

71	2	3	4	5	6	7	8	9	10	11	12
B	A	D	CD	BCD	A	BCD	B	ACD	D	ABCD	C

- 1、 加法器采用先行进位的目的是（ ）  
A.优化加法器结构                      **B.提高进位产生速度**  
C.保证运算结果正确                      D.正确传递进位值
- 2、 十进制数 178.125 表示成 IEEE 754 单精度浮点数的结果是（ ）  
**A.43322000H**    B.43591000H    C.B3591000H    D.B3322000H
- 3、 指令所需要的操作数不会来自（ ）  
A.指令寄存器 IR    B.主存  
C.通用寄存器                      **D.变址寄存器**
- 4、 **[多选]**下列计算机系统性能评价的描述中正确的是  
A.程序 MIPS 值越高，计算机的性能越高  
B.程序的 CPI 值越低，计算机的性能越高  
**C.主频高的机器性能不一定高**  
**D.同一程序在不同机器的 MIPS 不一定相同**
- 5、 **[多选]**下列关于取指令阶段指令流的描述中，正确的是(            )  
A.取指流程中只有一条数据通路；  
**B.取指流程包含取指令和 PC 增量流程**  
**C.不同 PC 增量方式影响取指流程；**  
**D.CPU 内总线结构影响取指流程；**
- 6、 某机有 5 级中断 L4~L0，中断响应优先级从高到低的次序是 L4>L3>L2>L1>L0，设中断屏蔽字为 M4M3M2M1M0， $M_i=1$  ( $0 \leq i \leq 4$ ) 表示对 Li 级中断进行屏蔽。若要实现中断完成（处理）优先级从高到低的顺序 L4>L1>L3>L2>L0，则在 L3 的中断服务程序中，应把中断屏蔽字设置为(            )

A.01101    B.10010    C.11000    D. 00111

- 7、 **[多选]**下列关于 Cache 的说法中，错误的是
- A.采用直接映像时，Cache 无需考虑替换问题
  - B.如果选用最优替换算法，则 Cache 的命中率可达到 100%
  - C.提高 Cache 本身速度比提高 Cache 命中率更有利于提升存储器的等效访问速度
  - D.Cache 的容量与主存的容量差别越大，存储系统的等效访问速度越高
- 8、 在补码定点加减法运算的溢出判别中，下列说法错误的是(        )
- A.对于减法，符号不同的两个数相减可能发生溢出
  - B.对于加法，符号相同的两个数相加一定发生溢出
  - C.对于加法，符号不同的两个数相加永不会发生溢出
  - D.对于减法，符号相同的两个数相减永不会发生溢出
- 9、 **[多选]**单周期 MIPS 在一个时钟周期中可能完成的操作 (        )
- A. ALU 运算和向寄存器堆写数据
  - B.从数据存储器读数据和向数据存储器写数据；
  - C.更新 PC 内容和向数据存储器写数据
  - D.寄存器堆读数据，ALU 运算和数据存储器写数据
- 10、 某型 MIPS32 指令架构的单周期 CPU，其数据通路结构如下图：



得分	评卷人

## 二、分析问答题（12 分）

在 CRC 编码实验中，待传输 16 位数据位 = 0000 0000 0000 0111，采用 CRC 循环冗余校验码进行数据校验，生成多项式为 1101111。

- 1) 试根据 CRC 校验码的编码规则给出该编码的 CRC 余数。

$$111000000 \bmod 1101111 = 010011$$

- 2) 假设接收方接收到的最终编码为 0000 0000 0000 0111 0000 11，假设最多发生一位错，最低位为第 1 位，结合课程实验中所用到的方法说明 CRC 编码如何定位错误并纠正错误。

$$111000011 \bmod 1101111 = 010000 \quad \text{发生错误}$$

根据余数应该右起第 5 位错

- 3) 课程实验中并行 CRC 编码电路的基本思路是什么？

提前计算若干编码的余数，然后将编码数据拆解成若干编码的排列组合，将余数异或加即可得到余数。

- 4) 在 CRC 编码流水传输中是如何区分一位错还是两位错的？对于两位错，流水线是如何处理的？

1、可以引入偶校验位，也可以直接利用 CRC 编码余数进行判断。。

2、对于两位错，要求清除两个接口的数据，并在发送端重新回滚。

得分	评卷人

### 三、分析计算题（12 分）

计算分析题（10 分）已知 $[x]_{\text{补}}=11111011$ ， $[y]_{\text{补}}=01010111$ ，用补码一位乘法计算 $[x \times y]_{\text{补}}=?$ （单符号位）将答案填写在下面，并将计算过程填写在表格中。

