兰州大学信息科学与工程学院实验报告

实验成绩：

学生姓名：               杨添宝

学　　号：         320170941671

年级专业：  2017级计算机基地班

指导老师：               赵继平

实验课程：         计算机组成原理实验

实验题目：           综合实验的调试 1

一、实验目的

（1）通过使用软件LCACPT，了解程序编译、加载的过程。

（2）通过微单步、单拍调试，理解模型机中的数据流向。

二、实验连线

各模块控制信号连接表：（或者使用提供的连接板）

（1）总线和内存单元：

|  |  |
| --- | --- |
| BUS | M21 |
| RM | M21 |
| WM | M22 |
| EMCK | PLS4 |
| DJ1 | ALU-IN |
| DJ2 | ALU-OUT |
| DJ3 | R-IN |
| DJ4 | R-OUT |
| DJ5 | RA-IN |
| DJ6 | RA-OUT |
| DJ7 | PC-IN |
| AJ1 | PC-OUT |

（2）微程序控制单元：

|  |  |
| --- | --- |
| MLD | M23 |
| MCK | PLS1 |
| MOCK | PLS3 |
| MD0 | VCC |
| MD1 | VCC |
| MD2 | I4 |
| MD3 | I5 |
| MD4 | I6 |
| MD5 | I7 |
| MD6 | GND |
| MD7 | GND |

（3）寄存器组单元：

|  |  |
| --- | --- |
| SA | I0 |
| SB | I1 |
| RR | M15 |
| WR | M14 |
| RCK | PLS4 |
| X0 | M12 |
| X1 | M11 |
| ERA | M10 |
| RA-O | M9 |
| RACK | PLS4 |

（4）算术逻辑单元：

|  |  |
| --- | --- |
| EDR1 | M8 |
| EDR2 | M7 |
| ALU-O | M6 |
| CN | M5 |
| M | M4 |
| S3 | M3 |
| S2 | M2 |
| S1 | M1 |
| S0 | M0 |
| D2CK | PLS4 |
| D1CK | PLS4 |
| CCK | PLS4 |
| ZD | JZ |
| CY | JC |

（5）指令寄存器：

|  |  |
| --- | --- |
| EIR1 | M20 |
| EIR2 | M19 |
| IR1CK | PLS4 |
| IR2CK | PLS4 |
| IR2-O | M18 |
| PC-O | M17 |
| ELP | M16 |
| PCCK | PLS2 |
| JZ | ZD |
| JC | CY |
| JS0 | I2 |
| JS1 | I3 |

（6）启停单元：

|  |  |
| --- | --- |
| HCK | PLS2 |
| HALT | M13 |

三、指令系统

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 指令助记符 | 指令功能 | 指令编码 | 微周期 | 微操作 |
| 取指微指令 |  |  | T0： | PC->地址总线->RAM RAM->数据总线->IR1 |
| ADD A,R0  ADD A,R1  ADD A,R2  ADD A,R3 | (A)+(Ri)->A | 0C  0D  0E  0F | T0：  T1：  T2：  T3： | A->数据总线->DR1  Ri->数据总线->DR2  ALU->数据总线->A、置CY  取指微指令 |
| SUB A,R0  SUB A,R1  SUB A,R2  SUB A,R3 | (A)-(Ri)->A | 1C  1D  1E  1F | T0：  T1：  T2：  T3： | A->数据总线->DR1  Ri->数据总线->DR2  ALU->数据总线->A、置CY  取指微指令 |
| MOV A,@R0  MOV A,@R1  MOV A,@R2  MOV A,@R3 | (Ri)->A | 2C  2D  2E  2F | T0：  T1：  T2： | Ri->数据总线->IR2  IR2->地址总线->RAM->A  取指微指令 |
| MOV A,R0  MOV A,R1  MOV A,R2  MOV A,R3 | (Ri)->A | 3C  3D  3E  3F | T0：  T1： | Ri->数据总线->A  取指微指令 |
| MOV R0,A  MOV R1,A  MOV R2,A  MOV R3,A | (A)->Ri | 4C  4D  4E  4F | T0：  T1： | A->数据总线->Ri  取指微指令 |
| MOV A,#data | Data->A | 5F | T0：  T1： | RAM->数据总线->A  取指微指令 |
| MOV R0,#data  MOV R1,#data  MOV R2,#data  MOV R3,#data | Data->Ri | 6C  6D  6E  6F | T0：  T1： | RAM->数据总线->A  取指微指令 |
| LDA addr | (addr)->A | 7F | T0：  T1：  T2： | RAM->数据总线->IR2  IR2->地址总线，RAM->A  取指微指令 |
| STA addr | (A)->addr | 8F | T0：  T1：  T2： | RAM->数据总线->IR2  IR2->地址总线，A->RAM  取指微指令 |
| RLC A | C、A左移1位 | 9F | T0：  T1： | A<<1、置CY  取指微指令 |
| RRC A | C、A右移1位 | AF | T0：  T1： | A>>1、置CY  取指微指令 |
| JZ addr | A=0 ，Addr->PC | B3 | T0：  T1： | 条件成立：RAM->PC  取指微指令 |
| JC addr | Cy=0，Addr->PC | B7 | T0：  T1： | 条件成立：RAM->PC  取指微指令 |
| JMP addr | Addr->PC | BF | T0：  T1： | RAM->PC  取指微指令 |
| ORL A,#data | (A)或data->A | CF | T0：  T1：  T2：  T3： | A->数据总线->DR1  RAM->数据总线->DR2  ALU->数据总线->A  取指微指令 |
| ANL A,#data | (A) 与data->A | DF | T0：  T1：  T2：  T3： | A->数据总线->DR1  RAM->数据总线->DR2  ALU->数据总线->A  取指微指令 |
| HALT | 停机 | FF | T0： | 停机 |

