

# ECE 3561

## Advanced Digital Design

Department of Electrical and  
Computer Engineering  
The Ohio State University

# Today

- Syllabus
- The Course
- Intro
  - Syllabus detail discussion

# Course Philosophy and Objective

- Familiarize students with advanced digital design principles and practice
- Learn to use actual chips for designing practical digital circuits
- Learn modern design technologies with Quartus software and programmable chips
- See the role HDLs have played in design methodology

# Modern Digital Design

- Real logic designs are too large to solve by straight theoretical approach
- Today's methodology
  - Requires use of subdivision of system into Logic Building Blocks. Far above the gate level of AND/OR gates but far below the processor level.
  - Use of CAD
  - Use of PLDs and FPGA – state of the art programmable chips.

# Course Topics

- Review of combination and sequential logic
- Analysis of sequential circuits
- Logic Building Blocks and applications
  - Counters, shift registers, comparators
- Review of traditional approaches to sequential design
- FPGAs and CPLDs

# Course Topics (2)

- **Structured Sequential Design**
  - Based on Logical Building Blocks (LBBs)
  - Complex System = Sum of smaller systems
  - Organize functions, inputs, outputs from word description of problem
  - Art (creative process) – choose LBBs and organize
  - Science – function and timing
- **Design Technology**
  - Using modern CAD
  - Use programmable chips – PLDs and FPGAs
  - Use of HDLs – VHDL, Verilog, System Verilog, System C

# Combination Logic Design

- In today's world digital circuits, both combinational and sequential, have millions of gates and several hundred, if not thousands, of inputs and outputs.
- How do you handle this?
  - Challenges the scope of human comprehension.
  - How much information can a human comprehend?

# Documentation

- “Good documentation is essential for correct design” (from text)
- Design specification must be accurate, complete and understandable
- The starting place is a good specification of the circuit or system.



# The Specification

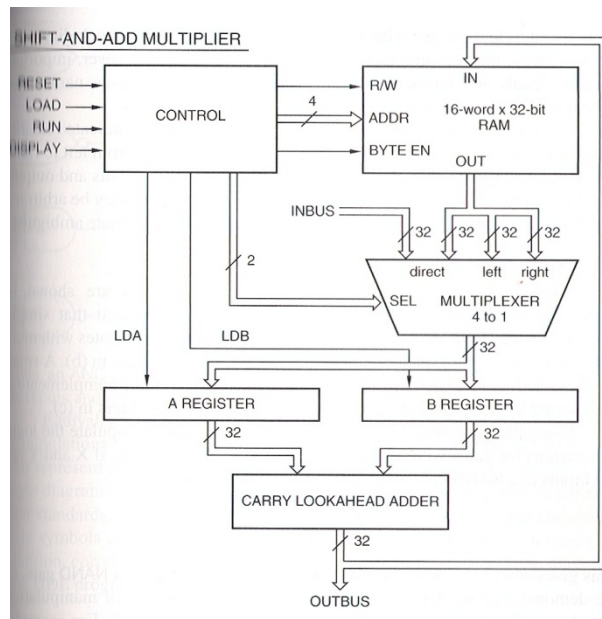
- Describes exactly what the circuit of system is supposed to do.
- All inputs and outputs are accurately specified.
- The internal function performed is fully specified.
  - Algorithm implemented is documented
  - Data format and transformations are specified
- Timing is clear and precise

# Other aspects of documentation

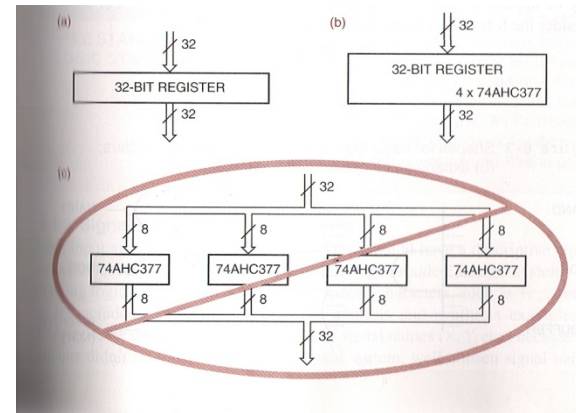
- Block Diagram
- Schematic Diagram
- Timing Diagram
- Structured Logic Description
  - HDL description both documents and allows for simulation and synthesis of the design
- Circuit Description

