

分 数	
评卷人	

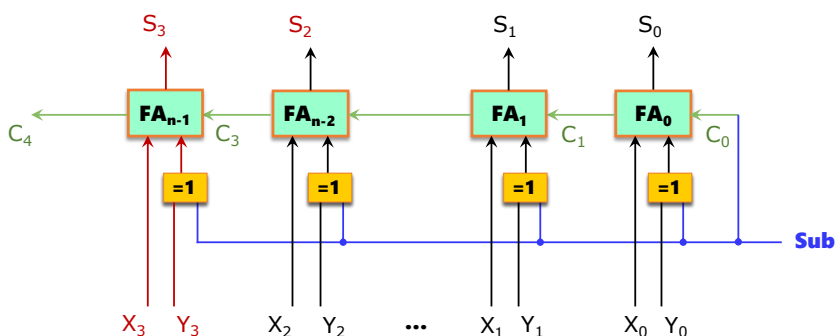
一、要基于一位全加器 FA 串联设计一个 4 位无符号补码可控加减法器，两个运算操作数分别为 $X = X_3X_2X_1X_0$, $Y = Y_3Y_2Y_1Y_0$ ，进位输入信号为 C_0 ，运算结果输出为 $S = S_3S_2S_1S_0$ ， C_4 为进位输出，运算控制信号为 Sub，试完成下列各题。（12 分）

1) 请设计一位全加器 FA 的电路，运算操作数为 X, Y，进位信号为 C_{in} ，输出为运算结果 S，进位输出 C_{out} ，给出所有输出信号逻辑表达式。

$$S_i = X \oplus Y \oplus C_{in} \quad (2 \text{ 分})$$

$$C_{out} = XY + (X \oplus Y)C_{in} \quad \text{或} \quad C_{out} = XY + (X + Y)C_{in} \quad (2 \text{ 分})$$

2) 以一位全加器 FA 为基础, 设计一个 4 位串行无符号补码加减法器, 请给出其电路图, 并说明其工作原理。(5 分)



3) 假设所有门电路时间延迟均为 $1T$, 则一位全加器 FA 的时延为 $3T$, (1 分) 该 4 位可控加减法器的关键路径延迟为 $13T$ 。(13T 给 1 分, 10T 给 2 分)

分 数	
评卷人	

二、某校验码编码长度 15 位，采用了海明码进行校验，编码左到右依次为 $H_5H_4H_{13}\dots H_1$ ，海明校验组采用偶校验，试完成下列各问。（14 分）

1) 根据海明校验的原理, 请用打钩的方式在下表中标记出 15 位海明码中的校验位。

H ₁₅	H ₁₄	H ₁₃	H ₁₂	H ₁₁	H ₁₀	H ₉	H ₈	H ₇	H ₆	H ₅	H ₄	H ₃	H ₂	H ₁
		(2 分)					✓				✓		✓	✓

2) 根据海明码定义, 该编码应该分为四组, 请给出每组中校验位的逻辑表达式。

[illegible]

(上表为编码设计辅助表格, 可以自行使用, 不做判分依据) (4分)

$$H1 = H3 \oplus H5 \oplus H7 \oplus H9 \oplus H11 \oplus H13 \oplus H15$$

$$H2 = H3 \oplus H6 \oplus H7 \oplus H10 \oplus H11 \oplus H14 \oplus H15$$

$$H4 = H5 \oplus H6 \oplus H7 \oplus H12 \oplus H13 \oplus H14 \oplus H15$$

$$H8 = H9 \oplus H10 \oplus H11 \oplus H12 \oplus H13 \oplus H14 \oplus H15$$

3) 假设指错字为 $G_4G_3G_2G_1$, 如果校验码最多只有一位错, 如何判断错误并纠正错误, 如校验码为 010101101101010, 请进行出错情况判断, 给出计算过程。

指错字值=0 表示没有错误, 否则表示出错位的位置, 只需将对应为取反即可纠错 (2分)

$$G1 = 0+0+0+1+1+0+0+0 = 0$$

$$G2 = 1+0+1+1+1+0+1+0 = 1$$

$$G3 = 1+0+1+1+1+0+1+0 = 1$$

$$G4 = 0+1+1+0+1+0+1+0 = 0$$

$$G_4G_3G_2G_1=6 \text{ 因此 } H6 \text{ 出错} \quad (2分)$$

4) 该编码纠错的前提是什么, 假设没有三位错, 如何识别一位错, 两位错?

