Name Key

EE3610 Midterm Exam (Fall 2012). 6 Pages. Open book. Open Notes. Closed Internet.
Cheating will not be tolerated and will result in grade of 0 for this exam.

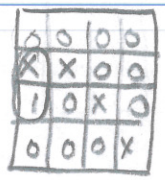
Part 1. Short Answer.

1. (2 pts) There are two predominant hardware description languages. Name them.
   (1) VHDL
   (2) Verilog
2. (3 pts) What is the difference between a structural and a behavioral model?
   Behavioral describes what the model does, Structural describes how (components & interconnect).
3. (2 pts) If the order of two concurrent statements in a VHDL model are changed:
   (a) The behavior of the model is certain to change.
   (b) The behavior of the model may change.
   (c) The behavior of the model is certain not to change.
   (d) A paradox is generated that may destroy the fabric of the universe.
4. (2 pts) (true/false) A process that is supposed to model combinational logic should drive all the outputs each time it runs, no matter what the inputs are.
5. (2 pts) (true/false) A process that is supposed to model sequential logic should drive all the outputs each time it runs, no matter what the inputs are.
6. (2 pts) (true/false) Pipelining is a technique that decreases latency (the time it takes to get a result) but also decreases throughput (the number of results per second).
7. (2 pts) (true/false) The first programmable electronic logic device was the ROM.
8. (2 pts) (true/false) The principal difference between a PLA and a PAL is the PLA has a programmable OR array, but in the PAL the OR array is fixed.
9. (6 pts) Suppose a 22V10 is used to implement a state machine with 7 states, 4 inputs and 2 outputs.
   (a) How many output pins must be dedicated to state bits plus outputs? 5
   (b) How many input pins are needed (don't forget clock and reset)? 6
   (c) How many unused pins will there be? 11
10. (2 pts) (true/false) an FPGA consists predominantly of programmable logic blocks and programmable interconnections.
11. (2 pts) What does BCD stand for?
    Binary Coded Decimal
12. (2 pts) Under the IEEE 1164 standard, what is the state of a signal that is simultaneously driven with a '0' and an 'Z'? '0'
13. (2 pts) (true/false) In VHDL, it is impossible to have a constant, signal or variable that is an array without declaring the array type first.
14. (3 pts) (true/false) I would like another 3 points on this exam.

Part 2 - Problems

15. (5 pts) Loop and give the simplest SOP expression for the given Variable Entered map

CD \ AB	00	01	11	10
00	$\bar{E}$	$\bar{E}$	0	0
01	X	1	E	E
11	1	0	X	0
10	0	0	0	X



F = $\bar{A}\bar{C}\bar{E} + \bar{C}DE + \bar{A}\bar{B}D$

16. (5 pts) Write a VHDL process (not the whole module) to model a T flip-flop with an asynchronous reset.

```
process (clk, rst)
begin
   if rst = '1' then
      Q <= '0';
   elsif rising_edge(clk) then
      if T = '1' then
         Q <= not Q;
      end if;
   end if;

end process;
```

17. (5 pts) Write a concurrent statement to model a 4-bit adder with carry input. Assume that addends (A and B) are unsigned(3 downto 0) and the carry in (Ci) is std_logic. The output is a 5-bit sum (S), whose type is unsigned(4 downto 0).

```
S <= ('0' & A) + B + unsigned'(0 => Ci);
```

18. (15 pts) Each of the VHDL code fragments below can be used to model a physical device. Tell what kind of device it is (e.g. D- flip-flop, MUX, gate, shift register, etc.) and name its features, if any (e.g. synchronous reset, asynchronous reset, clock enable, parallel load, number of MUX inputs, etc.)

