实验一 监控程序与汇编语言程序设计实验

1. **实验目的**

【1】学习和了解TEC-XP16教学实验系统监控命令的用法；

【2】学习和了解TEC-XP16教学实验系统的指令系统；

【3】学习简单的TEC-XP16教学实验系统汇编程序设计。

1. **实验内容、结果分析**

**【例1】**

程序代码：

2000: IN 81

2001：SHR R0

2002：SHR R0

2003：JRNC 2000

2004：IN 80

2005: OUT 80

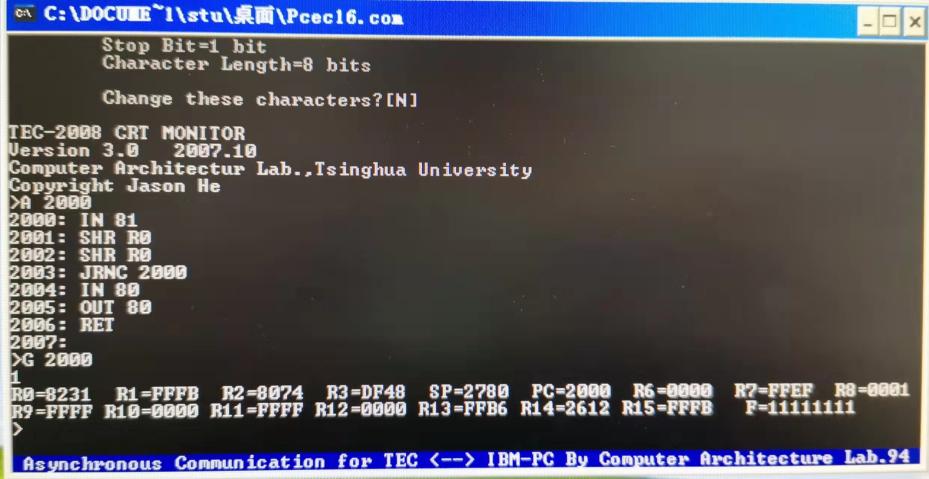
2006: RET

2007: ↙

实验数据：输入0

运行结果：在键盘上敲击0后，收到TEC的返回值0。还显示了各寄存器的值

实验现场图：



【例2】

程序代码：

2020:MVRD R2,000A

2022:MVRD R0,0030

2024:OUT 80

2025:DEC R2

2026:JRZ 202E

2027:PUSH R0

2028:IN 81

2029:SHR R0

202A:JRNC 2028

202B:POP R0

202C:INC R0

202D:JR 2024

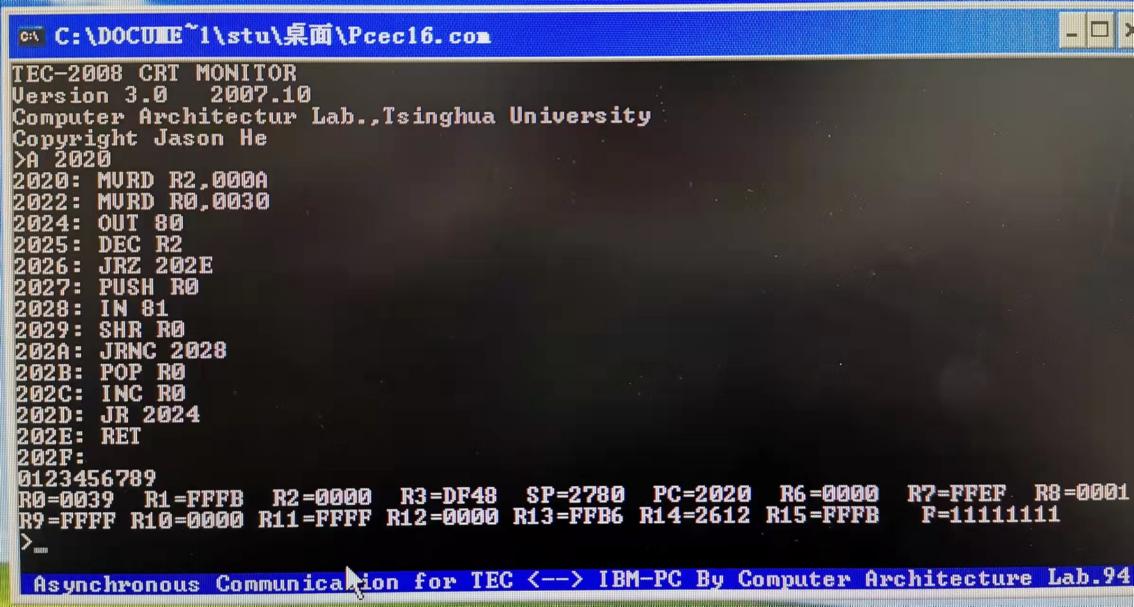
