

实验四 多周期CPU

zjx@ustc.edu.cn

2020.5.13

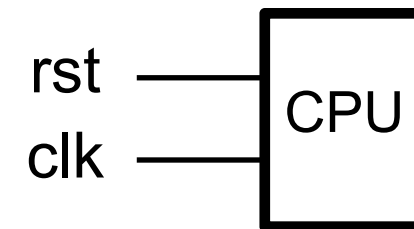
实验目标

- 理解计算机硬件的基本组成、结构和工作原理
- 掌握数字系统的设计和调试方法
- 熟练掌握数据通路和控制器的设计和描述方法

实验内容

1. 设计实现多周期CPU，可执行如下6条指令：

- add, addi
- lw, sw
- beq, j

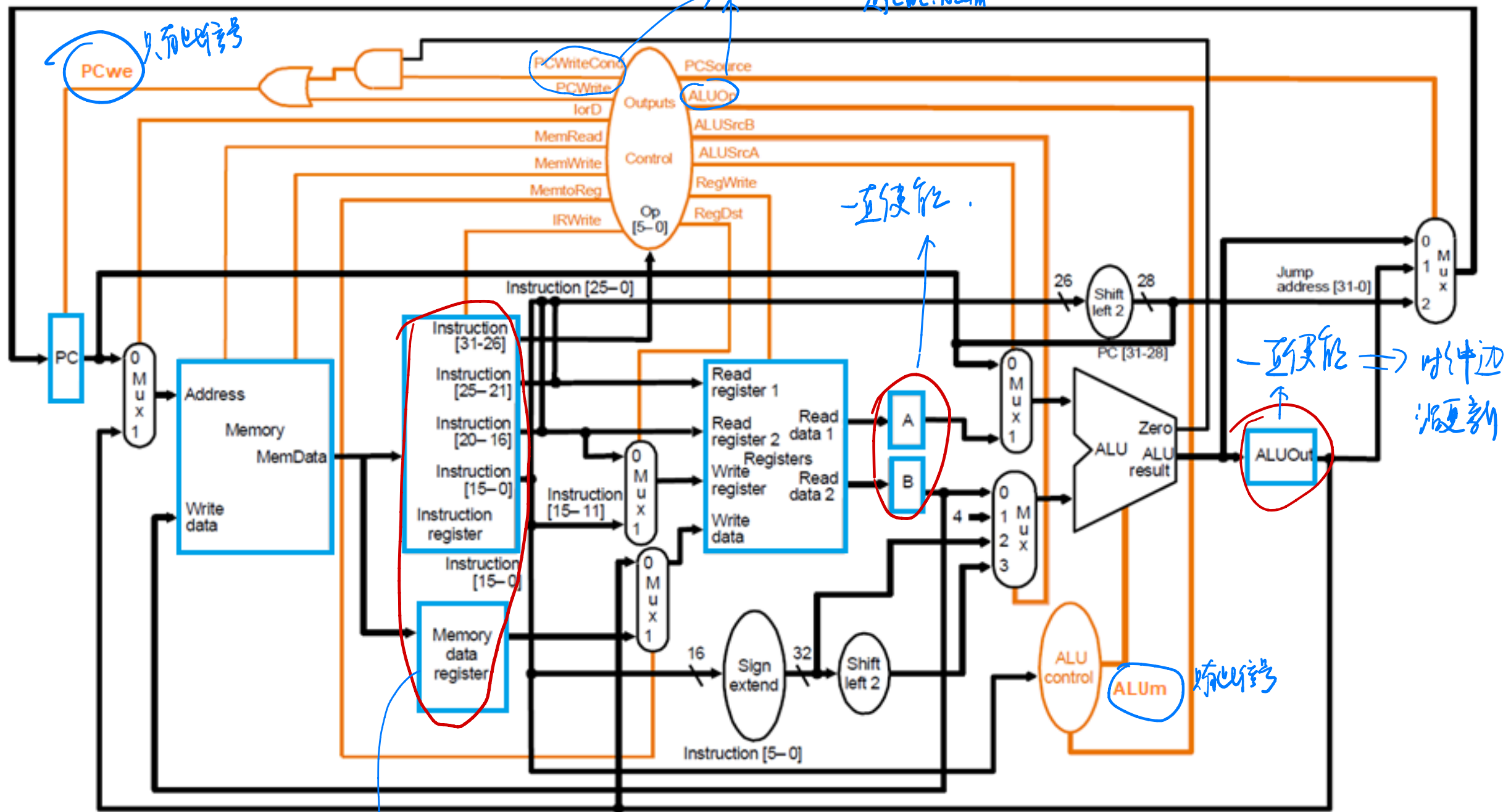


- 分析指令功能，设计数据通路和控制器
- ALU和寄存器堆：实验1和实验2设计的模块
- 指令和数据存储共用一个RAM存储器，采用IP例化实现，容量为512 x 32位的分布式存储器