假设只有一位错才能纠错 (1分)

可以引入总校验位, (1分)

指错字=1, 总检错位=1 时, 表示一位错, 否则表示两位错。 (2分)

分 数	
评卷人	

三、某计算机系统主存容量为 256B, 按字节编址, 其高速缓冲存储器 (Cache) 数据存储体的容量为 32B, 假定主存和 Cache 每个数据块大小均为 4B。
(16 分)

1) 若 Cache 采用 2 路组相联, 请给出主存地址向 Cache 地址映射时主存地址划分图, 分别给出标记字段 (Tag)、索引字段 (Index) 和块偏移字段 (Offset) 的位数。 (3 分)

Tag (4bits)	Index (2bits)	Offset (2bits)
-------------	---------------	----------------

2) 假定 Cache 采用 LRU 替换策略, 且 Cache 的初始内容为空; 画出 N=10 时, 执行下列代码后 Cache 各组各行中保存的数组数据情况 (按映射方法直接将 v[i] 写在 Cache 特定组的特定行, i 要用 0-9 中具体的值代替, 如 v[1] 等)。(注意: int 类型为 4 个字节, 假定代码执行时数组 V 被加载到主存地址 0 开始的连续存储器地址中, 变量 i, sum 编译时分配到寄存器中)

```
int sumv(int v[N])
{
    int i, sum=0;
    for (i=0; i<N; i++)
        sum += v[i];
    return sum;
}
```

(8 分)

组号	组内行号	内容
0	0	V[8]
	1	V[4]
1	0	V[9]
	1	V[5]
2	0	V[2]
	1	V[6]
3	0	V[3]
	1	V[7]

3) 结合 Cache 工作原理和存储体系构建的基本原理, 简要说明存储体系中设置 Cache 的目的是什么? 分析上述代码执行过程中 Cache 作用是否得到了发挥? 给出你认为能提高上述代码执行过程中 Cache 作用有效发挥的办法。

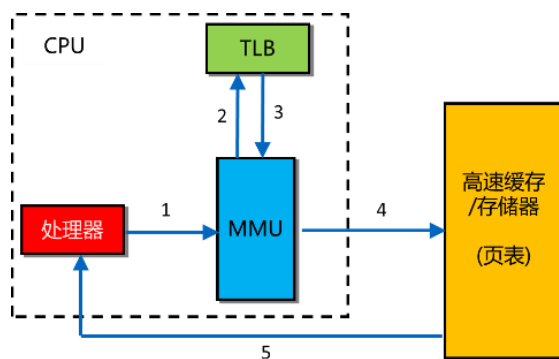
基于局部性原理, 提高 CPU 在 Cache 中访问数据的命中率来缓解 CPU 与主存间的速度差异, 从而提高存储系统的访问速率。 (2 分)

代码在执行过程, Cache 的作用没有发挥出来, 因为数据块大小刚好就是一个整数, 所构建的存储体系没有体现局部性的思想, 导致 CPU 对数据的访问没有一次能在 Cache 中命中。(1 分)

可行的办法: 提高数据块的大小 (2 分)

分 数	
评卷人	

四、下图为虚拟存储器的工作原理图。(14 分)



- 1) 页式虚拟存储器工作过程中涉及到 VA (虚拟地址)、PA (物理地址)、VPN (虚拟页号)、PPN (物理页号) 等概念。根据页式虚拟存储器的工作原理, 给出 VA、PA、VPN、PPN 在图中的编号。(4 分)

VA	1	VPN	2
PA	4	PPN	3

- 2) 如果不使用 TLB 会导致什么问题, 简要说明原因?

