**汇编作业四**

2151644 周一韬

Github代码仓库地址：https://github.com/FDWJTZSBJEP/2023TJ\_Assembly

**一 宏汇编的使用**

要求：用macro实现替换过程调用

目的：理解并运用宏汇编，比较宏汇编与过程调用的区别与共性，进一步理解汇编的底层逻辑

**1 概念解释**

C中的define可以替换掉一部分东西，然后在实际的实现中将define的别名换成实际内容，最后程序才能跑起来，这里也一样，我们可以定义一个宏，然后在编程中直接调用宏，便于我们的编程，主要是减少代码的长度。在最后的功能实现上，宏和过程调用实现的目的是类似的，但是在具体的实现过程上有所不同。

**2 思路分析**

为了理解宏的概念，这里我以第三次作业的第一题为例（即接收输入的时间并输出），将其中使用的两个过程调用改用宏来实现。宏的使用和过程调用很类似，需要先进行宏定义然后进行宏调用。

宏定义的实现如下，有参数和无参数的定义略有不同，将在源程序中多次使用到同一个程序段定义成一个“指令”

|  |
| --- |
| Assembly language ;无参数 宏名 macro  　　宏体  endm  ;有参数 宏名 macro 形参1,形参2,形参...  　　宏体  endm |

宏调用的实现如下所示，有参数和无参数的定义略有不同，用宏指令名来代替原来的程序段

|  |
| --- |
| Assembly language ;不带参数 宏名   ;带参数 宏名 实参1,实参2,实参.... |

可以看到在使用和过程调用非常类似，但是可以增加输入参数的部分。

**3 具体实现**

这里我选择的是在hw3\_1的基础上实现，即输入日期返回时间。

原程序使用了Disp和GetNum两个过程来实现，这里我将这两个过程都改用宏汇编实现

这里以Disp过程为例，修改后的代码如下

|  |
| --- |
| Assembly language Disp MACRO   ; 输出输出提示  MOV AH, 9  MOV DX, OFFSET OUTPUT\_STRING  INT 21H    MOV AH, 9  MOV DX, OFFSET STRING\_Y  INT 21H ;     MOV AH, 2  MOV DL, '-'  INT 21H   MOV AH, 9  MOV DX, OFFSET STRING\_M  INT 21H ;    MOV AH, 2  MOV DL, '-'  INT 21H   MOV AH, 9  MOV DX, OFFSET STRING\_D  INT 21H ;  ENDM |

然后我们在使用的时候，只需要将源程序call Disp的语句，修改为Disp即可，然后即可正常运行。

**4 心得体会**

宏和过程两者虽然最后实现的结果是一样的，但是在具体的实现上存在着些许不同。

过程在被调用的时候只有一份程序副本出现，而宏在被调用的时候，每一次对宏的调用都会出现一次宏代码的副本。过程通常会保存和恢复寄存器的内容，而宏通常不会去保存和恢复寄存器的内容。过程倾向于节省内存空间，而宏倾向于节省运行时间。

使用宏最大的好处是可以实现传参，但是在使用时不能调用原有寄存器中的值。而过程调用则正好之于相反，两者在编写汇编代码时是一个互补的关系，各有优势。

**二 多模块实现**

要求：把大文件拆分成多个源文件

目的：大型源文件难于管理且汇编速度慢，可以把单个文件拆分为多个子文件，但是，对其中任何子文件的修改仍需对所有的文件进行整体汇编。更好的方法是把一个程序按模块（module）（汇编单位）分割。每个模块可以单独汇编，因此，对一个模块源代码的修改就只需要重汇编这个模块。通过本次作业，学会使用多模块编程，理解其作用与好处。

**1 概念解释**

汇编多模块编程是指使用汇编语言编写的程序，由多个模块组成的一种编程方式。在这种编程方式中，程序被分割成多个独立的模块，每个模块负责完成特定的任务或功能。这种模块化的设计使得程序更加结构化、可维护性更高，并且有助于团队协作。

**2 具体实现**

本次作业是在上一次输出工资的基础上实现的，之前的这个程序比较复杂，适合我们利用多模块来拆分为多个子程序，简化每个程序的内容。

在上次的作业中，程序主要分为三部分，一个是主函数部分，负责初始化以及调用过程；一个是GetNum过程，将data表中的数据存放到table表中；一个是Disp过程，负责输出table表中的数据，实现可视化输出。这里也基于这种想法，将原本的 hw3\_2.asm 文件分为hw4\_2.asm（主程序），hw4\_2-1.asm（GetNum过程）,hw4\_2-2.asm(Disp过程) 然后即可实现

最后在编译运行的时候，需要先把这三个asm文件利用masm指令转换为.obj文件，然后在link的时候可以用“link a b c”的方法，link多个文件，最后生成一个可执行的hw4\_2.exe文件，其结果和上一次的作业相同。

**3 心得体会**

在汇编中，多模块编程是一个很有用的技巧，可以极大简化我们原本复杂的程序，在分为多个子程序后实现原本的功能。这种思想也类似于高级语言程序编程中的多模块思想，虽然目前来看所写的程序还没有必要使用这样的方法，但是相信之后遇到更复杂的程序，使用这种方法是很有必要的。