**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 设计一个小程序, 从键盘上接收一个字符并在屏幕上输出显示该字符。**

【1】在命令行提示符状态下输入：

A 2000↙；

屏幕将显示：

2000：

输入如下形式的程序：

2000: IN 81 ；判键盘上是否按了一个键

2001：SHR R0 ；即串行口是否有了输入的字符

2002：SHR R0

2003：JRNC 2000 ；未输入完则循环测试

2004：IN 80 ；接收该字符

2005: OUT 80↙；在屏幕上输出显示字符‘6’

2006: RET↙；每个用户程序都必须用RET指令结束

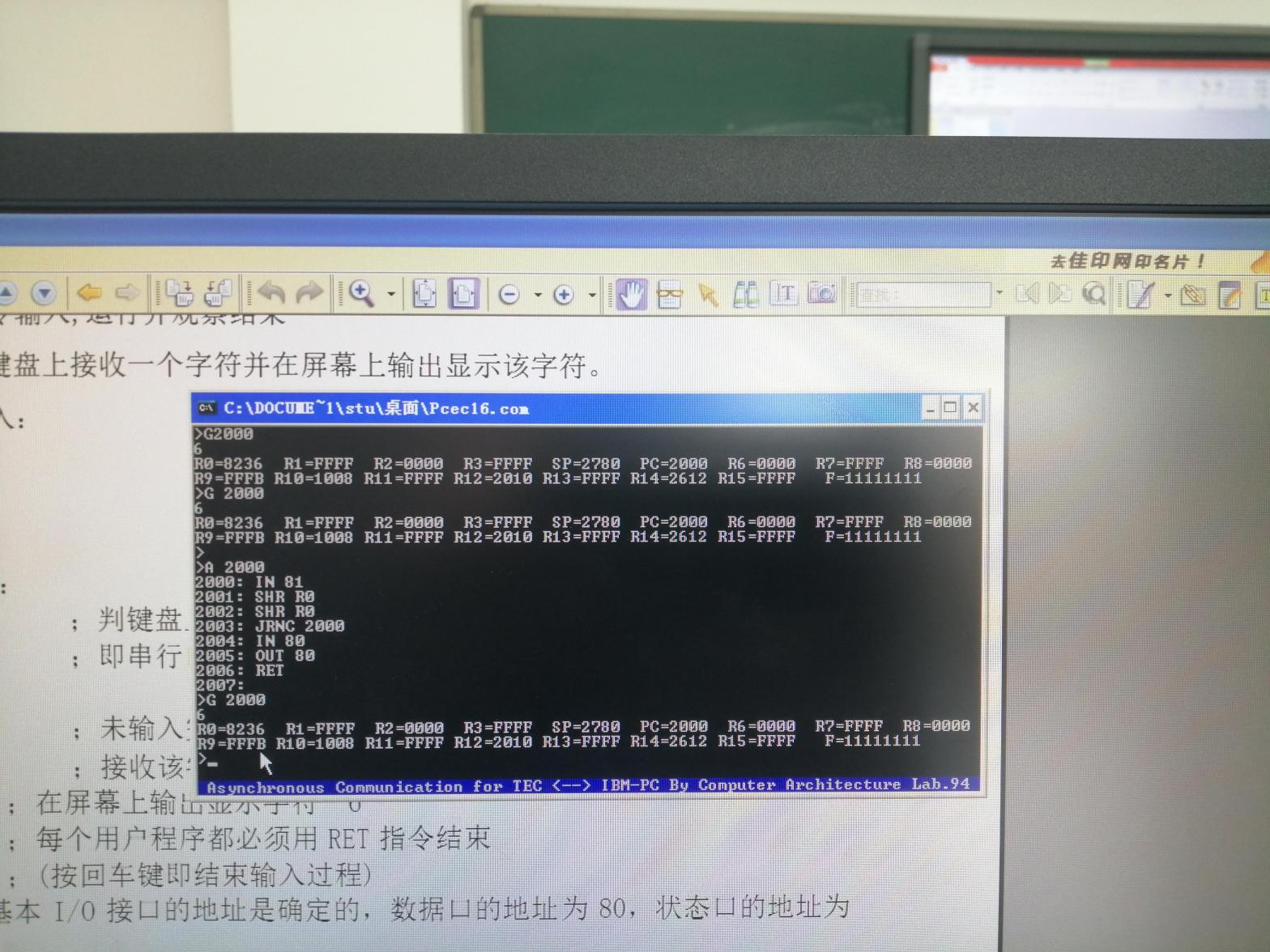
2007：↙；(按回车键即结束输入过程)

注：在十六位机中，基本I/O接口的地址是确定的，数据口的地址为80，状态口的地址

为81。

【2】用“G”命令运行程序

在命令行提示符状态下输入：G2000↙

****执行上面输入的程序，结果是什么？

答：用户输入端输入的是什么，就会显示什么。例如运行中我输入了6，那么就会在下一行显示6.

