苏州大学实验报告

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 院、系 | 计算机学院 | | 年级专业 | | 17计科 | | 姓名 | 魏军杰 | 学号 | 1727405175 |
| 课程名称 | | 计算机组成与系统结构 | | | | | | | 成绩 |  |
| 指导教师 | | 屈蕴茜 | | 同组实验者 | | 无 | | 实验日期 | 2019/6/12 | |

|  |  |
| --- | --- |
| 实 验 名 称 | 基本模型机设计与实现 |

1. 实验目的

1. 掌握一个简单CPU的组成原理

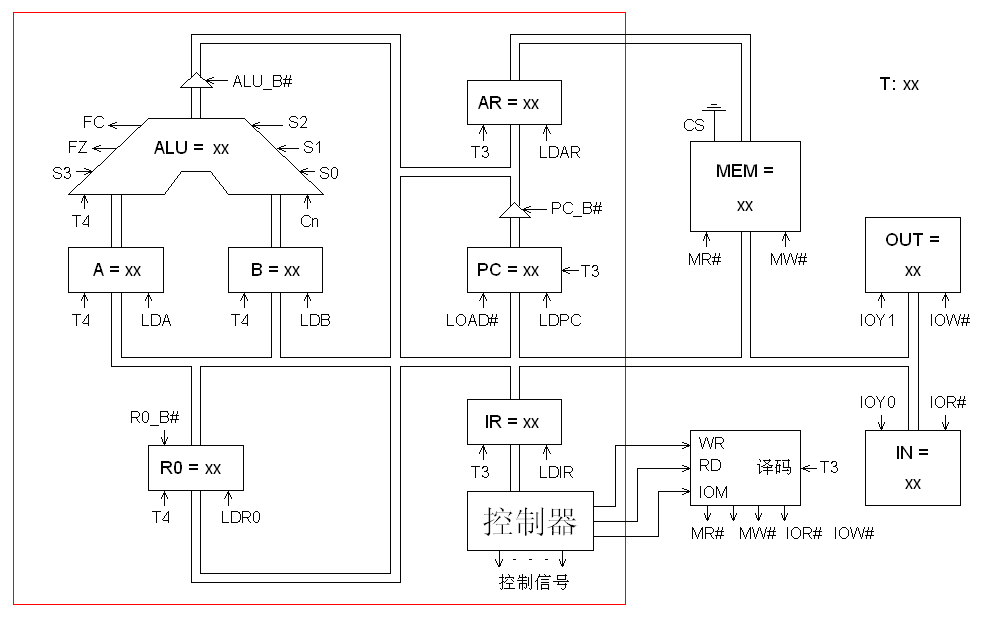
2. 在掌握部件单元电路实验的基础上，进一步将其构造成一台基本模型计算机。

3. 为该模型机定义五条机器指令，并编写相应的微程序，调试掌握整机概念。

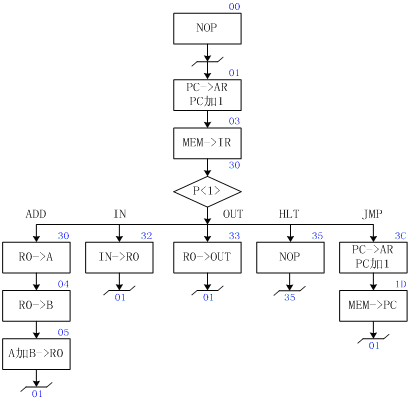
1. 实验设备

实验设备采用TDN-CMA教学实验箱

1. 实验原理

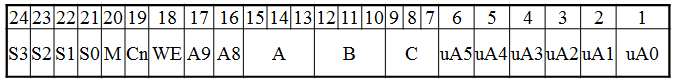


数据通路框图



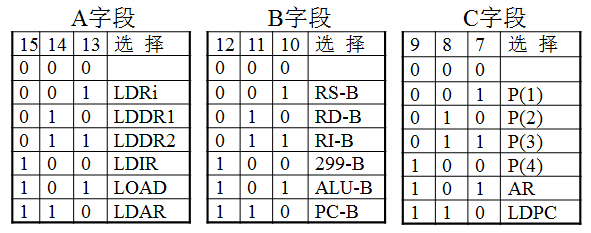
微程序流程图

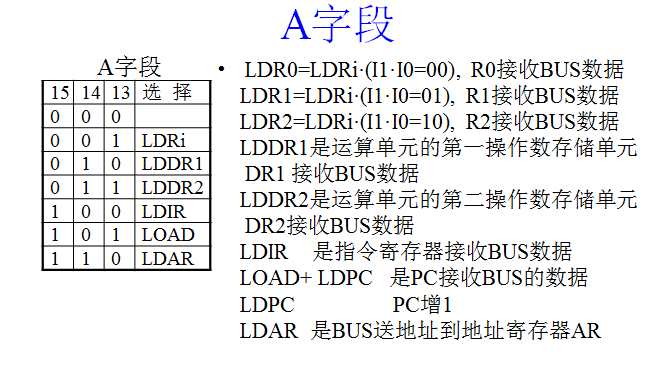
微指令格式：微指令字长共24位,uA5~uA0为6位的后继地址。A,B,C为三个译码字段

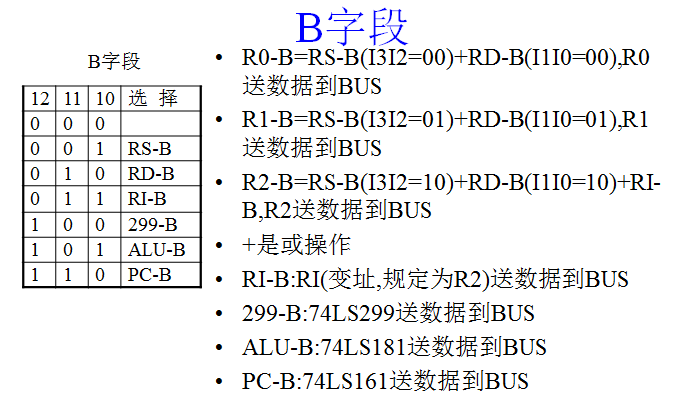


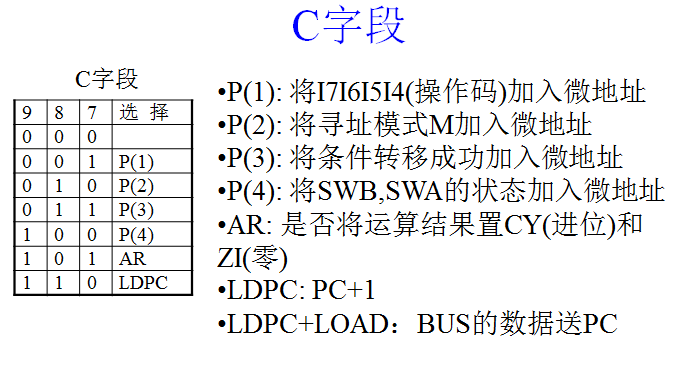
**微指令解释：**

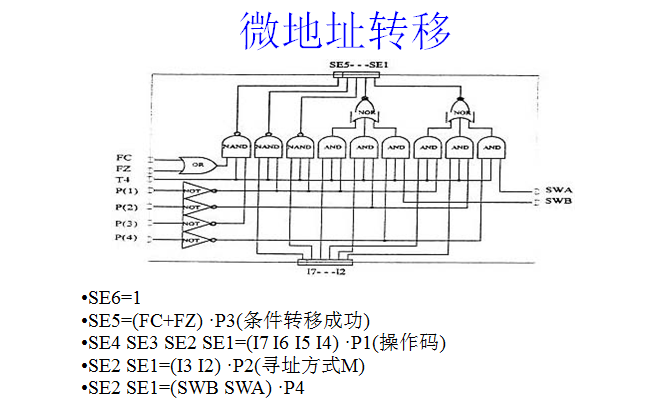
* S3,S2,S1,S0,M,Cn是74LS181的控制信号:
  + 000000且ALU-B为A加1
  + 000001且ALU-B为F=A
  + 000011且ALU-B为F=A’
  + 001111且ALU-B为F=0
  + 100101且ALU-B为A加B
  + 101110且ALU-B为F=AB
* WE是存储器的读/写控制信号:
  + WE=0为读信号 WE=1为写信号











