

计算机组成原理

实验报告

学号：57119101 姓名：王晨阳

学号：57119102 姓名：庄严

二0二0 年十二月

**实验四 数据通路的组织**

**一、实验内容**

* 设计一个单总线结构的数据通路，支持Demo\_IS指令系统的取数(LD)、减法 (SUB)、双字长分支 (JNZ)指令。
* 将测试程序存入主存，根据程序执行过程的OPCmd微操作命令）序列，控制所设计数据通路，来验证数据通路的正确性。

**二、电路设计与实现**

**1、指令功能分析**

本实验要求支持Demo\_IS指令系统中的取数(LD)、减法(SUB)、双字长分支(JNZ)指令，其功能分别为：RD←M[(RS)]、RD←(RD)－(RS)、ZF＝0时PC←Addr。其中，RD、RS表示寄存器编号，(Rx)、M[(Rx)]表示寄存器、存储单元的内容，Addr为直接寻址方式的地址码，ZF为上一条指令所产生的结果状态标志。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Demo\_IS Instructions | | | | | |
| Code | Byte 1 | | | Byte 2 | Specification |
| op | rd | rs | immediate/address |
| 7..4 | 3..2 | 1..0 | 7..0 |
| MOV | 0000 | rd | 00 | imme | rd ← imme |
| LD | 0010 | rd | rs | - | rd ← M[(rs)] |
| ST | 0011 | rd | rs | - | M[(rs)] ← (rd) |
| ADD | 0100 | rd | rs | - | rd ← (rd) + (rs) |
| 0101 | rd | rs | - | rd ← (rd) + M[(rs)] |
| SUB | 0110 | rd | rs | - | rd ← (rd) - (rs) |
| INC | 1000 | rd | 00 | - | rd ← (rd) + 1 |
| DEC | 1001 | rd | 00 | - | rd ← (rd) - 1 |
| JNZ | 1100 | 00 | 00 | addr | if(!ZF) PC←addr |
| 1101 | disp | | - | if(!ZF) PC←(PC)+disp |

由Demo\_IS的指令格式及上述3条指令功能约定，可得到如下分析结果：

* 数据类型只有8位整数一种，采用定点格式（补码编码）表示；
* 数据操作只有8位的加法、减法两种，需产生状态标志ZF；
* 数据寻址有寄存器寻址、寄存器间接寻址两种方式，地址无需计算；
* 指令寻址有直接寻址、隐含寻址两种方式，地址计算方法为8位加法；
* 寄存器有4个，长度为8位，每条指令最多2次读、1次写操作；
* 存储器按字节编址、地址空间为8位，每条指令最多有1次读/写操作。

基于数据通路，每个指令的功能可表示为一个微操作步序列，每个微操作步由一个或几个可同时实现的微操作组成。只要按序实现某微操作步序列中各微操作的功能，即可实现该指令的功能。

**2、数据通路的设计与实现**

数据通路由通路部件、部件互连两部分组成，互连结构有总线结构、点点结构两种类型。本实验的数据通路要求采用单总线结构。

**（1）功能部件设计**

为了满足Demo\_IS中3条指定指令的要求，数据通路的功能部件应包括ALU、寄存器组GPRs、状态寄存器PSR、存储器MEM、程序计数器PC、指令寄存器IR，以及地址寄存器MAR、数据寄存器MDR。

由Demo\_IS的分析结果可见，ALU应具有加法、减法功能，需产生状态标志ZF，可使用实验二设计的ALU来实现；GPRs应包含4个8位寄存器，具有1个读端口和1个写端口，可使用实验一设计的寄存器组来实现；MEM的容量应≤256×8位，可使用实验三设计的存储模块（容量为128×8位）来实现。为了简化控制，假设PC具有计数功能，可使用Quartus II提供的lpm\_counter模块来实现。

**（2）部件互连设计**

单总线结构的数据通路中，所有部件的数据入端、数据出端都连接在同一个总线上。为了保证数据传送的正确性，部件的出端需通过三态门连接到总线；部件的入端及出端中，只有1个可以直接连接总线，其余都需通过锁存器连接到总线。

本实验中，数据通路的宽度为8位，数据通路的组成如下图所示，TS0~TS3为三态门，GPRs的读地址引脚前增加选择器MUX，可以满足SUB指令中2次读GPRs的需求，当前读地址（RS或RD）用控制信号GRsel来选择。



图8 单总线结构的数据通路组成

图中的MEM需要存放测试程序，故要求使用实验三设计的存储模块（前64B为ROM空间）来实现，MEM片选引脚的连接也类似于实验三。由于实验三的存储模块读/写端口分离、未设置读操作控制信号、可在一个时钟周期内完成读写操作，故MDR由MDRI及MDRO实现，控制信号MemRd可以用来实现MDRO的写入控制（MDRinB）。

