

汇编语言程序设计

# 汇编语言的常量表达



# 常量 (Constant)

- 常量是程序中使用的一个确定数值
  - ▶ 在汇编阶段就可以确定，直接编码于指令代码中
  - ▶ 不是保存在存储器中可变的变量
- 汇编语言支持多种常量的表达形式
  - ▶ 常数：直接给出数值本身
  - ▶ 字符和字符串：使用**ASCII**码值
  - ▶ 符号常量：用标识符替代数值
  - ▶ 数值表达式：经过运算得到数值



# MASM常量表达：常数

- 十、十六和二进制形式表达的数值
  - ▶ 以后缀字母（**D**、**H**和**B**）区分，十进制数可以不加
- 以字母**A**~**F**开头的十六进制常数，要加前导**0**
  - ▶ 以便区别于以字母开头的标识符

十进制数	100, 255D
十六进制数	64H, 0FFH, 0B800H
二进制数	01101100B

“AH”是指寄存器

“0AH”表示十六进制常数（即十进制常数10）



# MASM常量表达：字符和字符串

- 英文缩略号（单引号）或双引号括起来的单个或多个字符
- 每个字符的数值是对应的**ASCII**码值

字符	'd' ( = 64H )
字符串	'Hello, Assembly !' "Let's have a try." 'Say "Hello", my baby.'

单引号、双引号没有区别

字符和字符串也没有本质上的区别



# MASM常量表达：符号常量

## ➤使用标识符表达一个数值

符号名 = 数值表达式

符号名 **equ** 数值表达式

符号名 **equ** <字符串>

等号 =	STD_INPUT_HANDLE = -10 STD_OUTPUT_HANDLE = -11
等价EQU	NULL equ 0 WriteConsole equ < WriteConsoleA>



# MASM常量表达：数值表达式

➤用运算符连接各种常量构成的算式

▶常用的算术运算符：

＋（加） －（减） \*（乘） /（除）

➤数值用作地址，数值表达式成为地址表达式

▶地址表达式只能使用加减

▶指示地址移动常量表示的若干个字节存储单元

地址＋常量

地址－常量

算术表达式	67h-52h
地址表达式	var+4



# 常量表达

## 1. 常数

- ▶ 十、十六和二进制形式表达的数值

## 2. 字符和字符串

- ▶ 英文单引号或双引号括起的单个或多个字符

## 3. 符号常量

- ▶ 使用标识符表达一个数值

## 4. 数值表达式

- ▶ 用运算符连接各种常量构成的算式

常量具有确定数值



汇编语言程序设计

# 数据表达程序





# 常量表达

## 1. 常数

- ▶ 十、十六和二进制形式表达的数值

## 2. 字符和字符串

- ▶ 英文单引号或双引号括起的单个或多个字符

## 3. 符号常量

- ▶ 使用标识符表达一个数值

## 4. 数值表达式

- ▶ 用运算符连接各种常量构成的算式

常量具有确定数值



# 数据表达程序—1

00000000

64 64 64 64 64

机器指令

const1 byte 100,100d,01100100b,64h, 'd'

相对地址

汇编语句

不同进制和形式表达了同一个数值：**100 (=64H)**  
无论源程序如何表达，计算机内部都是二进制编码

(8位) 字节变量定义伪指令: **byte**

# 数据表达程序—2

**00000005      01 7F 80 80 FF FF**

**const2   byte 1,+127,128,-128,255,-1**

**0000000B      69 97 20 E0 32 CE**

**const3   byte 105,-105,32,-32,32h,-32h**

无符号数**255**和有符号数**-1**: **FFH**

无符号数**128**和有符号数**-128**: **80H**

**105**的补码: **69H**, **-105**的补码: **97H**



# 数据表达程序—3

**00000011      30 31 32 33 34 35**

**36 37 38 39 61 62**

**63 78 79 7A 41 42**

**43 58 59 5A**

**const4   byte '0123456789','abcxyz','ABCXYZ'**

定义字符串：是每个字符的**ASCII**码



# 数据表达程序—4

**00000027      0D 0A 00**

**crlf          byte 0dh,0ah,0**

数值**0DH**（回车符）和**0AH**（换行符）  
注意前导零不能省略（否则成为**DH**和**AH**）  
数字“**0**”用于字符串结尾



# 数据表达程序—5

**= 0000000A minint = 10**

**= 000000FF maxint equ 0ffh**

**0000002A 0A 0F FA F5**

**const5 byte minint,minint+5**

**byte maxint-5,maxint-minint**

符号**MININT**为**10**，**MAXINT**为**255**，不占主存空间  
表达式定义**CONST5**的每个参数，实质还是常量



# 数据表达程序—6

**0000002E    10 56 15 EB**

**const6   byte 4\*4,34h+34,67h-52h,52h-67h**

表达式定义**CONST6**的每个参数，但实质还是常量



# 数据表达程序—7

**mov eax,offset const4** ;从**CONST4**开始显示  
**call dispmsg**

```
D:\MASM>make32 eg0201
```

```
.....
```

```
D:\MASM>eg0201.exe
```

操作过程

运行结果

```
0123456789abcdefghijklmnopqrstuvwxyzABCXYZ
```





# 数据表达程序

;数据段

**const1**    **byte 100,100d,01100100b,64h, 'd'**

...

**const4**    **byte '0123456789','abcxyz','ABCXYZ'**

**crlf**      **byte 0dh,0ah,0**

...

;代码段

**mov eax,offset const4**

**call dispmsg**

**0123456789abcxyzABCXYZ**

运行结果

