1. 一个程序在一台计算机上运行时需要100秒，其中80秒的时间用于乘法操作，通过将乘法操作的速度改进到只需16秒，从而把程序的运行速度提高到5倍。这里改进性能所使用到的是哪个伟大设计思想 。

A.通过预测提高性能 B.通过流水线提高性能 C.加速大概率事件 D.通过并行提高性能

1. 对某一芯片只提高工作电压，则其功耗 ：

A. 提高； B. 下降；C. 不确定； D. 保持不变；

1. 下面的图表代表的是哪条MIPS指令? **（ ）**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| op | rs | rt | rd | shamt | funct |
| 0 | 8 | 9 | 10 | 0 | 34 |

A. sub $t0, $t1, $t2 B. add $t2, $t0, $t1

C. sub $t2, $t1, $t0 D. sub $t2, $t0, $t1

1. 假设$s1中的值是0xD0000000,给定下列$s0的值,执行下列指令是否会产生溢出？

**（ ）**（1）$s0=0x70000000, 执行add $s0,$s0,$s1, A.会溢出 B.不会溢出

**（ ）**（2）$s0=0x80000000, 执行sub $s0,$s0,$s1, A.会溢出 B.不会溢出

**（ ）**（3）$s0=0x7FFFFFFF, 执行sub $s0,$s0,$s1, A.会溢出 B.不会溢出

1. 假设$t0=0xBEADFEED, $t1= 0xDEADFADE. 求执行下面指令后寄存器$t2的值。
2. sll $t2, $t0, 4

or $t2, $t2, $t1 则$t2= **（ ）**

（2） sll $t2, $t0, 4

andi $t2, $t2, -1 则$t2= **（ ）**

（3） srl $t2, $t0, 3

andi $t2, $t2, 0xFFEF 则$t2= **（ ）**

1. 如果要将乘法指令结果的高32位保存在$t1，低32位保存在$t2中，需要使用两条指令来完成，它们是
2. 为了调用函数myfunc1，应该使用指令 **，**函数返回时应该使用指令返回到调用函数处的下一跳指令。
3. 如果指令“beq $t0,$t1, 32”指令位于0x1000地址，执行该指令时PC= **，**若$t0=14,$t1=23，则下一条被执行的指令位于：
4. 把下面的C代码翻译成MIPS代码。假设变量f,g,h,i和j分别赋值给寄存器$s0,$s1,$s2,$s3和$s4。假设数组A和B的基地址分别存放在$s6和$s7中。假设数组A和B中的元素均为4字节的字：

1）B[8]=A[i]+A[j];

2）f=g-A[B[4]];

1. 将以下C语句转换为MIPS汇编指令序列，假设变量a, b, i, j分别对应寄存器$s0, $s1, $t0, $t1, $s2保存着数组D的起始地址。

a）for ( i=0; i<10; i++)

a += b;

b) while (a<10) {

D[a] = b + a;

a += 1;

}

1. ~~课本1-7题~~
2. 把下面的MIPS代码翻译成C代码。假设变量f,g,h,i和j分别赋值给寄存器$s0,$s1,$s2,$s3和$s4，数组A和数组B的基地址分别存放在$s6和$s7中。

addi $t0,$s6,4

add $t1,$s6,$0

sw $t1,0($t0)

lw $t0,0($t0)

add $s0,$t1,$t0

1. 假定字变量f, g, h, i, j分别对应寄存器$s0, $s1, $s2, $s3, $s4, 并且字数组A和B的起始地址分别存放在寄存器$s6, $s7中，请分别采用MIPS汇编指令实现其功能：

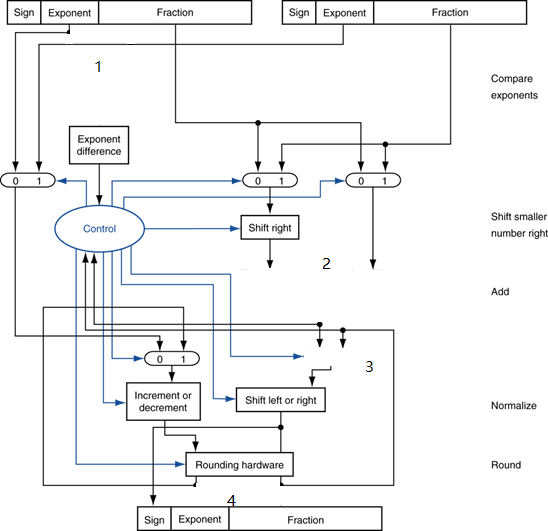
a) f=g+h+B[4]; b) f=g-A[B[4]];

c) f=g+h+B[1]; d) f=A[B[g]+1];

1. 画出课本第一版的乘法器结构图（含部件名称、位长、是否具有移位以及移位方向，部件间的连接），并填写0101\*1011的计算细节步骤。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 步骤 | 乘数 | 被乘数 | 乘积 |
| 1 |  |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 2 |  |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 3 |  |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |

1. 画出课本中改进的除法器结构图（含部件名称、位长、是否具有移位以及移位方向，部件间的连接）
2. 请补充完成下面的浮点加法器的结构图，将图中数字1~4的部位绘制完整。



1. 请将以下C代码转换成MIPS汇编，假设degree存放在$f12，结果存放到f0，常数3.0和27.0分别保存在gp+200和gp+208地址。

double fun (doube degree)

{ return 3.0\*(degree-27.0);