3 8086 的寻址方式和指令系统

- 8086 指令格式
- 8086 的寻址方式
- 8086 的指令系统

3.1 8086 指令格式

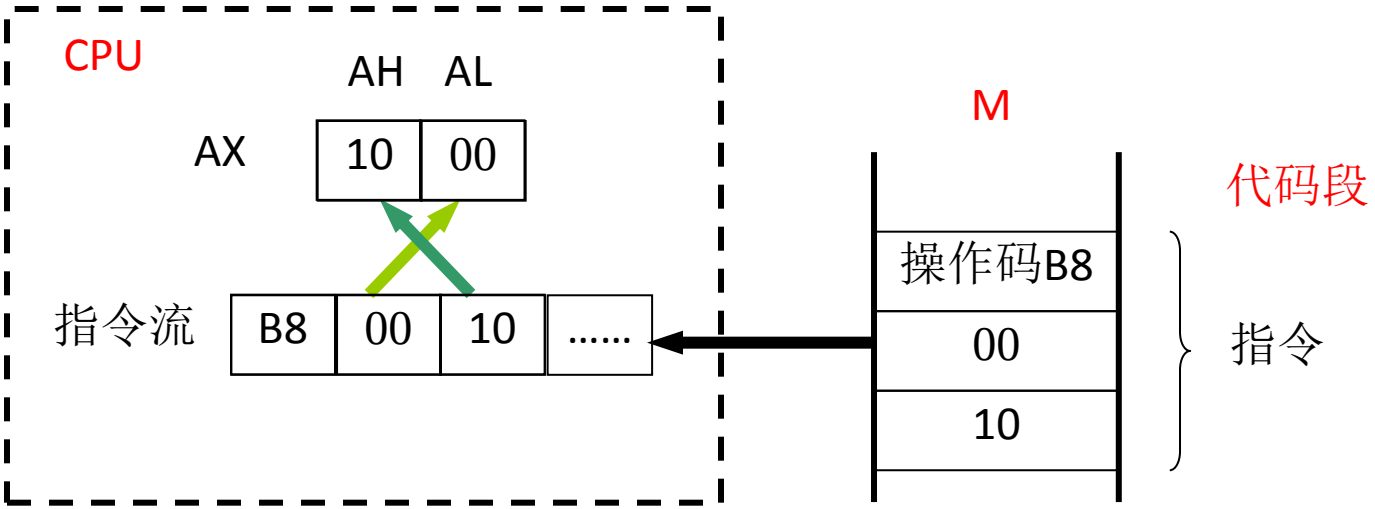
[标号:] [前缀指令] 指令助记符 操作数 [;注释]