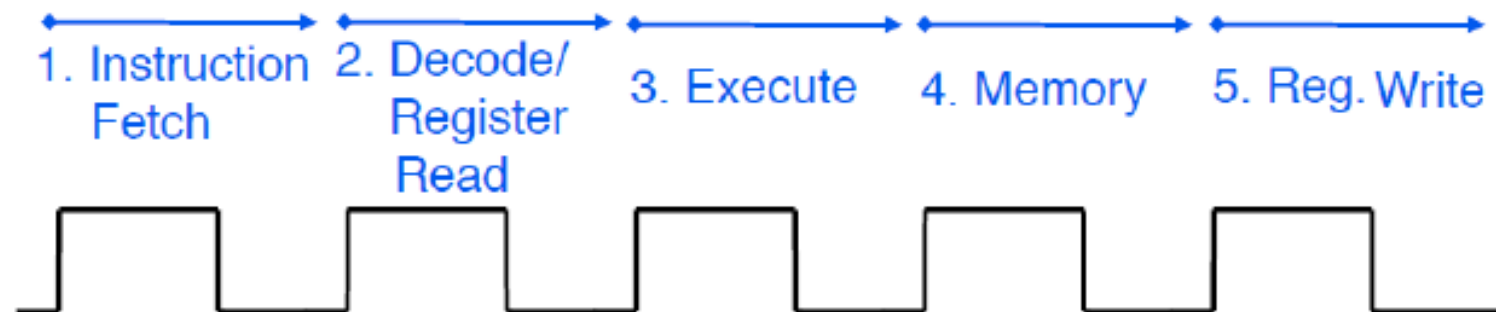
⇒ 控制信号在不同阶段产生 (状态机)

数据通路

分段, 加 key, 实现不同时期实现一条指令
(蓝色为不清晰元, 其余均为组合逻辑)

