

EECC 341 Learning Objectives

For each topical area, students should be able to

Boolean algebra

- Prove or disprove simple 1,2,3-variable Boolean algebraic theorems
- Explain the duality property (note the difference between the duality for equations versus that for expressions)
- Use AND, OR, INV and XOR operators to express a logical function
- Convert a Sum-of-Product expression to a Product-of-Sum expression and vice versa
- Simplify a Boolean expression following the Boolean algebraic rules
- Translate a simple problem specification into a truth table
- Determine the Sum-of-Minterm (Canonical Sum) and the Product-of-Maxterm (Canonical Product) expressions (in both the algebraic expression form and the notation form) from a given truth table
- Show that NAND and NOR are both self-sufficient

Numbering Systems and Binary Arithmetic

- Convert between binary, decimal, and hexadecimal numbers (for both integers and fractional numbers)
- Perform binary addition and subtraction
- Explain the sign-magnitude, 1's complement and 2's complement systems
- Express an integer number using the sign-magnitude, 1's complement and 2's complement systems (with given number of digits)
- Determine the range supported by a 2's complement number with given number of digits
- Provide the formulas that determine the value of a binary number or a 2's complement number
- Perform addition and subtraction for 2's complement numbers
- Design a full adder circuit and a n-bit carry-ripple adder using the full adder
- Explain and compare the difference between a carry-ripple and a carry-look-ahead adder
- Design a n-bit adder/subtractor for 2's complement numbers
- Design an overflow detection circuit for 2's complement adder/subtractor
- Use the Cout bit to correct the invalid SUM result if an overflow were to occur

Karnaugh Map

- Fill in K-map from a given Boolean expression or from a given Canonical Sum/Product notation form (with or without Don't cares)
- Use 3,4,5-variable K-map to derive a minimum Sum-of-Product expression and a minimum Product-of-Sum expression
- Explain what 6-variable K-map may be used
- Use K-map with Don't-cares
- Explain what static-0 and static-1 hazards are for 2 level OR-AND and AND-OR circuits
- Derive a hazard-free 2-level AND-OR (OR-AND) circuit

Circuit

- Derive a Boolean expression from a given circuit diagram
- Draw a circuit diagram from a given Boolean expression (with or without restriction on the types of gates one can use)
- Convert a 2-level AND-OR circuit into a 2-level NAND-NAND circuit

- Convert a 2-level OR-AND circuit into a 2-level NOR-NOR circuit
- Implement a general circuit using only NAND gates and INVs (or only NOR gates and INVs), with or without the restriction on the number of inputs
- Realize a Boolean expression using different types of logic gates, including AND, OR, INV, NAND, NOR and XOR
- Design a circuit with multiple outputs using common logic gates
- Explain what generalized XOR (with more than 2 inputs) is
- Explain how to use XOR as an inverter and as a buffer (output equals the input)

Decoder/Encoder/Mux/Demux

- Explain the functionality of the binary n -to- 2^n decoder, the binary 2^n -to- n encoder, the 2^n -to-1 multiplexer and the 1-to- 2^n demultiplexer with or without the enable signal
- Design the binary n -to- 2^n decoder, the 2^n -to-1 multiplexer and the 1-to- 2^n demultiplexer (with or without the enable signal) using basic logic gates
- Design a general digital circuit using n -to- 2^n decoder or 2^n -to-1 multiplexer, where the number of inputs to the circuit may be n or $n+1$ and there may be multiple output signals from the circuit
- Design a demultiplexer using a decoder (with or without the enable signal)

Latches and Flipflops

- Explain the concept of metastability and how it is used to implement a memory element
- Explain the operation of rising/falling edge triggered (flipflops) and level-1/level-0 triggered (latches) sequential circuit components
- Design and explain the functionality of the S-R latch, the D-latch, the D-FF, the J-K FF and the T-FF with basic logic gates
- Explain the effect of propagation delay to the operation of latches and flipflops, i.e., the setup and holding times
- Explain and use the ACTIVE HIGH or ACTIVE LOW preset, reset and clock signals

Sequential Circuit Analysis and Design

- Explain the difference between a combinational and a sequential circuit
- Explain the difference between a Moore and a Mealy machine
- Design binary counters using T-FFs (note the various version of binary counters)
- Design shift-registers using D-FFs
- Determine the various equations (i.e., characteristic, excitation and output equations), the state table, the state diagram and an exemplary timing diagram for a given sequential circuit
- Determine the state diagram and the state table based on a timing diagram
- Design a correct state diagram and the associated circuit based on a sequential circuit description (for simple sequential circuits)

Laboratory Activities

- Explain the use of voltage to represent logic high and low
- Explain how the input and output stages work and used for testing digital logic
- Draw a schematic circuit diagram using PSPICE
- Design, compile and simulate a circuit using Altera Max+Plus
- Explain what a characteristic curve is and draw a conceptual characteristic curve for a 2-input AND gate with one input tied to Vcc or Gnd
- Explain the effect of loading on TTL IC circuit voltage output
- Interpret the data sheets for basic TTL ICs

- Wire TTC ICs to implement (and debug) a digital logic circuit and test it using simple switches and LEDs
- Explain, design and use ICs with ACTIVE LOW pins