3.2 8086 的寻址方式

- 寄存器寻址方式
- 立即寻址方式
- 直接寻址方式
- 寄存器间接寻址方式
- 寄存器相对寻址方式
- 基址变址寻址方式
- 相对基址变址寻址方式

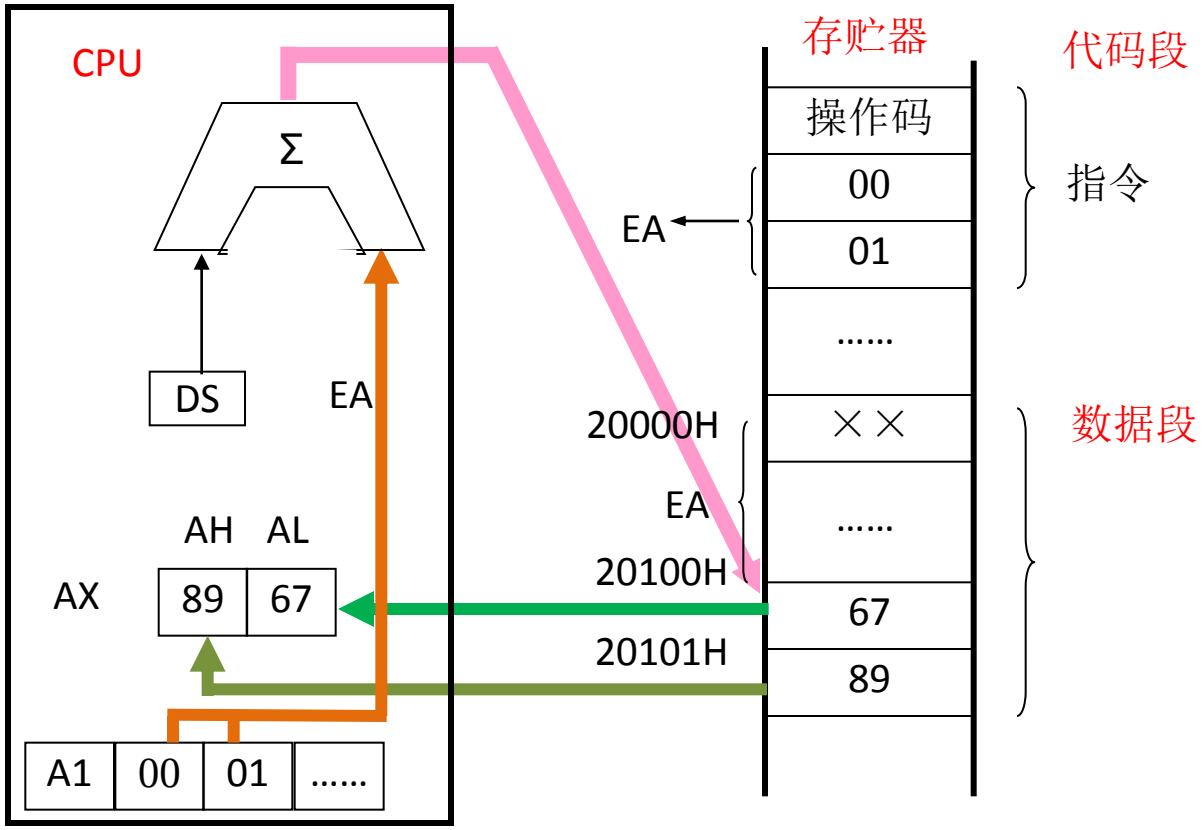
立即寻址方式

MOV AX，1000H



直接寻址方式

MOV AX，[100H]

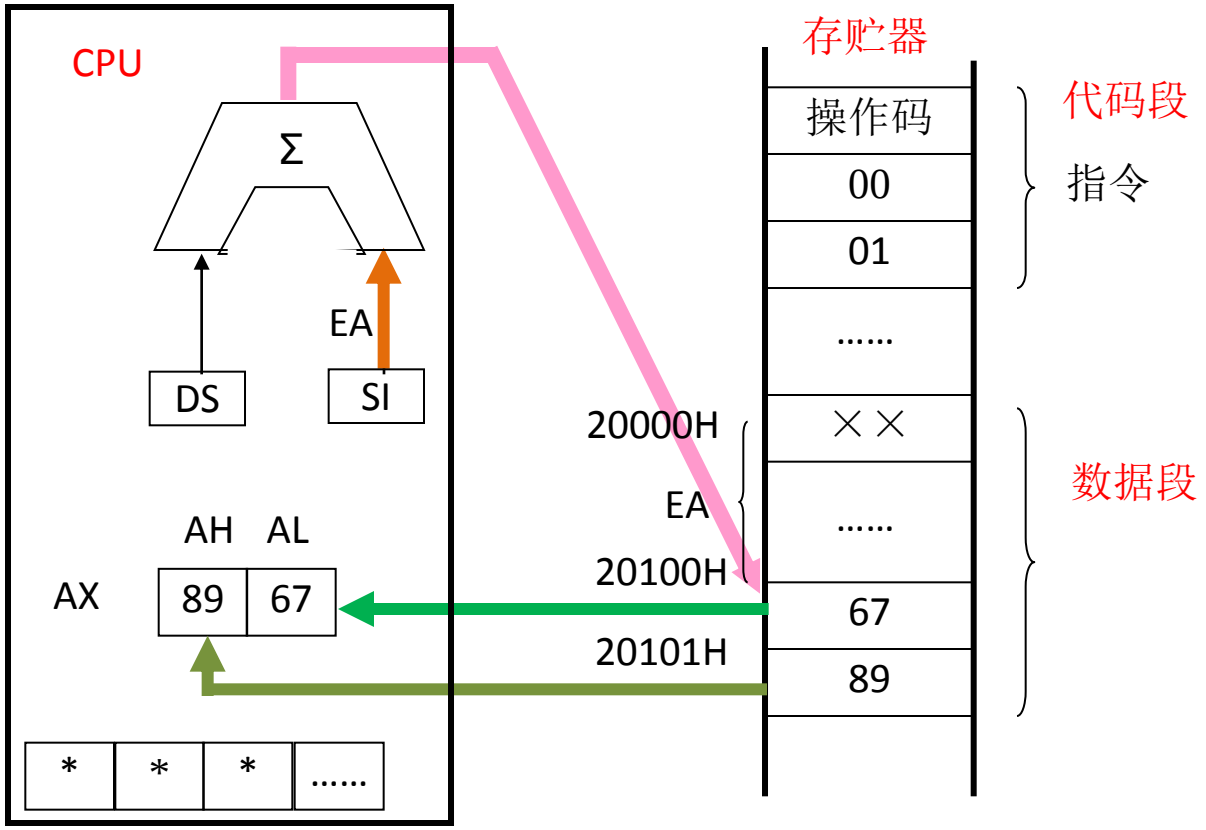


寄存器间接寻址方式

EA = BX / BP / SI / DI

MOV SI，100H

MOV AX，[SI]