(a)
```
process(A, E)
begin
    if E = '1' then
        Y <= A;
    else
        Y <= 'Z';
    end if;
end process;
```

Device Tri-state buffer

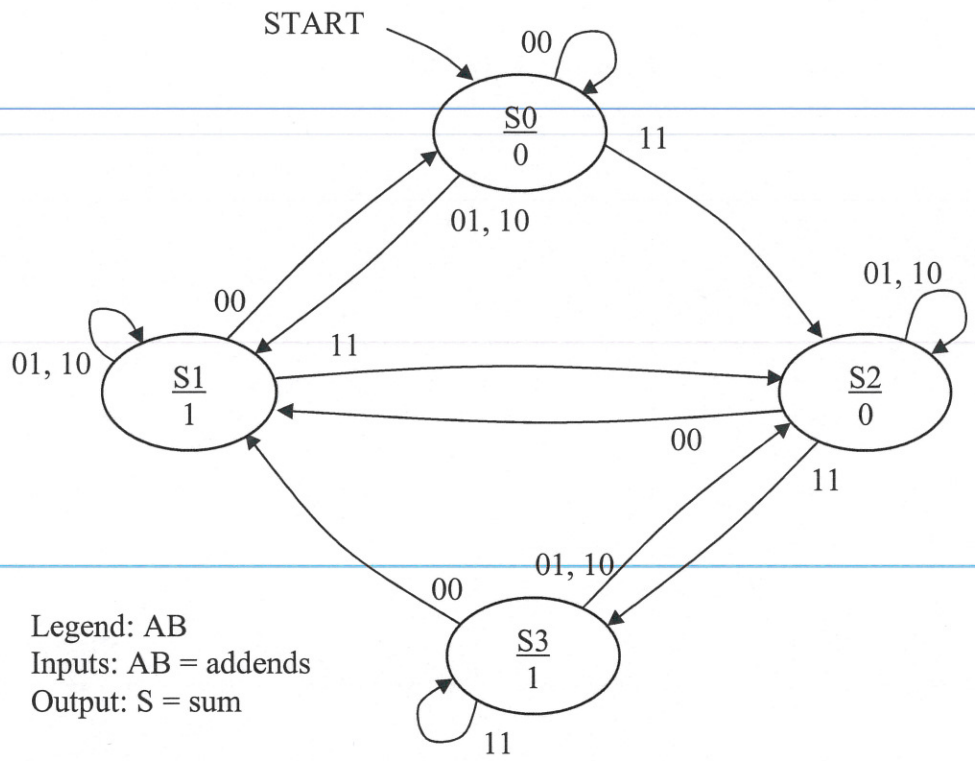
(b)
```
process(CLK)
begin
    if rising_edge(CLK) then
        if rst = '1' then
          Q <= (others=>'0');
        elsif PE = '1' then
          Q <= ParallelData;
        elsif SH= '1' then
          Q <= DSR & Q(7 downto 1);
        end if;
    end if;
end process;
```

Device Parallel-in Shift register with Synchronous Reset

(c)
```
Y <=  I0 when S= "00" else
      I1 when S= "01" else
      I2 when S= "10" else
      I3;
```

Device 4-1 MUX

19. The state diagram below represents a serial adder. (Addends A and B arrive one bit at a time, least significant bit first, and the sum, S, is delivered in the same manner.)



Legend: AB
Inputs: AB = addends
Output: S = sum

(a). (4 pts) Write the entity declaration for this module. Give the module the name ser_add.

```
entity ser_add is
  port (clk, rst, A, B : in std_logic; S : out std_logic);
end ser_add;
```

(b). (4 pts) Declare any signals, types, constants, etc. you will need for this module between the architecture and begin keywords below:

```
architecture behavioral of ser_add is
  type state_type is (S0, S1, S2, S3);
  signal D, Q : state_type;

begin
```

(c). (4 pts) Write VHDL code to model the state register.

```
Process (clk, rst)
begin
  if rst = '1' then
     Q <= S0;
  elsif rising_edge(clk) then
     Q <= D;
  end if;
end process;
```

