汇编重点

题型：

1. 填空题：20x2=40分
2. 定义数据段：5分（5个变量）

   字符型变量

   字变量/字节（不同进制）

   相同初值
3. 画数据段xx空间图（5分）

   不同类型变量

   不同进制数赋初值

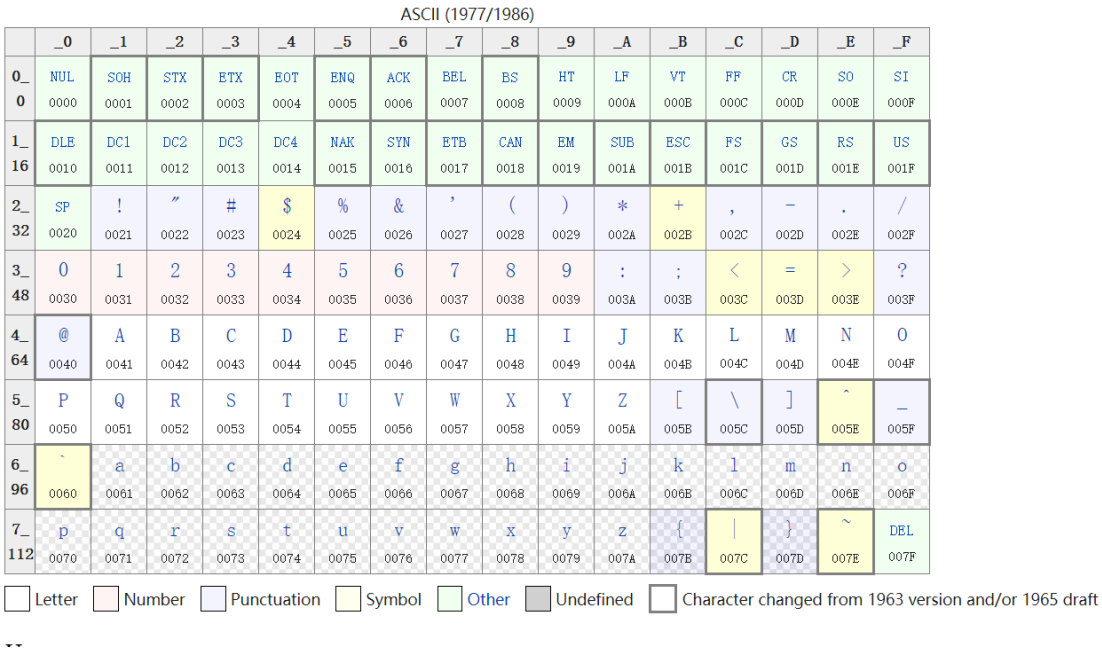
   变量赋初值
4. 阅读程序：4x7=28分: 最终结果(不同指令)
5. 编程（22分）两题

   Hello World

第一章 汇编语言基础

- 进制转换：10进制→2、8、16进制

  书写方法：后缀法（P29
- 原码、反码、补码：知道补码写真值
- ASCII：20H，30H



第二章 8086微处理器

- 组成：EU 3部件: 寄存器组+算术逻辑部件+标志寄存器(又称程序状态寄存器 **P6**

BIU 5部件: 地址加法器、段寄存器、指令指针IP、指令队列和总线控制逻辑 **P7**

- 寄存器:
  - 8 : (8个16位**通用寄存器**) : 包括数据寄存器(AX，BX，CX，DX)和地址寄存器(SI，DI，BP，SP)两种。各个寄存器都有自己特定的功能，在程序中常用来临时存放中间结果。 **P15**
  - 2: (2个专用寄存器):
    - IP(Instruction Pointer)为指令指针寄存器，又称指令指针，用于存放代码段中汇编指令的入口地址(即偏移地址)
    - FLAGS为标志寄存器，又称程序状态寄存器(Program Status Word, PSW)，共16位，用于存放当前程序指令执行的状态和运算结果的特征。标志寄存器中的每一位又称标志位，8086使用其中的9个标志位，各标志位分布如下图所示:

标志寄存器中标志位分布图

可分为条件标志位（CF—进位标志、ZF—零标志、SF—符号标志、OF—溢出标志、PF—奇偶标志、AF—辅助进位标志）和控制标志位（TF—单步中断允许标志、IF—外中断屏蔽标志、DF—方向标志）

