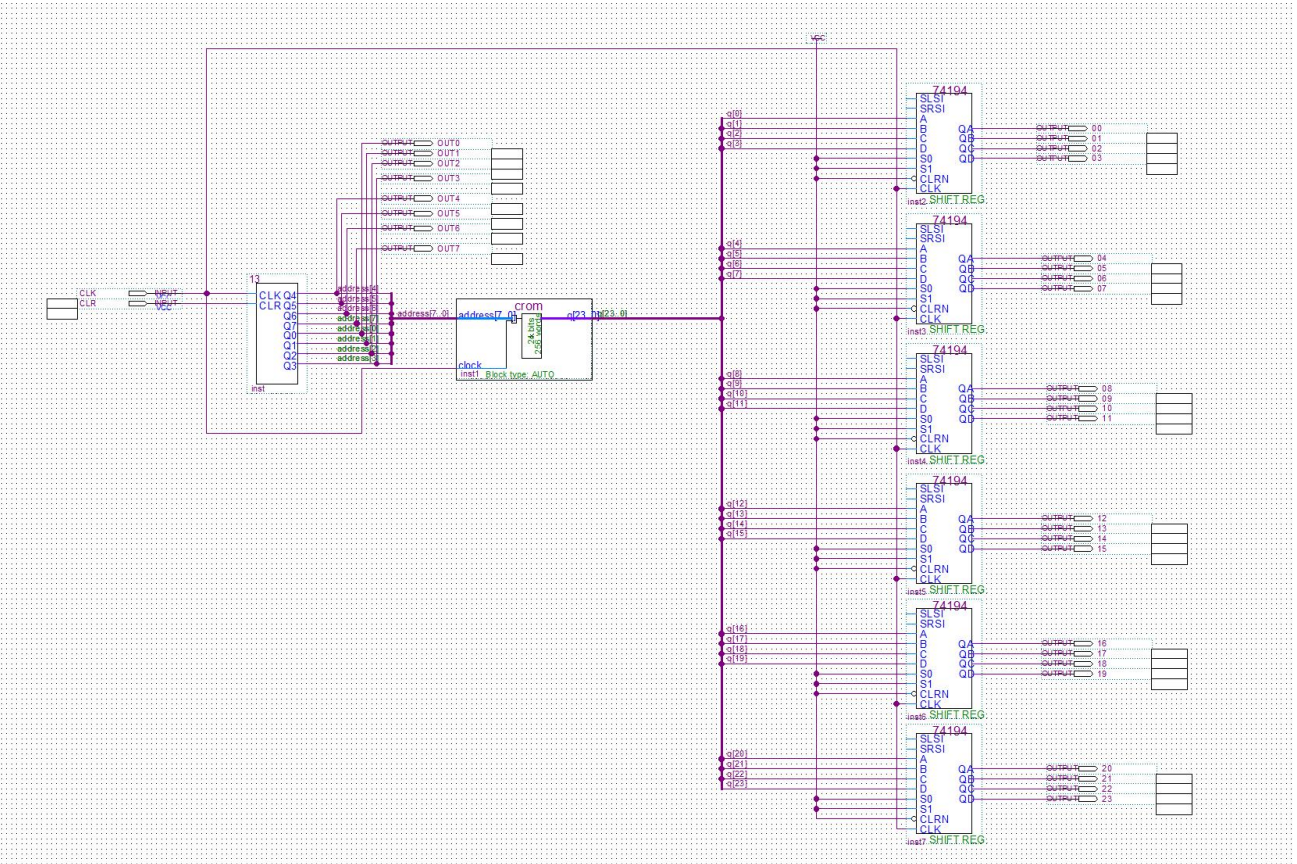


学号：202100130022	姓名：郭家宁	班级：2021 级数据班
实验题目： 控制器实验		
实验学时：	实验日期： 2022-12-4	
实验目的： 了解设计控制器的微程序设计方法		
硬件环境：1. 实验室台式机 2. 计算机组成与设计实验箱		
软件环境：QuartusII 软件		
实验内容与设计：		
1、实验内容		
<p>目前控制器设计大都采用微程序设计方法，又称存储逻辑控制器。微程序控制器电路结构如图 13-1 所示。它由控制存储器 CROM、微程序 μPC 计数器和微指令寄存器 μIR 构成。</p> <p>其中，微程序计数器 μPC 向控制存储器提供 8 位微地址，在控存读信号 $\overline{\mu RD}$ 的作用下，读出一条长 24 位的微指令代码，并在打入命令 $CP\mu IR$ 的作用下，送入 μIR。</p>		
		
