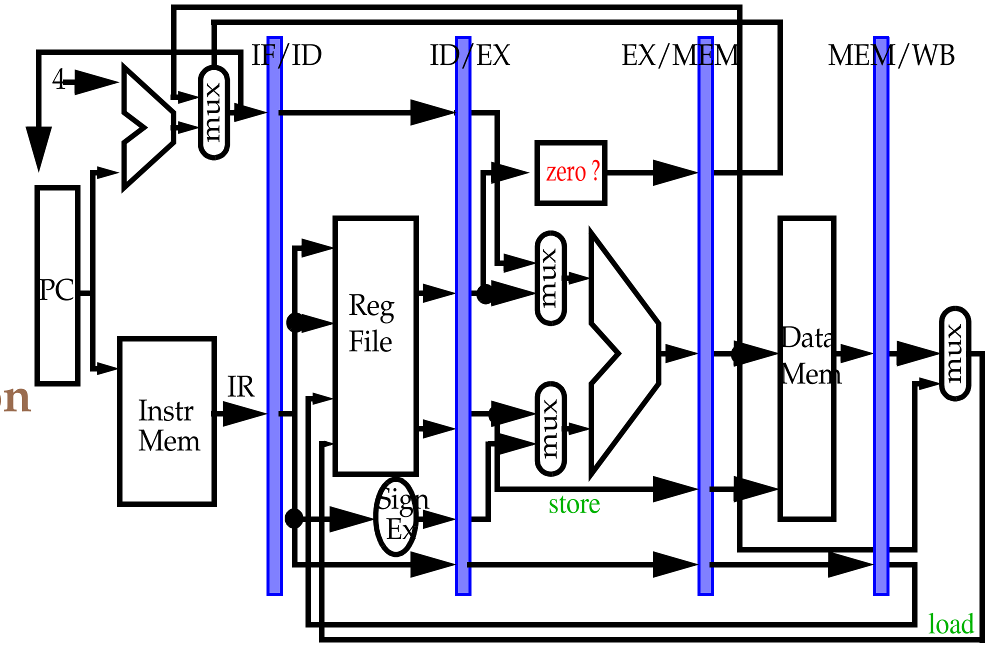

```
entity ALU32 is port (
    A, B: in bit_vector(31 downto 0);
    .....) end ALU32;
```

```
if (a=b) then
    sum <= '0' ;
else
    sum <= (a or b);
end if;
```