寄存器相对寻址方式

MOV BP，200H

MOV AX，[BP+34H]

$$EA = \begin{pmatrix} BX \\ BP \\ SI \\ DI \end{pmatrix} + disp$$

基址变址寻址方式

MOV SI，100H

MOV BX，10H

MOV AX，[BX+SI]

$$EA = \begin{pmatrix} BX \\ BP \end{pmatrix} + \begin{pmatrix} SI \\ DI \end{pmatrix}$$

相对基址变址寻址方式

MOV SI，100H

MOV BX，10H

MOV AX，[BX+SI+2]

$$EA = \underset{\text{I}}{\begin{pmatrix} BX \\ BP \end{pmatrix}} + \underset{\text{II}}{\begin{pmatrix} SI \\ DI \end{pmatrix}} + \underset{\text{III}}{disp}$$

3.3 8086 的指令系统

- 数据传送指令
- 算术运算指令
- 逻辑运算和移位指令
- 控制转移指令
- 字符串处理指令
- 处理器控制指令

MOV

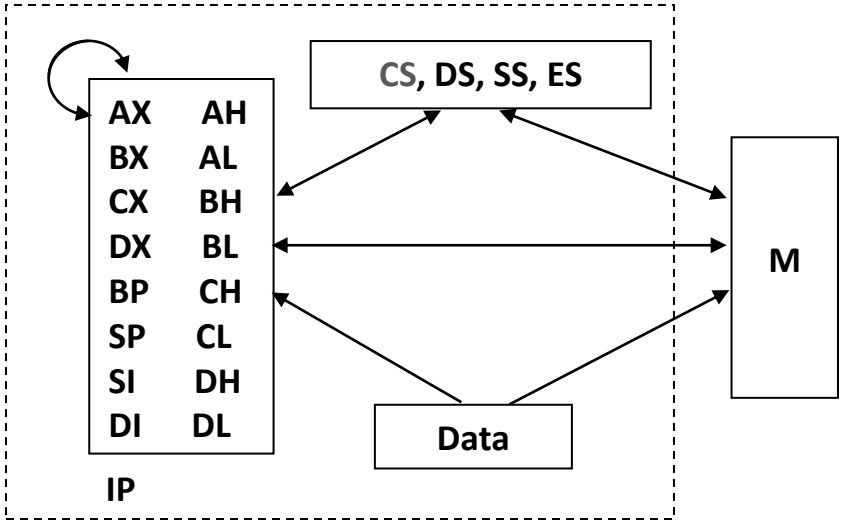
格式： MOV DST, SRC

SRC： Register, Memory, Segment Register, Immediate.

DST： Register, Memory.

功能： SRC→DST

FR： No influence .



注意：

CS和IP不能作为目的操作数。

目的操作数不允许用立即数形式。

立即数不能直接传送到段寄存器。

不许在两个段寄存器之间直接传送信息。

不许在两个存储单元之间直接传送数据。

MOV

Example：

1. MOV BL，AX
2. MOV 100H，AX
3. MOV DS，2000H
4. MOV ES，DS
5. MOV CS，3000H
6. MOV [BX]，[1000H]
7. MOV ES：AX，2000H
8. MOV VAR[SI][DI]，AX
9. MOV AL，F0H
10. MOV AX，DS
11. MOV IP，100H
12. MOV AX，IP
13. MOV [BX]，BL
14. MOV [BX]，10
15. MOV [DX]，10

PUSH

格式：　PUSH SRC

SRC：　Register, Segment Register, Memory.

功能：　SP—2→SP, SRC →SS：[SP].

Flags：　No influence.

POP

格式：　POP DST

DST：　Register, Segment Register, Memory.

操作：　SS:[SP] →DST, SP+2→SP .

Flags：　No influence.

PUSH/POP

Example：

设SS＝3000H, SP＝20H, AX＝1234H, BX＝0ABCDH,

依次执行下列指令：

① PUSH AX

② PUSH BX

③ POP AX

分析堆栈中数据和SP的变化.

PUSHF

格式：　　PUSHF

功能：　　SP—2→SP,　FR →SS：[SP].

Flags：　　No influence.

POPF

格式：　　POPF

功能：　　SS:[SP] →FR,　SP+2→SP.

Flags：　　Affected.

XCHG

格式：　　XCHG　OD1, OD2

操作数：　　Register, Memory.

功能：　　R←→R/M.

Flags：　　No influence .

LEA

格式：　　LEA REG, SRC

SRC：　　Memory.

DST：　　Register.

功能：　　offset address of SRC→REG.

Flags：　　No influence.

LDS

格式：　LDS DST, SRC

SRC：　Memory.

DST：　Register.

功能：　M→DS & R .

Flags：　No influence.

