

NEW YORK CITY COLLEGE OF TECHNOLOGY
The City University of New York

DEPARTMENT: Electrical and Telecommunications
Engineering Technology

**SUBJECT CODE
AND TITLE:** TCET 3120/TC 520 Switching and Automata Theory

COURSE DESCRIPTION: The course covers synchronous state machines. VHDL techniques are used to cover state transition analysis, synthesis and optimization techniques. VHDL concepts are used to develop simulation wave-shapes of all the circuits involved. The course involved with the study of combinational networks, counters, shift registers and sequential machines.

PREREQUISITE: TCET 2242/TC 430 or EET 2261/ET 482

TEXTBOOK: Digital Design with CPLD Applications and VHDL
By Robert Dueck - Second Edition

**COURSE OBJECTIVES/
COURSE OUTCOMES:** Upon successful completion of this course, students should be able to:

1. Analyze, simplify, and design Combinational Logic Circuits (ABET Criteria 2a, 2b, 2f, 2k).
2. Describe the structure of several types of Programmable Logic Devices-PLDs (ABET Criteria 2a, 2b, 2c, 2k, 2l).
3. Use Altera's Quartus II Software to design, simulate, and implement PLDs (ABET Criteria 2a, 2b, 2f, 2k, 2l).
4. Understand the basics of VHDL and use it in the programming of digital systems (ABET Criteria 2a, 2b, 2c, 2f, 2k, 2l).
5. Analyze and design Combinational Logic Functions such as Decoders, Encoders, Multiplexers, and Comparators (ABET Criteria 2a, 2b, 2c, 2f, 2k, 2l).
6. Analyze and design Sequential Logic Circuits such as Latches, Flip-flops, Counters, and shift registers.
7. Understand the concepts of State Machine Design and develop Mini-Processors (ABET Criteria 2a, 2b, 2c, 2d, 2e, 2f, 2k, 2l)
8. Understand the electrical characteristics of logic gates, Timing Parameters, and Arithmetic Circuits (ABET Criteria 2a, 2b, 2c, 2d, 2e, 2f, 2k, 2l).

TOPICS: Topics include Basic Principles of Digital Systems, Logic Functions and Gates, Boolean Algebra and Combinational Logic, Introduction to PLDs and Quartus II, Introduction to VHDL, Combinational Logic Functions, Digital Arithmetic and Arithmetic Circuits, Introduction to Sequential Logic, Counters and Shift Registers, State Machine Design, Logic Gate Circuitry.

CLASS HOURS: 2

LAB. Hours: 3

CREDITS: 3

Prepared by: Professor H. Marandi
October 2006

Course Coordinator: Professor H. Marandi
(718) 260-5312 E-mail: Hmarandi@citytech.cuny.edu

EXPERIMENT: TCET 3120/TC 520

1. DIP Integrated Circuits
2. Expanding Logic Gates
3. Pulsed Operation of Logic Gates
4. Boolean Algebra: SOP Forms
5. Boolean Algebra: DeMorgan Equivalent Forms
6. Introduction to Quartus II
7. Introduction to VHDL
8. Binary Decoders
9. Binary and 7-Segment Decoders in VHDL
10. Priority Encoders
11. Multiplexers

GRADING POLICY: TCET 3120/TC 520

Homework and class participation	10%
Lab Report and Presentation	25%
Exams and Midterm Exam:	40%
Final Exam	25%

<u>Letter Grade</u>	<u>Numerical Grade Ranges</u>	<u>Quality</u>
A	93-100	4.0
A-	90-92.9	3.7
B+	87-89.9	3.3
B	83-86.9	3.0
B-	80-82.9	2.7
C+	77-79.9	2.3
C	70-76.9	2.0
D	60-69.9	1.0
F	59.9 and below	0.0