微指令：

|  |  |  |
| --- | --- | --- |
| 助记符 | 机器指令码 | 说明 |
| IN | 0010 0000 | IN R0 |
| ADD | 0000 0000 | R0+R0 R0 |
| OUT | 0011 0000 | R0 OUT |
| JMP addr | 1110 0000 xxxx xxxx | addr PC |
| HLT | 0101 0000 | 停机 |

程序：

$P0020

$P0100

$P0230

$P03E0

$P0400

$P0550

微程序：

$M00000001

$M03107070

$M04002405

$M0504B201

$M30001404

$M32183001

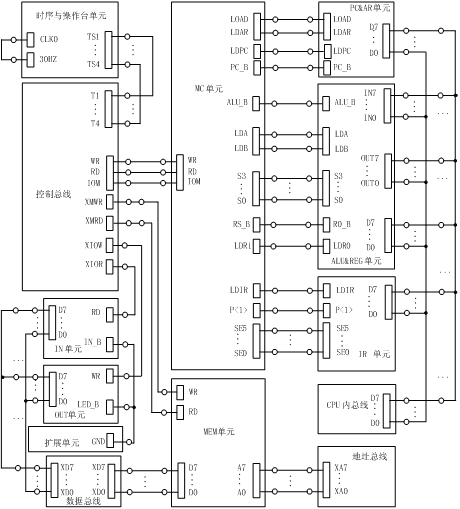
$M33280401

$M35000035

$M3C006D5D

1. 实验步骤

1. 按实验接线图连接实验线路，查线无误，接通电源。



2. 联机运行

将时序与操作台单元的开关KK1和KK3置为‘运行’档，单击【开始】/【程序】/TangDu/CMA/CMA的程序如图1-1-6所示。

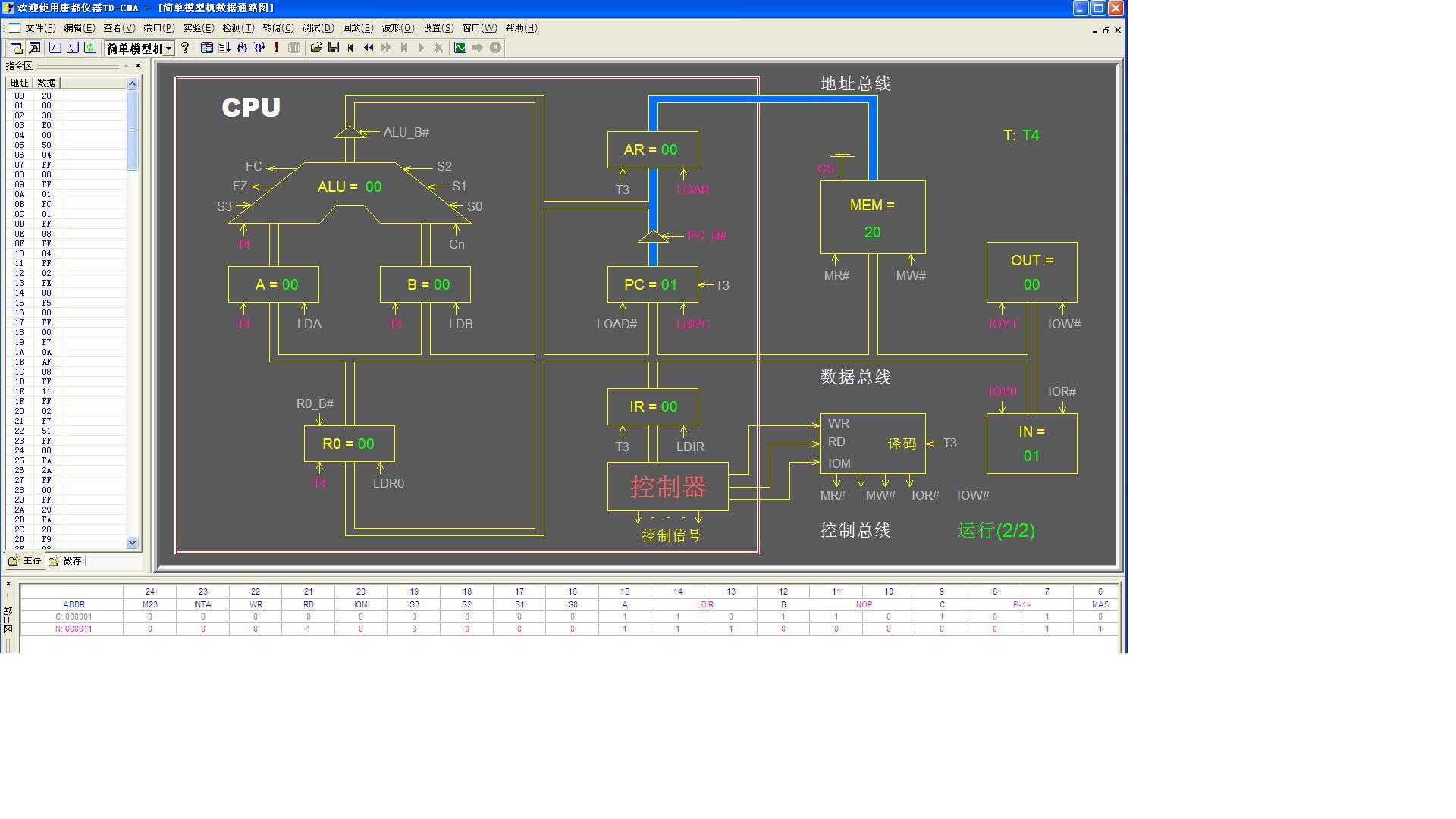
用联机软件的“【转储】—【装载C:/TangDu/CMA/CMA/Sample/CPU与简单模型机设计实验.Txt】”文件装载入实验系统。装入过程中，在软件的输出区的‘结果’栏会显示装载信息，如当前正在装载的是机器指令还是微指令，还剩多少条指令等。

