

# KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

## COMPUTER ENGINEERING DEPARTMENT

### COE 202: Digital Logic Design Term 162 (Spring 2017)

**Instructor:** Dr. Abdulaziz Barnawi

**Class time:** U.T.R.: 11:00-11:50AM

**Class location:** Bldg. 24-255

**Office hours:** Mon, Thu. 9:00-10:00AM (tentative) or by appointment

**Office Location:** 59/1040

**Office-Tel:** 1880

**E-mail:** [barnawi AT kfupm](mailto:barnawi AT kfupm)

**Web site:** <http://faculty.kfupm.edu.sa/coe/barnawi>

#### Course Description:

Introduction to information representation and number systems. Boolean algebra and switching theory. Manipulation and minimization of completely and incompletely specified Boolean functions. Physical properties of gates: fan-in, fan-out, propagation delay, timing diagrams and tri-state drivers. Combinational circuits design using multiplexers, decoders, comparators and adders. Sequential circuit analysis and design, basic flip-flops, clocking and timing diagrams. Registers, counters, RAMs, ROMs, PLAs, PLDs, and FPGA's. Introduction to Verilog.

**Pre-requisite:** PHSY 102 - General Physics II

#### Textbook:

Alan B. Marcovitz , *Introduction to Logic Design*, Third Edition, McGraw-Hill, 2010.

#### Tentative Grading Policy:

Homework Assignments	08%	
Class Discussions	02%	
Quizzes	10%	
Major Exam I	20%	(Tentative: Saturday, March 11, 2017)
Major Exam II	25%	(Tentative: Saturday, April 29, 2017)
Final Exam	35%	
Bonus for 100% attendance	03%	

#### IMPORTANT NOTES:

- All KFUPM regulations and standards will be enforced. Attendance will be checked each class.
- Prompt attendance in classes shows how keen you are to benefit from this course and enhances your performance and grade. Therefore, three late attendances are counted as one absent. Also, more than 5 minutes' latecomers will not be counted as absent.
- KFUPM rule pertaining to a DN grade will be strictly enforced (i.e. > **6 absences** will result in a DN grade).
- Only university approved/certified excuses will be accepted, and should be presented **no later than 1 week** after absence.
- Assignments are submitted **at the beginning of class** at the due date. A late assignment will be accepted only if it is within 24 hours from the due date but you will be penalized 20%.
- You have **48 hours** to object to the grade of homework, a quiz, or a major exam from the end of the class time in which the graded papers have been distributed back.
- Check the course webpage and your Blackboard for updates, emails and announcements.
- *Check your exam schedule carefully.* **NO make up exams without an official excuse that includes Exams. You must inform the instructor on the same day of missed exam/quiz.**

**Course Objectives:**

After successfully completing the course, students will be able to:

1. Use math and Boolean algebra in performing computations in various number systems and simplification of Boolean algebraic expressions.
2. Design efficient combinational and sequential logic circuit implementations from functional description of digital systems.
3. Use CAD tools to simulate and verify logic circuits.

**Course Learning Outcomes:**

<b>Course Learning Outcomes</b>	<b>Outcome Indicators and Details</b>	<b>Assessment Methods and Metrics</b>	<b>Min. Weight</b>	<b>ABET 2000</b>
Ability to use math and Boolean algebra in performing computations in various number systems and simplification of Boolean algebraic expressions.	<ul style="list-style-type: none"> <li>• Represent integer and fractional values in various number systems</li> <li>• Convert number representation from one system to another</li> <li>• Perform arithmetic operations in various number systems</li> <li>• Represent data in different binary codes including error detecting codes</li> <li>• Simplify Boolean expressions using Boolean algebra &amp; identities</li> </ul>	<ul style="list-style-type: none"> <li>• Quizzes</li> <li>• Assignments</li> <li>• Exams</li> </ul>	20%	A(H)
Ability to design efficient combinational and sequential logic circuit implementations from functional description of digital systems.	<ul style="list-style-type: none"> <li>• Derive gate-level implementation of a given Boolean expression and vice versa</li> <li>• Ability to build larger combinational functions using predefined modules (e.g., decoders, multiplexers, adders, Magnitude comparators.)</li> <li>• Ability to build a state diagram / table for both Moore &amp; Mealy models from functional description</li> <li>• Ability to design &amp; implement Moore &amp; Mealy model synchronous sequential circuits using different Flip-Flop types.</li> <li>• Ability to draw timing diagrams for major signals of both sequential and combination circuits</li> </ul>	<ul style="list-style-type: none"> <li>• Quizzes</li> <li>• Assignments</li> <li>• Exams</li> </ul>	50%	C(H)
Ability to use CAD tools to simulate and verify logic circuits.	<ul style="list-style-type: none"> <li>• Ability to simulate and verify the operation of combinational circuits</li> <li>• Ability to simulate and verify the operation of sequential circuits</li> </ul>	<ul style="list-style-type: none"> <li>• Assignments</li> </ul>	5%	K(L)