<p style="text-align: center;"><u>Assessment</u></p> <p>The following assessment techniques are correlated to the course objectives as follows: In addition, each assessment technique incorporates one or more of the following ABET Criterion 2 outcomes (2a, 2b, 2c, 2d, 2e, 2f, 2k, 2l).</p> <p><u>Course Objectives</u></p> <p>1. Analyze, simplify, and design combinational logic circuits.</p>	<p style="text-align: center;"><u>Assessment</u></p> <p>Using Digital Design Software, the student will be able to:</p> <ol style="list-style-type: none"> 1.1 Describe the configuration of several basic logic gates in DIP Integrated Circuits. 1.2 Determine how basic logic gates can be used to enable or inhibit time-varying digital signals by examining the gate truth tables. 1.3 Monitor the pulsed behavior of logic gates with LESs and with an oscilloscope. 1.4 Use Boolean algebra to simplify a logic gate network and to prove that two gate networks are equivalent. 1.5 Use DeMorgan equivalent forms of logic gates to simplify the Boolean expression of a logic gate network.
<p>2. Describe the structure of several types of Programmable Logic Devices (PLDs).</p>	<ol style="list-style-type: none"> 2.1 Create a project in Quartus II. 2.2 Use Quartus Block Editor to enter a graphical design in Quartus II.
<p>3. Use Altera's Quartus II software to design, simulate, and implement PLDs.</p>	<ol style="list-style-type: none"> 3.1 Compile and simulate the design. 3.2 Program an Altera CPLD with the Design. 3.3 Test the design on a CPLD test Board to determine the truth table.
<p>4. Understand the basics of VHDL and use it in the programming of digital systems.</p>	<ol style="list-style-type: none"> 4.1 Enter a simple combinational logic circuit in VHDL using the Quartus Text Editor. 4.2 Assign target device and pin numbers and compile a VHDL design file. 4.3 Write simulation criteria for a VHDL design entity and create a Simulation to verify the Correctness of the design.

<p>5. Analyze and design Combinational Logic Functions such as Decoders, Encoders, Multiplexers, and Comparators.</p>	<p>5.1 Enter the design for a binary decoder in Quartus II as a Block Diagram File.</p> <p>5.2 Create a Quartus II simulation of a binary decoder.</p> <p>5.3 Enter the design for a binary decoder in Quartus II as a VHDL design entity.</p> <p>5.4 Use VHDL to create 7-segment decoder.</p> <p>5.5 Enter a VHDL design for a BCD priority encoder.</p> <p>5.6 Write simulation criteria for the BCD priority encoder and create a simulation in QuartusII.</p> <p>5.7 Enter the logic circuit of a 4-to-1 multiplexer (MUX) as a block diagram file, using Altra's Quartus II CPLD design software.</p> <p>5.8 Create a Quartus II simulation file for the 4-to-1 multiplexer.</p>
<p>6. Analyze and design Sequential Logic Circuits such as Latches, Flip- Flops, Counters, and Shift Registers.</p>	<p>6.1 Draw circuits, function tables, and timing diagrams of edge-triggered D, JK, and T flip-flops.</p> <p>6.2 Use Quartus II to create simple circuits and simulations with D latches and D, JK, and T flip-flops.</p> <p>6.3 Use simple latch and flip-flop designs using VHDL.</p> <p>6.4 Implement various counter control functions, such as parallel load, clear, count enable, and count direction both in Block Diagram Files and in VHDL.</p> <p>6.5 Design shift registers, ring counters, Johnson counters using the Quartus II simulation tool.</p>

<p>7. Understand the concepts of State Machine Design and develop Mini-Processors.</p>	<p>7.1 Describe the components of a state machine.</p> <p>7.2 Distinguish between Moore and Mealy implementations of state machines.</p> <p>7.3 Create simulations in Quartus II to verify a state machine design.</p> <p>7.4 Design state machine applications, such as switch debouncer, single- pulse generator, and traffic light controller.</p>
<p>8. Understand the electrical characteristics of logic gates, Timing Parameters, and Arithmetic Circuits.</p>	<p>8.1 Derive the logic gate circuits for full and half adders and subtractors, given their truth tables.</p> <p>8.2 Draw circuits to perform BCD arithmetic and explain their operation.</p> <p>8.3 Use VHDL to program CPLD devices to perform various arithmetic functions, such as parallel adders and 1s complementers.</p>