<p>图 13-1 微程序控制器框图</p> <p>每当按一次脉冲键便产生一个负脉冲，该脉冲的作用是：</p> <ul style="list-style-type: none">• 作为读控存的命令 $\overline{\mu RD}$。• 负脉冲当作 $CP\mu IR$ 将读出的微指令打入微指令寄存器 μIR。• 负脉冲的上升沿使 $\mu PC+1$ 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。 <p>微程序时序如图 13-2。</p> 		
<p>图 13-2 微程序时序</p>		

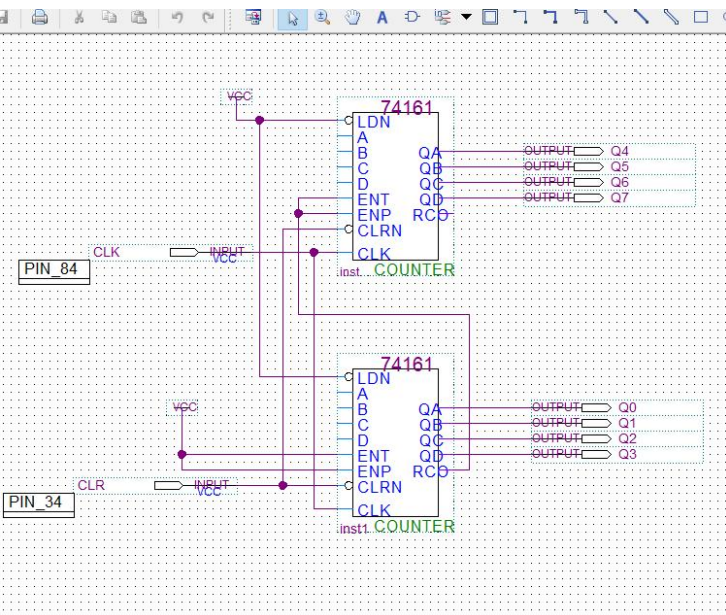
2、实验原理图



3、实验步骤

. 设计 PC 计数器:

PC 控制器逻辑图



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 μ PC 的计数脉冲 CPPC、 μ RD、CP μ IR。

* 将 μ PC 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 μ PC 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 + 1 为读下条微指令做准备。

