Review Guide
FINAL EXAM

Topics and Questions

CSE 2301 – Spring 2022

INTRODUCTION

The final exam for this course is cumulative. You'll be responsible for all content over the course of the semester, which means that most topics from module 1 through module 18 are fair game. This guide is intended to sharpen your focus with respect to what you really need to emphasize while studying. If it's not here, you're not going to need it for the exam.

That being said, this is a guide to ***potential*** questions. Attempt to at least skim each problem, even if you don't have the time to work it out completely. There are 40 questions. Completing this entire packet would take a considerable amount of time.

Practice question difficulty, assuming content familiarity:

★☆☆☆☆ Trivial	Very easy. These should be "free" points on your exam.
★★☆☆☆ Average	Most questions will fall into this category. You must be capable of completing these independently and efficiently.
★★★☆☆ Moderate	This is what you should expect from higher value problems.
★★★★☆ Hard	At least one problem of this level will appear on the exam. These are meant to test the knowledge of good students.
★★★★★ Very hard	Harder than any exam questions you will face. Correctly answering these is a testament to your preparedness.

TOPICS

This list is all encompassing. Some of the topics mentioned on this list are described in detail in the vocabulary section. I have also attempted to have at least one question for most of the problems discussed.

Exam 1 Content

- **Module 1**: Refamiliarize yourself with Ohm's law and basic circuit analysis tools like summing resistance over series and parallel circuits. You do not need to know how resistors are labelled physically.
- **Module 2:** Truth tables for AND, OR, NOT, NAND, and XOR gates. Recall the terminology for basic circuits. Memorize the TTL voltage levels (0v to 0.8v low, 2v to Vcc high).
- **Module 3:** Definitions of radix and other number systems. Know how to convert freely between any combination of unsigned binary, ternary, signed magnitude, 2's complement, hexadecimal, and octal. You should also be able to do IEEE 754 floating point conversions.

- **Module 4**: Review the fundamentals of reducing equations. That means identity functions, DeMorgan's Theorem, reduction with XOR, and switching algebra. Terminology for sums and products. You do not need to memorize Gray codes or ASCII tables, but you should be familiar with what BCD and XS3 mean.
- **Module 5:** You should know Karnaugh maps like the back of your hand. Expect a 5-variable problem on the exam. Quine-McCluskey is not on the exam.
- **Module 5.5:** Troubleshooting terminology.

Exam 2 Content

- **Module 6:** Understand timing diagrams and static hazards, including how to add logic to prevent a static hazard by using a Karnaugh map. You should also be able to interpret the change in a node (on a timing diagram) based on VHDL code or a schematic. Additionally: line codes. You may be expected to transpose signals with unipolar NRZ/RTZ or Manchester codes.
- **Module 7:** Recall POSTNET. You should also be extremely familiar with the parity checking/correcting codes for a 4-data-bit number. Review hamming and RAID terminology. Triangle parity may also appear on the exam, but you are not responsible for knowing intelligent bar codes or similar non-POSTNET systems.
- **Module 8:** Multiplexors and multiplexing tables. You do not need to be familiar with buffers and 3-state gates. You should be able to create a PLA diagram from a series of equations, or vice versa. Basic terminology.
- **Module 9:** Develop a theoretical understanding of adders and ALUs. Fully comprehend multiplication by partial products. You do not need to know demultiplexors/decoders or carry-lookahead.
- **Module 9.5:** Understand the basics of writing and deciphering code in VHDL. You don't need to be able to build entity/architecture, but writing logical statements with delays should be familiar. Recognize the difference between inertial and transport delay.