$$[x \times y]_{\text{补}} = \underline{7E4D} \quad (\text{十六进制}) \quad x \times y = \underline{-435} \quad (\text{10 进制})$$

如果 $[x]_{\text{补}}=1000000$  时， $[y]_{\text{补}}=00000001$ ，运算结果等于多少？结果是否正常，为什么？

=80，结果不正常，因为 $-x$  溢出了，无法进行运算。

$$[-x]_{\text{补}}=00000101$$

#	运算	部分积	移出位	判断位 $y_n y_{n+1}$
1		00000000		<u>010101110</u>
2	$+ [-x]_{\text{补}}$	00000101		
3	=	00000101		
4	$\rightarrow$	00000010	1	<u>01010111</u>
5	+0	0		
6	=	00000010	1	
7	$\rightarrow$	00000001	01	<u>0101011</u>
8	+0	0		
9	=	00000001	01	
10	$\rightarrow$	00000000	101	<u>010101</u>
11	$+ [x]_{\text{补}}$	11111011		
12	=	11111011	101	
13	$\rightarrow$	11111101	1101	<u>01010</u>
14	$+ [-x]_{\text{补}}$	00000101		
15	=	00000010	1101	
16	$\rightarrow$	00000001	01101	<u>0101</u>
17	$+ [x]_{\text{补}}$	11111011		
18	=	11111100	01101	
19	$\rightarrow$	11111110	001101	<u>010</u>
20	$+ [-x]_{\text{补}}$	00000101		
21	=	00000011	001101	
22	$\rightarrow$	00000001	1001101	<u>01</u>
23	$+ [x]_{\text{补}}$	11111011		
24	=	11111100	1001101	
25				

得分	评卷人

#### 四、综合分析题（12 分）

一个包含 10 个元素的一维数组顺序存放在主存连续单元，主存每个存储单元存放一个数组元素，机器的 Cache 分为指令 Cache 和数据 Cache，其中数据 Cache 有 4 行，每行可存放两个数组元素，Cache 的初始状态为空，采用 LRU 替换算法，某程序的伪代码如下：

```
SUM :=0
for j:=0 to 9 do
  SUM:= SUM +A(j)
end
Ave:= SUM/8
for i:=9 to 0 do
  A(i) := A(i)/Ave
end
```

- 1) 将 Cache 分为指令 Cache 和数据 Cache 的架构称为哈佛结构，这种结构有什么好处？
- 2) 计算全相联映射方式下 Cache 读操作的命中率。
- 3) 假定主存的存取时间为 200ns，Cache 的存取时间为 20 ns，求该存储系统的平均存取时间。

参考答案：

- 1) 将 Cache 分为指令 Cache 和数据 Cache 时,CPU 可并行访问指令和数据，缩短程序的执行时间。（2 分）
- 2) 全相联系方式下 Cache 中的内容变化如下图所示  
a)第一轮循环读命中情况

	j=0-7	命中	j=8,9	命中
C0	a[0],a[1]	a[1]	a[8],a[9]	a[9]
C1	a[2],a[3]	a[3]	a[2],a[3]	
C2	a[4],a[5]	a[5]	a[4],a[5]	
C3	a[6],a[7]	a[7]	a[6],a[7]	

b)第二轮循环命中的情况

循环前	j=9-2	命中	j=1, 0	命中
a[8],a[9]	a[8],a[9]	a[8],a[9]	a[1],a[0]	a[0]
a[2],a[3]	a[2],a[3]	a[3],a[2]	a[2],a[3]	
a[4],a[5]	a[4],a[5]	a[5],a[4]	a[4],a[5]	
a[6],a[7]	a[6],a[7]	a[7],a[6]	a[6],a[7]	

