Ppt1：

后顾：被除数是除数宽度的两倍，积是两个乘数宽度的两倍

逻辑指令：

逻辑指令包括逻辑运算和移位指令

1. 逻辑运算指令

逻辑运算是按位操作的，包括，AND,OR,NOT,XOR,和TEST指令

1. And 目的操作数， 源操作数

功能：目的操作数and源操作数 = 目的单元

操作数寻址方式： reg, reg/mem/imm

Mem, reg/imm

根据结果置SF,ZF,PF, cf,of由于没有进位所以肯定是0，af没有定义，不一定

1. OR 目的操作数，源操作数

功能：目的操作数or源操作数 = 目的单元

寻址方式和and相仿，

Eflags也和and相仿

1. XOR 目的操作数，源操作数

功能：目的操作数xor 源操作数 = 目的单元

寻址方式和结果eflags都和and和or相仿

1. NOT 目的操作数

功能：按位取反，结果放在目的操作数的目的单元

寻址方式：reg/mem

不影响标志位

1. TEST 目的操作数，源操作数

功能：按位逻辑与进行测试，根据目的操作数与源操作数的逻辑运算结果设置标志位

寻址方式和and一样

但是不保存结果，只是置标志位

总结：1. 将指定位清0，用and，2. 将指定位置1，用or，3. 指定位取反，用xor

指定位取反时，需要变反的位赋1，不变的位赋0

1. 移位指令

按规定的方式，对目的操作数执行向左或者向右移动的若干个二进制位的操作

1. SHL 目的操作数，移动位数

功能：目的操作数逻辑左移，最后移出的位进入CF，最低位用0填充

操作数寻址方式为： SHL reg/mem，imm/CL，而imm只能是1

如果需要移动比1大的数字，就把那个数字放到CL里面

影响cf，of，sf，zf，pf，但是af不确定

左移一位之后，若符号为改变,OF = 1，否则OF = 0

1. SHR 目的操作数，移动位数

功能：目的操作数右移，最后移出的位进入CF，最高位用0填充

操作数寻址方式和SHLxiangfang

影响标志和SHL相仿

OF在右移一位的时候才有效，否则不确定

右移一位后如果符号改变，OF = 1，否则OF = 0

1. SAL 目的操作数，移动位数

功能：算术左移指令

SAL和SHL同一条指令，只是名字不同

1. SAR 目的操作数，移动位数

功能：和SHR差不多，但是高位用符号为填充

1. ROL 目的操作数，移动位数

循环左移，最后移出的位到CF里面

1. 同理ROR循环右移
2. RCL 目的操作数，移动位数

功能，目的操作数和cf一起循环左移

1. RCR和RCL相仿，带进位的循环右移（为什么这就是带进位的）
2. IN 累加器，端口地址

功能：1/2/4字节端口数据 -》 AL/AX/EAX

——端口范围：000h-ffffh

——Imm8（立即数八位）型式：指令端口范围：00h-ffh

——用DX存放端口，指令中的端口：0000h-ffffh

——和内存操作数不同的是，in指令的源操作数据来自IO端口

1. OUT 端口地址，累加器

功能：AL/AX/EAX -》 1/2/4字节端口数据

就是in的反过来

用海伦公式计算三角形面积

DATA SEGMENT

P DW ?

A DW 3

B DW 4

C DW 5

P1 DW ?

P2 DW ?

P3 DW ?

S DW ?

DATA ENDS

CODE SEGMENT

ASSUME CS:CODE, DS:DATA

MAIN:

MOV AX, DATA

MOV DS, AX

MOV AX, A

ADD AX, B

ADD AX, C

SAR AX, 1 ; 算术右移

MOV P, AX

MOV DX, AX

SUB DX, A

MOV P1, DX

MOV DX, AX

SUB DX, B

MOV P2, DX

MOV DX, AX

SUB DX, C

MOV P3, DX

MOV AX, P

MUL P1

MUL P2

MUL P3

MOV S, AX

MOV AX. 4C00H

INT 21H

CODE ENDS

END

上机题:

MOV AX, A

IMUL B

MOV DI, AX

MOV AX, C

MOV BX, D

ADD BX, AX

MOV AX, DI

CWD

IDIV BX ;商在ax，余数在dx

MOV SI, AX ;AX现在为商

MOV BX, A

SUB BX, B

MOV AX, 5

IMUL BX

SAR AX, 1

ADD SI, AX ;前面两项相加

MOV AX, C

ADD AX, D

ADD SI, AX ;前面三项相加

MOV AX, SI

1. 几个DOS系统功能调用
2. 01h 输入一个字符，有回显

入口 ah = 01h

出口 al = 输入字符的ascii码

1. 02h 输出一个字符

入口 ah = 02h

Dl = 待输出字符的ascii码

出口 无

指的是从键盘输入一个字符到cl（显示）

MOV AH, 01H

INT 21H

MOV CL, AL

MOV AH, 02H

MOV DL, CL

INT 21H

1. 忘了
2. 09h 输出一个字符串

入口：ah = 09h

Ds：dx = 字符串地址（‘$’作为结束标志）

出口：无

MOV AH, 9

MOV DX, OFFSET S

INT 21H

1. 0Ah 读入字符串（以回车结束，有回显）

入口：AH = 0Ah

Ds：dx = 输入缓冲区地址（字节0允许字符数）

出口：输入缓冲区字节1：字符数，字节2：字符串

MOV AH, 0AH

MOV DX, OFFSET BUF

INT 21H

1. 4Ch
2. 转移类指令：

CMP X, 0

JNG LOC2

MOV X, 256

JMP LOC3

LOC2:

MOV X, 512

LOC3: