Department of Electrical and Computer Engineering
University of Alabama in Huntsville

EE/CPE 422/522 — Advanced Logic Design
Summer 2003
Final Exam

Instructor: Dr. Aleksandar Milenkovic

Date: July 30, 2003

Place: EB 207

Time: 6:00 PM — 8:30 PM

Note: Work should be performed systematically and neatly. This exam is closed book, closed notes, closed neighbour(s). Allowable items include exam, pencils, straight edge, and calculator. Best wishes.

Question	Points	Score
1	10	
2	20	
3	15	
4	10	
5	10	
6	10	
7	15	
8	10	
Sum		

Please print in capitals:

Last name:_____

First name: _____

1. (10 points)

a. (2) Explain VHDL GENERATE statement.

b. (2) Give VHDL statement to declare a text file named test_file opened in read mode. External file name is C:\test_file.txt . Show a VHDL code segment that reads an integer from the test_file?

c. (3) Give results of the following expressions assuming that A = 110 , B= 111 , C= 011000 , and D= 111011 (A, B, C, and D are bit vectors).

A & (not B)	
C ror 2	
D sra 2	
(D and C) = 011001	
A sla 1	
010 & 1	

d. (3) Fill in the following table.

```
type 2D_Array is array (0 to 4, 5 downto 1) of bit;
signal ms : 2D_Array;
```

Expression	Result
ms left(1)	
ms right(2)	
ms high(2)	
ms range(1)	
ms reverse_range(2)	
ms length(2)	

2. (20 points)

(a) (2) The following two code fragments are equivalent (Yes / No).

Fragment #1:

```
wait until Clk = '1';
```

Fragment #2:

```
wait on Clk until Clk = '1';
```

(b) (3) Circle the fragment of code below that suspends a process for a *maximum* of 100 us.

Fragment #1:

```
wait on sig_a, sig_b for 100 us;
```

Fragment #2:

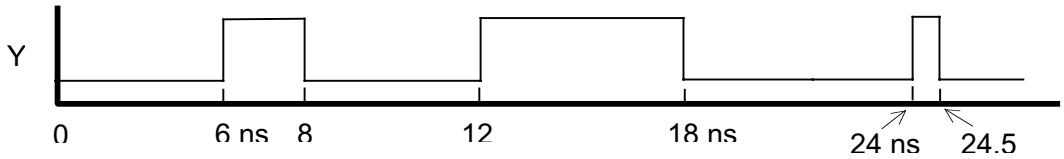
```
wait on sig_a, sig_b;
wait for 100 us;
```

(c) (4) The following two architectures are given. Tell if the two architectures are equivalent, i.e. do they give exactly the same behaviour. Briefly explain why or why not.
Assume that clk, q, and d are defined in the entity declaration.

```
architecture first of dff is
begin
process (clk)
begin
if (clk'event and clk='1')
     then q <= d after 2 ns;
end if;
end process;
end first;
```

```
architecture second of dff is
begin process
begin
if (clk'event and clk='1')
     then q <= d after 2 ns;
end if;
wait on clk;
end process;
end second;
```

(d) (6) Draw waveforms for the following signal assignments given the input waveform below. Be sure to clearly label the time when an event occurs.

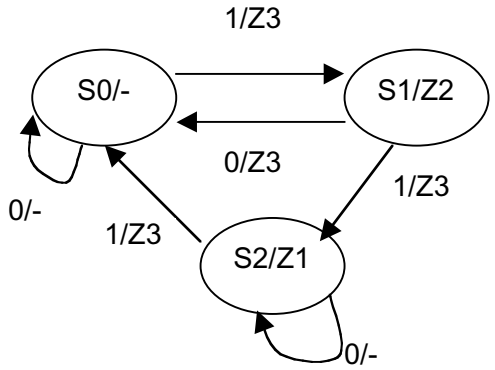


(i) (2)
```
X <= Y after 5 ns;
```

(ii) (2)
```
X <= transport Y after 5 ns;
```

(iii) (2)
```
X  <= reject 1 ns inertial Y after 5 ns;
```

(e) (5) Derive an SM chart (State Machine Flowchart) for the following state diagram.



3. (15 points)
(a) (10) Write a VHDL function that converts a 5-bit bit_vector to an integer.
How much simulated time will it take for your function to execute?

```
function vec2int(signal vec5 : bit_vector(4 downto 0)) return integer is

begin









end vec2int;
```

(b) (5) Write a VHDL function vec2int that places no restrictions on the size and range of the input vector.