选择联机软件的“【转储】—【刷新指令区】”可以读出下位机所有的机器指令和微指令。可以通过联机软件单独修改某个单元的微指令，先用鼠标左键单击指令区的‘微存’，然后再单击需修改单元的数据，此时该单元变为编辑框，输入6位修改数据并回车，编辑框消失，并以红色显示写入的数据

进入软件界面，选择菜单命令“【实验】—【简单模型机】”，打开简单模型机数据通路图。

3. 按动CON单元的总清按钮CLR，然后通过软件运行程序，选择相应的功能命令，即可联机运行、监控、调试程序，当模型机执行完JMP指令后，检查OUT单元显示的数是否为IN单元值的2倍。在数据通路图和微程序流中观测指令的执行过程，并观测软件中地址总线、数据总线以及微指令显示和下位机是否一致。

这是调试界面



1. 参照机器指令及微程序流程图，将屏幕显示的数据流图与理论分析比较，验证系统执行指令的正确性。
2. 实验内容：

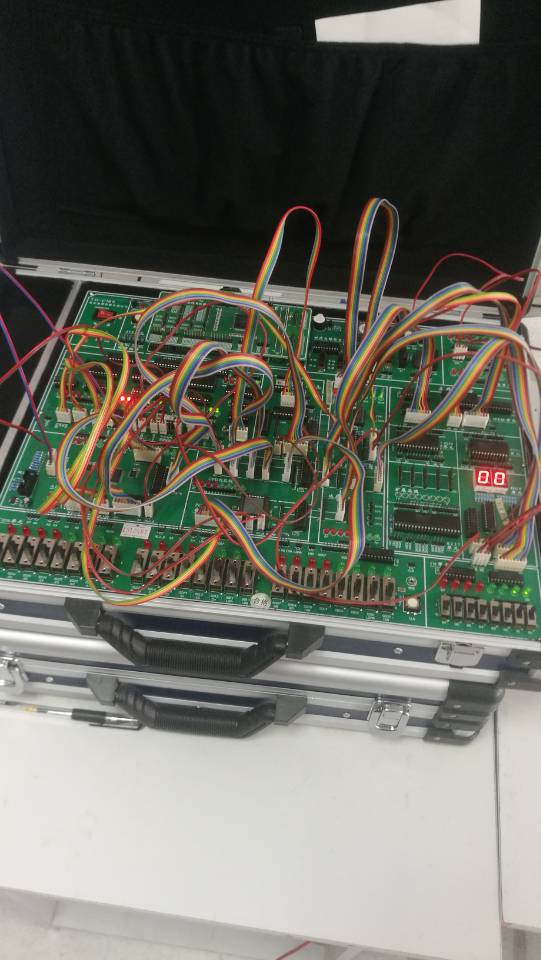
1). 从IN单元读入一个数据，存于R0，将R0和自身相加，结果存于R0，再将R0的值送OUT单元显示、停机。

