CSC 203 – Digital Design

**COURSE PARTICULARS**

- **Course Code:** CSC 203
- **Course Title:** Digital Design
- **No. of Units:** 3
- **Course Duration:** Two hours of theory and 3 hours of practical per week for 13 weeks.
- **Status:** Compulsory
- **Course Email Address:** csc203@futa.edu.ng
- **Course Webpage:** [http://www.csc.futa.edu.ng/courseschedule.php?coursecode=CSC%20204](http://www.csc.futa.edu.ng/courseschedule.php?coursecode=CSC%20204)
- **Prerequisite:** NONE

**COURSE INSTRUCTORS**

**Prof. S.O. Falaki**  
*Dept. of Computer Science,  
The Federal University of Technology, Akure, Nigeria.*  
**Phone:** +2348034034202  
**Email:** osfalaki@futa.edu.ng

**Dr. A.O. Adetunmbi**  
*Dept. of Computer Science,  
The Federal University of Technology, Akure, Nigeria.*  
**Phone:** +08039617525  
**Email:** aoadetunmbi@futa.edu.ng

**Mr. E.A. Ibidunmoye**  
*Dept. of Computer Science,  
The Federal University of Technology, Akure, Nigeria.*  
**Phone:** +08066851683  
**Email:** eaibidunmoye@futa.edu.ng

and

**Mr. B.S. Ige**  
*Dept. of Computer Science,  
The Federal University of Technology, Akure, Nigeria.*  
**Phone:** 08067113906  
**Email:** bsige@futa.edu.ng
COURSE DESCRIPTION

This course provides an introduction to the design of computer components. Students will learn the necessary mathematical background to carry out logic design. Students will design and implement simple combinational and sequential circuits.

COURSE OBJECTIVES

The specific objectives of this course are to learn:
- Basics of Information and Digital abstractions – number systems and codes,
- CMOS technology and Boolean algebra;
- Combinational, sequential and register transfer logic,
- Hardware Descriptive language, Arithmetic/logic unit, Memories,
- Basics of computer organization, Input-output and Microprocessors.

COURSE LEARNING OUTCOMES / COMPETENCIES

Upon completion of CSC 203, students will be able to:

1). perform arithmetic operations in many number systems
2). Characterize the logic function of combinational devices using CMOS, ROM, or PLA technologies.
3). Explain synthesis issues for combinational devices using CMOS, ROM, or PLA technologies from their functional specification.
4). Explain synthesis of acyclic circuits from combinational components.
5). Calculate performance characteristics of acyclic circuits with combinational components.
6). Explain and calculate performance characteristics of single-clock sequential circuits.
7). Design, debug, and test combinational circuits of the complexity of an arithmetic logic unit.
8). Design, debug, and test a controller for a finite-state machine.
9). Pipeline a combinational circuit for improved throughput.
10). Understand issues affecting microprocessor instruction set design.
11). Complete and debug the design of a simple CPU with a given RISC-based instruction set.
12). Measure the memory access performance of a processor, and tune cache design parameters to improve performance.
13). Design and conduct experiments, as well as to analyze and interpret data,
14). Design a system, component, or process to meet desired needs within realistic constraints,
15). Identify, formulate, and solve engineering problems,

GRADING SYSTEM FOR THE COURSE
This course will be graded as follows:

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignments</td>
<td>10%</td>
</tr>
<tr>
<td>Practical</td>
<td>25%</td>
</tr>
<tr>
<td>Test(s)</td>
<td>15%</td>
</tr>
<tr>
<td>Final Examination</td>
<td>50%</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>

**GENERAL INSTRUCTIONS**

**Attendance:** It is expected that every student will be in class for lectures and also participate in all practical exercises. Attendance records will be kept and used to determine each person’s qualification to sit for the final examination. In case of illness or other unavoidable cause of absence, the student must communicate as soon as possible with any of the instructors, indicating the reason for the absence.

**Academic Integrity:** Violations of academic integrity, including dishonesty in assignments, examinations, or other academic performances are prohibited. You are not allowed to make copies of another person’s work and submit it as your own; that is plagiarism. All cases of academic dishonesty will be reported to the University Management for appropriate sanctions in accordance with the guidelines for handling students’ misconduct as spelt out in the Students’ Handbook.