四、微指令表

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 指令助记符 |  | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |
|  | 微地址 | MLD | WM | RM | EIR1 | EIR2 | IR2-O | PC-O | ELP | RR | WR | HALT | X0 | X1 | ERA | RA-O | EDR1 | EDR2 | ALU-O | CN | M | S3 | S2 | S1 | S0 | 16进制 |
|  | 有效值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | \* | \* | 0 | 0 | 0 | 0 | 0 | \* | \* | \* | \* | \* | \* |  |
|  | 00H | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
| 取指微指令 | 01H |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 02H |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| ADD A,RI | 03H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | FFFCF9 |
|  | 04H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | FF7F79 |
|  | 05H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | FFFBA9 |
|  | 06H | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
| SUB A,RI | 07H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | FFFCD6 |
|  | 08H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | FE7F56 |
|  | 09H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | FFFB86 |
|  | 0AH | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
| MOV A,@RI | 0BH | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | F77FFF |
|  | 0CH | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | DBFBFF |
|  | 0DH | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
|  | 0EH |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| MOV A,RI | 0FH | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | FF7BFF |
|  | 10H | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
|  | 11H |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 12H |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| MOV RI,A | 13H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7FBDFF |
|  | 14H | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
|  | 15H |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 16H |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 指令助记符 |  | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |
|  | 微地址 | MLD | WM | RM | EIR1 | EIR2 | IR2-O | PC-O | ELP | RR | WR | HALT | X0 | X1 | ERA | RA-O | EDR1 | EDR2 | ALU-O | CN | M | S3 | S2 | S1 | S0 | 16进制 |
|  | 有效值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | \* | \* | 0 | 0 | 0 | 0 | 0 | \* | \* | \* | \* | \* | \* |  |
| MOV A,#data | 17H | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | DDFBFF |
|  | 18H | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
|  | 19H |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 1AH |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| MOV Ri,#data | 1BH | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | DDBFFF |
|  | 1CH | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
|  | 1DH |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 1EH |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| LDA A,addr | 1FH | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | D5FFFF |
|  | 20H | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | DBFBFF |
|  | 21H | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
|  | 22H |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| STA addr | 23H | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | D5FFFF |
|  | 24H | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | BBFDFF |
|  | 25H | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
|  | 26H |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| RRC | 27H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | FFF1EF |
|  | 28H | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
|  | 29H |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 2AH |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| RLC | 2BH | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | FFE9EF |
|  | 2CH | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | CDFFFF |
|  | 2DH |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 2EH |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 指令助记符 |  | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |
|  | 微地址 | MLD | WM | RM | EIR1 | EIR2 | IR2-O | PC-O | ELP | RR | WR | HALT | X0 | X1 | ERA | RA-O | EDR1 | EDR2 | ALU-O | CN | M | S3 | S2 | S1 | S0 | 16进制 |
|  | 有效值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | \* | \* | 0 | 0 | 0 | 0 | 0 | \* | \* | \* | \* | \* | \* |  |
| JZ addr | 2FH | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | D4FFFF |
| JC addr | 30H | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
| JMP addr | 31H |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 32H |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| ORL A,#data | 33H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | FFFCFE |
|  | 34H | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | DDFF7E |
|  | 35H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | FFFBBE |
|  | 36H | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
| ANL A,#data | 37H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | FFFCFB |
|  | 38H | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | DDFF7B |
|  | 39H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | FFFBBB |
|  | 3AH | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4DFFFF |
|  | 3BH |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 3CH |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 3DH |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 3EH |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| HALT | 3FH | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | FFDFFF |

五、程序调试

实验、实现普通的加、减法指令

在软件LCACPT中，输入以下程序，并且编译、加载到实验平台中。

MOV A,#55

RAM->A

MOV R0,#66

RAM->R0

ADD A,R0

(A)+(R0)->A

MOV R1#33

RAM->R1

SUB A,R1

(A)-(R1)->A

STA 10

HALT

A->RAM

停机

或通过键盘（键盘使用方法见第四章）输入微程序及以下程序：

|  |  |  |  |
| --- | --- | --- | --- |
| 内存地址 | 指令助记符 | 指令码 | 说明 |
| 00H | MOV A,#55 | 5FH | 立即数55H->A |
| 01H |  | 55H |  |
| 02H | MOV R0,#66 | 6CH | 立即数66H->寄存器R0 |
| 03H |  | 66H |  |
| 04H | ADD A,R0 | 0CH | A内容+R0内容->A |
| 05H | MOV R1,#33 | 6DH | 立即数33H->寄存器R1 |
| 06H |  | 33H |  |
| 07H | SUB A,R1 | 1DH | A内容-R1内容->A |
| 08H | STA 10 | 8FH | 将A内容写入RAM地址10H |
| 09H |  | 10H |  |
| 0AH | HALT | FFH | 停机 |

运行结果为：RAM 10H单元中的内容为88H

运行程序：

通过软件LCACPT的微单步功能可观察各个变量的变化。

微单步运行过程显示如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 微地址 | 数据流程 | 数据总线 | 地址总线 | 操作寄存器 |
| 00H | 取指微指令 RAM->BUS->IR1 | 5FH | 00H | IR1=5FH |
| MOV A,#55 | |  |  |  |
| 17H | BUS-> A | 55H | 01H | A=55H |
| 18H | 取指微指令 RAM->BUS->IR1 | 6CH | 02H | IR1=6CH |
| MOV R0,#66 | |  |  |  |
| 1BH | RAM->寄存器R0 | 66H | 03H | 寄存器R0=66H |
| 1CH | 取指微指令 RAM->BUS->IR1 | 0CH | 04H | IR1=0CH |
| ADD A,R0 | |  |  |  |
| 03H | A->锁存器DR1 | 55H | 无效 | DR1=55H |
| 04H | 寄存器R0->锁存器DR2 | 66H | 无效 | DR2=66H |
| 05H | ALU-> A | BBH | 无效 | A=BBH |
| 06H | 取指微指令 RAM->BUS->IR1 | 6DH | 05H | IR1=6DH |
| MOV R1,#33 | |  |  |  |
| 1BH | RAM->寄存器R1 | 33H | 06H | 寄存器R1=33 |
| 1CH | 取指微指令 RAM->BUS->IR1 | 1DH | 07H | IR1=1DH |
| SUB A,R1 | |  |  |  |
| 07H | A->锁存器DR1 | BBH | 无效 | DR1=BBH |
| 08H | 寄存器R1->锁存器DR2 | 33H | 无效 | DR2=33H |
| 09H | ALU->A | 88H | 无效 | A=88H |
| 0AH | 取指微指令 RAM->BUS->IR1 | 8FH | 08H | IR1=8FH |
| STA 10 | |  |  |  |
| 23H | RAM->BUS->IR2 | 10H | 09H | IR2=10H |
| 24H | A->RAM(10H) | 88H | 10H | RAM(10)=88H |
| 25H | 取指微指令 RAM->BUS->IR1 | FFH | 0AH | IR1=FFH |
| HALT | |  |  |  |
| 3FH | 置模型机为停止状态 | 无效 | 无效 | 置停止状态 |

