

一、(15分) IEEE 754 标准包括半精度浮点数, 它使用 16 位来存储, 由 1 位符号位、5 位阶码 (偏移量为 15)、以及 10 位尾数构成 (含隐藏位), 其余格式规范与单精度和双精度浮点数相同。已知两个半精度浮点数的二进制表示, $X = 0111\ 1000\ 1100\ 1101$, $Y = 1111\ 0110\ 1010\ 1001$, 回答以下问题。

- 1) X 的阶码十进制值为_____, 尾数真值的二进制表示为_____;
Y 的阶码十进制值为_____, 尾数真值的二进制表示为_____。
- 2) 规格化的半精度浮点数的绝对值最小数为_____, 非规格化的半精度浮点数的绝对值最小数为_____。
- 3) 计算 $X - Y$, 按照 IEEE 754 标准运算结果为_____ (16 进制), 给出关键计算过程, 采用两位保留附加位和 0 舍 1 入法。最终运算结果_____ (溢出/不溢出)。

$X = 0111\ 1000\ 1100\ 1101$, 阶码十进制值为 15, 尾数二进制真值为 +1.00 1100 1101;
 $Y = 1111\ 0110\ 1010\ 1001$, 阶码十进制值为 14, 尾数二进制真值为 -1. 10 1010 1001。 (4 分)

规格化的半精度浮点数的绝对值最小数为 2^{-14} , 非规格化的为 2^{-24} 。 (4 分)

计算 $X - Y$ 需要先对阶, 将 Y 右归一位, 阶码值为 15, 尾数真值为 -0.11 0101 0100 1

$$\begin{array}{r} 1.00\ 1100\ 1101\ 00 \\ +\ 0.11\ 0101\ 0100\ 10 \\ \hline = 10.00\ 0010\ 0001\ 10 \end{array}$$

运算结果右归一位, 阶码值加 1 = 1.000 0010 000 11

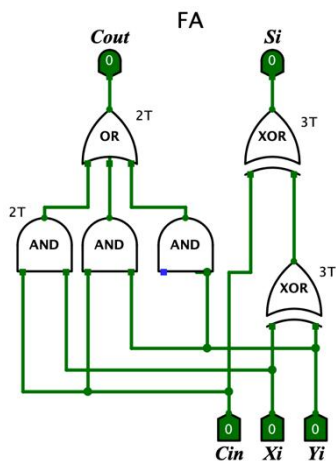
0 舍 1 入 = 1.000 0010 001

阶码十进制值为 $15+1 = 16$, 移码二进制表示为 11111, 运算上溢 (无穷大)。 (7 分)

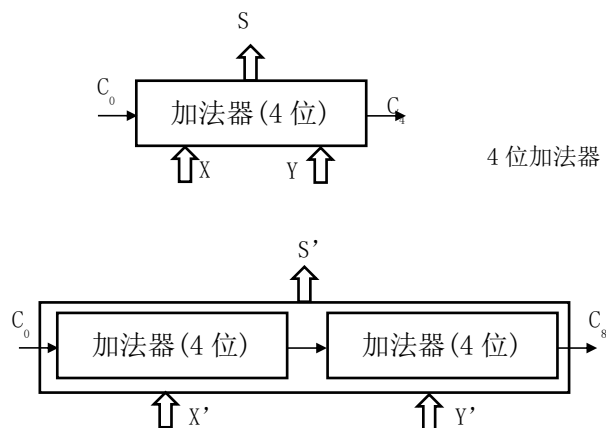
最终结果 0 11111 000 0010 001=7C11H

分 数	
评卷人	

二、(14分) 用如图所示的 1 位全加器 FA (左图) 串联构建了一个 4 位加法器 (右上)。该 4 位加法器输入分别为被加数 $X = X_4 \sim X_1$, 加数 $Y = Y_4 \sim Y_1$, 进位输入 C_0 , C_4 为最高位进位, 输出 $S = S_4 \sim S_1$, 第 4 位为最高位。请完成下列各题。



1 位全加器 FA



8 位加法器

1) 1 位全加器 FA 内部采用如左图结构所示。图中，有一个与门输入未给出，请在图上补充连接完成，并给出 C_{out} 的表达式 $XY + XC_{in} + YC_{in}$ (2 分)

