Using the CTAG Rubrics for Digital Electronics

Presented by
Randy Storms
Assistant Professor
North Central State College

CTEET002 – Digital Electronics
4 Semester

- No Prerequisites
- 4 semester hours
    - 5 contact hours per week for 15 weeks – 3 Lecture/2 lab
    - 75 hours “in seat” time per semester
    - OBR requires 2 hours work outside of class per credit hour minimum.
    - Minimum of 8 hours homework.
    - Weekly time required of students is a minimum of 13 hours.

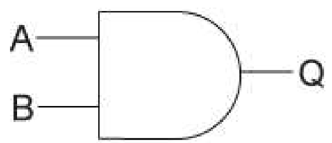
Digital "Outcomes" - *Required – Must Be Taught

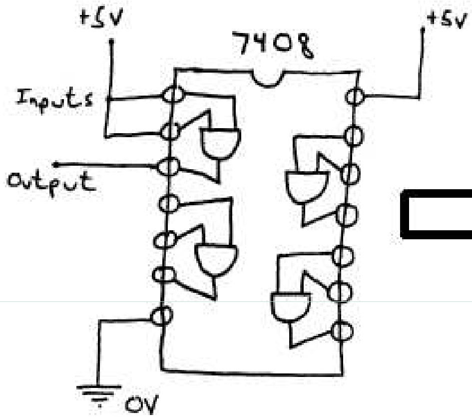
- Number systems, operations and codes*
- Logic gates*
- Boolean Algebra *
- DeMorgan's theorem and logic simplification*
- Combinational logic circuits*
- Encoders/decoders*
- Multiplexers/demultiplexers*
- Adders, subtractors, ALUs*
- Flip-flops and related devices*
- Counters*
- Shift registers*
- Memory and storage*
- Integrated circuit technologies*

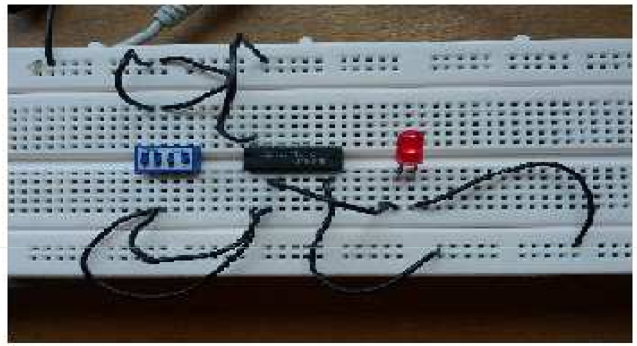
Submitted course work must include proof of laboratory component

- Students will not fully grasp the subject matter until they must physically transfer the schematic to the circuit board.
- Multi-SIM and other simulators are great additional learning tools, but do not replaces “hands on” circuit wiring.
- It takes time to develop the skills of successfully transferring the abstract “symbology” to a protoboard or circuit board.
- Besides, it is a required CTAG component.

From this to this to this







What I have learned of the TAG/CTAG process

- Make sure your syllabus outcomes and material line up with the outcomes of the listed CTAG.
- Borrow from the rubrics on the CEMS descriptions.
- Be clear on the number of hours spent on each outcome.
- Be clear on the homework assignments, i.e., what problems from what chapter.
- Be clear on the labs.
- You WANT approval on the first submission.

Outcome 1
Number systems, operations, and codes.*

Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
• Contrast binary versus Gray code potential error generation. • Choose error detection codes for an application.	• Perform addition and subtraction operations in binary and hexadecimal. • Convert fractional binary numbers.	• Convert between binary, decimal and hexadecimal by hand. • Convert between decimal and BCD by hand.	• Define Binary, Decimal and Hexadecimal. • Describe BCD, Gray code, and ASCII.

- Your students must be able to:
  - count to 16_{10} for purposes of building truth tables.
  - convert 4-bit binary numbers to decimal, Hexadecimal and BCD by memory.
  - convert single digit Decimal, Hexadecimal and BCD to a 4-bit binary value by memory.
  - an 8-bit binary number to decimal, and Hexadecimal by hand and visa versa.
- Students MUST understand how to convert using successive division.
- Encourage the use of Engineering calculator apps for a smartphone and using the calculator provided with Windows.

