



**实验**报告

**课程名称 计算机组成原理**

**题目名称 计算机组成原理**

**学生学院 计算机学院**

**专业班级 19级软件工程（1）班**

**指导教师 叶林锋**

**学 号 3119005028**

**学生姓名 魏耀辉**

2020年12月04日



计算机 学院 软件工程 专业 （1） 班\_\_\_\_\_组、学号3119005028

姓名 魏耀辉 协作者 陆锦源 教师评定\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

实验题目 实验一：基础汇编语言程序设计

一、实验目的：

1、学习和了解TEC-XP+教学实验系统监控命令的用法；

2、学习和了解丁EC-XP+教学实验系统的指令系统；

3、学习简单的TEC-XP+教学实验系统汇编程序设计。

二、实验设备与器材：

1.一台串口工作良好的PC机；

2.TEC-XP一台。

三、**实验**说明和原理**：**

实验说明：使用教学机前，应先熟悉教学机的各个组成部分，及其使用方法。

实验原理：利用TEC-XP+系统进行汇编语言的程序设计。

四、实验内容：

1.学习联机使用 TEC-XP+教学实验系统和仿真终端软件PCEC；

2.学习使用 WINDOWS界面的串口通讯软件；

3.使用监控程序的R命令显示/修改寄存器内容、D命令显示存储器内容、E命令修改存储器内容；

4.使用A命令写一小段汇编程序，U命令反汇编刚输入的程序，用G命令连续运行该程序，用T、P命令单步运行并观察程序单步执行情况。

**五、**实验步骤：

（1）准备一台串口工作良好的PC机；

（2）将TEC-XP放在实验台上，打开实验箱的盖子，确定电源处于断开状态；

（3）将黑色的电源线一端接220V交流电源，另一端插在TEC-XP实验箱的电源插座里；

（4）取出通讯线，将通讯线的9芯插头接在TEC-XP实验箱上的串口“COM”或“C0M2”上，另一端接到PC机的串口上；

（5）将 TEC-XP实验系统左下方的六个黑色的控制机器运行状态的开关置于正确的位置，在找个实验中开关应置为001100（连续、内存读指令、组合逻辑、联机、16位、MACH）控制开关的功能在开关上、下方有标识：开关拨向上方表示“1”，拨向下方表示“0”，“X表示任意，其它实验相同；

（6）打开电源，船形开关和5V电源指示灯亮；

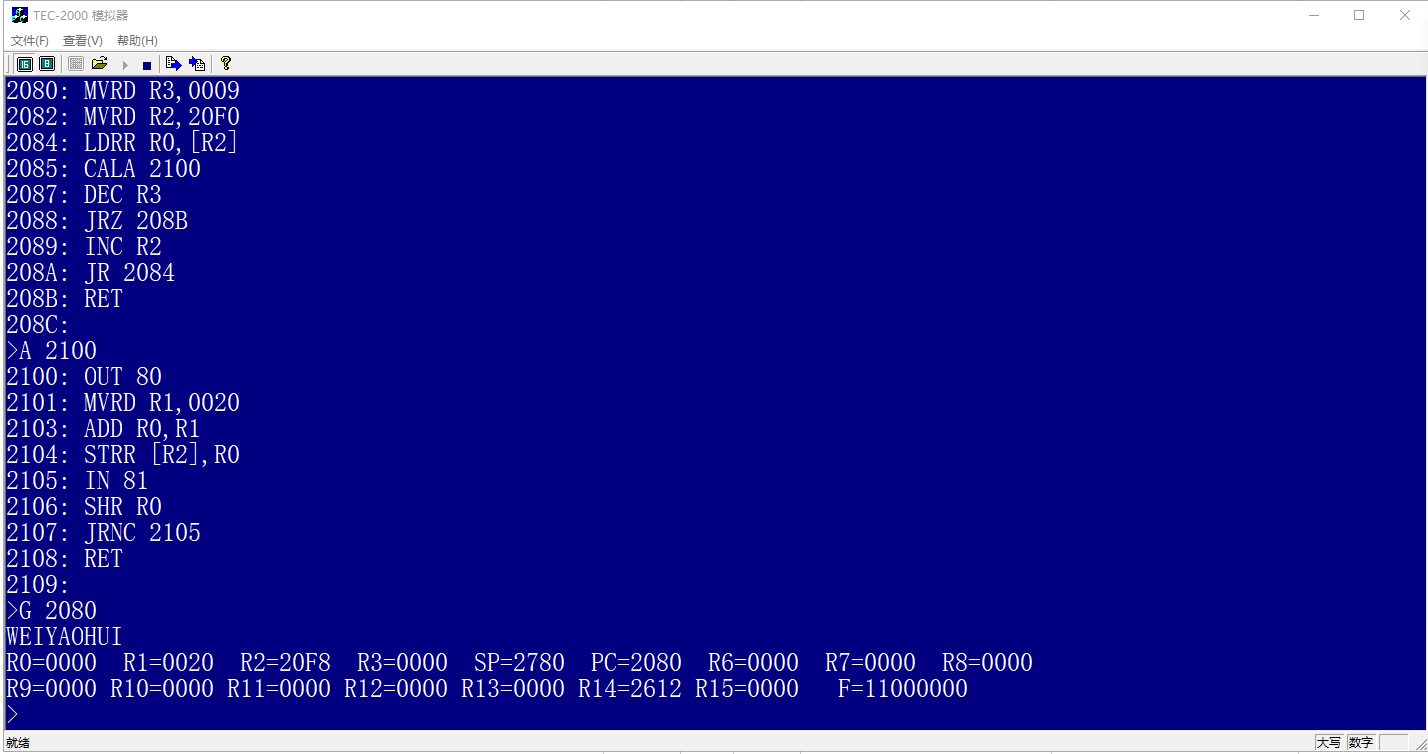
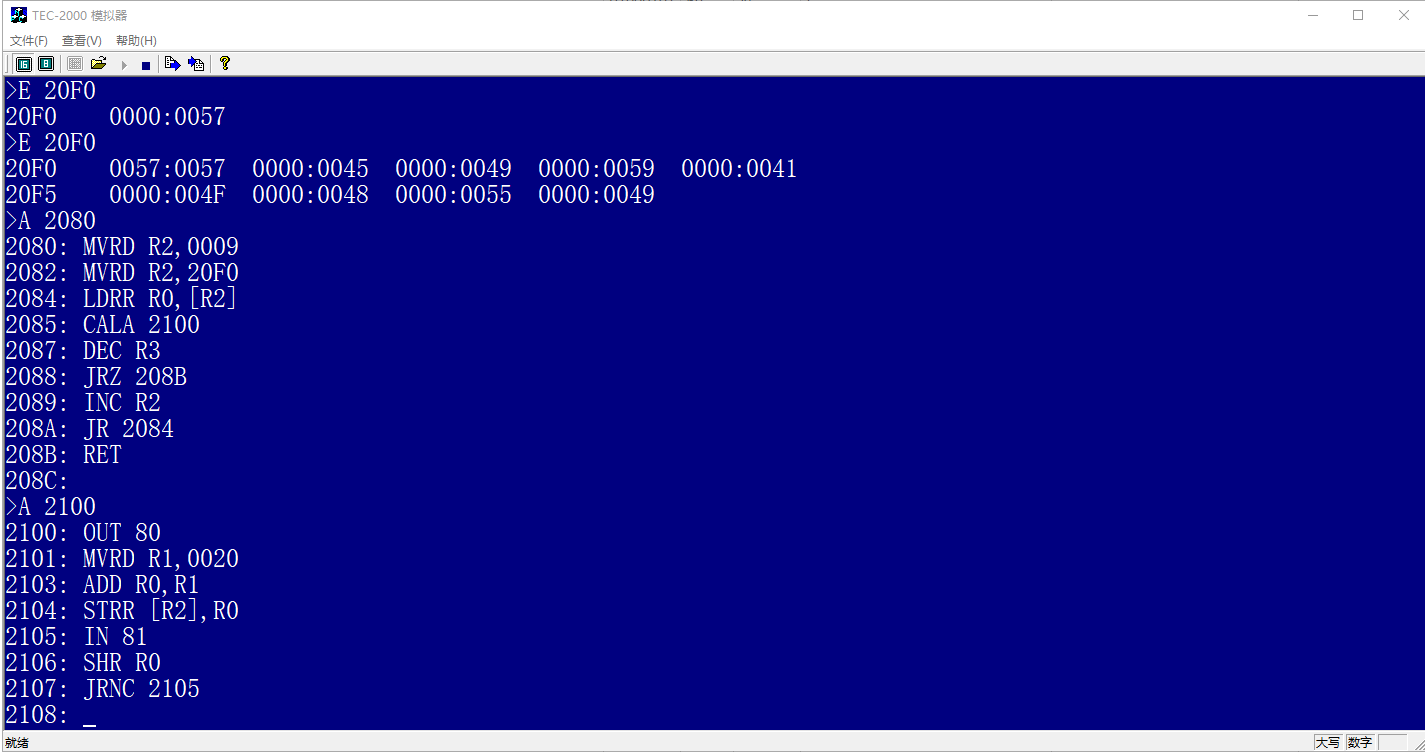
（7）在PC机上运行PCEC16.EXE文件，根据连接的PC机的串口设置所用PC机的串口为“1”或“2”，直接回车即可；

（8）按一下“ RESET”按键，再按一下“ START”按键；

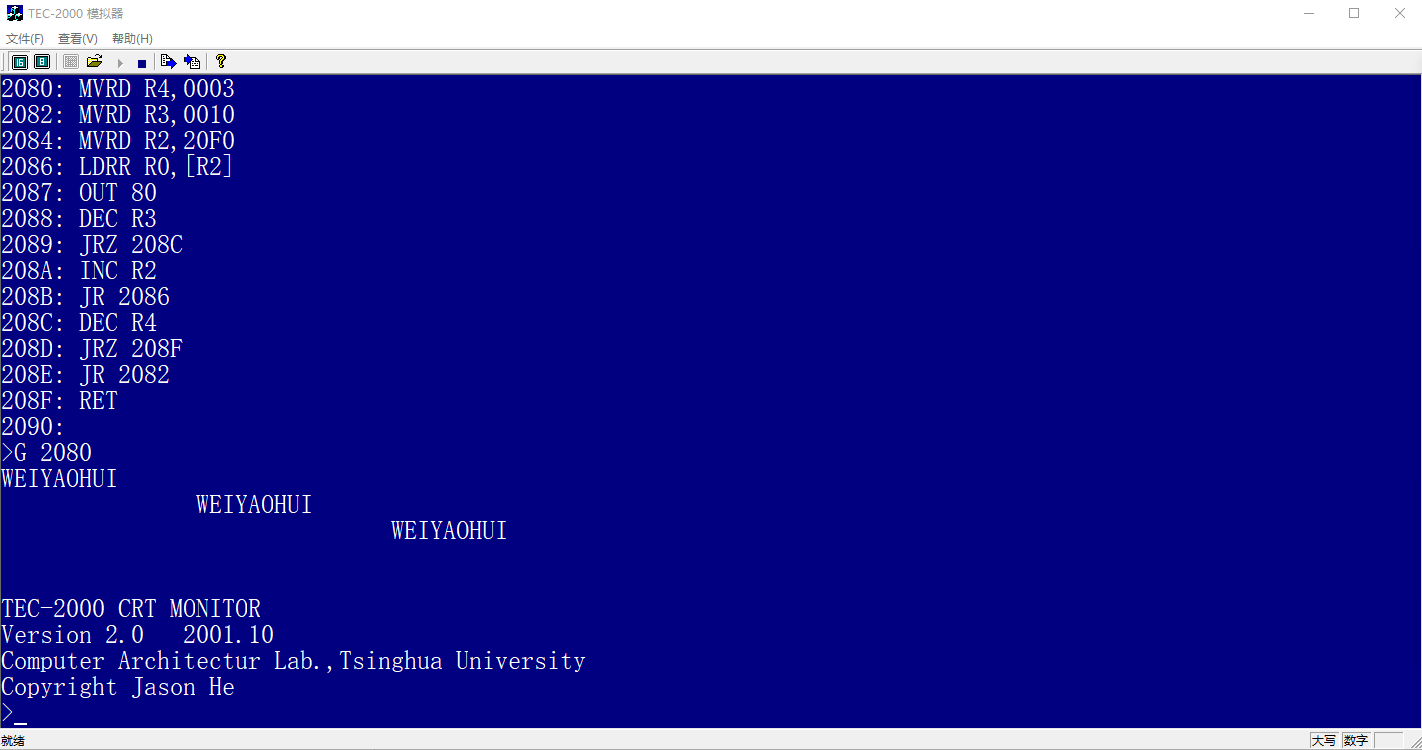
（9）进行汇编语言的程序设计。

六、思考题：

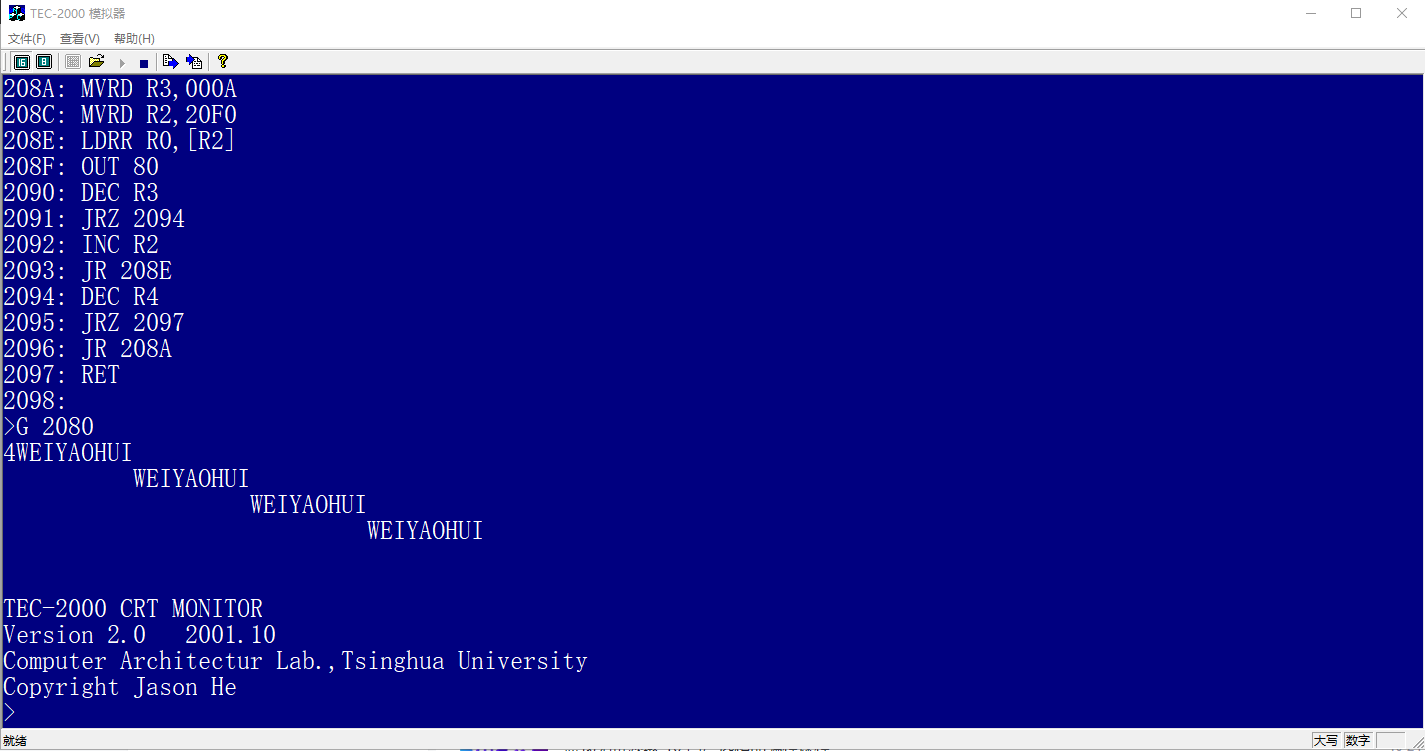
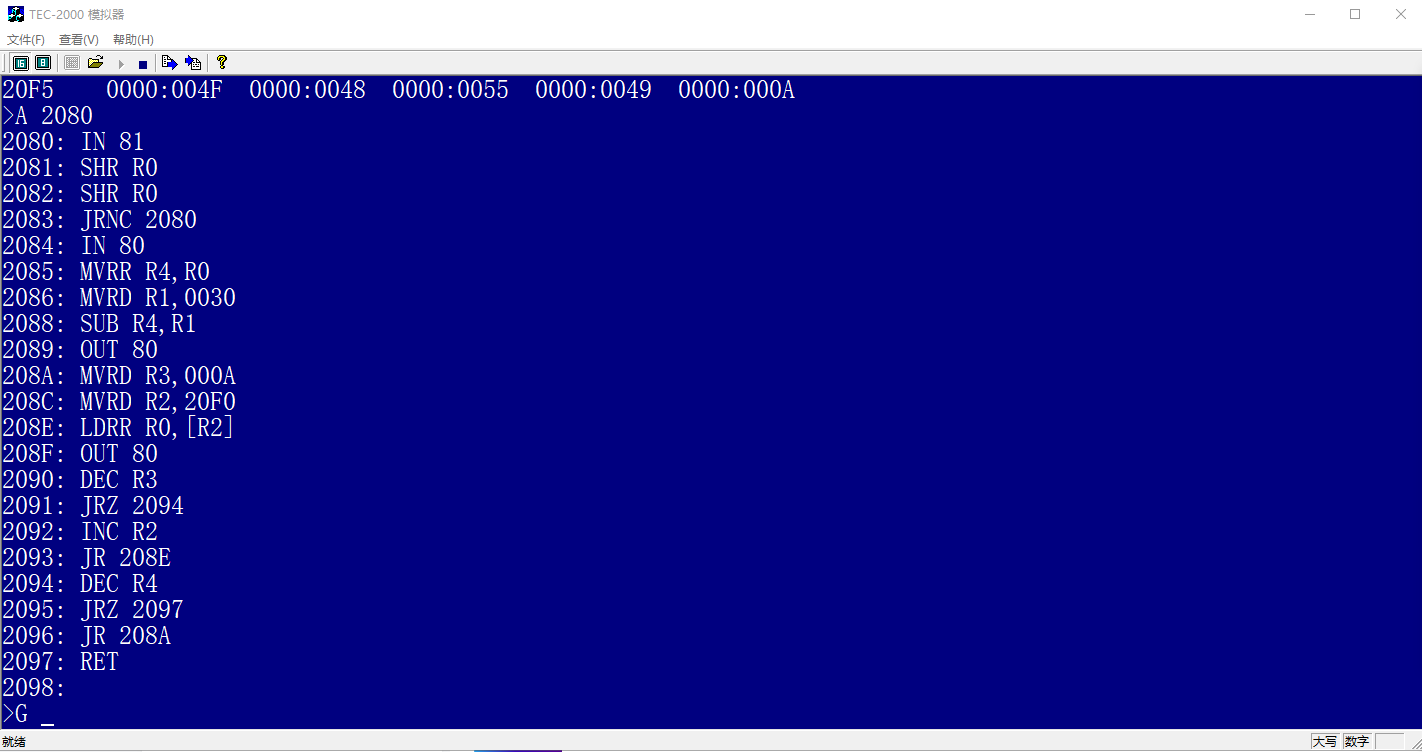
思考题一：运行程序，输出自己的名字；



思考题二：运行程序，输出自己的名字三次；



思考题三：运行程序，输入次数，输出自己的名字对应的次数。



七、实验心得：

本次实验中，由于TEC-XP教学机存在故障问题无法使用，只能使用TEC-2000模拟器进行模拟操作。因为初次接触汇编语言程序设计，所以对它的语法和程序结构并不了解，前期未能迅速的解决思考题。后来经过上网查找资料以及比对实验指导书了解了汇编的基本语法。对于实验希望改进下程序设计的速度。



计算机 学院 软件工程 专业 （1） 班\_\_\_\_\_组、学号3119005028

姓名 魏耀辉 协作者 陆锦源 教师评定\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

实验题目 实验二：脱机运算器实验

一、实验目的：

1、深入了解AM2901算器的功能与具体用法；

2、深化运算器部件的组成、设计、控制与使用等知识实验内容。

二、实验设备与器材：

1.TEC-XP一台。

三、**实验**说明和原理**：**

实验说明：脱机运算器实验，是指让运算器从教学计算机整机中脱离出来，此时，它的全部控制与操作均需通两个12位的微型开关来完成，这就不上执行指令，只能通过开关按键控制教学机的运算器完成定的运算功能，并通过指示灯观察运算结果。

1.12位微型开关的具体控制功能分配如下：

A口、B口地址：送给AM2901器件用于选择源与目的操作数的寄存器编号。

I8-I0：选择操作数来源、运算操作功能、选择操作数处理结果和运算器输出内容的3组3位的控制码

SCi、SSH和SST：用于确定运算器最低位的进位输入、移位信号的入/出和怎样处理AM2901产生的状态标志位的结果。

2.开关位置说明：

做脱机运算器实验时，要用到提供24位控制信号的微动开关和提供16位数据的拨动开关。微动开关是红色的，一共有三个一个动开关可以提供12位的制信号，三个开关分别标有 SWI micro switch sW2 micro switch和sw3 micro switch

1. 开关检测：

红色微动开关是该系使用命最的器件开关的测方法比较简单，用户将六个控制工作方式的开关置1XX000前第三个和第三的开关处于任意位置。然后将两个微动的24个小纽子依置为1（开关拨到上方为1。看対应的指示灯是否亮。如果有一个或数个指示灯不亮，一般是开关出了问题）

实验原理：利用TEC-XP+系统进行脱机操作。

四、实验内容：

1.在脱机方式下，对于给定指令分析其执行过程中运算的步骤，通过对AM2901运算器所需控制号的设置，使之完成运算，并核对运算结果实验要求。

**五、**实验步骤：

在脱机方式下，通过设置SM、SW2、SW3各微码字段和数据开关，可实现多种运算。以实现 AAAAH→R为例说明操作过程。

1. 按照表中的微码和数据开关，对运算器的功能进行设置。操作步骤如下：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 微动开关 | | | | | | | | 数据开关 |
| I8-I6 | I5-I3 | I2-I0 | SST | SSH | SCI | B | A | D15-D0 |
| 011 | 000 | 111 | 001 | 0 | 00 | 0001 | 不用 | AAAH |

1. 将教学机左下方的6个拨动开关置为1XX00（单步、16位、脱机、MACH）：先按一下 RESET按键，再按一下START按键，进行初始化。
2. 通过16个数据开关设置立即数 AAAAH

（3）通过SW、SW2、SW3设置各微码。

2、按一次 START键，立即数XXXH置入R，通过显示灯察看按 START键后的输出。若要进行其他操作：

（1）重新设置SW、SW2、SW3、数据开关，通过显示灯观察输出。

（2）然后按 START键执行操作，通过显示灯观察按下 START键后的输出，检查运算结果是否正确。

3、完成表中的各种运算，记录按 START键前和按 START键后的ALU输出及标志位C、Z、V、S的值。

问题：分析比较各指令按 START键前和按 START键后的值，是否有不同并解释。

六、思考题：

思考题一：完成下表

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 运算 | I8-I6 | I5-I3 | I2-I0 | SST | SSH | SCI | B | A | 压START前 | | 压START后 | |
| ALU  输出 | CZVS | ALU输出 | CZVS |
| 0101→R0 | 011 | 000 | 111 | 001 | 0 | 00 | 0000 | / | 0101 | / | 0101 | 0000 |
| 1010→R1 | 011 | 000 | 111 | 001 | 0 | 00 | 0001 | 0000 | 1010 | 0000 | 1010 | 0000 |
| R0+R1→R0 | 011 | 000 | 001 | 001 | 0 | 00 | 0000 | 0001 | 1111 | 0000 | 2121 | 0000 |
| R0-R1→R0 | 011 | 001 | 001 | 001 | 0 | 01 | 0000 | 0001 | 0101 | 0000 | F0F1 | 0000 |
| R1-R0→R1 | 011 | 001 | 001 | 001 | 0 | 01 | 0001 | 0000 | 0F0F | 1000 | 0E0E | 1000 |
| R0∨R1→R0 | 011 | 011 | 001 | 001 | 0 | 00 | 0000 | 0001 | 0F0F | 1000 | 0F0F | 1010 |
| R0∧R1→R0 | 011 | 100 | 001 | 001 | 0 | 00 | 0000 | 0001 | 0F0F | 1010 | 0F0F | 1010 |
| R0⊕R1→R0 | 011 | 110 | 001 | 001 | 0 | 00 | 0000 | 0001 | 0000 | 1010 | 0F0F | 1010 |
| ┐（R0⊕R1）→R0 | 011 | 111 | 001 | 001 | 0 | 00 | 0000 | 0001 | F0F0 | 1010 | 0000 | 1001 |
| 2\*R0→R0 | 111 | 000 | 011 | 001 | 0 | 00 | 0000 | / | F0F0 | 1001 | E1E0 | 0001 |
| R0/2→R0 | 101 | 000 | 011 | 001 | 0 | 00 | 0000 | / | E1E0 | 0001 | 70F0 | 0001 |

问题解答：压START前和压START后有区别，前应该是一种预览状态，而压之后是对同操作下一次压START前的预览情况。

七、实验心得：

本次实验中，因为指令微码的输入必须是连续的才能观察到对应的操作结果，若途中发生错误则必须重新开始，所以在做实验时必须全神贯注。通过这次的实验，加深了对AM2901运算器的了解，清楚了AM2901的功能与具体用法，深化运算器部件的组成、设计、控制与使用。



计算机 学院 软件工程 专业 （1） 班\_\_\_\_\_组、学号3119005028

姓名 魏耀辉 协作者 陆锦源 教师评定\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

实验题目 实验三：存储器部件教学实验

一、实验目的：

1、熟悉ROM芯片和RAM芯片在功能和使用方法等方面的相同和差异之处：学习用编程器设备向 EEPROM芯片内写入一批数据的过程和方法；

2、理解并熟悉通过字、位扩展技术实现扩展存储器系统容量的方案；

3、了解静态存储器系统使用的各种控制信号之间正常的时序关系；

4、了解如何通过读、写存储器的指令实现对58C65ROM芯片的读、写操作；

5、加深理解存储器部件在计算机整机系统中的作用。

二、实验设备与器材：

1.一台串口工作良好的PC机；

2.TEC-XP一台。

三、**实验**说明和原理**：**

实验说明：

（1）内存储器原理

内存储器是计算机中存放正在运行中的程序和相关数据的部件。在教学计算机存储器部件设计中，出于简化和容易实现的目的，选用静态存储器芯片实现内存储器的存储体，包括唯读存储区（ROM，存放监控程序等）和随读写存储区（RAM）两部分，ROM存储区选用4片长度8位、容量8KB的58C65芯片实现，RAM存储区选用2片长度8位、容量2KB的6116芯片实现。