如果在运行微单步时，发现有错误或对微单步中的时序过程不清楚，可用时序单元中的按钮来手动给出4个节拍。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 微周期 | 数据流程 | 节拍 | 数据总线 | 地址总线 | 操作寄存器 |
| T0： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址清零  PLS2：置模型机运行  PLS3：取指微指令输出  PLS4：BUS->IR1 | 无效  无效  5FH  5FH | 无效  无效  00H  00H | 微地址：00H  PC=00H  锁存微指令  IR=5FH |
| MOV A,#55 | | | | | |
| T0： | BUS->A | PLS1：置微地址  PLS2：PC+1  PLS3：微指令输出  PLS4：BUS->A | 5FH  5FH  55H  55H | 00H  00H  01H  01H | 微地址：17H  PC=01H  锁存微指令  A=55H |
| T1： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址+1  PLS2：PC+1  PLS3：取指微指令输出  PLS4：BUS->IR1 | 55H  55H  6CH  6CH | 01H  01H  02H  02H | 微地址：18H  PC=02H  锁存微指令  IR1=6CH |
| MOV R0,#66 | | | | | |
| T0： | RAM->寄存器R0 | PLS1：置微地址  PLS2：PC+1  PLS3：取指微指令输出  PLS4：BUS->寄存器R0 | 6CH  6CH  66H  66H | 02H  02H  03H  03H | 微地址：1BH  PC=03H  锁存微指令  寄存器R0=66H |
| T1： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址+1  PLS2：PC+1  PLS3：取指微指令输出  PLS4：BUS->IR1 | 66H  66H  0CH  0CH | 03H  03H  04H  04H | 微地址：1CH  PC=04H  锁存微指令  IR1=0CH |
| ADD A,R0 |  |  |  |  |  |
| T0： | A->锁存器DR1 | PLS1：置微地址  PLS2：PC+1  PLS3：取指微指令输出  PLS4：BUS->锁存器DR1 | 0CH  0CH  55H  55H | 04H  04H  无效  无效 | 微地址：03H  PC=05H  锁存微指令  DR1=55H |
| T1： | 寄存器R0->锁存器DR2 | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：BUS->锁存器DR2 | 55H  55H  66H  66H | 无效  无效  无效  无效 | 微地址：04H  PC=05H  锁存微指令  DR2=66H |
| T2： | ALU->A | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：BUS->A | 66H  66H  BBH  BBH | 无效  无效  无效  无效 | 微地址：05H  PC=05H  锁存微指令  A=BBH |
| T3： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：BUS->IR1 | BBH  BBH  6DH  6DH | 无效  无效  05H  05H | 微地址：06H  PC=05H  锁存微指令  IR1=6DH |
| MOV R1,#33 | | | | | |
| T0： | RAM->寄存器R1 | PLS1：置微地址  PLS2：PC+1  PLS3：微指令输出  PLS4：BUS->IR1 | 6DH  6DH  33H  33H | 05H  05H  06H  06H | 微地址：1BH  PC=06H  锁存微指令  寄存器R1=33 |
| T1： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址+1  PLS2：PC+1  PLS3：微指令输出  PLS4：BUS->IR1 | 33H  33H  1DH  1DH | 06H  06H  07H  07H | 微地址：1CH  PC=07H  锁存微指令  IR1=1DH |
| SUB A,Ri |  |  |  |  |  |
| T0： | A->锁存器DR1 | PLS1：置微地址  PLS2：PC+1  PLS3：微指令输出  PLS4：BUS->DR1 | 1DH  1DH  BBH  BBH | 07H  07H  无效  无效 | 微地址：07H  PC=08H  锁存微指令  DR1=BBAH |
| T1： | 寄存器R1->锁存器DR2 | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：BUS->DR2 | BBH  BBH  33H  33H | 无效  无效  无效  无效 | 微地址：08H  PC=08H  锁存微指令  DR2=33H |
| T2： | ALU->A | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：ALU->A | 33H  33H  88H  88H | 无效  无效  无效  无效 | 微地址：09H  PC=08H  锁存微指令  A=88H |
| T3： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：BUS->IR1 | 88H  88H  8FH  8FH | 无效  无效  08H  08H | 微地址：0AH  PC=08H  锁存微指令  IR1=8FH |
| STA 10 |  |  |  |  |  |
| T0： | RAM->BUS->IR2 | PLS1：置微地址  PLS2：PC+1  PLS3：微指令输出  PLS4：BUS->IR2 | 8FH  8FH  10H  10H | 08H  08H  09H  09H | 微地址：23H  PC=09H  锁存微指令  IR2=10H |
| T1： | A->RAM(10H) | PLS1：微地址+1  PLS2：PC+1  PLS3：微指令输出  PLS4：BUS->RAM | 10H  10H  88H  88H | 09H  09H  10H  10H | 微地址：24H  PC=0AH  锁存微指令  RAM(10)=88H |
| T2： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：BUS->IR1 | 88H  88H  FFH  FFH | 10H  10H  0AH  0AH | 微地址：25H  PC=0AH  锁存微指令  IR1=FFH |
| HALT | | | | | |
| T0： | 置模型机为停止状态 | PLS1：置微地址  PLS2：停机  PLS3：微指令输出  PLS4：BUS->IR2 | FFH  无效 | 0AH  无效 | 微地址：3FH  PC=0BH |

