

班级排序个人报告

八班 第六组 唐嘉良 2020K8009907032

1. 硬件设计

Table 1 A human sorter computer consisting of five components

Component	Implemented by	Functionality of the component
控制器(CU)	一名同学	负责翻译代码，根据各种指令产生信号或者对各个部件的操作进行控制
存储器 (Memory)	待排序的数据，每个数据由一位同学表示	负责存储数据（和指令，但是在本实验中不体现），在指令执行的时候运算器可以直接对存储器中的数据进行操作
运算器(ALU)	一名同学	负责进行算术和逻辑运算，执行一些比较操作，也可以执行一些算术运算操作
寄存器(REG)	五个寄存器，由一位同学控制	负责储存一些临时变量
标志寄存器 (LT)	一位同学	负责存储每次执行完 <code>compare</code> 指令之后会根据比较结果产生的信号参量

2. 指令集

Table 2 Instruction set of the student's human sorter computer

Opcode	Operand1	Operand2	Operand3	Explanatory remarks
Search_interval_left	Imm	Reg2	N/A	由运算器搜索 Imm 所指的地址开始第一个未蹲下的同学，并将地址号码存入 reg2 寄存器中，若找不到就存最后一个蹲下了的同学
Search_interval_right	Reg1	Reg2	N/A	搜索从 reg1 寄存器所指的地址开始往后的地址空间，直到遇到结尾或者遇到下一个蹲下的同学，并将下一个蹲下同学的前一个地址空间存入 reg2,或者一直没有同学蹲下的话，也将结尾的地址存入 reg2
Random_mark	Reg1	Reg2	Reg3	从 reg1 寄存器所指的地址到 reg2 寄存器所指的地址间随机产生一个地址存入 reg3 寄存器中,并让 reg3 寄存器所指的同学蹲下
MemExchange	Address1	Address2	N/A	将 Address1 所指的地址内容和 Address2 所指的地址内容做交换
Blez	Reg1	Address	N/A	如 reg1 寄存器里的值小于或等于零，则跳转到 address 指的地址位置

Cmp	Reg1	Reg2	N/A	比较寄存器 reg1 和寄存器 reg2 所指向的内存单元的内容，如果 reg1<reg2，将 LZ（less than）标志位设置为 1
Exchangelz	Reg1	Reg2	N/A	条件交换指令，如果 lz 标志位为 1，交换 reg1 和 reg2 所指的内存单元的值
Move	Imm1/Reg1	Reg2	N/A	将 Imm1 或者 reg1 的内容写到 reg2 中
Inc_lz	reg	N/A	N/A	lz 标志位为 1 就将 reg 的内容加一
Inc	reg	N/A	N/A	将 reg 寄存器的内容加一，再写回 reg 寄存器
Jump	Imm	N/A	N/A	无条件跳转，跳转执行 Imm 所指的指令
Sub	Reg1	Reg2	Reg3	将 reg1 的内容减去 reg2 的内容写入寄存器 reg3
Halt	N/A	N/A	N/A	停机，排序结束

3. 快速排序指令序列

Table 3 Assembly language code of the sort program executed by the human sorter computer

No.	Instruction	Explanatory Comments
1	Search_interval_left \$1, %v1	由运算器搜索地址 1 开始第一个未蹲下的同学，并将地址号码存入 v1 寄存器中，若找不到就存最后一个蹲下了的同学
2	Search_interval_right %v1, %v2	搜索从 v1 寄存器所指的空间开始往后的地址空间，直到遇到结尾或者遇到下一个蹲下的同学，并将下一个蹲下同学的前一个地址空间存入 v2,或者一直没有蹲下同学的话，也将结尾的地址存入 v2
3	Sub %v2 %v1 %v3	将 v2 寄存器减去 v1 寄存器里的值，并存入 v3 寄存器
4	Blez %v3,18	如果 v3 寄存器里的值小于或等于零，则跳转执行指令 18

5	Random_mark %v1,%v2,%v3	从 v1 寄存器所指的地址到 v2 寄存器所指的地址间随机产生一个地址存入 v3 寄存器中,并让 v3 寄存器所指的同学蹲下
6	MemExchange (%v3) (%v2)	将随机数交换到该区间的末尾
7	Move %v1,%v5	将 v5 寄存器存 left 值
8	Move %v1 %v4	将寄存器 v1 的值给 v4 寄存器，作为循环变量
9	Bge %v4,%v2, 16	当 v4 寄存器的值大于等于 v2 寄存器的值的时候，跳转执行指令 16
10	Move \$0,%lz	清零标志寄存器
11	Cmp (%v4), (%v2)	比较 v4 寄存器和 v2 寄存器所指向的内存单元的内容，如果 v4< v2 设置标志位为 1
12	Exchangelz (%v5), (%v4)	如果 lz 标志位为 1，交换 v5 和 v4 所指的内存单元的值
13	Inc_lz %v5	lz 标志位为 1 就将 v5 加一
14	Inc %v4	循环变量加一
15	Jump 9	跳转执行指令 9
16	MemExchange (%v5), (%v2)	将标杆到正确的位置
17	Jump 1	跳转执行指令 1
18	Halt	停机，排序完毕