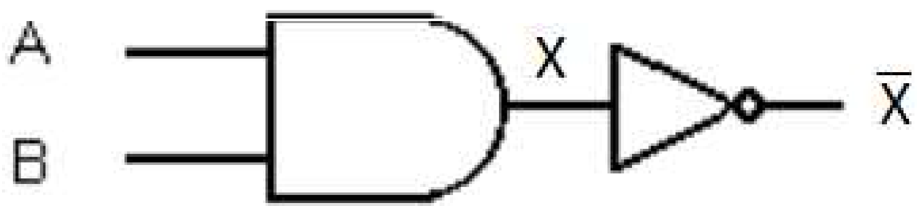
Outcome 2
Logic Gates

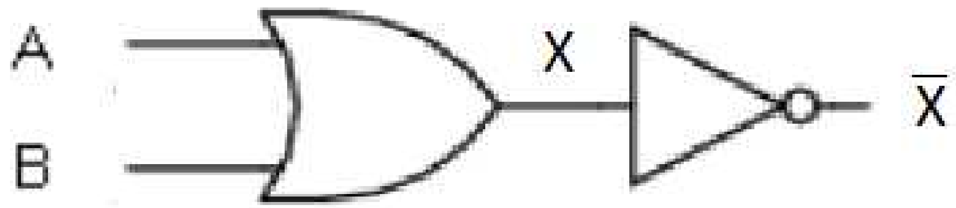
Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
• Construct and analyze logic gates with more than 2 inputs. • Measure voltages and logic levels (high, low, invalid) at inputs and outputs and compare to data sheets.	• Verify the 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.	• Construct truth tables for the 7 common logic gates. • Identify pin numbers and pinouts of logic gate ICs. • Interpret data sheets.	• Identify truth tables and the operation symbols for the 7 common logic gates.

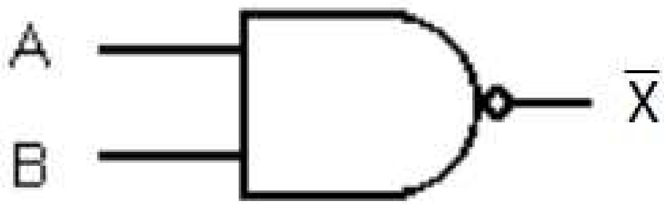
- It is important that students learn how to properly use a digital logic probe.
  - Students will typically assume that 0 volts, ground, and no connection amount to the same thing.
  - A voltmeter cannot distinguish between a ground and no connection.
  - A voltmeter does no easily show you if there is a signal pulse on an I/O.
- It is important that students trace problems with test instruments and not just recheck/redo wiring.

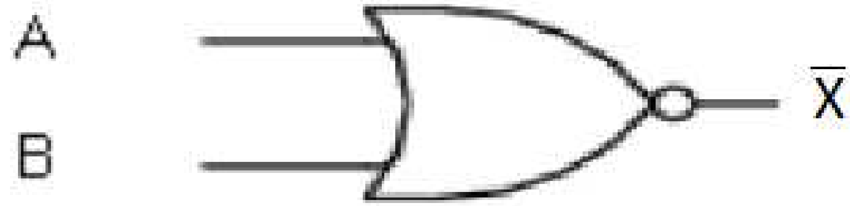
A	B	X		$\overline{X}$
0	0	0		1
0	1	0		1
1	0	0		1
1	1	1		0

A	B	X		$\overline{X}$
0	0	0		1
0	1	1		0
1	0	1		0
1	1	1		0









Outcome 3
Boolean Algebra

Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
• Prove 12 basic rules of Boolean algebra • Use 12 basic rules of Boolean algebra • Develop Boolean Algebra equations for combinational logic circuits. • Develop SOP and POS Boolean Algebra equations from Truth Tables.	• Apply Boolean addition and multiplication • Relate Boolean operations to appropriate logic gates • Construct a Truth Table output for a combinational circuit using Boolean Algebra.	• Evaluate sum and product terms • Describe commutative, associate and distribute laws • Apply & compute Boolean Algebra operators to the 7 common logic gates.	• Define variable and literal • Identify Boolean addition and multiplication • Identify & explain Boolean Algebra operators.

