EE 434
Lecture 23

Logic Design
This lecture will focus on the various hierarchical levels used in the design of digital systems. Comments will be made about Hardware Description Languages (HDL) which is an integral part of essentially all digital system design. There are two widely used hardware description languages. One is VHDL and the other is Verilog. In this lecture comments will be made about VHDL. In the balance of this course we will focus on Verilog. There is still debate about which is most popular and that continues in this department. We will focus on Verilog in the balance of this course because we believe most ISU students have more familiarity with Verilog than with VHDL. Both will be used in industry and designers will usually be expected to be comfortable working in both.
Design Domains:

Behavioral:

Structural:

Physical:

Behavioral: Describes what a system does or what it should do.

Structural: Identifies constituent blocks and describes how these blocks are interconnected and how they interact.

Physical: Describes the constituent blocks to both the transistor and polygon levels.
Ex:
Example

\[ \frac{1}{L} \quad \omega_L = 4 \]

\[ \frac{1}{L} \quad \omega = 8 \quad \frac{1}{L} \quad \omega = 1 \quad \frac{1}{L} \quad \omega = 2 \]

Diagram:

\[ \omega_1, L \quad \omega_1 + \omega_2, L \]
Example:

\[ A \rightarrow \Delta \rightarrow C \]

\[ C = A \oplus B \]

\[ = A \overline{B} + B \overline{A} \]
In each domain, multiple levels of abstraction are generally used.

**Consider Physical Domain**

- Consider lowest level to highest

  0 - placement of diffusions, thin oxide regions, field oxide, etc. on a substrate.

  1 - polygons identify all mask information (not unique)

  2 - transistors (not unique)

  3 - gate level (not unique)
Structural Level

- HDL
- Netlists

- DSP

- Blocks (Adders, Memory, Registers, etc.)

- Gates

- Transistor
Behavioral Level (top down)

- Application
- Programs
- Subroutines
- Boolean Expressions
Example!

Behavioral Level: System which will give a command to dial an emergency number if a temperature indicator identifies a problem and a humidity indicator indicates no problem or if the temperature indicator indicates no problem but the humidity indicator indicates a problem.

hum: humidity

tem: temp

call: time to dial emergency number.
architecture controller is
begin
    call co (hum xor ten);
end
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
port (hum, ten : in; out : out);
entity alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
architecture controller of alarm is
architecture controller of alarm is
entity alarm is
entity alarm is
Structural Level

\[ \text{Call} = \text{hum} \oplus \text{tem} \]

Physical Level

\[ \Theta_1(\text{call}, \text{hum}, \text{tem}) \]

Static CMOS gates
Representation of Digital Systems

Standard Approach to Digital Circuit Design

1. Behavioral Description
   - technology independent

2. RTL Description
   (must verify 1) and (2)

3. RTL Compiler
   Registers & Combinational Logic Functions

4. Logic Optimizer

5. Logic Synthesis
   generally use a standard cell library for synthesis
6. Place & Route
   (physically locates all gates & registers
    and interconnects them)

7. Layout Extraction
   DRC
   Back Annotation

8. Post layout simulation
   may necessitate a return
   to a higher level in
   the design flow