J  | K | Q(T) | Q(T+1)

0    x     0        0

1    x     0        1

x    1     1        0

x    0     1        1

--Nor and Nand are universal gates

— When doing jk flip flop truth table, you must use the present states to simplify j/k

--module JK\_FF(input J, K, Q, clk, reset, output reg Qn); always @ (posedge clk, reset) begin if (reset) Qn <= 0; else Qn <= J && !Q || !K && Q; end endmodule

(Accumulator code) module acumulator(input clk,reset,output[3:0] y);

     wire [3:0] w1,w2;

     adder\_inst(adder\_inst(.A(y),.B(w2),.Y(w1));

     counter\_inst(.reset(reset),.clk(clk),.count(w2));

     reg\_inst(.din(w1),.dout(y),.reset(reset),.clk(clk));

endmodule

module adder(input [3:0] A,B,output [3:0] y);

module counter (input reset,clk,output reg[3:0] count);

module register(input[3:0] din,clk,reset output [3:0] dout);

--Which of the statements about Verilog is/are true? (A) Any Verilog code that simulates well is correct and can be synthesized. (B) In Verilog, one module can instantiate other modules, and can have multiple instances of another module. (C) Always @ \* can be used in describing Combinational logic. (D) Always @ \* can be used in describing memory made of flip-flops and registers. (E) Always @ \* can be used preceding a case statement. **ANS: BCE**

--Pi and Gi are the Propagate and Generate functions that we discussed in class for a full adder. Suppose Ci is the CARRY-OUT and Ci-1 is the CARRY-IN. Which of the following is the correct Boolean expression for Ci? (A) (Gi’Pi + Gi’Ci-1’)’ (B) (Gi’Pi’ + Gi’Ci-1’)’ (C) (Gi’Pi + Gi’Ci-1’) (D) (Gi’Pi’ + Gi’Ci-1’) (E) Pi + GiCi-1 **ANS B**

--Which of the following regarding a sequential circuit (s.c.) is correct ?(A) A s.c. with six outputs requires at least 3 flip-flops to build. (B) A s.c. with six inputs requires at least 3 flip-flops to build. ( C ) It is possible that a s.c. with three outputs has one input only. (D) A s.c. has five states, 2 inputs and 2 outputs can be built using 2 flip flops. **ANS C**

--Which of the following regarding a sequential circuit (s.c.) is **not** correct ?(A)The outputs of a Moore machine do not depend on the states explicitly.(B) The outputs of a Mealy machine depend on both the inputs and the states.( C ) A s.c. may have no input.(D)In order that a s.c. works properly, the data on an input must be present and stable for a certain amount of time before the clock transition. **ANS A**

--Which is regarding a flip-flop is **not** correct ?(A) If the  Q` output of a flip-flop Q is connected to an inverter, the inverter's output will always be the same as the Q output of the flip-flop. (B) A flip-flop is a primitive device that has memory. (C )The next state of a T flip-flop is given by Q(t+1) = T(t)®Q(t)’ (D) A flip-flop has two states.(E) none of the above is **false**. **ANS A OR C**

--Design a sequential circuit with two D flip-flops, QA and QB, and one input, X. When X = 0, the state of the circuit remains the same. When X = 1, the circuit goes through the state transitions from 00 to 01 to 11 to 10 back to 00, and repeats. Which of the following incorrect ?(A)QA(t+1) = QA(t)X' + QB(t)X (B) DA(t+1) = QA(t)X' + QB(t)X( C) QB(t) = QA(t)X' + QB(t)X(D)  QB(t+1) = QA(t)X' + QB(t)X(E) DB(t+1) = QA(t)X' + QB(t)X **ANS A**

--This question is to test your computation thinking capability. Imagine buying 100 pounds of potatoes and being told that they are 99 percent water. After the potatoes have been left outdoors for a day, you are told that they are now 98 percent water. What is the weight of these slightly dehydrated potatoes in pounds? **ANS <60**

--An 8 x 1 multiplexer has inputs A, B, and C connected to the selection inputs s2, s1, and s0, respectively. The data inputs, I0 through I7are as follows: I1 = I2 = I 7 = 0; I3 = I 5 = 1; I 0 = I 4 = D; and I 6 = D'. Determine the Boolean function that the multiplexer implements. **ANS** F(A,B,C,D) = summation of **(1, 6, 7, 9, 11, 12 )**

—You have used a *ATF16V8B* chip to contruct a 7-segment hex decoder. What is the minimum number of *ATF16V8B* chips you need to use to build a 4x16 decoder if no other kinds of chips are used? **ANS = 2**

