DE Blueprint

The purpose of this assessment is to …

Table of Specifications

Days & % of Coverage: Unit	Days & % of Coverage: Lesson	Unit	Lesson	Knowledge and Skills	Item Type (ex. multiple choice, performance, true false, essay, etc.)	Complexity Webb's DOK: #Level 1 (Recall)	Complexity Webb's DOK: #Level 2 (Skill/ Concept)	Complexity Webb's DOK: #Level 3 (Strategic Thinking)	Item Name(s)	Total # of Items	% Lesson	% Unit
40	20	1	1	K1 – Recognize safety hazards associated with electrical circuits and know the best practices of working safely in an electronics lab environment.						0	#DIV/0!	#DIV/0!
				K2 – Identify the equipment and know how to effectively use the equipment in an electronics lab.						0	#DIV/0!	#DIV/0!
23%	12%			K3 – Know scientific notation, engineering notation, and System International (SI) notation.						0	#DIV/0!	#DIV/0!
				K4 – Know formulas for Ohm's Law, Kirchhoff's Voltage Law, and Kirchhoff's Current Law.						0	#DIV/0!	#DIV/0!
				K5 – Know the characteristics of series and parallel sections of a circuit.						0	#DIV/0!	#DIV/0!
				K6 – Identify digital and analog components and recognize the schematic symbol representation.						0	#DIV/0!	#DIV/0!
				K7 – Know resistor color codes for labeling values.						0	#DIV/0!	#DIV/0!
				K8 – Know capacitor labeling codes.						0	#DIV/0!	#DIV/0!
				K9 – Know the characteristics of LEDs and how to locate LED datasheets.						0	#DIV/0!	#DIV/0!
				K10 – Recognize combinational logic gates.						0	#DIV/0!	#DIV/0!
				K11 – Recognize sequential logic gates.						0	#DIV/0!	#DIV/0!
				K12 – Recognize types of integrated circuits and know where to find manufacturer data sheets.						0	#DIV/0!	#DIV/0!
				K13 – Relate schematic symbols to logic gates and logic gates to schematic symbols.						0	#DIV/0!	#DIV/0!
				K14 – Relate truth tables to logic gates and logic gates to truth tables.						0	#DIV/0!	#DIV/0!
				K15 – Know base 2 and base number systems.						0	#DIV/0!	#DIV/0!
				K16 – Know the best practices of soldering and de–soldering components.						0	#DIV/0!	#DIV/0!
				S1 – Practice proper safety and best practices while working with electronics.						0	#DIV/0!	#DIV/0!
				S2 – Accurately take measurements with a Digital Multimeter (DMM).						0	#DIV/0!	#DIV/0!
				S3 – Express numbers in scientific notation, engineering notation, and System International (SI) notation.						0	#DIV/0!	#DIV/0!
				S4 – Solve for unknown values within circuits (series, parallel, and combination circuits) using Ohm's Law, Kirchhoff's Voltage Law, and Kirchhoff's Current Laws.						0	#DIV/0!	#DIV/0!
				S5 – Utilize Circuit Design Software (CDS) and to validate hand calculations of analog circuit solutions.						0	#DIV/0!	#DIV/0!
				S6 – Identify and describe the function of common components used in electronics.						0	#DIV/0!	#DIV/0!
				S7 – Demonstrate series and parallel circuits on a breadboard.						0	#DIV/0!	#DIV/0!
				S8 – Identify a resistor's nominal value by reading its color code.						0	#DIV/0!	#DIV/0!
