

ELEX 7660 : Digital System Design
Term 201810

FINAL EXAM
1:30 – 4:30 PM
Friday, April 20, 2018

This exam has five (5) questions on ten (10) pages. The marks for each question are as indicated. Answer all questions. Write your answers and all rough work in this paper and nowhere else. Show your work. Books and notes are allowed. No electronic devices other than calculators are allowed.

This exam paper is for:

Exam 1 A00123456

Each exam is equally difficult.
Answer your own exam.

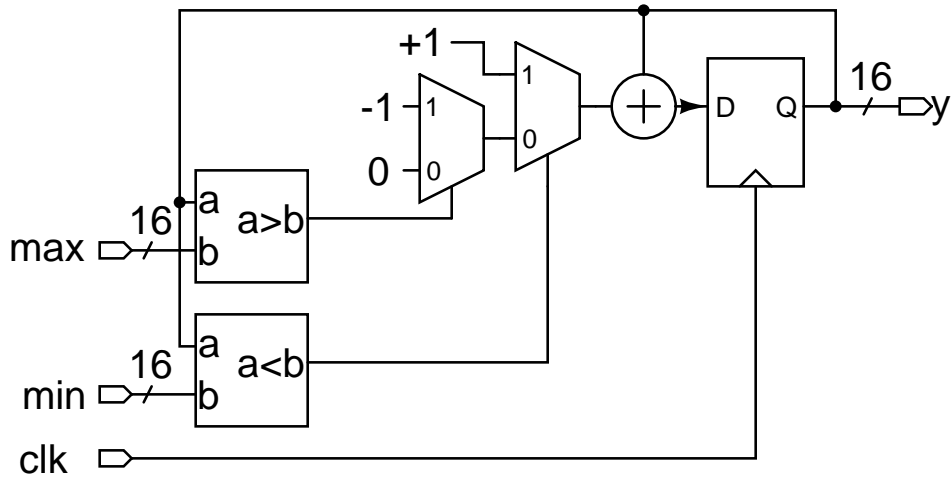
Do not start until you are told to do so.

Name: _____
BCIT ID: _____
Signature: _____

Question	Mark	Max.
1		5
2		5
3		5
4		5
5		4
Total		24

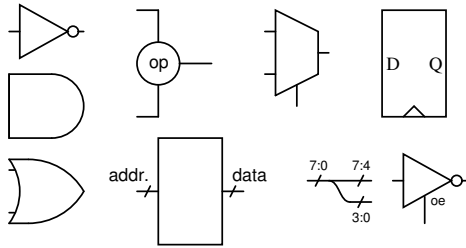
Question 1 (5 marks)

On the facing page write a System Verilog module that models the circuit shown below. Use logic signal types throughout. Include all code from module to endmodule. Where the order of declarations or statements is not significant you may use any order.



Question 2 (5 marks)

Using the following symbols:



draw a schematic for the following Verilog module on the facing page. Label the inputs and outputs. Indicate the bus width if it's more than 1 bit.

```

module check
  ( output logic [7:0] sr,
    input logic nb, dup,
    input logic reset, clk ) ;

  logic [7:0] sr_next ;

  always_comb begin
    if ( reset )
      sr_next = 8'hdd ;
    else
      if ( dup )
        for ( int i=0 ; i<8 ; i++ )
          sr_next[i] = nb ;
        else
          sr_next = sr*2 + (sr[0]^sr[4]) ;
    end

  always@(posedge clk)
    sr <= sr_next ;

endmodule

```


Question 3 (5 marks)

A controller has two 1-bit inputs: `button` and `clk` and two 8-bit outputs: `a` and `b`. The outputs change only on the rising edge of `clk`.

The controller can be in one of three states numbered 0 to 2. On each rising edge of the `button` input (when it changes from 0 to 1) the controller switches to the next state. From state 2 the controller returns to state 0. The state should change only once for each rising edge of `button`.

Outputs `a` and `b` change as follows in each state:

state	behaviour
0	<code>a</code> is set to <code>8'hff</code> and <code>b</code> is set to <code>8'h00</code>
1	<code>a</code> decreases by one and <code>b</code> increases by one on each rising edge of <code>clk</code>
2	<code>a</code> and <code>b</code> do not change

You can assume the controller starts off in state 0. You need not handle overflow or underflow of `a` or `b`.

On the facing page write a *synthesizable* System Verilog module called `bcontrol` that operates as described above. Use logic signal types throughout. Include all code from `module` to `endmodule`.

Question 4 (5 marks)

A C function to compute the greatest common divisor (GCD) of the integers a and b is:

```
void gcd(int a, int b)
{
    int done = 0 ;
    while ( !done ) {
        if ( a > b )
            a = a - b ;
        else if ( a < b )
            b = b - a ;
        else
            done = 1 ;
    }
}
```

On the facing page write a *synthesizable* System Verilog module with the declaration:

```
module gcd (
    input logic [15:0] a_in, b_in,
    output logic [15:0] a, b,
    output logic done,
    input logic clk, reset ) ;
```

that computes the GCD of a_in and b_in .

On each rising edge of clk , if $reset$ is asserted then a and b are set to a_in and b_in respectively and $done$ is set to 0; otherwise the next values of a , b and $done$ are computed.

Use logic signal types throughout. Include all code from module to endmodule.

Question 5 (4 marks)

An asynchronous input drives a two-flip-flop synchronizer. The input changes at an average rate of 1 kHz and the flip-flop is clocked at 50 MHz. For this flip-flop the values of C_1 and C_2 are both 2 ns. Assuming t_{CO} of 5 ns, what is the mean time between invalid inputs at the synchronizer's second flip-flop? Show your work.