— Design a combinational circuit that converts a 4-bit Gray code number to a 4-bit straight binary number. Suppose the inputs are A, B, C, D and the outputs are a, b, c, d. Also, suppose you have found out that a = A. Which of the following is true ? **ANS**E. **d = c ® D**

— Which of the following is **false** ? The symbol ® means exclusive OR. **ANS** E. **(X ® Y)' = (Y ® X)X'Y + XY'**

— In the 1.15 format, where 2's complement representation is used, which of the following represents **-0.75**? **ANS**B. **1010000000000000**

— How many address lines and input-output data lines are needed in a 64M x 16 memory unit ? **ANS**B. **26, 16**

— A pc uses RAM chip of 2048 x 1 capacity. How many chips needed to provide memory capacity of 16K bytes ? **ANS B. 68**

— How many flip-flops will be complemented in a 10-bit binary ripple counter to reach the next count after the count 0011111111 ? **ANS C) 9**

— A number of 128 x 8 RAM chips are used to provide a memory capacity of 2048 bytes. What kind of decoder do you use to accomplish this ? **ANS**D. **4 x 16**

— A 7-bit data is encoded using the Hamming code and is transmitted. The received codeword is : 1 0 1 0 0 1 1 1 1 0 1 . Suppose the leftmost position is position 1. Where is the corrupted bit if there's any. ANS **B. Position 9**

— An 8-bit data is encoded using the Hamming code and is transmitted. The received codeword is : 1 0 1 0 0 1 1 1 0 0 0 0. Suppose the leftmost position is position 1. Where is the corrupted bit if there's any. **ANS A. Position 11**

— A 12-bit Hamming-code word containing 8 bits of data and 4 parity bits is read from memory. What was the original 8-bit data that was written into memory if the 12-bit word read out is 000011101010 ? **ANS**C. **01011010**

— Consider a Moore machine with two inputs, one flip-flop, and four outputs. Which of the following regarding the state, transition, and output bounds is/are **not** true? (Your answer could be only one choice.)

**ANS** C.  **The minimum number of state transitions that can end in a given state is 1.**

— A flip-flop has a 10 nanosecond delay from its clock input going from 1 to 0 to the time when the output is complemented. What is the maximum frequency a 20-bit binary ripple counter that uses these flip-flops can operate? **ANS**B.**5 MHz**

— What is the Hamming-codeword for the message 0010 ? **ANS**A. **0101010**

— What is the minimum number of check bits required to encode a message with 120 bits using the Hamming Code? **ANS B. 7**

— Which of the following is **not** true if Hamming Code is used to encode a message and an extra parity bit is added to the coded-word to enhance the functions? **ANS**C. **It cannot correct the error if it occurs a the parity bit.**

— What is the minimal size of the ROM ( number of words and number of bits per word ) that will accommodate the truth table for the combinational circuit of a binary multiplier that multiplies two 8-bit numbers. **ANS**E. **64K x 16**

**— Parity bit/ Hamming code check: First you need to label all the parity bits which are powers of 2, so 1,2,4,8. After that do checks, for #1 you start at 1, skip 1, start at 1 skip 1… etc. Then for Check #2, start at 2, get 2, skip 2, get 2, skip 2, etc. For check #3, you need to start at position 4, get 4, skip 4. etc and same thing for 8, start at 8 skip 8. Even number of 1 gives 0 Odd gives 1. After, check if all parity bits = to the ones you got. If one or more don’t agree, add up the positions. Say if we got parity bit 2 and 8 wrong. Add them up, then position 10 is the wrong one. All we have to do is switch 1/0 or 0/1 to make it right.**

module mux\_2to1(input wire [1:0] D, input wire S, output wire out); assign out = S ? D[1] : D[0]; endmodule || module mux\_4to1(input wire [3:0] D, input wire [1:0] S, output Y); wire [1:0] O; mux\_2to1 m0(D[1:0], S[0], O[0]); mux\_2to1 m1(D[3:2], S[0], O[1]); mux\_2to1 m2(O, S[1], Y); endmodule

module tb\_mux\_4to1; reg [3:0] D; reg [1:0] S; wire Y; mux\_4to1 uut(.D(D), .S(S), .Y(Y)); initial begin D = 0; S = 0; $display("I3 I2 I1 I0 | S1 S0 | Y"); $display("------------|-------|--"); $monitor("%2d %2d %2d %2d | %2d %2d | %d", D[3], D[2], D[1], D[0], S[1], S[0], Y); #630 $finish; end always begin #10 S = S + 1'b1; end always begin #40 $display("------------|-------|--"); end always begin #40 D = D + 1'b1; end endmodule