

The purpose of this guide is to provide a range of knowledge at which students can demonstrate proficiency for each objective. Subsequent college course success depends strongly on courses taught primarily at the “some applied skills present” and “applied skills strongly demonstrated” levels.

Objective	Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
1. Number systems, operations and codes*	<ul style="list-style-type: none"> • Contrast binary versus Gray code potential error generation. • Choose error detection codes for an application. 	<ul style="list-style-type: none"> • Perform addition and subtraction operations in binary and hexadecimal. • Convert fractional binary numbers. 	<ul style="list-style-type: none"> • Convert between binary, decimal and hexadecimal by hand. • Convert between decimal and BCD by hand. 	<ul style="list-style-type: none"> • Define Binary, Decimal and Hexadecimal. • Describe BCD, Gray code, and ASCII.
2. Logic gates*	<ul style="list-style-type: none"> • Construct and analyze logic gates with more than 2 inputs. • Measure voltages and logic levels (high, low, invalid) at inputs and outputs and compare to data sheets. 	<ul style="list-style-type: none"> • Verify the physical functionality of the 7 common logic gates in a laboratory setting. • Contrast ideal electrical behavior versus real world measurements based on data sheets. • Construct and analyze timing diagrams. 	<ul style="list-style-type: none"> • Construct truth tables for the 7 common logic gates. • Identify pin numbers and pinouts of logic gate ICs. • Interpret data sheets. 	<ul style="list-style-type: none"> • Identify truth tables and the operation symbols for the 7 common logic gates.
3. Boolean Algebra *	<ul style="list-style-type: none"> • Prove 12 basic rules of Boolean algebra • Use 12 basic rules of Boolean algebra • Develop Boolean Algebra equations for combinational logic circuits. 	<ul style="list-style-type: none"> • Apply Boolean addition and multiplication • Relate Boolean operations to appropriate logic gates • Construct a Truth 	<ul style="list-style-type: none"> • Evaluate sum and product terms • Describe commutative, associate and distribute laws • Apply & compute Boolean Algebra 	<ul style="list-style-type: none"> • Define variable and literal • Identify Boolean addition and multiplication • Identify & explain Boolean Algebra operators.

The purpose of this guide is to provide a range of knowledge at which students can demonstrate proficiency for each objective. Subsequent college course success depends strongly on courses taught primarily at the “some applied skills present” and “applied skills strongly demonstrated” levels.

Objective	Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
	<ul style="list-style-type: none"> Develop SOP and POS Boolean Algebra equations from Truth Tables. 	Table output for a combinational circuit using Boolean Algebra.	operators to the 7 common logic gates.	
4. DeMorgan’s theorem and logic simplification*	<ul style="list-style-type: none"> Simplify a logic expression by applying Boolean algebra and DeMorgan’s theorem Simplify a logic expression by applying K map. Simplify Boolean Algebra equations using the 12 basic laws of simplification & DeMorgan’s theorem. Predict simplified SOP equations from K-maps. Predict simplified POS equations from K-maps. Prove simplified equations match original equations. 	<ul style="list-style-type: none"> Develop a truth table and K map from a Boolean expression Compare the circuits to match both sides of the 12 basic laws of simplification. Deduce how to group 1’s in a SOP K-maps. Deduce how to group 0’s in a POS K-maps. 	<ul style="list-style-type: none"> Explain the equivalency between NAND and Negative-OR gate and NOR and Negative-AND gate using DeMorgan’s theorem Evaluate a sum-of-products (SOP) expression Apply DeMorgan’s theorem to combinational logic circuits. Show how to place 1’s in a SOP K-map. Show how to place 0’s in a POS K-map. 	<ul style="list-style-type: none"> Describe DeMorgan’s theorem Derive logic expression for a given logic circuit Define DeMorgan’s theorem for NAND and NOR gates. Identify the 12 basic laws of simplification. Draw & label 2,3,4, & 5 variable K-maps.

The purpose of this guide is to provide a range of knowledge at which students can demonstrate proficiency for each objective. Subsequent college course success depends strongly on courses taught primarily at the “some applied skills present” and “applied skills strongly demonstrated” levels.

