



**COURSE OUTLINE : ENGR 132**

**D Credit – Degree Applicable**

**COURSE ID 010217**

**Cyclical Review: July 2020**

**COURSE DISCIPLINE :** ENGR  
**COURSE NUMBER :** 132  
**COURSE TITLE (FULL) :** Introduction to Digital Electronics  
**COURSE TITLE (SHORT) :** Intro to Digital Electronics

**CATALOG DESCRIPTION**

ENGR 132 provides an introduction to the basic concepts and theory of digital electronics. Binary number system is covered along with Boolean functions and their simplification. Combinational and sequential logic and digital system design using simulation software is included. Students also learn how to build, test, and analyze simple digital circuits in the laboratory, using test equipment such as power supply, multimeter, signal/function generator, and oscilloscope.

Total Lecture Units: 3.00

Total Laboratory Units: 1.00

**Total Course Units: 4.00**

Total Lecture Hours: 54.00

Total Laboratory Hours: 54.00

Total Laboratory Hours To Be Arranged: 0.00

**Total Contact Hours: 108.00**

**Total Out-of-Class Hours: 108.00**

Recommended Preparation: MATH 102 or MATH 110, or equivalent.



**ENTRY STANDARDS**

	<b>Subject</b>	<b>Number</b>	<b>Title</b>	<b>Description</b>	<b>Include</b>
1	MATH	102	Trigonometry	solve trigonometric equations;	Yes
2	MATH	102	Trigonometry	apply the laws of sines and cosines to solve application problems;	Yes
3	MATH	110	Precalculus	solve equations including rational, linear, polynomial, exponential, absolute value, radical, and logarithmic;	Yes
4	MATH	110	Precalculus	solve linear, non-linear, and absolute value inequalities;	Yes
5	MATH	110	Precalculus	solve linear and non-linear systems of equations and inequalities;	Yes
6	MATH	110	Precalculus	apply the laws of sines and cosines to solve application problems;	Yes
7	MATH	110	Precalculus	represent a vector (a quantity with magnitude and direction) in the form $\langle a,b \rangle$ and $ai+bj$ .	Yes

**EXIT STANDARDS**

- 1 Demonstrate digital circuit elements such as logic gates, counters, and flip flops;
- 2 design, build, analyze, and test digital electrical circuits with logic gates, counters, and flip flops;
- 3 use laboratory test equipment such as power supply, multimeter, signal/function generator, and oscilloscope, to test electrical circuits.
- 4 convert from different number systems and demonstrate the applications of each system;
- 5 perform operations using Boolean algebra and differentiate the representation of logical functions;
- 6 analyze sequential circuits and differentiate between synchronous and asynchronous types;
- 7 draw state tables of sequential circuits;
- 8 identify the types of shift registers and apply them to data transfer circuit designs;
- 9 explain the basic units of input/output, memory, control, and arithmetic logic unit (ALU) that make up a computer.

**STUDENT LEARNING OUTCOMES**

- 1 design combinational circuits using logical operations from Boolean algebra;
- 2 analyze number systems, computer arithmetic, and binary codes;
- 3 design, build and analyze digital electrical circuits using electrical engineering software.



**COURSE CONTENT WITH INSTRUCTIONAL HOURS**

	<b>Description</b>	<b>Lecture</b>	<b>Lab</b>	<b>Total Hours</b>
1	<b>Introductory Concepts</b> <ul style="list-style-type: none"> <li>• Introduction to digital systems</li> <li>• Introduction to numbering systems</li> <li>• Conversions of numbers</li> <li>• Binary numbers</li> <li>• Numerical representation</li> <li>• Memory</li> <li>• Applications to computing</li> </ul>	6	6	12
2	<b>Fundamentals of Boolean Algebra</b> <ul style="list-style-type: none"> <li>• Theorems of Boolean Algebra</li> <li>• Binary number system</li> <li>• Hexadecimal number system</li> <li>• Boolean functional notation</li> <li>• Logic Gates</li> <li>• Logic operations</li> <li>• Integrated Circuits (IC's)</li> </ul>	7	7	14
3	<b>Gate-Level Minimization</b> <ul style="list-style-type: none"> <li>• optimal gate-level implementation</li> <li>• Two, three and four variable K-map (Karnaugh map)</li> <li>• Prime implicant chart</li> <li>• Representation of logical function by logical expression</li> <li>• Canonical sum of products and products of sums</li> <li>• Design of combinational logic</li> <li>• Integrated circuits (IC's) definition</li> </ul>	7	7	14