Electrical and Telecommunications Engineering Technology_TCET3120/TC520

WEEK	TOPIC	READING ASSIGNMENT	HOMEWORK
1 & 2	<ul style="list-style-type: none"> • Course outline, lateness, absence, and classroom policy • Review of the basic principles of digital systems • Review of logic functions and gates • DeMorgan equivalent forms • Review of Boolean Algebra • Describe the behavior of tristate buffers. 	<p>Chapter 1 & 2</p> <p>Lab.#1 DIP Integrated Circuits</p>	<p>1.20, 1.21, 1.22, 1.23, 1.24, 1.25, 1.26</p>
3	<ul style="list-style-type: none"> • Introduction to Combinational logic design • Explain the relationship between Boolean expression, logic diagram, and truth table of a logic gate network • Write sum of products and product of sums forms • Simplify logic diagrams using the Karnaugh map technique • Redraw a logic diagram using all-NAND or all-NOR implementations 	<p>Chapter 3</p> <p>Lab. #2 Expanding Logic Gates</p>	<p>2.22, 2.24, 2.25, 2.33, 2.35, 2.37, 2.39</p>
4	<ul style="list-style-type: none"> • Introduction to Programmable Logic Devices (PLDs) • Introduction to Quartus Design Software • Describe the structure of Programmable Logic Arrays (PALs) • Draw fuses on the logic diagram of a PAL to implement logic functions • Use Quartus II PLD design software to enter simple combinational circuits using schematic capture 	<p>Chapter 4</p> <p>Sections 4.1-4.6</p> <p>Lab. #3 Pulse Operation of Logic Gates</p>	<p>4.4, 4.8, 4.12, 4.14, 4.16,</p>
5	<ul style="list-style-type: none"> • More on PLDs and Quartus II design software • Assign device and pin numbers to schematic designs • Program Altera PLDs via a parallel port download cable 	<p>Chapter 4 Sections 4.7-4.10</p>	<p>4.16, 4.24, 4.25, 4.27, 4.28</p>

	error-checking system and draw simple parity-generation and checking circuits	Lab. #6 Introduction to VHDL	
9	<ul style="list-style-type: none"> • Introduction to sequential circuits and difference between combinational and sequential circuits • Define the set and reset functions of an RS latch • Draw circuits, function tables and timing diagrams of NAND and NOR latches • Explain the effect of each possible input combination to a NAND and NOR latch, including set, reset, and no change functions, as well as the ambiguous or forbidden input conditions • Design circuit applications that employ NAND and NOR latches • Describe the use of ENABLE input of a gated RS or D latch as an Enable/Inhibit function and as a synchronizing function 	<p>Chapter 8</p> <p>Sections 8.1 - 8.3</p> <p>Lab. #7 Standard Wiring Configuration for Altera</p>	8.1, 8.7, 8.15, 8.19,8.20
10	<ul style="list-style-type: none"> • Explain the concept of edge-triggering and why it is an improvement over level-sensitive enabling • Draw circuits, function tables, and timing diagrams of edge-triggered D, JK, T flip-flops • Describe the toggle function of a JK flip-flop and a T flip-flop • Describe the operation of the asynchronous preset and clear functions of D, JK, and T flip-flops and be able to draw timing diagrams showing their functions • Use Quartus II to create simple circuits and simulations with D latches 	<p>Chapter 8</p> <p>Sections 8.4-8.8</p>	8.26, 8.32, 8.34, 8.36, 8.46

	<ul style="list-style-type: none"> and D, JK, and T flip-flops • Create simple latch and flip-flop design using VHDL • Explain the structure of an Output Logic MarcoCell (OLMC) • State the differences between the Generic Logic Array (GAL) and standard PAL 	<p>Lab. #8 Binary Decoders</p>	
11	<ul style="list-style-type: none"> • Introduction to Digital Counters and Shift Registers • Explain the concept of a synchronous counter • Determine the modulus of a counter • Draw the count sequence table, state diagram, and timing diagram of a counter • Determine the recycle point of a counter sequence • Calculate the frequencies of each counter output, giving the input clock frequency • Draw a circuit for a full-sequence synchronous counter • Use Quartus II and VHDL to design and simulate synchronous counters 	<p>Chapters 9 Sections 9.1-9.6</p> <p>Lab. #9 Binary and Seven-Segment Decoders</p>	<p>9.2, 9.14, 9.16, 9.27, 9.36</p>
12	<ul style="list-style-type: none"> • Introduction to Shift Registers • Draw logic circuit of a serial shift register and determine its counts over time giving any input data • Draw a timing diagram showing the operation of a serial shift register • Draw a logic circuit and timing diagram of a general parallel load shift register • Draw the general logic circuit and timing diagrams of bi-directional shift register and explain the concepts of right-shift and left-shift • Describe the operation of a universal shift register 	<p>Chapters 9 Sections 9.9-9.9</p>	<p>9.48, 9.54, 9.55</p>

