

二.

9.

表 1

指令	w_r_s	imm_s	rt_imm_s	wr_data_s	ALU_OP	Write_Reg	Mem_Write	IO_R	IO_W	PC_s
5)	00	-	0	00	0100	1	0	-	0	00
7)	01	1	1	00	0100	1	0	-	0	00
8)	-	1	0	-	0101	0	0	-	0	00/10
9)	-	-	-	-	-	0	0	-	0	11
10)	-	1	1	-	0100	0	1	-	0	00

10.

表 2

指令	功能描述
xori rt, rs, imm	逻辑异或: $rs \oplus imm \rightarrow rt$
In rt, [PortAr]	读入端口地址为 PortAr 的字存入 rt 中, 即: $(PortAr) \rightarrow rt$
Out [PortAr], rt	输出 rt 寄存器的内容到端口地址为 PortAr 的设备: $(rt) \rightarrow PortAr$

11.

表 3

指令	w_r_s	imm_s	rt_imm_s	wr_data_s	ALU_OP	Write_Reg	Mem_Write	IO_R	IO_W	PC_s
in	01	-	-	11	-	1	0	1	0	00
out	-	-	-	-	-	0	0	-	1	00

12.

- 图 2 所示的 MIPS 模型机的存储器是哈佛结构, 指令和数据存储于不同的存储器 (指令存储器和数据存储器) 中; 而冯·诺依曼体系结构的存储器, 则将指令和数据存储于同一个存储器中。
- 在图 2 所示的 MIPS 模型机结构中, 也体现了冯·诺依曼体系结构的特点:
 - 将指令和数据事先存放在存储器中, 并且采用二进制表示信息。
 - 采用存储程序和程序控制的思想; 由控制器控制从指令存储器中取出指令并执行, 在该 MIPS 单周期 CPU 中, 每次取指是取出一条指令。
 - 计算机硬件系统包括运算器、控制器、存储器、输入设备、输出设备五大部件。在图 2 所示的 MIPS 系统中, 运算 ALU 和寄存器堆以及标志位可以算作运算器部分; PC、PC_NEW、指令寄存器、译码及控制单元、用于地址计算的两个 ALU 等都可以算作控制器; 存储器采用哈佛结构, 分为指令

存储器和数据存储器。

13. (20) 使用频度高的指令分配较短的操作码，而频度低的指令分配较长的操作码

14. (21) 极间电容 (22) 慢 (23) 低

15. (24) 6400

16.

(25) 0000000 H~ 3FFFFFFH。

(26) 16

(27)