两轮循环中数组中的元素共被访问 20 次，两轮循环中从数据 Cache 中共命中 14 次，命中率为  $14/20 = 70\%$

3)  $T = 20 \text{ ns} \times 0.7 + 200 \text{ ns} \times (1-0.7) = 74 \text{ ns}$  （4 分）

得分	评卷人

## 五、设计分析题（12 分）

某计算机数据线 32 位，地址线 18 位，按字节编址。主存 字 地址空间分配如下：0000H - 3FFFH 为 ROM 区，4000H - FFFFH 为 RAM 区，现要求用 8K×8 位的 RAM 和 ROM 芯片构成上述 32 位的存储器。完成下列问题：

（1） 计算所需要的 RAM 和 ROM 芯片数量；

（1） ROM 有 16K 地址，需要  $\frac{16}{8} \times \frac{32}{8} = 8$  片 8K×8 位的 ROM

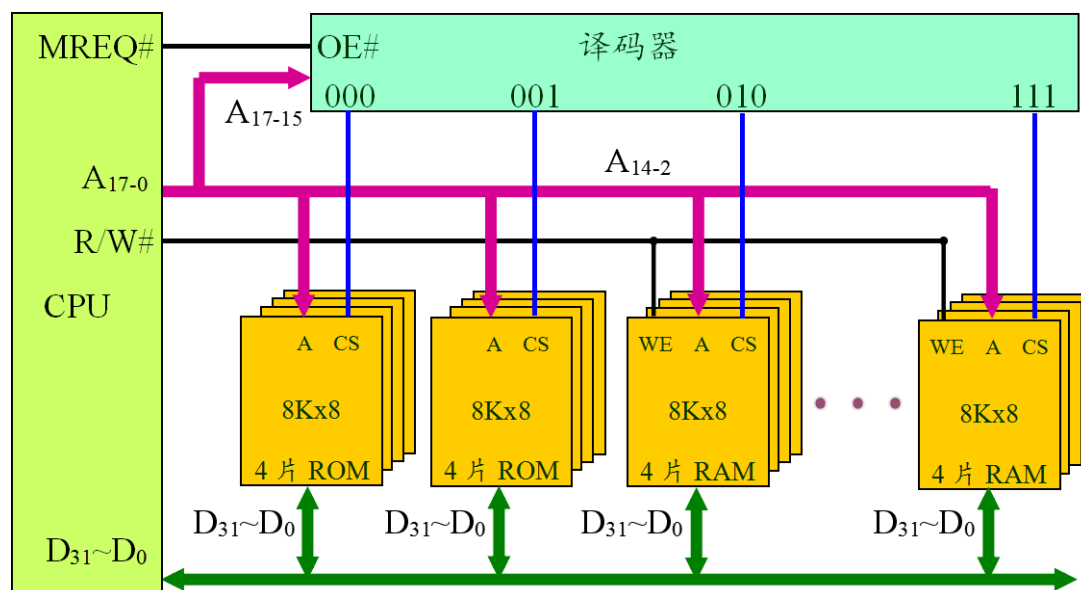
RAM 有 48K 地址，需要  $\frac{48}{8} \times \frac{32}{8} = 24$  片 8K×8 位的 RAM

（2） 如何保证存储器按字边界对齐？

该 32 位存储器总共有 64K 地址，需要 16 根地址线访问，为了保证存储器按字边界对齐可以采用以下方法：即计算机的 18 位地址线高 16 位连接存储器，低 2 位地址则不连接存储器，而是用于字节访问。

（3） 画出 CPU 与存储器的连接，注意片选信号和不同芯片的顺序。

CPU 与存储器的连接如下图所示：



得分	评卷人

## 六、设计分析题（12 分）

现针对云计算开发了一款64位的MIPS指令集的变体，为了更适应大数据场景，操作码Opcode字段为7位，通用寄存器的数目由32个变成了128个，寄存器数据位宽64位，假设R型指令中64位字长的指令字多余位均扩展到最右侧的Funct字段，I型指令中多余位均扩展到Immediate字段。