**例2 设计一个小程序, 用次数控制在终端屏幕上输出'0'到'9'十个数字符。**

【1】在命令行提示符状态下输入：A 2020↙

屏幕将显示：

2020：

从地址2020H开始输入下列程序:

2020:MVRD R2,00OA ；送入输出字符个数

2022:MVRD R0,0030 ；“0”字符的ASCII码送寄存器R0

2024:OUT 80 ；输出保存在R0低位字节的字符

2025:DEC R2 ；输出字符个数减1

2026:JRZ 202E ；判10个字符输出完否,已完，则转到程序结束处

2027:PUSH R0 ；未完，保存R0的值到堆栈中

2028:IN 81 ；查询接口状态，判字符串行输出完成否,

2029:SHR R0 ；

202A:JRNC 2028 ；未完成, 则循环等待

202B:POP R0 ；已完成, 准备输出下一字符并从堆栈恢复R0的值

202C:INC R0 ；得到下一个要输出的字符

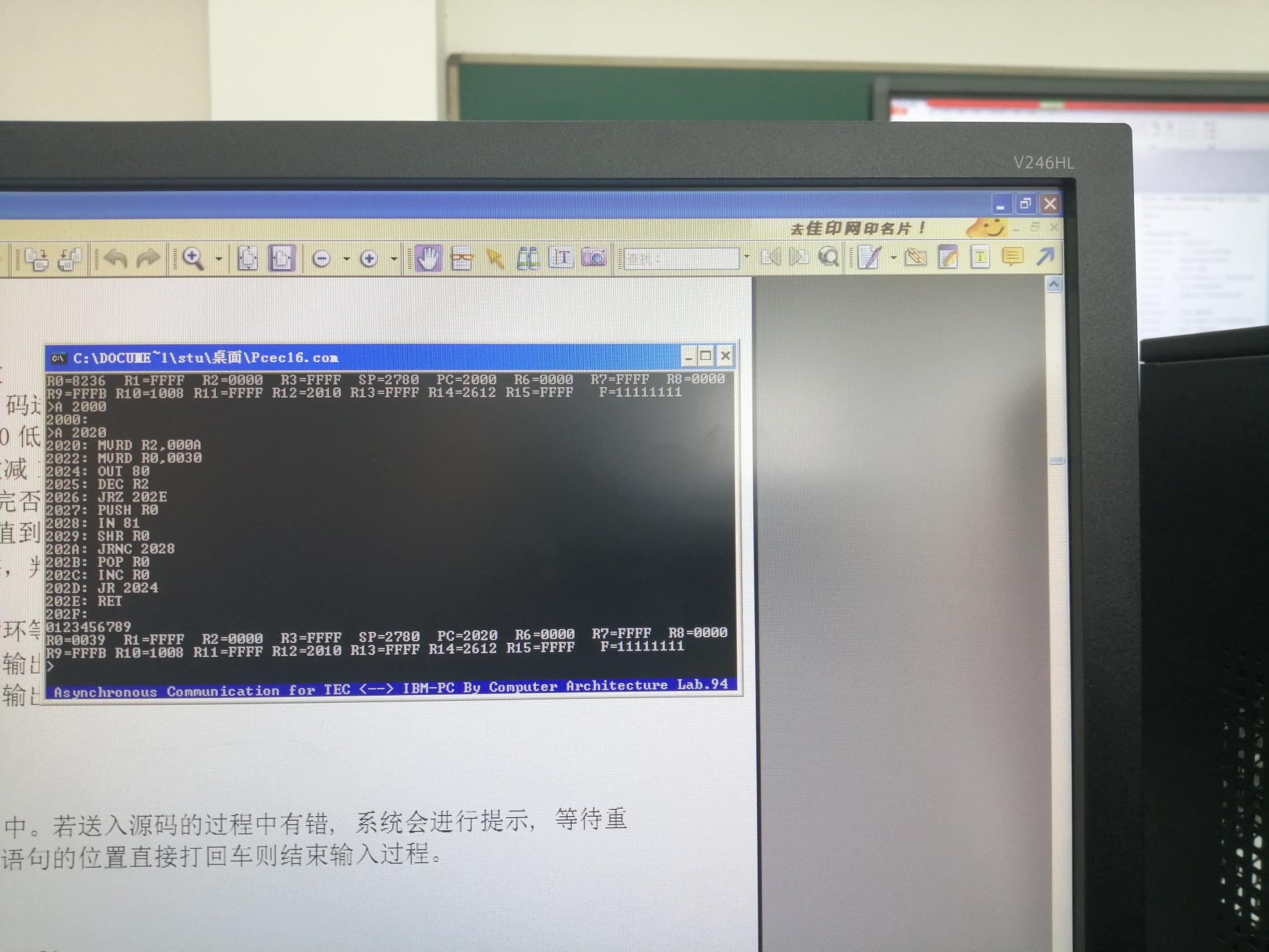
202D:JR 2024 ；转去输出字符

202E:RET

202F：↙

【2】用“G”命令运行程序

在命令行提示符状态下输入：G2020



执行结果为？

答：输出0123456789.

**例3 从键盘上连续打入多个属于‘0’到‘9’的数字符并在屏幕上显示, 遇到非数字**

**字符结束输入过程。**

【1】在命令行提示符状态下输入：A 2040↙

屏幕显示如下：2040：从地址2040H开始输入下列程序:

(2040) MVRD R2,0030 ；用于判数字符的下界值

MVRD R3,0039 ；用于判数字符的上界值

(2044) IN 81 ；判键盘上是否按了一个键,

SHR R0 ；即串行口是否有了输入的字符

SHR R0

JRNC 2044 ；没有输入则循环测试

IN80 ；输入字符到R0

MVRD R1,00FF

AND R0,R1 ；清零R0的高位字节内容

CMP R0,R2 ；判输入字符≥字符'0'否

JRNC 2053 ；为否, 则转到程序结束处

CMP R3,R0 ；判输入字符≤字符'9'否

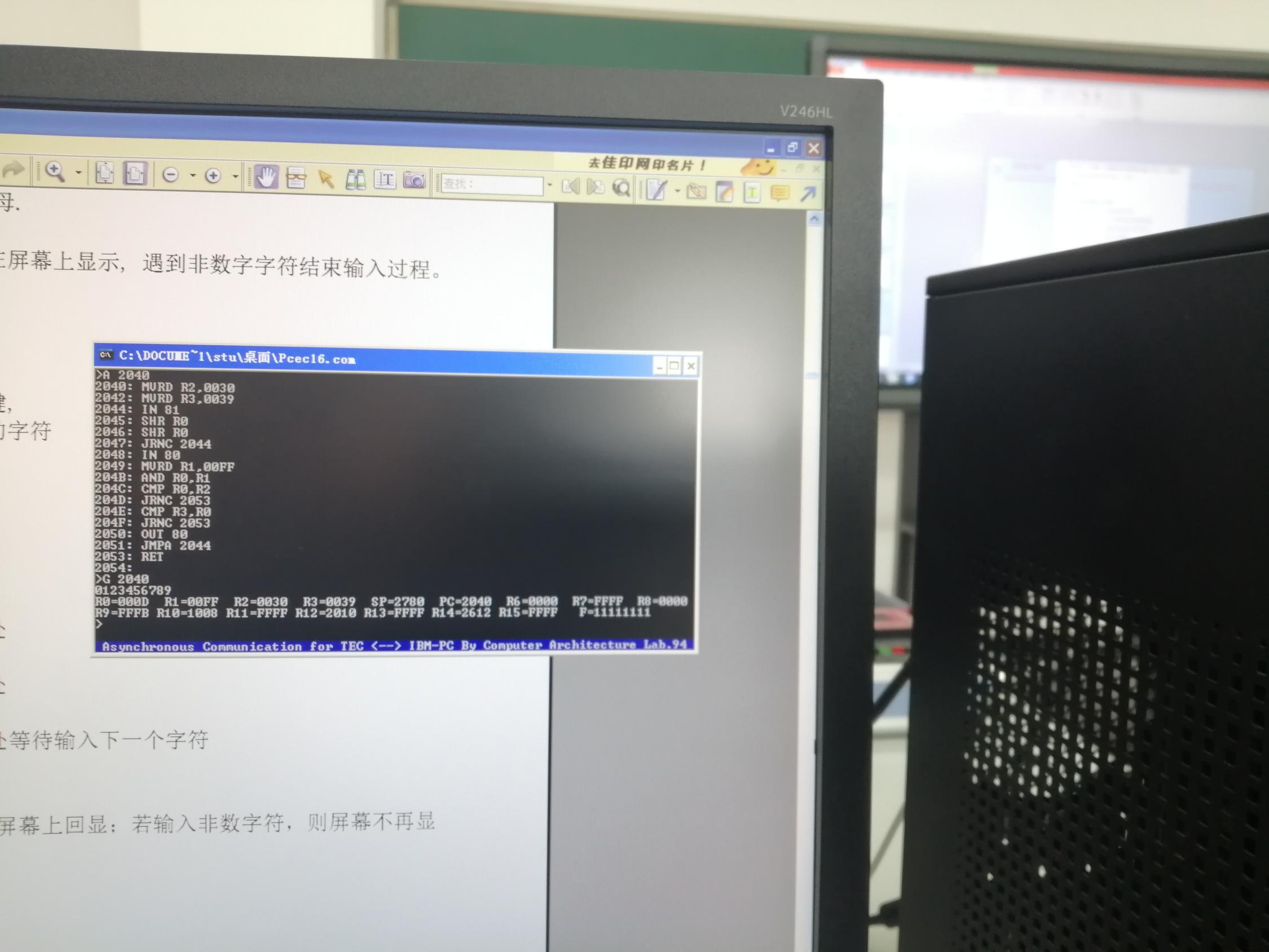
JRNC 2053 ；为否, 则转到程序结束处

OUT 80 ；输出刚输入的数字符

JMPA 2044 ；转去程序前边2044处等待输入下一个字符

(2053) RET

【2】在命令行提示符状态下输入：G 2040↙



执行结果为？

答：连续输入0123456789后回车，回车为非数字符，结束输入。

**例4 计算1到10的累加和。**

【1】在命令行提示符状态下输入：

A 2060↙屏幕将显示：

2060：从地址2060H开始输入下列程序：

(2060) MVRD R1,0000 ；置累加和的初值为0

MVRD R2,00OA ；最大的加数

