|  |  |
| --- | --- |
| 001 |  |

**合肥工业大学**

**计算机与信息学院**

**作业**

**课 程：汇编语言程序设计**

**专业班级： 物联网一班**

**学 号：2023212388**

**姓 名： 敬成超**

**成 绩：**

1-7：

1. 10010001 -> 91

2. 10001001-> 89

3. 00110110-> 36

4. 10010000 -> 90

5. 00001000 -> 8

6. 10010111 -> 97

7. 10000001 -> 81

8. 00000010 -> 2

#### 1-8

#### (1) 0

原码：00000000

反码：00000000

补码：00000000

**(2) -127**

原码：11111111

反码：10000000

补码：10000001

**(3) 127**

原码：01111111

反码：01111111

补码：01111111

**(4) -5**

原码：11111011

反码：10001001

补码：10001010

**(5) 126**

原码：01111110

反码：01111110

补码：01111110

**(6) -126**

原码：10000010

反码：01111110

补码：01111111

**(7) -128**

原码：10000000

反码：01111111

补码：10000000

**(8) 6**

原码：00000110

反码：00000110

补码：00000110

### 1-11

### 如果把它认为是无符号数

"01100001" 作为无符号数转换为十进制数是 97。

如果认为它是BCD码

0110 对应的十进制数是 6

0001 对应的十进制数是 1

所以，"01100001" 作为BCD码转换为十进制数是 61。

如果它是某个ASCII码

ASCII码 "01100001" 代表的字符是 'a'。

1-19

在8086微处理器中：

**逻辑地址**由段地址和偏移地址两部分组成。它是由程序设计者定义的地址，用于访问内存。

**物理地址**是实际的内存地址，由内存管理单元（MMU）将逻辑地址转换而来。逻辑地址到物理地址的转换

逻辑地址转换为物理地址的过程如下： 物理地址=(段地址×16)+偏移地址物理地址=(段地址×16)+偏移地址 其中，段地址左移4位（即乘以16），然后加上偏移地址。

给定的逻辑地址及其转换为物理地址：

1. **FFFF:0**
   * 段地址：FFFF（65535），偏移地址：0000
   * 物理地址：(65535×16)+0=1048576+0=1048576(65535×16)+0=1048576+0=1048576
   * 十六进制表示：0x100000
2. **40:17**
   * 段地址：0040（64），偏移地址：0017（23）
   * 物理地址：(64×16)+23=1024+23=1047(64×16)+23=1024+23=1047
   * 十六进制表示：0x0417
3. **2000:4500**
   * 段地址：2000（8192），偏移地址：4500（17664）
   * 物理地址：(8192×16)+17664=131072+17664=148736(8192×16)+17664=131072+17664=148736
   * 十六进制表示：0x24000
4. **B821:4567**
   * 段地址：B821（47649），偏移地址：4567（17743）
   * 物理地址：(47649×16)+17743=762384+17743=780127(47649×16)+17743=762384+17743=780127
   * 十六进制表示：0xBDC67

1-24

1. 立即寻址（Immediate Addressing）：

操作数直接作为立即数给出，例如：MOV AX, 1234H。

1. 直接寻址（Direct Addressing）：

操作数的地址直接给出，例如：MOV AX, [1234H]。

1. 寄存器寻址（Register Addressing）：

操作数直接在寄存器中，例如：MOV AX, BX。

1. 寄存器间接寻址（Register Indirect Addressing）：

操作数的地址存储在寄存器中，例如：MOV AX, [BX]。

1. 基址寻址（Base Addressing）：

使用基址寄存器加上偏移量来形成操作数的地址，例如：MOV AX, [BX+1234H]。

1. 变址寻址（Indexed Addressing）：

使用变址寄存器加上偏移量来形成操作数的地址，例如：MOV AX, [BX+SI]。

1. 基址址寻址（Base Plus Indexed Addressing）：

结合基址寄存器和变址寄存器以及偏移量来形成操作数的地址，例如：MOV AX, [BX+SI+1234H]。

1.mov dx,[1234H]

寻址方式：直接寻址

有效地址EA：1234H（因为操作数直接给出

2.mov dx,1234H

寻址方式：立即寻址

有效地址EA：不适用（操作数是立即数）

3.mov dx,bx

寻址方式：寄存器寻址

有效地址EA：不适用（操作数在寄存器中）

4.mov dx,[bx]

寻址方式：寄存器间接寻址

有效地址EA：BX寄存器的值（如果BX=2000H，则EA=2000H）

5.mov dx,[bx+1234H]

寻址方式：基址加立即数寻址

有效地址EA：BX + 1234H（如果BX=2000H，则EA=2000H + 1234H = 3234H）

6.mov dx,[bx+di]

寻址方式：基址加变址寻址

有效地址EA：BX + DI（如果BX=2000H且DI=40H，则EA=2000H + 40H = 2040H）

7.mov dx,[bx+di+1234H]

寻址方式：基址加变址加立即数寻址

有效地址EA：BX + DI + 1234H（如果BX=2000H且DI=40H，则EA=2000H + 40H + 1234H = 3234H）