  - 4： (4个段寄存器）：
    - 代码段CS，用来存放当前正在运行的程序代码。
    - 数据段DS，用来存放当前运行程序所用的数据，如果程序中使用了串处理指令，则其源操作数必存放在数据段中
    - 堆栈段SS，堆栈段定义了堆栈的所在区域，堆栈是一种数据结构，是一个特殊的数据存储区，并以先进后出的方式来访问数据，它常用来存放子程序的入口地址。
    - 附加段ES，附加段是附加的数据段，它是一个辅助的数据存储区，也是串处理指令的目的操作数存放区。

- 存储器：字节：低字节低地址，高字节高地址 **P27**
- 物理地址(寻址形式):

  物理地址 = 段地址×10H + 偏移地址

  PA=(CS)×10H+(IP)

  PA = (DS或ES)×10H十该存储单元的偏移地址

  PA = (SS)×10H+(SP)

- Debug调试工具命令:
  - R: “R”命令后，即可显示出各寄存器的名称及内容。“R”是检查和修改寄存器内容的命令
- U: 输入D和U命令后，即可查看内存单元的内容
  - E: 输入E命令后，即可修改内存单元的内容
  - T: 跟踪命令, 每执行一条指令就显示运行结果，使程序员可以细致地观察程序的执行情况。
    从指定地址起执行一条或数值参数指定条数的指令后停下来，每条指令执行后都要显示所有寄存器和标志位的值以及下一条指令。如未指定地址则从当前的CS : IP开始执行。注意给出的执行地址前有一个等号，否则会被认为是被跟踪指令的条数（数值）。
  - G: 运行命令, 从指定地址处开始运行程序，直到遇到断点或者程序正常结束。
    程序遇到断点(实际上就是断点中断指令INT 3)，停止执行，并显示当前所有寄存器和标志位

的内容、以及下一条将要执行的指令(显示内容同R命令)，以便观察程序运行到此的情况。程序正常结束，将显示“Program terminated normally”

- 堆栈：先进后出/指针变化　**SP栈顶指针**
- 程序示例：背下来　　**P48**

汇编语言实现字符串显示1

```
DATA    SEGMENT          ；数据段
BUF     DB  0AH, 0DH, " HELLO WORLD ! $"
DATA    ENDS
STACK   SEGMENT  STACK ；堆栈段
        DB  200  DUP(0)
STACK   ENDS
CODE    SEGMENT          ；代码段
ASSUME  CS:CODE, DS:DATA
  START: MOV   AX,  DATA ；将数据段首址
DATA→AX(MOV为传送指令)
          MOV   DS,   AX  ；将数据段首址DATA置入数据
段寄存器DS
```

汇编语言实现字符串显示2

```
        LEA   DX,  BUF ；将变量BUF的偏移地址传给DX
          MOV   AH,  9    ；将立即数9传给AH
          INT  21H         ；DOS中断调用
          MOV   AH , 4CH ；4CH→AH
          INT   21H        ；DOS中断调用，这两条指令执行
完后，计算机将结
                          ；束本程序的运行，返回DOS状态
  CODE   ENDS
        END   START
```

第三章 寻址方式

- 有效地址组成：　**P4**

  EA = 基址 + (变址×比例因子) + 位移量（X）

  (除比例因子是固定值外，其他三个成分都可正可负，以保证指针移动的灵活性

1. 位移量(displacement)是存放在指令中的一个8位、16位和32位的数，但它不是立即数，而是一个地址。
2. 基址(base)是存放在基址寄存器中的内容。它是有效地址中的基址部分，通常用来指向数据段中数组或字符串的首地址。
3. 变址(index)是存放在变址寄存器中的内容。它通常用来访问数组中的某个元素或字符串中的某个字符。
4. 比例因子是386及其后继机型中新增加寻址方式中的一个术语，其值可为1，2，4或8。在寻址中，可用变址寄存器的内容乘以比例因子来取得变址值。这类寻址方式对访问元素长度为2，4，8字节的数组特别有用

- 指令寻址：

  寻址方式是什么： 寻找数据和指令存放地址的方式称为寻扯方式

  - 立即寻址
  - **直接寻址**
  - **寄存器寻址**
  - **寄存器间接寻址**
  - **寄存器相对寻址**
  - **基址加变址寻址**
  - **相对基址加变址寻址**
  - 相对比例变址寻址方式
  - 相对基址比例变址寻址方式

  段内直接寻址

  段内间接寻址

- 操作数存放在哪里： 物理地址是由段地址和偏移地址两部分组成，段地址存放在相应的段寄存器中，偏移地址存储在相应的地址寄存器中，偏移地址又称有效地址

物理地址计算：

寄存器与存储区数据存放位置：

CS：IP怎么变化，值是多少

跨段的有关问题：

第四章 汇编语言程序

- 语句类型：
  - 指令语句：指令语句是可执行语句，在汇编时可产生供机器执行的二进制目标代码
  - 伪指令语句：伪指令语句是不可执行语句，在汇编时不产生目标代码，汇编程序主要利用它分配存储单元和定义程序段等
  - 宏指令语句：要求先定义后使用，它是一个宏名代替宏体的过程。（后面章节详细介绍）
- 语句格式：指令语句和伪指令语句具有相似的语句格式，都由4部分组成，一般格式为：