- Example：

  设 DS=1200H, [12450H]=0F346H, [12452H]=0A90H,
  说明LDS SI, [450H] 指令执行后，SI和DS的内容.

XLAT

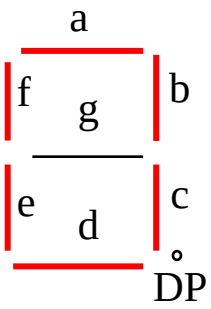
格式： XLAT

功能： DS：[BX＋AL]→AL.

Flags： No influence.

Example： 若十进制数字0～9的LED七段码对照关系如表，存于TABLE开始的内存单元中，试用XLAT指令求数字2的七段码值.

BCD	0	1	2	3	4	5	6	7	8	9
7-segment	3FH	06H	5BH	4FH	66H	6DH	7DH	07H	7FH	6FH



In / Out

直接寻址 (n<256)

```
IN    AL, n          IN    AX, n
OUT   n , AL         OUT   n , AX
```

间接寻址

```
IN    AL, DX         IN    AX, DX
OUT   DX , AL        OUT   DX , AX
```

功能: I/O ←→AL/AX.

ADD

格式：　ADD DST, SRC

SRC：　Register, Memory, Immediate .

DST：　Register, Memory .

功能：　DST+SRC→DST

Flags：　OF, SF, ZF, AF, PF, CF.

ADC

格式：　ADC DST, SRC

SRC：　Register, Memory, Immediate .

DST：　Register, Memory .

功能：　DST+SRC+CF→DST

Flags：　OF, SF, ZF, AF, PF, CF.

INC

格式：　　INC OPR

OPR:　　Register, Memory.

功能：　　OPR+1→OPR

Flags：　　OF, SF, ZF, AF, PF.

DAA/AAA

格式：　　DAA

功能：　　压缩BCD码加法调整.

Flags：　　SF, ZF, AF, PF, CF.

Note：　　操作数为AL.

格式：　　AAA

功能：　　非压缩BCD码加法调整.

Flags：　　AF, CF.

Note：　　操作数为AX.

SUB

格式：　SUB　DST, SRC

SRC：　Register, Memory, Immediate .

DST：　Register, Memory .

功能：　DST－SRC→DST

Flags：　OF, SF, ZF, AF, PF, CF.

SBB

格式：　SBB　DST, SRC

SRC：　Register, Memory, Immediate .

DST：　Register, Memory .

功能：　DST－SRC－CF→DST

Flags：　OF, SF, ZF, AF, PF, CF.

DEC

格式：　DEC OPR

OPR：　Register, Memory.

功能：　OPD － 1→OPD

Flags：　OF, SF, ZF, AF, PF.

例：设（BX）=1000H，DS：[1000H]=0200H, 说明下列指令分别执行后BX或内存的内容？

DEC BX　　DEC BL

DEC WORD PTR[BX]　　DEC BYTE PTR[BX]

NEG

格式：　　NEG OPR

OPR：　　register, memory.

功能：　　0－OPR→OPR.

Flags：　　SF, ZF, AF, PF, OF, CF.

CMP

格式：　CMP OPR1, OPR2

OPR2：　Register, Memory, Immediate .

OPR1：　Register, Memory .

功能：　OPD1—OPD2

Flags：　OF, SF, ZF, AF, PF, CF.

MUL/IMUL

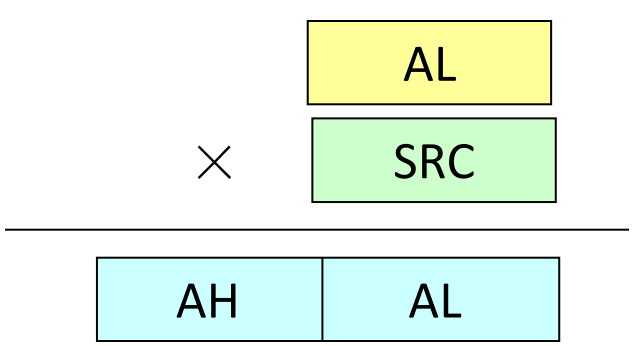
格式：　MUL SRC　（无符号数）

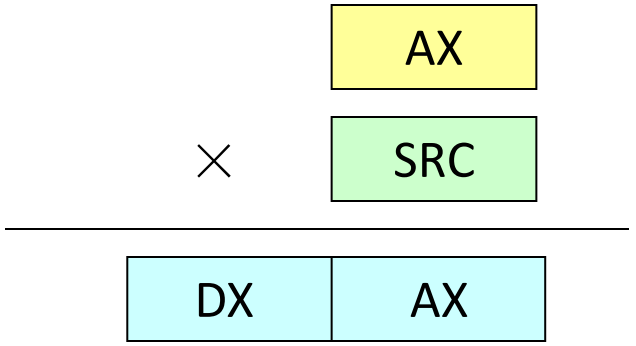
IMUL SRC　（有符号数）

SRC：　Register, Memory.