2) 假设 OR、AND 门电路时间延迟均为 $2T$ ，XOR 门电路延迟为 $3T$ ，则该一位全加器的关键时延为 $6T$ 。(2 分)

3) 请用文字说明解释一下 4 位串行加法器中关键时延路径是什么？(2 分)

4 个一位全加器通过进位位产生逻辑先后关联（后一个全加器必须等待前一个全加器给出进位位后才能进一步通过一个 AND 和 OR 逻辑门生成正确进位输出）。显然，这条沿着进位输出的路径是时延最长的路径也即关键实验路径，具体为： $AND + OR + AND + OR + AND + OR + AND + OR$

4) 上述 4 位串行加法器关键路径的具体时延为 $16T$ 。(2 分)

5) 上述 4 位串行加法器，如果采用单符号补码，则 S 最大时机器码为 7 (16 进制)，S 最小时机器码为 8 (16 进制)，S 为最小数时其真值为（视为纯小数） -1 。(3 分)

6) 将上述 2 个 4 位加法器串联为一个 8 位的加法器，被加数、加数与和数分别为 X' 、 Y' 和 S' 。采用双符号补码，如 $X' = -0.110111$ ， $Y' = -0.101100$ ，和数 S' 的机器码为 _____ (16 进制)。请写出简要的计算过程。 S' 是否溢出？给出判断理由。(3 分)

11.001001
 $+ 11.010100$

 110.011101 S' 溢出，因为双符号位相异。

分 数	
评卷人	

三、(12分) 某计算机主存容量要求为 128K 个字，按字编址，其地址空间分为三个部分：00000 H ~ 03FFF H 是 ROM 区域，0C000 H ~ 0FFFF H 是保留区域，10000 H ~ 1FFFF H 是 RAM 区域。现有多个 16K × 4 位的 ROM 芯片和 RAM 芯片，它们的存储周期为 $T=50\text{ns}$ 。要求为该计算机设计一个数据传输带宽不低于 80MB/s 的主存储器，请回答以下问题。

- 1) 该主存的字长最少需要 32 位，地址线 17 位，共需要 8 片 ROM 芯片和 32 片 RAM 芯片，才能满足存储容量和数据传输带宽需求。

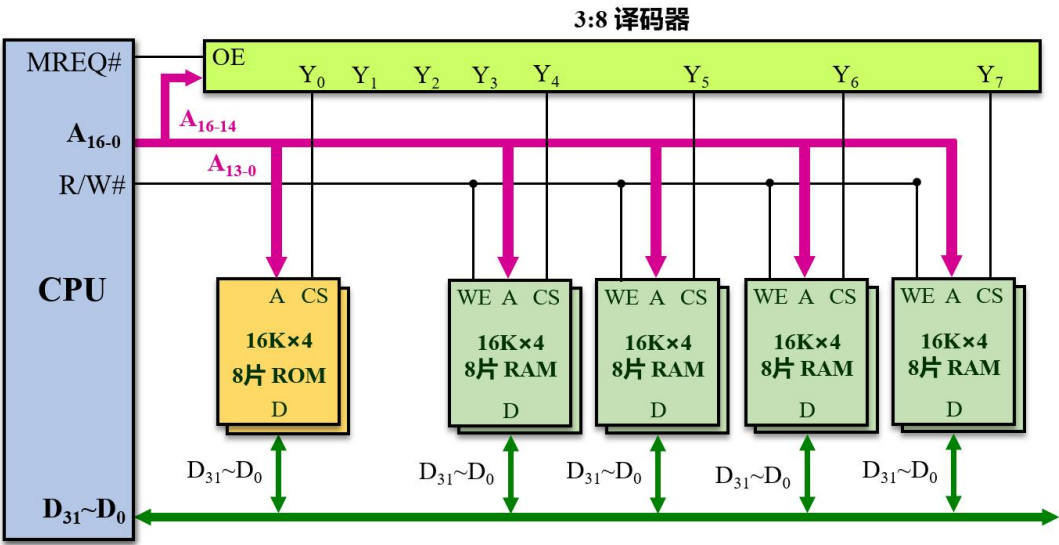
每个存储芯片的位宽为 4 bit，存储周期为 50ns，那么存储带宽为 10MB/s。要求总带宽为 80MB/s，因此应该字长扩展至 32 位。

