

1. $(3A58)_{16} = (0011\ 1010\ 0101\ 1000)_2$,

尾数=1.001011000 (符合规格化要求), 阶码=0,01110

真值=1.001011000 $\times 2^{0.01110} = -0.110101 \times 2^{+1110}$

2. 一) 原码一位乘:

$x^* = 0.110111$, $y^* = 0.101110$

$x_0 = 0$, $y_0 = 1$, $z_0 = x_0 \oplus y_0 = 0 \oplus 1 = 1$

原码一位乘:

部分积		乘数 y^*					
	0.000 000	.	1	0	1	1	1 0 — +0
→1	0.000 000	0.	1	0	1	1	1 — + x^*
+	0.110 111						
	0.110 111						
→1	0.011 011	1	0.	1	0	1	1 — + x^*
+	0.110 111						
	1.010 010						
→1	0.101 001	0	1	0	.	1	0 1 — + x^*
+	0.110 111						
	1.100 000						
→1	0.110 000	0	0	1	0.	1	0 — +0
→1	0.011 000	0	0	0	1	0.	1 — x^*
+	0.110 111						
	1.001 111						
→1	0.100 111	1	0	0	0	1	0

$x^* \times y^* = 0.100111100010$

加符号位, $[x \times y]_{\text{原}} = 1.100\ 111\ 100\ 010$

二) 补码一位乘:

$[x]_{\text{补}} = x = 00.110111$, $[y]_{\text{补}} = 1.010010$, $[-x]_{\text{补}} = 11.001001$

补码一位乘的计算过程如下:

	部分积	乘数 Y_n	附加位 Y_{n+1}
	00.000000	1.01001 <u>0</u>	<u>0</u>
→1	00.000000	0 1 0100 <u>1</u>	<u>0</u>
	+ <u>11.001001</u>		
	11.001001		
→1	11.100100	10 1010 <u>0</u>	<u>1</u>
	+ <u>00.110111</u>		
	00.011011		
→1	00.001101	110 101 <u>0</u>	<u>0</u>
→1	00.000110	1110 10 <u>1</u>	<u>0</u>
	+ <u>11.001001</u>		
	11.001111		
→1	11.100111	11110 1 <u>0</u>	<u>1</u>
	+ <u>00.110111</u>		
	00.011110		
→1	00.001111	011110 <u>1</u>	<u>0</u>
	+ <u>11.001001</u>		
	11.011000	011110	

说明：最后一步不移位，得 $[x \times y]_{\#} = 1.011\ 000\ 011110$ 。

3. (1) 8341H = 1000 0011 0100 0001

寻址方式：相对寻址； EA = (PC) + 2 + 形式地址 A = 5431 + 2 + 0041H = 5474H

(2) 1468H = 0001 0100 0100 1000

寻址方式：直接寻址； EA = 形式地址 A = 0068H

(3) 8100H = 1000 0001 0000 0000

寻址方式：寄存器间接寻址； $EA = (R1) = 3525H$

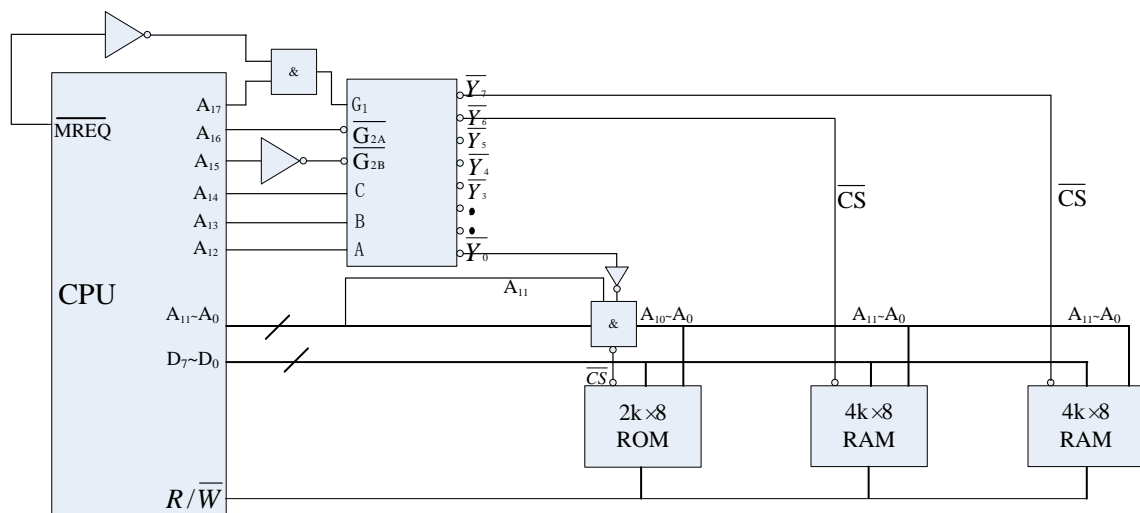
(4) $6264H = 0110\ 0010\ 0110\ 0100$

寻址方式：变址寻址； $EA = (R2) + \text{位移量 } D = 6783H + 0064H = 67E7H$

4.

中断源	屏蔽字				
	0	1	2	3	4
L_0	1	0	0	1	0
L_1	1	1	1	1	1
L_2	1	0	1	1	0
L_3	0	0	0	1	0
L_4	1	0	1	1	1

5.



6. 当CPU发出主存地址后,地址映射机构按照全相联映射方式将主存地址标记与Cache所有字块的标记进行比较,以判断出所访主存字(主存地址的内容)是否已在Cache中。若命中,直接访问Cache,将该字送至CPU;若未命中,一方面要访问主存,将该字传送给CPU,与此同时,要按照全相联映射方式转换的Cache地址将该字所在的主存块装入Cache,如果此时Cache已满,就要执行替换算法,腾出空位才能将新的主存块调入。

7.CPU可从时间和空间两个层面来区分访存取来的指令和数据。

1) 时间层面:在取指周期(或运行取指微程序)内,由PC提供访存地址,取来的即为指令;在执行周期(或运行执行周期相对应的微程序段)内,由指令的地址码部分提供访存地址,取来的即为操作数,也就是数据。

2) 空间层面:取来的机器指令应存放在指令寄存器,而取来的数据(或操作数)则应该存放在以累加器为代表的通用寄存器内。