|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | |  |  |
|  |  |  |  |  | |  |  |
| Addi | 001000 | Rs | | Rt | | | Imm16 |
| Andi | 001100 | Rs | | Rt | | | Imm16 |
| Xori | 001110 | Rs | | Rt | | | Imm16 |
| Bne | 000101 | Rs | Rt | Offset16 | | | |
| Beq | 000100 | Rs | Rt | Offset16 | | | |
| Lw | 010111 | Rs | Rt | Offset16 | | | |
| Sw | 101011 | Rs | Rt | Offset16 | | | |
| Add | 000000 | Rs | Rt | Rd(rs+rt) | 00000 | | 010100 |
| Or | 000000 | Rs | Rt | Rd(rs|rt) | | 00000 | 011001 |
| J | 000010 | Target26 | | | | | |
| Jal | 000011 | Target26 | | | | | |
| Jr | 000000 | Rs | 000000000000000(15) | | | | 001000 |
| Sll | 000000 | 00000 | Rt | Rd | | Shamt5 | 000000 |
| Srl | 000000 | 00000 | Rt | Rd | | Shamt5 | 000010 |

Imm: immediate

信号表：

|  |  |  |
| --- | --- | --- |
| ALUCon | 000 | 加法 |
|  | 001 | 左移 |
|  | 010 | 右移 |
|  | 011 | 异或 |
|  | 100 | 与 |
|  | 101 | 或 |
|  | 110 | 减 |
| PCsrc | 00 | pc来自A寄存器（对应jr指令） |
|  | 01 | pc来自pc-extend（对应无条件转移指令j和jar 和 jr） |
|  | 10 | pc直接来自运算器结果（对应pc = pc + 1） |
|  | 11 | pc来自ALUOut寄存器（对应条件转移指令beq和bne） |
| PCWC1 | 0 | / |
|  | 1 | 对应相等则转移（beq） |
| PCWC2 | 0 | / |
|  | 1 | 对应不相等则转移（bne） |
| PCW | 0 | / |
|  | 1 | 对应无条件转移(j jal jr pc+1) |
| Regsrc | 00 | Wd来自MDR（对应读内存指令lw） |
|  | 01 | Wd来自pc（对应jal指令） |
|  | 10 | Wd来自ALUOut（对应所有计算性结果） |
| ALUsrca | 00 | A操作数来自pc （对应beq bne pc + 1 ） |
|  | 01 | A操作数来自A寄存器 (branch也用) |
|  | 10 | A操作数来自IR[10:6]（对应移位指令） |
| ALUsrcb | 00 | B操作数来自B寄存器（branch也用） |
|  | 01 | B操作数来自sign-extend（对应bne beq lw sw \*\*\*i） |
|  | 10 | B操作数是1（pc = pc + 1） |
| RegWrite | 0 | / |
|  | 1 | Regfile可以写 |
| RegDst | 00 | Wa来自IR[20:16] |
|  | 01 | Wa来自IR[15:11] (对应and or sll srl) |
|  | 10 | Wa为31（对应jal存pc当前值到$ra） |
| MEMW | 0 | / |
|  | 1 | Memory可以写 |
| MEMR | 0 | / |
|  | 1 | Memory可以读 |
| IorD | 0 | 访存地址来自PC（对应取指周期） |
|  | 1 | 访存地址来自ALUOut（对应sw） |
| IRWrite | 0 | / |
|  | 1 | IR可以写（对应取指周期） |