（2）扩展教学机的存储器空间

教学机的主板上，预留了两片存储器芯片的28个引脚的器件插座，可以插上28个引脚的58C65ROM（8KB容量）器件，也可以插上24个引脚的6161RAM（2KB容量）器件。每片器件都只有8位数据线，为构建16位的字长的存储体，需要使用2片存储器器件，这就是通常说的存储器的位数（字长）扩展技术。为了保证新插上去的芯片与原有存储器芯片有正确的连接关系，需要为这2片存储器芯片分配地址空间范围，这就是通常说的存储器的容量（字数）扩展技术，通过把对内存高位地址译码产生的不同信号连接到这两个芯片的片选信号引脚完成。除此之外，还要向这2片存储器芯片提供正确的读写控制信号、使能控制信号（/OE）。再考虑到58C65ROM器件和6161RAM器件的引脚数不同，使用的控制信号也不完全相同，可能还有几个跳接线需要连通或者断开。对于用2片58C65ROM器件扩展8K字的ROM存储区的实验，可以在使用编程器设备向该存储器芯片内写入程序和数据之后，再把芯片插到器件插座中在唯读的操作方式运行其中的程序或使用其中的数据，检查结果的正确性。通过这个实验可以学习编程器的使用方法和向器件内写入信息的操作步骤。也可以直接用教学机的指令向58C65ROM器件内写入信息，例如用教学机的监控命令E向器件内输入某些信息，此时需要保证向器件提供正确的读写控制信号和使能控制信号，并确保每一次写操作要维持约1毫秒左右的时间，此方式下的58C65ROM器件提供了类似于RAM器件的功能，这只能算是一种变通的用法。用2片6161RAM器件扩展2K字的RAM存储区的实验。此时需要把24脚的RAM芯片按照与器件插座右对齐的方式查到插座中，并适当的进行必要的跳接线连通或者断开操作。全部接线都连接正确之后，就可以对这个存储区执行写入或者读出数据的操作功能。

实验原理：利用TEC-XP+系统和内置的存储器模块进行实验。

四、实验内容：

1.要完成存储器容量拓展的教学实验，需为拓展存储器选择一个地址，并注意读写和OE等控制信号的正确状态；

2.用监控程序的D、E命令对存储器进行读写，比较（6116）、 EEPROM（28系列芯片）、 EPROM（27系列芯片）在读写上的异同；

3.用监控程序的A命令编写一段程序，对RAM（611）进行读写，用D命令查看结果是否正确；

4.用监控程序的A命令编写一段程序，对扩展存储器 EEPROM（28系列芯片）进行读写，用D命令查看结果是否正确：如不正确，分析原因，改写程序，重新运行。

**五、**实验步骤：

1.接通教学机电源；

2.将教学机左下方的6个拨动开关置为001100连续从内存读指令、组合、联机、16位、MACH）；

3.按下 “RESET”按键，接着按“START”；

4.RAM（6116）支持即时读写，可直接用A、E命令向扩展的存储器输入程序或改变内存单元的值。RAM中的内容在断电后会消失，重新启动实验机后会发现内存单元的值发生了改变；

（1）用E命令改变内存单元的值并用D命令观察结果

1）在命令行提示符状态下输入：

E 2020

屏幕将显示：2020内存单元原值：

按如下形式键入：2020原值：2222（空格）原值：3333（空格）原值：4444（空格）原值：5555

2）在命令行提示符状态下输入：

D 2020

屏幕将显示从2020内存单元开始的值，其中2020H-~2023H的值为：

2222 3333 4444 5555

问题：断电后重新启动教学实验机，用D命令观察内存单元2020~2023的值会发现

什么问题，为什么？

（2）用A命令输入一段程序，执行并观察结果。

在命令行提示符状态下输入：

A2000

屏幕将显示：2000：

按如下形式键入：

2000: MVRD R0, AAAA

MVRD R1, 5555

AND R0, R1

RET

问题：采用单步和连续两种方式执行这段程序，察看结果，断电后发生什么情况？

5.将扩展芯片上方标有 EXTROMH和 EXTROML的“cs信号用自锁紧线短接，然后短接到 MEMDC138片的上方的标有“4000-5ff地址单元。

6.58C65的读操作和一般的RAM一样，而其写操作，需要一定的时间，大约为1毫秒因此，需要编写一延迟子程序，在对 EPROM进行写操作时，调用该子程序，以完成正确的读写；

1. 用E命令改变内存单元的值并用D命令观察结果。

1）在命令行提示符状态下输入：

E 5000

屏幕将显示：5000 内存单元原值：

按如下形式键入：

5000原值：2424（按空格）原值：3636（按空格）原值：4848（按空格）原值：5050

2）在命令行提示符状态下输入：

D 5000

屏幕将显示5000H-507FH内存单元的值，从5000开始的连续四个内存单元的值依次

为2424 3636 4848 5050。

1. 断电后重新启动，用D命令察看内存单元50050的值，会发现这几个单元的值没有发生改变，说明 EEPROM的内容断电后可保存。

（2）58C65存储器不能直接用A命令输入程序，单字节的指令可能会写进去，双字节指令的低位会出错，本实验将程序放到RAM（6116）中，调用延时子程序，访问58C65中的内存地址。下面给出的程序，在5000H-500FH单元中依次写入数据0000H、0001H、...000FH

从2000H单元开始输入主程序：

（2000）MVRD R0,0000

MVRD R2，0010；

MVRD R3，5000；

（2006）STRR[R3]，R0；

CALA 2200；

INC R0；

INC R3；

DEC R2；

JRN 2006；

RET

从2200H单元开始输入延时子程序：

（2200）PUSH R3

MVRD R3, FFFF

（2203）DEC R3

JRNZ 2203

POP R3

RET

运行主程序，在命令提示符下输入：G 2000。程序执行结束后，在命令提示符下输入：D 5000；可看到从5000H开始的内存单元的值变为

5000:0000 0001 0002 0003 0004 0005 0006 0007

5008:0008 0009 000A 000B 000C 000D 000E 000F

六、思考题：

思考题一：为何能用E命令直接写58C65的存储单元，而A命令则有时不正确；

答：A命令时的单字节指令可能会写进去，双字节指令的低位会出错。

思考题二：修改延时子程序，将其延时改短，可将延时子程序中R3的内容赋成00FF或0FFF等，再看运行结果。

答：运行结果不变，运行速度加快。

七、实验心得：

通过本次试验的难度在于怎样弄清楚ROM芯片和RAM芯片在功能和使用方法等方面的相同和差异之处：学习编程器设备向EEPROM芯片内写入一批数据的过程跟方法的工作原理，在做实验的时候也遇到一些相应的疑问，RAM和EPROM、EEPROM存储器芯片在读写控制跟写入时间等方面的同异之处，并正确建立连接关系和在过程中完成正确的读写过程。



计算机 学院 软件工程 专业 （1） 班\_\_\_\_\_组、学号3119005028

姓名 魏耀辉 协作者 陆锦源 教师评定\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

实验题目 实验四：组合逻辑控制器部件教学实验

一、实验目的：

1、深入理解计算机控制器的功能、组成知识；

2、深入地学习计算机各类典型指令的执行流程；

3、对指令格式、寻址方式、指令系统、指令分类等建立具体的总体概念

4、学习组合逻辑控制器的设计过程和相关技术。

二、实验设备与器材：

1.一台串口工作良好的PC机；

2.TEC-XP一台。

三、**实验**说明和原理**：**

1、TEC-XP16教学机的组合逻辑控制器主要由MACH器件组成；

2、TEC-XP16教学机上已实现了29条基本指令；

3、应了解监控程序的A命令只支持基本指令，扩展指令应用E命令将指令代码写入到相应的存储单元中不能用T、P命令单步调试扩展指令，只能用G命令执行有扩展指令的程序；

4、要明白TEC-XP16学机支持的指令格式及指令执行流程分组情况：理解TEC-XP教学机中已经设计好并正常运行的各类指令的功能、格式和执行流程，也包括控制器设计与实现中的具体线路和控制信号的组成；

5、要明确自己要实现的指令格式、功能、执行流程设计中必须遵从的约束条件。

（1）确定指令格式和功能，要受到教学机已有硬件的约束，应尽量与已实现指令的格式和分类办法保持一致；

（2）划分指令执行步骤并设计每一步的执行功能，设计节拍状态的取值，应参照已实现指令的处理办法来完成，特别要注意的是，读取指令的节拍只能用原来已实现的，其他节拍的节拍状态也应尽可能地与原用节拍的状态保持一致和相近；

（3）在指令流程表中填写每一个控制信号的状态值基本上是个查表填数的过程，应该特别仔细，并有意识地体会这些信号的控制作用；

（4）在给出的MACH的源文件中添加扩展指令的控制信号的逻辑表达式，编译适配后下载到MACH器件中；

（5）写一个包含你设计的指令的程序，通过运行该程序检查执行结果的正确性，初步判断你的设计是否正确：如果有问题，通过几种办法查出错误并改正，（比如手动置指令，单步调试每个节拍对应的控制信号）继续调试，直到完全正确。

四、实验内容：

1.完成控制器部件的教学实验，生自己设计几条指令的的功能、格式和执行流程，并在教学计算机上实现、调试正确；

2.看懂TEC XP+学计算机的功能部件组成和线路逻辑关系，分析教学计算机中已经设计 好并正常运行的几条典型指令的功能、格式和执行流程；

3.设计几条指令的功能、格式和执行流程，并在教学计算机上实现、调试正确。

4.单条运行指令，查看指令的功能、格式和执行流程。

5.用监控程序的A、E（扩展指令必须用E命令置入）命令编写一段小程序，观察运行结果。实验时将教学机左下方的5个拨动开关置为00110，运行编写的小程序观察终端显示的结果，检验设计的指令是否正确，若与预定结果不符，可查看指令的功能、格式、执行、流程设计的是否正确。