## Tentative Class and Lab Schedule

Week	Topics
1	<ul style="list-style-type: none"> <li>• Introduction. Information Processing and representation. Digital vs. Analog quantities.</li> <li>• Number systems: Binary system.</li> <li>• Weighted Number Systems. Decimal, Binary, Octal and Hexadecimal.</li> <li>• Number base conversion (Dec to Bin, Oct, and Hex, General). Conversion (Bin to OCT, Hex).</li> </ul>
2	<ul style="list-style-type: none"> <li>• <b><u>Quiz# 1</u></b></li> <li>• Arithmetic in Binary and Hex (addition, subtraction &amp; Multiplication)</li> <li>• Decimal Codes: BCD, Excess-3, other codes, Character Storage, ASCII Code. Error Detection, Parity Bit.</li> <li>• Binary logic and gates, Truth tables, Boolean Algebra, Basic identities of Boolean algebra. Principle of duality, DeMorgan's Theorem.</li> <li>• Algebraic manipulation of Boolean expressions.</li> </ul>
3	<ul style="list-style-type: none"> <li>• <b><u>HW# 1 is due</u></b></li> <li>• Algebraic manipulation of Boolean expressions.</li> <li>• Canonical and Standard forms, Minterms, Maxterms, Sum of products &amp; Products of Sums.</li> <li>• 2-Level gate implementation SOP, POS, AOI, OAI), and multi-level logic.</li> <li>• From Truth tables to Boolean Expressions</li> </ul>
4	<ul style="list-style-type: none"> <li>• <b><u>Quiz# 2</u></b></li> <li>• Physical properties of gates: fan-in, fan-out, propagation delay. Timing diagrams.</li> <li>• Introduction to Verilog: Verilog Syntax, Definition of a Module, Gate Level Modeling, Using Modelsim simulation tool. Module Instantiation, Propagation Delay, Behavioral Modeling, Boolean Equation-Based Behavioral Models of Combinational Logic, Assign Statement, Propagation Delay &amp; Continuous Assignment, Test Bench Example.</li> <li>• Map method of simplification: 2, 3 and 4-variable K-Map.</li> </ul>
5	<ul style="list-style-type: none"> <li>• <b><u>HW# 2 is due</u></b></li> <li>• Implicants, Prime Implicants, Essential Prime Implicants.</li> <li>• POS simplification</li> <li>• Don't care conditions and simplification.</li> </ul>
	<b>Major I: Saturday, March 11, 2017</b>
6	<ul style="list-style-type: none"> <li>• Universal gates (NAND, NOR)</li> <li>• Implementation using <i>NAND and NOR gates</i>: 2-level &amp; multilevel implementation.</li> <li>• Exclusive-OR (XOR) and Equivalence (XNOR) gates, Odd and Even Functions, Parity generation and checking.</li> </ul>

7	<ul style="list-style-type: none"> <li>• <b><u>Quiz#3</u></b></li> <li>• Combinational Circuit Design Procedure &amp; Examples:</li> <li>• Code Converter.</li> <li>• BCD to 7-Segment Display Conversion</li> <li>• Half and Full Adders,</li> <li>• Ripple Carry Adder design and <i>Delay</i> analysis of RCA</li> </ul>
8	<ul style="list-style-type: none"> <li>• Signed Numbers: sign-magnitude, 1`s complement, and 2`s complement.</li> <li>• Signed Binary Arithmetic. (Addition and Subtraction in 2`s complement).</li> <li>• Binary Adder-Subtractor.</li> <li>• Carry Look-ahead adder.</li> </ul>
<b>Mid-Term Vacation: 2-6 April, 2017</b>	
9	<ul style="list-style-type: none"> <li>• <b><u>HW# 3 is due</u></b></li> <li>• Decoders 2x4, 3x8, 4x16. Designing large decoders from smaller decoders. Function implementation using decoders.</li> <li>• Encoders: Priority Encoders. Applications of decoders and priority encoders.</li> </ul>
10	<ul style="list-style-type: none"> <li>• <b><u>Quiz#4</u></b></li> <li>• Multiplexers: 2x1, 4x1. Constructing large MUXs from smaller ones. Function implementation using multiplexers.</li> <li>• Function implementation using multiplexers.</li> <li>• Magnitude Comparator</li> <li>• MSI Design Examples</li> </ul>
11	<ul style="list-style-type: none"> <li>• <b><u>HW#4 is due</u></b></li> <li>• Introduction to Verilog: Verilog Operators, Behavioral Description of an Adder, Always block, Procedural Assignment, If Statements, Case Statements, Comparator, Arithmetic &amp; Logic Unit. Multiplexor, Encoder, Priority Encoder, Decoder, Seven Segment Display Decoder.</li> </ul>
<b>Major 2: Saturday, April 29, 2017</b>	
12	<ul style="list-style-type: none"> <li>• <i>Sequential Circuits</i>: Latches, Clocked latches: SR, D, T and JK. Race problem in clocked JK-Latch.</li> <li>• <i>Flip-Flops</i>: Master-Slave, D-FF.</li> <li>• Designing flip-flops from D-FF.</li> </ul>
13	<ul style="list-style-type: none"> <li>• <b><u>Quiz#5</u></b></li> <li>• Analysis of Sequential Circuits. State table, State diagram.</li> <li>• Mealy vs. Moore machine.</li> <li>• <i>Sequential Circuit Design</i>. Design procedure, State diagrams and state tables.</li> <li>• <i>Asynchronous/Direct Clear and Set Inputs</i>. Setup, Hold, FF propagation delay. Calculation of maximum clock frequency.</li> </ul>

14	<ul style="list-style-type: none"> <li>• Verilog modeling of D-Latch, D Flip Flop – Synchronous Set/Reset, D Flip Flop–Asynchronous Set/Reset. Verilog Structural modeling of sequential circuits, Verilog FSM modeling.</li> <li>• Registers, Registers with parallel load.</li> <li>• Shift Registers. Bi-directional shift register. Applications of shift registers.</li> <li>• Synchronous Binary Counters: Up-Down Counters</li> <li>• Counters with Parallel load, enable, synchronous clear and asynchronous clear.</li> </ul>
15	<ul style="list-style-type: none"> <li>• <b><u>HW#5 is due</u></b></li> <li>• Use of available counters to build counters of different count.</li> <li>• Design with unused States</li> <li>• Combinational Circuit Implementation with ROM.</li> <li>• Sequential Circuit Implementation using ROMs.</li> <li>• <b><i>Review</i></b></li> </ul>