Exam 3 Content

- **Module 10:** Sequential logic, including the core definitions of serial and parallel operation, D flip-flops, JK flip-flops, SR latches. PRE/SET and CLR operations, shift registers, and counters. We will not ask about T flip-flops. Understand the principles of PRBS generators.
- **Module 11:** Advanced VHDL. You should be able to comprehend branching and statements conditioned on events, as well as processes, but it is unlikely that we will ask you *write* code of this complexity.
- **Module 12:** Everything related to state diagrams, from drawing nodes to notation to transition tables. We will not ask you about Mealy and Moore state machines.
- **Module 13:** You should be able to identify gates built from CMOS. Recall also the voltage ranges of CMOS. Unit loads and fan-in/fan-out will not be included in exam content, but they are good to know.
- **Module 14:** You should be able to identify gates built from diodes, understand debouncers, and potentially build other gates from NANDs only. Recount the differences between 54 and 74 series chips. The mathematical specifics of Zener diodes will not be mentioned.

- **Module 15:** A useful module that gives a great overview of practical circuit building and reviews how many systems interact, but there is nothing new here.

New Content

- **Module 16:** Mostly definitions, which are enumerated in the vocabulary section. You should absolutely review Nyquist's work and conversion between digital and analog. We may have some general questions on the specifics of compact disk reading and hazard avoidance. For example, you should know the difference between DRAM and SRAM.
- **Module 17:** Be capable of working out a sequential multiplication example step-by-step.
- **Module 18:** There is a lot of content in this module. You should be able to identify the *basic* properties of memory, especially with respect to the volatility of certain systems. Vocabulary is given below.
- Modules 19 and 20 have been cut from the final.

VOCABULARY

This list is exhaustive. Though it does not include every term covered in the modules, it includes every *definition* that may be relevant to your final exam.

Definitions

- Analog vs. digital: Waveform vs. pits and lands.
- Combinational Logic: Digital logic implemented in the form of Boolean circuits, where the output is purely a function of the input.
- Radix: Base or number of values per digit (10 in decimal, 2 in binary)
- Radix economy: How efficiently a value can be represented in a certain radix. The most efficient base is e.
- Literal: A variable or its complement (A or A', etc.)
- Minterm: Synonymous with "product term". The product of multiple literals via AND operations (ABC).
- Maxterm: Synonymous with "sum term". The sum of multiple literals via OR operations (A+B+C).
- Canonical sum: Also known as a "sum of products" (AB + CDE + B'D').
- "Don't Care": A state represented by X on a Karnaugh map or transition table which represents some state or product term that isn't being used and/or has undefined behavior. Can be treated as a zero or one on a K-map.
- Fault Stimulation: Troubleshooting technique. Set all input conditions such that toggling one switch triggers a change in some observable output. If it does not change, an error is present.
- Path sensitization: Observation of node values.
- Hamming Distance: The number of bits in error. As hamming distance increases, you need more parity bits to both detect and correct those errors.
- RAID-0: Data is "striped" or distributed across disks.
- RAID-1: Data is mirrored between two independent disks.

- RAID-5: Four disk with distributed parity bits. Can restore a downed disk.
- Sequential logic: Outputs are based on memory from previous inputs as well as existing values.
- Serial I/O: Also known as synchronous. Concerns loading bits to registers via the D input and the operation of the clock.
- Parallel I/O: Also known as asynchronous. Changes of state via PRE/SET and CLR inputs, or simultaneously reading all flip-flop outputs, independently of the clock.
- Sampling rate: The frequency at which measurements or comparisons are made for an ADC. Higher frequencies result in greater accuracy.
- Resolution: The number of bits used to represent a waveform.
- Nyquist Criterion: In order to accurately sample a waveform, you need to sample at twice the rate of the highest recorded analog frequency.
- Aliasing: Jagged, irregular sampling that occurs when the sampling rate is less than the frequency suggested by the Nyquist criterion.
- Memory Volatility: Magnetic systems maintain their states without power. Delay tubes and vacuum tubes are volatile. Semiconductors can be volatile or non-volatile. Most need power to maintain their states.

