Design a digital system that will compute the GCD of two numbers A and B.

- A, B are 32-bits binary number

You are not allowed to use the following operators (*,/, +). You are allowed to use the mod operator but it should be implemented as a separate entity.

Inputs:

1. Clk
2. Rst: 1→ asynchronous and set all registers to a value of 0.
3. A: A 32-bits binary number
4. B: A 32-bits binary number

Outputs:

- G: A 32-bits binary number representing the greatest common divisor.
- A_out: A 32-bits binary number the A for which the output is calculated
- B_out: A 32-bits binary number the B for which the output is calculated
- V: 1 bit that indicates if the output is valid

Calculating the GCD is done as follows

```
var R;
X = A;
Y = B;
while ((X % Y) > 0) {
  R = X % Y;
  X = Y;
  Y = R;
}
G = Y
```

Each of the iterations should be carried out in one cycle, in addition to one cycle to read the inputs.

You should submit a do file run.do to show the waveform for the test cases below

A	B	GCD
16#62	16#38	16#E

16#798F20	16#12B87CA0	16#20
16#FFFFFFFF	16#FFFFFFFF	16#FFFFFFFF
16#0	16#9	16#9