VLSI System Design.

1) Algorithm 1: GCD Computation.

```
1.  a = read() ;  // 8-bit data
2.  b = read() ;  // 8-bit data
3.  while (a != b)
        if (a < b)
            b = b - a ;
        else
            a = a - b ;
        end if ;
    end while
4.  print (a).
```

Test bench i/P

data_in1 = 42

data_in2 = 16

Clk-period = 110ns. (50% duty cycle)

@ 10 ns ⇒ rst = 1

@ 30 ns ⇒ rst = 0

@ 65 ns ⇒ go = 1.

@ 185 ns ⇒ go = 0.

Plot the output waveform after simulation.

Mention the time in ns when final o/p is generated.

(2) Algorithm 2: Counting number of 1 in a 16-bit data input

```
1. a = read(); //16-bit data.

2. while (a != 0)
       ~~bit~~ msb = msb-read(a);
       if (msb == 1)
           c = c+1; [c is set to zero initially.]
       else
           c = c;

       ls(a); // left shift operation on a.

3. print(a);
```

Test Bench i/P.

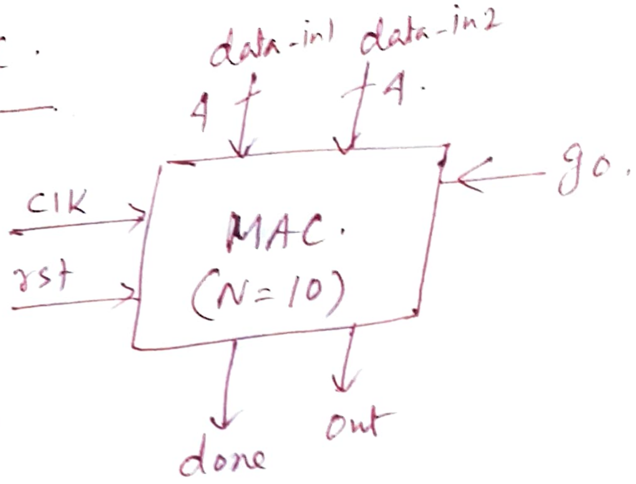
data-in = 1100110100110010

clk-period = 110 ns (50% duty cycle).

@10ns => rst = 1
@30ns => rst = 0
@65ns => go = 1
@175ns => go = 0.

Plot the o/p waveform after simulation and mention the time (in ns) on which the final o/p is produced.

(3) Algorithm 3 : programmable MAC.



1. S = 0 , N = 10 (Fixed parameter).
2. for i = 1 to 10 ;

   a = read() ; // data_in1

   b = read() ; // data_in2

   p = ~~a+b~~ a * b ;

   S = S + p ;
3. print (s) ;

Test Bench i/p

clk-period = 110 ns (50% duty cycle)

@ 10 ns ⇒ rst = 1 | @ 65 ns ⇒ go = 1

@ 30 ns ⇒ rst = 0 | @ 185 ns ⇒ go = 0

---

	data-in1	data-in2
@ 140 ns ⇒	2	3
@ 250 ns ⇒	3	4
@ 350 ns ⇒	4	5
@ 450 ns ⇒	5	6
@ 550 ns ⇒	6	7
@ 650 ns ⇒	7	8
@ 750 ns ⇒	8	9
@ 850 ns ⇒	1	1

Plot the o/p waveform & Show the final result.