Acronyms

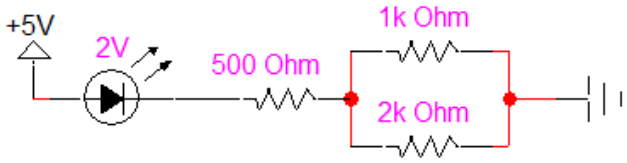
- TTL: Transistor-Transistor Logic
- SIL: Single in-line (rows of resistors)
- DIL: Dual in-line (switches)
- SSI: Small scale integration (<10 gates)
- MSI: Medium scale integration (10-100 gates)
- LSI: Large scale integration (thousands of gates)
- VLSI: Very large-scale integration (millions of gates)
- BCD: Binary Coded Decimal
- NRZ: Non-return-to-zero
- RTZ: Return-to-zero (high signal returns to low halfway through clock)
- PLA: Programmable Logic Array
- ASIC: Application Specific ICs (pre-programmed)
- FPGA: Field Programmable Gate Array
- RAID: Redundant Array of Independent/Inexpensive Disks
- ALU: Arithmetic Logic Unit
- VHDL: [Very high speed integrated circuit] Hardware Description Language
- PRBS: Pseudo-Random Binary Sequence
- ADC: Analog to Digital Converter
- DAC: Digital to Analog Converter
- ROM: Read-only memory
- PROM: Programmed ROM (non-volatile)
- RAM: Random Access Memory
- DRAM: Dynamic RAM (needs to be actively refreshed)
- SRAM: Solid RAM (stored as long a power is applied)

SECTION 1:

Modules 1-5.5 Practice

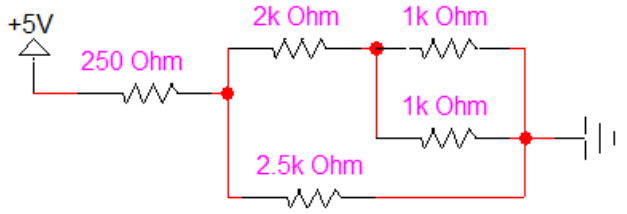
Module 1: Circuits *Difficulty ★☆☆☆☆*

1. Find the current through the LED for the circuit below. Assume that the LED is on since the power is running from anode to cathode.



Module 1: Circuits *Difficulty ★★☆☆☆*

2. Find the equivalent resistance through the entire circuit given below, and the power through the 2.5kΩ resistor.



Module 4: Algebra
Difficulty ★☆☆☆☆

3. Determine the results of the following Boolean algebra statements.

- $A + 0$
- $A + A'$
- $0 * 1$
- $A * A'$
- $A \oplus A'$
- $1 \oplus A$

Module 4: Algebra
Difficulty ★★☆☆☆

4. Reduce the following equations to their simplest form.

- $AB + BC$
- $AB'C + ABC'$
- $A + AB + ABC$
- $AB'(C' + D) + A'B(C' + D)$
- $AB + A'B'$
- $(A + C)' + A'B$
- $A(B'(CD))'$

Module 2/4: NAND *Difficulty ★★★☆☆*

5. Construct AND, OR, NOT, and XOR gates using only 2-input NANDs. Then build a circuit of the form $F = ABC' + B'(A \oplus C)$.

Module 3: IEEE 754 *Difficulty ★★★★☆*

6. Convert the decimal number -31.415 to IEEE 754 format with 8 bits of precision. Include as many bit in the mantissa as are necessary.

Module 3: Twos Comp *Difficulty ★★☆☆☆*

7. Add $10101110 + 01111000$ in unsigned binary and two's complement. Does overflow occur? Define what it means to have overflow in each case.

Module 3: Radix *Difficulty ★★☆☆☆*

8. Determine the radix economy of 111_{10} in binary, octal, and decimal. Then do the same for 73_{10} in ternary and hexadecimal.

Module 3: Arithmetic

Difficulty ★☆☆☆☆

9. Perform the following operations in unsigned binary. Specifically, your work should be done in binary.

- $8 - 3 =$
- $11 + 13 =$
- $5 * 4 =$
- $15 / 4 =$

Module 3: Conversion

Difficulty ★★☆☆☆

10. For each value, convert to decimal, hexadecimal, binary, and octal.

- 150_8
- CF_{16}
- $0110\ 1100_2$
- 175_{10}

Module 5: K-maps *Difficulty ★★★☆☆*

11. Reduce a 5-variable K-map where $\sum m(2, 4, 5, 11, 14, 16, 18, 28, 30)$ are active high, $\sum M(0, 1, 6, 9, 13, 15, 17, 20, 22, 23, 25, 29, 31)$ are active low. Remaining states are "don't care". Your final result should have 4 minterms.

Module 5: K-maps ***Difficulty ★★★★★***

12. **The K-Map of Doom**: Reduce a 6-variable Karnaugh map to its absolute minimum. An optimal solution also uses XORs. F is generated by the following function:

F is equal to 1 if when zero or five variables are high. When one or four variables are high, we don't care about F. F is zero when two or six variables are high. Some elements shown.

U	V	W	X	Y	Z	F
0	0	0	0	0	0	1
					1	1
				1	1	0

U	V	W	X	Y	Z	F
		1	1	1	1	X
	1	1	1	1	1	1
1	1	1	1	1	1	0

The outputs when three variables go high are special, as defined below.

U	V	W	X	Y	Z	F
			1	1	1	X
		1		1	1	0
		1	1		1	1
		1	1	1		0
	1			1	1	1
	1		1		1	1
	1		1	1		0
	1	1			1	0
	1	1		1		X
	1	1	1			0

U	V	W	X	Y	Z	F
1				1	1	X
1			1		1	0
1			1	1		0
1		1			1	0
1		1		1		X
1		1	1			1
1	1				1	0
1	1			1		0
1	1		1			X
1	1	1				0

SECTION 2:

Modules 6-9.5 Practice

Module 7: POSTNET *Difficulty* ★★☆☆☆

13. Is the given POSTNET code correct? If not, identify the location of the error and the correct value. Then draw that value in POSTNET.

Module 6: Parity *Difficulty* ★☆☆☆☆

14. Which of the following logic expressions checks for even parity among A, B, C, and D?

a) *(D xor C) or (D xor A)*
b) *D xor C xor B xor A*
c) *(D xor C)' xor (B xor A)'*
d) *(D or C) and (B or A)*

Module 6: Hazards *Difficulty* ★★☆☆☆

15. For a 4-variable K-map, we currently have the equation F = D'B' + C'BA. Add another minterm to prevent the possibility of a static hazard.

Module 6: Signals *Difficulty* ★★☆☆☆

16. Draw the binary string 00101110 in Manchester code and Unipolar RTZ.

Module 6: Timing *Difficulty* ★★★☆☆

17. Given the following VHDL code, complete a timing diagram where A is changed to '1' at t=30, then back to '0' at t=40. B changes to '1' at t=50. Everything is initialized to zero. Delay is inertial. Complete table up to 100 ns.

```
C <= not C after 20 ns;
D <= A or B after 20 ns;
E <= C and D after 10 ns;
```