202E:RET

202F:↙

实验数据：无手动输入

运行结果：输出0123456789共10个数字

实验现场图：



【例3】

程序代码：

MVRD R2,0030

MVRD R3,0039

IN 81

SHR R0

SHR R0

JRNC 2044

IN 80

MVRD R1,00FF

AND R0,R1

CMP R0,R2

JRNC 2053

CMP R3,R0

JRNC 2053

OUT 80

JMPA 2044

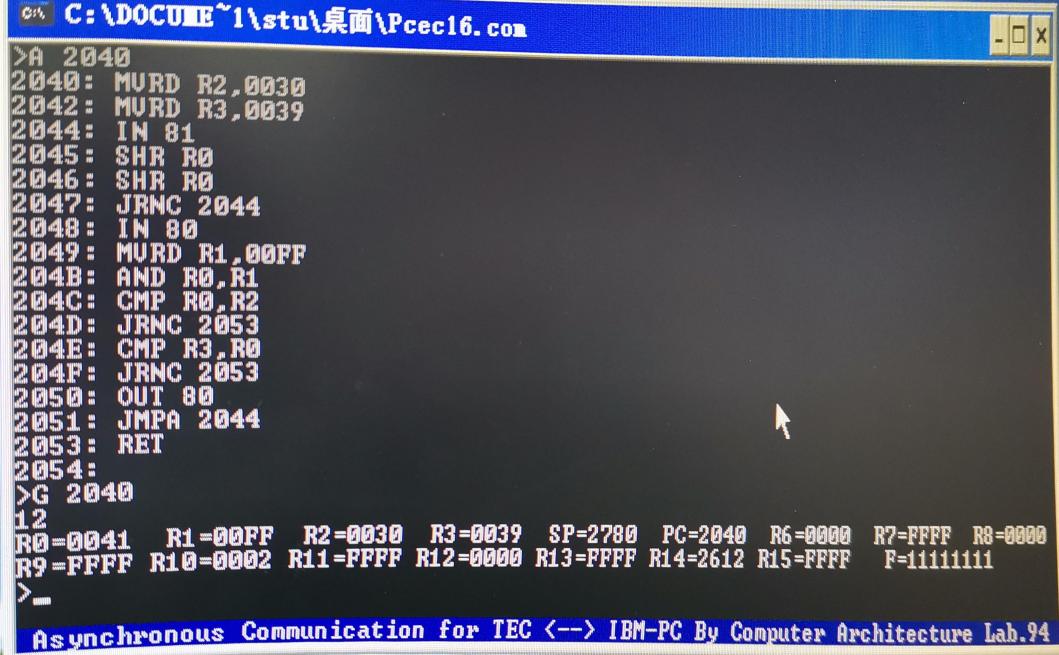
RET

↙

实验数据：输入12A

运行结果：输入1、2时会有回显，当输入A时直接退出输入状态

实验现场图：



【例4】

程序代码：

MVRD R1,0000

MVRD R2,00OA

MVRD R3,0000

INC R3

ADD R1,R3

CMP R3,R2

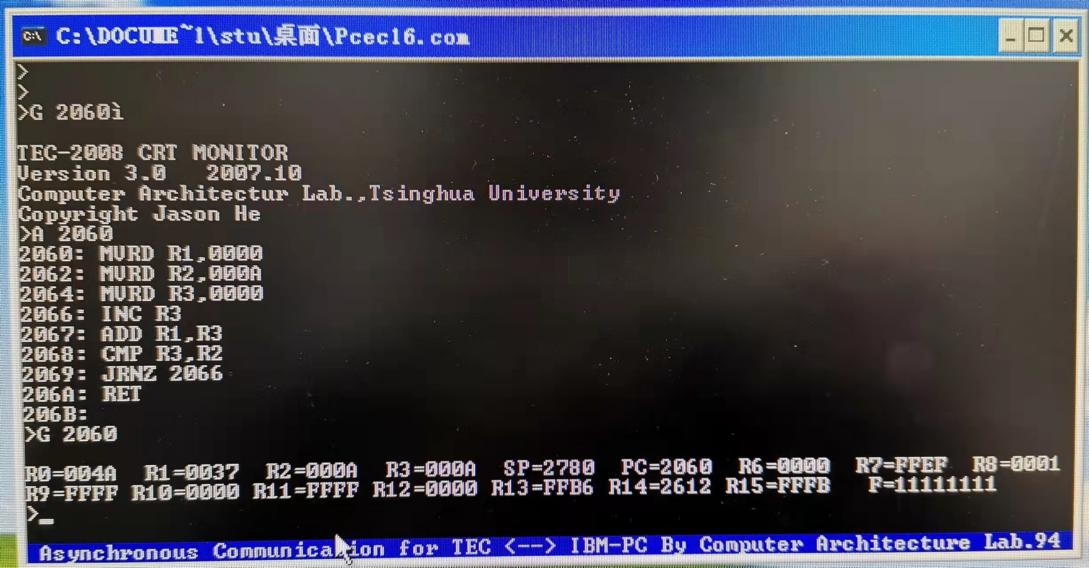
JRNZ 2066

RET

实验数据：无手动输入

运行结果：R2寄存器中存储了1到10的累加和

实验现场图：



【例5】

2080H开始

MVRD R3,0006

MVRD R2,20F0

LDRR R0,[R2]