2). 2+3=

3). 7-5=

1. 实验结果

连线：



两数相加代码：

; //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* //

; // //

; // CPU 与简单模型机实验指令文件 //

; // //

; // By TangDu CO.,LTD //

; // //

; //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* //

; //\*\*\*\*\*\* Start Of Main Memory Data \*\*\*\*\*\* //

$P 00 20 ;  START: IN R0 从 IN 单元读入数据送 R0

$P 01 00 ;  ADD R0,R0 R0 和自身相加，结果送 R0

$P 02 30 ;  OUT R0 R0 的值送 OUT 单元显示

$P 03 E0 ;  JMP START 跳转至 00H 地址

$P 04 00 ;

$P 05 50 ;  HLT 停机

; //\*\*\*\*\*\*\* End Of Main Memory Data \*\*\*\*\*\*\* //

; //\*\*\*\* Start Of MicroController Data \*\*\*\* //

$M 00 000001 ;  NOP

$M 01 006D43 ;  PC->AR,PC 加 1

$M 03 107070 ;  MEM->IR, P<1>

$M 04 182005 ;  IN->B

$M 05 04B201 ;  A 加 B->R0

$M 1D 105141 ;  MEM->PC

$M 30 001404 ;  R0->A

$M 32 183001 ;  IN->R0

$M 33 280401 ;  R0->OUT

$M 35 000035 ;  NOP

$M 3C 006D5D ;  PC->AR,PC 加 1

//\*\* End Of MicroController Data \*\*//

两数相减代码：

; //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* //

; // //

; // CPU 与简单模型机实验指令文件 //

; // //

; // By TangDu CO.,LTD //

; // //

; //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* //

; //\*\*\*\*\*\* Start Of Main Memory Data \*\*\*\*\*\* //

$P 00 20 ;  START: IN R0 从 IN 单元读入数据送 R0

$P 01 00 ;  ADD R0,R0 R0 和自身相加，结果送 R0

$P 02 30 ;  OUT R0 R0 的值送 OUT 单元显示

$P 03 E0 ;  JMP START 跳转至 00H 地址

$P 04 00 ;

$P 05 50 ;  HLT 停机

; //\*\*\*\*\*\*\* End Of Main Memory Data \*\*\*\*\*\*\* //

; //\*\*\*\* Start Of MicroController Data \*\*\*\* //

$M 00 000001 ;  NOP

$M 01 006D43 ;  PC->AR,PC 加 1

$M 03 107070 ;  MEM->IR, P<1>

$M 04 182005 ;  IN->B

$M 05 05B201 ;  A 减 B->R0

$M 1D 105141 ;  MEM->PC

$M 30 001404 ;  R0->A

$M 32 183001 ;  IN->R0

$M 33 280401 ;  R0->OUT

$M 35 000035 ;  NOP

$M 3C 006D5D ;  PC->AR,PC 加 1

//\*\* End Of MicroController Data \*\*//

1. 实验总结

通过本次实验，我掌握了一个简单CPU的组成原理，进一步将其构造一台基本模型计算机，进一步了解了CPU的工作原理，这次实验和上次实验过程差别不大，主要是设计微程序的这个过程耗时较多，同时收获也更多。