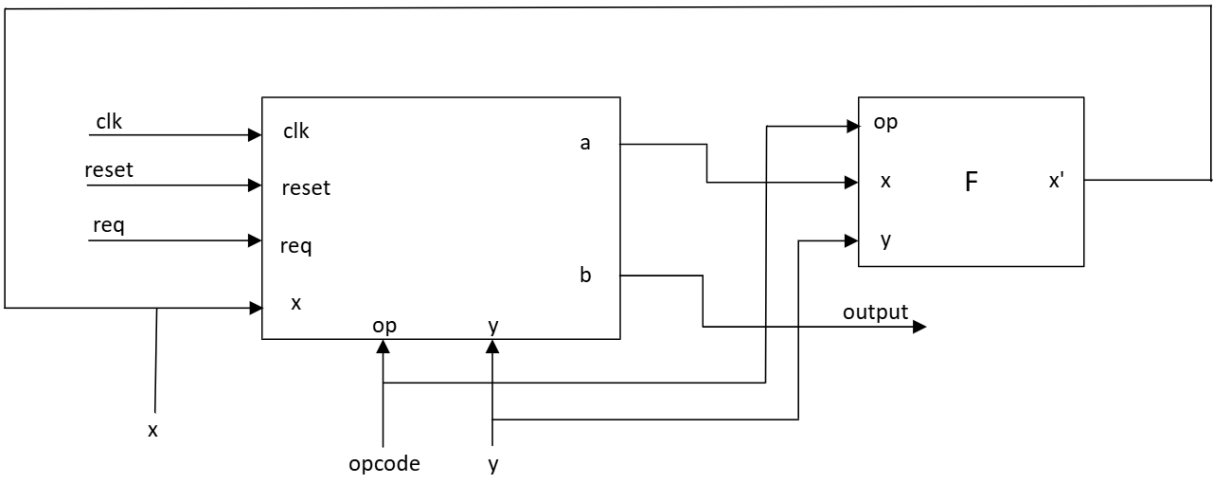
FSM in Verilog

Given the following device



Device protocol

1. Inputs:
   - clk: Clock signal.
   - reset: Resets signal.
   - req: Request signal to determine paths *a* or *b*.
   - x and y: Operands inputs.
   - op: Operation code.
2. Process:
   a. The device performs a computation (F) based on the opcode.
   b. It selects either a or b, depending on the req signal:
      - When req = 1, selected bus *a* (computation).
      - When req = 0, selected bus *b* (output).
   c. On a positive edge of clk, the computation (F) happens based on the current opcode.
   d. If reset != 1 device uses as *x* a result from previous computation.

3. Output:
   a. The result is shown when req = 0.
4. Reset:
   a. Reset signal sets all registers and inputs to 0.
5. Opcode encoding:
   - opcode = 00: AND (x & y).
   - opcode = 01: NAND ~(x & y).
   - opcode = 10: NOR ~(x | y).
   - opcode = 11: XOR (a ^ b).

Tasks

1. Draw the FSM diagram for this device.

   Hint. Answer the following questions:

   a. What states this device can be in?
   b. How does the transition between states happen (e.g. what signals change)?
2. Write a Verilog code.

Code Run

```
iverilog -o FSM_sim solution.v computation_device_tb.v
vvp FSM_sim
```