# Block Diagrams

- Shows inputs and outputs and functional modules



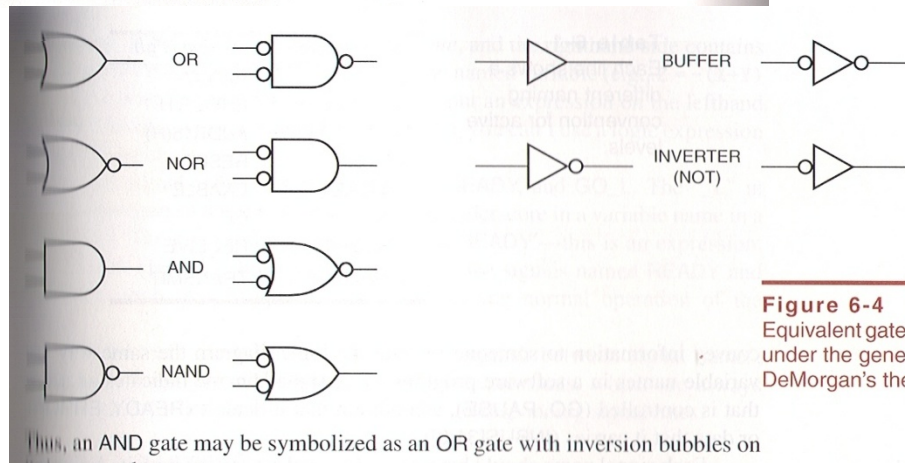
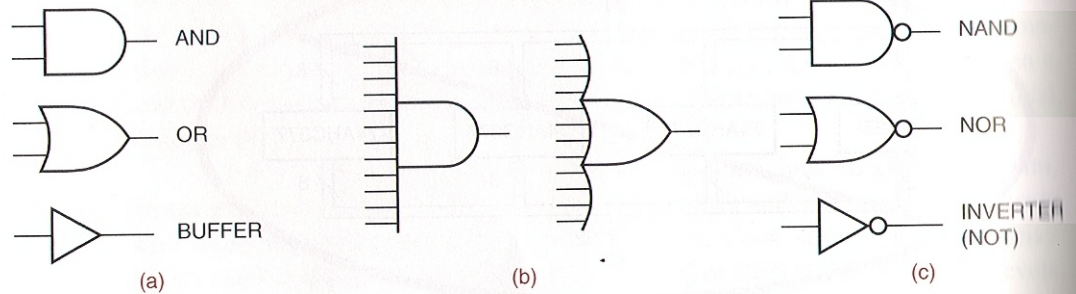
**Figure 6-1**  
Block diagram for a digital design project.



**Figure 6-2**  
A 32-bit register block: (a) realization unspecified; (b) chip specified; (c) too much detail.

# Gate Symbols

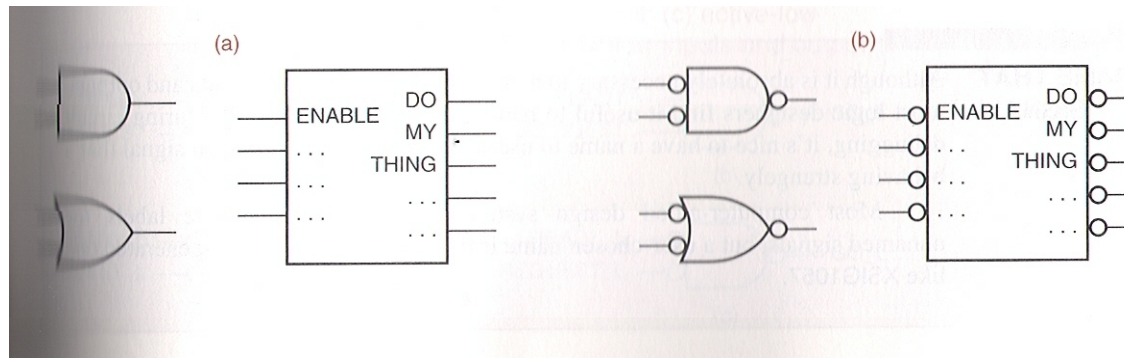
**Figure 6-3** Shapes for basic logic gates: (a) AND, OR, and buffers; (b) expansion of inputs; (c) inversion bubbles.



