Lecture 3 (Chapter 2)

Introduction to VHDL

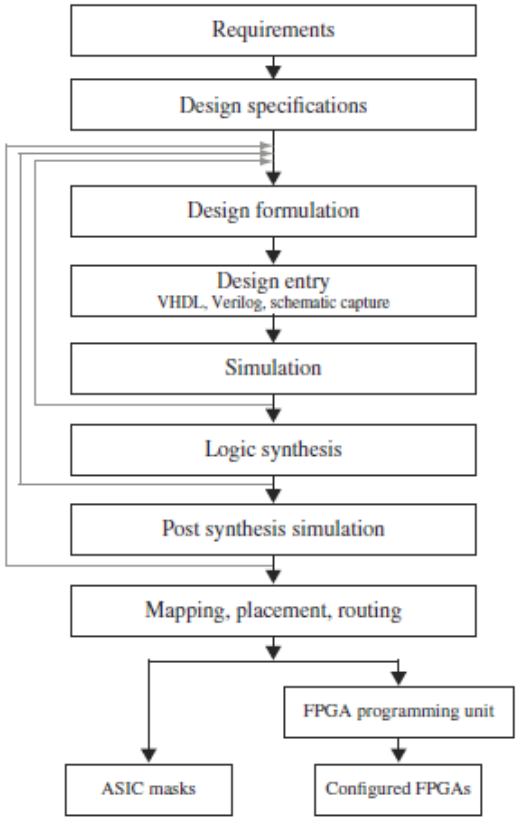
Khaza Anuarul Hoque
ECE 4250/7250

Introduction

- With the allowance of more and more components on a chip, digital systems have continued to grow in complexity.
- Technology improvements have advanced the very large scale integration (VLSI) field. Despite changes in integration ability, the term VLSI remains popular.
- Early integrated circuits belonged to small-scale integration (SSI), medium-scale integration (MSI), or large-scale integration (LSI) groups depending on the density of integration.

Computer-Aided Design (CAD)

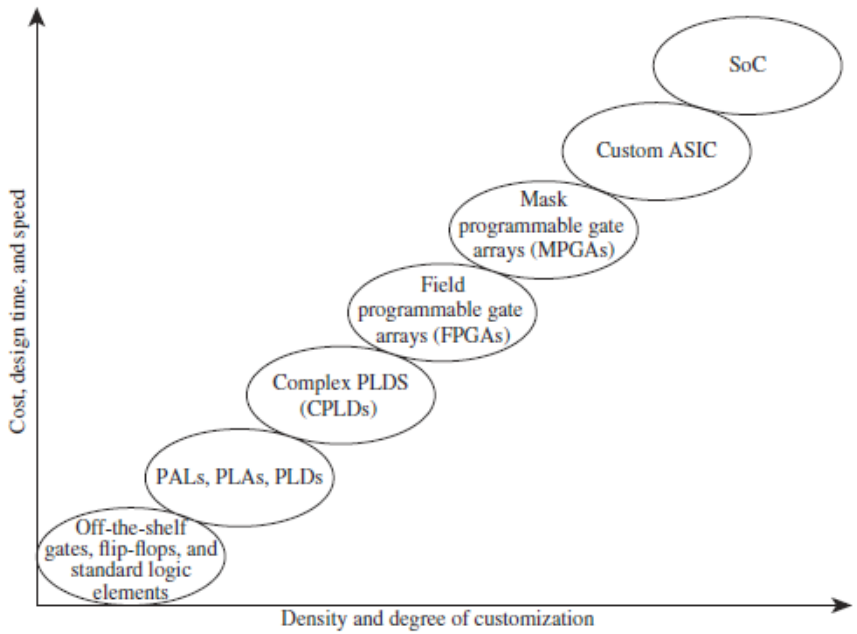
- Steps in modern digital system design:



CAD (continued)

- Target technologies that are available:

FIGURE 2-2: Spectrum of Design Technologies



- Most common: field programmable gate arrays (FPGAs) and application-specific integrated circuits (ASICs).

Hardware Description Languages (HDLs)

- Two popular HDLs—VHDL and Verilog.
- VHDL is an HDL used to describe the behavior and/or structure of digital systems.
- VHDL can describe a digital system at several different levels—behavioral, data flow, and structural.
- VHDL leads naturally to a top-down design methodology.

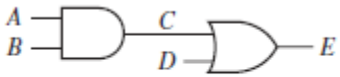
VHDL Description of Combinational Circuits

- VHDL models combinational circuits using **concurrent** statements.
- Concurrent statements:
  - Statements that are **always ready** to execute.
  - Evaluated any time and every time a signal on the right side of the statement changes.
  - Execute repeatedly as if they were in a loop.

VHDL Description of Combinational Circuits (continued)

- Concurrent statements (continuous assignments) example:

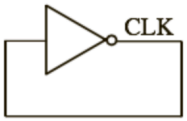
FIGURE 2-3: A Simple Gate Circuit




```
C <= A and B after 5 ns;
E <= C or D after 5 ns;
```

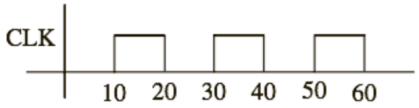
```
C <= A and B;
E <= C or D;
```

VHDL Description of Combinational Circuits (continued)

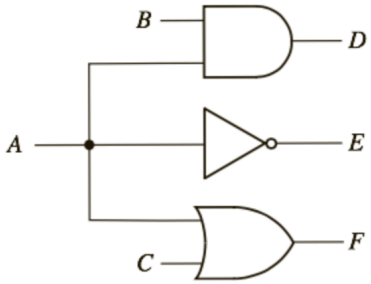



```
CLK <= not CLK after 10 ns;
```

✓



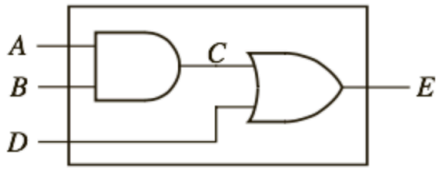

```
CLK <= not CLK;
```




```
-- when A changes, these concurrent
-- statements all execute at the
-- same time
  D <= A and B after 2 ns;
  E <= not A after 1 ns;
  F <= A or C after 3 ns;
```

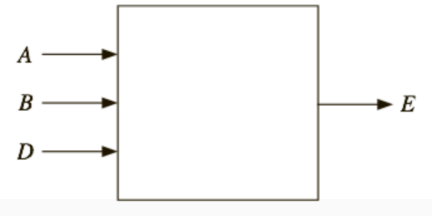
VHDL Modules

- The general structure of a VHDL module is an **entity** description and an **architecture** description.
- **Entity** description declares the input and output signals.
- **Port** declaration specifies the inputs and outputs to the module.
- **Architecture** specifies the internal operation of a module.




```
entity two_gates is
  port(A, B, D: in bit; E: out bit);
end two_gates;

architecture gates of two_gates is
signal C: bit;
begin
  C <= A and B; -- concurrent
  E <= C or D;  -- statements
end gates;
```



VHDL Modules (continued)

- **Mode** indicates the direction of information
- Use the **in**, **out**, and **inout** keywords to specify the direction of port signals.
- **Type** specifies the data type that can be communicated

VHDL Modules (continued)

- Other modes:
  - **buffer**: similar to **inout**, can be read and written into the entity, helpful if the signal is an output.
  - **linkage:** useful when VHDL entities are connected to non-VHDL entities.