May 15, 2007

COMPUTER ENGINEERING DEPARTMENT

COE 405

DESIGN & MODELING OF DIGITAL SYSTEMS

Major Exam II

Second Semester (062)

Time: 7:00-9:30 PM

OPEN BOOK EXAM

Student Name : ____

Student ID. : ____

Question	Max Points	Score
Q1	20	
Q2	30	
Q3	20	
Q4	30	
Total	100	

Dr. Aiman El-Maleh

[20 Points]

(Q1) Given the following Entity Description for a JK Flip-Flop:

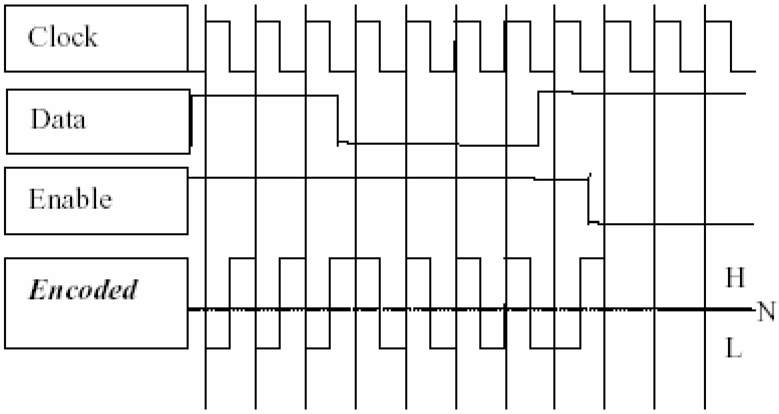
```
ENTITY JKFF IS
   GENERIC (delay : TIME := 4 NS);
   PORT (j, k, clk, reset : IN BIT; q, qb : OUT BIT);
END JKFF;
```

(i) Model the JK-FF using a **BLOCK** statement with a GUARD expression and **slective signal assignment,** assuming that reset is synchronous and the JK-FF is rising-edge triggered.

(ii) Model the JK-FF using a **BLOCK** statement with a GUARD expression and **conditional signal assignment**, assuming that reset is asynchronous and the JK-FF is rising-edge triggered.

[30 Points]

(Q2) Manchester bit-encoding is used in Ethernet and IEEE 802.3 standard. You are to generate a signal with bit information in Manchester encoding. Each bit has a fixed length, *bit_time*. A bit 0 is encoded as high, H, for the first half of a *bit_time* and low, L, for the second half. A transition from H to L indicates the presence of a 0 on a line. Likewise, bit 1 is encoded as a transition from L to H. When no data is on a line, the line stays at the null or N value. This scheme of coding combines a clock signal, a data signal, and an enable line all into one signal. As in the example shown below, a data signal that is synchronized with the rising edge of a clock signal, along with its synchronizing clock and its enable line, are combined into a line containing bit-encoded information, *be*. You are to write a VHDL model to generate the Manchester encoded signal *be*. The data signal is sampled on the rising edge of the clock and if it is 1, a LH transition is made on the *be* line, and if it is 0, a HL transition is made. If the enable is off, the *be* line remains at the N value.



(i) Given the entity **Encode** shown below, write an architecture **Manchester** to produce the encoded signal using **conditional signal assignment**.

Entity Encode **IS**
  Port (clock, data, enable: **IN** bit; be: **OUT** : bit_encode);
  Generic (bit_time: Time := 100 ns);
End Encode;

Assume that type *bit_encode* is defined as shown below and that it is stored in a package *test_utilities* in the work library:
Type *bit_encode* **IS** (N, L, H);

(ii) Write a test bench to test the Architecture **Manchester** of Entity **Encode.** Assume that the input data will be read from a file **data.txt** and it has the format shown below. Assume that the default bit_time = 100 ns is used. You need to generate the clock signal inside the test bench with a period of bit_time and 50% duty cycle.

