模块定义

1. NPC
2. 基本描述

Npc的主要作用是根据npc\_op和其他的信号来计算下一条指令的地址，同时，对于跳转类指令，计算pc+4

(2)模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| Zero[31:0] | I | alu计算出的数据，做各位或判断是否跳转 |
| Npc\_op[3:0] | I | 判断当前的npc执行那条指令  `define PC\_J 4'b0001  `define PC\_JAL 4'b0010  `define PC\_BEQ 4'b0011  `define PC\_JR 4'b0100  `define PC\_BNE 4'b0101  `define PC\_BLEZ 4'b0110  `define PC\_BGTZ 4'b0111  `define PC\_BGEZ 4'b1000  `define PC\_BLTZ 4'b1001  `define PC\_JALR 4'b1010 |
| Imm26[15:0] | I | 指令传输的26位立即数，当需要跳转时使用 |
| Pc[31:2] | I | 当前的pc值 |
| Pc\_reg[31:0] | I | 寄存器传来的值，jr指令用到 |
| Npc[31:2] | O | 输出下一条指令的地址 |
| Pc­­\_4[31:2] | O | Pc+4的值，当jal指令有用 |

(3)功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | 计算下一条指令 | 计算下一条指令 |
| 2 | 输出pc+4 | Jal指令需要存储pc+4， |

1. ALU
2. 基本描述

运算单元，可以进行addu，ori，subu，slt四种运算，根据输入的op来对num1，num2进行相应的处理并输出。同时，alu可以通过减法运算判断beq两个数值是否相等，从而判断是否应该跳转。

1. 模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| Num1[31:0] | I | 需要计算的数据，来自busA |
| Num2[31:0] | I | 需要计算的数据，来自busB或者sign\_imm16 |
| Sa | I | 进行移位操作是用到 |
| Alu\_op[2:0] | I | 决定alu的行为  000：addu  001：subu  010：ori  011:slt |
| Out[31:0] | O | Alu的计算结果 |

1. 功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | 无符号加法 | 计算无符号加法 |
| 2 | 无符号减法 | 无符号减法  同时，计算beq指令是否跳转 |
| 3 | 或立即数 | 计算或立即数的结果 |
| 4 | 小于置1 | Slt指令，num1<num2时,输出1 |

1. GPR
2. 基本描述

寄存器堆，由32个寄存器构成，能够存储数据，读取数据，写入数据等功能，

(2)模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| Busw[31:0] | I | 需要写入的数据，可能来自alu或者dm |
| Rw[4:0] | I | 要写入数据的地址 |
| Regwr | I | 写使能信号  1：寄存器可以写入  0：寄存器只读状态 |
| Clk | I | 时钟信号 |
| Reset | I | 复位信号  1：复位  0：不复位 |
| Ra[4:0] | I | 要输出的数据的地址，来自rs。 |
| BUSA[31:0] | O | 输出的数据，由rs决定 |
| Rb[4:0] | I | 要输出的数据的地址，来自rt |
| BUSB[31:0] | O | 输出的数据，由rb决定 |

(3)功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | 写入数据 | 将计算的结果或者从内存中读取的值存入相应寄存器 |
| 2 | 读取数据 | 根据地址，输出相应寄存器存储的值 |
| 3 | 复位 | 将所有寄存器赋值为0x0000\_0000 |

1. EXT
2. 基本描述

符号扩展，分别为无符号扩展(0扩展)，有符号扩展(根据最高位对数据扩展)，lui扩展(加载16位数据到高位)。符号扩展是为了使所有的数据都变成需要的位数，方便alu的计算。

(2)模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| Imm[15:0] | I | 需要扩展的16位数据 |
| Ext\_op[1:0] | I | 扩展的方式;  00：0扩展  01：有符号扩展  10：lui扩展 |
| Extout[31:0] | O | 扩展后的32位数 |

(3)功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | 无符号扩展 | 在16位数之前全部填成0 |
| 2 | 符号减法 | 根据16位数的最高位来扩展 |
| 3 | Lui扩展 | 将16位数放在32位数的高位 |

5.CONTROL

(1)基本描述

控制单元，通过控制各个多路选择器和输入信号，来控制数据的流向

(2)模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| Instr[31:0] | I | Mips指令 |
| Rw\_sel | O | 控制写入寄存器的值所在的地址  00：st  01：rd  10：31号 |
| Alu\_sel | O | 控制alu第二个数据的来源  1：ext  0：busb |
| busw\_sel | O | 控制写会寄存器的值来自于alu还是dm还是pc+4  00：alu  01：dm  10：pc+4 |
| Reg\_wr | O | 控制寄存器文件的写使能信号  0：不能写入  1：可以写入 |
| Dm\_we | O | 控制dm文件的写使能信号  0：不能写入  1：可以写入 |
| npc\_sel | O | 判断是否是beq信号  1：是  0：不是 |
| Ext\_op[1:0] | O | 控制ext原件的行为  00：0扩展  01：有符号扩展  10：lui扩展 |
| ALU op | O | 控制alu的行为  `define ALU\_ADD 3'b000  `define ALU\_SUB 3'b001  `define ALU\_ORI 3'b010  `define ALU\_SLT 3'b011 |
| Load\_op | O | 出现load指令时，对load的数据处理的信号 |
| Store\_op | O | 出现store类指令时，对store的数据进行处理的控制 |
| mult\_op; | O | 乘除法器行为控制 |
| intreq | I | 中断发生信号 |
| cp0\_we | O | Cp0写使能 |
| exlset, | O | 开启全局中断 |
| exlclr | O | 关闭全局中断 |
| epc\_we | O | epc写使能 |

