King Fahd University of Petroleum and Minerals
College of Computer Sciences and Engineering
Department of Computer Engineering

COE 202 Fundamentals of Computer Engineering (3-0-3)

Instructor: Dr. Marwan Abu-Amara
Office: 22-148-1
Phone: 1632
E-mail: marwan@kfupm.edu.sa
Term: 081 (1st term 2008–2009)
Prequisite: PHSY 101 (General Physics I)
Textbook: *Logic and Computer Design Fundamentals*, Morris Mano and Charles Kime, Third Edition, Prentice Hall International, 2004.
Course Notes: PDF files with 5 units divided into lessons.
Online CD: The CD contains all course lectures with animations and sound. The material is divided into 6 units with several lessons in each unit.
Office Hours: SMW 09:30 AM – 10:30 AM (or by appointment)
Web Site: http://faculty.kfupm.edu.sa/COE/marwan

Tentative Grading Policy:

- Homeworks **5%** (Each homework may carry a different weight)
- CAD Assignments **5%** (Each CAD assignment may carry a different weight)
- Quizzes **10%** (Each quiz may carry a different weight)
- Major Exam I **20%** (Monday November 10, 2008 from 07:00 PM to 09:00 PM)
- Major Exam II **25%** (Saturday December 27, 2008 from 07:00 PM to 09:00 PM)
- Final Exam **35%** (***Comprehensive***)

IMPORTANT NOTES:

- All KFUPM regulations and standards will be enforced. Attendance will be checked each class. The KFUPM rule pertaining to a DN grade will be strictly enforced (i.e. > **9 absences** will result in a DN grade). *Check your university e-mail regularly for warnings regarding your absences.*
- If you are late to the class for more than 5 minutes (i.e. arrive after 08:05 AM), you will **NOT be allowed to enter** the classroom and you will be considered absent for that class.
- Only university approved/certified excuses will be accepted, and should be presented **no later than 1 week** after absence.
- Use of cell phones and PDAs during class period and during exams is absolutely **prohibited**.
- Homeworks are to be submitted **in class** on the due date during the class period. Late homeworks will **NOT be accepted** (i.e. will get 0 credit).
- You have 48 hours to object to the grade of a homework, a quiz, or a major exam from the end of the class time in which the graded papers have been distributed back. If for some reason you cannot contact me within this period, send me an email requesting an appointment. The email should be sent within the 48-hour time period.
- **NO make up exams**. ALL homeworks and quizzes will be counted towards your grade.
- Final exam is **comprehensive**.

Tentative schedule:

