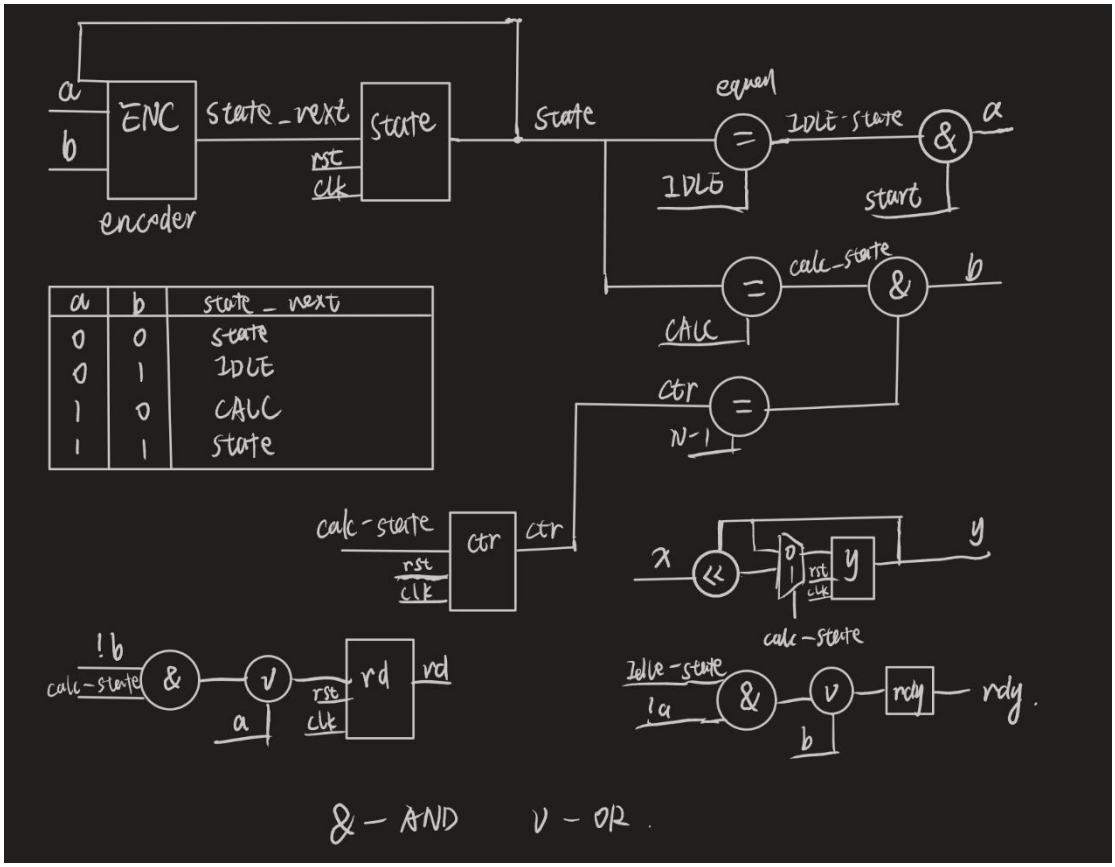
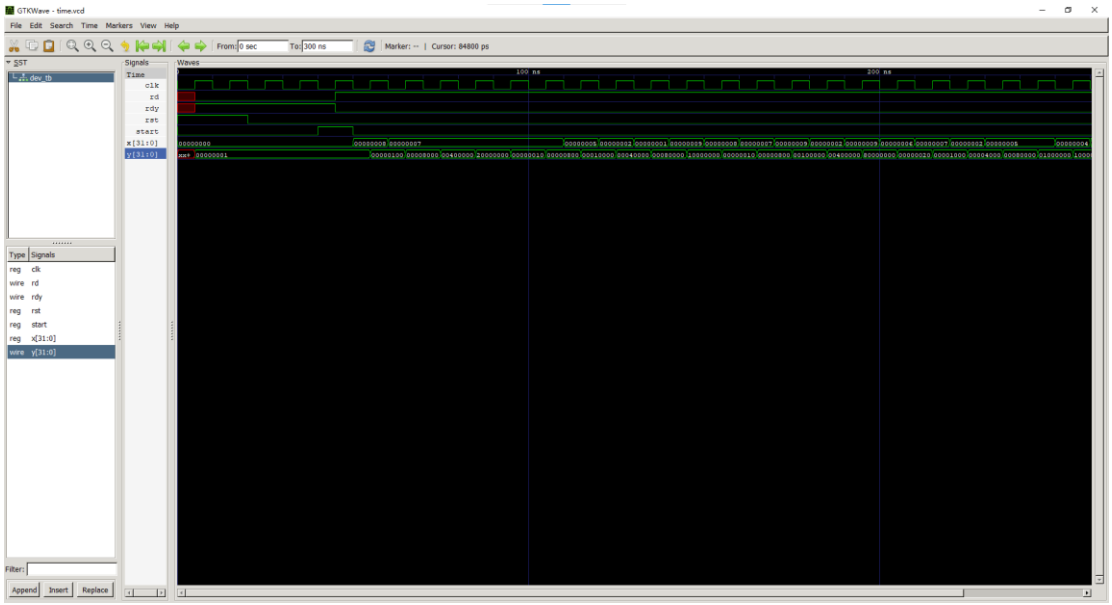
Functional electronic circuits Lab1