功能：　SRC×AL→AX

SRC×AX→DX, AX





Flags：　OF, CF.

DIV/IDIV

格式：　　DIV SRC　　（无符号数）

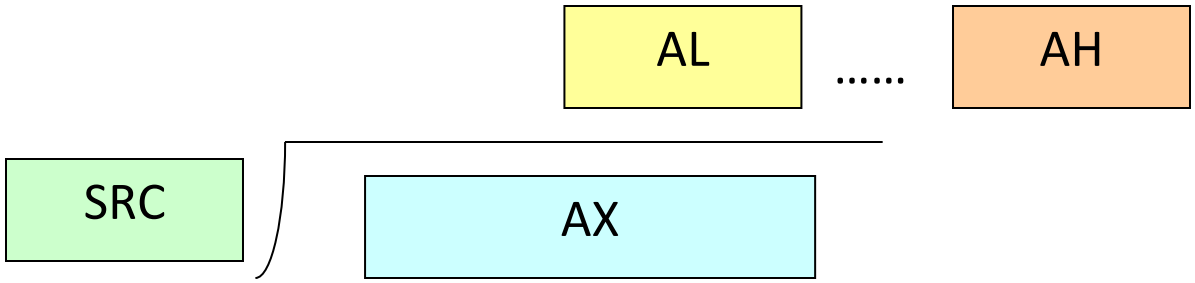
　　IDIV SRC　　（有符号数）

SRC：　　Register, Memory.

功能：　　AX÷SRC=AL……AH

　　DX, AX ÷SRC=AX……DX

Flags：　　undefined.



CBW/CWD

格式：　　CBW

　　CWD

功能：　　CBW—将AL的符号位扩展到AH.

　　CWD—将AX的符号位扩展到DX.

Flags：　　No influence.

NOT

格式：　　NOT DST

DST：　　register, memory .

功能：　　$\overline{\text{DST}}$ → DST.

Flags：　　No influence.

AND

格式：　　AND DST, SRC

SRC：　　immediate, memory, register .

DST：　　memory, register.

功能：　　DST ∧ SRC→DST.

Flags：　　CF , OF , PF, SF, ZF.

OR

格式：　　OR DST, SRC

SRC：　　immediate, memory, register .

DST：　　memory, register.

功能：　　DST V SRC→DST.

Flags：　　CF , OF, PF, SF, ZF.

XOR

格式：　　XOR　DST, SRC

SRC：　　immediate，　memory, register .

DST：　　memory, register.

功能：　　DST ∀ SRC→DST.

Flags：　　CF , OF , PF, SF, ZF.

TEST

格式： TEST DST, SRC

SRC： immediate, memory, register .

DST： memory, register.

功能： DST ∧ SRC.

Flags： CF , OF , PF, SF, ZF.

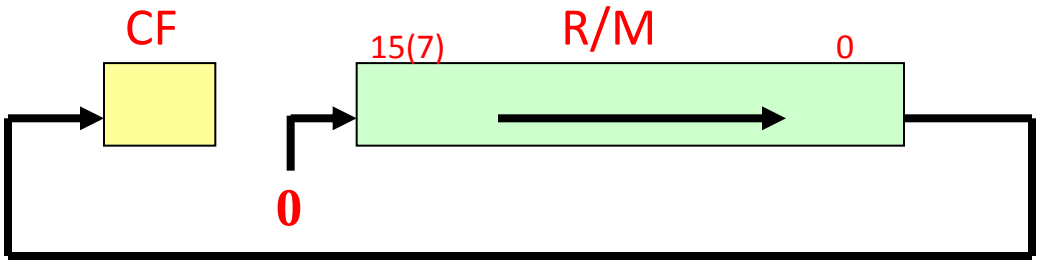
SHR

格式：　　SHR　DST, CNT

DST：　　memory, register.

CNT ：　　1, CL.

功能：



Flags：　　CF, PF, SF, ZF, OF .

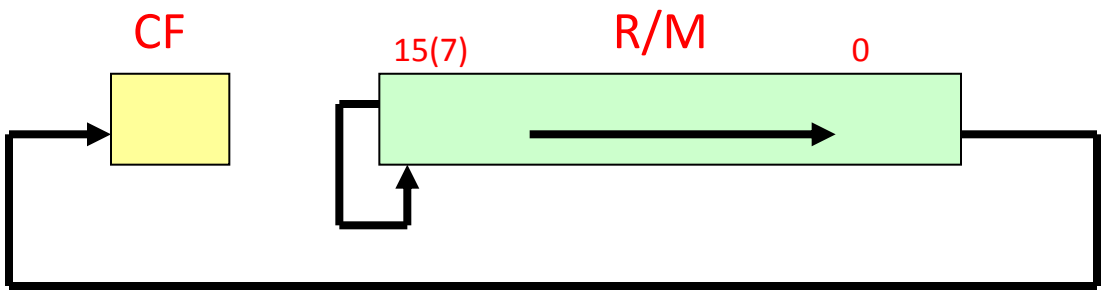
SAR

格式：　　SAR DST, CNT

DST：　　memory, register.