六、实验思考

**1．为什么指令5FH置入口微地址为17H。**

指令5FH对应指令助记符MOV A,#data，查微指令表可知它的微地址有效值为17H，查阅下表可得微程序首地址：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 按操作码散转 | | | | | | |
| 指令操作码 | | | | | | 微程序首地址 |
| MD7、MD6 | I7 | I6 | I5 | I4 | MD1、MD0 | MD7～MD0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 003H |
| 0 | 0 | 0 | 0 | 1 | 1 | 007H |
| 0 | 0 | 0 | 1 | 0 | 1 | 00BH |
| 0 | 0 | 0 | 1 | 1 | 1 | 00FH |
| 0 | 0 | 1 | 0 | 0 | 1 | 013H |
| 0 | 0 | 1 | 0 | 1 | 1 | 017H |
| 0 | 0 | 1 | 1 | 0 | 1 | 01BH |
| 0 | 0 | 1 | 1 | 1 | 1 | 01FH |
| 0 | 1 | 0 | 0 | 0 | 1 | 023H |
| 0 | 1 | 0 | 0 | 1 | 1 | 027H |
| 0 | 1 | 0 | 1 | 0 | 1 | 02BH |
| 0 | 1 | 0 | 1 | 1 | 1 | 02FH |
| 0 | 1 | 1 | 0 | 0 | 1 | 033H |
| 0 | 1 | 1 | 0 | 1 | 1 | 037H |
| 0 | 1 | 1 | 1 | 0 | 1 | 03BH |
| 0 | 1 | 1 | 1 | 1 | 1 | 03FH |

**2．写出对应的微程序并通过键盘输入，记录运行结果。**

微程序：

MOV A,#55

RAM->A

MOV R0,#66

RAM->R0

ADD A,R0

(A)+(R0)->A

MOV R1#33

RAM->R1

SUB A,R1

(A)-(R1)->A

STA 10

HALT

A->RAM

停机

运行结果：

|  |  |  |  |
| --- | --- | --- | --- |
| 内存地址 | 指令助记符 | 指令码 | 说明 |
| 00H | MOV A,#55 | 5FH | 立即数55H->A |
| 01H |  | 55H |  |
| 02H | MOV R0,#66 | 6CH | 立即数66H->寄存器R0 |
| 03H |  | 66H |  |
| 04H | ADD A,R0 | 0CH | A内容+R0内容->A |
| 05H | MOV R1,#33 | 6DH | 立即数33H->寄存器R1 |
| 06H |  | 33H |  |
| 07H | SUB A,R1 | 1DH | A内容-R1内容->A |
| 08H | STA 10 | 8FH | 将A内容写入RAM地址10H |
| 09H |  | 10H |  |
| 0AH | HALT | FFH | 停机 |