Objective	Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
5. Combinational logic circuits*	<ul style="list-style-type: none"> • Design and build a simplified combinational circuit from a Boolean output expression • Troubleshoot a combinational circuit with appropriate tools • Construct & evaluate a Combinational circuit from a schematic. • Construct & evaluate a Combinational circuit from a written logic scenario. 	<ul style="list-style-type: none"> • Derive a logic circuit from a given truth table or a K map 	<ul style="list-style-type: none"> • Produce a Truth Table for a Combinational circuit. • Construct a K map from a truth table or logic circuit • Write the Boolean output expression for a combinational circuit 	<ul style="list-style-type: none"> • Identify various logic gates in a combinational circuit • Define combinational logic circuits • List all input combinations for a circuit. • Draw schematics with correct symbols with ECAD.
6. Encoders/decoders*	<ul style="list-style-type: none"> • Build and troubleshoot a 74LS47 7-segment display circuit • Design a logic circuit to decode or encode 	<ul style="list-style-type: none"> • Analyze how to cascade encoders and decoders. • Develop truth table based on the function of decoders and encoders 	<ul style="list-style-type: none"> • Explain the number of input and output bits for a decoder and encoder • Identify the gates needed for a simple encoder and decoder. 	<ul style="list-style-type: none"> • Define encoder and decoder • Describe the function of binary-to-decimal decoder and decimal-to-BCD encoder
7. Multiplexers/demultiplexers*	<ul style="list-style-type: none"> • Wire and troubleshoot multiplexers/demultiplexers circuits • Design a multiplexer to satisfy a logic 	<ul style="list-style-type: none"> • Expand multiplexers/demultiplexers to handle more data lines • Simplify the output 	<ul style="list-style-type: none"> • Develop truth table based on the operation of multiplexers/demultiplexers 	<ul style="list-style-type: none"> • Describe the operation and function of multiplexers/demultiplexers

The purpose of this guide is to provide a range of knowledge at which students can demonstrate proficiency for each objective. Subsequent college course success depends strongly on courses taught primarily at the “some applied skills present” and “applied skills strongly demonstrated” levels.

Objective	Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
	scenario. <ul style="list-style-type: none"> • Design a demultiplexer to satisfy a logic scenario. 	expression of multiplexers/demultiplexers <ul style="list-style-type: none"> • Analyze how to use a multiplexer as a function generator. 	<ul style="list-style-type: none"> • Describe 74HC157 multiplexer and 74HC154 demultiplexer • Draw the logic diagram of multiplexers/demultiplexers • Explain the uses of a multiplexer & a demultiplexer. 	<ul style="list-style-type: none"> • Describe applications of multiplexers/demultiplexers • Define a multiplexer & a demultiplexer.
8. Adders, subtractors, ALUs*	<ul style="list-style-type: none"> • Design and build adder and other ALU circuits with proper logic gates. • Troubleshoot the ALU circuits with proper tools. 	<ul style="list-style-type: none"> • Draw logic diagrams of half-adder, full-adder and other ALUs • Expand adders to multiple bits. • Analyze & apply commercial adders. 	<ul style="list-style-type: none"> • Develop the truth tables of half-adder, full-adder and other ALUs • Simplify the output expression of half-adder, full-adder and other ALUs • Apply adders to solve multi-bit addition. • Explain two's complement use for negative numbers. • Apply two's complement to convert negative binary numbers. 	<ul style="list-style-type: none"> • Describe the function of a half-adder and full-adder • Describe the function of other ALUs • Explain how adders can be used to subtract.

The purpose of this guide is to provide a range of knowledge at which students can demonstrate proficiency for each objective. Subsequent college course success depends strongly on courses taught primarily at the “some applied skills present” and “applied skills strongly demonstrated” levels.