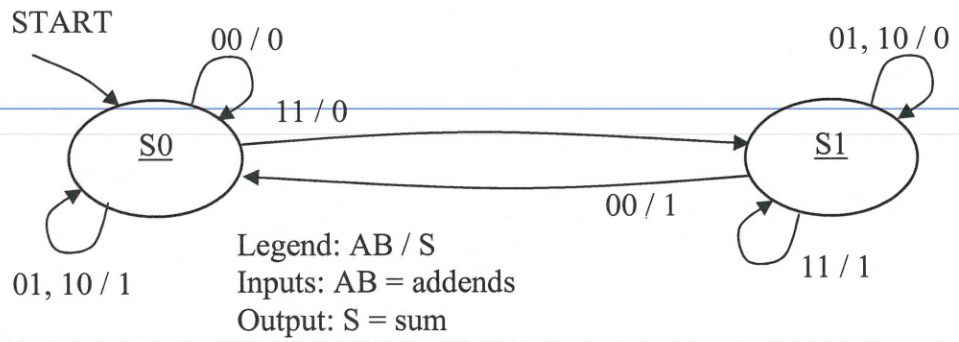
(d). (8 pts) Write VHDL code to model the next state decoder. To save time, you may replace states S1 and S2 in the case statement with <u>when S1=>…</u> and <u>when S2=>…</u> respectively.

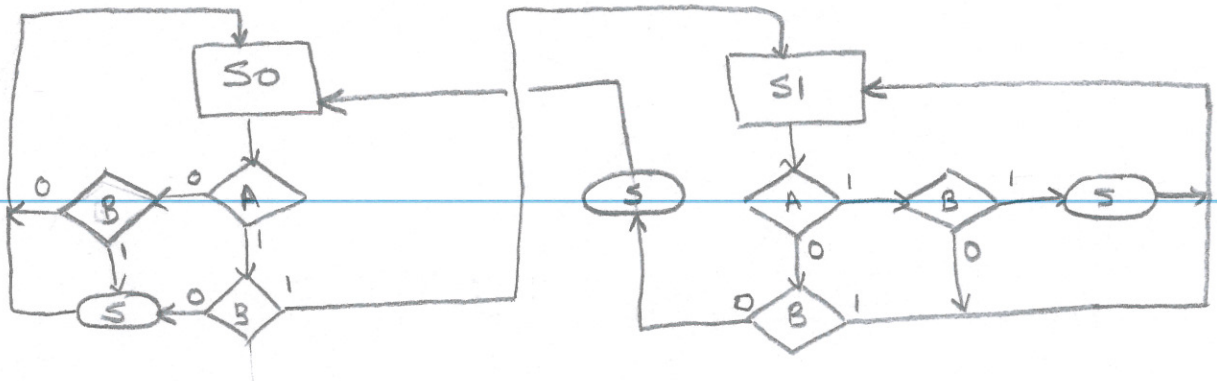
```
Process (Q, A, B)
begin
   Case Q is

      When S0 =>
         if A&B = "00" then D <= S0;
         elsif A&B = "11" then D <= S2;
         else D <= S1;
         end if;
      When S1 => ...
      when S2 => ...

      When S3 =>
         if A&B = "00" then D <= S1;
         elsif A&B = "11" then D <= S3;
         else D <= S2;
         end if;
   end case;
end process;
```

(e). (4 pts) Write a VHDL statement to model the output decoder

```
S <= '1' when Q = S1 or Q = S3 else '0';
```

20. (10 pts) A Mealy version of the serial adder is given below. Draw an SM chart that is equivalent to this state diagram.





21. (2 pts) Comparing the Moore state machine in problem 19 with the Mealy state machine in problem 20:
(a) The Moore state machine generates its output earlier than the Mealy state machine.
(b) The Mealy state machine generates its output earlier than the Moore state machine.
(c) Both state machines generate their output at the same time.

Extra Credit (5 pts) Draw a state diagram similar to the one in problem 20 that implements a serial subtractor.