  [<名字>] <操作码> [<操作数>] [;<注释>]

  []可选，<>必选，[<>]可选中的必选，|或选，前后内容必选其一

  在汇编语言中，允许使用如下语言成分：

  - 字母：a～z A～Z
  - 数字：0～9

    - 字符：？；　：　，　@ $ [ ]
- 常量类型：数值型常量、符号常量和字符型常量

  字符型常量与数值型常量可以相互通用

- EQU “=”

  对经常引用的数值型常量，可以用等价伪指令EQU或等号伪指令“=”给它定义一个名字，然后在语句中用这个名字来代表该常量。这个名字称为符号常量

- 变量属性：段属性、偏移属性、类型
- 变量赋初值

  1. 数值表达式
  2. ASCII字符串（通常选用DB类型
  3. 地址表达式(只适用于DW和DD两个伪指令)；DW：偏移地址，DD：段首址和EA
  4. ？(表示所定义的变量无确定初值)
  5. 重复子句。其格式为：n DUP（表达式）
  6. 可以是以上表达式组成的序列：逗号隔开，变量名是首地址，表达式个数即存储单元个数
- 标号类型

  NEAR类型: 只能在定义该标号的段内引用，且可以省略

  FAR类型: 标号段间引用，不可省

- 数值表达式：由各种常量与数值运算符连接而成的式子，称为数值表达式。数值表达式的计算结果是一个数值，它只有大小而没有属性
- 地址表达式：由常量、变量、标号、寄存器和数值运算符、地址运算符组合而成的有意义的式子，称为地址表达式，单个变量、标号是地址表达式的最简形式
- 地址运算符：属性分离算符、属性定义算符
    - 高低? 分离 16位 定义

- 8086微处理器的指令系统大约具有100条基本指令，这些指令可以分为以下六类
    - 数据传送指令：
        - 数据传送指令：**MOV**、XCGH
        - 堆栈操作指令：PUSH、POP
        - 标志寄存器：PUSHF、**POPF**、LAHF、**SAHF**
        - 地址传送指令：**LEA、LDS**、LES
        - 输入输出指令：IN、OUT

      除SAHF、POPF两指令外，其他均不影响标志位
    - 算数运算指令：
        - 加法：**ADD、ADC(带进位加法)**、INC
        - 减法：DEC、SUB、**CMP(很重要)、NEG(求补？)、SBB(借位减法)**
        - 乘法指令：**MUL** 、IMUL(有符号乘法) 另一个操作数
        - 除法指令：**DIV** 、IDIV 另一个操作数
        - 符号扩展指令：**CBW**、CWD、CDQ
    - 位操作指令：
        - 逻辑运算指令：AND（屏蔽？），OR，NOT（按位取反）XOR（某些位取反），TEST（相与，结果不保存，根据其特征设置条件码）
        - 移位指令：

          算数移位指令：

          算数左移和逻辑左移：SAL、SHL

算数右移：SAR

逻辑右移：SHR

循环移位指令：

循环左移指令 ROL
循环右移指令 ROR
带进位的循环左移指令 RCL
带进位的循环右移指令 RCR

- CF　SAL/R SHL/R ROL/R(理解含义)

- 控制转移指令：
  - 无条件转移指令：JMP <转向地址>
  - 条件转移
  - 循环
- 其他
- 处理机控制指令
  1. 标志处理指令: 设置和清除CF、DF、IF三个标志位
  2. 其他处理机指令

- 循环：LOOP

  其他告诉意思?

- 伪指令 $

第五章 汇编语言程序设计

- 分支结构程序设计
- 循环结构程序设计
- 常用DOS中断调用
- 程序调用、返回

参数传递方法：1、2、9、10号调用意思

第六章 汇编语言高级编程

- 宏定义:

```
<宏名> MACRO [<形式参数表>]
        <语句1>
          ┆          宏体
        <语句n>
ENDM

```

- 结构定义

```
;结构定义格式为:
<结构名> STRUC
<字段说明>
<结构名> ENDS
```

- 伪指令

第七章 IO程序设计

- 端口三种类型: 数据端口、控制端口和状态端口
- 4指令:
  - IN
  - OUT
  - INS
  - OUTS

第八章 中断及中断系统

例8.4