

计算机组成原理实验说明

VANITY FAIR^a, KAEDE^a

^aUniversity of Science and Technology Beijing, 30 XUEYUAN ROAD, HAIDIAN DISTRICT, Beijing, 100083, Beijing, China

Abstract

本文旨在帮助完成 2021 级计组实验课程的考核内容，其中包括了实验需要拓展的指令，以及相关提示信息。

Keywords: 计算机组成原理, 实验, 单周期CPU

目录

1 简介	1
2 提交要求	1
3 扩展指令JAL	1
3.1 格式	3
3.2 描述	3
3.3 操作	3
4 扩展指令JR	3
4.1 格式	5
4.2 描述	5
4.3 操作	5
5 扩展指令BNE	5
5.1 格式	7
5.2 描述	7
5.3 操作	7
6 扩展指令SLTI	7
6.1 格式	9
6.2 描述	9
6.3 操作	9

7 扩展指令SLTIU	9
7.1 格式	11
7.2 描述	11
7.3 操作	11

1. 简介

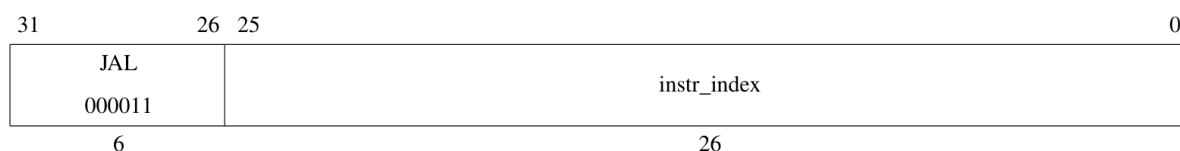
本次实验为了不受之前串口打印实验的影响¹，在CG平台上已经提供了正确添加 `ori`和`lui`指令的CPU，经过测试可以正常在串口输出信息。本次实验基于该工程，但如果你确认自己正确添加了命令，修改了Router文件，那么可以选择使用你自己的CPU工程，此处不做限制。**注意，如果使用自己的项目，请确保提交文件中任何位置都不会在控制台输出字符，否则无法通过测评。**

2. 提交要求

请提交`ALU.v`, `ALUct.v`, `Control.v`, `Core.v`, `IFU.v`. 共五个文件打包而成的压缩包，也可以自行增加模块或更改上述文件名²。并确保所有模块请放置于同一目录等级，也就是说，提交的压缩包中不要有任何文件夹，也不要试图提交`*.data`³。

3. 扩展指令JAL

本文要求扩展的指令为 JAL (Jump and Link)。



3.1. 格式

JAL target

3.2. 描述

在 31 号寄存器（GPR31）中存放返回地址，并跳转至 `target` 位置执行。由于目标地址为形式地址零扩展到 32 位，并左移两位得到 (`target << 2`)，因此有效目标地址在一个 256 MB 的空间中，有关此指令与延迟槽（delay slot）的关系，请查阅 *MIPS32™ Architecture For Programmers Volume II: The MIPS32™ Instruction Set*

¹据本文作者所知，许多人没有正确实现串口打印功能，这将会导致本实验验收无法正常通过

²不建议这么做

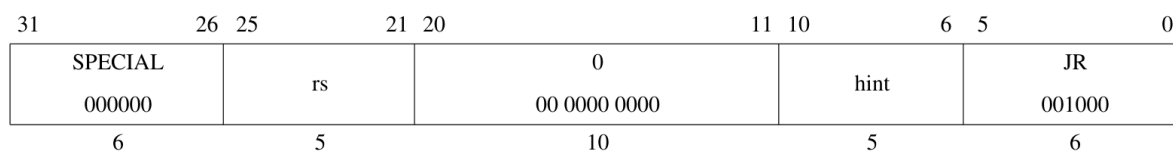
³这不会对你的分数产生任何影响

3.3. 操作

$$\text{GPR}[31] \leftarrow \text{PC} + 4$$
$$\text{PC} \leftarrow 0^4 \parallel \text{target} \parallel 0^2$$