4	<p>Sequential Logic Circuits</p> <ul style="list-style-type: none"> <li>• Synchronous and asynchronous circuits</li> <li>• AND, NAND (not and), OR, and NOR (not or) gate circuits</li> <li>• Clocked set-reset</li> <li>• D flip-flop circuits (Data flip-flop)</li> <li>• Toggle flip-flops</li> <li>• J-K (set-reset) flip-flop circuits</li> <li>• State table and state reduction and assignment</li> <li>• Synthesizable Hardware Descriptive Language (HDL) models of sequential circuits</li> <li>• Latches</li> <li>• Mealy and Moore models of finite state machines</li> </ul>	7	7	14
5	<p>Combinational Logic</p> <ul style="list-style-type: none"> <li>• Definition and analysis of combinational circuits</li> <li>• Binary adder-subtractor and multiplier</li> <li>• Decimal adder</li> <li>• Gate-level modeling</li> <li>• Data-level modeling</li> <li>• Overflow</li> <li>• Encoders and decoders</li> <li>• Multiplexers</li> <li>• Behavior modeling and logic simulation</li> <li>• Logic simulation</li> <li>• Writing a simple test-bench</li> </ul>	7	7	14
6	<p>Registers</p> <ul style="list-style-type: none"> <li>• Counters</li> <li>• Shift registers</li> <li>• Ripple and synchronous counters</li> <li>• Types of counters</li> <li>• Hardware descriptive language (HDL) models</li> <li>• HDL models for registers and counters</li> <li>• HDL models of combinational circuits</li> </ul>	7	7	14



7	<p>Memory and Programmable Logic</p> <ul style="list-style-type: none"> <li>• Random-Access Memory (RAM)</li> <li>• Read-only memory</li> <li>• Decoding memory</li> <li>• Error detection and correction</li> <li>• Programming logic arrays</li> <li>• Array logic</li> <li>• Sequential devices</li> </ul>	7	7	14
8	<p>Register and Transfer Level Designing</p> <ul style="list-style-type: none"> <li>• Register transfer level (RTL)</li> <li>• Algorithmic state machines (ASM)</li> <li>• HDL</li> <li>• Multiplexers</li> <li>• Race-free design</li> <li>• Latch-free design</li> <li>• Control systems logic</li> <li>• Sequential binary multiplier</li> </ul>	6	6	12
				<b>108</b>

**OUT OF CLASS ASSIGNMENTS**

- 1 homework (e.g. calculations and circuit diagrams)
- 2 laboratory reports (e.g. lab report on digital logic gates experiment);
- 3 project (e.g. using an integrated circuit, IC, design and build a memory unit such as an IC RAM, Random Access Memory).

**METHODS OF EVALUATION**

- 1 quizzes;
- 2 midterm exam(s);
- 3 final exam.

**METHODS OF INSTRUCTION**

- Lecture
- Laboratory
- Studio
- Discussion
- Multimedia



**COURSE OUTLINE : ENGR 132**

**D Credit – Degree Applicable**

**COURSE ID 010217**

**Cyclical Review: July 2020**

- Tutorial
- Independent Study
- Collaboratory Learning
- Demonstration
- Field Activities (Trips)
- Guest Speakers
- Presentations

**TEXTBOOKS**

Title	Type	Publisher	Edition	Medium	Author	IBSN	Date
Digital Systems: Principles and Applications	Required	Harlow : Pearson, 2018	12	Print	Tocci, Ronald J., et al.	978129216 2003 129216200 7	2018
Digital Design	Required	Pearson	6	Print	M. Morris Mano	ISBN: 978013454 9897	2018