

**UNIVERSITY OF NEVADA, RENO**  
**ELECTRICAL ENGINEERING DEPARTMENT**

*Sample*

Ph.D. Qualifying Examination - Computer Engineering Section

- Design a combinational logic circuit which has three inputs  $x, y, z$  and two outputs  $F_0, F_1$ . Output [ $F_0$  is equal to  $(x \text{ XOR } y)$  when  $z = 1$ ] and [otherwise  $F_0$  is equal to 1 when  $z = 0$ ]. Output [ $F_1$  is equal to  $(x \text{ AND } y)$  when  $z = 0$ ] and [otherwise  $F_1$  is equal to 0]. Find the **minimized sum of products** equations for  $F_0$  and  $F_1$  and draw a logic diagram for the circuits needed to generate  $F_0$  and  $F_1$ .
- Mark **T** for true or **F** for false **in the boxes below**.

T/F	Indicate for each of the following statements if they are true or false.
	The <i>noise margin</i> of a logic gate defines the voltage interval between logic high and logic low.
	The <i>fanout</i> of a logic gate is the number of inputs that the gate can drive without exceeding its worst-case loading specifications.
	The <i>propagation delay</i> $t_p$ of a signal path is the amount of time that it takes for a change in the input signal to produce a change in the output signal.

- Use the Boolean theorems to simplify the following equation: (show each step and identify the theorems used).

$$q = x + (x'y')' (x' + y z' + x y z')$$

- A Moore sequential network has one input and one output. When the input sequence 011 occurs, the output becomes 1 and remains 1 until the sequence 011 occurs again in which case the output returns to 0. The output then remains 0 until 011 occurs a third time, etc. For example, the input sequence

$$X = 01011010110100111$$

has the output

$$Z = 00001111100000011$$

Derive the STATE GRAPH.  
 Derive the STATE/OUTPUT TABLE  
 Implement the circuit using D Flip-Flops

5. Describe the evolution from an SR latch to a positive edge triggered D flip-flop.
6. Describe in words, using the example function below, how the Quine-McCluskey method works.

$$F(x,y,z) = \sum (1,4,6,8)$$

7. Prove that a 32 X 8 ROM can be used to implement a circuit that generates the binary square of input 5-bit number with output bit B0 = input bit A0 and output bit B1 = 0. Draw a block diagram of the circuit and list the first four and last four entries of the ROM truth table.
8. Consider a hypothetical computer with one n-bit accumulator register and an instruction set of only two n-bit instructions. The most significant bit of each instruction specifies the opcode (0 = SUB X and 1 = JUMP X), and the remaining bits specify the address X of one of  $2^{n-1}$  n-bit words of main memory (assume the arithmetic unit uses two's complement integer arithmetic). This instruction set is surprisingly complete and most other instructions can be derived from the first two. The two instructions are:

SUB X     Subtract the contents of location X from the accumulator register, and store the result in location X **and** in the accumulator register,

JUMP X    Place address X in the program counter.

**Write a program using only the above two instructions to load the accumulator register from a memory location X.**

HINTS:     You can assume that you have some memory locations with known values pre-stored. For example, you can have a location which is known to contain zero. The program can be written in six lines of code.

9. Given the 8051 subroutine below, comment each line of the subroutine:

```

                                ORG          8100H
OUTCHAR:  MOV          C, P
                                CPL
                                MOV          ACC.7, C
AGAIN:    JNB          TI, AGAIN
                                CLR          TI
                                MOV          SBUF, A
                                CLR          ACC.7
                                RET
                                END

```