(3)功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | 分析指令 | 判断出本次执行到底是哪一条mips指令 |
| 2 | 控制 | 根据指令来输出信号，控制整个数据通路 |

1. DM

(1)基本描述

数据存储器，lw和sw两个指令用到,用于存放或者读取数据。

(2)模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| addr[31:0] | I | 输入的数据所要存放的地址或者读取的数据存放的地址 |
| din[31:0] | I | 要写入的数据 |
| Clk | I | 时钟信号 |
| we | I | Dm的写使能信号  0：无法写入  1：可以写入 |
| Dout[31:0] | O | Sw读取的的数据 |
| Be[3:0] | I | 控制存入的数据的位 |

(3)功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | Store | 将数据加载入dm |
| 2 | Load | 读取dm的数据 |

1. IM

(1)基本描述

指令存储期，接收到pc的值的时候，输出当前的指令

(2)模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| addr[31:0] | I | 输入pc的值 |
| Dout[31:0] | O | 当前执行的指令 |

(3)功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | 读入指令 | $readmemh("src\\code.txt",im); |
| 2 | 输出指令 | 输出当前执行的指令 |

1. Pc

(1)基本描述

存储当前的pc的值，并在时钟上升延改变成npc

(2)模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| npc[31:2] | I | 输入下一个pc的值 |
| Clk | I | 时钟 |
| Rst | I | 复位信号 |
| We | I | 使能信号 |
| pc[31:2] | O | 当前执行的指令的地址 |
| Epc | I | 恢复中断前地址 |

(3)功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | 存储地址 | 存储当前的指令的地址 |
| 2 | 改变地址 | 在时钟上升沿改变地址 |
| 3 | 复位 | 复位信号有效时，pc= 0x3000 |

1. ext\_be

(1)基本描述

遇到store 指令时，通过分析addr来控制输入数据到底存放在何处，对字节进行控制

(2)模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| Op | I | 控制到底是sw，sb,sh指令 |
| In | I | Alu计算出的地址的后两位，对字节进行控制 |
| Be | O | 输出dm的输入的控制 |

(3)功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | 控制保存数据 | 计算出哪些位保存 |

1. mult

(1)基本描述

计算乘除法

(2)模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| Mult\_op | I | 控制到底mult行为 |
| rs | I | 乘除法第一个参数 |
| Rt | I | 乘除法第二个参数，或者mtf0的参数 |
| Hi | O | 计算结果的高位 |
| Li | O | 计算结果的低位 |

(3)功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | 控制保存数据 | 计算出哪些位保存 |

1. cp0

(1)基本描述

中断处理器

1. 模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| Pc | I | 此时执行指令的地址 |
| Din | I | 输入要写入得值 |
| hwint | I | 各个设备的中断信号 |
| sel | I | 要写入或者读出的地址 |
| epc\_we | I | 计算结果的低位 |
| wen | I | Cp0写使能 |
| exlset | I | 开启全局中断 |
| exlclr | I | 关闭全局中断 |
| clk | I | Clk |
| rst | I | Rst |
| intreq | O | 中断请求信号 |
| epc | O | 输出epc的值，eret指令用到 |
| dou | O | 输出cp0某个寄存器的值 |

(3)功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | 产生中断 | 保存cp0，关闭中断，恢复中断等 |

1. Timer

(1)基本描述

计时器

(2)模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| Clk | I | Clk |
| rst | I | rst |
| addr | I | 写入或者读出的地址 |
| we | I | 写使能 |
| din | I | 输入要写的值 |
| Dout | O | 输出某个寄存器的值 |
| irq | O | 终端产生信号 |

(3)功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | 计时1 | 倒数并且在0产生中断 |
| 2 | 计时2 | 倒数，不产生中断 |

1. Bridge

(1)基本描述

桥，用来连接外部设备

1. 模块接口

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| Praddr | I | Cpu输出的地址 |
| Prrd | I | Cpu要输出的值 |
| Prwd | I | Cpu要读入的值 |
| cpu\_we | I | 写使能 |
| dev1\_addr | O | 第一个设备地址 |
| dev1\_we | O | 第一个设备使能 |
| dev1\_wd | O | 第一个设备写入的值 |
| dev1\_rd | O | 第一个设备读出的值 |

(3)功能定义

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 功能描述 |
| 1 | 连接cpu和外部设备 | 使cpu能和外部设备通信 |

问答：

Timer设置的时间不能低于中断处理所需要的周期数，否则程序将一直执行中断。代码无法被执行