**五、**实验步骤：

1、接通教学机电源。

2、将教学机左下方的6个拨动开关置111100

3、按一下 “RESET”按键。

4、通过16位的数据开关SWH、SWL置入16位的指令操作码。

5、在单步方式下，通过指示灯观察各类基本指令的节拍。

（1）选择基本指的A组指令中的ADD指令，观察其节拍流程：

1）置拨动开关SW=00000000 00000001；

2）按RESET按键；节拍指示灯T3~T显示1000；

3）按 START按键节拍指示灯T3~TO显示0000；

4）按START按键；节拍指示灯T3~TO显示0010；

5）按START按键：节拍指示灯T3~TO显示0011；A组指令的执行除公共节拍外，只需一步完成

（2）选择基本指令的B组指令中的PUSH指令，观察其节拍流程：

1）置拨动开关SW=10000101 00000000；

1. 按 RESET按键；节拍指示灯T3~TO显示1000；
2. 按START按键；节拍指示灯T3~T显示0000；
3. 按 START按键；节拍指示灯T3~TO显示0010；
4. 按 START按键；节拍指示灯T3~T0显示0110；
5. 按 START按键：节拍指示灯T3~T0显示0100；B组指令的执行除公共节拍外，需两步完成。
6. 选择基本指令的D组指中的CALA指令，观察其节拍流程：

1）拨置动开关SW=11001110 00000000；

2）按RESET按键；节拍指示灯T3~T0显示1000；

3）按START按键；节拍指示灯T3~T0显示0000；

4）按START按键；节拍指示灯T3~T0显示0010；

5）按START按键；节拍指示灯T3~T0显示0110；

6）按START按键；节拍指示灯T3~T0显示0100；

1. 按START按键；节拍指示灯T3~T0显示0111
2. 按START按键；节拍指示灯T3~T0显示0101；可以看到，D组指令CALA除公共节拍外，需四步完成；

6、单步方式下，通过指示灯观察各类基本指令的控制信号。

（1）选择指令的A组指令中的SHR指令，观察其执行过程中控制信号的变化，分析其作用。

1）置拨动开关SW=0000101100010000

2）先按“RESET”按键；再连续按 “START”按键，观察每一步的节拍及控制信号如下表：

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 节拍 | 指令 | 编码 | /MIO | REQ | /WE | A | B | Sci | SSH | I8-6 | I5-3 | I2-0 | SST | DC1 | DC2 |
| 1000 |  |  | 1 | 0 | 0 | 0101 | 0101 | 01 | 0 | 011 | 001 | 001 | 000 | 000 | 111 |
| 0000 |  |  | 1 | 0 | 0 | 0101 | 0101 | 01 | 0 | 010 | 000 | 011 | 000 | 000 | 011 |
| 0010 |  |  | 0 | 0 | 1 | 0000 | 0000 | 00 | 0 | 001 | 000 | 000 | 000 | 000 | 001 |
| 0011 | SHR | 0000 1001 | 1 | 0 | 0 | 0000 | 0001 | 00 | 0 | 101 | 000 | 011 | 101 | 000 | 000 |

（2）选择基本指令的B组指令中的JMPA指令，观察其执行过程中控制信号的变化，分析其作用。

1）置拨动开关SW=100000：（表示指令JMPA）

2）先按“RESET”按键；再连续按“START”按键，观察每一步的节拍及控制信号

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 节拍 | 指令 | 编码 | /MIO | REQ | /WE | A | B | Sci | SSH | I8-6 | I5-3 | I2-0 | SST | DC1 | DC2 |
| 1000 |  |  | 1 | 0 | 0 | 0101 | 0101 | 01 | 0 | 011 | 001 | 001 | 000 | 000 | 111 |
| 0000 |  |  | 1 | 0 | 0 | 0101 | 0101 | 01 | 0 | 010 | 000 | 011 | 000 | 000 | 011 |
| 0010 |  |  | 0 | 0 | 1 | 0000 | 0000 | 00 | 0 | 001 | 000 | 000 | 000 | 000 | 001 |
| 0110 | JMPA | 1000  0000 | 1 | 0 | 0 | 0101 | 0101 | 01 | 0 | 010 | 000 | 011 | 000 | 000 | 011 |
| 0100 | JMPA | 1000  0000 | 0 | 0 | 1 | 0000 | 0101 | 00 | 0 | 011 | 000 | 111 | 000 | 000 | 000 |

（3）选择基本指令的D组指令中的CALA指令，观察其执行过程中控制信号的变化，分析其作用.

1）置SW=11001110 00000000；

2）先按RESET按键：再连续按START按键，观察每一步的节拍及控制信号如下表

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 节拍 | 指令 | 编码 | /MIO | REQ | /WE | A | B | Sci | SSH | I8-6 | I5-3 | I2-0 | SST | DC1 | DC2 |
| 1000 |  |  | 1 | 0 | 0 | 0101 | 0101 | 01 | 0 | 011 | 001 | 001 | 000 | 000 | 111 |
| 0000 |  |  | 1 | 0 | 0 | 0101 | 0101 | 01 | 0 | 010 | 000 | 011 | 000 | 000 | 011 |
| 0010 | CALA | 1100  1110 | 0 | 0 | 1 | 0000 | 0000 | 00 | 0 | 001 | 000 | 000 | 000 | 000 | 001 |
| 0110 | CALA | 1100  1110 | 1 | 0 | 0 | 0101 | 0101 | 01 | 0 | 010 | 000 | 011 | 000 | 000 | 011 |
| 0100 | CALA | 1100  1110 | 0 | 0 | 1 | 0000 | 0000 | 00 | 0 | 000 | 000 | 111 | 000 | 000 | 000 |
| 0111 | CALA | 1100  1110 | 1 | 0 | 0 | 0000 | 0100 | 00 | 0 | 011 | 001 | 011 | 000 | 000 | 011 |
| 0101 | CALA | 1100  1110 | 0 | 0 | 0 | 0101 | 0101 | 00 | 0 | 010 | 000 | 010 | 000 | 001 | 000 |

7、在以上几步实验的基础上，选择几条扩展指令，设计出扩展指令的节拍 和每拍对应的控制信号。

（1）选择扩展指令的A组指令中的RCR指令，观察其节拍流程：

1）、置拨动开关SW＝00101011 00010000；

2）、按RESET按键；节拍指示灯T3 ~T0显示1000；

3）、按START按键；节拍指示灯T3~T0显示0000；

4）、按START按键；节拍指示灯T3~T0显示0010；

5）、按START按键；节拍指示灯T3~T0显示0011；可以观察到A组拓展指令除公共节拍外，只需一步完成。

（2）选择扩展指令的C级指令中的LDRA指令，观察其流程：

1）、置拨动开关SW＝11100100 00000000；

2）、按RESET按键；节拍指示灯T3 ~T0显示1000；

3）、按START按键；节拍指示灯T3~T0显示0000；

4）、按START按键；节拍指示灯T3~T0显示0010；

5）、按START按键；节拍指示灯T3~T0显示0110；

6）、按START按键；节拍指示灯T3~T0显示0111；

7）、按START按键；节拍指示灯T3~T0显示0101；可以观察到C组拓展指令除公共节拍外，只需三步完成。

8、设计几条扩展指令的控制信号如下：

（1）选择扩展指令ADC、STC、JRS、LDRX和JMPR。

（2）根据设计的控制信号的表格用ABEL语言编写MACH的逻辑表达式。

（3）将编好的程序MACHC.JED下载到MACH芯片内。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 节拍 | 指令 | 编码 | /MIO | REQ | /WE | A | B | Sci | SSH | I8-6 | I5-3 | I2-0 | SST | DC1 | DC2 |
| 1000 |  |  | 1 | 0 | 0 | 0101 | 0101 | 01 | 0 | 011 | 001 | 001 | 000 | 000 | 111 |
| 0000 |  |  | 1 | 0 | 0 | 0101 | 0101 | 01 | 0 | 010 | 000 | 011 | 000 | 000 | 011 |
| 0010 |  |  | 0 | 0 | 1 | 0000 | 0000 | 00 | 0 | 001 | 000 | 000 | 000 | 000 | 001 |
| 0011 | ADC | 00100000 | 1 | 0 | 0 | SR | DR | 10 | 0 | 011 | 000 | 001 | 001 | 000 | 000 |
|  | JRS | 01100100 | 1 | 0 | 0 | 0101 | 0101 | 00 | 0 | 0S1 | 000 | 101 | 000 | 010 | 000 |
|  | STC | 01101101 | 1 | 0 | 0 | 0000 | 0000 | 00 | 0 | 001 | 000 | 000 | 100 | 000 | 000 |
|  | JMPR | 01100000 | 1 | 0 | 0 | SR | 0101 | 00 | 0 | 011 | 000 | 100 | 000 | 000 | 000 |
| 0110 | LDRX | 11100101 | 1 | 0 | 0 | 0101 | 0101 | 01 | 0 | 010 | 000 | 011 | 000 | 000 | 011 |
|  | STRX | 11100110 | 1 | 0 | 0 | 0101 | 0101 | 01 | 0 | 010 | 000 | 011 | 000 | 000 | 011 |
| 0111 | LDRX | 11100101 | 0 | 0 | 1 | SR | 0000 | 00 | 0 | 001 | 000 | 101 | 000 | 000 | 011 |
|  | STRX | 11100101 | 0 | 0 | 1 | 0000 | DR | 00 | 0 | 011 | 000 | 111 | 000 | 000 | 000 |
| 0101 | LDRX | 11100110 | 0 | 0 | 0 | 0000 | DR | 00 | 0 | 001 | 000 | 011 | 000 | 001 | 000 |