每个 ROM/RAM 芯片都是 32K 个字，ROM 容量需求为 16K，RAM 容量需求为 64K，因此分别需要 8 片 ROM 和 32 片 RAM。

控制信号线需要 2 位，地址线需要 17 位，数据线需要 32 位，因此共需 51 个引脚。(6 分)

- 2) 请将 CPU 与存储芯片的连接电路示意图补充完整，图中译码器为 3:8 译码器，请标注译码线。

第二版答案



分 数	
评卷人	

四、(16分) 某 MIPS32 处理器，内存地址空间大小为 128MB，采用页式虚拟存储管理方式，按字节编址，虚拟地址为 32 位，页面大小 4KB；具有分离的指令 Cache 和数据 Cache，容量都是 32KB，主存块大小 64B，均采用 4 路组相联、LRU 替换策略和写穿 (Write Through) 写策略。开始时 Cache

均为空，请问：

(1) 指令 Cache 包括 512 cache 行，分为 128 组。

(2) 数据 Cache 标记字段(Tag)至少需要 14 位，组号(Index)需要 7 位，块内地址(Offset)需要 6 位。

(3) TLB 中虚拟页号 (VPN) 是 20 位，物理页号 (PPN) 是 15 位，页内地址 (Offset) 是 12 位。

(4) 相对于虚拟存储器，Cache 缺失带来的处理开销 小，而处理缺页的开销 大，原因是 缺页处理需要访问磁盘，而 Cache 缺失只要访问主存，主存的访问速度要比硬盘快很多。

(5) 有下列 C 程序：

```
for (i = 0; i < 1024, i++)
    s[i] = 2 * s[i];
```

若数组 s 及其变量 i 均为 int 型 (int 型数据为 32 位)，变量 i 分配在寄存器中，并且数组 s 已经调入主存，s[0] 已经放在 8195 物理页上，为了尽可能的减少缺失，则数组 s 在主存中的起始地址为 0x200 3000 (16 进制)，s[1025]会在数据 Cache 的第 0 组，该程序运行期间会进行 2048 次数据访问，产生 64 次数据缓存缺失，数据 Cache 的命中率为 96.875 %。

分 数	
评卷人	

五、（12分）某 16 位计算机的主存按字节编址，存取单位为 16 位。采用 16 位定长指令字格式，指令集分为 R 型、I 型和 J 型三类指令。其中 R 型指令格式如下图所示，图中 Md、Msl、Ms2 为寻址方式位，支持寄存器直接和寄存器间接两种寻址方式（0 表示寄存器直接寻址，1 表示寄存器间接寻址）；Rd、Rsl、Rs2 为寄存器编号，通用寄存器为 R0-R3（编号分别为 0、1、2、3）。

指令操作码	目的操作数		源操作数 1		源操作数 2	
OP	Md	Rd	Msl	Rsl	Ms2	Rs2

- （1）该指令系统最多可定义 128 条 R 型指令。
- （2）假设 sub 指令的操作码为 0x3，则“sub R3, (R1), R2”的机器码为 0000011 0 11 1 01 0 10 或 0x06EA（16 进制）。
- （3）Store 指令属于 I 型指令，假设 Store 指令的操作码长度不变，源操作数固定为寄存器寻址，目的操作数固定为变址寻址方式；若变址寄存器可使用任一通用寄存器，且偏移量用补码表示，则 Store 指令中偏移量的取值范围是 -16~15。
- （4）转移指令为一地址指令，采用相对寻址。假定取指令时，每取一字节 PC 自动加 1。若某转移指令所在主存地址为 0x2000，相对位移量字段的内容为 0x06，则该转移指令成功转移后的目标地址是 0x 2008。
- （5）该机器的指令系统属于 RISC 指令集，原因是 指令长度固定、指令格式种类少、寻址方式种类少等。或者：该机器的指令系统属于 CISC 指令集，原因是 除 Load/Store 指令外其他指令也可以访存，通用寄存器数量太少等。

分 数	
评卷人	

六、(20分) 某计算机字长 32 位，支持下表中的五条 MIPS32 指令，CPU 内部采用单总线结构，具体数据通路如图所示。除多路选择器选择控制信号外，图中所有控制信号为 1 时表示有效、为 0 时表示无效，控制信号功能说明见表。