MVRD R3,0000

(2066) INC R3 ；得到下一个参加累加的数

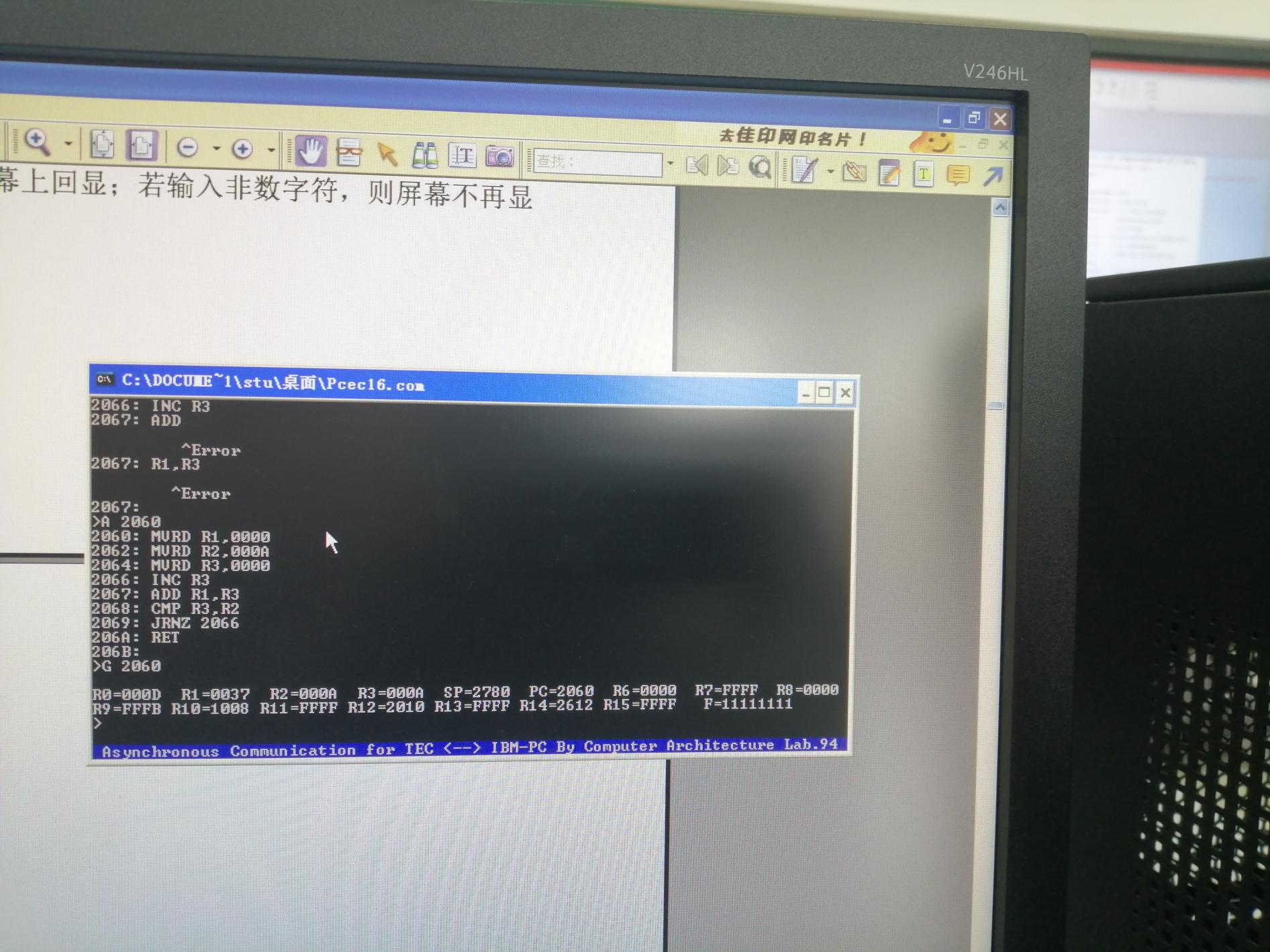
ADD R1,R3 ；累加计算

CMP R3,R2 ；判是否累加完

JRNZ 2066 ；未完, 开始下一轮累加

RET

【2】在命令行提示符状态下输入：G 2060↙

****

执行结果为？

答：R1=0037，R2=000A，R3=000A，计算出了每次累加的数值和最终1到10的累加。

**例5 设计一个有读写内存和子程序调用指令的程序,功能是读出内存中的字符，将其**

**显示到显示器的屏幕上，转换为小写字母后再写回存储器原存储区域。**

【1】将被显示的6个字符‘A’~‘F’送入到内存20F0H开始的存储区域中。

在命令行提示符状态下输入：

E 20F0↙屏幕将显示：

20F0 内存单元原值：

按下列格式输入：

20F0 内存原值：0041 内存原值：0042 内存原值：0043

内存原值：0044 内存原值：0045 内存原值：0046↙

在命令行提示符状态下输入：

从地址2080H开始输入下列程序：

(2080) MVRD R3,0006 ；指定被读数据的个数

MVRD R2,20F0 ；指定被读、写数据内存区首地址

(2084) LDRR R0,[R2] ；读内存中的一个字符到R0寄存器

