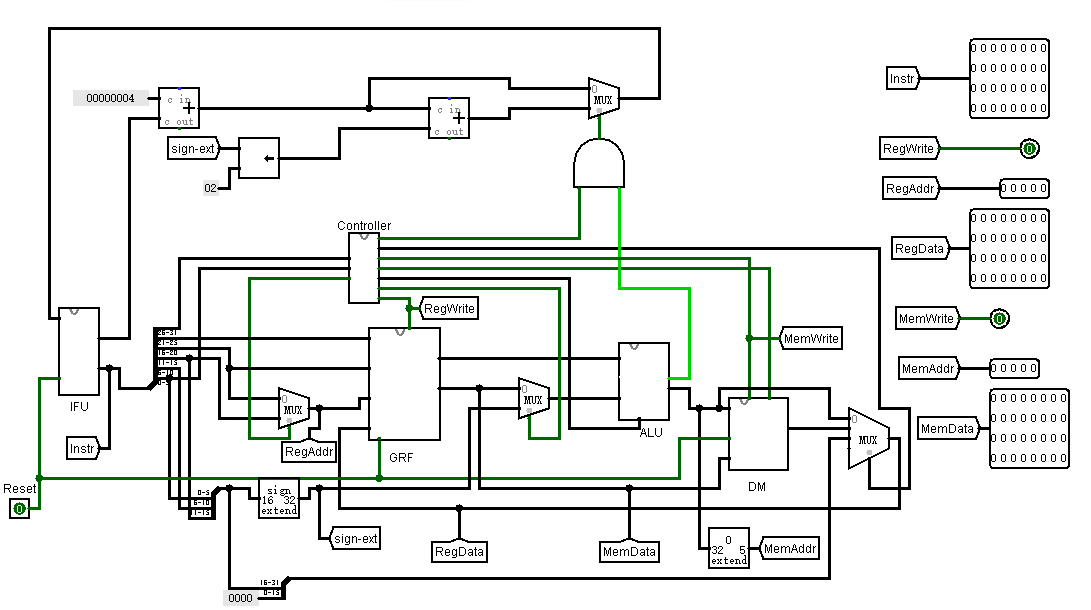
利用logisim开发支持八条指令的MIPS单周期处理器

1. 设计概述:

本实验利用logisim开发支持MIPS指令集中的addu、subu、ori、lw、sw、lui、beq、nop八条指令的单周期处理器。本实验设计的单周期CPU包含Controller(控制器)、IFU(取指令单元)、GRF(通用寄存器组/寄存器文件/寄存器堆)、ALU(算术逻辑运算单元)、DM(数据存储器)、EXT(位扩展)等基本部件，借助多路选择器Multiplexer和译码器Decoder以及Splitter等logisim内置器件组合连接成DataPath。



Logisim搭建的CPU顶层设计图

1. 模块设计：

本实验设计的CPU共有Controller、GRF、IFU、ALU、EXT、DM六个模块。下面给出各模块的模块接口以及功能定义。

1. IFU

模块接口：

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| Reset | I | 复位信号  1:复位  0：无效 |
| Clk | I | 时钟信号 |
| PC\_Update[31:0] | I | PC寄存器的输入端口，时钟信号有效时更新PC的值 |
| PC\_current[31:0] | O | 当前PC的值，指令寄存器IM的地址输入端 |
| Instruction[31:0] | O | 32位的MIPS指令 |

功能定义:

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 描述 |
| 1 | 复位 | 当复位信号有效时，PC<-0x00000000 |
| 2 | 取指令 | 依据PC[6:2],从IM取出指令 |
| 3 | 更新PC | 时钟上升沿(有效),PC寄存器写入新的PC值,获得下一条指令地址 |

1. GRF

模块接口:

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| ReadReg1[4:0] | I | 读寄存器地址1 |
| ReadReg2[4:0] | I | 读寄存器地址2 |
| WriteReg[4:0] | I | 写寄存器地址 |
| WriteEn | I | 读写控制信号  1：写操作  0：读操作 |
| Clk | I | 时钟信号 |
| reset | I | 复位信号  1:复位  0：无效 |
| WData[31:0] | I | 写入数据的输入 |
| RData1[31:0] | O | 32位输出数据1 |
| RData2[31:0] | O | 32位输出数据2 |

功能定义：

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 描述 |
| 1 | 复位 | 当复位信号有效时，GRF的所有寄存器值 0x00000000 |
| 2 | 读寄存器 | 根据读寄存器的输入地址读出数据 |
| 3 | 写寄存器 | 根据写寄存器的输入地址，在写信号有效时，把输入的数据写入指定寄存器中 |

1. ALU

模块接口:

|  |  |  |
| --- | --- | --- |
| 信号名 | 方向 | 描述 |
| A[31:0] | I | 32位输入数据1 |
| B[31:0] | I | 32位输入数据2 |
| ALUOp[1:0] | I | ALU控制信号  00：32位加法  01:32位减法  10：32位或运算 |
| Zero | O | 两个32位数据是否相等 |
| Outcome[31:0] | O | 32位数据输出 |

功能定义：

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 描述 |
| 1 | 加 | A+B |
| 2 | 减 | A-B |
| 3 | 或 | A|B |
| 4 | 相等 | A==B |

1. DM

模块接口：

|  |  |  |
| --- | --- | --- |
| 信号名称 | 方向 | 描述 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |