2024“龙芯杯”个人赛设计报告

学校：安徽新华学院
姓名：马中林

一、设计简介

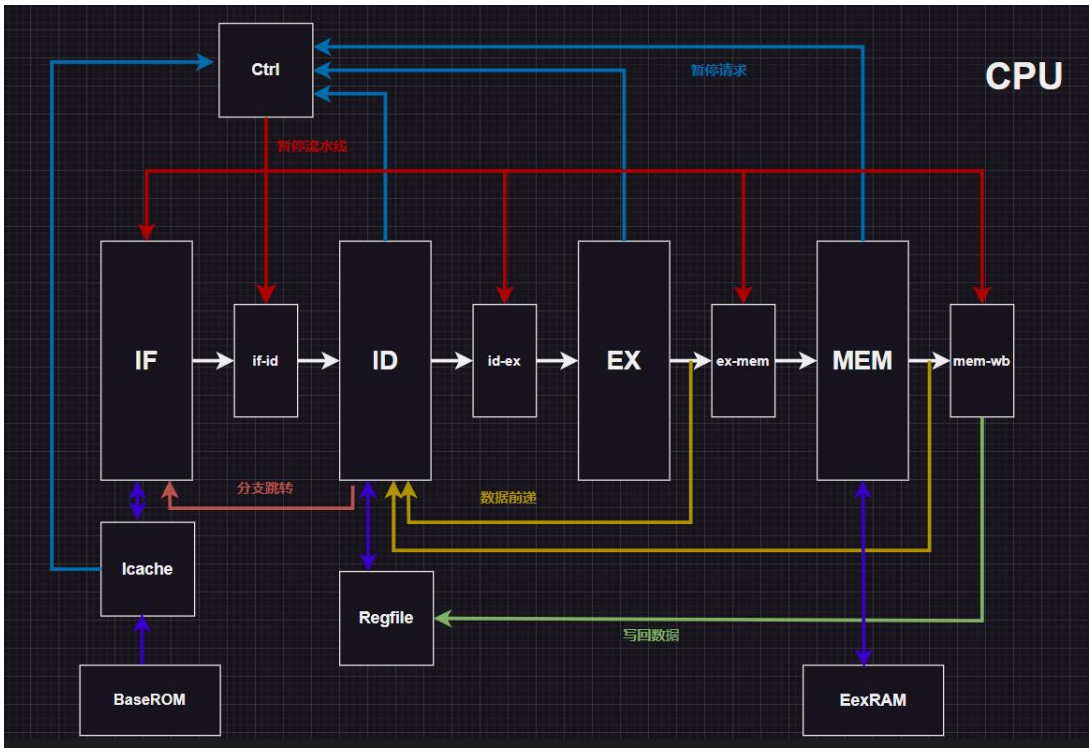
该设计是针对于 2024 年“龙芯杯”计算机系统能力培养大赛个人赛所设计的一个 32 位 MIPS CPU，设计采用顺序单发射机制和经典的五级流水结构，双周期访存，设计添加了 Icache 模块，实现了比赛要求的 MIPS 测评指令集中支持 MIPS-C1 指令集的 6 条指令、 MIPS-C2 指令集的 17 条指令（包含 3 条随机指令 ADD, SRA, BLEZ）、 MIPS-C3 指令集的 22 条指令。在数据相关问题上采用数据前递的方式进行解决，在 load 相关问题和指令数据访存冲突时通过暂停流水线进行解决。支持延迟槽，解决了分支跳转问题。本次设计的 CPU 在所有指令功能测试和性能测试满分通过的基础上，最高可以支持 104MHz 的时钟频率。

二、设计方案

（一）总体设计思路

总体设计：设计采用顺序单发射机制和经典的五级流水结构，主要有两个模块构成：

1. cpu 模块，用于实现一个 32 位的 MIPS CPU。
2. bus 模块，用于实现 SRAM 内存控制和串口控制。



2.1 CPU 模块设计

设计采用顺序单发射机制和经典的五级流水结构，双周期访存，主要由取指模块、译码模块、执行模块、访存模块、写回模块构成,并且添加 Icache 模块提高指令的访问效率。

1.取指模块：负责从指令存储器中读取指令，对 PC 值进行修改跳转指令跳转地址更新下一条指令地址。

端口	位宽	方向	作用
clk	1	输入	时钟信号
rst	1	输入	复位信号
stall	6	输入	暂停信号
branch_flag	1	输入	跳转信号
branch_target_address	32	输入	跳转地址
ce	1	输出	rom 使能信号
pc	32	输出	指令地址

2. 译码模块：接收取值模块传输的数据，将其数据进行译码。根据译码结果进行对不同模块的数据得出相应的操作，将相关数据传递到取值模块和执行模块。

端口	位宽	方向	作用
rst	1	输入	复位信号
pc	32	输入	指令地址
Inst	32	输入	指令
reg1_Data	32	输入	寄存器数据
reg2_Data	32	输入	寄存器数据
ex_we_En	1	输入	译码阶段写使能
ex_we_Data	5	输入	译码阶段写数据
ex_we_Address	32	输入	译码阶段写地址
mem_we_En	1	输入	访存阶段写使能
mem_we_Data	5	输入	访存阶段写数据
mem_we_Address	32	输入	访存阶段写地址
is_in_delayslot_i	1	输入	是否延迟指令
pre_inst_is_load	1	输入	load 信号
branch_flag	1	输出	跳转信号
branch_target_address	32	输出	跳转地址
reg1_Read_En	1	输出	寄存器使能信号
reg2_Read_En	1	输出	寄存器使能信号

reg1_Address	5	输出	寄存器地址
reg2_Address	5	输出	寄存器地址
alu_op	12	输出	ALU 操作
alu_op_Data1	32	输出	ALU 操作数
alu_op_Data2	32	输出	ALU 操作数
we_Address	5	输出	寄存器写地址
we_En	1	输出	寄存器写使能
Offset_Data	32	输出	扩展数
link_Address	32	输出	写入寄存器地址数据
mem_op	4	输出	存储器操作
next_inst_in_delayslot_o	1	输出	下条指令是否为延迟槽
is_in_delayslot_o	1	输出	延迟标志
id_stop	1	输出	流水线暂停信号

3. 执行模块：接收译码模块传输的数据，通过 ALU 对数据进行相关运算，输出结果。将结果数据传递到访存模块。

端口	位宽	方向	作用
rst	1	输入	复位信号
alu_op	12	输入	ALU 操作
alu_op_Data1	32	输入	ALU 操作数
alu_op_Data2	32	输入	ALU 操作数
Offset_Data	32	输入	扩展数
we_En_i	1	输入	寄存器写使能
we_Address_i	5	输入	寄存器写地址
link_address_i	32	输入	写入寄存器地址数据
mem_op_i	4	输入	存储器操作
is_in_delayslot_i	1	输入	延迟标志
pre_inst_is_load	1	输出	load 信号
we_En_o	1	输出	寄存器写使能
we_Address_o	5	输出	寄存器写地址
we_Data_o	32	输出	寄存器写数据
mem_op_o	4	输出	存储器操作
mem_Address	32	输出	存储器写地址
mem_Data	32	输出	存储器写数据
ex_stop	1	输出	流水线暂停信号

4. 访存模块：接收执行模块传输的数据，进行内存相关数据的交互。根据执行模块传输的数据，由访存操作对流水线请求暂停，进行对内存数据的访存。将相关数据传递到写回模块。

端口	位宽	方向	作用
clk	1	输入	时钟信号
rst	1	输入	复位信号
we_En_i	1	输入	寄存器写使能
we_Address_i	5	输入	寄存器写地址
we_Data_i	32	输入	寄存器写数据
mem_op	4	输入	存储器操作
mem_Address_i	32	输入	存储器操作地址
mem_Result_Data_i	32	输入	写入存储器数据
mem_Data_i	32	输入	存储器取出数据
we_En_o	1	输出	寄存器写使能
we_Address_o	5	输出	寄存器写地址
we_Data_o	32	输出	寄存器写数据
mem_Address	32	输出	写存储器地址
mem_We	1	输出	存储器写使能
mem_Sel	4	输出	存储器片选信号
mem_Data	32	输出	写存储器数据
mem_Ce	1	输出	存储器使能信号
Stop	1	输出	流水线暂停信号

5. 写回模块： 接收访存模块传输的数据，将相关数据写回到寄存器文件。

端口	位宽	方向	作用
clk	1	输入	时钟信号
rst	1	输入	复位信号
stall	6	输入	暂停信号
mem_En	1	输入	寄存器写使能
mem_Address	5	输入	寄存器写地址
mem_Data	32	输入	寄存器写数据
wb_En	1	输出	写回寄存器使能
wb_Address	5	输出	写回寄存器地址
wb_Data	32	输出	写回寄存器数据

