

# Answer 3

## A1

opcode 有 64 条, 则需要 6bit 才能表示完整。register 有 56 个, 也需要 6bit 才能表示完整。则 IMM 可用位数为:  $32 - 6 - 6 - 6 = 14\text{bit}$  则表示范围:  $-2^{13} < IMM < 2^{13} - 1$

## A2

前 4 个都比较显然, 第 5 个注意循环 `R0` 循环加 2 直到加到 x8000 后循环中止, 再加上后面的 3 和 1, 得到 x8004.

X	Does the program halt?	Value stored in <code>R0</code>
000000010	Yes	2
000000001	Yes	3
000000000	Yes	6
111111111	No	--
111111110	Yes	x8004 ( $-2^{15} + 4$ )

## A3

当 `R1` 加 `R2` 的和为正数或 0 时, 之后会执行位于 `x3009` 的指令.

## A4

1. ADD(0001) 和 AND(0101) 有了更大的立即数范围。但 NOT(1001) 没有获益。
2. LD(0010) 和 ST(0011) 能有更多的一位去寻址。
3. BR(0000) 中没有寄存器, 因此没有获益。

## A5

101, 011, 010, 110, 110

内存单元内存储的值只需看最后一次写入时的输入  $D_{in}$ , 而  $D_{out}$  看最后一次操作即可

若回答 111, 011, 010, 111, 111 , 则判此题完全不得分

## A6

此题需要严密的推理。

Operation No.	R/W	MAR	MDR
1	W	x4000	11110
2	R	x4003	10110
3	W	x4001	10110
4	R	x4002	01101
5	W	x4003	01101

Operations on Memory

Address	Before Access 1	After Access 3	After Access 5
x4000	01101	11110	11110
x4001	11010	10110	10110
x4002	01101	01101	01101
x4003	10110	10110	01101
x4004	11110	11110	11110

Contents of Memory locations

## A7

- a.  $2 \times 10^8$
- b.  $2.5 \times 10^7$
- c.  $\approx 2 \times 10^8$

## A8

以下用  $a \oplus b$  表示 a XOR b,  $a + b$  表示 a OR b,  $a \cdot b$  表示 a AND b,  $\bar{a}$  表示 NOT a, 则有  $a \oplus b = a \cdot \bar{b} + \bar{a} \cdot b = \overline{a \cdot \bar{b} + \bar{a} \cdot b} = \overline{\bar{a} \cdot \bar{b}} \cdot \overline{a \cdot b}$ .

Address	Instruction	Comments
x3000	1001 111 001 111111	NOT R7, R1
x3001	1001 110 010 111111	NOT R6, R2
x3002	0101 101 111 000 010	AND R5, R7, R2
x3003	0101 100 110 000 001	AND R4, R6, R1
x3004	1001 001 101 111111	NOT R1, R5
x3005	1001 010 100 111111	NOT R2, R4
x3006	0101 000 001 000 010	AND R0, R1, R2
x3007	1001 011 000 111111	NOT R3, R0

## A9

其中第2条可以作为NOP，因为其偏移设置为0

1. 将R1中的值与立即数2相加并存放R2中
2. NOP
3. 检查标志位N和P，若其中一个被置位则跳转到对应的位置
4. 对R7求反存放R2中
5. TRAP指令，读取内存单元x0023中的内容作为服务程序入口

## A10

BR指令无法跳转到PC寄存器和偏移量之和范围以外的指令，即其最大范围为距离当前指令+256到-255的地址空间

对于JMP指令，其能够将对应寄存器的内容装入PC寄存器，使得程序执行流可以跳转至内存空间的任意位置。