

**ECE 464 / ECE 520/ DS 510P**  
Midterm 2002

Each question is worth 2 points. The correct answer earns 2 points, the incorrect 0. You have up to two hours to take this test. Answer all of the questions. Off-campus students, please return this test completed by the end of March.

The exam is open book, open notes.

Write your name and student number on the bubble sheets in the specified locations. I would like to display final scores against your student number. If you do NOT wish this for your test results, please give your name to the TAs during the exam.

**Question 1**

Consider the following code fragment

```
always@(A or B)
    if (B) C = A;
always@(posedge clock)
    if (D) E <= C;
```

There is a potential synthesis problem with this code fragment. Which is the potential problem and fix?

A. Unintentional latches.

```
always@(A or B)
    if (B) C = A;
    else C = ~A;
always@(posedge clock)
    if (D) E <= C;
```

B. Unintentional latches.

```
always@(A or B)
    if (B) C = A;
    else C = ~A;
always@(posedge clock)
    if (D) E <= C;
    else E = ~C;
```

C. Incorrect use of blocking assignment.

```
always@(A or B)
    if (B) C <= A;
always@(posedge clock)
    if (D) E <= C;
```

D. Incomplete sensitivity list.

```
always@(A or B)
```

```
        if (B) C = A;
always@(posedge clock or C)
        if (D) E <= C;
```

E. None of the above

### Question 2

Which of the following outcomes are most likely to be predicted by Moore's Law:

- A. That by the year 3000 AD, chips will have more transistors in them than there are atoms in the Universe.
- B. That the number of transistors manufacturable in a chip will stop increasing at the traditional rate within eighteen months.
- C. That the semiconductor industry will continue to deliver new chips every year with novel capabilities and increased performance over their predecessors.
- D. The cost of a computer will halve every 18 months.
- E. None of the above.

### Question 3

Which of the following statements are **not** true about FPGAs vs. Standard Cell ASICs?

- A. FPGAs provide lower functional density and clock performance than standard cell designs.
- B. It is very hard and time consuming to recast a design implemented in an FPGA to standard cells.
- C. FPGAs have lower non-recurring engineering (NRE) costs.
- D. FPGAs cost more per chip than the equivalent gate-count standard cell chip.
- E. All of these are false.

### Question 4

It is often valuable to simulate the chip's function in a high level language such as C or Matlab before starting the logic design because

- A. It saves time overall to establish that the algorithm to be implemented in the chip is fully debugged before starting hardware design.
- B. It is trivial to convert C to Verilog.
- C. It moves the job of hardware optimization to the high level "algorithm" people who know the most about the intrinsic parallelism of Hardware.
- D. The instructor in your University course on ASIC design said to do it this way and you always do exactly as he advises.
- E. It is not valuable.

### Question 5

A "hold violation" refers to which of the following:

- A. The clock frequency is so slow that logic transitions can race through successive logic stages (stage = register+logic) within one clock period.
- B. The logic is so fast that a transition can race through successive logic stages in the hold+skew time associated with a single clock edge.
- C. The logic is so slow that a transition can be delayed at the following register to the extent that it occurs during the hold time of a flip-flop.

- D. A penalty resulting in suspension under World Wrestling Federation Rules.
- E. None of the above.

### Questions 6, 7

These questions refer to the following logic diagram. The minimum and maximum delays between each set of successive gates are marked as #(min: typical: max) ns and are marked on the output node of the driving gate. You also need the following:

- $T_{\text{setup}} = \#(1: 2 : 3)$  ns.
- $T_{\text{hold}} = \#(1: 1.5 : 2)$  ns.
- $T_{\text{skew}} = \#(1: 1 : 1)$  ns for the clock.
- $T_{\text{clock-Q}} = \#(2 : 3 : 6)$  ns.
- $T_{\text{logic}} = \#(1 : 3 : 5)$  ns for each and every logic gate.

Consider the logic equivalent to the following code (implemented exactly as described):

```
reg [2:0] A;
wire [2:0] B;

always@(posedge clock)
    A <= B;

assign B = {(A[0] ^ A[1]) | A[2], A[0] & A[2], A[1]};
```

### Question 6

What is the fastest possible clock period?

- A. 4 ns
- B. 15 ns
- C. 20 ns
- D. 21 ns
- E. None of the above are correct.

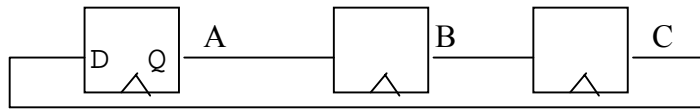
### Question 7

Is there a potential hold violation?

- A. Yes, the fastest logic is 2 ns too fast.
- B. Yes, the fastest logic is 1 ns too fast.
- C. No, there is a 1 ns safety margin.
- D. No, there is a 2 ns safety margin.
- E. None of the above are correct.

### Question 8

Which code fragment correctly captures the following logic.



- A. 

```
always@(posedge clock)
begin
    A <= B;
    B <= C;
    C <= A;
end
```
- B. 

```
always@(posedge clock)
begin
    B = A;
    C = B;
    A = C;
end
```
- C. 

```
always@(posedge clock)
begin
    C = B;
    B = A;
    A = C;
end
```
- D. 

```
always@(posedge clock)
begin
    A = C;
    B = A;
    C = B;
end
```
- E. None of the above

### Question 9

Which alternative best describes the behavior of the logic in the following verilog fragment.

```
wire [3:0] A, B;
assign B = A << 2;
```

- If A=4'b0111, then
- A. B=4'b1100
  - B. B=4'b0001
  - C. B=4'b1111
  - D. B=4'b1101;
  - E. None of the above

**Question 10**

Which alternative best describes the behavior of the logic in the following verilog fragment.

```
wire [4:0] A;  
wire [2:0] B;  
assign B = {&A[2:0]; {2{A[4] | A[3]}} };
```

If A =5'b10111, then

- A. B=3'b000;
- B. B=3'b011;
- C. B=3'b100;
- D. B=3'b111;
- E. None of the above.

**Question 11**

Consider the following code.

```
always@(posedge clock)  
    C <= A+B;
```

If it is synthesized with the following constraints

```
Create_clock -period 10 -waveform {0 5} clock  
set_clock_skew -uncertainty 1.0 clock  
  
set_input_delay 2.0 -clock clock all_inputs() - clock
```

then what is the maximum allowed delay for the logic A+B if the setup time for the flip-flops with output C is 1 ns?

- A. 1 ns
- B. 4 ns
- C. 6 ns
- D. 10 ns
- E. None of the above

**Question 12**

Which alternative best describes the behavior of the logic in the following verilog fragment.

```
reg [3:0] A, B, C;

always@(posedge clock)
begin
    B <= {A[1:0], A[3:2]};
    C <= A + B;
end
```

If A =4'b1101, and B=4'b0011 before the positive edge of the clock, then after the positive edge.

- A. B=4'b0011; C=4'b0001;
- B. B=4'b0111; C=5'b10001;
- C. B=4'b0111; C=4'b0001;
- D. B=4'b0111; C=4'b1001;
- E. None of the above.

**Question 13**

Which of the following is most correct?

- A. When HDLs were introduced, they lead to a quantum leap in design productivity;
- B. When synthesis was introduced, it lead to a quantum leap in design productivity;
- C. VHDL and Verilog are very different. Totally different logic design approaches are required for each language;
- D. The synthesizable subset of Verilog is so vast, that it takes a long time to learn it.
- E. All of the above are terribly wrong.