



哈爾濱工業大學（深圳）

Harbin Institute of Technology, Shenzhen

课程实验报告

课程名称：汇编语言程序设计

实验名称：实验三

学生所在院（系）：计算机科学与技术学院

学号：190110716

学生姓名：朱海峰

上交日期：2020.12.13

指导老师：裴文杰

目 录

1. 问题描述（将题目转换为计算机问题进行描述）	1
2. 解决方案（包括程序具体设计和流程图）	1
3. 具体实现（源代码和程序调试步骤、程序运行截图）	2
4. 总结（实验中遇到的问题和收获）	4

1. 问题描述（将题目转换为计算机问题进行描述）

题目：

将数据段的 3 个字节类型无符号数分别转换成二进制和十六进制，并显示在屏幕上。

要求：程序必须用到宏或子程序。

计算机问题：

将字节类型的无符号数放入 AX 寄存器后，通过子程序的累除取余得到不同进制的各位数，压栈后逐个输出即可

2. 解决方案（包括程序具体设计和流程图）

程序中用到子程序和宏。

子程序实现的功能是将 NUMBER[BX] 通过累除得到各个余数压栈转化为 K 进制数，弹出并显示出来。宏实现换行回车功能。

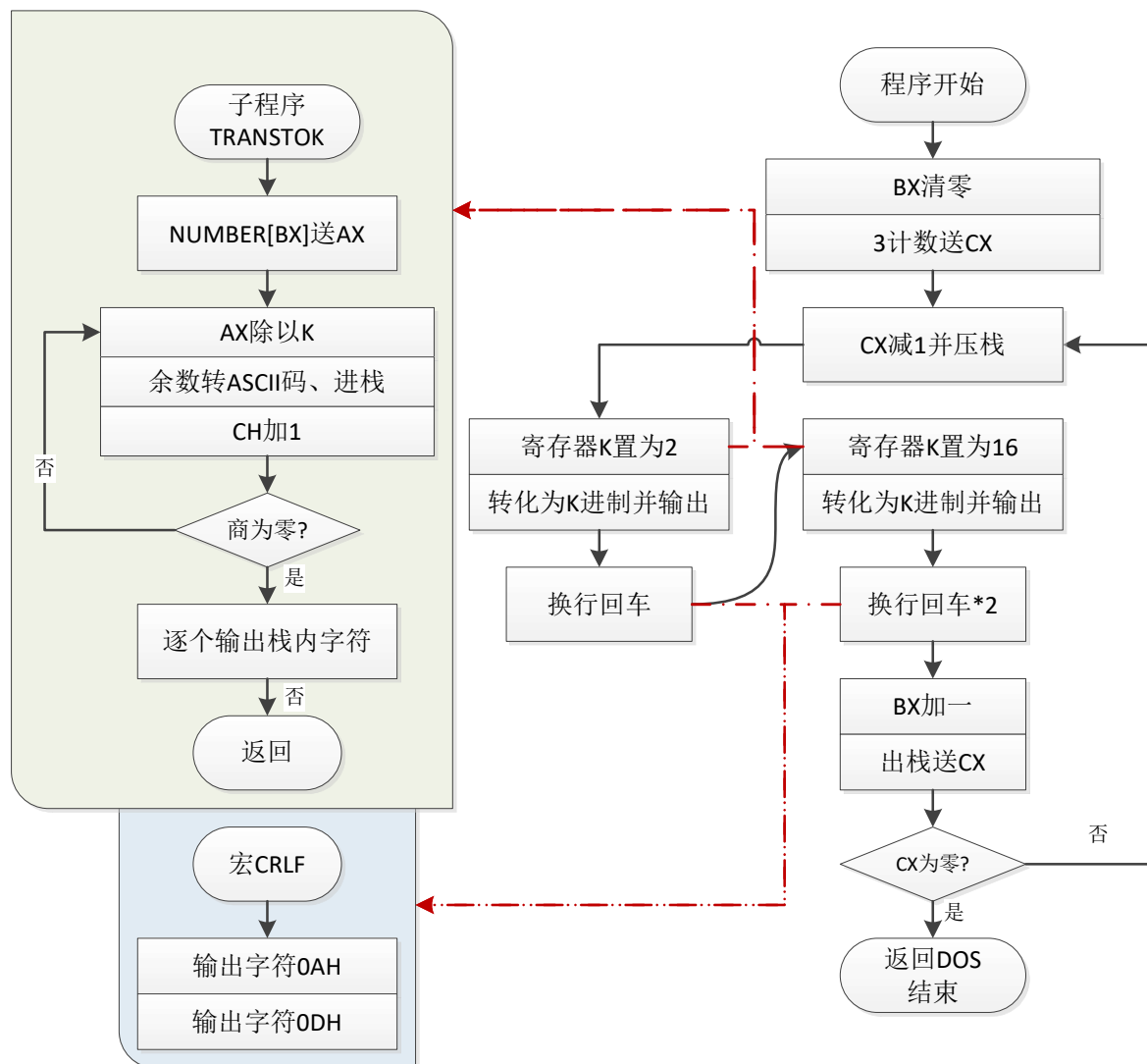


图 1

3. 具体实现（源代码和程序调试步骤、程序运行截图）

源代码：

```

;-----换行回车
CALF MACRO
    PUSH DX
    MOV DL,0AH
    MOV AH,2
    INT 21H
    MOV DL,13
    MOV AH,2
    INT 21H;
    POP DX
ENDM

DATAS SEGMENT
    NUMBER DB 117,36,77
    INPUT DW 0
    K DB 16
DATAS ENDS

STACKS SEGMENT
    DW 20 DUP(?)
STACKS ENDS

;-----主程序
CODES SEGMENT
    ASSUME CS:CODES,DS:DATAS,SS:STACKS
START:
    MOV AX,DATAS
    MOV DS,AX
    MOV AX,STACKS
    MOV SS,AX
    XOR BX,0
    MOV CX,3H      ;外部循环次数
AGAIN:
    PUSH CX
    MOV K,2
    CALL TRANSTOK ;转化为 2 进制显示
    CALF          ;换行回车
    MOV K,16
    CALL TRANSTOK ;转化 16 进制显示
    CALF
    CALF

    INC BX          ;下一个数
    POP CX
    LOOP AGAIN

    MOV AH,4CH
    INT 21H

;-----将 NUMBER[BX] 用 k 进制显示
TRANSTOK PROC

```

```

MOV AL,NUMBER[BX]
XOR CX,CX
CLASSIFY:
    XOR AH,AH ;置零
    DIV K
    ADD AH,30H ;转化 ASCII 码
    CMP AH,3AH ;区分字母与数字
    JL PUS
    ADD AH,7H
PUS:
    PUSH AX ;压栈
    INC CH ;记录字符数
    CMP AL,0 ;商为零则结束
    JNZ CLASSIFY
DISPLAY:
    MOV CL,8
    POP DX
    SHR DX,CL ;字符送 DL
    MOV AH,02H
    INT 21H
    DEC CH
    JNZ DISPLAY

RET
TRANSTOK ENDP
CODES ENDS
END START

```

调试步骤:

DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

0774:000A	83F300	XOR	BX,+00
0774:000D	B90300	MOV	CX,0003
0774:0010	51	PUSH	CX
0774:0011	C606050002	MOV	BYTE PTR [0005],02
0774:0016	E83A00	CALL	0053