				S9 – Measure a resistor's actual value by reading its resistance with a Digital Multimeter (DMM).						0	#DIV/0!	#DIV/0!
				S10 – Identify a capacitor's nominal value by reading its labeled nomenclature.						0	#DIV/0!	#DIV/0!
				S11 – Identify commonly used electronic components given their part number or schematic symbol.						0	#DIV/0!	#DIV/0!
				S12 – Obtain manufacturer datasheets and extract information for components commonly used in digital electronics.						0	#DIV/0!	#DIV/0!
				S13 – Identify various integrated circuit (IC) package styles.						0	#DIV/0!	#DIV/0!
				S14 – Recognize the fundamental differences between combinational and sequential logic.						0	#DIV/0!	#DIV/0!
				S15 – Identify and describe the function of AND, OR, and INVERTER gates.						0	#DIV/0!	#DIV/0!
				S16 – Convert numbers between the binary and decimal number systems.						0	#DIV/0!	#DIV/0!
				S17 – Count from 0–15 in binary.						0	#DIV/0!	#DIV/0!
				S18 – Demonstrate proper soldering/de–soldering techniques to solder and de–solder components on a printed circuit board.						0	#DIV/0!	#DIV/0!
				S19 – Properly tin the tip of a soldering iron and distinguish good solder joints from bad solder joints.						0	#DIV/0!	#DIV/0!
	20	1	1	K1 – Know formulas for Ohm's Law, Kirchhoff's Voltage Law, and Kirchhoff's Current Law.						0	#DIV/0!	#DIV/0!
				K2 – Know the characteristics of series, parallel, and combination circuits.						0	#DIV/0!	#DIV/0!
	12%			K3 – Identify digital and analog components.						0	#DIV/0!	#DIV/0!
				K4 – Know the characteristics and differences between analog and digital signals and circuits.						0	#DIV/0!	#DIV/0!
				K5 – Measure characteristics of a circuit using a DMM.						0	#DIV/0!	#DIV/0!
				K6 – Know the formulas for period, frequency, and duty cycle.						0	#DIV/0!	#DIV/0!
				K7 – Relate schematic symbols to logic gates and logic gates to schematic symbols.						0	#DIV/0!	#DIV/0!
				K8 – Relate truth tables to logic gates and logic gates to truth tables.						0	#DIV/0!	#DIV/0!
				K9 – Relate logic expressions to logic gates and logic gates to logic expressions.						0	#DIV/0!	#DIV/0!
				K10 – There is a formal design process for translating a set of design specifications into a functional circuit.						0	#DIV/0!	#DIV/0!
				S1 – Solve for unknown values within circuits (series, parallel, and combination circuits) using Ohm's Law, Kirchhoff's Voltage Law, and Kirchhoff's Current Laws.						0	#DIV/0!	#DIV/0!
				S2 – Utilize Circuit Design Software (CDS) to validate hand calculations to analog circuit solutions.						0	#DIV/0!	#DIV/0!
				S3 – Demonstrate series and parallel circuits on a breadboard.						0	#DIV/0!	#DIV/0!
				S4 – Analyze simple analog circuits using a digital multimeter.						0	#DIV/0!	#DIV/0!
				S5 – Analyze and interpret the amplitude, period, frequency, and duty cycle of analog and digital signals based on instrumentation and calculations.						0	#DIV/0!	#DIV/0!
				S6 – Interpret the design of a simple 555 Timer oscillator and how the analog components affect the wave generated.						0	#DIV/0!	#DIV/0!