The VLSI Design Process

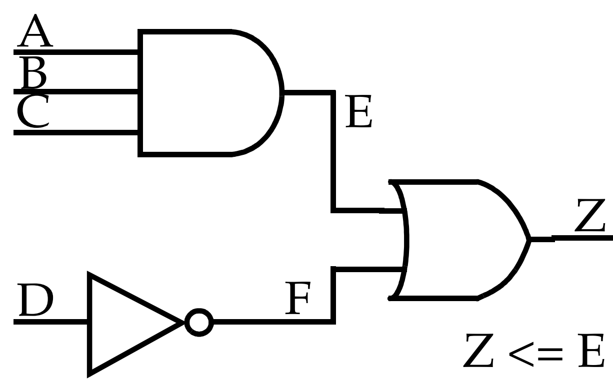
Register Transfer Level Design

RTL Simulation



Logic Design

Logic Simulation

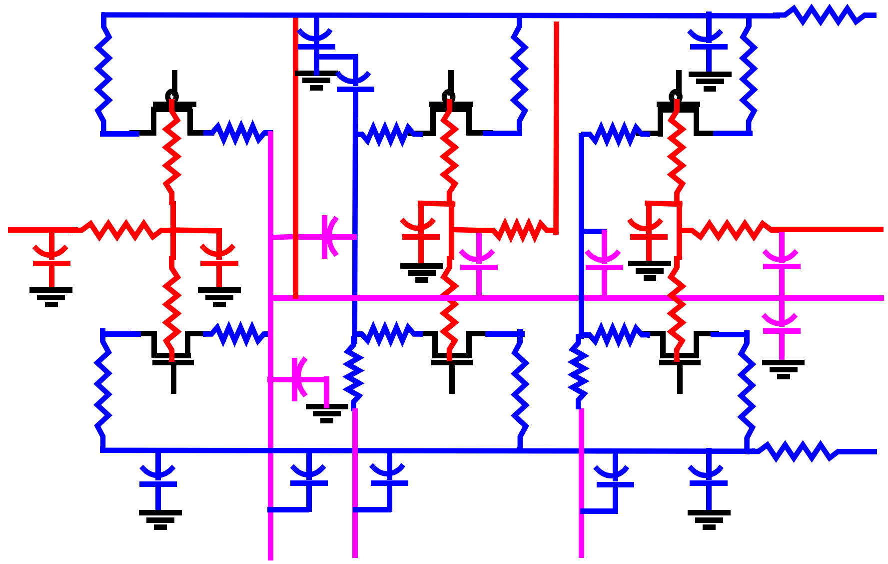


Z <= E **OR** F

The VLSI Design Process

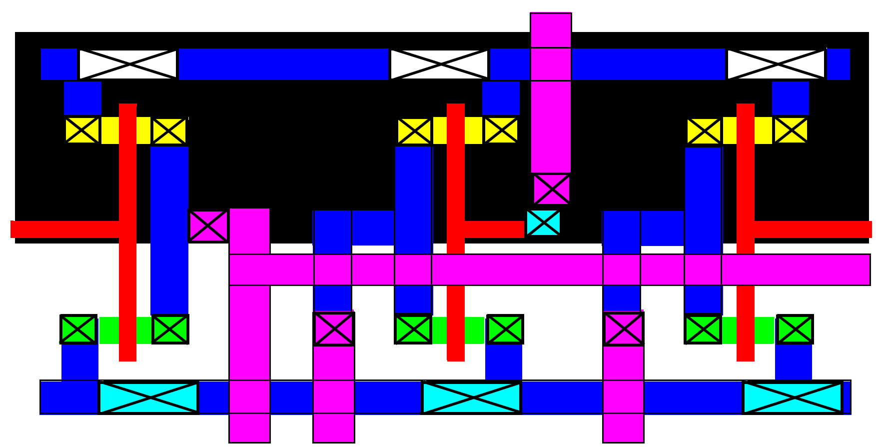
Circuit Design

Timing Simulation



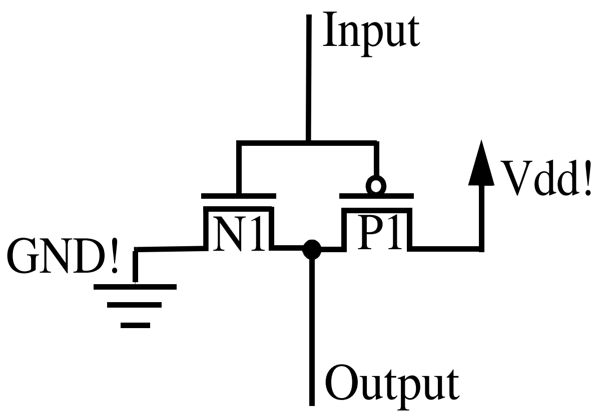
Physical Design

Design Rule Checking



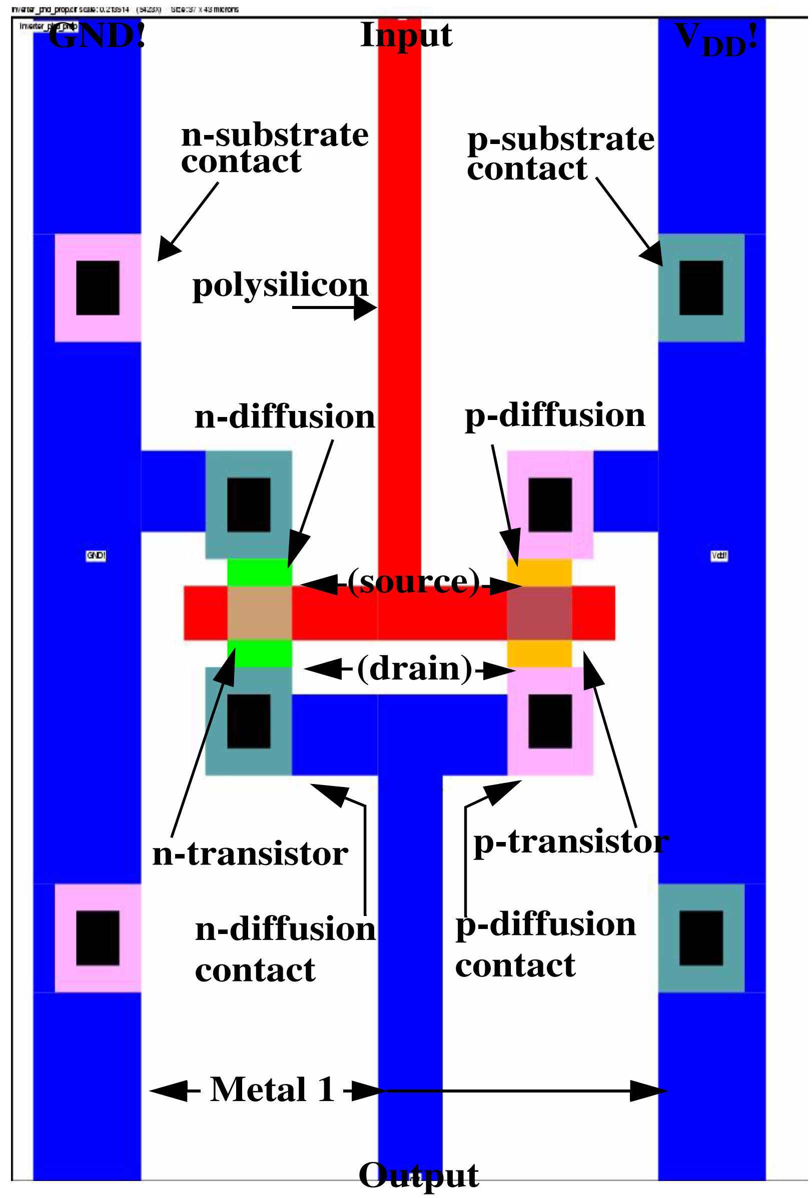
What is CMOS?

A CMOS Inverter



Inverter Schematic

Inverter Layout



Hierarchy and Abstraction

Moore's Law: Integration density doubles every 18 months.

For example, Microprocessors:

The million transistor/chip barrier crossed in '88 with the 486. Today we have more than 100 million transistors on a single chip.

Impact of this revolution on design:

Hand crafting not possible anymore for designing a pentium IV (as was done for the 4004).

Hierarchy is used in the design of complex VLSI circuits.

A large system can be partitioned into many units. Each unit can have functional blocks, blocks are built from cells, cells are ultimately constructed from transistors.

The processor is a collection of modules each composed of cells. Re-use of cells reduces design effort.

Abstraction is also used in digital designs.

It is critical for dealing with the design complexity.

Hierarchy and Abstraction

Entire CAD design frameworks are based on this design philosophy.

- These have made it possible to achieve current design complexity.

Examples of CAD tools for digital design are:

- Simulators that work at various complexity levels.
- Design verification tools.
- Place and Route tools. (Layout generation)
- Logic synthesis tools.

Standard cells are a popular design style that makes layout generation easy.

- Layouts of basic gates such as AND, OR, NAND, NOR, and NOT as well as arithmetic and memory modules are provided as input.

- These cells are designed with similar characteristics, such as constant height, and can be manipulated easily to generate a layout.

- Place-and-Route tools can use these libraries and generate layouts using logic level description of the design.

Digital Circuit Design

If design automation solves all the problems, why be concerned with digital circuit design?

Reality is more complex and a knowledge of digital circuit design will be important for some time to come.