Objective	Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
			<ul style="list-style-type: none"> • Explain carry/borrow inputs and outputs. • 	
9. Flip-flops and related devices*	<ul style="list-style-type: none"> • Wire flip-flops with understanding of “preset”, “clear” and “clock” • Discuss the operating characteristics such as propagation delay, hold time and set-up time. • Interpret the applications such as timers • Construct timing diagrams for latches and F-F’s. • Design & construct latches & F-F’s for various applications. • Design power on reset (POR) circuitry for latches & F-F’s. 	<ul style="list-style-type: none"> • Identify and draw logic diagrams of various flip-flops • Recognize the difference among S-R, D and J-K flip-flops • Explain the difference between combinational and sequential circuits • Analyze & Compare asynchronous preset & clear operations. 	<ul style="list-style-type: none"> • Explain clock pulses and edge-triggered flip-flops • Explain the function of pulse transition detector • Distinguish between a positive and negative edge-triggered flip-flops • Draw Truth tables for latches & F-F’s. • Produce Timing Diagrams for latches & F-F’s. 	<ul style="list-style-type: none"> • Describe structure, operation and application of various types of latches • Distinguish between latches and flip-flops • Identify latches & F-F’s by their schematic symbols. • Explain latch & F-F operations.
10. Counters*	<ul style="list-style-type: none"> • Determine and modify the modulus of a counter • Identify and wire 	<ul style="list-style-type: none"> • Analyze the operation of decade asynchronous counters 	<ul style="list-style-type: none"> • Construct truth table for a sequential logic circuit • Analyze counter 	<ul style="list-style-type: none"> • Define the counters • Describe the operation of a 2 bit asynchronous counter

The purpose of this guide is to provide a range of knowledge at which students can demonstrate proficiency for each objective. Subsequent college course success depends strongly on courses taught primarily at the “some applied skills present” and “applied skills strongly demonstrated” levels.

Objective	Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
	various types of counters such as up/down counters <ul style="list-style-type: none"> • Design a counter with specified sequence states • Construct timing diagrams for std. & truncated counters. 	<ul style="list-style-type: none"> • Analyze the operation of synchronous counters • Analyze the difference between asynchronous and synchronous counters 	timing diagrams <ul style="list-style-type: none"> • Explain & apply synchronous & asynchronous cascading. 	<ul style="list-style-type: none"> • Identify counters by their schematic symbols. • Explain event counting & timing. • Explain synchronous & asynchronous counters. • Define binary & decimal (BCD) counters.
11. Shift registers*	<ul style="list-style-type: none"> • Interpret applications of shift registers such as counters, time delay and data converter • Wire and troubleshoot shift register • Construct timing diagrams for various shift registers. • Design & construct shift registers for various applications. 	<ul style="list-style-type: none"> • Analyze the operation of other shift registers such as bi-directional. • Draw the wave forms of the output of shift registers 	<ul style="list-style-type: none"> • Describe the structure and operation of serial in/serial out, serial in/parallel out, parallel in/serial out and parallel in/parallel out shift register • Identify & explain Johnson & Ring counters and their use. 	<ul style="list-style-type: none"> • Explain how a flip-flop stores and transfer data • Identify logic symbols of various shift registers • Identify basic forms of data movement in shift registers\ • Explain event counting & timing.
12. Memory and storage*	<ul style="list-style-type: none"> • Describe the unique structure and performance of flash memory 	<ul style="list-style-type: none"> • Explain what RAMs are made of and how they work • Explain what ROMs 	<ul style="list-style-type: none"> • Describe the function of three types of buses such as address, data and control 	<ul style="list-style-type: none"> • Describe the basic organization of a memory • Explain the capacity

The purpose of this guide is to provide a range of knowledge at which students can demonstrate proficiency for each objective. Subsequent college course success depends strongly on courses taught primarily at the “some applied skills present” and “applied skills strongly demonstrated” levels.