- I take several weeks mix with other objectives to teach Boolean, DeMorgan, SOP, and POS.
- Homework problems, homework problems, homework problems.
- Verify the Boolean Algebra laws in MultiSim.
- How many gates required to implement $X=(A+B)C$
- What Law? $B + (F + G) = F + (B + G)$
- Develop equations

Outcome #4
DeMorgan’s Theorem & Logic Simplification

- Outcomes 3 and 4 require the Memorization and Application of:
  - 3 Laws
  - 12 Rules
  - DeMorgan’s Theorem
  - Maurice Karnaugh’s map
- I spread all this out over the first 7-8 weeks of the semester.
- Students not only need time for homework exercises but for the memory “writes” to take hold.
- Considering that most are either taking remedial math or college level algebra…

Applied skills strongly demonstrated	**Some applied skills present**	**Little applied skills present**	**No applied skills Present**
• 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.	• 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.	• 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.	• 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. •

1) Students have a very difficult time with understanding the DeMorgan's Law. It is a 3 step process:

- **(a) Invert the literals**
- **(b) Change the operator (AND - OR)**
- **(c) Invert the function Thus, $X'Y \rightarrow (XY') \rightarrow (X+Y') \rightarrow (X+Y')'$**

2) Also, no textbook presents a very good stepwise approach to Boolean reduction. They just present the laws and let the students loose. But, here is a better way!!!

Excerpt from Ralph D. Whaley, Jr., Ph.D.'s presentation in 2014 Cincinnati State Technical and Community College

(1) Convert with DeMorgan's if needed

$(A + B)' = A'B'$; $(AB)' = A' + B'$

(2) Expand terms

$C(A+B) = CA + CB$

(3) Simplify "simple" terms

$AA'BC = 0$; $ABC'C' = ABC'$

$A + AB = A$; $1 + X = 1$

(4) Group terms and reduce

$AB'C + AB' = AB'(C+1) = AB'$

(5) Look for any terms with A + A'B "2nd law"

$ABC + B'C = C(B' + BA) = C(B'+B)(B'+A)=C(B'+A)$

(6) Possibly consider adding "1"

$AB = AB(C+C') = ABC + ABC'$

Excerpt from Ralph D. Whaley, Jr., Ph.D.'s presentation in 2014
Cincinnati State Technical and Community College

Outcome 5
Combinational Logic Circuits

Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
• 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.	• Derive a logic circuit from a given truth table or a K map	• 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	• 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.

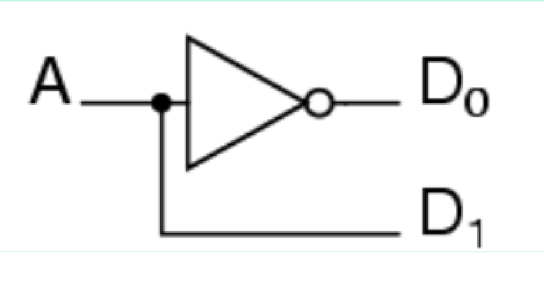
Design a circuit such that a hall light may be controlled by 2 different switches.