ELEX 7660 : Digital System Design
Term 201810

FINAL EXAM
1:30 – 4:30 PM
Friday, April 20, 2018

This exam has five (5) questions on ten (10) pages. The marks for each question are as indicated. Answer all questions. Write your answers and all rough work in this paper and nowhere else. Show your work. Books and notes are allowed. No electronic devices other than calculators are allowed.

This exam paper is for:

Exam 2 A00123456

Each exam is equally difficult.
Answer your own exam.

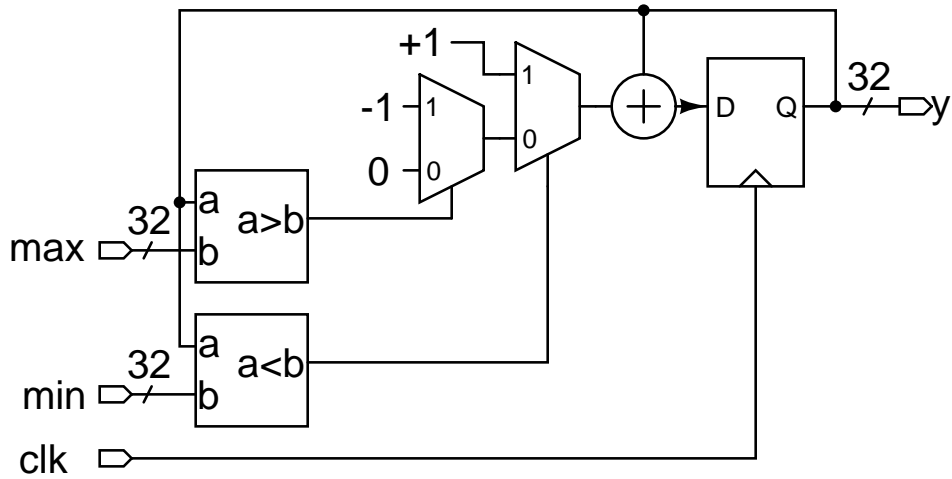
Do not start until you are told to do so.

Name: _____
BCIT ID: _____
Signature: _____

Question	Mark	Max.
1		5
2		5
3		5
4		5
5		4
Total		24

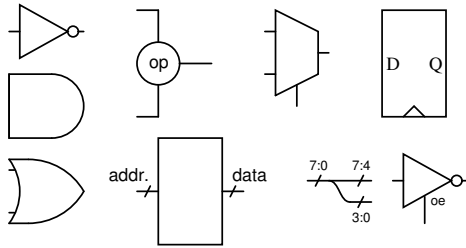
Question 1 (5 marks)

On the facing page write a System Verilog module that models the circuit shown below. Use logic signal types throughout. Include all code from module to endmodule. Where the order of declarations or statements is not significant you may use any order.



Question 2 (5 marks)

Using the following symbols:



draw a schematic for the following Verilog module on the facing page. Label the inputs and outputs. Indicate the bus width if it's more than 1 bit.

```

module check
  ( output logic [7:0] sr,
    input logic nb, dup,
    input logic reset, clk ) ;

  logic [7:0] sr_next ;

  always_comb begin
    if ( reset )
      sr_next = 8'h55 ;
    else
      if ( dup )
        for ( int i=0 ; i<8 ; i++ )
          sr_next[i] = nb ;
      else
        sr_next = sr*2 + (sr[0]^sr[3]) ;
    end

    always@(posedge clk)
      sr <= sr_next ;

endmodule

```


Question 3 (5 marks)

A controller has two 1-bit inputs: `button` and `clk` and two 8-bit outputs: `a` and `b`. The outputs change only on the rising edge of `clk`.

The controller can be in one of three states numbered 0 to 2. On each rising edge of the `button` input (when it changes from 0 to 1) the controller switches to the next state. From state 2 the controller returns to state 0. The state should change only once for each rising edge of `button`.

Outputs `a` and `b` change as follows in each state:

state	behaviour
0	<code>a</code> increases by one on each rising edge of <code>clk</code> , <code>b</code> does not change
1	<code>b</code> increases by one on each rising edge of <code>clk</code> , <code>a</code> does not change
2	<code>a</code> and <code>b</code> are set to zero

You can assume the controller starts off in state 0. You need not handle overflow or underflow of `a` or `b`.

On the facing page write a *synthesizable* System Verilog module called `bcontrol` that operates as described above. Use logic signal types throughout. Include all code from module to `endmodule`.

Question 4 (5 marks)

A C function to compute the greatest common divisor (GCD) of the integers a and b is:

```
void gcd(int a, int b)
{
    int done = 0 ;
    while ( !done ) {
        if ( a > b )
            a = a - b ;
        else if ( a < b )
            b = b - a ;
        else
            done = 1 ;
    }
}
```

On the facing page write a *synthesizable* System Verilog module with the declaration:

```
module gcd (
    input logic [31:0] a_in, b_in,
    output logic [31:0] a, b,
    output logic done,
    input logic clk, reset ) ;
```

that computes the GCD of a_in and b_in .

On each rising edge of clk , if $reset$ is asserted then a and b are set to a_in and b_in respectively and $done$ is set to 0; otherwise the next values of a , b and $done$ are computed.

Use logic signal types throughout. Include all code from module to endmodule.

Question 5 (4 marks)

An asynchronous input drives a two-flip-flop synchronizer. The input changes at an average rate of 1 kHz and the flip-flop is clocked at 50 MHz. For this flip-flop the values of C_1 and C_2 are both 2 ns. Assuming t_{CO} of 5 ns, what is the mean time between invalid inputs at the synchronizer's second flip-flop? Show your work.