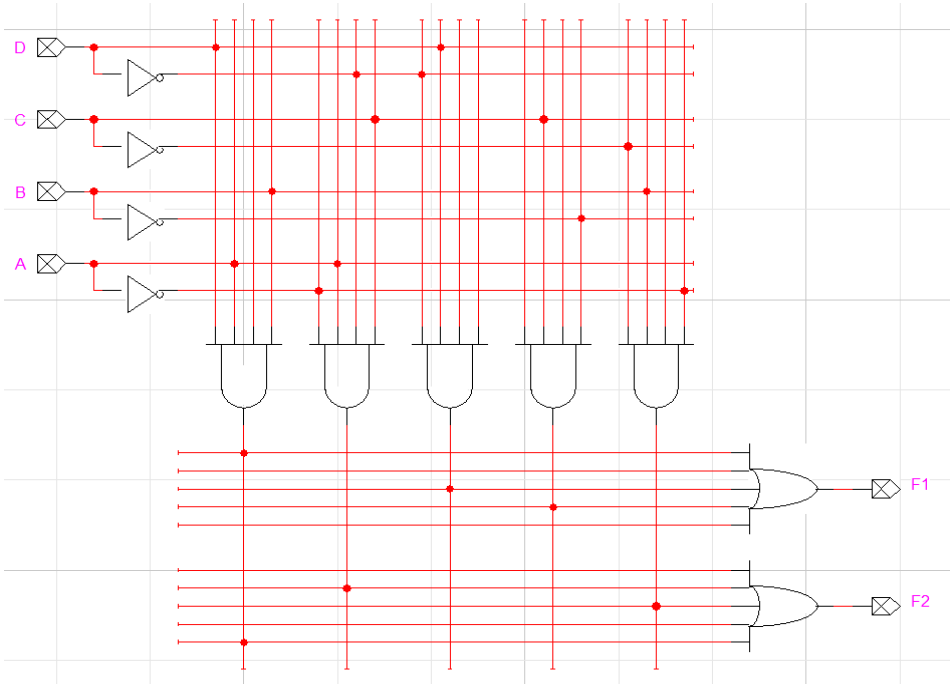
Module 6: Timing *Difficulty* ★★★★☆

18. Draw a timing diagram based on the circuit above and the following information:

- *Inverters have a delay of 1, AND gates a delay of 3, and OR gates a delay of 2.*
- *At t = 0, all inputs are '0'.*
- *C becomes '1' at t = 1, then goes back to '0' at t = 6*
- *B becomes '1' at t = 5*
- *A becomes '1' at t = 2, then '0' again at t = 4*
- *Assume TRANSPORT delay.*
  Also assume that if a variable evaluates as '0' or '1' at t = 0, there is no delay.

Module 8: PLAs *Difficulty* ★★☆☆☆

19. Find the canonical sums for F1 and F2 in the PLA diagram provided above.



Module 8: Multiplexors *Difficulty* ★★★☆☆

20. Given the truth tables below, determine the inputs for an 8-1 multiplexor as a function of A using only {A, A', 0, 1}. That is, find m0, m1, …, m7.

D	C	B	A	F
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1

D	C	B	A	F
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

Module 8: Multiplexors *Difficulty* ★★★★☆

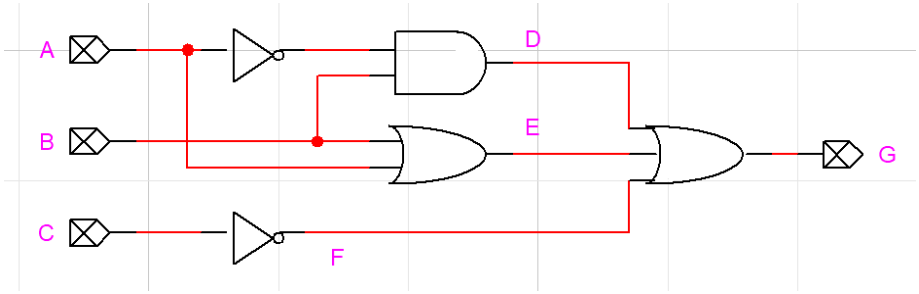
21. Create a truth table for an 8-1 multiplexor as a function of B based on the following: **m0 = 0, m1 = B, m2 = B', m3 = 1, m4 = 0, m5 = B, m6 = B, m7 = 0.**

Module 9.5: VHDL *Difficulty* ★★★☆☆

22. Draw a circuit that represents the VHDL code F <= A when X = '1' else B and not (C or D); Then, based on your answer, what would the output F be when A = 1, B = 1, C = 0, D = 0, and X = 0?

Module 9.5: VHDL *Difficulty* ★★★☆☆

23. Write VHDL code that represents the circuit below given that inverters have 10 ns delay, but other gates have no delay. You do not have to include entity and architecture definitions, just the logic.