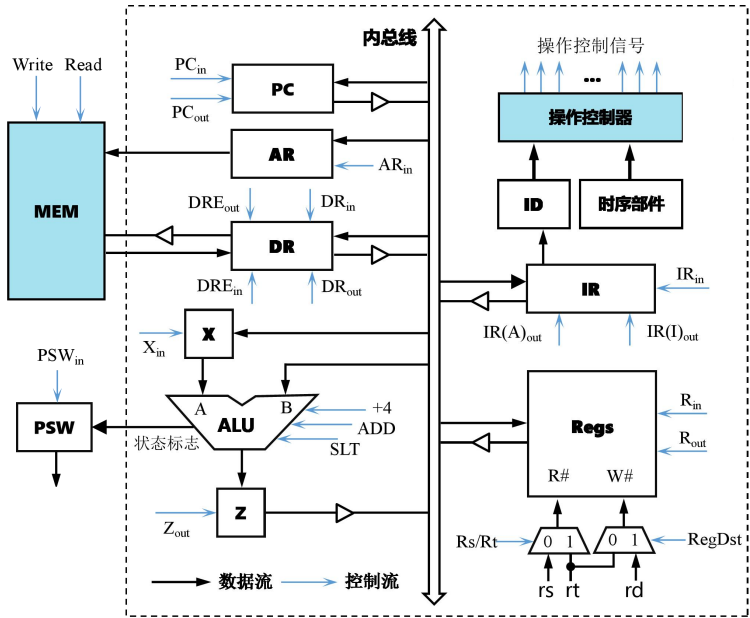


表 1、指令功能描述

#	指令	汇编代码	指令类型	RTL 功能说明
1	lw	lw rt,imm(rs)	I 型	$R[rd] \leftarrow M[R[rs] + \text{SignExt}(imm)]$
2	sw	sw rt,imm(rs)	I 型	$M[R[rs] + \text{SignExt}(imm)] \leftarrow R[rt]$
3	beq	beq rs,rt,imm	I 型	$\text{if}(R[rs] == R[rt]) \quad PC \leftarrow PC + 4 + \text{SignExt}(imm) << 2$
4	addi	addi rt,rs,imm	I 型	$R[rt] \leftarrow R[rs] + \text{SignExt}(imm)$
5	add	add rd,rs,rt	R 型	$R[rd] \leftarrow R[rs] + R[rt]$

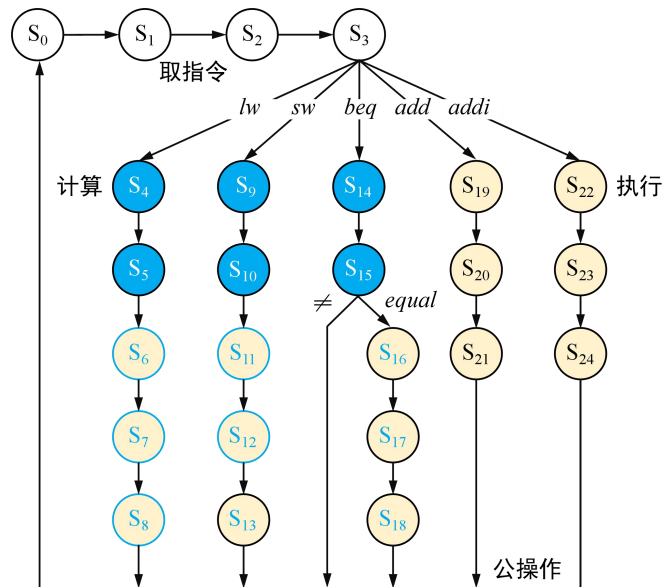
表 2、控制信号功能描述

#	控制信号	功能说明
1	PCin	控制 PC 接收来自内总线的的数据，需配合时钟控制
2	PCout	控制 PC 向内总线输出数据
3	ARin	控制 AR 接收来自内总线的的数据，需配合时钟控制
4	DRin	控制 DR 接收来自内总线的的数据，需配合时钟控制
5	DRout	控制 DR 向内总线输出数据
6	DREin	控制 DR 接收从主存读出的数据，需配合时钟控制
7	DREout	控制 DR 向主存输出数据，以便最后将该数据写入主存
8	Xin	控制暂存寄存器 X 接收来自内总线的的数据，需配合时钟控制
9	+4	将 ALU A 端口的数据加 4 输出
10	ADD	控制 ALU 执行加法，实现 A 端口和 B 端口的两数相加
11	SUB	控制 ALU 执行减法运算
12	PSWin	控制状态寄存器 PSW 接收 ALU 的运算状态，需配合时钟控制
13	Zout	控制 Z 向内总线输出数据
14	IRin	控制 IR 接收来自内总线的指令，需配合时钟控制
15	IR(A)out	控制 IR 中的分支目标地址输出到内总线，指令字中的立即数要转换成目标地址需要相应逻辑