如果不使用 TLB, 会降低存储系统的访问速率 (或增加存储系统的访问时间), 因为实现虚拟地址与物理地址的转换需要增加一次访问主存/高速缓冲存储器。(2 分)

- 3) 假定某虚拟页式存储器页大小为 1024B, 物理空间为 64KB。结合下表求对应于十进制虚拟地址 2050 和 3080 的主存物理地址 (十进制)。(第一列为有效位, 1 表示有效) (4 分)

1	000010
1	000110
1	000111
0	000100

页表

VA (10 进制)	PA (10 进制)
2050	7170
3080	缺页

根据虚拟页式存储器页面大小 1024B, 可知页内偏移地址为 10 位;

$(2050)_{10} = (10\ 000000010)_B$, 对应的虚页号为 2, 查页表得到物理页号为 000111, 且有效位为 1, 因此可得到物理地址为: $(000111\ 0000000010)_B = 7170$

$(3080)_{10} = (11\ 0000001000)_B$, 对应的虚页号为 3, 查页表得该页有效位为 0, 因此本次访问实效, 本次访问不能获得与虚拟十进制地址 3080 对应的物理地址, 将发生缺页异常。

- 4) 访问 TLB 不命中时一定会发生缺页异常吗? 如果 TLB 命中, cache 是否一定命中? 简单分析原因。

TLB 不命中, 只是表示对应页表项不在 TLB 中, 只有访问主存系统中的页表项提示缺页时才会发生缺页异常; TLB 命中, 只能说明要访问的页在主存, 页载入主存和数据块载入 cache 并不同步, 所以 cache 有可能命中, 也有可能缺失。 4 分

分 数	
评卷人	

五、某计算机采用 16 位定长指令字格式,其 CPU 中有一个标志寄存器,其中包含进位/借位标志 CF、零标志 ZF 和符号标志 NF。假定为该机设计了条件转移指令,其格式如下:

15~11	10	9	8	07~00
00000	C	Z	N	OFFSET

其中, 00000 为操作码 OP; C、Z 和 N 分别为 CF、ZF 和 NF 的对应检测位, 某检测位为 1 时表示需检测对应标志, 需检测的标志位中只要有一个为 1 就转移, 否则不转移, 例如, 若 C=1, Z=0, N=1, 则需检测 CF 和 NF 的值, 当 CF=1 或 NF=1 时发生转移; OFFSET 是相对偏移量, 用补码表示。转移执行时, 转移目标地址为 (PC)+2+OFFSET×2; 顺序执行时, 下条指令地址为 (PC)+2。请回答下列问题。(11 分)

1) 该计算机存储器按字节编址还是按字编址? 该条件转移指令向前和向后最多可跳转多少条指令?

按字节编址 (2 分)

Offset 采用 8 位补码表示, 所以表示范围位-128~127, 可以向前跳跃 128, 向后跳跃 127 条指令 (1 分)

2) 某条件转移指令的地址为 200CH, 指令内容如下图所示, 若该指令执行时 CF=0, ZF=0, NF=1, 则该指令执行后 PC 的值是多少? 若该指令执行时 CF=1, ZF=0, NF=0, 则该指令执行后 PC 的值又是多少? 请给出计算过程。

15~11	10	9	8	07~00
00000	0	1	1	1 1 1 0 0 0 1 1

指令中 C = 0, Z = 1, N = 1, 故应根据 ZF 和 NF 的值来判断是否转移。当 CF=0, ZF=0, NF=1 时, 需转移。(1 分)

已知指令中偏移量为 1110 0011B=E3H, 符号扩展后为 FFE3 H, 左移一位 (乘 2) 后为 FFC6 H, 故 PC 的值 (即转移目标地址) 为 200CH+2+FFC6H=1FD4H。(2 分)

当 CF = 1, ZF = 0, NF = 0 时不转移。(1 分)

PC 的值为: 200CH+2=200EH。(1 分)

3) 实现“无符号数比较小于等于时转移”功能的指令中, C、Z 和 N 应各是什么?