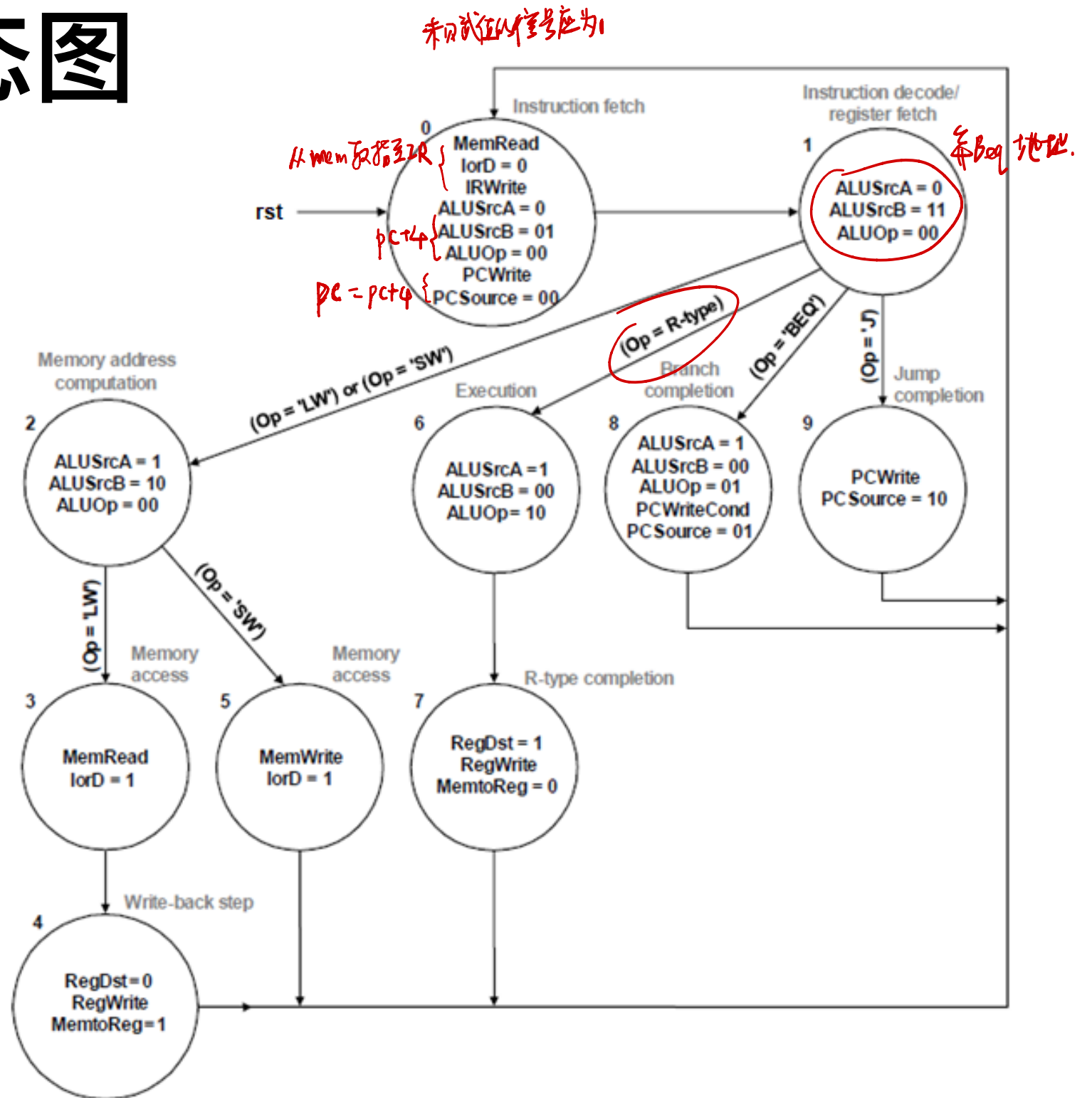


多周期操作

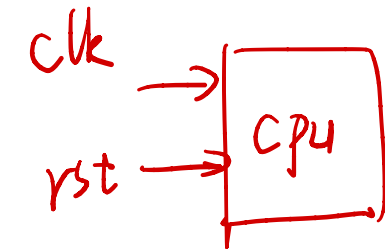


Step	R-Type	lw/sw	beq/bne	j
IF	$IR = Mem[PC]$ $PC = PC + 4$			
ID	$A = Reg[IR[25-21]]$ $B = Reg[IR[20-16]]$ $ALUOut = PC + (SE(IR[15-0]) \ll 2)$			
EX	$ALUOut = A \text{ op } B$	$ALUOut = A + SE(IR[15-0])$	If $(A == B)$ then $PC = ALUOut$	$PC = PC[31-28] \parallel (IR[25-0] \ll 2)$
MEM	$Reg[IR[15-11]] = ALUOut$	$MDR = Mem[ALUOut]$ $Mem[ALUOut] = B$		
WB		$Reg[IR[20-16]] = MDR$		