**3．记录微单步运行过程。**

微单步运行过程显示如下：

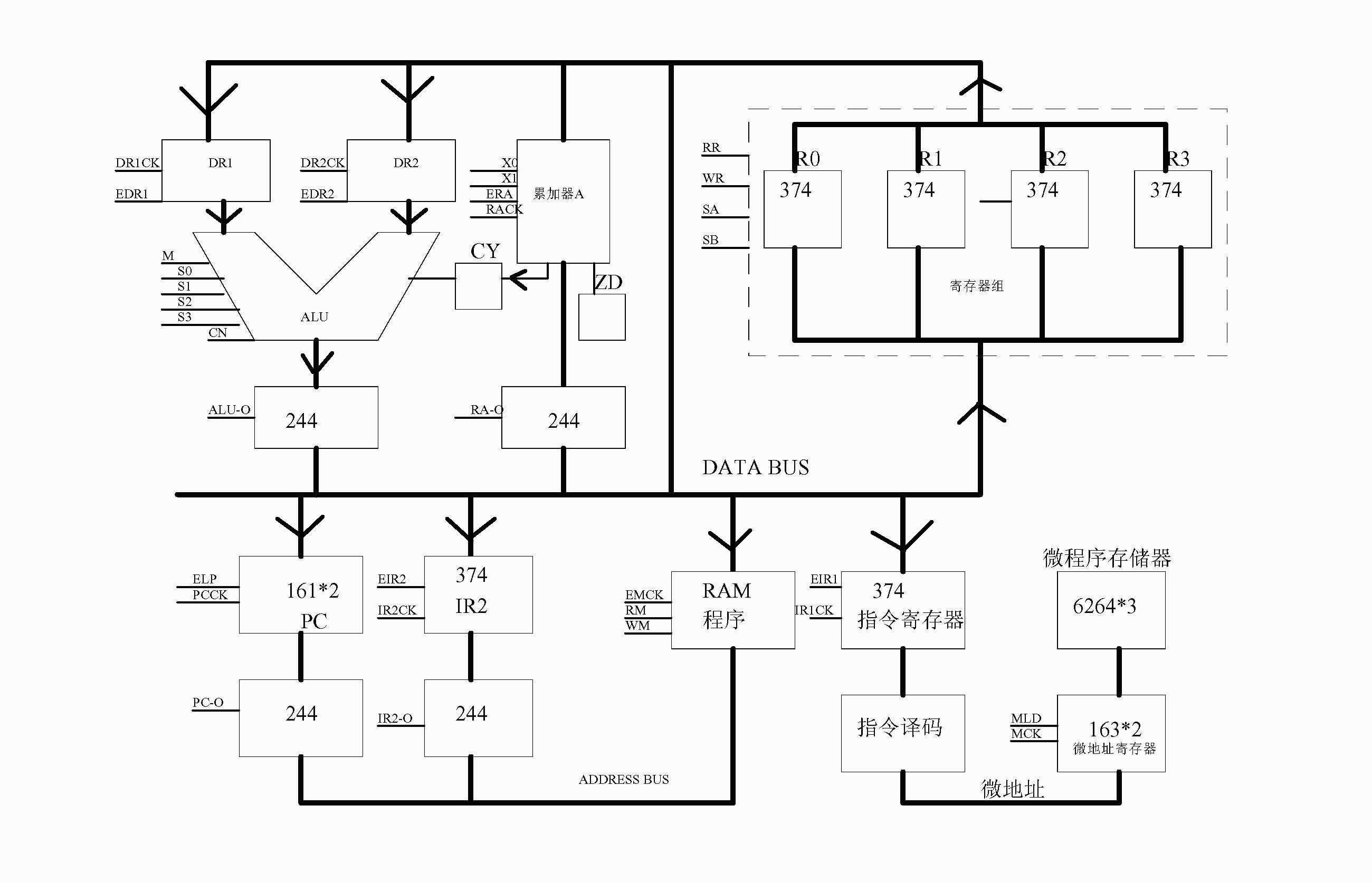
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 微地址 | 数据流程 | 数据总线 | 地址总线 | 操作寄存器 |
| 00H | 取指微指令 RAM->BUS->IR1 | 5FH | 00H | IR1=5FH |
| MOV A,#55 | |  |  |  |
| 17H | BUS-> A | 55H | 01H | A=55H |
| 18H | 取指微指令 RAM->BUS->IR1 | 6CH | 02H | IR1=6CH |
| MOV R0,#66 | |  |  |  |
| 1BH | RAM->寄存器R0 | 66H | 03H | 寄存器R0=66H |
| 1CH | 取指微指令 RAM->BUS->IR1 | 0CH | 04H | IR1=0CH |
| ADD A,R0 | |  |  |  |
| 03H | A->锁存器DR1 | 55H | 无效 | DR1=55H |
| 04H | 寄存器R0->锁存器DR2 | 66H | 无效 | DR2=66H |
| 05H | ALU-> A | BBH | 无效 | A=BBH |
| 06H | 取指微指令 RAM->BUS->IR1 | 6DH | 05H | IR1=6DH |
| MOV R1,#33 | |  |  |  |
| 1BH | RAM->寄存器R1 | 33H | 06H | 寄存器R1=33 |
| 1CH | 取指微指令 RAM->BUS->IR1 | 1DH | 07H | IR1=1DH |
| SUB A,R1 | |  |  |  |
| 07H | A->锁存器DR1 | BBH | 无效 | DR1=BBH |
| 08H | 寄存器R1->锁存器DR2 | 33H | 无效 | DR2=33H |
| 09H | ALU->A | 88H | 无效 | A=88H |
| 0AH | 取指微指令 RAM->BUS->IR1 | 8FH | 08H | IR1=8FH |
| STA 10 | |  |  |  |
| 23H | RAM->BUS->IR2 | 10H | 09H | IR2=10H |
| 24H | A->RAM(10H) | 88H | 10H | RAM(10)=88H |
| 25H | 取指微指令 RAM->BUS->IR1 | FFH | 0AH | IR1=FFH |
| HALT | |  |  |  |
| 3FH | 置模型机为停止状态 | 无效 | 无效 | 置停止状态 |