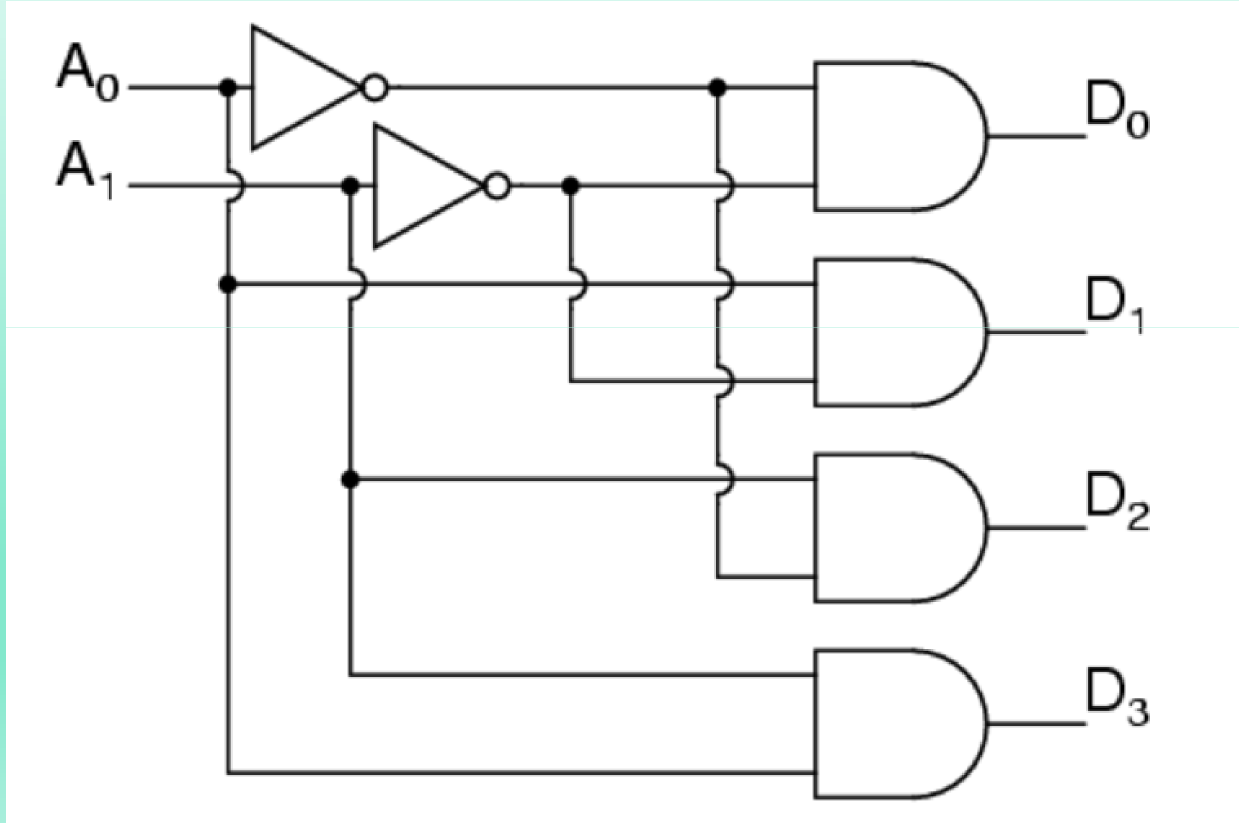
- Develop the equation.
- Develop the truth table
- Design the circuit.
- Build the circuit.
- Test against the truth table.

Outcome 6
Encoders and Decoders

Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
• Build and troubleshoot a 74LS47 7-segment display circuit • Design a logic circuit to decode or encode	• Analyze how to cascade encoders and decoders. • Develop truth table based on the function of decoders and encoders	• Explain the number of input and output bits for a decoder and encoder • Identify the gates needed for a simple encoder and decoder.	• Define encoder and decoder • Describe the function of binary-to-decimal decoder and decimal-to-BCD encoder

- Demonstrate how a single line in can select between 2 lines out.
- Decoders take an n-digit binary number and decodes it into 2^n data lines.

