
32位MIPS处理器实验需求文档

Cache小组

2015年1月8日

目录

1	引言	3
1.1	编写目的	3
1.2	背景	3
1.3	定义	3
1.4	参考资料	3
2	功能需求	5
2.1	CPU	5
2.1.1	ALU	5
2.1.2	乘法器	6
2.1.3	CPO	6
2.1.4	MMU与TLB	7
2.1.5	异常中断处理	9
2.1.6	其他功能部件	11
2.1.6.1	PC	11
2.1.6.2	寄存器堆	11
2.1.6.3	串口	12
2.1.7	扩展部分	12
2.1.8	指令集与数据通路	12
2.2	BIOS	13
3	性能需求	14
4	运行环境需求	15
4.1	设备	15
4.2	控制	15

5 附录	16
5.1 指令系统	16

1 引言

1.1 编写目的

计算机组成原理32位MIPS实验是在计原16位实验的基础上的扩展。在实验原理方面与多门课程相结合，涉及到操作系统、软件工程、编译原理等等多个方面，实验初期学习曲线较陡，在原理方面比较难以掌控。在编程实现方面涉及到大规模VHDL代码的书写，需要对数字逻辑设计有比较清晰的思路，工作量非常大。

因此，为控制整个开发过程，明确项目需求与目标，编写此需求文档。

文档预期读者为任务提出者：刘卫东老师、李山山老师、白晓颖老师。

1.2 背景

系统名称：32位MIPS处理器

任务提出者：计算机组成原理课程：刘卫东老师、李山山老师

软件工程课程：白晓颖老师

开发者：计23 李天润

计23 胡津铭

计23 孙皓

1.3 定义

以下是此次32位MIPS实验中需要用到的专业名词定义

表 1: 定义列表

名词	描述
MIPS	无内部互锁流水级的微处理器
CPU	中央处理器
CPO	协处理器0
MMU	内存管理单元
TLB	翻译后备缓冲
ALU	算术逻辑单元
RAM	存储程序的硬件，断电不保存信息
ROM	存储程序的硬件，断电可保存信息
Flash	存储程序的硬件，断电可保存信息，实验中用作硬盘
BIOS	主板上的启动程序，负责初始化硬件引导操作系统
BootLoader	一段特殊程序，将操作系统从Flash中加载到内存中，并且开始执行

1.4 参考资料

实验指导文档

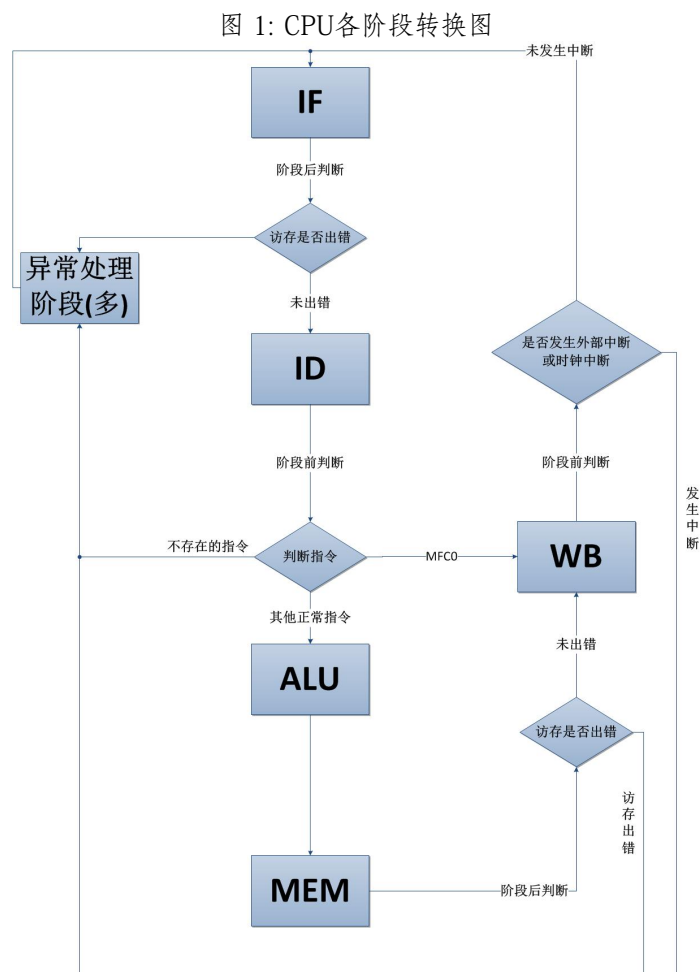
OsLab实验参考文档

计算机组成原理综合实验报告 贾开
《计算机组成和设计 硬件/软件接口》
《See MIPS Run》

2 功能需求

2.1 CPU

CPU整体工作流程图



2.1.1 ALU

功能需求:

1. 完成数据和地址的算术、逻辑和移位运算，输入两个数据根据ALUOp得出输出结果。
2. 根据指令系统的需求，完成指令系统中乘法之外的算术指令。

实现方式：

1. ALU两个输入数据，ALUOp目前整理出9种运算。
2. 输出最终运算结果，三个标志位。

2.1.2 乘法器

功能需求：

1. 完成乘法功能，结果保存在LO和HI寄存器中，可以通过MT和MF类指令访问和修改计算结果。

实现方式：

1. 使用IPCore实现。
2. 直接与主ALU相结合，在ALUOp中加入乘法的操作码，实现乘法的计算。
3. 乘法指令需要的计算时间较长，无法在一个时钟上升沿中完成计算。但是由于是多周期CPU，指令执行的控制过程比较灵活，可以在乘法指令时增加多个时钟上升沿，当乘法完成后在继续执行下一条指令
4. MFLO, MFHI, MTLO, MTHI, MULT指令查看与修改。在ALUOp中也加入支持MTHI、MTLO的操作码，实现对LO和HI的修改。同时LO和HI始终处于可读的状态，连接至写回阶段，MFLO与MFHI通过对写回阶段RegValue的控制来实现。

2.1.3 CPO

功能需求：

辅助操作系统对硬件进行管理。

1. 内存管理：辅助操作系统，完成虚拟地址和物理地址的转换、不同进程之间的内存切换、用户态与内核态内存的分离
2. 异常处理：检测指令执行过程中可能出现的异常、对异常进行分类、记录异常出现的指令地址或错误的内存地址
3. 外部中断：负责检测外部中断，辅助实现CPU与外设之间的交互

实现方式：

1. CPO记录CPU当前状态，通过MFCO、MTCO指令等进行读写操作，与通用寄存器建立联系
2. 在异常状态发生时保存异常的返回地址，异常类型等信息
3. 在内存映射中对TLB表项进行支持，为TLB充填等功能提供硬件支持
4. CPO寄存器的写方式有两种：

表 2: 需要实现的CPO寄存器

序号	名称	功能
0	Index	用于TLBWI指令访问TLB入口的索引序号
1	EntryLo0	作为TLBWI及其他TLB指令接口,管理偶数页入口
2	EntryLo1	作为TLBWI及其他TLB指令接口,管理奇数页入口
3	EntryHi	TLB异常时,系统将虚拟地址部分写入EntryHi寄存器中用于TLB匹配
4	BadVAddr	捕捉最近一次地址错误或TLB异常(重填、失效、修改)时的虚拟地址
5	Count	每隔一个时钟周期增加1,用作计时器,并可使能控制
6	Compare	当Count值与Compare相等时,SI_TimerInt 引脚是变高电平直到有数值写入Compare, 用于定时中断
7	Status	表示协理器的操作模式、中断使能及诊断状态
8	Cause	记录最近依次异常的原因,控制软件中断请求以及中断协理派分的向量
9	EPC	存储异常处理之后程序恢复执行的地址
10	EBase	识别多协理器系统中不同的协理器异常向量的基地址

- (a) 正常执行状态下, 通过MTC0指令, 将CPO寄存器编号, CPO写入值, CPO控制信号传递给CPO寄存器, 进行写入操作。
- (b) 异常状态下, 通过异常处理模块到CPO的连接, 直接将异常原因、异常指令地址、访存错误地址等信息, 写入cause、EPC、Bad-VAddr寄存器中

5. CPO寄存器的读方式有两种:

- (a) 正常执行状态下, 通过MFC0指令, 将CPO寄存器编号传递给CPO寄存器, 之后读取的值通过写回阶段写到通用寄存器中
- (b) 针对TLBWI等指令对CPO寄存器的读需求(需要同时读取5个CPO寄存器), 将CPO寄存器与TLB相关的直接连接到MMU单元, 通过TLBWI的使能信号控制写入TLB

