

## Lab2

逻辑设计

有限状态机(fifo.v)

电路设计

fifo

register file

核心代码

register file

fifo

仿真结果

register file

distmen & blkmem

fifo

结果分析

实验总结

意见建议

思考题

# Lab2

---

## 逻辑设计

---

### 有限状态机(fifo.v)

```

1 always @(*) begin
2     if(IN & (count < 31)) next_state = PUSH;
3     // 2
4     else if(OUT & (count > 0)) next_state =
5     POP; // 3
6     else next_state = KEEP;
7     // 1
8     next_state = 4*(current_state%4==3) +
9     next_state;
10    // if c_state==POP then n_state+=4
11    // use (n_state >= 4) to confirm output
12    as it's 1T late
13 end

```

## 电路设计

### fifo

通过循环队列以存储, `Head`, `Tail` 表示头/尾

### register file

先对PUSH/POP信号取边沿

每次上升沿初始化为0; 如果要改变且不是寄存器0则改变对应的寄存器

## 核心代码

---

### register file

```

1 always @(posedge clk) begin
2     if(we & wa) begin
3         // wa: write address
4         // we: write enable
5         REGS[wa] = wd;
6     end
7 end

```

## fifo

```

1 assign count = (TAIL - HEAD) % 32;
2 // 由于count是无符号数，因此TAIL<HEAD没问题
3 always @(*) begin
4     dout = 0;
5     {we, en} = 2'b00;
6     if(rst) begin
7         {HEAD, TAIL, addr, en, we} = 17'b0;
8     end
9     else begin
10         case (current_state%4)
11             KEEP: {en, we} = 2'b0;
12             PUSH: begin
13                 addr = TAIL;
14                 TAIL = (TAIL+1) % 32;
15                 we = 1;
16             end
17             POP: begin
18                 addr = HEAD;
19                 we = 0;
20                 HEAD = (HEAD+1) % 32;
21             end
22             default: ;

```

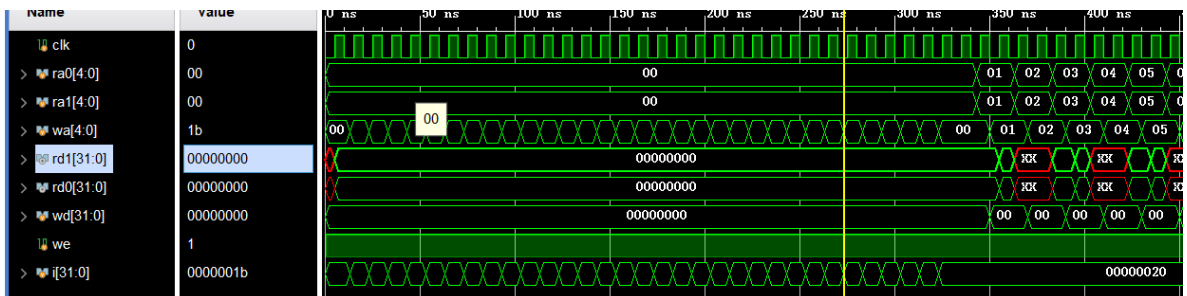
```

23         endcase
24         if(current_state>3) dout = DOUT;
25         // 这里表示如果上一次是POP，那么这次输出
dout(因为延迟一个周期)
26         // 因此c_state%3表示本次运行状态，
c_state>=4表示是否输出
27         // 不过在实验检查也说到这样会在dout输出开
始有个小毛刺
28         // 小毛刺是DOUT(不是输出的dout,这个接在
RAM端口)在输出前一瞬间的值
29     end
30 end

```

## 仿真结果

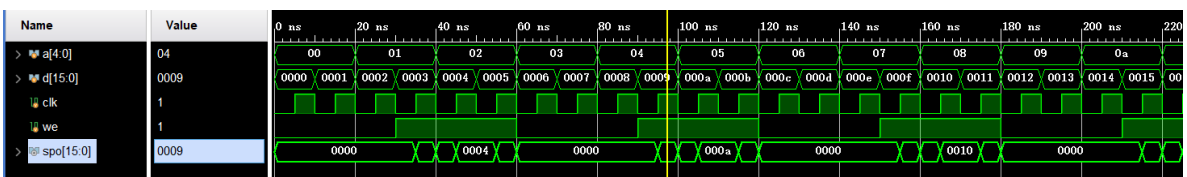
## register file



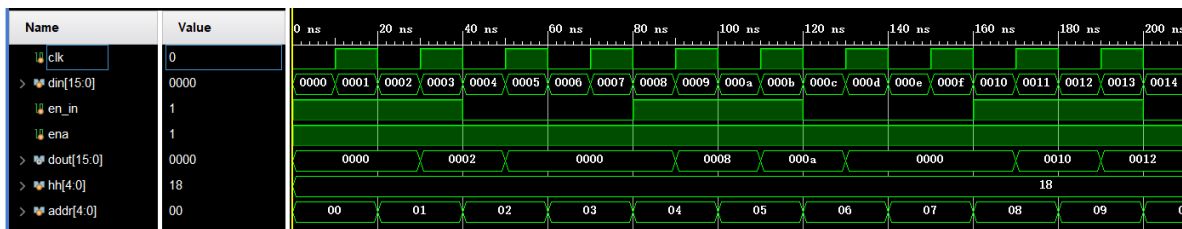
## 高阻态是因为当时正在执行写操作

## distmen & blkmem

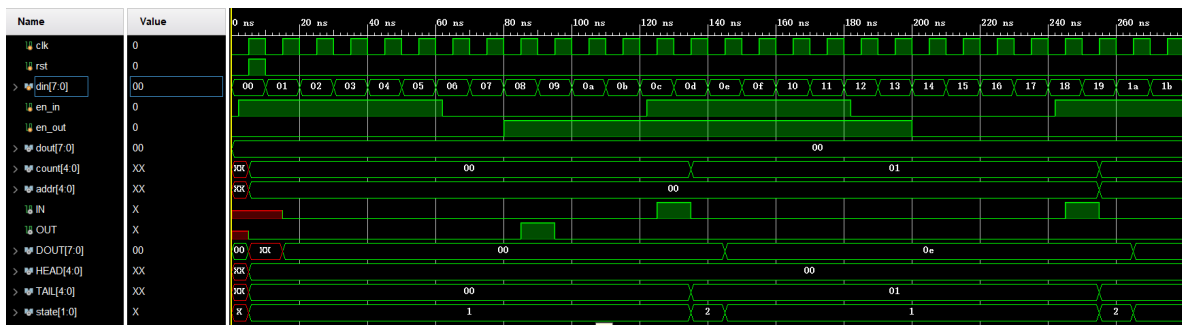
dist:



blkmem:



## fifo



仿真时引出了一些"内部变量"作为端口

## 结果分析

- ALU可以正常存取数据
- blkmem内置了addr寄存器因此读写操作会慢一个周期
- fifo能正常PUSH/POP

## 实验总结

1. 初始化的操作可放进仿真文件
2. blk读写会落后一个周期
3. 一些情况下会出现毛刺

## 意见建议

关于fifo, 我觉得用distmem可能更为恰当

# 思考题

---

连上接口, 直接按照冒泡排序for循环即可