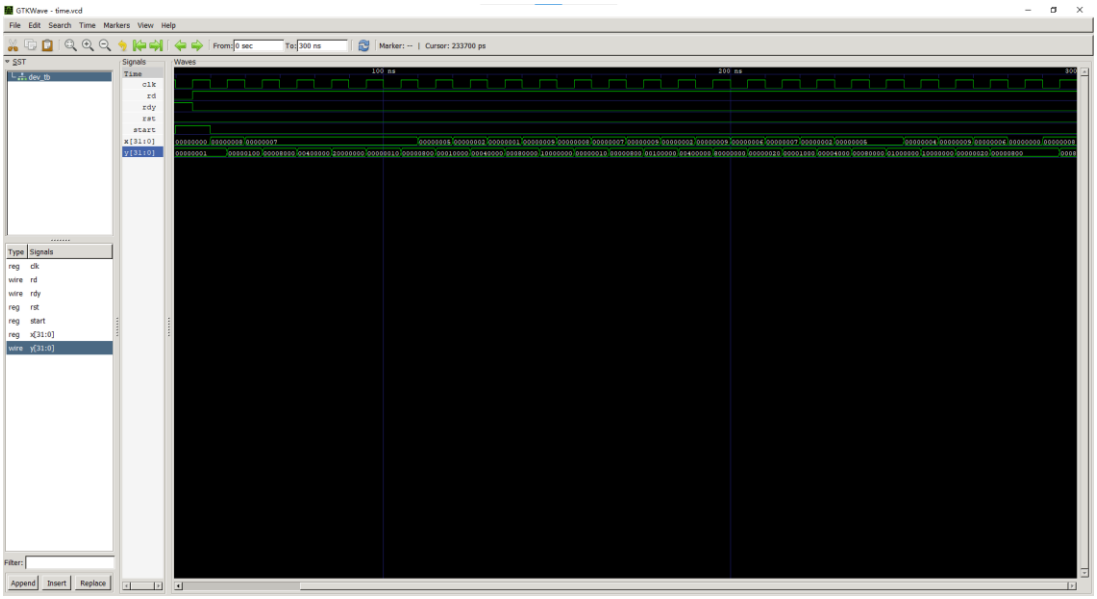
Student Name: CAO Xinyang
Student ID: 20321308

The picture with microarchitecture:



The timing diagram with simulation results:





Code of the testbench and the device:

dev.v

```
module dev(
    input [31:0] x,
    input start,
    input rst,
    input clk,
    output reg rd,
    output reg [31:0] y,
    output reg rdy
);

parameter N= 10;
parameter IDLE = 0;
parameter CALC = 1;


reg state;
reg[2:0] ctr;

always@(posedge clk)
    if(rst) begin
        state <= IDLE;
        ctr <= 0;
        rdy <= 1;
        rd <= 0;
```

```
        y <= 1;
    end else begin
        case(state)
            IDLE:
                if(start) begin
                    state <= CALC;
                    rdy <= 0;
                    rd <= 1;
                end
            CALC:
                begin
            ctr <= ctr + 1;
                y<=(y << x) | (y >> (32 - x));

         if(ctr == (N-1))begin
                  rd <= 0;
              state <= IDLE;
              rdy = 1;
         end
         end

            endcase
        end

endmodule
```

dev_tb.v

```
`timescale 1ns/1ps

module dev_tb;

reg [31:0] x;
reg start,rst,clk;
wire rd, rdy;
wire [31:0] y;

dev uut(
    .x(x),
    .rd(rd),
    .start(start),
    .rst(rst),
    .clk(clk),
```

```
    .y(y),
    .rdy(rdy)
);

always #5 clk = ~clk; // each 5 ns we will get opposite value of clk
sigr

always@(negedge clk)
    if(rd)
        x = ($random % 10) & 8'hFF;

initial begin
    $dumpfile("time.vcd");
    $dumpvars(1, dev_tb);
    clk = 0;
    rst = 1;
    start = 0;
    x = 0;
    #20
    rst = 0;
    #20
    start = 1;
    #10
    start = 0;
    #250
    $finish;
end

endmodule
```