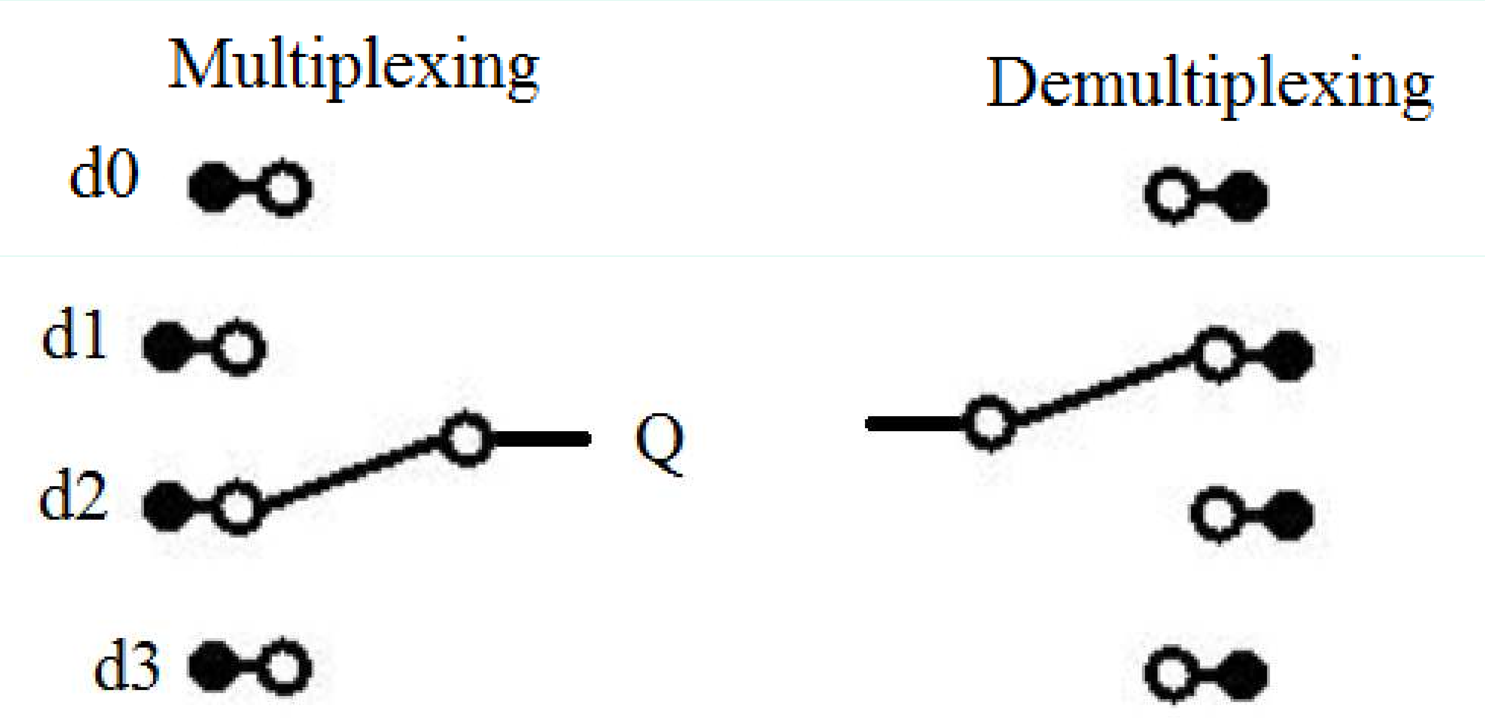




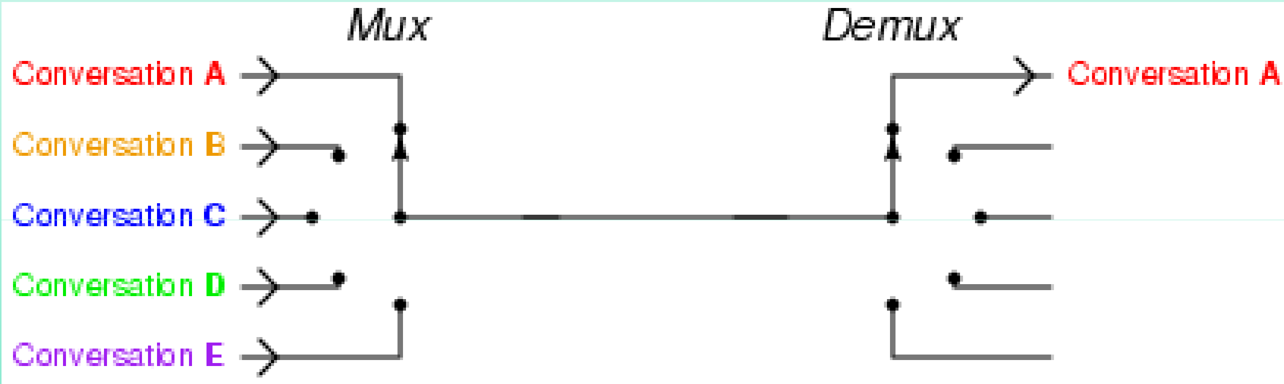
Outcome 7
Multiplexers and Demultiplexers

Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
• Wire and troubleshoot multiplexers/demultiplexers circuits • Design a multiplexer to satisfy a logic scenario. • Design a demultiplexer to satisfy a logic scenario.	• Expand multiplexers/demultiplexers to handle more data lines • Simplify the output expression of multiplexers/demultiplexers • Analyze how to use a multiplexer as a function generator.	• Develop truth table based on the operation of multiplexers/demultiplexers • Describe 74HC157 multiplexer and 74HC154 demultiplexer • Draw the logic diagram of multiplexers/demultiplexers • Explain the uses of a multiplexer & a demultiplexer.	• Describe the operation and function of multiplexers/demultiplexers • Describe applications of multiplexers/demultiplexers • Define a multiplexer & a demultiplexer.

- Although it used to make more of a connection with students when we still had analog tv's with a channel selector I still use rotary switches to demonstrate multiplexing and demultiplexing.