CALA 2100

DEC R3

JRZ 208B

INC R2

JR 2084

RET

2100H开始

OUT 80

MVRD R1，0020

ADD R0，R1

STRR [R2],R0

IN 81

SHR R0

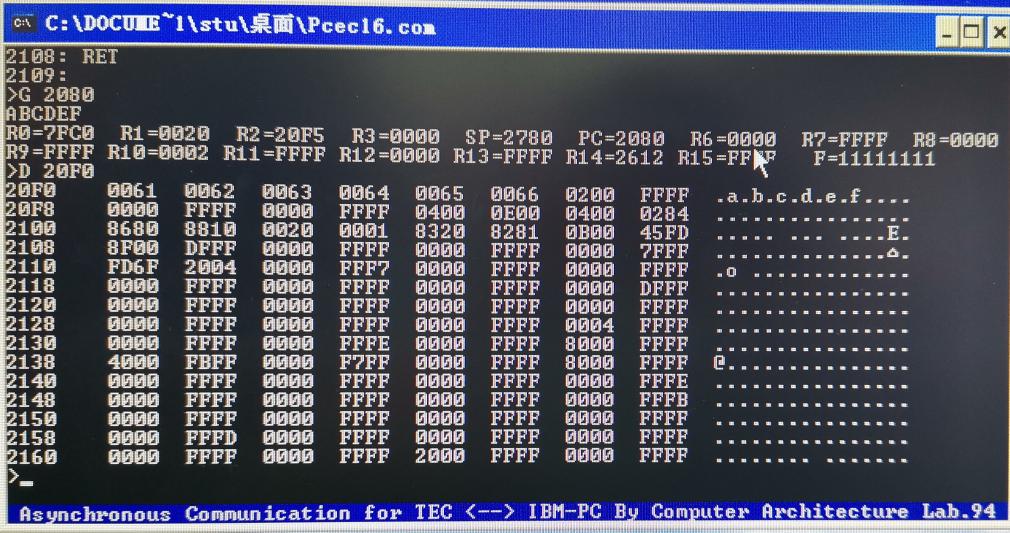
JRNC 2105

RET

实验数据：无手动输入

运行结果：输出了ABCDEF，运行完后内存中为abcdef

实验现场图：



【例6】

程序代码：

20A0: MVRD R1,7E

20A2: MVRD R0,20

20A4: OUT 80

20A5: PUSH R0

20A6: IN 81

20A7: SHR R0

20A8: JRNC 20A6

20A9: POP R0

20AA: CMP R0,R1

20AB: JRZ 20AE

20AC: INC R0

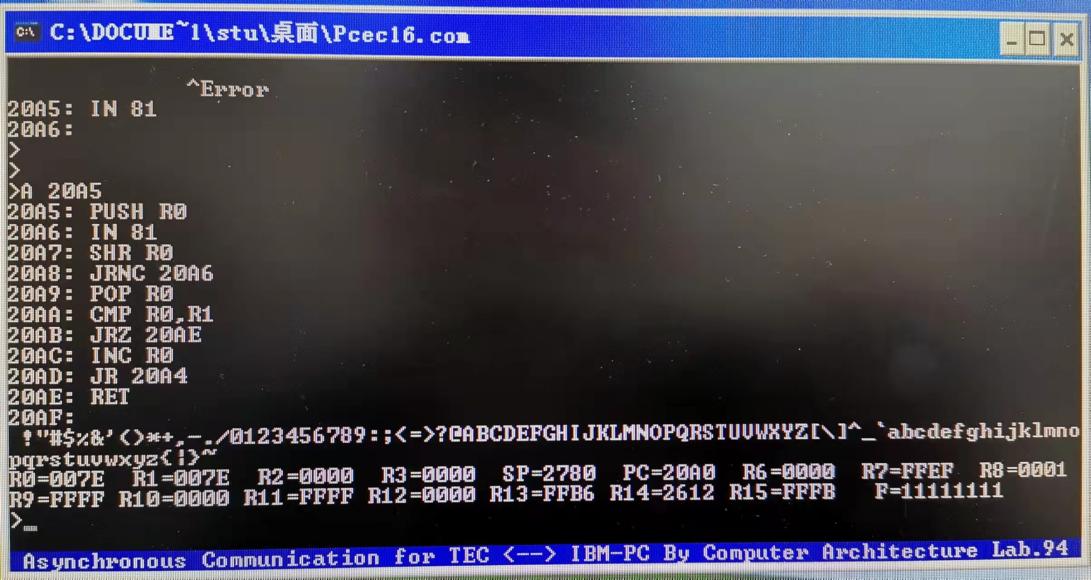