4. (10 points) Consider the following fragment of a VHDL code.

```
type My4 is ('a', 'b', 'c', 'd');
type My4_vector is array(natural range <>) of My4;
function myresolve(s: My4_vector) return My4;
subtype My4R is myresolve My4;
...

signal R : My4R;
```

The resolution function **myresolve** realizes the signal resolution using the following table.

	a	b	c	d
a	a	a	a	a
b	a	b	b	b
c	a	b	c	c
d	a	b	c	d

Give values of the signal R in time interval from 0 ns to 20 ns assuming that it s driven by the following concurrent signal assignment statements.

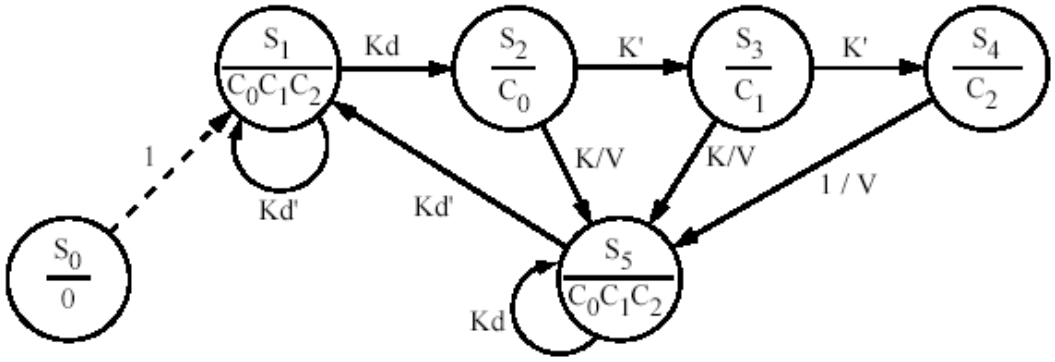
```
R <= transport 'a' after 2 ns, 'b' after 4 ns;
R <= transport 'c' after 2 ns, 'd' after 2 ns;
R <= transport 'b' after 1 ns, 'c' after 5 ns;
```

5. (10 points) Write a VHDL architecture of the following register file with two read ports and one write port.

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_unsigned.all;
entity REGS is
  port (
    CLK    : in  std_logic; -- clock signal
    WREN   : in  std_logic; -- write enable
    WAD    : in  std_logic_vector( 3 downto 0); -- write address port
    DIN    : in  std_logic_vector(31 downto 0); -- data write port
    RAD1   : in  std_logic_vector( 3 downto 0); -- read address port 1
    RAD2   : in  std_logic_vector( 3 downto 0); -- read address port 2
    DOUT1  : out std_logic_vector(31 downto 0); -- read data port 1
    DOUT2  : out std_logic_vector(31 downto 0)); -- read data port 2
end REGS;
architecture RTL of REGS is
--- write your code here
```

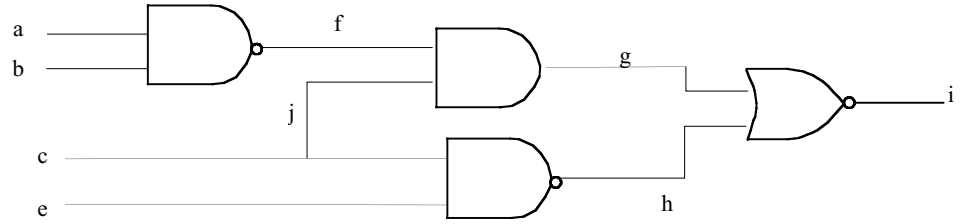
6. (10 points) Write a generic VHDL model (the size is specified during instantiation) that describes a counter with the following characteristics. Signal **Cln** (Clear) is asynchronous and active low. All other state changes occur on the **falling** edge of the **Clock** signal. If the control inputs S0=S1=1 the counter is loaded in parallel. If S0=0, S1 = 1, the counter counts up, if S0=1, S1=0 it counts down. If S0=S1=0, no action occurs.

7. (15 points) Write a VHDL code for the following FSM.



8. (10 points)

a. (5 points) Determine the necessary inputs to the following network to test **f** for **stuck-at-0**. Give the value of the output **I** when the fault in the presence and absence of the fault on line f.



b. (5 points) Determine the necessary inputs for the following network to test signal **e** for **stuck-at-1**. Determine which other stuck-at faults can be tested with this test vector.