指令缓存模块：负责存储指令，根据相关操作需要对流水线请求暂停，对指令进行缓存，为取值模块输送指令，提高指令的访问效率。

Cache 设计： cache 大小 2kb。

cache 块大小为一个字。

缓存映射方式：组相联映射，使用 2 路组相联映射共有 32 组。

替换算法：采用 LRU 替换算法。

2.2 BUS 模块设计

Bus 模块：主要功能为对来自 CPU 模块相关数据的信息请求进行处理，对内存地址内存数据、处理串口收发数据、连接 BaseRAM 和 ExtRAM 进行数据交互。

三、设计结果

（一）设计交付物说明

```
├──constrs_1                          //约束文件
│    └──new
│            thinpad_top.xdc
│
├──sim_1
│
└──sources_1
     ├──ip                            // ip 核
     │    └──pll_example
     │
     │
     │
     │
     └──new                           //设计源文件
            ALU.v
            async.v
            Bus.v
            cpu_top.v
            Ctrl.v
            defines.v
            EX.v
            ex_mem.v
            ID.v
            id_ex.v
            IF.v
            if_id.v
            I_Cache.v
            MEM.v
            Mem_S.v
            mem_wb.v
            Regfile.v
            SEG7_LUT.v
            thinpad_top.v
            vga.v
```

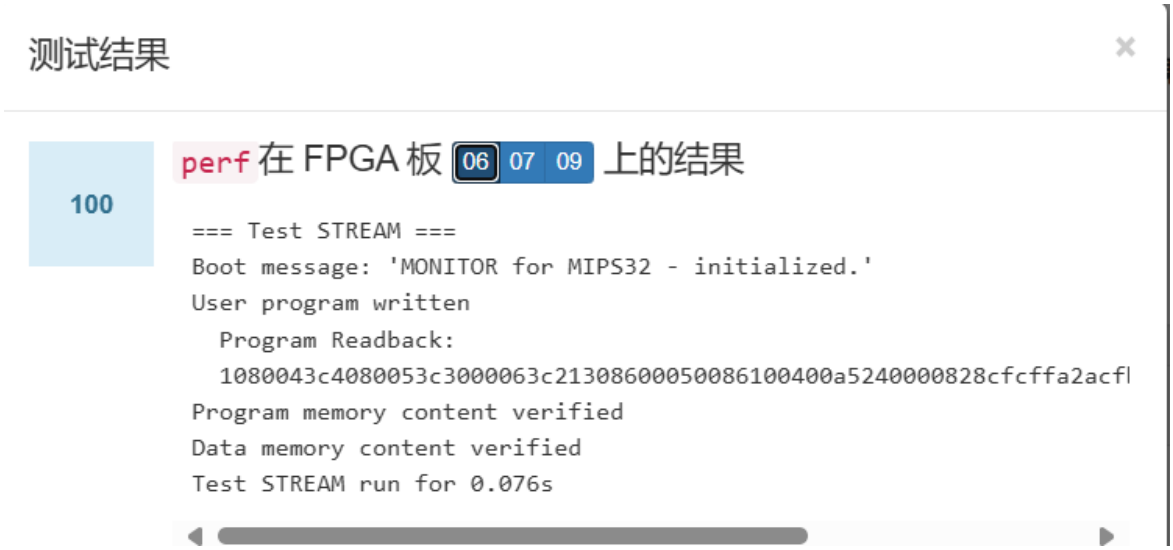
（二）设计演示结果

测试分数表

测试	分数
一级测试	100
二级测试	100
三级测试	100
性能测试	100

性能测试运行时间表

	运行时间
STREAM	0.076s
MATRIX	0.112s
CRYPTONIGHT	0.268s
总用时	0.456s



性能测试 Test STREAM

性能测试 Test MATRIX

性能测试 Test CRYPTONIGHT

四、参考设计说明

1. CPU 的流水线框借鉴《自己动手写 CPU》.(雷思磊)的五级流水框架
2. SRAM 内存控制参考“ 2023“龙芯杯 ”个人赛的开源代码

五、参考文献

[1] 雷思磊. 自己动手写 CPU. 电子工业出版社
[2] 杨全胜. CPU 设计实践教程：从数字电路到计算机组成. 清华大学出版社
[3] 汪文祥,刑金璋.CPU 设计实战. 机器工业出版社
[4] 李月娥,马阿宁,彭宏. 超标量处理器设计. 清华大学出版社