CPO具体实现方式参照下面两个小节。
此次实验中需要实现的CPO寄存器见表2

2.1.4 MMU与TLB

功能需求:

1. 实现虚拟地址(线性地址)到物理地址的转换。只对部分地址([0x20000000~0x80000000]和[0xC0000000~0xFFFFFFFF])进行地址映射, 其余部分直接映射。
2. 实现内核态与用户态内存的区分。
3. 实现相对应的异常处理: TLBS、TLBL、TLB Modified。

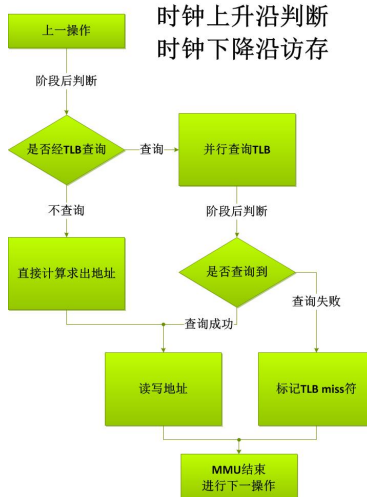
实现方式：

1. (a) TLB共16项，每项可以将连续两个虚页映射到两个不同的物理页，相当于可以处理32个页
- (b) 虚拟地址高19位作为虚页号VPN进行选择TLB表项，第20位通过奇偶判断，选择两个物理页之一
- (c) 出于简化的考虑，不维护进程号ASID、全局标记G，默认所有G均为1，只维护每个物理地址的D标记
2. 硬件支持：
 - (a) 虚拟地址高19位在TLB表项中并行查询EntryHi部分。
 - (b) 如果查找到，根据第20位选择EntryLo，直接与低12位结合得到真实的物理地址。
 - (c) 如果未查找到，触发TLBMiss异常：设置Cause寄存器中的ExcCode为TLB异常，EPC寄存器为当前指令的地址，BadvAddr寄存器为错误的地址，之后触发一个异常，PC跳转到EBase的异常处理基地址，之后由操作系统接管。
 - (d) 对于地址不对齐异常，在内存映射阶段对虚拟地址offset部分最低两位进行检测如果不为00，则触发地址不对齐异常，交由操作系统处理

图 2: TLB电路结构

	VPN2	PFN1	V1	D1	PFN2	V2	D2
范围	62-44	43-24	23	22	21-2	1	0

图 3: 内存访问流程图
时钟上升沿判断
时钟下降沿访存



3. 操作系统：

- (a) 进入异常处理向量 (trap/vector.S)。
- (b) 跳转到处理函数部分 (trap/exception.S)。
- (c) 先保存异常现场，之后进入mips_trap函数 (trap/trap.c)。
- (d) 根据cause寄存器的值进行分类，调用handle_tlbmiss函数。
- (e) 得到异常地址对应的物理页号，之后进行tlb_refill (include/thu-mips_tlb.h)。
- (f) 操作系统维护index，写EntryLo0、EntryLo1、EntryHi、PageMask、Index五个寄存器，之后用tlbwi写到TLB的第Index项。此处需要硬件提供MTCO、MFCO、TLBWI指令的支持。
- (g) 异常返回，重新执行取地址命令。

2.1.5 异常中断处理

功能需求：

- 1. 实现对异常和外部中断的处理。
- 2. 实现的异常有内存访问异常、地址不对齐异常、系统调用、未定义的指令异常、未定义的寄存器异常。
- 3. 外部中断：键盘、通讯端口。

实现方式：

1. 硬件支持

- (a) 异常处理，在数据通路上添加异常处理部分，在多周期的任意一个周期检测到异常后，在Cause设置异常代码、在EPC寄存器设置异常处理的返回地址，将Status寄存器的EXL为设置为1，表示进入内核态处理异常之后根据EBase跳转到异常处理向量基地址，之后操作系统接管。
- (b) 外部中断处理，设置外部中断检查信号，外部设备触发时，异步地将检查信号置1；每个指令IF阶段检查该信号，若为1则进入外部中断处理，否则正常执行指令。
- (c) 时钟中断处理。当寄存器Count和Compare寄存器中的数值相等时，触发一个时钟异常，交由操作系统处理。