CNT：　　1, CL.

功能：



Flags：　　CF, PF, SF, ZF, OF .

SHL/SAL

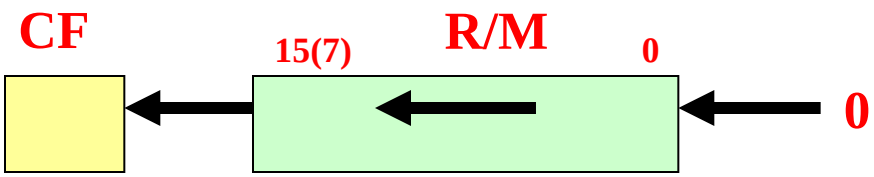
格式：　　SHL　DST, CNT

　　SAL　DST, CNT

DST：　　memory, register.

CNT ：　　1, CL.

功能：



Flags：　　CF, PF, SF, ZF, OF .

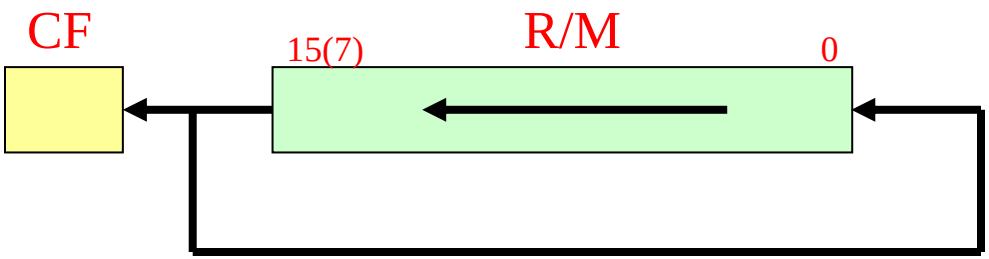
ROL

格式：　　ROL DST, CNT

DST：　　memory, register.

CNT：　　1, CL.

功能：



Flags：　　CF, OF .

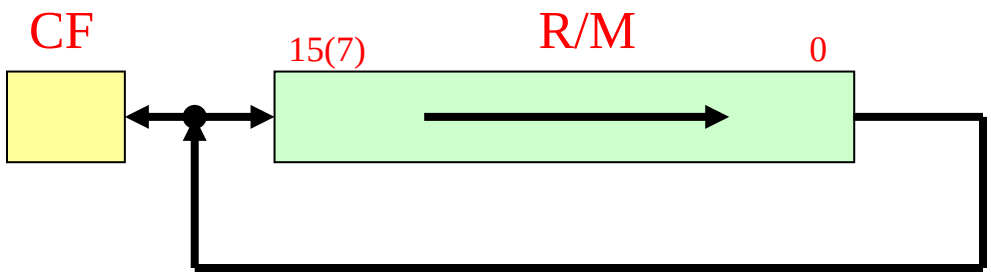
ROR

格式：　　ROR DST, CNT

DST：　　memory, register.

CNT：　　1, CL.

功能：



Flags：　　CF, OF .

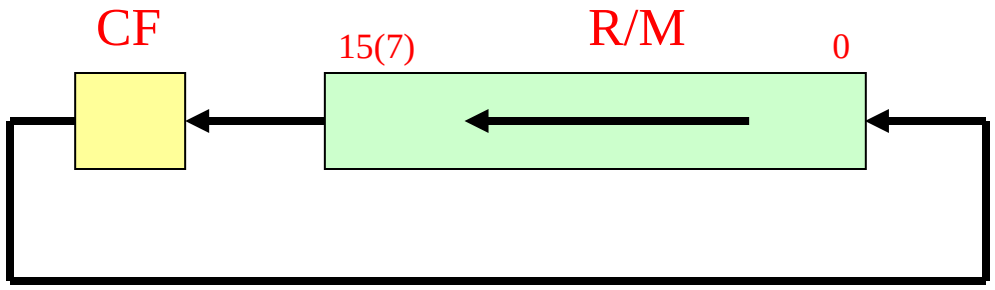
RCL

格式：　　RCL　DST, CNT

DST：　　memory,　register.

CNT：　　1,　CL.

功能：



Flags：　　CF, OF .