20AD: JR 20A4

20AE: RET

实验数据：无手动输入

运行结果：输出了95个可打印字符

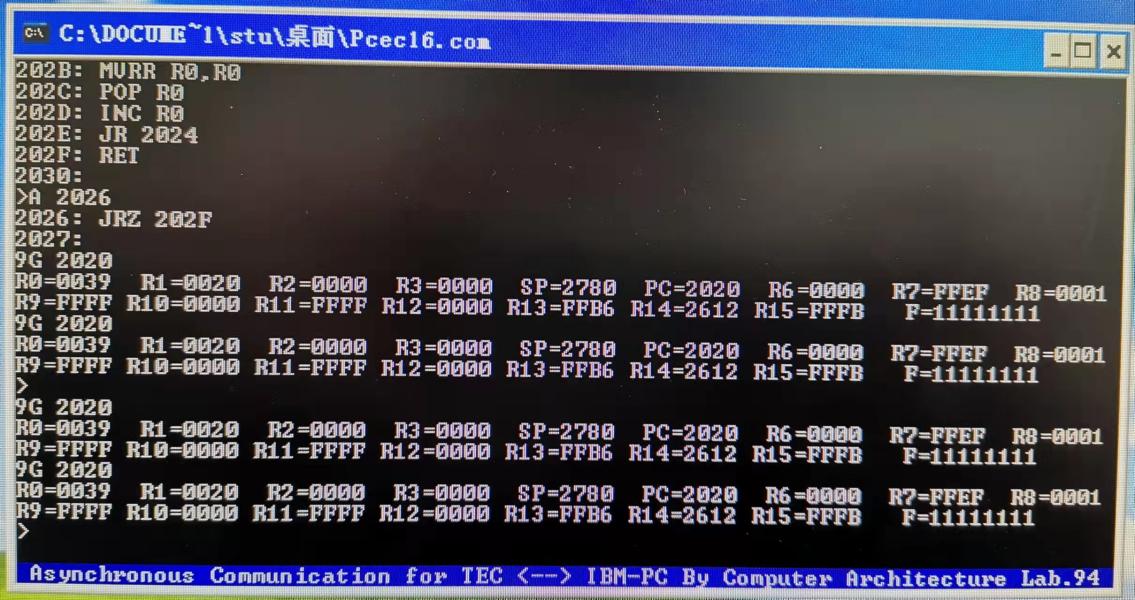
实验现场图：



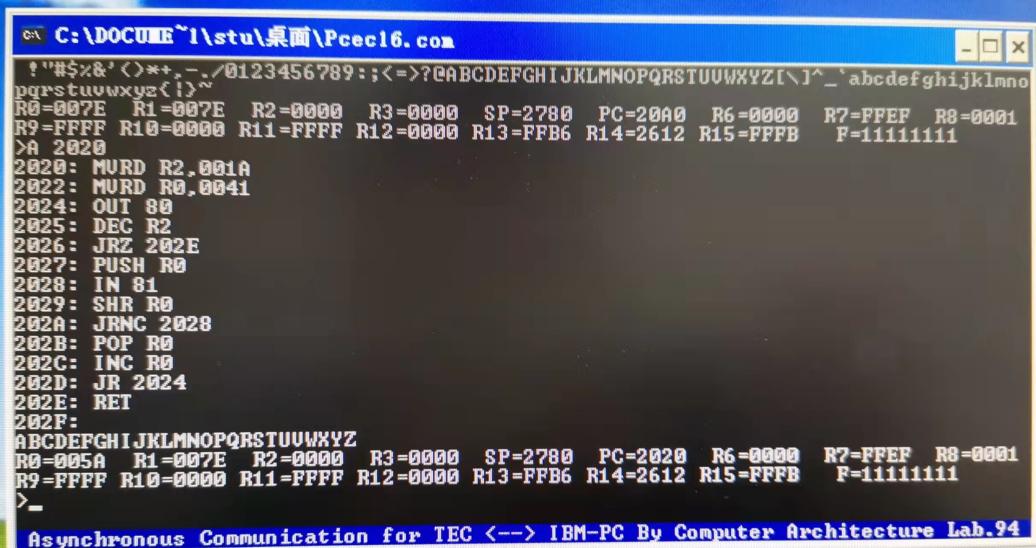
1. **作业与思考题**

1. 把例2中的IN 81, SHR R0, JRNC 2028三个语句换成4个MVRR R0，R0语句

该程序会无限循环地输出乱码，是因为程序控制权最后未交到RET语句，需要将2026处的代码改为“JRZ 202F”，使程序正常退出

****

2. 在终端屏幕上输出'A'到'Z'共26个英文字母



将2020处要输出的字符数改为26，即(001A)H；将输出的起始字符改为A的ASCII码65，即(0041)H。就可以实现输出A到Z共26个字符

个人体会与总结

本次实验使用清华大学设计制作的“迷你计算机”TEC进行了一些汇编语言层面的测试，理解了程序执行的基本原理和基础的汇编语言的设计思路。

在输入命令之前，都要先输入A address指令来指示接下来要输入的命令要被存在内存的什么位置，这充分体现了计算机的冯氏结构的特点——程序都存在内存中，一条条指令被存在主存（即内存）中。

在发出G address指令后，程序控制权交到了位于address处的命令，若没有跳转类的命令，控制权就从前往后按照顺序交给各条命令，当碰到跳转类的命令时（例如JRZ address）会将控制权交到对应的语句，而后继续一条一条地从前往后执行内存中的程序。

当程序控制权转移到RET时，表示整个程序的结束，将会停止执行程序。

从使用到的一些命令来看，一般情况下，运算器进行运算时所能使用的数据都只能来自于寄存器，对于要和内存中的数据交互的，要先把数据从内存中读取到寄存器中，再进行运算使用。可见此处在底层对算法有很大的优化空间，由于寄存器存取速度是最快的，而寄存器和其他部分进行数据交换显然比直接使用寄存器中的数据慢很多，因此在设计程序时，要尽量减少寄存器与外部的数据交换以提高程序的执行效率。