C=Z=1, N=0 (3 分)

2) 执行指令阶段(5 分)

时钟	数据通路	控制信号
T1	$R[rs] \rightarrow X$	R_{out}, X_{in}
T2	$X - R[rt] \rightarrow PSW$	$R_{out}, Rs/Rt, SUB, PSW_{in}$
T3	$PC \rightarrow X$	PC_{out}, X_{in}
T4	$IR(A) + X \rightarrow Z$	$IR(A)_{out}, ADD$
T5	If (PSW.equal) $Z \rightarrow PC$	$Z_{out}, PC_{in} = PSW.equal$

- 2) 常见 MIPS 指令实现中是没有程序状态寄存器 PSW 的，为什么在本题中的 CPU 架构中需要设计程序状态字 PSW？

beq 指令需要分别计算比较结果和分支地址，本题的主机架构中只有一个运算器，所以需要两次使用运算器，因此必须暂存比较结果于 PSW 中。 (3 分)

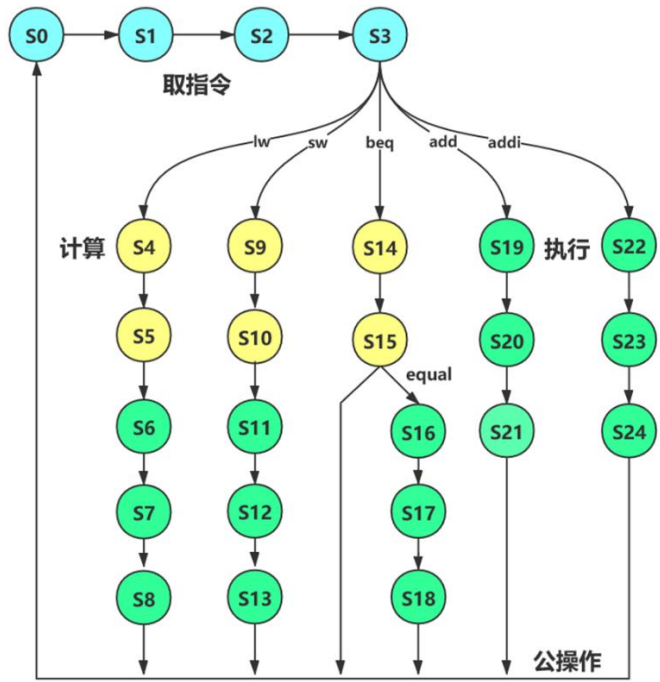
- 3) 在定长指令周期三级时序单总线 CPU 实验中，测试程序预期功能是在 0x80 的内存数据单元进行排序，请问这个排序是降序还是升序，是有符号比较还是无符号比较？为什么实际 Educoder 平台上通关的结果是在 0x00 处进行内存单元数据排序的，而且代码区部分代码会被覆盖？

降序排序，有符号比较 (2 分)

Z 寄存器没有锁存控制，才用定长周期时，刚刚计算完地址应该直接送 AR，但由于计算周期和执行周期之间插入了空周期，所以导致送入 AR 中的地址错误。 (1 分)

分 数	
评卷人	

七、对于上题中的 CPU 数据通路，如果采用现代时序方式，其指令执行状态图如下所示，如果采用微程序方式实现控制器，尝试回答如下问题。(17 分)



1) 如果采用下址字段法，直接表示的水平微指令，则微指令操作控制字段长度为 21/22 位，判别测试位至少 3 位，下址字段至少为 5 位，实现上题中五条机器指令共需要多少条 25 微指令。如果改用计数器法，则判别测试位至少 3 位。如果采用对微指令字中的操作控制字段采用编码方式缩短字长，则该字段最短为 18 位。

(最后一空 2 分，其他各 1 分，共 7 分)

2) 如果要为该 CPU 增加单级中断处理机制，需要增加哪些硬件单元，简要叙述增加的硬件单元的功能。

- EPC 保存断点
- 中断使能寄存器 IE 开关中断
- 中断识别控制逻辑 中断识别 (6 分)

