

# 汇编语言程序设计

Saturday 18<sup>th</sup> February, 2023

# 目录

<b>I</b>	<b>寄存器</b>	<b>3</b>
<b>1</b>	<b>通用寄存器</b>	<b>3</b>
1.1	数据寄存器	3
1.1.1	累加寄存器 AX (Accumulator)	3
1.1.2	基址寄存器 BX (Base)	3
1.1.3	计数寄存器 CX (Count)	3
1.1.4	数据寄存器 DX (Data)	3
1.2	指针寄存器	3
1.2.1	栈指针寄存器 SP (Stack Pointer)	3
1.2.2	基指针寄存器 BP (Base Pointer)	3
1.3	变址寄存器	3
1.3.1	源变址寄存器 SI (Source Index)	3
1.3.2	目的变址寄存器 DI (Destination Index)	3
<b>2</b>	<b>专用寄存器</b>	<b>3</b>
2.1	指令指针寄存器 IP (Instruction Pointer)	3
2.2	标志寄存器 FLAG	4
<b>3</b>	<b>段寄存器</b>	<b>4</b>
3.1	代码段寄存器 CS (Code Segment)	4
3.2	数据段寄存器 DS (Data Segment)	4
3.3	堆栈段寄存器 SS (Stack Segment)	4
3.4	附加段寄存器 ES (Extra Segment)	4
<b>II</b>	<b>标志位</b>	<b>4</b>
<b>4</b>	<b>状态标志位</b>	<b>4</b>
4.1	0 进位标志 CF(Carry Flag)	4
4.2	6 零标志 ZF(Zero Flag)	5
4.3	2 奇偶标志 PF(Parity Flag)	5
4.4	4 辅助进位标志 AF(Auxiliary Carry Flag)	5
4.5	7 符号标志 SF(Sign Flag)	5
4.6	11 溢出标志 OF(Overflow Flag)	5
<b>5</b>	<b>控制标志位</b>	<b>5</b>
5.1	10 方向标志 DF(Direction Flag)	5
5.2	9 中断允许标志 IF(Interrupt-enable Flag)	5
5.3	8 陷阱标志 TF(Trap Flag)	5

<b>6</b>	<b>内存操作数寻址方式</b>	<b>6</b>
6.1	直接寻址方式 (direct addressing) . . . . .	6
6.2	寄存器间接寻址方式 (register indirect) . . . . .	6
6.3	寄存器相对寻址方式 (register relative) . . . . .	6
6.4	基址变址寻址方式 (based indexed) . . . . .	6
6.5	相对基址变址方式 (relative based indexed) . . . . .	6

## Part I

# 寄存器

## 1 通用寄存器

### 1.1 数据寄存器

#### 1.1.1 累加寄存器 AX (Accumulator)

可分为 AH 和 AL 两个 8 位寄存器

#### 1.1.2 基址寄存器 BX (Base)

可分为 BH 和 BL 两个 8 位寄存器

#### 1.1.3 计数寄存器 CX (Count)

可分为 CH 和 CL 两个 8 位寄存器

#### 1.1.4 数据寄存器 DX (Data)

可分为 DH 和 DL 两个 8 位寄存器

### 1.2 指针寄存器

与 SS(3.3) 联合使用

#### 1.2.1 栈指针寄存器 SP (Stack Pointer)

栈顶的偏移地址

#### 1.2.2 基指针寄存器 BP (Base Pointer)

数据在堆栈段中的基地址

### 1.3 变址寄存器

#### 1.3.1 源变址寄存器 SI (Source Index)

#### 1.3.2 目的变址寄存器 DI (Destination Index)

## 2 专用寄存器

### 2.1 指令指针寄存器 IP (Instruction Pointer)

代码段中指令的偏移地址，与 CS(3.1) 联合使用

## 2.2 标志寄存器 FLAG

## 3 段寄存器

用来确定该段在内存中的起始地址。用途特定，不可分开使用。段的长度不超过  $2^{16} = 64K$

### 3.1 代码段寄存器 CS (Code Segment)

与 IP(2.1) 联合使用

### 3.2 数据段寄存器 DS (Data Segment)

### 3.3 堆栈段寄存器 SS (Stack Segment)

与指针寄存器 (1.2) 联合使用

### 3.4 附加段寄存器 ES (Extra Segment)

## Part II

# 标志位

DEBUG 模式下的表示

标志位	1	0
OF	OV	NV
DF	DN	UP
IF	EI	DI
SF	NG	PL
ZF	ZR	NZ
AF	AC	NA
PF	PE	PO
CF	CY	NC

## 4 状态标志位

记录当前运算结果的状态信息，是 CPU “自动” 完成的

### 4.1 0 进位标志 CF(Carry Flag)

当运算结果的最高有效位有进位（加法）或借位（减法）时，CF=1

#### 4.2 6 零标志 ZF(Zero Flag)

若运算结果为 0, ZF=1

#### 4.3 2 奇偶标志 PF(Parity Flag)

当运算结果最低 8 位中“1”的个数为偶数时, PF=1

#### 4.4 4 辅助进位标志 AF(Auxiliary Carry Flag)

运算时低 4 位有进位或借位时, AF=1

#### 4.5 7 符号标志 SF(Sign Flag)

运算结果最高位为 1, SF=1

#### 4.6 11 溢出标志 OF(Overflow Flag)

若算术运算的结果有溢出, OF=1 (对无符号数而言, OF=1 并不意味着结果出错)

### 5 控制标志位

存放控制 CPU 工作方式的标志信息

#### 5.1 10 方向标志 DF(Direction Flag)

用于串操作指令中, 控制地址的变化方向:

设置 DF=0, 串操作的存储器地址自动增加;

设置 DF=1, 串操作的存储器地址自动减少。

#### 5.2 9 中断允许标志 IF(Interrupt-enable Flag)

用于控制外部可屏蔽中断是否可以被处理器响应:

设置 IF=1, 则允许中断;

设置 IF=0, 则禁止中断。

#### 5.3 8 陷阱标志 TF(Trap Flag)

用于控制处理器是否进入单步操作方式:

设置 TF=0, 处理器正常工作;

设置 TF=1, 处理器单步执行指令。

单步执行指令: 处理器在每条指令执行结束时, 便产生一个编号为 1 的内部中断

## 6 内存操作数寻址方式

括号内只能有这四种 BX、BP、SI、DI 四种寄存器

(段寄存器):(基址)+(变址)+(位移量)(三种寻址至少有一)

(CS 或 DS 或 SS 或 ES:)([BX 或 BP])([SI 或 DI])([8bits 或 16bits])

若未指定段寄存器:

若出现 BP, 默认操作 CS; 否则默认操作 SS

### 6.1 直接寻址方式 (direct addressing)

[8bits 或 16bits]

### 6.2 寄存器间接寻址方式 (register indirect)

[BX 或 BP 或 SI 或 DI]

### 6.3 寄存器相对寻址方式 (register relative)

[BX 或 BP 或 SI 或 DI][8bits 或 16bits]

### 6.4 基址变址寻址方式 (based indexed)

[BX 或 BP][SI 或 DI]

### 6.5 相对基址变址方式 (relative based indexed)

[BX 或 BP][SI 或 DI][8bits 或 16bits]