**Figure 6-4**  
Equivalent gate symbols  
under the generalized  
DeMorgan's theorem.

# Active high and Active low

<i>Active Low</i>	<i>Active High</i>
READY-	READY+
ERROR.L	ERROR.H
ADDR15(L)	ADDR15(H)
RESET*	RESET
ENABLE~	ENABLE
~GO	GO
/RECEIVE	RECEIVE
TRANSMIT_L	TRANSMIT



# Circuit Timing

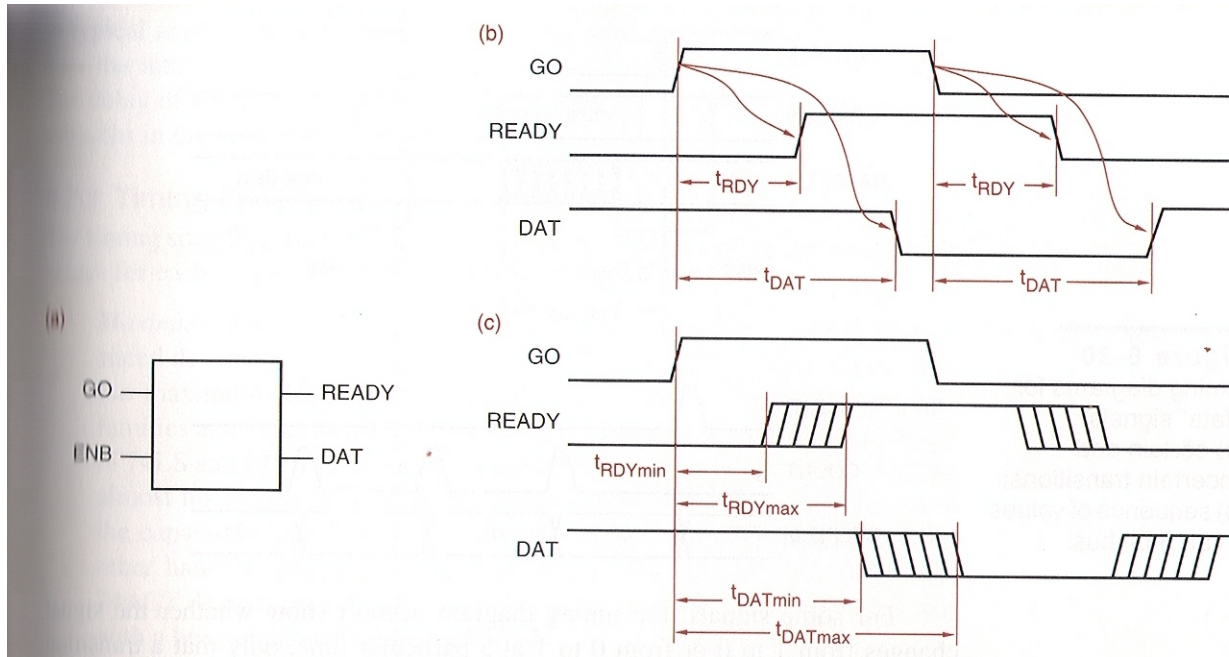
- “Timing is everything”
  - In Comedy
  - In Investing
  - In digital design
- When the inputs change the output of the gate will respond to that change. The output will change (if it does) after the internal circuitry of the gate settles to the new output state.

# Circuit Timing (2)

- Glitches on the output occur when the inputs do not arrive simultaneously.
- It is almost impossible to design a combinational logic circuit that is 100% glitch free. It is possible to create a design in which the glitches that do occur are insignificant.
- This is why synchronous systems have a minimum clock period.

# Timing Analysis Tools

- Circuit timing waveforms





# Expansion into this offering

- This slide will explain the specifics of how each course objective will be achieved.
- Slide 5 – Logic building blocks
  - The datapath of a simple microcontroller will be implemented by creating small components from gates and then integrating them together.
- Sequential circuits
  - The controller for the simple microcontroller will be implemented, illustrating the complexity of controllers of modern processors.

# From Wikipedia

- 1971 TMS 1000 – 8000 transistors – 8um
- 1971 Intel 4004 – 2300 transistors – 10 um
- 1972 Intel 8008 – 3500 transistors
- 1979 Motorola 68000 – 68,000 transistors
- 1993 Pentium – 3.1 M transistors - .8um
- 2000 Pentium 4 – 55 M
- 2006 Pentium 4 Cedar Mill – 184 M - 65nm
- 2008 Core i7 – 731 M - 45nm
- 2010 Quad Core Itanium – 2 B
- 2015 IBM z13 Storage Controller – 7.1 B 22nm

# Significance of last slide???

- All these processors are constructed from arrangement of basic components
  - Multiplexors
  - Adders
  - Selectors – priority encoders
  - Registers, latches and flip-flops
  - Memory arrays
  - Bus drivers

# Assignment

- None today – Read in text
  - Read Chapter 9
  - Will start in Chapter 11 and come back for Chapter 10
- Assignments will be due 2 classes after assigned to the drop box on Carmen.  
No paper submissions – all are electronic.