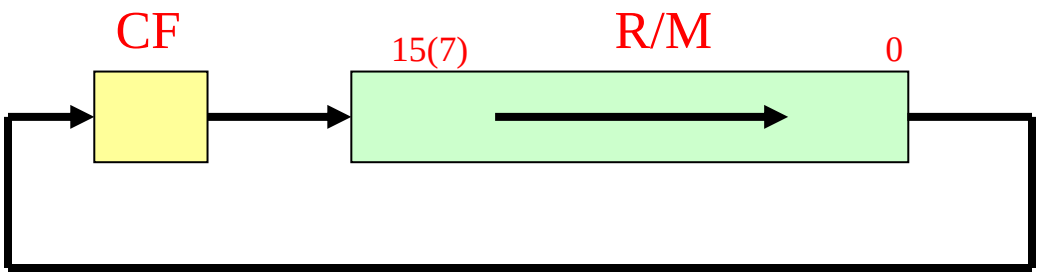
RCR

格式：　　RCR　DST, CNT

DST：　　memory,　register.

CNT：　　1,　CL.

功能：



Flags：　　CF, OF .

JMP

格式：　　JMP label　　IP

功能：　　无条件地转移到目标单元.

Flags：　　No influence.

Example：在偏移地址为0035H的内存单元有一条两字节的短转移指令JMP SHORT ADDR，如果其中位移量为①14H、②ECH，问目标地址的偏移量？

条件转移指令

格式：　　J*　label

功能：　　满足条件'*', 则转移到目标单元.

Flags：　　No influence.

修改IP；　　转移范围：-128~127

条件转移指令——单个标志

标志	指令助记符	测试条件	指令功能
CF	JC	CF=1	Jump if carry set
	JNC	CF=0	Jump if no carry
ZF	JZ/JE	ZF=1	Jump if equal / jump if zero set
	JNZ/JNE	ZF=0	Jump if not equal / jump if not zero
SF	JS	SF=1	Jump if sign set / jump if negative
	JNS	SF=0	Jump if no sign / jump if positive
OF	JO	OF=1	Jump if overflow set
	JNO	OF=0	Jump if no overflow
PF	JP/JPE	PF=1	Jump if parity set / Jump if parity even
	JNP/JPO	PF=0	Jump if no parity / Jump if parity odd

条件转移指令

A—Above, B—Below, E—Equal, N—not

G—Greater than, L —Less than, E—Equal, N—not

	指令	测试条件	指令功能
无符号数	JA/JNBE	CF∨ZF=0	Jump if above / not bellow or equal (a>b)
	JAE/JNB	CF=0	Jump if above or equal / not bellow (a≥b)
	JB/JNAE	CF=1	Jump if bellow / not above or equal (a<b)
	JBE/JNA	CF∨ZF=1	Jump if bellow or equal / not above (a≤b)
有符号数	JG/JNLE	(SF∀OF) ∨ZF=0	Jump if greater than/ not less than or equal (a>b)
	JGE/JNL	SF∀OF=0	Jump if greater than or equal / not less than (a≥b)
	JL/JNGE	SF∀OF=1	Jump if less than/ not greater than or equal (a<b)
	JLE/JNG	(SF∀OF) ∨ZF=1	Jump if less than or equal / not greater than (a≤b)

条件转移指令

Example： 设指令CMP AX，BX后跟着一条J* NEXT条指令，其中*为GE、LE、AE、NBE、BE，如果AX和BX的值分别如下：

AX	4000H	2000H	F000H	F000H
BX	4000H	3000H	E000H	2000H

判断对每组给出的AX和BX数据，使用哪几种条件转移指令可引起程序转移到NEXT地址.

Example： 符号函数Y＝f(X)，设任意给定的X值存放在XX单元，函数Y的值存放在YY单元，试编写一段指令序列实现，根据X的不同取值给Y赋值。

$$Y=\begin{cases}1 & \text{当}X>0\\ 0 & \text{当}X=0\\ -1 & \text{当}X<0\end{cases}$$

LOOP/LOOPZ/LOOPNZ

格式：　　LOOP label

功能：　　CX-1→CX

CX≠0则转移至label，CX=0退出循环.

指令助记符	循环退出条件
LOOP	CX=0
LOOPE/LOOPZ	CX=0 or ZF=0
LOOPNE/LOOPNZ	CX=0 or ZF=1

CALL

格式： CALL PROCEDURE

功能：断点地址入栈;子程序入口地址装入(CS:)IP.

near CALL:

SP-2→SP,断点地址入栈;子程序入口地址→IP .

far CALL:

SP-4→SP,断点地址入栈;子程序入口地址→CS:IP.

Flags： No influence.

RET

格式：　　RET

功能：　　SS:[SP]→IP, SP+2→SP.