- Someone has to design and implement the module libraries.
  - Porting from technology generation to technology generation (different feature sizes) is NOT automatic.
  - This occurs approximately every two years!
- Creating an adequate *model* of a cell/module requires an in-depth understanding of its internal operation.
- The library-based approach does NOT work for all situations, i.e. high performance sub-systems in designs like microprocessors.

Digital Circuit Design

- The abstraction-based approach is only correct to a certain degree.
  Performance of a module, i.e. an adder, is substantially influenced by the way it is connected in its environment (*interconnect parasitics*).

- Scaling tends to emphasize other deficiencies of the abstraction-based approach.
  Global entities, such as clock signals and supply lines, are significantly affected by scaling.

- New design issues emerge over time.
  Power dissipation issue periodically re-emerges.

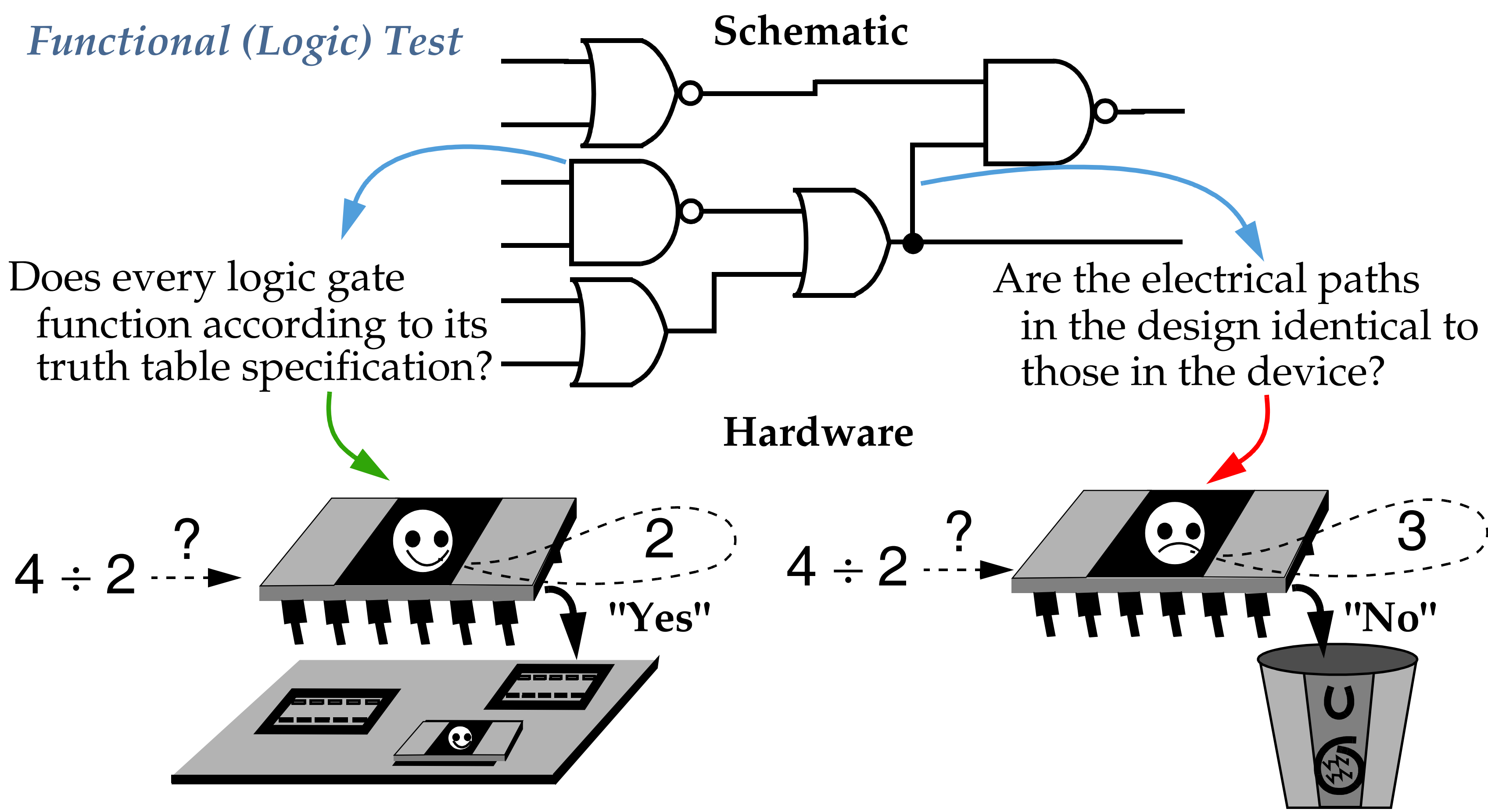
- Trouble shooting an erroneous design requires circuit expertise.