Module 7: Hamming

Difficulty ★★☆☆☆

24. Repair the following hamming codes (distance = 1) using **even** parity. Indicate the position of the error and the corrected value

- 0101011
- 1101111
- 0010011
- 0001000

Module 7: Hamming

Difficulty ★★☆☆☆

25. Find Co through C7 for the following triangle code. Parity is odd.

$0\ 1\ 0\ 0\ 1\ 0\ 0\ C_0$
$1\ 0\ 0\ 0\ 1\ 1\ C_1$
$0\ 1\ 1\ 0\ 1\ C_2$
$0\ 1\ 0\ 0\ C_3$
$1\ 1\ 0\ C_4$
$0\ 0\ C_5$
$1\ C_6$
C_7

SECTION 3:

Modules 10-15 Practice

Module 10: Flip-Flops *Difficulty* ★★★☆☆

26. Write the truth tables for a D flip-flop, J-K flip flop, and S-R latch. The tables you need to complete are provided below. You may use don't care states.

D Flip-Flop		
Q	Q+	D
0	0	
0	1	
1	0	
1	1	

J-K Flip Flop			
Q	Q+	J	K
0	0		
0	1		
1	0		
1	1		

S-R Latch			
Q	Q+	S	R
0	0		
0	1		
1	0		
1	1		

Module 14: TTL *Difficulty* ★★☆☆☆

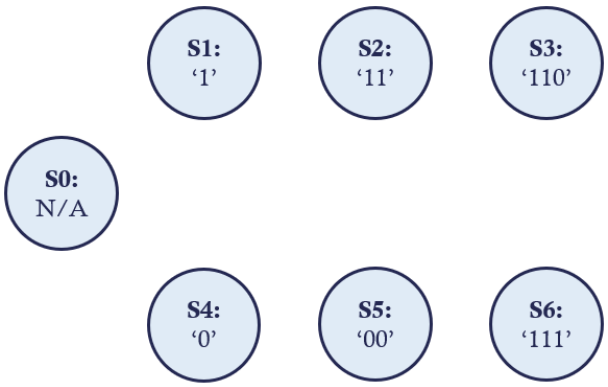
27. Describe how a debouncer works, specifically the NAND implementation that you developed in lab. What kind of named circuit is this analogous to?

Module 12: Sequence *Difficulty* ★★★☆☆

28. Design a state diagram to detect the sequence '0110'. Assume that overlapping is possible, and include a quiescent state. How many D flip-flops would you need for this?

Module 12: Sequence *Difficulty* ★★★★☆

29. Design a state diagram capable of detecting the sequences 1101, 1110, and 001. The "N/A" state is considered quiescent. You must also observe overlap.



Module 12: Transition *Difficulty* ★★★☆☆

30. Create a transition table describing all state transitions for the sequence diagram in the previous question. Start by evaluating how many D flip-flops will you need. Are there any "don't care" states on the table? You can make K-maps for Q^+ and the output if you'd like some additional practice.

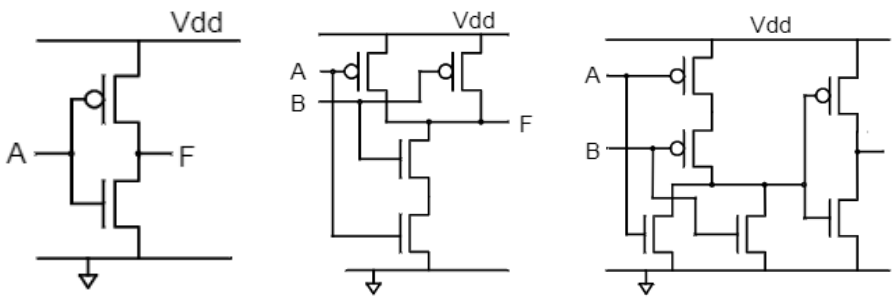
Module 10: JK FF *Difficulty* ★★★★☆

31. Design a counter using ***J-K flip flops*** that counts up by two when input A is a 1, and decrements by 1 when A is a 0. The counter should have a maximum value of 7. Your task is to create the associated transition table.