CALA 2100 ；指定子程序地址为2100,调用子程序，完成显示、

转换并写回的功能

DEC R3 ；检查输出的字符个数

JRZ 208B ；完成输出则结束程序的执行过程

INC R2 ；未完成，修改内存地址

JR 2084 ；转移到程序的2086处，循环执行规定的处理

(208B) RET

从地址2100H开始输入下列程序：

(2100) OUT 80 ；输出保存在R0寄存器中的字符

MVRD R1，0020

ADD R0，R1 ；将保存在R0中的大写字母转换为小写字母9

STRR [R2],R0 ；写R0中的字符到内存，地址同LOD所用的地址

(2105) IN 81 ；测试串行接口是否完成输出过程

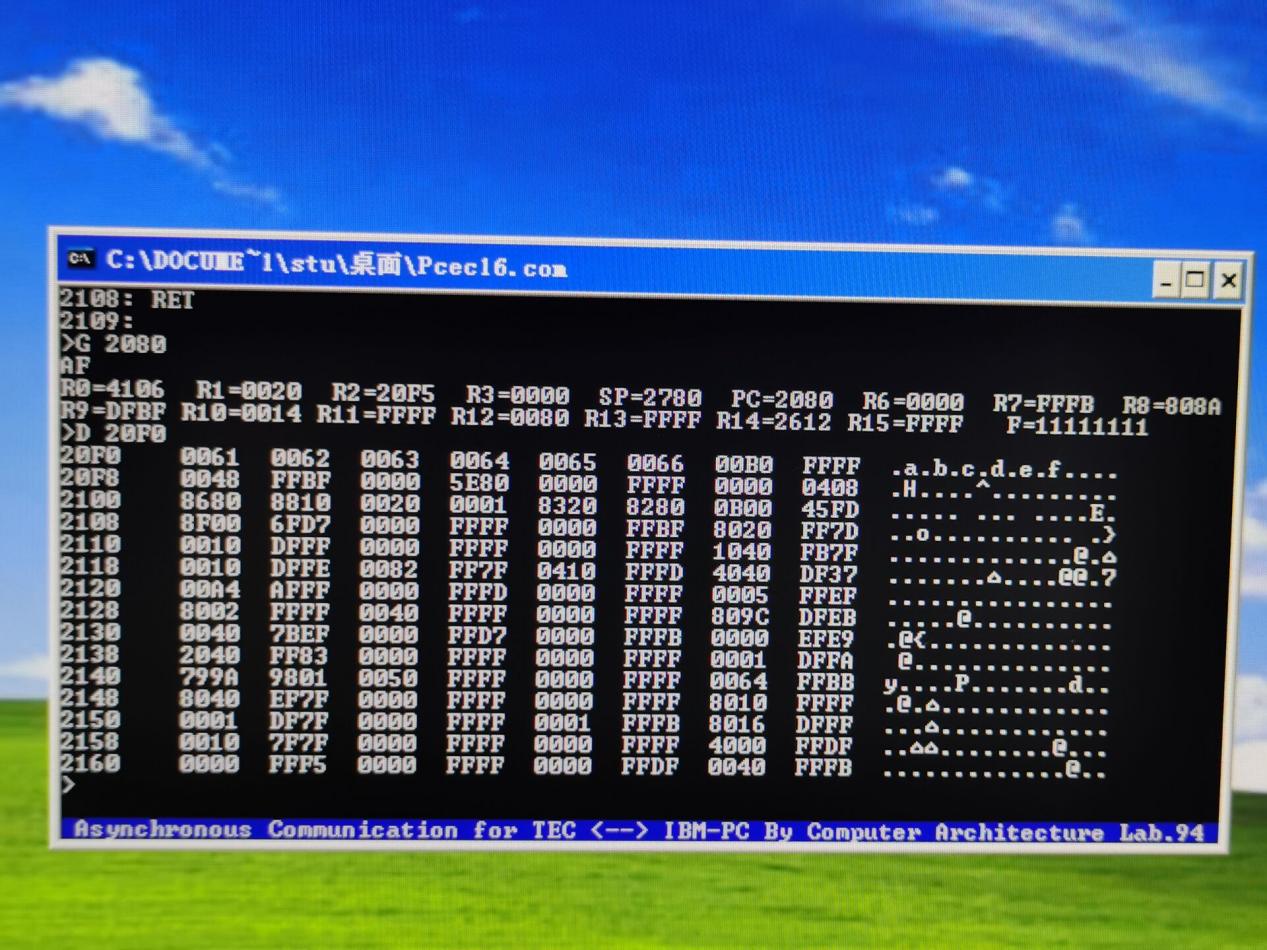
SHR R0

JRNC 2105 ；未完成输出过程则循环测试

RET ；结束子程序执行过程，返回主程序

【2】在命令行提示符状态下输入：G 2080↙

屏幕显示运行结果为？



运行结果为：AF

【3】在命令行提示符状态下输入：D 20F0↙

20F0H~20F5H内存单元的内容为？

答：20F0H~20F5H内存单元的内容为：0061、0062、0063、0064、0065、0066（a、b、c、d、e、f）

