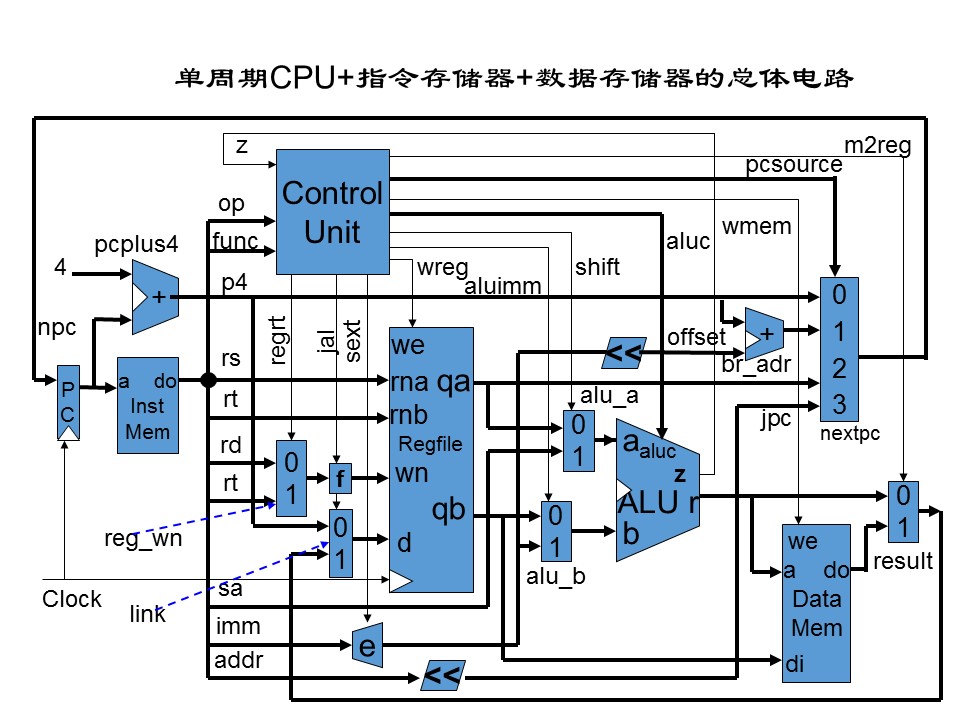
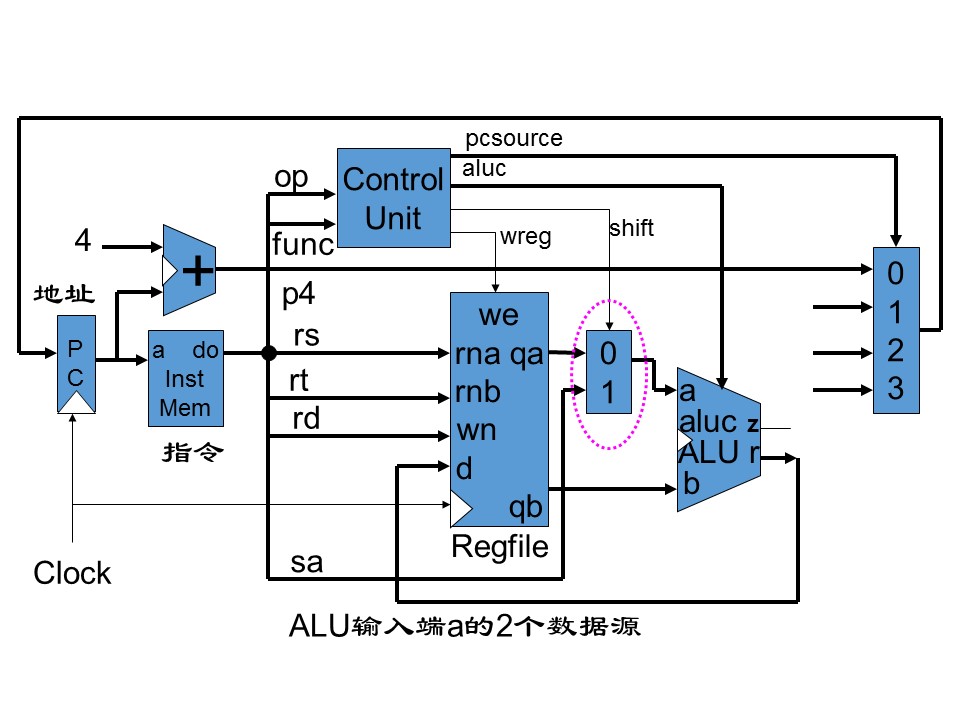
|  |  |  |
| --- | --- | --- |
| 控制信号 | 功能 | 信号值作用 |
| wreg | 写寄存器 | 1—写寄存器，0—不写 |
| regrt | 目的寄存器号是rt | 1—选择rt，0—选择rd |
| jal | 子程序调用 | 1—指令为jal，0—不是 |
| m2reg | 存储器数据写入寄存器 | 1—选择存储器数据，0—选择ALU结果。 |
| shift | ALU a使用移位位数 | 1—使用移位位数，0—使用寄存器数据 |
| aluimm | ALU b使用立即数 | 1—使用立即数，0—使用寄存器数据 |
| sext | ALU操作控制 | 1—符号扩展，0—0扩展 |
| aluc[3:0] | 写存储器 |  |
| wmem | 写存储器 | 1—写存储器，0—不写 |
| pcsource[1:0] | 下一条指令地址选择 | 00—pc+4; 01—选转移地址  10—选寄存器内的地址; 11 —选跳转地址 |

该电路必须能够执行列出的20条指令



**ALU的a输入端**

* 控制信号shift
* 以下面2条寄存器操作指令为例说明。
  + add rd,rs,rt ;rd<--rs+rt
  + sll rd,rt,sa ;rd<--rt<<sa
* 从上面的注释中可以看出，rd,rt是相同的，而rs,sa是不同的。
* 根据前面设计的ALU32可知，移位运算时，sa是从a输入端输入，寄存器算术逻辑时a输入端输入1个数，故a输入端有2个数据源。而rt作为ALU的b输入端，两者皆可用。
* 控制信号shift
* 以下面2条寄存器操作指令为例说明。
  + add rd,rs,rt ;rd<--rs+rt
  + sll rd,rt,sa ;rd<--rt<<sa
* 从上面的注释中可以看出，rd,rt是相同的，而rs,sa是不同的。
* 设计的ALU32移位运算时，sa是从a输入端输入，寄存器算术逻辑时a输入端输入1个数，故a输入端有2个数据源。而rt作为ALU的b输入端，两者皆可用。



|  |  |
| --- | --- |
| shift | 条件 |
| 0 | add|| sub ||and||or||xor |
| 1 | sll || srl || sra |