- *And:* You need to know it for doing the class assignments and project.

The VLSI Testing Process (CMPE 418)

A process applied to hardware devices whose goal is to determine if the device is free of fabrication defects that would otherwise cause the device to violate its *functional* or *parametric* specifications.

Functional (Logic) Test



Parametric tests

Based on the analysis of a *continuous* circuit parameter, in contrast to functional test which analyzes *logic signals*

.

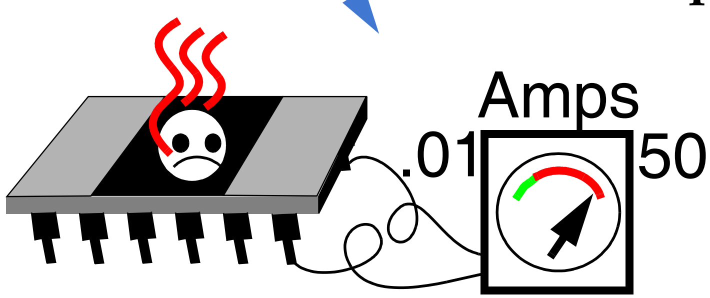
Parametric Tests

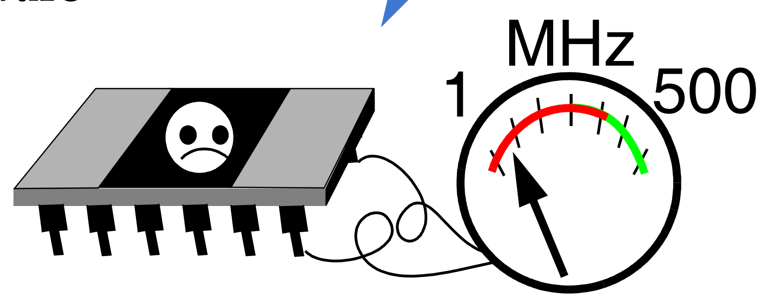
Is the steady-state current requirements of the device excessive?
(I_{DDQ})

Are the effects of process variations within tolerance?
(TSA)

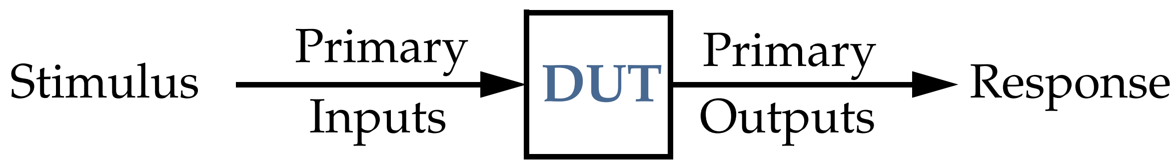
Are the performance requirements met?
(Delay fault)

Hardware





The Testing Process



When Device Under Test (DUT) is digital logic device, the stimuli are called **test patterns** or **test vectors**.

A device test consists of applying the test patterns one at a time (by a tester) to the Primary Inputs of the DUT.

The test patterns are defined in a **test program** that describes the waveforms to be applied, the voltage levels and the clock frequency.

A new part is automatically fed to the tester and a probe card or DUT board is used to connect the inputs and outputs of the tester to the pins of the die or package.

The Testing Process