DE Blueprint
The purpose of this assessment is to …

Table of Specifications

Days & % of Coverage Unit	Days & % of Coverage Lesson	Unit	Lesson	Knowledge and Skills	Item Type (ex. multiple choice, performance, true false, essay, etc.)	Complexity Webb's DOK #Level 1 (Recall)	#Level 2 (Skill/ Concept)	#Level 3 (Strategic Thinking)	Item Name(s)	Total # of Items	% Lesson	% Unit
				S7 – Utilize the Circuit Design Software (CDS) to simulate and test a complete analog design.						0	#DIV/0!	#DIV/0!
				S8 – Use Circuit Design Software (CDS) to simulate and test a simple combinational logic circuit designed with AND, OR, and INVERTER gates.						0	#DIV/0!	#DIV/0!
				S9 – Identify and describe the function of a D flip–flop.						0	#DIV/0!	#DIV/0!
				S10 – Use Circuit Design Software (CDS) to simulate and test a simple sequential logic circuit design with D flip–flops.						0	#DIV/0!	#DIV/0!
				S11 – Utilize the Circuit Design Software (CDS) to simulate and test a complete design containing both combinational and sequential logic.						0	#DIV/0!	#DIV/0!
51	20	2	2	K1 – Know the formal design process for designing combinational logic circuits.						0	#DIV/0!	#DIV/0!
				K2 – Know the truth tables and logic expressions associated with AND gates, OR gates, and INVERTER gates.						0	#DIV/0!	#DIV/0!
30%	12%			K3 – Know rules and laws of Boolean Algebra including DeMorgan’s Theorems.						0	#DIV/0!	#DIV/0!
				K4 – Know that a truth table can be interpreted into an algebraic expression representing the output of the circuit.						0	#DIV/0!	#DIV/0!
				K5 – Know that a simplified logic expression can produce the same outputs with fewer gates.						0	#DIV/0!	#DIV/0!
				K6 – Recognize sum–of–product expressions and product–of–sum expressions.						0	#DIV/0!	#DIV/0!
				S1 – Translate design specifications into truth tables.						0	#DIV/0!	#DIV/0!
				S2 – Generate un–simplified logic expressions from truth tables.						0	#DIV/0!	#DIV/0!
				S3 – Construct truth tables from logic expressions.						0	#DIV/0!	#DIV/0!
				S4 – Formulate simplified logic expressions using the rules and laws of Boolean algebra, including DeMorgan’s Theorems.						0	#DIV/0!	#DIV/0!
				S5 – Analyze AOI (AND/OR/INVERTER) combinational logic circuits to compare their equivalent logic expressions and truth tables.						0	#DIV/0!	#DIV/0!
				S6 – Translate a set of design specifications into a functional AOI combinational logic circuit following a formal design process.						0	#DIV/0!	#DIV/0!
				S7 – Simulate and prototype AOI logic circuits using Circuit Design Software (CDS) and a Digital Logic Board (DLB).						0	#DIV/0!	#DIV/0!
				S8 – Identify the IC number and recognize the related wiring diagram for AOI Logic.						0	#DIV/0!	#DIV/0!
	14	2	2	K1 – Identify NAND and NOR gates and recognize them as universal gates.						0	#DIV/0!	#DIV/0!
				K2 – Know that universal gates may provide the opportunity for a more efficient design.						0	#DIV/0!	#DIV/0!
	8%			K3 – Relate AOI logic to NAND only logic.						0	#DIV/0!	#DIV/0!
				K4 – Relate AOI logic to NOR only logic.						0	#DIV/0!	#DIV/0!
				K5 – Know the rules associated with the K–Mapping Technique.						0	#DIV/0!	#DIV/0!
				S1 – Translate a set of design specifications into a functional NAND or NOR combinational logic circuit following a formal design process.						0	#DIV/0!	#DIV/0!
				S2 – Compare and contrast the quality of combinational logic designs implemented with AOI, NAND, and NOR logic gates.						0	#DIV/0!	#DIV/0!
				S3 – Use Circuit Design Software (CDS) to simulate and prototype NAND and NOR logic circuits.						0	#DIV/0!	#DIV/0!
				S4 – Use the K–Mapping technique to simplify combinational logic problems containing two, three, and four variables.						0	#DIV/0!	#DIV/0!
				S5 – Solve K–Maps that contain one or more don’t care conditions.						0	#DIV/0!	#DIV/0!
				S6 – Use current technology to convert AOI designs to universal gate designs.						0	#DIV/0!	#DIV/0!
	10	2	2	K1 – Know the rules governing base 10 number systems.						0	#DIV/0!	#DIV/0!
				K2 – Know the rules governing base 8 number systems.						0	#DIV/0!	#DIV/0!
	6%			K3 – Know the rules governing base 16 number systems.						0	#DIV/0!	#DIV/0!
				K4 – Know the rules governing two’s complement addition.						0	#DIV/0!	#DIV/0!
				K5 – Recognize a half–adder.						0	#DIV/0!	#DIV/0!
				K6 – Recognize a full–adder.						0	#DIV/0!	#DIV/0!
				K7– Label the seven segments of a seven segment display.						0	#DIV/0!	#DIV/0!
				K8 – Identify Common Cathode and Common Anode Seven Segment Displays and know the characteristics of each.						0	#DIV/0!	#DIV/0!
				K9 – Know the formal design process used to translate design specifications to a functional combinational logic circuit.						0	#DIV/0!	#DIV/0!
				K10 – Recognize a multiplexer and de–multiplexer.						0	#DIV/0!	#DIV/0!
				K11 – Describe the benefits of using a multiplexer and de–multiplexer in a circuit design.						0	#DIV/0!	#DIV/0!
				S1 – Convert numbers between the hexadecimal or octal number systems and the decimal number system.						0	#DIV/0!	#DIV/0!
				S2 – Use a seven–segment display in a combinational logic design to display alpha/numeric values.						0	#DIV/0!	#DIV/0!
				S3 – Select the correct current limiting resistor and properly wire both common cathode and common anode seven–segment displays.						0	#DIV/0!	#DIV/0!
				S4 – Design binary half–adders and full–adders using XOR and XNOR gates.						0	#DIV/0!	#DIV/0!
				S5 – Use the two’s complement process to add and subtract binary numbers.						0	#DIV/0!	#DIV/0!
				S6 – Describe how the addition of two binary numbers of any bit length can be accomplished by cascading one half–adder with one or more full adders.						0	#DIV/0!	#DIV/0!
				S7 – Design and implement binary adders using SSI and MSI ICs.						0	#DIV/0!	#DIV/0!
				S8 – Use a formal design process to translate a set of design specifications for a design containing multiple outputs into a functional combinational logic circuit.						0	#DIV/0!	#DIV/0!
				S9 – Design AOI, NAND, & NOR solutions for a logic expression and select the solution that uses the least number of ICs to implement.						0	#DIV/0!	#DIV/0!