9、用教学机已实现的基本指令和扩展的几条指令编写程序并运行，测试扩展的几条指令是否正确。

六、实验心得：

通过本次组合逻辑控制部件实验， 对组合逻辑控制器部件的功能和组成知识有了更进一 步的了解和认知，在此过程中学习了计算机各类典型指令的执行流程，对指令格式、寻址方式、指令系统、指令分类等建立了具体概念。



计算机 学院 软件工程 专业 （1） 班\_\_\_\_\_组、学号3119005028

姓名 魏耀辉 协作者 陆锦源 教师评定\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

实验题目 实验五：微程序控制器部件教学实验

一、实验目的：

1、全面掌握计算机各部件组成及相互联接关系；

2、深入理解计算机微程序控制器的功能、组成知识；

3、深入地学习计算机各类典型指令的执行流程；

4、对指令格式、寻址方式、指令系统、指令分类等建立具体的总体概念；

5、学习微程序控制器的设计过程和相关技术。

二、实验设备与器材：

1.一台串口工作良好的PC机；

2.TEC-XP一台。

三、**实验**说明和原理**：**

控制器设计是学习计算机总体组成和设计的最重要的部分。要在TEC-XP+教学计算机完成这项实验，必须比较清楚地懂得：

1、TEC-XP+教学机的微程序控制器主要由微程序定序器AM2910、产生当前微地址和下地址的微控存和MACH器件组成；

2、TEC-XP+教学机上已实现的全部基本指令和留给用户实现的19条扩展指令的控制信号都是由微控存和MACH给出的；

3、应了解监控程序的A命令只支持基本指令，扩展指令应用E命令将指令代码写入到相应的存储单元中不能用T、P命令单步调试扩展指令，只能用G命令执行扩展指令；

4、要明白TEC-XP+教学机支持的指令格式及指令执行流程分组情况：理解TEC-XP+教学机中已经设计好并正常运行的各类指令的功能、格式和执行流程也包括控制器设计与实现中的具体线路和控制信号的组成。

1. 要明确自己要实现的指令格式、功能、执行流程设计中必须遵从的约束条件。为了完成自己设计几条指令的的功能、格式和执行流程，并在教学计算机上实现、调试正确的实验字段的使用和分配要受教学机已有硬件的约束，应尽量与已实现指令的格式和分类办法保持一致。 内容，具体过程包括

（1）确定指格式和功能，包括确定要用的操作码，指令中其它字段的内容分配与使用。指令中；

（2）按新指的功能和格式，设计指令的执行流程划分指令执行步骤并设计每一步的执行功能，设计微地址和下地址的取值，应参照已实现指令的处理办法来完成；

（3）在指令流程表中填写每一个控制信号的状态值基本上是个查表填数的过程，应该特别细，并有意识地体会这些信号的控制作用；

（4）将设计好的微码，装入控制存储器的相应单元；

（5）写一个包含你设计的指令的程序，通过运行该程序检查执行结果的正确性，来初步判断你的设计是否正确：如果有问题，通过几种办法查出错误并改正，继续调试，直到完全正确。

四、实验内容：

1、完成控制器部件的教学实验，主要内容是自己设计几条指令的功能、格式和执行流程，并在教学计算机上实现、调试正确；

2、首先是看懂TEC-XP+教学计算机的功能部件组成和线路逻辑关系，然后分析教学计算机中已经设计好并正常运行的几条典型指令（的功能、格式和执行流程，注意各操作功能所对应的控制信号的作用；

3、设计几条指令的功能、格式和执行流程，并在教学计算机上实现、调试正确；

4、单条运行指令，查看指令的功能、格式和执行流程；

1. 用监控程序的A、E（扩展指令必须用E命令置入）命编写一段小程序，观察运行结果

**五、**实验步骤：

1、接通教学机电源 ；

2、将教学机左下方的六个拨动开关置为110100；

3、按下“RESET”按键；

4、通过16位的数据开关SWH、SWL置入指令操作码

5、在单步方式下，通过指示灯观察各类基本指令的微码。

1. 选择基本指令的A组指令中的ADD指令，观察其节拍流程

1）置拨动开关S=00000000 00000001；

2）按RESET按键指示灯 Microp亮，其它灯全灭；

3）按START按键：指示灯CI3~0、SCC3~0显示1110 0000和下址的指示灯全灭；

4）按START按键：指示灯CI3~0、SCC3~0显示1110 0000，微址指示灯显示0000 0001，下址的指示灯全灭；

5）按START按键：指示灯CI3~0、SCC3~0显示1110 0000，微址指示灯显示0000 0010，下址的指示灯全灭；

6）以上三步为公共操作，其它指令同；

7）按START按键：指示灯CI3~0、SCC3~0显示0010 0000，微址指示灯显示0000 0011，下址的指示灯显示0000 0100；

8）按START按键：指示灯CI3~0、SCC3~0显示0011 0000，微址指示灯显示0000 0100，下址的指示灯显示0011 0000；

9）按START按键：指示灯CI3~0、SCC3~0显示0011 0010，微址指示灯显示0011 0000，下址的指示灯显示0011 1010；

10）按START按键：指示灯CI3~0、SCC3~0显示0011 0000，微址指示灯显示0011 0001，下址的指示灯显示0000 0010；

（2）选择基本指令的B组指令中的MVRD指令，观察其节拍流程。

（3）选择基本指令的D组指令中的CALA指令，观察其节拍流程。

6、在连续方式下，用A命令键入程序并运行；

7、设计几条指令的功能、格式和执行流程，设计每条微指令各字段的具体编码值，包括控制码的各字段、下地址字段、形成下址用到的条件码；

（1）扩展几条指令，确定各步的控制信号。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 指令 | 操作功能 | 微址 | 下址 | CI3-0 | SCC3-0 | MRW | I2-0 | I8-I6 | I5-3 | B | A | SST | SSH  SCI | DC2 | DC1 |
| ADC | DR+SR+CF->DR | 50 | 30 | 0011 | 0000 | 100 | 001 | 011 | 000 | 0000 | 0000 | 001 | 010 | 000 | 000 |
| STC | STC | 57 | 30 | 0011 | 0000 | 100 | 000 | 001 | 000 | 0000 | 0000 | 100 | 000 | 000 | 000 |
| LDRA | PC->AR  PC+1->PC | 5B | 00 | 1110 | 0000 | 100 | 011 | 010 | 000 | 0101 | 0101 | 000 | 001 | 011 | 000 |
|  | MEM->AR | 5C | 1C | 0011 | 0000 | 001 | 111 | 001 | 000 | 0000 | 0000 | 000 | 000 | 011 | 000 |
| CALR | SP-1->SP,AB | 64 | 00 | 1110 | 0000 | 100 | 011 | 011 | 001 | 0100 | 0000 | 000 | 000 | 011 | 000 |
|  | PC->MEM | 65 | 00 | 1110 | 0000 | 000 | 100 | 001 | 000 | 0000 | 0101 | 000 | 000 | 000 | 001 |
|  | SP->PC | 66 | 30 | 0011 | 0000 | 100 | 100 | 011 | 000 | 0101 | 0000 | 000 | 000 | 000 | 000 |

8、在单步方式下，通过指示灯观察各类扩展指令的微码。

9、用A、E键入程序连续运行。

六、实验心得：

本次实验让我对指令的格式、寻址方式以及指令系统有了一个总体概念，特别是对于扩展指令的认识与使用得到了加深，并且可以自主实现自设微码调用微指令，初步掌握了微程序控制器的设计过程。通过对指令的执行，更为深入地知晓汇编典型指令的执行流程。当我们使用扩展指令时，需要用E命令输入。



计算机 学院 软件工程 专业 （1） 班\_\_\_\_\_组、学号3119005028

姓名 魏耀辉 协作者 陆锦源 教师评定\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

实验题目 实验六：输入/输出接口拓展实验

一、实验目的：

1、了解串行接口与计算机主机之间的数据传送方式；

2、理解串行接口芯片的内部组成和传送数据的运行过程；

3、了解串行接口在投入运行之前必须执行的初始化操作的作用及完成初始化操作的具体方案。

二、实验设备与器材：

1.两台串口工作良好的PC机；

2.TEC-XP两台。

三、**实验**说明和原理**：**

1) 串行通信是指外设和计算机间使用一根数据信号线,数据在一根数据信号线上按位进行传输，每一位数据都占据一个固定的时间长度。

2) 串行端口的本质功能是作为CPU和串行设备间的编码转换器。当数据从 CPU经过串行端口发送出去时，字节数据转换为串行的位。在接收数据时，串行的位被转换为字节数据。