**4．手动控制4个节拍，记录运行过程。**

用时序单元中的按钮来手动给出4个节拍。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 微周期 | 数据流程 | 节拍 | 数据总线 | 地址总线 | 操作寄存器 |
| T0： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址清零  PLS2：置模型机运行  PLS3：取指微指令输出  PLS4：BUS->IR1 | 无效  无效  5FH  5FH | 无效  无效  00H  00H | 微地址：00H  PC=00H  锁存微指令  IR=5FH |
| MOV A,#55 | | | | | |
| T0： | BUS->A | PLS1：置微地址  PLS2：PC+1  PLS3：微指令输出  PLS4：BUS->A | 5FH  5FH  55H  55H | 00H  00H  01H  01H | 微地址：17H  PC=01H  锁存微指令  A=55H |
| T1： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址+1  PLS2：PC+1  PLS3：取指微指令输出  PLS4：BUS->IR1 | 55H  55H  6CH  6CH | 01H  01H  02H  02H | 微地址：18H  PC=02H  锁存微指令  IR1=6CH |
| MOV R0,#66 | | | | | |
| T0： | RAM->寄存器R0 | PLS1：置微地址  PLS2：PC+1  PLS3：取指微指令输出  PLS4：BUS->寄存器R0 | 6CH  6CH  66H  66H | 02H  02H  03H  03H | 微地址：1BH  PC=03H  锁存微指令  寄存器R0=66H |
| T1： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址+1  PLS2：PC+1  PLS3：取指微指令输出  PLS4：BUS->IR1 | 66H  66H  0CH  0CH | 03H  03H  04H  04H | 微地址：1CH  PC=04H  锁存微指令  IR1=0CH |
| ADD A,R0 |  |  |  |  |  |
| T0： | A->锁存器DR1 | PLS1：置微地址  PLS2：PC+1  PLS3：取指微指令输出  PLS4：BUS->锁存器DR1 | 0CH  0CH  55H  55H | 04H  04H  无效  无效 | 微地址：03H  PC=05H  锁存微指令  DR1=55H |
| T1： | 寄存器R0->锁存器DR2 | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：BUS->锁存器DR2 | 55H  55H  66H  66H | 无效  无效  无效  无效 | 微地址：04H  PC=05H  锁存微指令  DR2=66H |
| T2： | ALU->A | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：BUS->A | 66H  66H  BBH  BBH | 无效  无效  无效  无效 | 微地址：05H  PC=05H  锁存微指令  A=BBH |
| T3： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：BUS->IR1 | BBH  BBH  6DH  6DH | 无效  无效  05H  05H | 微地址：06H  PC=05H  锁存微指令  IR1=6DH |
| MOV R1,#33 | | | | | |
| T0： | RAM->寄存器R1 | PLS1：置微地址  PLS2：PC+1  PLS3：微指令输出  PLS4：BUS->IR1 | 6DH  6DH  33H  33H | 05H  05H  06H  06H | 微地址：1BH  PC=06H  锁存微指令  寄存器R1=33 |
