**实验报告**

专业：电气工程及其自动化

姓名： 严旭铧

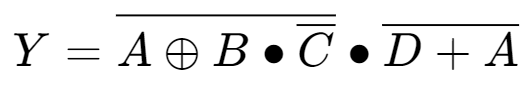
学号：\_3220101731

地点：紫金港东三406

课程名称：\_\_\_微机原理与应用综合实验\_\_\_\_指导老师：\_\_ 胡斯登\_\_\_ \_成绩：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

实验名称：\_\_\_\_\_\_\_\_BCD控制\_\_\_\_\_实验类型：\_微机实验\_\_\_\_\_同组学生姓名：褚玘铖

# 实验2 BCD控制

1. **题目要求**
   1. 30H与31H中存放4位BCD码数字，编写程序将数字倒序排列
   2. 在RAM 31H单元存放一组8位带符号数，字节个数放在30H中，请编写程序统计出其中正数，负数以及0的数目，结果存放在41H，42H以及43H中。
   3. 模拟下列逻辑运算编写程序并将运算结果转换为显示码后进行显示。设:A=63H、B=82H、C=0C5H、D=36H
2. **代码及结果实现**

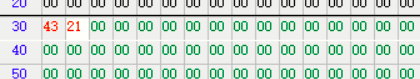
**#注：**代码用了截图和OpenDocument Text对象插入，后者双击即可编辑。其中注释部分有加 **#** 的部分是值得注意的地方。

1. **EX1**
   1. 代码



* 思路：
* 先各数倒序，再两数交换。利用XCH指令完成两个地址之间的内容互换，再利用SWAP对每个地址里面的2为BCD码进行互换。
* 需要注意，XCH和SWAP都是对A进行操作的指令，因此需要用A作为中介，将两个寄存器的内容互换，并且处理高低位。
  1. 结果

输入：30H输入12,31H输入34，预期输出为43 21

输出：

达到实验预期效果

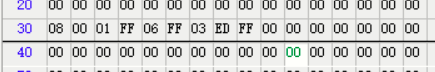
1. **EX2**
   1. 代码



* 思路：
* 正负判断：利用JB ACC.7,rel这条指令进行负数的判断。
* 0的判断：利用CJNE A,#0判断;也可使用 JZ rel判断。两种方法的逻辑是相反的。这两条指令是最核心的指令。

这里在写的时候没有使用RET。可以使用ACALL/LCALL+RET进行子程序的调用和返回，可以用RET结尾来实现跳转回去。

* 1. 结果

输入：8个带符号数，1个0,3个正数，4个负数，41H~43H应该分别为03,04,01

输出：

得到所求结果

1. **EX3**
   1. 代码



* 思路：
* 逻辑运算题需要考虑逻辑运算顺序，一般是从里到外，最后化为 A？Y的形式，进行最后一步的逻辑运算。逻辑运算不难，慢慢拆分组合就行。
* 本题还要求用显示码表示结果，但是这个显示码的意思我不是很清楚。我自己理解成给他可视化放入地址中。由于是逻辑运算，结果必然是8位的0或1。这里我选择把最后储存在A里面的结果（88H）逐位表示出来，在41H到48H中显示。这样每一位的逻辑取值会很清晰。这一步的核心是RLC A + JC PUT1指令，实现了逐位的01判断。也可以改成JNC，那就是反一下，CY为0跳转
  1. 结果

输入部分直接在代码了，也可以改成自己输入值再存进去

输出：

如果没算错的话最后的结果是（1000 1000）B，也就是88H

1. **心得体会**

BCD码显示可以将不太熟悉的16进制转换成十进制的形式显示出来，非常方便。这次三道题的核心我觉得是要熟练运用A和CY位，因为都是涉及到位的判断和处理的。JC等对位判断跳转的指令在这类场景下非常有用。