#	控制信号	功能说明
16	IR(I)out	控制 IR 中的立即数输出到内部总线，指令字中的立即数符号扩展为 32 位才能输出
17	Write	存储器写命令，需配合时钟控制
18	Read	存储器读命令
19	Rin	控制寄存器堆接收来自内总线的的数据，写入 W#端口对应的寄存器中，需配合时钟控制
20	Rout	控制寄存器堆输出指定编号 R#寄存器的数据，该寄存器组为单端口输出
21	Rs/Rt	控制多路选择器选择送入 R#的寄存器编号，为 0 时送入指令字中 rs 字段，为 1 时送入 rt
22	RegDst	控制多路选择器选择送入 W#的寄存器编号，为 0 时送入指令字中的 rt 字段，为 1 时送入 rd

当 ALU 两操作数相等时，equal 状态信号输出为 1，否则为零，equal 信号与运算无关。

- 1) 单总线结构中 PSW 的输出应输出到图中 **操作控制器** 部件，ID 部件的功能是 **指令译码**。如果采用定长指令周期设计三级时序硬布线控制器，一个指令周期应包含 **3** 个机器周期，每个机器周期包含 **4** 个时钟节拍。如果采用变长指令周期设计三级时序设计硬布线控制器，最少需要 **2** 个机器周期，最多需要 **3** 个机器周期，每个机器周期最短需要 **2** 个时钟节拍。(7 分)
- 2) 如果采用现代时序设计硬布线控制器，CPU 状态机如下图所示，假设状态寄存器输出为 $Q_4Q_3Q_2Q_1Q_0$ ，请给出硬布线控制器 ADD 信号将在哪几个状态产生，给出逻辑表达式。(2 分)



$$ADD = S_5 + S_{10} + S_{17} + S_{20} + S_{23}$$

3) 采用微程序设计操作控制器，计数器法，微指令格式如下表所示，左侧为最高位，请给出取指微指令微程序已经给出，请用 16 进制完成如下填空。（7 分）

		<div><div><div>PC_{out}</div><div>DR_{out}</div><div>Z_{out}</div><div>R_{out}</div><div>IR(I)_{out}</div><div>IR(A)_{out}</div><div>DRE_{out}</div><div>PC_{in}</div><div>AR_{in}</div><div>DRE_{in}</div><div>DR_{in}</div><div>X_{in}</div><div>R_{in}</div><div>IR_{in}</div><div>PSW_{in}</div><div>Rs/Rt</div><div>RegDst</div><div>ADD</div><div>+4</div><div>SUB</div><div>Write</div><div>P_{IR}</div><div>P_{equal}</div><div>P_{end}</div></div></div>																								
功能	微地址	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	P ₀	P ₁	P ₂
取指	0	1								1			1													
	1																			1						
	2			1					1		1											1				
	3		1												1											
add	19				1								1				1									
	20				1														1							
	21			1										1				1								1