0774:0019	52	PUSH	DX
0774:001A	B20A	MOV	DL,0A
0774:001C	B402	MOV	AH,02
0774:001E	CD21	INT	21
0774:000A	83F300	XOR	BX,+00
0774:000D	B90300	MOV	CX,0003
0774:0010	51	PUSH	CX
0774:0011	C606050002	MOV	BYTE PTR [0005],02
0774:0016	E83A00	CALL	0053
0774:0019	52	PUSH	DX
0774:001A	B20A	MOV	DL,0A
0774:001C	B402	MOV	AH,02
0774:001E	CD21	INT	21
0774:0020	B20D	MOV	DL,0D
0774:0022	B402	MOV	AH,02
0774:0024	CD21	INT	21
0774:0027	C606050010	MOV	BYTE PTR [0005],10

图 2 宏展开

```

DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0774:006A 50          PUSH    AX
-t
AX=3100 BX=0000 CX=0600 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0770 ES=0760 SS=0771 CS=0774 IP=006B  NU UP EI NG NZ AC PO CY
0774:006B FEC5          INC     CH
-t
AX=3100 BX=0000 CX=0700 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0770 ES=0760 SS=0771 CS=0774 IP=006D  NU UP EI PL NZ NA PO CY
0774:006D 3C00          CMP     AL,00
-t
AX=3100 BX=0000 CX=0700 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0770 ES=0760 SS=0771 CS=0774 IP=006F  NU UP EI PL ZR NA PE NC
0774:006F 75E8          JNZ     0059
-t
AX=3100 BX=0000 CX=0700 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0770 ES=0760 SS=0771 CS=0774 IP=0071  NU UP EI PL ZR NA PE NC
0774:0071 B108          MOV     CL,08
-d ss:ffe0
0771:FFE0  00 00 00 00 00 31 00 00-71 00 74 07 A6 01 00 31  .....1..q.t....1
0771:FFF0  01 31 03 31 07 30 0E 31-1D 30 3A 31 19 00 03 00  .1.1.0.1.0:1....

```

图 3 堆栈段数据存各位二进制字符

运行结果:

```

DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Pro...
1110101
75
100100
24
1001101
4D
Press any key to continue

```

图 4

4. 总结（实验中遇到的问题和收获）

本次实验中，对子程序、宏指令的设计与使用有了一些了解。此外，实验中涉及到转换进制问题。由于转换为 2 进制和十六进制，显然可以直接用移位操作得到各个位再转化 ASCII 码，这种子程序的实现用到了较多的参数，情况较多，考虑的很容易不周全，而且迁移性比较差。比如如果要转换其他进制就可能不适用，于是我选择的不是移位操作，而是直接用累除的方式，这样的话转换 16 以内的进制就都可行，只需要设置不同的 K 即可。调试过程中我在寻找堆栈段内的数据内容，发现 ss: 0 都没有数据，再结合 sp，sp 是个 fff-，发现堆栈的数据顺序是反过来存的。