2. 操作系统

- (a) 进入异常处理向量 (trap/vector.S)。
- (b) 跳转到处理函数部分 (trap/exception.S)。
- (c) 先保存异常现场，之后进入mips_trap函数 (trap/trap.c)。

图 4: 异常中断处理流程
操作系统异常处理简介 (软件)

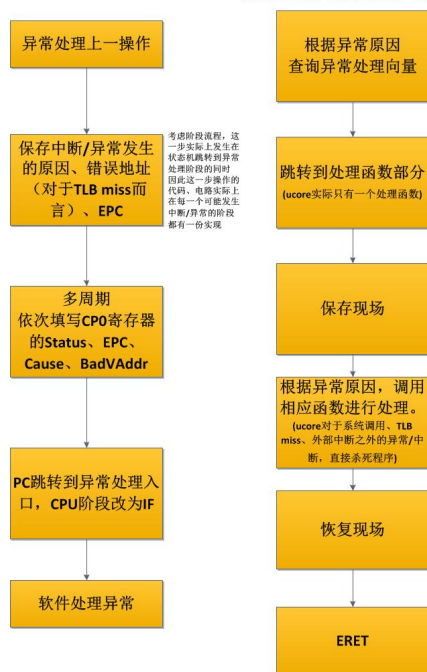


表 3: 需要实现的异常

异常号	异常名	描述
0	Interrupt	外部中断,异步发生,由硬件引起
1	TLB Modified	内存修改异常,发生在Memory阶段
2	TLBL	读未在TLB中映射的内存地址触发的异常
3	TLBS	写未在TLB中映射的内存地址触发的异常
4	ADEL	读访问一个非对齐地址触发的异常
5	ADES	写访问一个非对齐地址触发的异常
8	SYSCALL	系统调用
10	RI	执行未定义指令异常
11	Co-Processor Unavailable	试图访问不存在的协处理器异常

- (d) 根据cause寄存器的值进行分类, 调用interrupt_handler、syscall等等函数进行处理, 如果是地址不对齐异常, 未定义的指令或未定义的异常, 直接退出。
- (e) 使用ERET指令从异常返回, 将EXL位重新设置为0, 表示进入用户态 重新执行取地址命令

2.1.6 其他功能部件

2.1.6.1 PC

功能需求:

1. 实现PC的多种变换方式: 正常执行、分支、跳转。
2. 实现异常处理时对PC的处理, 跳转到异常处理向量部分。

实现方式

1. 通过多路选择器选择ALU的输入数据, 与当前PC进行运算实现PC的多种变化方式。
2. 针对异常状态、从异常返回两种情况的PC变化, 从CPO寄存器中引出EPC与EBase两个地址到PC计算单元。 通过当前的状态进行判断PC的取值。
3. 下一条指令的PC在本条指令的写回阶段得到确定, 保证在下一条指令的取指令时钟上升沿开始之前, 正确送到MMU进行取指令。

2.1.6.2 寄存器堆

功能需求:

1. 实现通用寄存器, 以及在数据通路中的读写控制。

实现方式

1. 在CPU中实现寄存器堆并实现读写控制。
2. 在解码指令阶段，根据指令的rs、rt、rd位置读取通用寄存器。与写回模块相配合，完成寄存器堆的写入

2.1.6.3 串口

功能需求：

1. 实现与PC机通信，通过计算机键盘输入数据，向计算机输出数据。

实现方式

1. 使用开源的ASync transmitter and receiver，通过在每个信号周期内多次采样，并滤波得到稳定的信号。CPLD仅负责TX/RX端口数据的转发，对数据的编码与解码在FPGA中进行。这样设计提高了数据传送的准确率，并且串口数据不需要占用内存数据线，无需处理冲突，简化了设计。数据通信协议在开发过程中设计，有待进一步说明。