控制器状态图

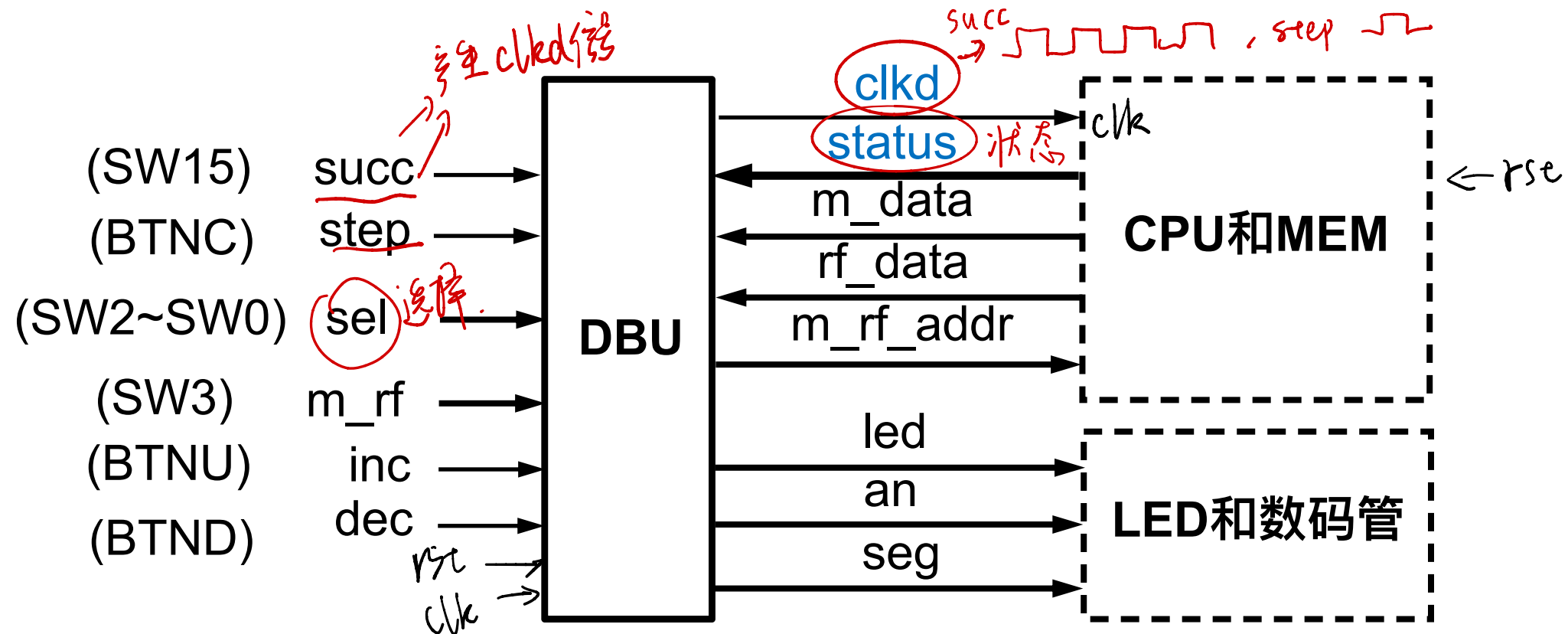


实验内容 (续1)



2. DBU: Debug Unit, 调试单元

- 下载测试时, 用于控制运行方式, 显示运行状态和运行结果



【图中省略了 *clk* (*clk* 100mhz 降频) 和 *rst* (BTNL) 信号】

实验内容 (续2)

- **控制CPU运行方式**

- $\text{succ} = 1$: clkd 输出连续的周期性脉冲信号 (可以作为CPU的时钟信号), 控制CPU连续执行指令
- $\text{succ} = 0$: 每按动step一次, clkd 输出一个脉冲信号, 控制CPU执行一个时钟周期

- **$\text{sel} = 0$: 查看CPU运行结果 (存储器或者寄存器堆内容)**

- m_rf : 1, 查看存储器(MEM); 0, 查看寄存器堆(RF)
- m_rf_addr : MEM/RF的调试读口地址(字地址), 复位时为零
- inc/dec : m_rf_addr 加1或减1
- rf_data/m_data : 从RF/MEM读取的数据字
- 16个LED指示灯显示 m_rf_addr
- 8个数码管显示 rf_data/m_data

实验内容（续3）

- **sel = 1 ~ 7：查看CPU运行状态（status）**
 - 16个LED指示灯(SW15~SW0)依次显示控制器的控制信号
PCSource(2)、PCwe、IorD、MemWrite、IRWrite、RegDst、
MemtoReg、RegWrite、ALUm(3)、ALUSrcA、ALUSrcB(2) 和
ALUZero
 - 8个数码管显示由sel选择的一个32位数据
 - sel = 1：PC，程序计数器
 - sel = 2：IR，指令寄存器
 - sel = 3：MD，存储器读出数据寄存器
 - sel = 4：A，寄存器堆读出寄存器A
 - sel = 5：B，寄存器堆读出寄存器B
 - sel = 6：ALUOut，ALU运算结果寄存器
 - sel = 7：

实验步骤

1. 结构化描述多周期CPU的数据通路和控制器，并进行功能仿真
2. 将CPU和DBU下载至FPGA中测试

实验检查

- 检查多周期CPU的功能仿真
- 检查CPU和DBU下载至FPGA后的运行功能

思考题

- **修改数据通路和控制器，扩展对其他MIPS指令的支持，并进行功能仿真和下载测试**

The End