（1） 请给出R型指令和I型指令字段划分。

Opcode	Rs	Rt	Rd	Shamt	Funct
7	7	7	7	6	30

Opcode	Rs	Rt	Imm
7	7	7	43

（2） 该MIPS指令系统最多可以支持多少条指令，给出计算表达式。

$$2^7 - 1 \text{ 非R型号指令} + 2^{30} \text{ R型指令}$$

（3） 执行一条R型指令后，PC的增量是多少，给出计算表达式；

$$\$pc = \$pc + 8;$$

（4） 请问有条件分支指令Beq最多能将PC寄存器的值增加多少？给出计算表达式。

$$\begin{aligned} &\text{Imm字段位宽 } 43\text{位, 补码最大整数为 } 2^{42} - 1, \text{ 跳转指令数最大为 } (2^{42} - 1) \ll 3 \\ &\$pc = \$pc + 8 + (2^{42} - 1) \ll 3 = \$pc + 2^{45} \end{aligned}$$



得分	评卷人

## 七、工程设计题（16 分）

在 CPU 设计实验中我们实现了如下 8 条 MIPS 指令的多周期 CPU，指令功能描述如下表所示，后页给出了主机数据通路图。

#	MIPS 指令	RTL 功能描述
1	add \$rd,\$rs,\$rt	$R[\$rd] \leftarrow R[\$rs] + R[\$rt]$
2	slt \$rd,\$rs,\$rt	$R[\$rd] \leftarrow R[\$rs] < R[\$rt]$ 小于置 1，有符号比较
3	addi \$rt,\$rs,imm	$R[\$rt] \leftarrow R[\$rs] + \text{SignExt}_{16b}(\text{imm})$
4	lw \$rt,imm(\$rs)	$R[\$rt] \leftarrow \text{Mem}_{4B}(R[\$rs] + \text{SignExt}_{16b}(\text{imm}))$
5	sw \$rt,imm(\$rs)	$\text{Mem}_{4B}(R[\$rs] + \text{SignExt}_{16b}(\text{imm})) \leftarrow R[\$rt]$
6	beq \$rs,\$rt,imm	if( $R[\$rs] = R[\$rt]$ ) $PC \leftarrow PC + \text{SignExt}_{18b}(\{\text{imm}, 00\})$
7	bne \$rs,\$rt,imm	if( $R[\$rs] \neq R[\$rt]$ ) $PC \leftarrow PC + \text{SignExt}_{18b}(\{\text{imm}, 00\})$
8	syscall	系统调用，这里用于停机

- 1) 根据主机数据通路图的信息请给出 sw 指令在取指令阶段和执行指令阶段的数据通路和控制信号。

### 1) 取指令阶段

时钟	数据通路	控制信号
T1	<b>Mem[PC]→IR</b> <b>PC+4→PC</b>	<b>IRwrite=1 IorD=0 PcWrite=1 MemRead</b> <b>ALU_OP ALU_srcA ALU_SrcB</b>
T2	<b>Reg→A、B</b> <b>PC+4+IMM16&lt;&lt;2→C</b>	<b>ALU_OP ALU_srcA ALU_SrcB</b>
T3		
T4		

### 2) 执行指令阶段

时钟	数据通路	控制信号
T1	<b>A+B→C</b>	<b>ALU_OP ALU_srcA ALU_SrcB</b>
T2	<b>B→Mem[C]</b>	<b>IorD=1 MemWrite</b>
T3		
T4		

- 2) 在 Logisim 环境中支持 Ctrl+R 进行系统总复位，为什么电路中还要增加一个 Rst 复位信号？

Ctrl+R 复位会清空 RAM 存储器，导致程序丢失，增加的 Rst 信号可避免 RAM 数据丢失

- 3) 如果采用微程序构造控制器，采用水平型直接表示法，微指令分为哪几个字段，各字段长度多少，各字段包括哪些信息，

操作控制字段 17 位

P 字段 1 位

下址字段 4 位

- 4) 微程序通常是串行执行的，简要叙述取指微程序执行完毕后系统是如何跳转到当前指令对应的微程序入口地址的？

P 字段为 0 时，下址字段提供下条微指令地址，P 字段为 1 时，由地址转移逻辑根据指令功能给出下条微指令地址。