　　&　　SS:[SP]→CS, SP+2→SP. (far)

格式：　　RET　disp16

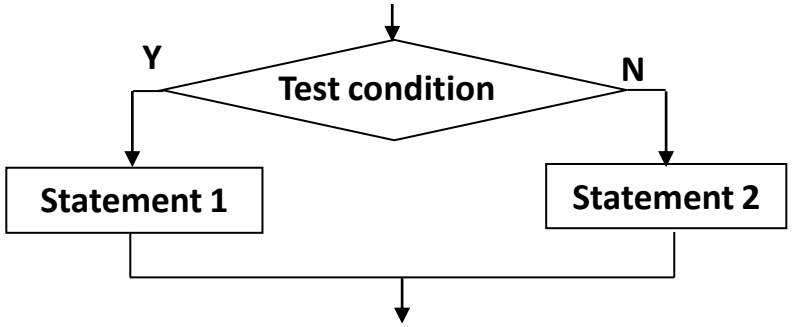
功能：　　RET;SP+disp16→SP.

Flags：　　No influence.

汇编语言程序控制流程

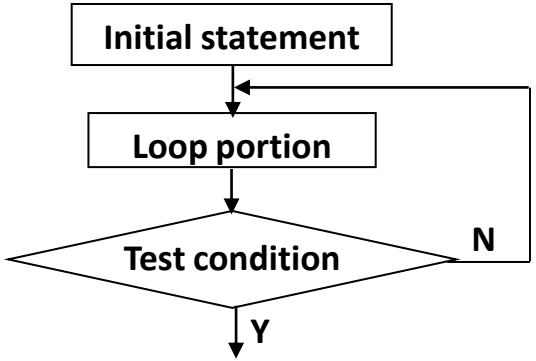
- 分支程序

IF－THEN－ELSE

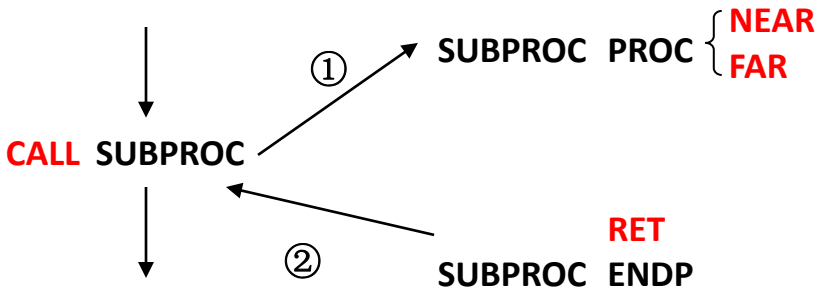


- 循环程序

DO－UNTIL loops



- 子程序



串操作指令

段地址：　源串——DS, 可段超越；目的串——ES.

偏移地址：　Source——SI；Destination——DI.

元素个数：　CX

处理方向：　DF

地址改变：　SI, DI 自动修改.

重复前缀：　REP/REPE/REPNE

	BYTE	WORD
DF=0	SI＋1 DI＋1	SI＋2 DI＋2
DF=1	SI－1 DI－1	SI－2 DI－2

Instruction Code	Condition for repeat	Condition for Exit
REP	CX≠0	CX=0
REPE/REPZ	ZF=1且CX≠0	CX=0或ZF=0
REPNE/REPNZ	ZF＝0且CX≠0	CX=0或ZF=1

MOVS/MOVSB/MOVSW

格式：　　MOVSB

　　MOVSW

功能：　　DS:[SI] →ES:[DI];

　　SI , DI自动修改.

Flags：　　No influence.

Example：把内存中数据段中以SRC开始的字符串”HELLO！”传送到附加段中以DST开始的单元中.

CMPS/CMPSB/CMPSW

格式：　　CMPSB

　　CMPSW

功能：　　DS:[SI] — ES:[DI];

　　SI , DI自动修改..

Flags：　　AF, CF, OF, PF, SF, ZF.

Example：　内存中PASSWORD处放有密码，INWORD处为键盘输入密码，检查输入密码是否正确，正确在RESULT处存1，不对存 -1.

SCAS/SCACB/SCASW

格式：　　SCASB

　　SCASW

功能：　　AL (or AX) — ES:[DI]

　　修改DI.

Flags：　　AF, CF, OF, PF, SF, ZF .

Example：　在BLOCK开始的数据块中查找字母”M”，若没找到，DI＝0；若找到，DI存查找次数.

LODS/LODSB/LODSW

格式：　　LODSB

　　LODSW

功能：　　DS:[SI] → AL (or AX)

　　修改SI.

Flags：　　No influence.

STOS/STOSB/STOSW

格式：　　STOSB

　　STOSW

功能：　　AL (or AX) → ES:[DI]

　　修改DI.

Flags：　　No influence.

Example：　设ES=3000H，DI=0200H，数据区长度100字节，对该数据区清0.

处理器控制指令

- 标志操作指令

  Carry Flag：　CLC, CMC, STC

  Direction Flag：CLD, STD

  Interrupt Flag：CLI, STI

- 停机指令　HLT
- 空操作指令　NOP