Objective	Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
	<ul style="list-style-type: none"> • Design & construct circuitry for memory expansion. • Design & construct memory circuits for various applications. 	<p>are made of and how they work</p> <ul style="list-style-type: none"> • Compare the RAM and ROM for their advantages and disadvantages. • Analyze techniques for memory expansion. 	<ul style="list-style-type: none"> • Describe the basic read and write operation • Identify & explain RAM & ROM inputs and outputs. • Compute address size and organization of memory from inputs & outputs. 	<p>and address of a memory</p> <ul style="list-style-type: none"> • Identify & explain RAM & ROM inputs and outputs. • Compute address size and organization of memory from inputs & outputs.
13. Integrated circuit technologies*	<ul style="list-style-type: none"> • Compare CMOS and TTL in term of their performance • Build and measure a few logic gates with transistors 	<ul style="list-style-type: none"> • Interpret the operation of various logic gates such as inverters, NAND and NOR gates implemented by MOSFETs • Interpret the operation of various logic gates such as inverters, NAND and NOR gates implemented by BJTs • Compare performance parameters of logic families. 	<ul style="list-style-type: none"> • Read and obtain information from the data sheet of IC devices • Explain the basic operation of MOSFETs and BJTs • Identify MOSFETs and BJTs by their symbols • 	<ul style="list-style-type: none"> • Discuss basic IC characteristics such as logic levels, noise margin and fan-out • Explain how propagation delay affects the circuit speed • List various logic families. • Identify various packaging styles. • Define complexity SSI through ULSI (Gates through microprocessors).

The purpose of this guide is to provide a range of knowledge at which students can demonstrate proficiency for each objective. Subsequent college course success depends strongly on courses taught primarily at the “some applied skills present” and “applied skills strongly demonstrated” levels.

Objective	Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
14. VHDL topics	<ul style="list-style-type: none"> • Program a logic circuit using both text and graph • Program, compile, simulate, download and run a logic circuit 	<ul style="list-style-type: none"> • Explain the entire design flow • Explain each step in the design such as simulation, synthesis and download • Compare GAL's and PLA's. 	<ul style="list-style-type: none"> • Distinguish between schematic and text entry • Discuss the function of a compiler • Produce HDL descriptions 	<ul style="list-style-type: none"> • Define programmable logic device • Define hardware description language • Define HDL. • Identify HDL programmers.
15. Introduction to microprocessors, computers and buses	<ul style="list-style-type: none"> • Identify different levels of programming languages • Discuss how the interrupts work in a computer • Write simple machine language programs. 	<ul style="list-style-type: none"> • Describe the sequence of a standard process in a microprocessor • Identify three types of buses • Compare high level language and machine language. • Compare polling and interrupt operation. 	<ul style="list-style-type: none"> • Describe what each component of a computer does • List the basic components of a microprocessor • Define and compare an OS and a user program. • Explain the Fetch/execute system. 	<ul style="list-style-type: none"> • List the basic components of a computer • Discuss the function of software and hardware • List & explain std. computer buses. • List & explain a std. computer's CPU, Memory, and I/O. • Define DMA, co-processors, & multitasking.
16. Introduction to digital signal processing (DSP)	<ul style="list-style-type: none"> • Name the basic elements in a DSP • Explain how a DSP works • Wire and troubleshoot ADC and DAC 	<ul style="list-style-type: none"> • Explain the operation of various types of ADC and DAC • Analyze errors caused by conversions. 	<ul style="list-style-type: none"> • Discuss the sampling theorem • Discuss the purpose of filtering and sample-and-hold function 	<ul style="list-style-type: none"> • State the purpose of ADC and DAC • Define DSP.

The purpose of this guide is to provide a range of knowledge at which students can demonstrate proficiency for each objective. Subsequent college course success depends strongly on courses taught primarily at the “some applied skills present” and “applied skills strongly demonstrated” levels.

Objective	Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
	circuits 		<ul style="list-style-type: none"> • Compute outputs for AtoD & DtoA converters. • Compare polling and interrupt operation. 	
17. Digital communications and transmission standards	<ul style="list-style-type: none"> • Interpret the operation of typical error detection technique in a digital transmission system • Distinguish between odd and even parity technique • Interpret a complete digital communication and transmission system and identify the basic components in it • Construct simple circuits to do data transmission to computers. 	<ul style="list-style-type: none"> • Interpret the function of multiplexer/demultiplexer in a digital transmission system • Compare Bit rate and Baud rate. • 	<ul style="list-style-type: none"> • Discuss the operation of serial and parallel data transfer • Describe the sampling theorem • Compute throughput and efficiency of transmission. 	<ul style="list-style-type: none"> • State the advantages of digital communication over analog one • Explain how the information is transferred by digital signals • List std. serial and parallel data transmission options. • List std. serial and parallel data transmission media. • Describe modulation techniques.