4. 扩展指令JR

本文要求扩展的指令为 JR (Jump Register)。



4.1. 格式

JR **rs**

4.2. 描述

跳转到寄存器 *rs* 中地址的位置。有关此指令与延迟槽（delay slot）的关系，请查阅 *MIPS32™ Architecture For Programmers Volume II: The MIPS32™ Instruction Set*

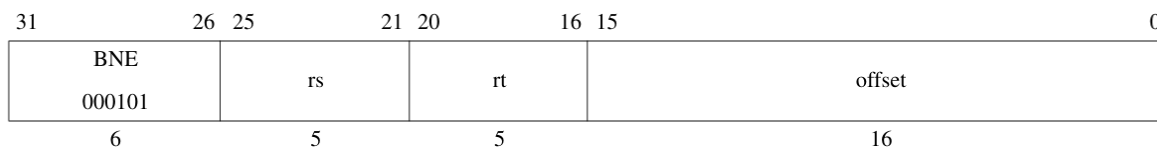
本指令要求 *rs* 的值是对齐的，否则会产生 Address Error exception.

4.3. 操作

$$\text{PC} \leftarrow \text{rs}$$

5. 扩展指令BNE

本文要求扩展的指令为 BNE (Branch on Not Equal)。



5.1. 格式

BNE rs, rt, offset

5.2. 描述

如果 $rs \neq rt$ ，则进行分支。

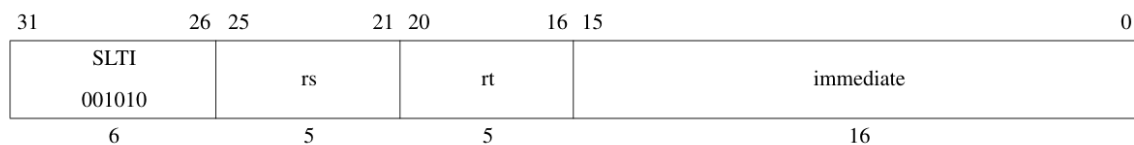
一个 18 位有符号偏移量（由 16 位 offset 左移两位得到）加到当前指令下一条指令的地址中，有关此指令与延迟槽（delay slot）的关系，请查阅 *MIPS32™ Architecture For Programmers Volume II: The MIPS32™ Instruction Set*

5.3. 操作

```
if GPR[rs] ≠ GPR[rt] then
    target_offset ← sign_extend (offset || 02)
    PC ← PC + 4 + target_offset
endif
```

6. 扩展指令SLTI

本文要求扩展的指令为 SLTI (Set on Less Than Immediate)。



6.1. 格式

SLTI rt, rs, immediate

6.2. 描述

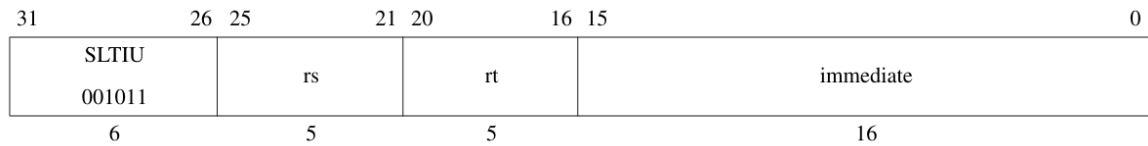
将 rs 的内容与 16 位有符号立即数比较，在 rt 中记录比较结果：如果 rs 小于立即数，结果为 1，否则为 0。

6.3. 操作

```
if GPR[rs] < sign_extend immediate then
    GPR[rt] ← 031 || 1
else
    GPR[rt] ← 032
endif
```

7. 扩展指令SLTIU

本文要求扩展的指令为 SLTIU (Set on Less Than Immediate Unsigned)。



7.1. 格式

SLTIU rt, rs, immediate

7.2. 描述

将 **rs** 的内容与 16 位无符号立即数进行无符号比较，在 **rt** 中记录比较结果：如果 **rs** 小于立即数，结果为 1，否则为 0。注意，为了简化操作，本指令与 MIPS 指令集行为不相同

7.3. 操作

```
if GPR[rs] < zero_extend(immediate) then
    GPR[rt] ← 031 || 1
else
    GPR[rt] ← 032
endif
```