◆连续按单脉冲键

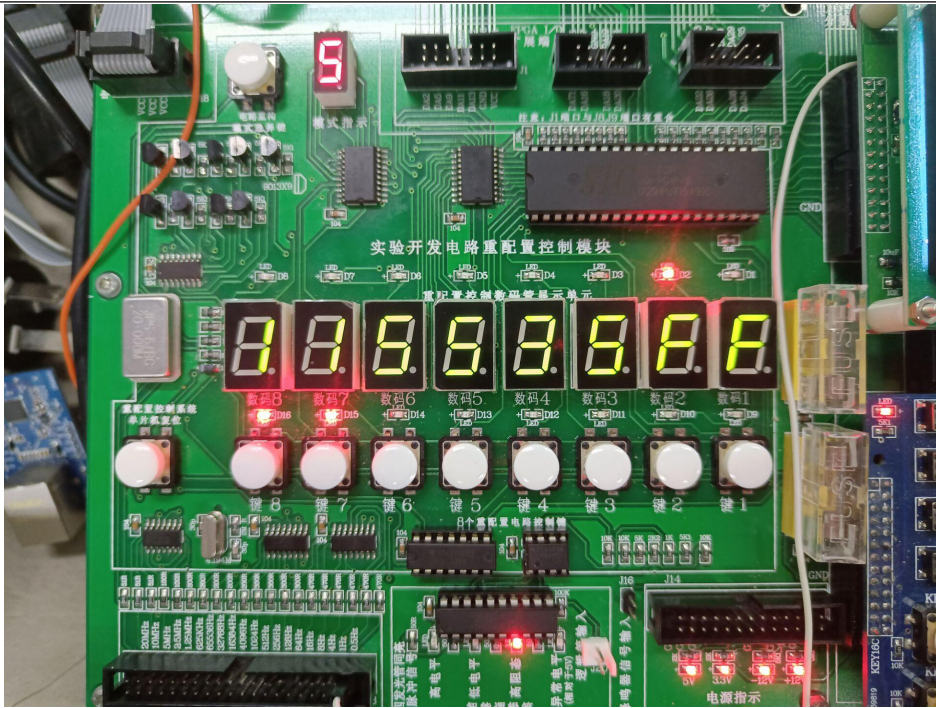
微指令连续从 ROM 中读出并显示。

注意: 在读出过程中要和原先写入的代码对比看是否正确

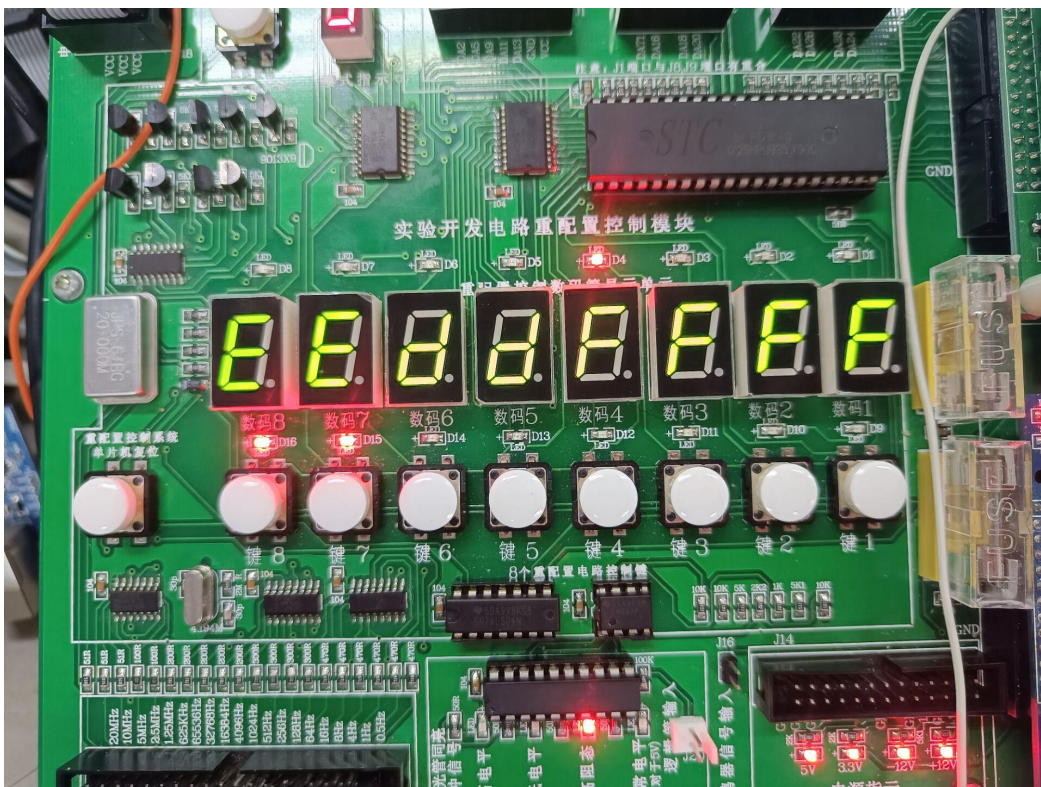
4、实验结果

Addr	+0	+1	+2	+3	+4	+5	+6	+7	ASCII
000	23FFEE	115535	554254	453435	453455	000000	AABBCC	EEDDFF
008	AA5555	BE4567	122345	432234	544554	678666	2223CC	EDCA34
010	000000	000000	000000	000000	000000	000000	000000	000000
018	000000	000000	000000	000000	000000	000000	000000	000000
020	000000	000000	000000	000000	000000	000000	000000	000000
028	000000	000000	000000	000000	000000	000000	000000	000000
030	000000	000000	000000	000000	000000	000000	000000	000000
038	000000	000000	000000	000000	000000	000000	000000	000000
040	000000	000000	000000	000000	000000	000000	000000	000000
048	000000	000000	000000	000000	000000	000000	000000	000000
050	000000	000000	000000	000000	000000	000000	000000	000000
058	000000	000000	000000	000000	000000	000000	000000	000000
060	000000	000000	000000	000000	000000	000000	000000	000000
068	000000	000000	000000	000000	000000	000000	000000	000000

rom 中对应提前输入的数据



地址（上边的一排 led 灯对应 8 位地址）为 02H 读出的数据经过译码为 115535，对应正确



地址为 08H 读出的数据为 EEDDF 也正确

清零复位后可以得出相同的实验结果

结论分析与体会：

对控制器的实现与逻辑有了更加深刻的理解与体会