2.1.7 扩展部分

在开发过程中需要分模块进行测试，因此在开发阶段计划实现一个分模块的测试工具。

1. 使用仿真工具ModelSim或者Xilinx自带的ISim进行仿真，在给定的输入波形下产生对应的输出波形，将输入与输出波形都输出到文件中。之后再使用该测试工具（用python或者c++等高级语言实现），对输出与标准输出进行比较，判断该模块是否正常工作。
2. 在模块测试阶段，使用Xilinx提供的ipcore生成一块片内RAM，作为临时的存储区域。能够做到在一个时钟上升沿内进行读写操作，操作比一般RAM简便。由此在MMU和MEM部分未完成的情况下，可以对其他模块进行独立的测试，而且可以进行内存的访问。
3. 每个模块正常工作之后，将各个模块进行组合，形成整体的CPU设计，再进行后续调试工作。

对于测试工具部分会实时与老师进行沟通，明确老师对于测试工具的要求。

如果操作系统已经能够在CPU上正常运行，而且距离项目结束还有比较长的时间，则可以考虑对网口进行扩展。

2.1.8 指令集与数据通路

指令集为标准MIPS系统子集

实现多周期CPU，针对指令集设计数据通路。仿照《软件硬件接口》书中的多周期CPU进行设计，在书中7条指令的原型上进行扩展，以支持标准MIPS系统的子集。

指令执行过程主要分为取指、解码、执行、访存、写回五个周期，某些指令可能需要其他特殊的周期支持，采用多周期实现比较灵活。

数据通路包括状态机与控制线设计，每条指令有不同的状态机变化，而且对指令进行解码，能够得到相应的控制线。在多周期的不同周期利用控制线对数据通路进行控制，使指令正常执行。

数据通路图比较大，因此不显示在需求文档中。另附图片文件显示。

2.2 BIOS

准备阶段：

1. 操作系统和其他程序烧写到Flash中
2. bootasm.S编译出的二进制文件写到ROM中
3. CPU代码烧写到FPGA中。

启动阶段：

1. 从ROM中的bootasm.S启动
2. 将Flash中的全部操作系统拷贝到RAM中，然后控制权交给操作系统。

操作系统启动阶段：

1. bootasm.S工作结束后，进入操作系统entry.S文件中进行寄存器初始化。
2. entry.S：通用寄存器清零，CPO寄存器设置为默认值，异常处理基地址进行初始化，之后跳转到init.c中的kern_init。
3. kern_init：完成TLB初始化（全部TLB表项清零），物理内存与虚拟内存的转换页表的初始化（创建初始页表）
4. 最后完成操作系统的进程和文件管理系统，至此ucore系统开始正常工作，

3 性能需求

实现多周期CPU，性能上可能与流水线存在一定差距， 尽量通过良好的内部设计弥补，减小各个模块之间的延时。

参照贾开学长 $12.5MHz$ 的时钟主频，将我们的主频目标暂定为 $6.25MHz$ ，主频可能根据具体实现过程进行调整。

4 运行环境需求

4.1 设备

表 4: 设备与外部接口

环境	描述
FPGA	Xilinx Spartan6 xc6slx100
RAM	32-bit字长, 4块, 共8MB
Flash	16-bit 字长, 共8MB
CPLD	与FPGA相连, 用于I/O
串口	2个
USB 串口	1个
ps/2 接口	1个
以太网接口	1个
VGA 接口	1个

4.2 控制

控制部分采用PS/2键盘作为输入设备, FPGA进行计算与处理之后通过VGA进行输出。如果有需要使用串口与PC进行通信, 扩展功能可以采用网口实现开发板与PC之间的通信。

5 附录

5.1 指令系统

指令格式	ADDU rd rs rt
指令功能	$R[d] \leftarrow R[s] + R[t]$
功能说明	将寄存器 rs 与寄存器 rt 的值求和, 结果保存到寄存器 rd 中

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	0	0	0	0	rs					rt					rd					0	0	0	0	0	1	0	1	0	1	0
指令格式	SLT rd rs rt																															
指令功能	if(R[s] < R[t]) then R[d] = 1, else R[d] = 0																															
功能说明	比较寄存器 rs 与寄存器 rt 的值并根据结果对寄存器 rd 赋值																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	1	0	1	0	rs						rt						immediate													
指令格式	SLTI rt rs immediate																															
指令功能	if(R[s] < Sign-extend(immediate)) R[t] = 1, else R[t] = 0																															
功能说明	比较寄存器 rs 的值与进行符号扩展后的立即数的值, 并根据结果对寄存器 rt 赋值																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	1	0	1	1	rs					rt					immediate															
指令格式	SLTIU rt rs immediate																															
指令功能	if(R[s] < Zero-extend(immediate)) R[t] = 1, else R[t] = 0																															
功能说明	比较寄存器 rs 的值与进行零扩展后的立即数的值, 并根据结果对寄存器 rt 赋值																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	0	0	0	0					rs					rt					rd		0	0	0	0	0	1	0	1	0	1
指令格式	SLTU rd rs rt																															
指令功能	if(R[s] < R[t]) R[d] = 1, else R[d] = 0																															
功能说明	比较寄存器 rs 与寄存器 rt 的值, 并根据结果对寄存器 rt 赋值																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	0	0	0	0					rs					rt					rd		0	0	0	0	0	1	0	0	0	1
指令格式	SUBU rd rs rt																															
指令功能	$R[d] \leftarrow R[s] - R[t]$																															
功能说明	用寄存器 rs 的值减寄存器 ry 的值, 结果保存到寄存器 rz 中																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	0	0	0	0					rs					rt		0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
指令格式	MULT rs rt																															
指令功能	HI LO ← R[s] * R[t]																															
功能说明	将寄存器 rs 与寄存器 rt 的值相乘，结果的高 32 位保存到 HI 寄存器中，低 32 位保存到 LO 寄存器中																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
--	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

[illegible]

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
指令 编码	0	0	0	0	0	0	rs					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1
指令格式	MTLO rs																																
指令功能	L0 ← R[s]																																
功能说明	将寄存器 rs 的值保存到 L0 寄存器中																																

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
指令 编码	0	0	0	0	0	0	rs					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
指令格式	MTHI rs																																
指令功能	HI ← R[s]																																
功能说明	将寄存器 rs 的值保存到 HI 寄存器中																																

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令 编码	0	0	0	1	0	0	rs					rt					immediate															
指令格式	BEQ rs rt immediate																															
指令功能	if(R[S] = R[T]) PC ← PC + Sign-extend(immediate)																															
功能说明	如果寄存器 rs 与寄存器 rt 的值相等，则跳转到目的地址执行，否则顺序执行下一条指令																															

[illegible]

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	0	1	1	1	rs					0	0	0	0	immediate																
指令格式	BG7 rs immediate																															
指令功能	if(R[s] > 0) PC ← PC + Sign-extend(immediate)																															
功能说明	如果寄存器 rs 的值大于 0, 则跳转到目的地址执行, 否则顺序执行下一条指令																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
指令编码	0	0	0	1	1	0	rs					0	0	0	0	0	immediate																
指令格式	BLEZ rs																																
指令功能	if (R[s] <= 0) PC ← PC + Sign-extend(immediate)																																
功能说明	如果寄存器 rs 的值小于等于 0，则跳转到目的地址执行，否则顺序执行下一条指令																																

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
指令编码	0	0	0	0	0	1	rs					0	0	0	0	0	immediate																
指令格式	BLTZ rs																																
指令功能	if (R[s] < 0) PC ← PC + Sign-extend(immediate)																																
功能说明	如果寄存器 rs 的值小于 0，则跳转到目的地址执行，否则顺序执行下一条指令																																

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	0	1	0	1	rs					rt				immediate																
指令格式	BNE rs rt																															
指令功能	if (R[s] != R[t]) PC ← PC + Sign-extend(immediate)																															
功能说明	如果寄存器 rs 与寄存器 rt 的值不相等，则跳转到目的地址执行，否则顺序执行下一条指令																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
指令编码	0	0	0	0	1	0	immediate																										
指令格式	J immediate																																
指令功能	PC ← Sign-extend(immediate)																																
功能说明	无条件跳转至目的地址执行																																

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
指令编码	0	0	0	0	1	1	immediate																										
指令格式	JAL																																
指令功能	PC ← Sign-extend(immediate), RA ← RPC																																
功能说明	无条件跳转至目的地址执行，将延时槽后一条指令的地址保存到 RA 寄存器中																																

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
指令编码	0	0	0	0	0	0	rs					0	0	0	0	0	rd				0				0	0	0	0	0	0	1	0	0	1
指令格式	JALR rs rd																																	
指令功能	PC ← R[s], R[d] ← RPC																																	
功能说明	无条件跳转至寄存器 rs 中所存地址执行，将延时槽后一条指令的地址保存到 RA 寄存器中																																	

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
指令编码	0	0	0	0	0	0	rs					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
指令格式	JR rs																																	
指令功能	PC ← R[s]																																	

功能说明	无条件跳转至寄存器 rs 中所存地址执行																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令编码	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															
	1	0	0	0	1	1	rs					rt					immediate																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令格式	LW rt rs immediate																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令功能	$R[t] \leftarrow \text{MEM}[R[s] + \text{Sign-extend}(\text{immediate})]$																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
功能说明	将寄存器 rs 的值与立即数 immediate 符号扩展后相加所得地址中的数据取出来保存至 rt 中																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令编码	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															
	1	0	1	0	1	1	rs					rt					immediate																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令格式	SW rt rs immediate																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令功能	$\text{MEM}[R[s] + \text{Sign-extend}(\text{immediate})] \leftarrow R[t]$																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
功能说明	将寄存器 rt 的值存入寄存器 rs 的值与立即数 immediate 符号扩展后相加所得地址中																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令编码	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															
	1	0	0	0	0	0	rs					rt					immediate																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令格式	LB rt rs immediate																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令功能	$R[t] \leftarrow \text{Sign-extend}(\text{MEM_Byte}[R[s] + \text{Sign-extend}(\text{immediate})])$																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
功能说明	将寄存器 rs 的值与立即数 immediate 符号扩展后相加所得地址中的第一个字节取出来符号扩展后保存在寄存器 rt 中																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令编码	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															
	1	0	0	1	0	0	rs					rt					immediate																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令格式	LBU rt rs immediate																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令功能	$R[t] \leftarrow \text{Zero-extend}(\text{MEM_Byte}[R[s] + \text{Sign-extend}(\text{immediate})])$																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
功能说明	将寄存器 rs 的值与立即数 immediate 符号扩展后相加所得地址中的第一个字节取出来零扩展后保存在寄存器 rt 中																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令编码	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															
	1	0	1	0	0	0	rs					rt					immediate																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令格式	SB rt rs immediate																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令功能	$\text{MEM_Byte}[R[s] + \text{Sign-extend}(\text{immediate})] \leftarrow \text{LOW_BYTE}[R[t]]$																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
功能说明	将寄存器 rt 的最低字节取出来保存在寄存器 rs 的值与立即数 immediate 符号扩展后相加所得地址中																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
指令编码	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															
	0	0	0	0	0	0	rs					rt					rd					0	0	0	0	0	1	0	0	1	0	0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															

指令编码	0	0	1	1	0	0	rs				rt				immediate											
指令格式	ANDI rt rs immediate																									
指令功能	$R[t] \leftarrow R[s] \& \text{Zero-extend(immediate)}$																									
功能说明	将寄存器 rs 的值与立即数零扩展后相与的结果保存至寄存器 rt 中																									

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
指令编码	0	0	1	1	1	0	0	0	0	0	rt					immediate																		
指令格式	LUI rt immediate																																	
指令功能	R[t] ← immediate * 65536																																	
功能说明	将 16 位立即数放至寄存器 rt 的高 16 位中																																	

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
指令编码	0	0	0	0	0	0	rs					rt					rd					0	0	0	0	0	1	0	0	1	1	1	1
指令格式	NOR rd rs rt																																
指令功能	$R[d] \leftarrow \sim(R[s] \mid R[t])$																																
功能说明	将寄存器 rs 与寄存器 rt 的值或非后的结果保存至寄存器 rd 中																																

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
指令编码	0	0	0	0	0	0	rs					rt					rd					0	0	0	0	0	1	0	0	1	0	1	1
指令格式	OR rd rs rt																																
指令功能	R[d] ← R[s] R[t]																																
功能说明	将寄存器 rs 与寄存器 rt 的值相或后的结果保存至寄存器 rd 中																																

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
指令编码	0	0	1	1	0	1	rs					rt					immediate																	
指令格式	ORI rt rs immediate																																	
指令功能	$R[t] \leftarrow R[s] \mid \text{Zero-extend(immediate)}$																																	
功能说明	将寄存器 rs 与立即数 immediate 零扩展后相或的结果保存至寄存器 rd 中																																	