Hint: If A is a 1, we should see the sequence 0-2-4-6-0… or 1-3-5-7-1…, depending on if the initial state is odd or even. If A is a 0, we should see the sequence 7-6-5-4-3-2-1-0-7…

Module 13: CMOS *Difficulty* ★★★☆☆

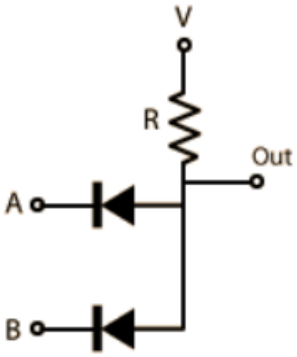
32. Identify the following CMOS chips:



Module 13: TTL/CMOS *Difficulty ★/★★*

33. Answer the following true/false questions:

- CMOS chips take less current than TTL.
- Resistance and current can never be zero.
- For a CMOS transistor, a LOW input results in a low impedance.
- Ohm's Law is $I = VR$
- To add resistance in series, sum the inverses.



Module 14: Diodes *Difficulty ★★☆☆☆*

34. Do the diodes on the right make an AND or an OR gate? Once you've identified it, draw the other one.

Module 10: Shift *Difficulty ★★★☆☆*

35. I have a 4-bit shift register make of D flip-flops. Each D flip-flop is bound to the same CLK input and the same CLR input, but have separate SETs. The register is $Q_3Q_2Q_1Q_0$, where serial input goes to Q_3, and each clock shifts right. Given the following sequence of inputs, determine the final state of the shift register, that is, Q_3 through Q_0.

i. CLR is toggled LOW, then back to HIGH. All SETs are HIGH.
ii. Serial input is HIGH. Two CLK positive edge-triggers are given.
iii. Serial input is LOW. Three CLK positive edge-triggers are given.
iv. SET is toggled LOW for Q_3.
v. One more CLK positive edge-triggers are given.
vi. What am I?

Module 10: PRBS *Difficulty ★★☆☆☆*

36. Create complete sequence diagrams (one input, one output per node) for a 4-bit PRBS where the serial input with each clock is defined by $\overline{Q_2 \oplus Q_0}$. The register is $Q_3Q_2Q_1Q_0$, where serial input goes to Q_3, and each clock shifts right. There will be more than one sequence diagram.

Module 11: VHDL ***Difficulty ★★★★★***

37. **VHDL Hell:** Develop VHDL modules for a D flip-flop, a J-K flip flop, and an S-R latch. You may assume that much of the infrastructure has already been established, but you will have to specify your input and output signals formally.

Note: This problem is practically a full lab. You will never be asked to answer a question of this complexity in an exam setting, especially without notes.

SECTION 4:

Additional Concepts

Module 16: Sampling *Difficulty ★☆☆☆☆*

38. We want to sample an audio wave that has a bandwidth of 10kHz. What frequency should we sample at to retain its quality? What is the name of the criterion you used?

Module 17: SeqMul *Difficulty ★★★☆☆*

39. Show all steps for multiplying 1010 (Multiplicand) and 0101 (Multiplier) using sequential multiplication. Perform addition and shift operations in separate steps.

Register A (Multiplicand)			
1	0	1	0

Step	C	Reg B (Product)				Reg Q (Multiplier)			
0	0	0	0	0	0	0	1	0	1
1									
2									
3									
4									
5									
6									

Module 18: Memory *Difficulty ★/★★*

40. Answer the following true/false questions:

- Semiconductors are non-volatile memory.
- EPROMs are erased by UV light.
- DRAM is used for computer firmware.
- Magnetic memory systems need to be powered to maintain their state.
- SRAM needs to be constantly refreshed to maintain its state.
- If a disk goes down in RAID-0 configuration, another partition can restore it.
- Vacuum tubes are volatile memory.