**Assignments and Group Work:** Students are expected to submit assignments as scheduled. Failure to submit an assignment, as at when due, will earn you zero for that assignment. Only under extenuating circumstances, for which a student has notified any of the instructors in advance, will late submission of assignments be permitted.

**Code of Conduct in Lecture Rooms and Laboratories:** Students should turn off their cell phones during lectures. Students are prohibited from engaging in other activities (such as texting, watching videos, etc.) during lectures. Food and drinks are not permitted in the laboratories.

**LUGGAGE, FOOD OR DRINKS** Luggage, food or drinks are not allowed in the laboratory. On entering the lab., there are lockers on the right hand side where all your luggage can be safely kept. Do not bring your laptops to the Laboratory during lab. hours.

**READING LIST**


VHDL Tutorial (2009) Peter J. Ashedent. Softcopy available online

Falaki S.O. Lecture Materials on Digital Design

COURSE OUTLINE

<table>
<thead>
<tr>
<th>Lectures #</th>
<th>Topic</th>
</tr>
</thead>
</table>
| 1          | Course overview and objectives,  
Basis of information, number systems and codes |
| 2          | Digital Arithmetic Operations  
CMOS technology; gate design, timing |
| 3          | Logic gates, Boolean Algebra  
Hardware description language |
| 4          | Canonical forms: synthesis , simplification  
Verilog hardware description language,  
combinational logic in Verilog – testbenches |
| 5          | Sequential Logic Circuits:  
Flip flops, registers , clocks  
Sequential circuits using HDL |
| 6          | Digital Arithmetic circuits  
Logical Operations with VHDL |
| 7          | Counters and Registers  
Simple VHDL implementation of counters |
<table>
<thead>
<tr>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finite state Machines, synchronization, metastability</td>
</tr>
<tr>
<td>VHDL implementation</td>
</tr>
<tr>
<td>Finite state Machines cont.</td>
</tr>
<tr>
<td>QUIZ</td>
</tr>
<tr>
<td>Memory Basics and Timing</td>
</tr>
<tr>
<td>Analog Building Blocks</td>
</tr>
<tr>
<td>Models of Computation, Programmable Logic Device Architecture</td>
</tr>
<tr>
<td>Verilog for Digital System Design</td>
</tr>
<tr>
<td>Simple design Project Discussions</td>
</tr>
<tr>
<td>Wrapup Lecture</td>
</tr>
</tbody>
</table>

**Laboratory Schedule**

<table>
<thead>
<tr>
<th>Lab. #</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab1</td>
<td>Demonstrating the operation and characteristics of a TTL logic gate (NAND gate-7400) and implementing the three basic logic functions using NAND gate.</td>
</tr>
<tr>
<td>Lab2</td>
<td>Demonstrating the operation and characteristics of a CMOS logic gate (NOR gate-4001) and implementing the three basic logic functions using NOR gate and using verilog.</td>
</tr>
<tr>
<td>Lab3</td>
<td>TTL and CMOS, NAND, NOR, and XOR gates are used to implement any logic functions and implementing the Boolean algebra to reducing logic circuits to their minimum configuration.</td>
</tr>
<tr>
<td>Lab4</td>
<td>Implementing and investigating the operation of a 2-bit full adder, 4-bit full adder, and 4-bit full subtractors from basic combinational 74LS logic and wired using Verilog on Xilinx Sparta 3E FPGA board</td>
</tr>
<tr>
<td>Lab5</td>
<td>Testing the operation of 74x138 decoder, and using it as demultiplexer. Implement the Boolean functions using 74x138 decode and 74151 multiplexer.</td>
</tr>
<tr>
<td>Lab 6</td>
<td>Demonstrating the operations and characteristics of D-type flip-flop and JK-type flip-flop. Verify that the flip-flop is a bistable multivibrator (has two stable state). And it has two complementary output states.</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Lab 7</td>
<td>Demonstrating the operations and characteristics of a binary counter (up counter /down counter).</td>
</tr>
<tr>
<td>Lab 8</td>
<td>Design Project commenced</td>
</tr>
<tr>
<td></td>
<td>Group of students will assign a Problem to solve which involves the application of materials or content of this course.</td>
</tr>
</tbody>
</table>