**例6 设计一个程序在显示器屏幕上循环显示95个(包括空格字符)可打印字符。**

【1】在命令行提示符状态下输入：A 20A0↙

屏幕将显示：20A0：

从地址20A0H开始输入下列程序：

A 20A0 ；从内存的20A0地址单元开始建立用户的第一个程序

20A0: MVRD R1,7E ；向寄存器传送立即数

20A2: MVRD R0,20 ；

20A4: OUT 80 ；通过串行接口输出R0低位字节内容到显示器屏幕

20A5: PUSH R0 ；保存R0寄存器的内容到堆栈中

20A6: IN 81 ；读串行接口的状态寄存器的内容

20A7: SHR R0 ；R0寄存器的内容右移一位，最低位的值移入标志位C

20A8: JRNC 20A6 ；条件转移指令，当标志位C不是1时就转到20A6地址

20A9: POP R0 ；从堆栈中恢复R0寄存器的原内容

20AA: CMP R0,R1 ；比较两个寄存器的内容是否相同，相同则标志位Z=1

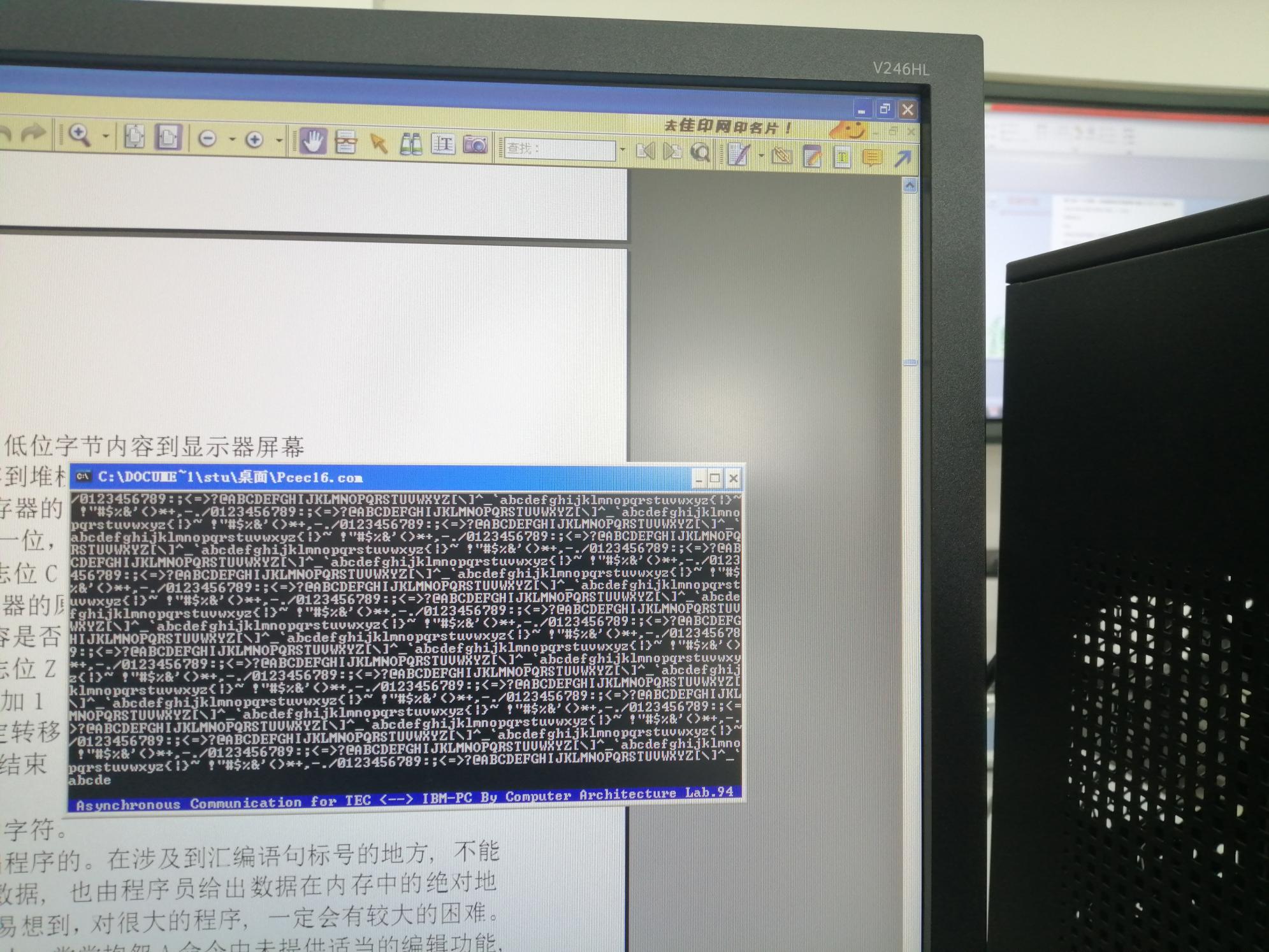
20AB: JRZ 20AE ；条件转移指令，当标志位Z为1时转到20AE地址,结束程序

20AC: INC R0 ；把R0寄存器的内容增加1

20AD: JR 20A4 ；无条件转移指令，一定转移到20A4地址

20AE: RET ；子程序返回指令,程序结束

【2】在命令行提示符状态下输入：G 20A0↙



运行过后, 可以观察到显示器上会显示出所有可打印的字符。

1. 执行例1输入的程序，结果是什么？

答：用户输入端输入的是什么，就会显示什么。例如运行中我输入了6，那么就会在下一行显示6.

1. 执行例2输入的程序，结果是什么？

答：输出0123456789.

1. 执行例3输入的程序，结果是什么？

答：连续输入0123456789后回车，回车为非数字符，结束输入。

1. 执行例4输入的程序，结果是什么？

答：R1=0037，R2=000A，R3=000A，计算出了每次累加的数值和最终1到10的累加。

5. 执行例5输入的程序，屏幕显示运行结果是什么？在命令行提示符状态下输入：D

20F0↙ 20F0H~20F5H内存单元的内容分别是什么？

答：屏幕运行结果为AF，20F0H~20F5H内存单元的内容为：0061、0062、0063、0064、0065、0066（a、b、c、d、e、f）

6. 执行例6输入的程序，结果是什么？

答：可以观察到显示器上会循环显示出所有可打印的字符。

1. **作业与思考题**

（1）若把例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 执行过程会出现什么现象? 试分析并实际执行一次。