- Stress the use of the n line to 2^n line decoder used in the demux.
- And the use of 2^n to n encoder used in the mux.



http://en.wikipedia.org/wiki/Multiplexer#/media/File:Telephony_multiplexer_system.gif

Outcome 8 - Adders, Subtractors, ALUs

Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
• Design and build adder and other ALU circuits with proper logic gates. • Troubleshoot the ALU circuits with proper tools.	• Draw logic diagrams of half-adder, full-adder and other ALUs • Expand adders to multiple bits. • Analyze & apply commercial adders.	• 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 compliment use for negative numbers. • Apply two's compliment to convert negative binary numbers. • Explain carry/borrow inputs and outputs.	• Describe the function of a half-adder and full-adder • Describe the function of other ALUs • Explain how adders can be used to subtract.

- Analyze the simple gate logic of 2 bit half and full adders.
- Beyond that we use the functional block diagrams.
- It is important that students understand how negative binary numbers are treated.
- Students should be able to cascade half-bit and full-bit adders

Outcome 9
Flip-Flops and Related Devices

- Done with Combinational logic
- Start with Sequential logic
- Outputs only change state with a clock or other enable/disable signals
- Timing diagrams are a MUST!
- Flip-flops are latches are memory.
- Registers, groups of FFs are the basis of computer memory.

Applied skills strongly demonstrated	**Some applied skills present**	**Little applied skills present**	**No applied skills Present**
• 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.	• 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.	• 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.	• 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.

Outcome 10 - Counters

Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
• Determine and modify the modulus of a counter • Identify and wire various types of counters such as up/down counters • Design a counter with specified sequence states • Construct timing diagrams for std. & truncated counters.	• Analyze the operation of decade asynchronous counters • Analyze the operation of synchronous counters • Analyze the difference between asynchronous and synchronous counters	• Construct truth table for a sequential logic circuit • Analyze counter timing diagrams • Explain & apply synchronous & asynchronous cascading.	• Define the counters • Describe the operation of a 2 bit asynchronous counter • Identify counters by their schematic symbols. • Explain event counting & timing. • Explain synchronous & asynchronous counters. • Define binary & decimal (BCD) counters.

- Waveform analysis of counters are absolutely necessary.
- Spend time on the data sheets!
- Understand the Clock inputs and which edge triggers the count.
- A great time to discuss and observe propagation delay.

Outcome 11
Shift Registers

Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
• 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.	• Analyze the operation of other shift registers such as bi-directional. • Draw the wave forms of the output of shift registers	• 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.	• 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.

Outcome 12
Memory and Storage

Applied skills strongly demonstrated	**Some applied skills present**	**Little applied skills present**	**No applied skills Present**
• Describe the unique structure and performance of flash memory • Design & construct circuitry for memory expansion. • Design & construct memory circuits for various applications.	• Explain what RAMs are made of and how they work • Explain what ROMs are made of and how they work • Compare the RAM and ROM for their advantages and disadvantages. • Analyze techniques for memory expansion.	• Describe the function of three types of buses such as address, data and control • Describe the basic read and write operation • Identify & explain RAM & ROM inputs and outputs. • Compute address size and organization of memory from inputs & outputs.	• Describe the basic organization of a memory • Explain the capacity and address of a memory • Identify & explain RAM & ROM inputs and outputs. • Compute address size and organization of memory from inputs & outputs.

- If there is time.
- This outcome belongs in the first microprocessor class, not digital.
- The hierarchy of gates to latches to FFs to registers should be emphasized.

Outcome 13
Integrated Circuit Technologies

Applied skills strongly demonstrated	Some applied skills present	Little applied skills present	No applied skills Present
• Compare CMOS and TTL in term of their performance • Build and measure a few logic gates with transistors	• 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.	• 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 •	• 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).

- Considering that digital is a no-prereq course, most students will have no semiconductor background knowledge.
- TTL voltage levels, fan-outs, package types, CMOS handling, reading data sheets, etc, should be covered.

The Wrap-up

- Digital is hard for many students
  - The student needs to spend a minimum of 8 hours, per week on reading assignments, solving problems and review.
  - 13 outcomes are to many for mastery in one 4 credit hour course.
  - The difficulty is compounded by a lack of prerequisites.
- The purpose of TAGs and CTAGS are to help insure student success in later courses.
- The purpose of the rubrics are to help guide you, the teacher, with *measurable objectives.*