四、实验内容：

1、为扩展I/O口选择一个地址，即将与COM2口相连的8251的/C与标有VOCS的一排插孔中的一个相连；

2.将COM2口与终端或另一台运行有PCEC16的PC机的串口相连；

3、用监控程序的A命令，编写一段小程序，先初始化COM2口，再向COM2口发送一些字符，也可从COM2口接收一些字符，或实现两个串口的通信。

**五、**实验步骤：

1、为扩展O口选择一个地址：将与COM2口相连的8251的/CS与标有O/CS的插孔中地址为A0-AF的一个相连；

2、将一台教学机COM1口与一台PC机相连，在PC机上启动PCEC16EXE

3、将另一台教学机COM1口与另一台PC机相连，同样启动PCEC16EXE

4、用一根串口线将第一台的教学机的COM2口和另一台教学机的COM2口相连

5、接通教学机电源。

6、将教学机左下方的6个拨动开关置为001100；

7、按一下“RESET”按键，接着按“START”按键。

8、在两台PC机对应的PCEC上分别输入以下程序：

从2000H单元开始输入下面的程序

2000：MVRD R0，004；

2002：OUT A1；

2003：MVRD R0，0037；

2005：OUT A1；

2006：IN 81；

2007：SHR R0；

2008：SHR R0；

2009：JRNC 200D；

200A：IN 80；

200B：OUT 80；

200C：OUT A0；

200D：IN A1；

200E: SHR R0；

200F: SHR R0；

2010：JRNC 2006；

2011：IN A0；

2012：OUT 80；

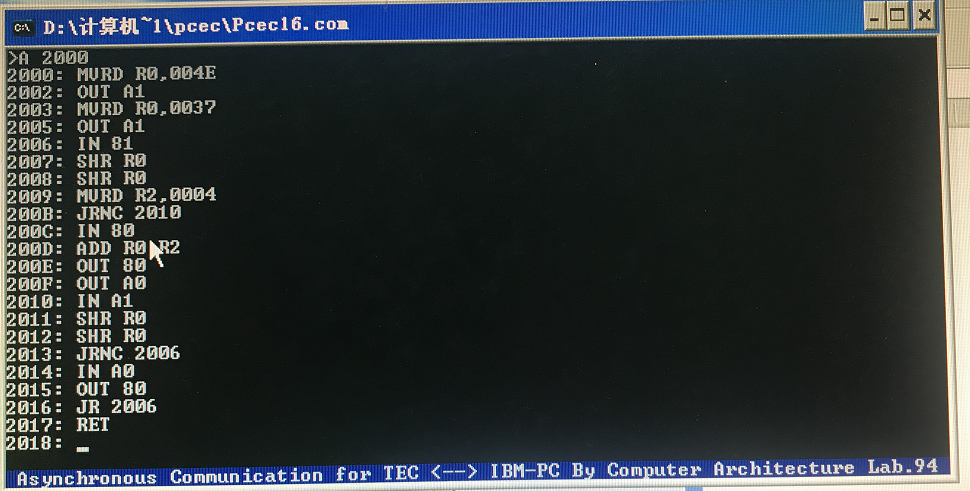
2013：JR 2006；

2014：RET

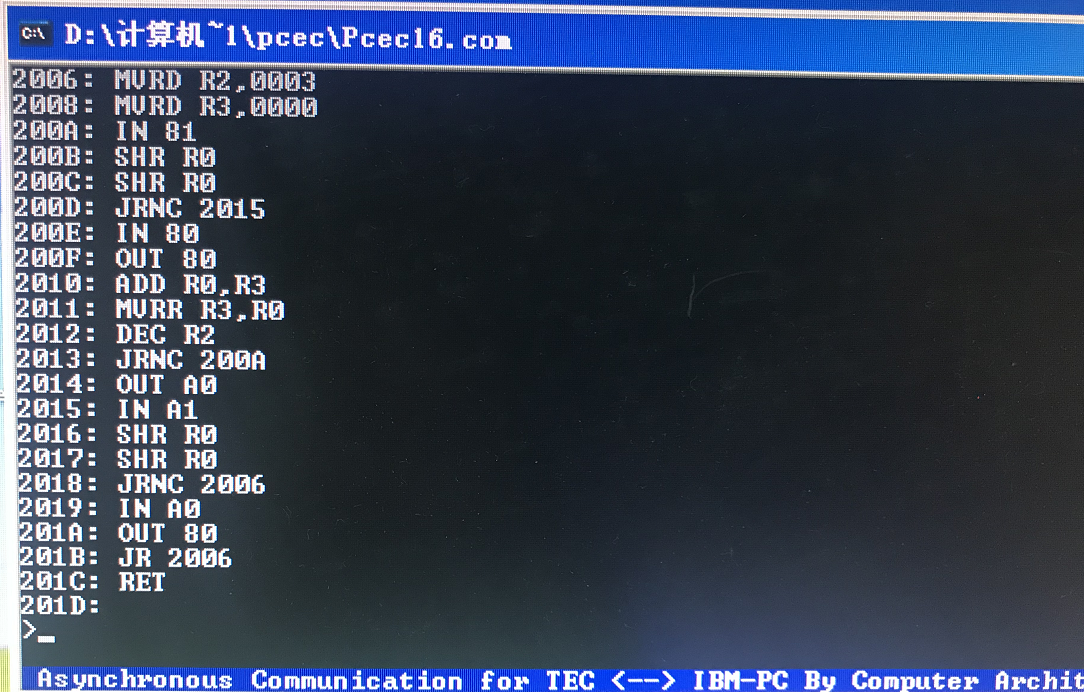
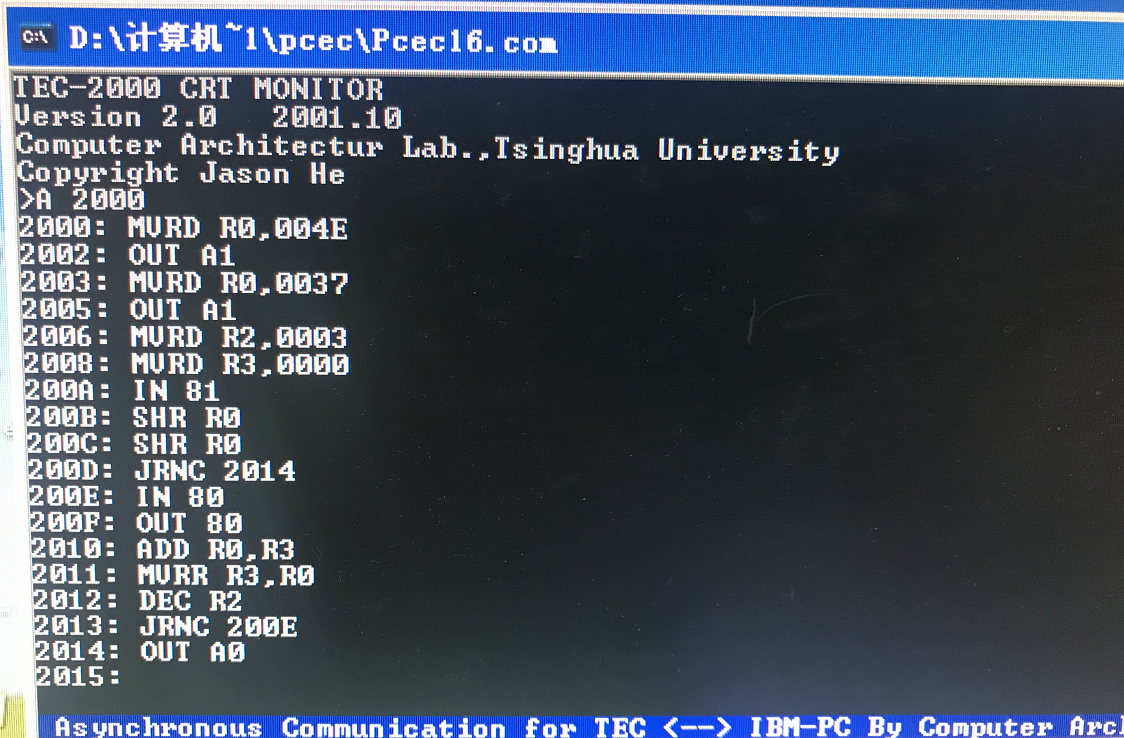
该程序完成两台教学计算机的第2个串行接口扩展操作并完成该串口初始化，启动两台教学机，都运行这个程序

六、思考题：

思考题一：运行程序，使两台PC机实现加密通信



思考题二：运行程序，使两台PC机实现加密和解密通信；



七、实验心得：

本次实验主要是实现两个PC机的串行通讯，需要解决的是两台TEC-XP教学机的COM口必须完好，在实验初期由于COM口的损坏导致排查原因耗费的时间十分多，由于有实验一的基础，在实现加密和解密的汇编语言程序设计时较为轻松，能够独立地完成加密解密通信的程序。



计算机 学院 软件工程 专业 （1） 班\_\_\_\_\_组、学号3119005028

姓名 魏耀辉 协作者 陆锦源 教师评定\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

实验题目 实验七：中断实验

一、实验目的：

1、加深理解计算机系统中断的工作原理及处理过程。

2、学习和掌握中断产生、响应、处理等技术。

3、掌握中断服务子程序的编写要点，进行一次硬、软件的综合调试。

二、实验设备与器材：

1.一台串口工作良好的PC机；

2.TEC-XP一台。

三、**实验**说明和原理**：**

（1）要求中断隐指令中执行关中断功能，如果用户中断服务程序允许被中断，必须 在中断程序中执行EI开中断命令。

（2）教学机的中断系统共支持三级中断，由三个无锁按键确定 从右到左依次为一、二、三级中断，对应的P1、P0的编码分别是01、10、11，优先级也依次升高，这决定 了它们的中断向量，为XXX4、XXX8、XXXC。可以看到，每级中断实际可用空间只有四个字节，故这个空间一般只存放一条转移指令，而真正的用户中断服务程序则存放在转移指令所指向的地址。

（3）用户需扩展中断隐指令、开中断指令、产中断指令、中断返回指令及其节拍。

四、实验内容：

1.扩展中断隐指令，为中断隐指令分配节拍，中断隐指令用到12个节拍，为了和一般指令相区别，应将其节拍T3设计为1.