Week	Subject
	Number System and Codes
1	Introduction. Information processing and representation. Digital vs Analog
	Number Systems. Binary, octal and hexadecimal numbers
2	Number base conversion (Dec to Bin, Oct, and Hex, General)
	Conversion (Bin, OCT, Hex), Binary & other System Arithmetic
	Signed Binary Number representation, Signed Mag, R's &(R-1)'s Complement
3	Signed Binary Addition and Subtraction. R's &(R-1)'s Complement
	Signed Binary Addition and Subtraction. R's &(R-1)'s Complement
	Codes. BCD, Excess-3, Parity Bits, ASCII & Uni-Codes
	Binary Logic & Gates
4	Binary logic and gates, Boolean Algebra, Basic identities of Boolean algebra.
	Boolean functions, Algebraic manipulation, Complement of a function.
	Canonical & Standard forms, Minterms & Maxterms, Sum of products, Product of Sums.
5	***Map method of simplification***: Two-, Three-, and Four-variable K-Map.
	Map method of simplification: Five, and Six-variable K-Map.
6	***Map manipulation:*** Essential prime implicants, Nonessential prime implicants, Simplification procedure.
	Don't care conditions and Simplification.
	Universal gates; ***NAND and NOR gates:*** 2-level implementation.
7	***Multilevel NAND Circuits***.
	Exclusive-OR (XOR) and Equivalence (XNOR) gates, Parity generation and checking.
	Combinational Logic
7	Combinational Logic, Design procedure. BCD-to-Excess 3 code Conversion.
8	BCD-to-7 Segment Display. Half and Full Adders.
	Design using MSI parts. Decoders, Decoder Expansion. Combinational Circuit implementation using decoders.
	Encoders & Priority Encoders *Magnitude Comparator*.
9	Multiplexers. Function Implementation using multiplexers, Demultiplexers
	Binary Adders: 4-Bit Ripple Carry Adder, Carry Look-Ahead Adder, Binary Adder-Subtractor.
	BCD Adder, Binary Multiplier.
	Sequential Circuits
10	***Sequential Circuits:*** Latches, SR and D-latch, Clocked latch.
	Flip-Flops: Master-Slave, Edge-Triggered. *Timing Diagrams*
11	***Flip-Flops Characteristic & Excitation Tables:*** D-FF, SR-FF, JK-FF, T-FF. ***Asynchronous/Direct Clear and Set Inputs***
	Setup, Hold, Enable times. Timing control and Clocks. Path delay constraints.
	Sequential Circuit Analysis: Input equations, State table.
12	***Sequential Circuit Design:*** Design procedure, Construction of state diagrams and state tables.
	Designing with D-FFs. Designing with unused states.
	Designing with JK-FFs, T-FFs
13	Sequential Circuit Design Examples.
	Registers & Counters
13	Registers, Registers with parallel load, Shift Registers.
	Shift register with parallel load, Bi-directional shift register.
14	***Ripple Counters:*** Up-Down Counters. Synchronous Binary Counters: ***Counters with JK-FF, Counters with D-FF.***
	Serial & Parallel Counter, Up-Down Binary Counter, Binary Counter with Parallel Load.
	Other Counters: BCD Counter, Arbitrary Count Sequence.
	Memory & PLDs
15	***Memory and Programmable Logic Devices:*** Read-Only Memory.
	Combinational Circuit Implementation with ROM.
	Programmable logic array, Programmable array logic, Prog. logic devices, FPGAs

COE 202 Digital Logic Design

Course Learning Outcomes Table

Course Learning Outcomes	Outcome Indicators & Details	Assessment Methods and Metrics	Min. Weight	ABET 2000
1. Ability to use math and Boolean algebra in performing computations in various number systems and simplification of Boolean algebraic expressions.	➢ Represent integer and fractional values in various number systems ➢ Convert number representation from one system to another ➢ Perform arithmetic operations in various number systems ➢ Represent data in different binary codes including error detecting codes ➢ Simplify Boolean expressions using Boolean algebra & identities	➢ Assignments ➢ Quizzes ➢ Exams	20%	A (H)
2. Ability to design efficient combinational and sequential logic circuit implementations from functional description of digital systems.	➢ Derive gate-level implementation of a given Boolean expression and vice versa ➢ Ability to build larger combinational functions using predefined modules (e.g., decoders, multiplexers, adders, Magnitude comparators.) ➢ Ability to build a state diagram / table for both Moore & Mealy models from functional description ➢ Ability to design & implement Moore & Mealy model synchronous sequential circuits using different Flip-Flop types. ➢ Ability to draw timing diagrams for major signals of both sequential and combination circuits	➢ Assignments ➢ Quizzes ➢ Exams	50%	C (H)
3. Ability to use CAD tools to simulate and verify logic circuits.	➢ Ability to simulate and verify the operation of combinational circuits ➢ Ability to simulate and verify the operation of sequential circuits	➢ Assignments	5%	K (L)

ABET 2000 COE Program Learning Outcomes

(a) an ability to apply knowledge of mathematics, science, and engineering

(b) an ability to design and conduct experiments, as well as to analyze and interpret data

(c) an ability to design a system, component, or process to meet desired needs

(d) an ability to function as an effective team member

(e) an ability to identify, formulate, and solve engineering problems

(f) an understanding of professional and ethical responsibility

(g) an ability to communicate effectively

(h) the broad education necessary to understand the impact of engineering solutions in a global and societal context

(i) a recognition of the need for, and an ability to engage in life-long learning

(j) knowledge of contemporary issues

(k) an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice

(l) Knowledge of Probability and Statistics and their applications in Computer Engineering

(m) Knowledge of Discrete Mathematics

(n) The ability to design a system that involves the integration of hardware and software components