

ECE 3270 - Digital Computer Design - Dr. Ligon

JAN 12 2024

ASIC: application specific integrated circuit

FPGA: Field programmable gate array

VHDL: hardware description language

textbook: digital design using vhdl a systems approach

room 309 room code: 327468

logical and physical specification

1. design entry types:

- truth tables, waveforms, state diagrams.
- Schematic capture.
- hardware description language.

netlist mixed-mode synthesis

JAN 17 2024

hierarchical design methodology involves creating segments of a product that are designed with different tools at differing levels of specificity.

synthesizer cad tool takes specification/hardware description language and creates design from the specification.

Lab portion of the class involves using synthesizer on HDL code.

schematic capture (drawing schematics) becomes untenable with larger and more complex designs

during process of synthesis, technology mapping involves putting specific hardware into design.

1. EDIF: Electronic Design interchange format.

2. Functional Simulation is final step that verifies design.

- functional simulators assume the time needed for signals to propagate through the logic gates is negligible.

- timing simulators must be used in tandem with functional simulators in order to obtain a complete test of the design.

- add timing bounds to portions of circuit: best, worst, typical

this allows for testing with all possible operation timings.

- Use spice for specific timing measurements/edge cases. Spice uses

numerical methods and has higher precision than simple digital logic simulators

1. Event-driven simulation:

- Zero delay simulators will not detect race and hazard conditions.
- Race condition: when two or more signals are changing simultaneously in a circuit which may result in an incorrect state when a condition is assumed to be stable.

1. Possible states in logic simulation:

- 0

- 1
- U (unknown)
- Z (high impedance) Relevant with TSB (tri-state buffer) and TG (transmission gate)

use reset or preset signal to set unknown back into a known state.

Mixed-mode simulation: logic and spice simulation together

JAN 22 2024

1. Design – Place and route – Schematic Capture

- pin data:
 1. placement
 2. electrical resistance
 3. signal name
 4. special flags
- Routing
 1. traditional : by gate
 2. floor planning : by structure
- Simulations to estimate real-world timing
 - identify critical paths
 - full logical simulation
 - full electrical simulation
 - leads to re-routing, or re-design

Lecture 1: The Digital Abstraction, Combinational Logic.

Associated reading: chapter 1, 3, 6

- The Digital Abstraction
 - representation
 - noise
- Low voltage CMOS logic:

[Damage] – -0.3V – 0.0V – 0.7V – 1.7V – 2.5V – 2.8V [Damage]
- 0.7 to 1.7 V is transition region for 2.5V CMOS logic.
- As opposed to analog systems, digital systems can:
 - Process, transport, and store info without noise distortion.
 - Possible because the signals are discrete
 - No loss of information with added noise until the noise becomes large enough to push the signal out of the valid range
- Digital signals are periodically restored to keep them in the valid range using a buffer.
- In analog systems, since all voltages are valid signals there is no way to restore the signal to a noiseless state between operations.
- Analog systems also limited in precision
 - Accuracy is limited by the background noise.
- All restoring logic devices guarantee that the outputs fall into a range that is narrower than the input range

- Larger noise margins are not necessarily better.

$$V_{NMH} = V_{OH} - V_{IH}$$

Jan 24 2024

- Combinational logic means that logical outputs are based solely on inputs, not state or memory.
 - closed under acyclic composition (as long as a feedback loop is not created when composing combinational logic circuits, then the composition is still combinational)
 - combinational, not combinatorial
- Sequential logic depends on memory and state.
- More than one possible logic equation for a given truth table.
- Demorgan's law:

$$\neg(x \wedge y) = \neg x \vee \neg y \rightarrow \neg(x \vee y) = \neg x \wedge \neg y$$