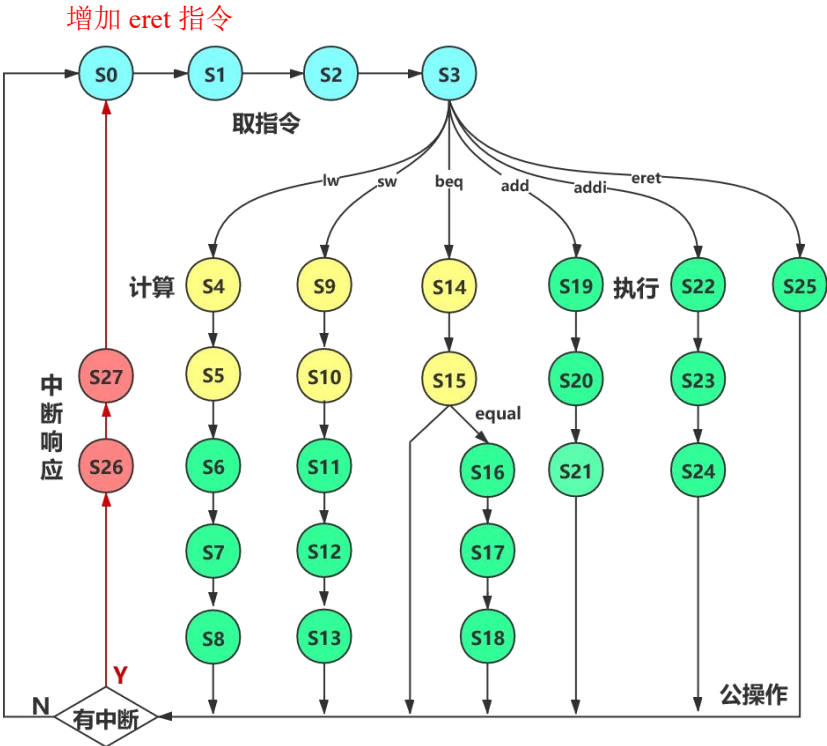
取指微程序第 4 条微指令控制字段为 100100，P 字段为 4

add 指令微程序第 1 条微指令控制字段为 40400，P 字段为 0 (20240)

add 指令微程序第 2 条微指令控制字段为 40050 / 40010

add 指令微程序第 3 条微指令控制字段为 80220，P 字段为 1

5) 如果需要在支持单级中断，需要增加什么指令，如何修改第 2 问中的状态机，请给出状态机图？（4 分）



分 数	
评卷人	

七、(12分) 有一台计算机 CPU 的主频为 2GHz。该计算机接有一台磁盘设备，该外部设备实现数据传输有两种选择：中断方式和 DMA 方式。请完成下面各题（该题中数据传输率、频率中 K/M/G 均以 10 为基数）。

1) 如果采用中断方式，请简述中断响应周期中主要要完成的几个动作并予以说明。（3 分）

（1）关中断：临时禁止中断请求，是为了保障中断响应周期以及中断服务程序中保护现场操作的完整性；

（2）保存断点：保存将来返回被中断程序的位置，对于已经执行完毕的指令其断点是下一条指令的位置（注意有可能不是顺序指令）；

（3）中断识别：就是根据当前的中断请求识别出中断来源，也就是发生了什么中断，并将对应中断的中断服务程序入口地址送程序计数器 PC。

2) 如果采用 DMA 方式，在 DMA 操作的准备阶段，主机需要通过 CPU 指令向 DMA 接口发送必要的传送参数，并启动 DMA 工作。具体需要传送的参数如下（请完善）：（3 分）

- （1）数据传送的方向，
- （2）数据块在主存的首地址，
- （3）数据在外设存储介质上的地址，
- （4）数据的传送量。

3) 假定该外设采用中断方式与主机进行数据传送。启动阶段（发送命令和参数）需要 100 个时钟周期，每次中断服务的开销要 400 个时钟周期（包括中断响应和中断处理，不含数据传输），实际数据传输需要 1500 个时钟周期，CPU 访问硬盘的速率是 20MB/s。如果中断方式下 CPU 的占用率要控制在 20%以下，一次中断至少要传输多少 Byte 数据（假设一次中断可以传送多个字节）？（3 分）

一次中断传输需要的时间开销： $(100+400+1500) / f = 2000 / (2 \times 10^9) = 10^{-6}$ 秒

一次传输总 CPU 时间（含非 IO 的其他时间）： $10^{-6} / 0.2 = 5 \times 10^{-6}$ 秒

一次中断需传输的数据量为： $20\text{MB} \times 5 \times 10^{-6} = 100\text{B}$

4) 如果改为 DMA 方式来传输，CPU 访问硬盘的速率仍是 20MB/s。如 DMA 的预处理阶段需要 100 个时钟周期，DMA 完成传输后的中断处理需要 400 个时钟周期，1 次 DMA 平均传输的数据长度为 10KB，则 DMA 方式下 CPU 的占用率是多少？（3 分）

一次 DMA 操作所需的时间是： $100T+400T=500T=500/f=500/(2 \times 10^9)=2.5 \times 10^{-7}$

一次传送 10KB 总的时间： $10\text{KB}/20\text{MB/s} = 0.5 \times 10^{-3}$

DMA 方式下 CPU 的占用率是： $2.5 \times 10^{-7} / 0.5 \times 10^{-3} = 0.05\%$