山东大学 计算机科学与技术 学院

计算机组成与设计 课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号：202200400053 | 姓名：王宇涵 | | 班级： 2202 |
| 实验题目：控制器实验 | | | |
| 实验学时：2 | | 实验日期: 2024-06-13 | |
| 实验目的：  掌握存储逻辑控制器的工作原理并设计运行 | | | |
| 实验软件和硬件环境：  软件环境：  QuartusII软件  硬件环境：  1.实验室台式机  2.计算机组成与设计实验箱 | | | |
| 实验原理和方法：  目前控制器设计大都采用微程序设计方法，又称存储逻辑控制器。微程序控制器电路结构如图13-1所示。它由控制存储器CROM、微程序PC计数器和微指令寄存器IR构成。  其中，微程序计数PC向控制存储器提供8位微地址，在控存读信号的作用下，读出一条长24位的微指令代码，并在打入命令CPIR的作用下，送入IR。    图13-1 微程序控制器框图  每当按一次脉冲键便产生一个负脉冲，该脉冲的作用是：  ·作为读控存的命令。  ·负脉冲当作CPIR将读出的微指令打入微指令寄存器IR。  ·负脉冲的上升沿使PC＋1形成下一条微指令的地址。  ·负脉冲反相后的上升沿作为寄存器打入脉冲。  （1）微程序计数器μPC的设计，完成8位具有加1功能和清除功能的计数器设计并封装，如图13-3。    图13-3 UPC的设计图  其中，CLR：清零端，，低电平有效；CLR=0时，Q7Q6Q5Q4Q3Q2Q1Q0=00000000；  LOAD：置数端，低电平有效；LOAD=0时，在CLK的上升沿，Q7Q6Q5Q4Q3Q2Q1Q0=D7D6D5D4D3D2D1D0；  当CLR=1，LOAD=1，ET=1，EP=1时，对CLK进行增1计数。  注意：本实验使用时，只有clk、CLR两引脚引出，其它引脚，ET、EP、LOAD接高电平。  （2）按图13-1完成微程序控制器的连线及引脚锁定。  \* 用单脉冲驱动μPC的计数脉冲CPPC、μRD、CPμIR。  \* 将μPC的8位输出锁定在A7－A0上。  \* 实验平台工作于模式5，将单脉冲锁定于键8，将μPC的复位端CLR锁定在键7，输出UIR的24位接数显3—数显8。  （3）设计并初始化控制存储器。         从ROM的0号单元开始写入、写入的微程序代码随意确定。  （4）读出微指令  ◆按CPU复位按键，清除微指令计数器。           ◆按单脉冲键           读出0号单元中的微指令代码送IR23-IR0同时显示在数显上；           将微指令计数器PC十1为读下条微指令做准备。          ◆连续按单脉冲键  微指令连续从ROM中读出并显示。                 注意：在读出过程中要和原先写入的代码比对看是否正确 | | | |
| 实验步骤：   1. 原理图输入：根据图所示电路，完成电路原理图设计。     （2）管脚锁定：完成原理图中输入、输出的管脚锁定。  引脚分配如下：  CLK——键8——PIO7——PIN84  CLR——键7——PIO6——PIN34  LOAD——键6——PIO5——PIN75  D7——DOUT8——DA7——PIO40——PIN135  D6——DOUT7——DA6——PIO41——PIN136  D5——DOUT6——DA5——PIO42——PIN137  D4——DOUT5——DA4——PIO43——PIN138  D3——DOUT4——DA3——PIO44——PIN141  D2——DOUT3——DA2——PIO45——PIN142  D1——DOUT2——DA0——PIO47——PIN144  D0——DOUT1——DA1——PIO46——PIN143  MOUT23——译码器6第4位——PIO39——PIN125  MOUT22——译码器6第3位——PIO38——PIN128  MOUT21——译码器6第2位——PIO37——PIN114  MOUT20——译码器6第1位——PIO36——PIN120  MOUT19——译码器5第4位——PIO35——PIN105  MOUT18——译码器5第3位——PIO34——PIN113  MOUT17——译码器5第2位——PIO33——PIN100  MOUT16——译码器5第1位——PIO32——PIN101  MOUT15——译码器4第4位——PIO31——PIN80  MOUT14——译码器4第3位——PIO30——PIN85  MOUT13——译码器4第2位——PIO29——PIN73  MOUT12——译码器4第1位——PIO28——PIN76  MOUT11——译码器3第4位——PIO27——PIN71  MOUT10——译码器3第3位——PIO26——PIN72  MOUT9——译码器3第2位——PIO25——PIN68  MOUT8——译码器3第1位——PIO24——PIN69  MOUT7——译码器2第4位——PIO23——PIN54  MOUT6——译码器2第3位——PIO22——PIN59  MOUT5——译码器2第2位——PIO21——PIN50  MOUT4——译码器2第1位——PIO20——PIN51  MOUT3——译码器1第4位——PIO19——PIN46  MOUT2——译码器1第3位——PIO18——PIN49  MOUT1——译码器1第2位——PIO17——PIN43  MOUT0——译码器1第1位——PIO16——PIN44  （3）原理图编译、适配和下载：在QuartusⅡ环境中选择EP4CE6/10器件，进行原理图的编译和适配，无误后完成下载。  （4）功能测试：利用输入开关及发光二极管LD测试逻辑运算部件的功能并记录测试结果。  我们初始化ROM如图, 将地址处的值设置为该地址    我们先进行PC = PC + 1的测试, 将LOAD置为1, CLR置为1, 不断按动脉冲键, 发现读出的值不断增加, 为预期的结果          接下来我们测试初始化地址为4时, LOAD置为0, 预期的输出为地址4对应的值4, 输出正确.      然后我们将LOAD置为1, 继续进行自增操作, 发现变为了5, 为预期结果, 成功!    （5）生成元件符号。 | | | |
| 仿真结果：  本次实验采取实际演示效果更明显, 因此不做仿真演示. | | | |
| 结论分析与体会：  通过实验，我深入理解了微程序控制器的工作原理。微程序控制器由控制存储器（CROM）、微程序计数器（PC）和微指令寄存器（IR）组成，能够实现复杂的控制逻辑。每次按下脉冲键产生的负脉冲，分别用于控制存储器的读取命令、微指令寄存器的加载和微程序计数器的递增。这个过程中，我学会了如何设计8位微程序计数器，包含加1和清除功能，并且明白了CLR、LOAD等控制信号的作用。  此外，通过将微程序控制器的各个部分连线并进行引脚锁定，我实际动手操作，掌握了硬件电路的连接与调试技巧。使用单脉冲驱动计数器、控制存储器读取、微指令寄存器加载等操作，使我更加理解了控制信号在微程序控制器中的作用。  在控制存储器初始化和微指令读取过程中，我学会了如何从ROM中写入和读出微指令代码，并通过与预先写入的代码进行比对，验证了读出过程的正确性。这部分实验强化了我对存储器和寄存器操作的理解，提升了调试和故障排除的能力。  总体而言，本次实验不仅增强了我对微程序控制器的理论理解，还通过实际操作提升了我的实践能力和问题解决能力。 | | | |
| 期间遇到的一些问题如下:   1. **如何初始化ROM?**   答 : 在创建的时候选择对应的初始化hex文件, 而不能中途修改的时候重新选择.   1. **LOAD的作用?**   答 : LOAD低电平有效, 为1的时候代表不进行初始化地址数据的读入, 进行自增操作, 为0的时候代表进行初始化地址数据的读入. | | | |