提示：该程序改变这三条语句后，若用T命令单条执行，会依次显示0~9十个数字。若

用G命令运行程序，程序执行速度快，端口输出速度慢，这样就会跳跃输出。

在命令行提示符状态下输G 2020，屏幕显示09。

**答：由于程序运行速度快，端口串行输出的速度慢，这样就会丢失某些输出的字行。如：在这部机器上，在命令行提示符状态下输入G 2020命令后，屏幕显示的是9，而不是0123456789的十个字符。三句变成四句之后，其对应的跳转语句地址也要变，由JRZ 202E变成JRZ 202F。**

****

（2）若要求在终端屏幕上输出'A'到'Z'共26个英文字母，应如何修改例1中给出的

程序? 请验证之。

A 2100

MVRD R2,001A

MVRD R0,0041

OUT 80

DEC R2

JRZ 210E

PUSH R0

IN 81

SHR R0

JRNC 2108

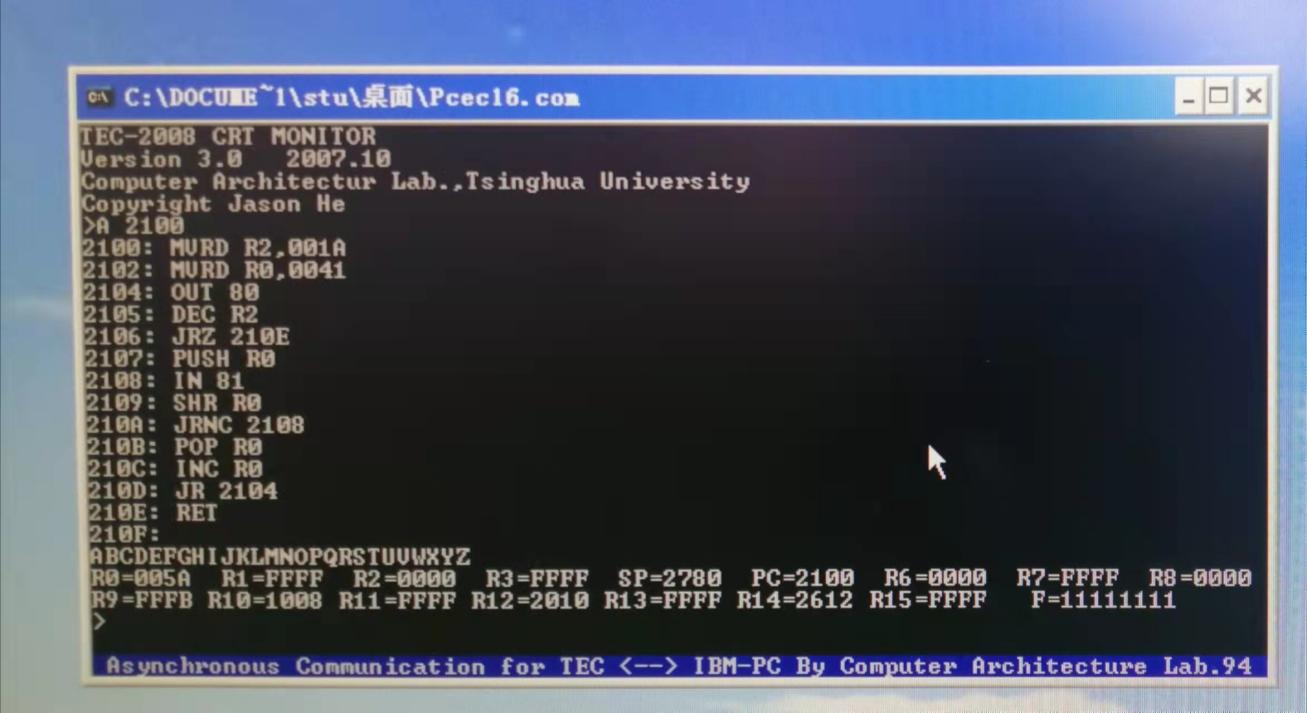
POP R0

INC R0

JR 2104

RET

验证：



结果：输出ABCDEFGHIJKLMNOPQRSTUVWXYZ

1. **个人体会与总结**

答：在完成思考题一时，在完成把IN 81, SHR R0, JRNC 2028三个语句换成4个MVRR R0，R0语句后，出现了一直循环乱码的现象。经过检查发现是由于在把三句代码

转换为四句时，其对应后面的跳转语句地址也要随之改变，由JRZ 202E变为JRZ

202F。改正之后，出现了正确结果。

在完成思考题二时，我们先由汇编语言的思想编写好程序，而后按照附录里的语句功能表再进行修正，发现与已学的汇编语言还是有一定的差距。比如跳转语句，

此处我们运用的是利用进位符号位CF，与汇编语言中的习惯相反，有进位借位时

为0，反之为1.开始由于写反了而总是得不到正确结果，经过改正，最终达到实验

要求。