

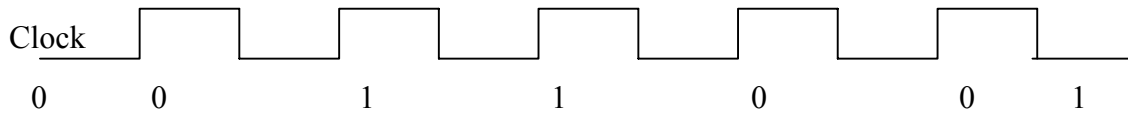


(b) Rewrite the answer using an XOR gate.

### Question 3

*The purpose of this question is to get you starting to design.*

Design a piece of synchronous logic that produces the following sequence:



ie. It inverts in value every two clock cycles. You can assume that the registers are initially set to any value you like.

### Question 4

*This is another exam-prep question.*

Which of the following statements is the most correct?

- A. Hardware Description Languages (HDLs) are useful because they allow design optimization (e.g. Karnaugh maps) to be done by textual manipulation.
- B. HDLs are useful because it is faster to write a description of a digital circuit than draw it.
- C. HDLs are useful because computer tools are available to optimize and generate a detailed design.
- D. Both B and C.
- E. None of the above is correct.

### Question 5

*Another exam prep question.*

What does the following code fregment represent ('&' represents an and gate)?

```
reg A, B, C;  
always@(posedge clock)  
    C <= A & B;
```

- A. A 2-input and gate with A, B, and clock as inputs and C as an output.
- B. A flip-flop whose input is connected to an AND gate fed by A and B.
- C. A comparison circuit. Is C less than or equal to A and B?
- D. A J-K flip-flop.
- E. None of the above.

**Question 6**

*Another exam prep question but this time, not multiple choice.*

Sketch the logic capture in the following code fragment.

```
reg A, B;
wire C;
always@(posedge clock)
    if (C) A <= B;
assign C = D & E;
always@(D or E or F)
    if (F) B = E;
    else B = F;
```

**Question 7**

*Again, exam preparation.*

Rewrite the code in Question 6 so that it is one procedural block and one assign statement.

**Question 8**

Draw the waveform specified in the following code fragment...

```
reg clock, A, B;
initial
    begin
        clock = 0;
        A = 0;
        B = 1;
        #11 A = 1;
        #10 B = 0; A = 0;
    end
always
    #5 clock = ~clock;
```