Time	Data	Enable
0 ns	0	1
280 ns	1	1
460 ns	0	1
840 ns	1	0

[20 Points]

(Q3) A generic **n-bit multiplier** can be modeled using BIT_VECTOR type as shown below:

```
Use work.exam_utility.all;

Entity MUL is

    Generic (n : integer :=4);

    Port (a, b : IN bit_vector(n-1 downto 0); c: OUT bit_vector(2*n-1 downto 0));

End MUL;

Architecture example of MUL is

Begin

    c <= a * b;

End example;
```

For this architecture to work properly, we need to define the following function in the **exam_utility** package.

Function "*" (x, y : BIT_VECTOR) Return BIT_VECTOR

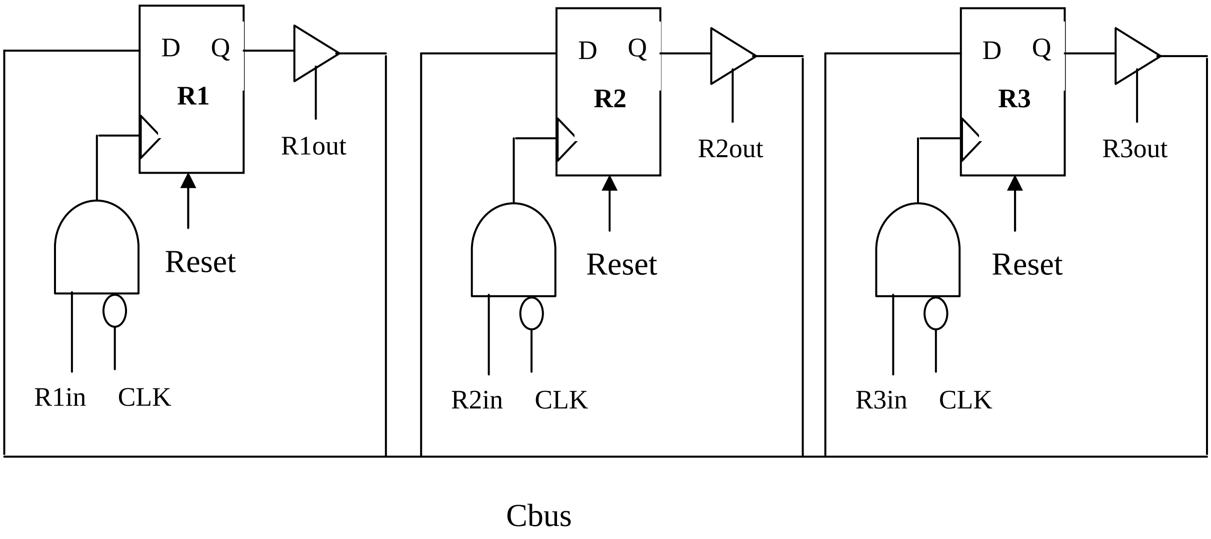
(i) Define the following function assuming **unsigned** multiplication:

Function "*" (x, y : BIT_VECTOR) Return BIT_VECTOR.

(ii) Define the package **exam_utility** and put all the required functions and procedures to be used for the MUL architecture in it.

[30 Points]

(Q4) The three **n-bit** registers R1, R2, and R3, are connected through a tri-state bus (Cbus) to allow the transfer of the content of any register to any other register as shown below:



Assume that the delay across a tri-state buffer is **4 ns** and the flip-flop propagation delay is **2 ns**. Assume that the signals are of type **qit** defined as: **Type qit is ('0', '1', 'Z', 'X');**

(i) Describe an Entity **Datapath** showing the inetrface signals assuming R1, R2, R3 as **output** signals, CLK, Reset, R1in, R2in, R3in, R1out, R2out, and R3out as **input** signals, Cbus as **input/output** signal. Assume that Reset is **Asynchronous** reset. Use Genric **n** for determining the width of registers, **bufferdelay** for the delay across a tri-state buffer, and **ffdelay** for the flip-flop propagation delay.

(ii) Describe all needed types, subtypes and functions in a **Utility** package and use that utility package.

(iii) Describe an architecture **dataflow** for the Entity.