PSR用于存放结果状态标志，供CU使用，由于本实验的μOPCmd由手工给出，故判断ZF＝0也由人工完成，PSR可以省略（用虚线表示）。同理，ID也可省略。

**（3）数据通路的实现**

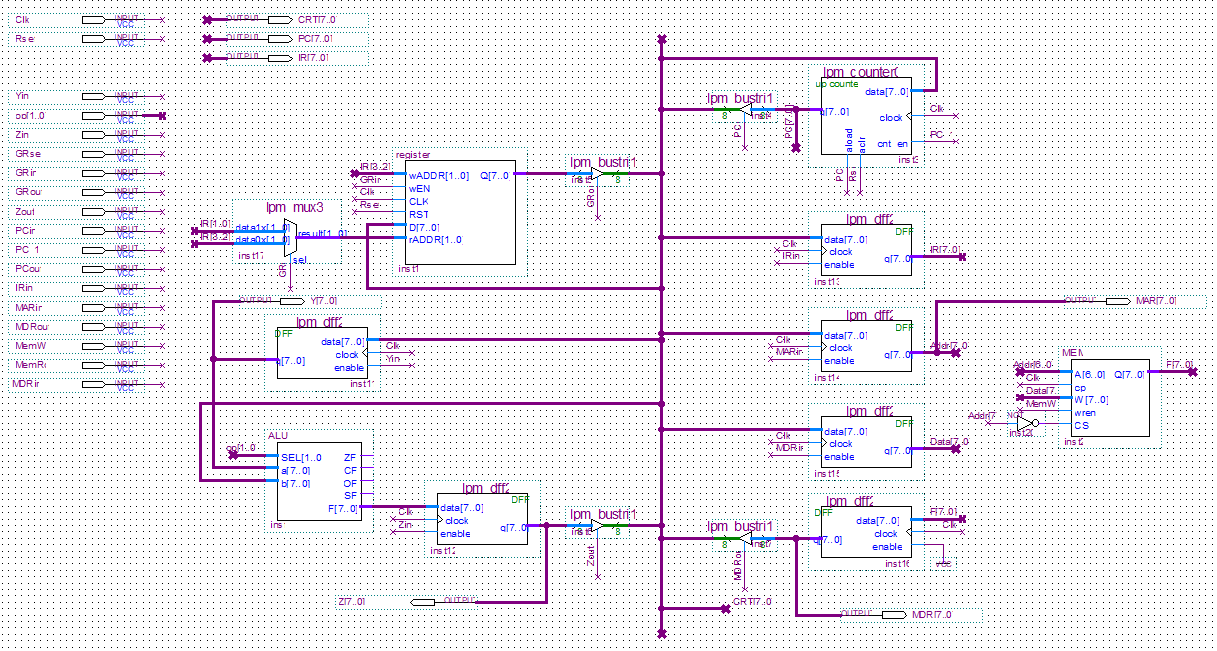
数据通路的实现需要编辑原理图文件（如DataPath.bdf），电路有多种实现方法，ALU、GPRs、MEM、PC都按照功能部件设计的方案来实现，IR、MAR、MDR、Y、Z应使用lpm\_dff实现，所有的部件控制信号都需设置为输入引脚。

数据传送μOP应采用电位-脉冲制来实现，即发送部件在时钟周期开始时就输出数据（如打开三态门），接收部件在时钟周期结束前接收数据并写入。如图9所示，μOPCmd由CU在时钟周期开始时（clock上升沿）通过电路产生（有一定时延[图9中①]），控制相应三态门打开（有一定时延[图9中②]），数据写入只能在时钟周期结束时（即下个clock上升沿[图9中④]）进行。

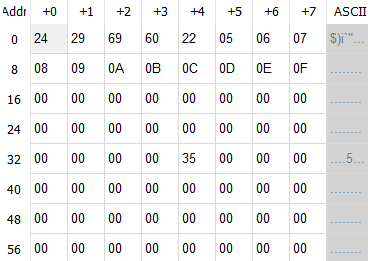
注意，MAR、MDRI、MDRO应在时钟周期结束时写入，由于图8中的MEM为同步RAM，故MEM操作应安排在时钟周期中部（clock下降沿）开始。

|  |  |  |  |
| --- | --- | --- | --- |
| 微操作 | 时钟周期 | OP | OPCmd |
| R1←M[(R0)] | t1 | MAR←(PC) | PCout、MARin |
| t2 | MDR←M[(MAR)],PC←(PC)＋1 | Read、WMFC，PC+1 |
| t3 | IR←(MDR) | MDRout、IRin |
| t4 | MAR←(R0) | GRout、Rsel、MARin |
| t5 | MDR←M[(MAR)] | Read、WMFC |
| t6 | R1←(MDR)，End←1 | MDRout、GRin，End |
| R2←M[(R1)] | t1 | MAR←(PC) | PCout、MARin |
| t2 | MDR←M[(MAR)],PC←(PC)＋1 | Read、WMFC，PC+1 |
| t3 | IR←(MDR) | MDRout、IRin |
| t4 | MAR←(R1) | GRout、Rsel、MARin |
| t5 | MDR←M[(MAR)] | Read、WMFC |
| t6 | R2←(MDR)，End←1 | MDRout、GRin，End |
| R2←(R2)－(R1) | t1 | MAR←(PC) | PCout、MARin |
| t2 | MDR←M[(MAR)],PC←(PC)＋1 | Read、WMFC，PC+1 |
| t3t | IR←(MDR) | MDRout、IRin |
| t4 | Y←(R2) | GRout、Yin ;Rsel=0 |
| t5 | Z←(Y)－(R1) | GRout、Rsel、op＝01、Zin |
| t6 | (R2)←(Z)，End←1 | Zout、GRin，End |
| JNZ 22H | t1 | MAR←(PC) | PCout、MARin |
| t2 | MDR←M[(MAR)],PC←(PC)＋1 | Read、WMFC，PC+1 |
| t3 | IR←(MDR) | MDRout、IRin |
| t4 | MAR←（PC） | PCout、MARin |
| t5 | MDR←M[(MAR)];PC←PC+1 | Read、WMFC |
| t6 | PC←(MDR);End←1 | MDRout、PCin、End |