DE Blueprint

The purpose of this assessment is to …

Table of Specifications

Days & % of Coverage Unit	Lesson	Unit	Lesson	Knowledge and Skills	Item Type (ex. multiple choice, performance, true false, essay, etc.)	Complexity Webb's DOK #Level 1 (Recall)	#Level 2 (Skill/ Concept)	#Level 3 (Strategic Thinking)	Item Name(s)	Total # of Items	% Lesson	% Unit
				S10 – Design electronics displays using seven–segment displays that utilize de–multiplexers.						0	#DIV/0!	#DIV/0!
				S11 – Use Circuit Design Software (CDS) and a Digital Logic Board (DLB) to simulate and prototype specific combinational logic circuits.						0	#DIV/0!	#DIV/0!
	7	2	2	K1 – Know the role Programmable Logic Devices (PLDs) play in circuit development today.						0	#DIV/0!	#DIV/0!
				K2 – Know the advantages to using PLDs.						0	#DIV/0!	#DIV/0!
				K3 – Know the types of Programmable Logic Devices.						0	#DIV/0!	#DIV/0!
				S1 – Design combinational logic circuits using a programmable logic device.						0	#DIV/0!	#DIV/0!
				S2 – Describe the advantages and disadvantages of programmable logic devices over discrete logic gates.						0	#DIV/0!	#DIV/0!
				S3 – Use Circuit Design Software (CDS) and a Digital Logic Board (DLB) to simulate and prototype combinational logic designs implemented with programmable logic.						0	#DIV/0!	#DIV/0!
56	6	3	3	K1 – Know the schematic symbols and excitation tables for the D and J/K flip–flops.						0	#DIV/0!	#DIV/0!
				K2 – Describe the function of the D and J/K flip–flops.						0	#DIV/0!	#DIV/0!
33%	**3%**			K3 – Describe the function of, and differences between, level sensitive and edge sensitive triggers.						0	#DIV/0!	#DIV/0!
				K4 – Describe the function of, and differences between, active high and active low signals.						0	#DIV/0!	#DIV/0!
				K5 – Describe the function of, and differences between, a flip–flop's synchronous and asynchronous inputs.						0	#DIV/0!	#DIV/0!
				S1 – Draw detailed timing diagrams for the D or J/K flip–flop's Q output in response to a variety of synchronous and asynchronous input conditions.						0	#DIV/0!	#DIV/0!
				S2 – Analyze and design introductory flip–flop applications such as event detection circuits, data synchronizers, shift registers, and frequency dividers.						0	#DIV/0!	#DIV/0!
				S3 – Use Circuit Design Software (CDS) and a Digital Logic Board (DLB) to simulate and prototype introductory flip–flop applications.						0	#DIV/0!	#DIV/0!
	25	3	3	K1 – Recognize asynchronous counters.						0	#DIV/0!	#DIV/0!
				K2 – Recognize that asynchronous counters are commonly referred to as ripple counters.						0	#DIV/0!	#DIV/0!
	15%			K3 – Recognize small scale integration (SSI) logic gates.						0	#DIV/0!	#DIV/0!
				K4 – Recognize medium scale integration (MSI) logic gates.						0	#DIV/0!	#DIV/0!
				K5 – Arrange asynchronous counters to count up or down over a specified range.						0	#DIV/0!	#DIV/0!
				S1 – Describe the advantages and disadvantages of counters designed using the asynchronous counter method.						0	#DIV/0!	#DIV/0!
				S2 – Describe the ripple effect of an asynchronous counter.						0	#DIV/0!	#DIV/0!
				S3 – Analyze and design up, down and modulus asynchronous counters using discrete D and J/K flip–flops.						0	#DIV/0!	#DIV/0!
				S4 – Analyze and design up, down and modulus asynchronous counters using medium scale integrated (MSI) circuit counters.						0	#DIV/0!	#DIV/0!