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
指令编码	0	0	0	0	0	0	rs					rt					rd					0	0	0	0	0	1	0	0	1	1	0	1	0
指令格式	XOR rd rs rt																																	
指令功能	$R[d] \leftarrow R[s] \wedge R[t]$																																	
功能说明	将寄存器 rs 与寄存器 rt 的值异或后的结果保存至寄存器 rd 中																																	

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
指令编码	0	0	1	1	1	0	rs					rt					immediate																	
指令格式	XORI rt rs immediate																																	
指令功能	$R[t] \leftarrow R[s] \wedge \text{Zero-extend(immediate)}$																																	
功能说明	将寄存器 rs 与立即数 immediate 零扩展后相异或的结果保存至寄存器 rd 中																																	

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	0	0	0	0	0	0	0	0	0																					
指令格式	SLL rd rt immediate																															
指令功能	$R[d] \leftarrow R[t] \ll \text{immediate}$																															
功能说明	将寄存器 rt 中的值逻辑左移立即数 immediate 位后的结果保存至寄存器 rd 中																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	0	0	0	0																										
指令格式	SLLV rd rt rs																															
指令功能	$R[d] \leftarrow R[t] \ll R[s]$																															
功能说明	将寄存器 rt 中的值逻辑左移寄存器 rs 中的值位后的结果保存至寄存器 rd 中																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	0	0	0	0	0	0	0	0	0																					
指令格式	SRA rd rt immediate																															
指令功能	$R[d] \leftarrow R[t] \gg \text{immediate}(\text{arithmetic})$																															
功能说明	将寄存器 rt 中的值算数右移立即数 immediate 位后的结果保存至寄存器 rd 中																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	0	0	0	0																										
指令格式	SRAV rd rt rs																															
指令功能	$R[d] \leftarrow R[t] \gg R[s](\text{arithmetic})$																															
功能说明	将寄存器 rt 中的值算数右移寄存器 rs 中的值位后的结果保存至寄存器 rd 中																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	0	0	0	0	0	0	0	0	0																					
指令格式	SRL rd rt immediate																															
指令功能	$R[d] \leftarrow R[t] \gg \text{immediate}(\text{logical})$																															
功能说明	将寄存器 rt 中的值逻辑右移立即数 immediate 位后的结果保存至寄存器 rd 中																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	0	0	0	0	0																										
指令格式	SRLV rd rt rs																															
指令功能	$R[d] \leftarrow R[t] \gg R[s](\text{logical})$																															
功能说明	将寄存器 rt 中的值逻辑右移寄存器 rs 中的值位后的结果保存至寄存器 rd 中																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
指令编码	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
指令格式	SYSCALL																																
指令功能	中断号 \leftarrow SYSCALL																																

功能说明	执行后除法中断
------	---------

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	1	0	1	1	1																											
指令格式	CACHE																															
指令功能	无																															
功能说明	不做 cache, 视为 NOP																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
指令格式	ERET																															
指令功能	$PC \leftarrow EPC$																															
功能说明	返回至 EPC 寄存器的地址执行, 并设置 Status 寄存器的 EXL 位为 0																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	1	0	0	0	0	0	0	0	0	0																					
指令格式	MFC0 rt rd																															
指令功能	$R[t] \leftarrow CP0[R[d]]$																															
功能说明	将协处理器 0 中的 rd 寄存器的值保存到 rt 寄存器中																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	1	0	0	0	0	0	1	0	0																						
指令格式	MTC0 rd rt																															
指令功能	$CP0[R[d]] \leftarrow R[t]$																															
功能说明	将寄存器 rt 的值保存到协处理器 0 中的 rd 寄存器中																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
指令格式	TLBWI																															
指令功能																																
功能说明	写索引 TLB 项																															

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
指令编码	1	0	0	1	0	1																										
指令格式	LHU rt rs immediate																															
指令功能	$R[t] \leftarrow \text{Zero-extend}(\text{MEM_HALFWORD}[R[s] + \text{Sign-extend}(\text{immediate})])$																															
功能说明	将寄存器 rs 的值与立即数 immediate 符号扩展后相加所得地址中的低两个字节取出来零扩展后保存在寄存器 rt 中																															