SOLUTION NOTES FOR EXAMINERS AND SUPERVISORS ONLY

ECAD 2000 Paper 4 Question 5 (SWM)

(a) In Verilog continuous assignment is effectively a wire connection.
For example:

```
  wire x1,x2,y,z;
  assign x1 = y && z;
  assign x2 = x1;
```

will result in x1 always being the combinational AND of y and z.
x2 had the identical value to x1.

Non-blocking assignment is register based assignment.
For example:

```
  reg x1,x2;
  wire y,z;
  always @(posedge clock) begin
    x1 <= y && z;
    x2 <= x1;
  end
```

will result in x1 being assigned the result of y AND z at the next clock edge. The state of x2 will be one clock cycle behind x1.

(b) In RandomBitsA:

```
  shift_reg[3]=shift_reg[2]=shift_reg[1]=shift_reg[0]=newbit
```

so

```
  newbit = ~(newbit ^ newbit ^ newbit)
         = ~newbit;
```

(to comply with the question a circuit diagram of this is required)

In RandomBitsB shift_reg really is a shift register made up of a D-latch for each bit in shift_reg.
(to comply with the question a circuit diagram of this is required)

(c) RandomBitsA is likely to oscillate asynchronously with respect to the clock.

To compute the output of RandomBitsB, let us assume that shift_reg starts in state 4'b0000:

current state	newbit	next state
0000	1	0001
0001	0	0010

```
        0010        0        0100
        0100        1        1001
        1001        1        0011
        0011        1        0111
        0111        1        1111
        1111        0        1110
        1110        1        1101
        1101        1        1011
        1011        0        0110
        0110        0        1100
        1100        0        1000
        1000        0        0000
other states:
        0101        0        1010
        1010        1        0101
```