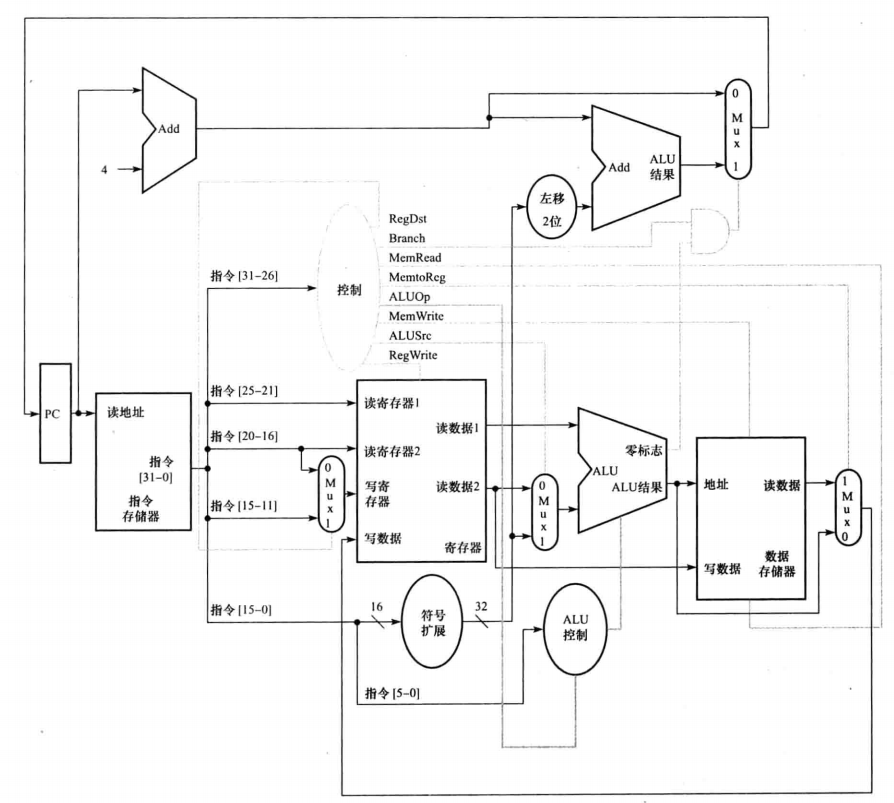
设计者学号： 21231511

设计者姓名：欧祖宇

1. **指定的指令如下**

|  |  |  |  |
| --- | --- | --- | --- |
| 指令 | 操作码 | 类型 | 含义 |
| ori | 001101 | I |  |
| lui | 001111 | I |  |
| addu | 000000  Func:  100001 | R |  |
| sub | 000000  Func:  100010 | R | 假定不产生溢出 |
| bne | 000101 | I |  |
| lw | 100011 | I |  |
| sw | 101011 | I |  |

1. **带控制信号的数据通路（用书上的即可）**



1. **ALU设计（需要设计者完成）**

为了设计ALU的控制信号，首先根据指令功能进行归纳总结，完成下表。

|  |  |  |
| --- | --- | --- |
| 指令 | 操 作 | ALU功能 |
| ori | op1和op2 按位或 | 或运算 |
| lui | op2 左移16位 | 左移运算 |
| addu | op1 和 op2 相加 | 加运算 |
| Sub | op1 和 op2 相减 | 减运算 |
| bne | op1 和 op2 判断相等 | 减运算后为0 zero输出1 |
| lw | op1 和 op2 相加 | 加运算 |
| sw | op1 和 op2 相加 | 加运算 |
| addiu | op1 和 op2 相加 | 加运算 |
| sll | Op2左移sa | 左移sa位 |
| Slt | op1 和 op2 判断大小 | 减运算输出sign符号位 |
| Beq | op1 和 op2 判断相等 | 减运算 |
| j | 无操作 | 无操作 |

