

指令	周期	功能	控制信号(默认全为0)
FETCH(取址周期)		从内存取出指令送到IR; 当前PC+4.	memread <= 1; irwrite <= 1; alusrcb <= 3'b001; pcwrite <= 1;
DECODE(译码周期)		根据指令op段决定下一状态; 提前计算offset+pc.	alusrcb <= 3'b011;
lw	MEMADR	基址寻址, 送地址到内存.	alusrca <= 2'b01; alusrcb <= 3'b010;
	LWRD	读内存, 送数据到寄存器.	memread <= 1; iord <= 1;
	LWWR	写寄存器.	regwrite <= 1; memtoreg <= 1;
sw	MEMADR	同lw中的MEMADR.	同上
	SWWR	写内存.	memwrite <= 1; iord <= 1;
add and sub or	RTYPEEX	根据funct对寄存器rs和rt进行相应运算.	alusrca <= 2'b01; aluop <= 2'b10;
	RTYPEWR	将运算结果写到寄存器rd.	regdst <= 2'b01; regwrite <= 1;
sll srl	SHAMT	寄存器rt左(右)移shamt位.	alusrca <= 2'b10; aluop <= 2'b10;
	RTYPEWR	同add中的RTPYEWR.	同上
addi	ADDIEX	寄存器rs和立即数imme进行加法运算.	alusrca <= 2'b01; alusrcb <= 3'b010; aluop <= 2'b00;
	ADDIWR	运算结果存至寄存器rt.	regwrite <= 1;
andi	ANDIEX	寄存器rs和立即数	alusrca <= 2'b01;

		imme进行与运算.	alusrcb <= 3'b100; aluop <= 2'b11;
	ANDIWR	运算结果存至寄存器rt.	regwrite <= 1;
beq	BEQEX	寄存器rs和rt相减, 如果相等则跳过offset条指令, 否则顺序执行.	alusrca <= 2'b01; aluop <= 2'b01; pcwritecond <= 1; pcsource <= 2'b01;
bne	BNEEX	寄存器rs和rt相减, 如果不相等则跳过offset条指令, 否则顺序执行.	alusrca <= 2'b01; aluop <= 2'b01; pcwritecond_notzero <= 1; pcsource <= 2'b01;
jal	JALCAL	计算PC+4.	alusrcb <= 3'b001;
	JALEX	PC+4结果存至寄存器\$31; PC跳转到target.	regdst <= 2'b10; regwrite <= 1; pcwrite <= 1; pcsource <= 2'b10;
jr	JR	将寄存器\$31的结果送至PC.	pcwrite <= 1; pcsource <= 2'b11;
j	JEX	将指令中的target字段左移两位后送至PC.	pcwrite <= 1; pcsource <= 2'b10;