				S5 – Describe where a count starts and where a count stops/repeats on a modulus asynchronous counter.						0	#DIV/0!	#DIV/0!
				S6 – Use Circuit Design Software (CDS) and Digital Logic Board (DLB) to simulate and prototype SSI and MSI asynchronous counters.						0	#DIV/0!	#DIV/0!
	25	3	3	K1 – Recognize synchronous counters.						0	#DIV/0!	#DIV/0!
				K2 – Recognize small scale integration (SSI) logic gates.						0	#DIV/0!	#DIV/0!
	15%			K3 – Recognize medium scale integration (MSI) logic gates.						0	#DIV/0!	#DIV/0!
				K4 – Arrange synchronous counters to count up or down over specified ranges.						0	#DIV/0!	#DIV/0!
				S1 – Describe the advantages and disadvantage of counters designed using the synchronous counter method.						0	#DIV/0!	#DIV/0!
				S2 – Analyze and design up, down and modulus synchronous counters using discrete D and J/K flip–flops.						0	#DIV/0!	#DIV/0!
				S3 – Analyze and design up, down and modulus synchronous counters using medium scale integrated (MSI) circuit counters.						0	#DIV/0!	#DIV/0!
				S4 – Describe where a count starts and where a count stops/repeats on a modulus synchronous counter.						0	#DIV/0!	#DIV/0!
				S5 – Use Circuit Design Software (CDS) and Digital Logic Board (DLB) to simulate and prototype SSI and MSI synchronous counters.						0	#DIV/0!	#DIV/0!
25	10	4	4	K1 – The basic function of a state machine.						0	#DIV/0!	#DIV/0!
				K2 – Identify the parts of a state graph and a state transition table.						0	#DIV/0!	#DIV/0!
				K3 – Recognize a state machine and identify examples of a state machine.						0	#DIV/0!	#DIV/0!
				K4 – Recognize a wide range of sensor inputs and outputs in real word systems.						0	#DIV/0!	#DIV/0!
15%	**6%**			S1 – Describe the components of a state machine.						0	#DIV/0!	#DIV/0!
				S2 – Draw a state graph and construct a state transition table for a state machine.						0	#DIV/0!	#DIV/0!
				S3 – Derive a state machine's Boolean equations from its state transition table.						0	#DIV/0!	#DIV/0!
				S4 – Implement Boolean equations into a functional state machine.						0	#DIV/0!	#DIV/0!
				S5 –Use Circuit Design Software (CDS) and a Digital Logic Board (DLB) to simulate and prototype state machine designs implemented with discrete and programmable logic.						0	#DIV/0!	#DIV/0!
	15	4	4	K1 – Identify everyday products that contain microcontrollers.						0	#DIV/0!	#DIV/0!
				K2 – Know a programming language for a microcontroller.						0	#DIV/0!	#DIV/0!
	9%			K3 – Identify and describe a servo motor.						0	#DIV/0!	#DIV/0!
				K4 – Know what Pulse Width Modulation (PWM) is and how it is used to control a motor.						0	#DIV/0!	#DIV/0!
				S1 – Program a microcontroller to control a servo.						0	#DIV/0!	#DIV/0!
				S2 – Program and test a microcontroller to control a real system based on inputs.						0	#DIV/0!	#DIV/0!

DE Blueprint
The purpose of this assessment is to …

Table of Specifications

Days & % of Coverage: Unit	Days & % of Coverage: Lesson	Unit	Lesson	Knowledge and Skills	Item Type (ex. multiple choice, performance, true false, essay, etc.)	Complexity Webb's DOK: #Level 1 (Recall)	Complexity Webb's DOK: #Level 2 (Skill/ Concept)	Complexity Webb's DOK: #Level 3 (Strategic Thinking)	Item Name(s)	Total # of Items	% Lesson	% Unit
				S3 – Use mathematics to calculate programming values.						0	#DIV/0!	#DIV/0!
Total Days				**Total**		0	0	0		0		
172				**% of Total**		#####	#####	#DIV/0!				