| T1： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址+1  PLS2：PC+1  PLS3：微指令输出  PLS4：BUS->IR1 | 33H  33H  1DH  1DH | 06H  06H  07H  07H | 微地址：1CH  PC=07H  锁存微指令  IR1=1DH |
| SUB A,Ri |  |  |  |  |  |
| T0： | A->锁存器DR1 | PLS1：置微地址  PLS2：PC+1  PLS3：微指令输出  PLS4：BUS->DR1 | 1DH  1DH  BBH  BBH | 07H  07H  无效  无效 | 微地址：07H  PC=08H  锁存微指令  DR1=BBAH |
| T1： | 寄存器R1->锁存器DR2 | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：BUS->DR2 | BBH  BBH  33H  33H | 无效  无效  无效  无效 | 微地址：08H  PC=08H  锁存微指令  DR2=33H |
| T2： | ALU->A | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：ALU->A | 33H  33H  88H  88H | 无效  无效  无效  无效 | 微地址：09H  PC=08H  锁存微指令  A=88H |
| T3： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：BUS->IR1 | 88H  88H  8FH  8FH | 无效  无效  08H  08H | 微地址：0AH  PC=08H  锁存微指令  IR1=8FH |
| STA 10 |  |  |  |  |  |
| T0： | RAM->BUS->IR2 | PLS1：置微地址  PLS2：PC+1  PLS3：微指令输出  PLS4：BUS->IR2 | 8FH  8FH  10H  10H | 08H  08H  09H  09H | 微地址：23H  PC=09H  锁存微指令  IR2=10H |
| T1： | A->RAM(10H) | PLS1：微地址+1  PLS2：PC+1  PLS3：微指令输出  PLS4：BUS->RAM | 10H  10H  88H  88H | 09H  09H  10H  10H | 微地址：24H  PC=0AH  锁存微指令  RAM(10)=88H |
| T2： | 取指微指令  RAM->BUS->IR1 | PLS1：微地址+1  PLS2：PC不变  PLS3：微指令输出  PLS4：BUS->IR1 | 88H  88H  FFH  FFH | 10H  10H  0AH  0AH | 微地址：25H  PC=0AH  锁存微指令  IR1=FFH |
| HALT | | | | | |
| T0： | 置模型机为停止状态 | PLS1：置微地址  PLS2：停机  PLS3：微指令输出  PLS4：BUS->IR2 | FFH  无效 | 0AH  无效 | 微地址：3FH  PC=0BH |

**5．在模型机逻辑框图中标示正确的数据流并描述。**



数据输入

运算结果

从数据总线获取数据

产生下一条微指令地址

将A写入RAM地址10H

进位

判0

**6．如何判断使用的指令寄存器是IR1还是IR2的值。**

可将指令寄存器IR1或IR2的值输出，和输入的指令做对比，使用的指令寄存器即为和输入的指令相同的那一个。

**7．总结微地址、PC地址、地址总线、数据总线、IR1、IR2、DR1、DR2、A等值的观察记录方式。**

微地址、IR1、IR2、DR1、DR2、A的值可以通过键盘和显示屏的操作进行观察记录，而PC地址、地址总线、数据总线的值则通过观察LED指示灯来记录。

**8．分析MD2、MD3、MD4、MD5与I4、I5、I6、I7互连的作用。**

在模型机中，用指令操作码的高4位作为核心扩展成8位的微程序入口地址 MD0~MD7，这种方法称为“按操作码散转”。MD2、MD3、MD4、MD5与I4、I5、I6、I7互连，通过I7-I4进行散转，可以产生下一条指令的微程序入口地址。