2.扩展开中断指令EI、关中断指令DI、中断返回指令IRET。

3.确定中断向量地址。中断微量的高12位由开关确定为（0001001000000）。三级中断对应的中断微量为2404H、2408H、240CH。当有中断请求且被响应后，将执行存放在该中断的中断微量所指向的内存区的指令。

4.真写中断微量表。在上述2404H、2408H、240CH地址写入三条JR转移指令，JR指令的OFFSSET是-128~127之间，但在PCEC16中输入时，用户不需要计算偏移量，直接输入要转向的绝对地址即可。

5.编写中断服务程序。中断服务程序可以放在中断微量表之后，中断服务程序可以实现在程序正常运行时在计算机屏幕上显示与优先级相对应的不同字符。

6.写主程序。可编写一死循环，要求先开中断。

**五、**实验步骤：

（1）填写中断向量表

1）将数据开关的高12位设置成：0010 0100 0000，即选择三级中断的中断向量为2404H、2408H、240CH。

2）中断向量一共有16位，高12位由数据开关SWH7-0和SWL7-4决定；后四位有无锁按键决定，分别为01、10、11，所以中断向量的16位为2404 、2408 、240C。

3）向量表指令填写：

（2404）JR 2420

（2408）JR 2430

（240C）JR 2440

（2）编写中断程序

用A、E命令从2420H单元开始输入下面和程序：（标有\*的语句表示要用E命令输入）

2420：PUSH R0

2421：PUSH R3

2422：MVRD R3,31

2424：JR 2450

2430：PUSH R0

2431：PUSH R3

2432：MVRD R3,32

2434：JR 2450

2440：PUSH R0

2441：PUSH R3

2442：MVRD R3,33

2444：JR 2450

\*2450：EI

2451：MVRD R0,0042

2453：CALA 2200

2455：MVRD R0,0049

2457：CALA 2200

2459：MVRR R0,R3

245A：CALA 2200

245C：IN 81

245E：SHR R0

245F：JRNC 245C

2460：IN 80

2461：MVRD R00045

2463：CALA 2200

2465：MVRD R0,0049

2467：CALA 2200

2469：MVRR R0,R3

246A：CALA 2200

246C：POP R3

246D：POP R0

\*246E：IRET

（3）编写子程序

2200：PUSH R0

2202：SHR R0

2203：JRNC 2201

2204：POP R0

2205：OUT 80

2206：RET

（4）编写主程序

\*2000：EI

2001：MVRE R0,0036

2003：CALA 2200

2005：MVRD R0,4000

2007：DEC R0

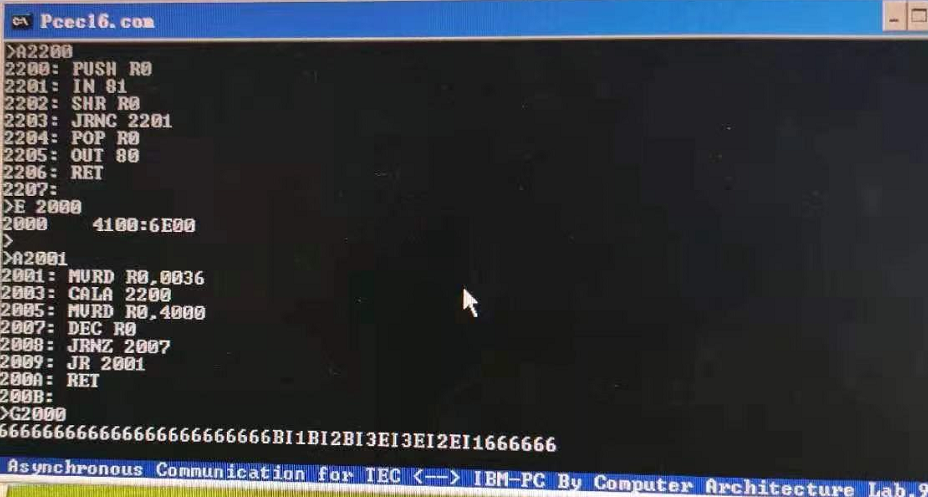
2008：JRNZ 2007

2009：JR 2001

200A：RET

（5）运行主程序，等待、响应中断。在命令行提示符状态下输入：G 2000

（6）屏幕将连续显示“6”。在程序执行过程中按下教学机右下方任意一个无锁按键。此时，教学机转向执行本级中断服务程序，在屏幕上显示BI以及按下的键对应的中断优先级。在接收键盘一个字符后，显示该字符并退出当前级的中断服务程序，恢复中断现场，接着执行断点处的程序。若在接收字符之前，又有更高一级的中断请求，则教学机转向执行高一级的中断服务程序，执行完后接着执行低级中断，然后 退出 执行主程序。需要注意的是若当前中断为高级中断，则不会响应低级中断简单的中断服务程序。



六、实验心得：

本次实验过程中，对中断的产生，响应以及处理机制加深了了解和熟悉。途中因为TEC-XP教学机无法正确与Pcec软件产生连接导致了实验一直未能完成，先熟悉流程及操作后借住别人的教学机进行实验，让我学会遇到问题可以先试着绕过它再解决它。



计算机 学院 软件工程 专业 （1） 班\_\_\_\_\_组、学号3119005028

姓名 魏耀辉 协作者 陆锦源 教师评定\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

实验题目 实验九：FPGA芯片实现非流水的CPU系统（综合实验）

一、实验目的：

1、进一步熟悉教学计算机的指令格式、指令编码、寻址方式和指令功能等内容。

2、进一步熟悉教学计算机的总体组成和各个部件的功能，理解控制器部件在计算机整机中的关键作用；

3、进一步理解和指令执行步骤的划分方案；

4、进一步熟悉教学计算机的硬连线控制器各个控制命令的控制功能，学习用VHDL语言描述节拍发生器和控制信号产生部件的功能。

5、进一步理解与熟悉在TH-union教学计算机控制器中处理原有指令和扩展指令的方案，提高对控制器功能能描述的理解程序。

二、实验设备与器材：

1.一台串口工作良好的PC机；

2.TEC-XP一台。

三、**实验**说明和原理**：**

1.用VHDL语言来描述的CPU的源码文件，经过专用工具软件的编译和综合后，经专用烧录器，把程序下载到FPGA芯片中，就得到了能够正常运行的CPU系统。

四、实验内容：

控制器实验可以在两个层次上进行：

1、第一个层次属于观察、验证性的实验，即通过多种方式，察看教学计算机指令的执行步骤、运行结果、各组控制信号在每一个执行步骤中的状态、指令之间的衔接等有关内容。这个层次的实验，重点在于学懂教学计算机中已有的设计结果，把实现基本指令的VHDL语言程序中的语句描述与教学机的运行结果对应清楚。

2、第二个层次是学生进行自己的设计与实现新的扩展指令的实验，即在教学机系统已有指令的基础上，由学生自己添加若干条新的指令进行，包括定义指令格式、功能。划分指令执行步骤和确定每一步的功能，确定每一执行步骤使用的全部控制信号的状态值，使用VHDL语句把新的设计结果描述正确并添加到已有的源程序代码中去，编译、下载并调试正确，写一个包含已有指令和刚刚实现的指令的小程序，检查运行结果的正确性，若发现错误则找出原因并设法改正，直到全部正确为止。

1. 实验步骤：

（1）按前述的步骤准备好实验机，连接好串口线和电源线，打开PCEC1.6EXE的仿真界面；将六个功能开关置为00X101；

（2）将TIMING GAL左方的波动开关置于“FPGA”端；

（3）确认标有“DataBus 15-8”和“DataBus 7-0”的数据总线的指示灯正文的插针断开；

（4）确认标有“AdressBus 15-8”和“AdressBus 7-0”的地址总线的指示灯下方的插针断开；

（5）将提供的带彩线的FPGA的下载线并口一端和计算机的并口连接，彩线一端按红色在左边的位置和大板上电源模块下方的一溶解度插针接好；

（6）打开实验机的电源；

（7）在PC机上打开ISE的软件；

（8）打开软件的下载界面，选择SERIAL方式，添加器件CPU.BIT，进行下载；

（9）下载完成关闭下载界面，启动PCEC界面，注意实验机不要断电。

（10）按一下“RESET”按键，再按一下“START”按键，主机上显示：

TEC-2000 CRT MONITOR

Version 1.0 April 2001

Computer Architectur Lab,Tsinghua University

Programmed by He Jia

>

（11）在FPGA构成的CPU的控制下将汇编语言程序设计的内容重新作一遍。

六、实验心得：

由本次实验可知，FPGA支持基本汇编指令集。FPGA系统则在指令系统、使用的软件资源等方面与小规模期间构成的实验箱兼容，能够使软件资源得到充分的应用。但是在对FPGA进行烧录的时候成功率并不是百分之一百的，默认下为了保护芯片，一旦发生数据烧录错误则必须从头开始烧录，为烧录带来了不少的麻烦。