# ALU的控制信号设计

从上面表来看，一共 4种操作，使用 二进制编码，在下表中给出定义。

|  |  |
| --- | --- |
| 操作 | 编码 |
| 加运算 | 000 |
| 减运算 | 001 |
| 或运算 | 010 |
| 左移运算 | 011 |
| 左移sa位 | 111 |

1. **控制单元分成两个部分，一个是cu，一个是alucontrol。**
2. 控制信号定义（仿照教材，所以直接给出）

|  |  |  |
| --- | --- | --- |
| 信号名 | “0”时定义 | “1”时定义 |
| RegDst | 写入寄存器来自rt | 写入寄存器来自rd |
| RegWrite | 无 | 往寄存器里写数据 |
| ALUSrc | 第二个操作数来自寄存器 | 第二个操作数是立即数扩展 |
| PCSrc | PC <-- PC+4 | PC <-- 分支地址 |
| MemRead | 无 | 读存储器 |
| MemWrite | 无 | 写存储器 |
| MemtoReg | ALU输出作为结果寄存器输入 | 存储器输出作为结果寄存器输入 |

1. 针对指令的cu输入/输出真值表

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 指令 | 操作码 | 类型 | RegDst | RegWrite | ALUSrc | PCSrc | MemRead | MemWrite | MemtoReg | ALUop |
| addu | 000000 | R | 1 | 1 | 0 | 00 | 0 | 0 | 1 | 0000 |
| sub | 000000 | R | 1 | 1 | 0 | 00 | 0 | 0 | 1 | 0000 |
| ori | 001101 | I | 0 | 1 | 1 | 00 | 0 | 0 | 1 | 0110 |
| lui | 001111 | I | 0 | 1 | 1 | 00 | 0 | 0 | 1 | 0111 |
| bne | 000101 | I | 0 | 0 | 0 | 01 | 0 | 0 | 0 | 0010 |
| lw | 100011 | I | 0 | 1 | 1 | 00 | 1 | 0 | 0 | 0001 |
| sw | 101011 | I | 0 | 0 | 1 | 00 | 0 | 1 | 0 | 0001 |
| Addiu | 001001 | I | 0 | 1 | 1 | 00 | 0 | 0 | 1 | 0001 |
| sll | 000000 | R | 1 | 1 | 0 | 00 | 0 | 0 | 1 | 0000 |
| slt | 000000 | R | 1 | 1 | 0 | 00 | 0 | 0 | 1 | 0000 |
| beq | 000100 | I | 0 | 0 | 0 | 10 | 0 | 0 | 0 | 0010 |
| j | 000010 | J | 0 | 0 | 0 | 11 | 0 | 0 | 0 | 0000 |

1. 针对指令的aluctrl输入/输出真值表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 指令 | 操作码 | 类型 | ALUop | Function字段 | ALU操作 | ALU控制码 |
| addu | 000000 | R | 0000 | 100001 | 加运算 | 000 |
| sub | 000000 | R | 0000 | 100010 | 减运算 | 001 |
| ori | 001101 | I | 0110 | \*\*\*\*\*\* | 或运算 | 010 |
| lui | 001111 | I | 0111 | \*\*\*\*\*\* | 左移运算 | 011 |
| bne | 000101 | I | 0010 | \*\*\*\*\*\* | 减运算 | 001 |
| lw | 100011 | I | 0001 | \*\*\*\*\*\* | 加运算 | 000 |
| sw | 101011 | I | 0001 | \*\*\*\*\*\* | 加运算 | 000 |
| Addiu | 001001 | I | 0001 | \*\*\*\*\*\* | 加运算 | 000 |
| Sll | 000000 | R | 0000 | 000000 | 左移sa位 | 111 |
| Slt | 000000 | R | 0000 | 101010 | 减运算 | 001 |
| Beq | 000100 | I | 0010 | \*\*\*\*\*\* | 减运算 | 001 |
| j | 000010 | J | 0000 | \*\*\*\*\*\* | \* | 000 |