电路图如下：



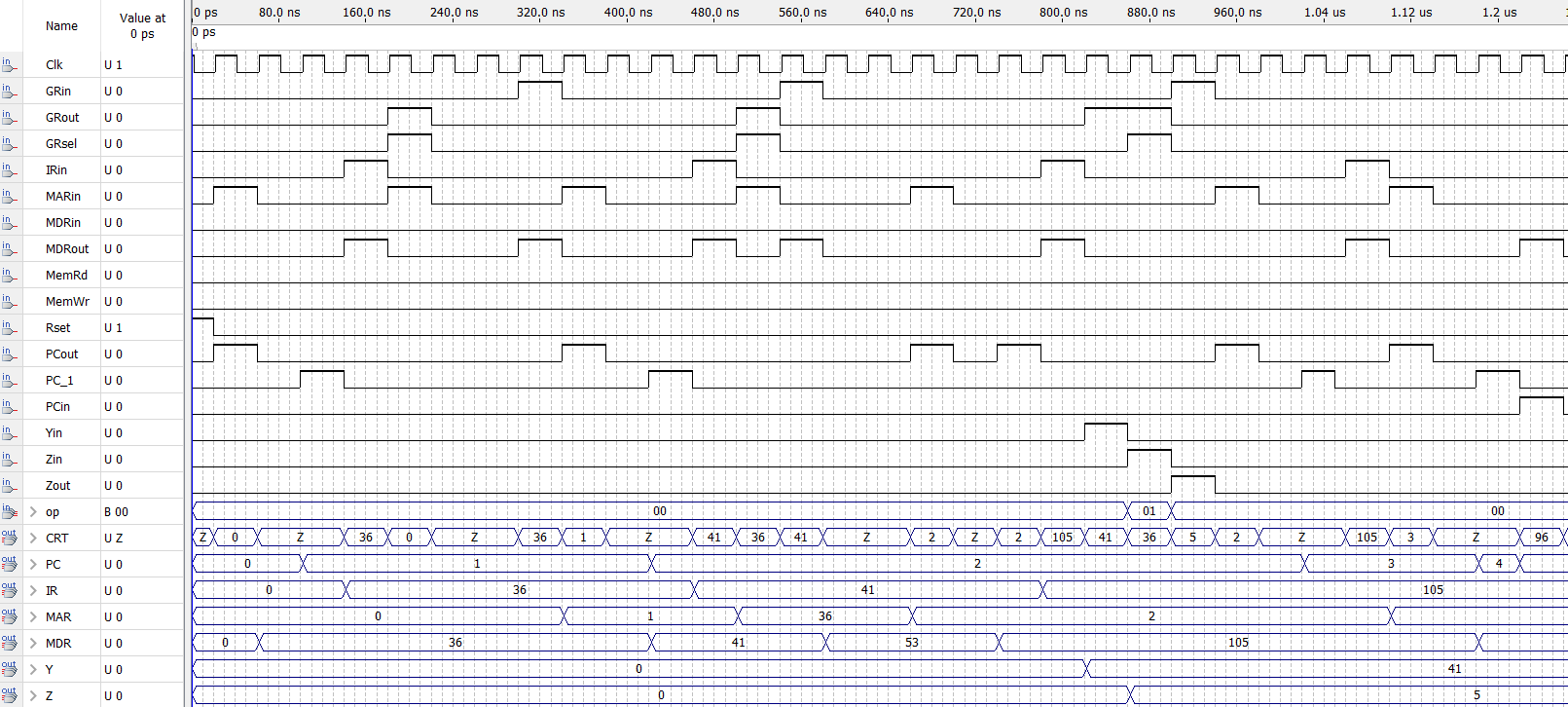
**图3 MEM电路图**



**图4 rom内容**

**三、电路正确性验证**

得到仿真结果如下所示



**图6 仿真波形图**

**四、实验小结**

通过这次试验，加深了对前三个实验的理解，将ALU运算器，寄存器，ram存储器组合应用；还掌握了CPU数据通路的逻辑组成，了解了指令功能的实现过程及其控制方法，练习了微操作序列的写法。

通过本次实验，小组成员巩固了所学习的计算机组成原理知识，对ALU的工作原理有了更加深刻的认识。小组成员也通过本次实验认识到了合作的重要性，锻炼了勇于面对困难与未知的精神。

**五、教师评语**

教师签字：

日期：