**《计算机硬件技术》实验作业**

**第三次仿真实验**

**学院：汽车学院**

**姓名：贾林轩**

**学号：1853688**

**软件：proteus8**

仿真实验三I/O接口电路

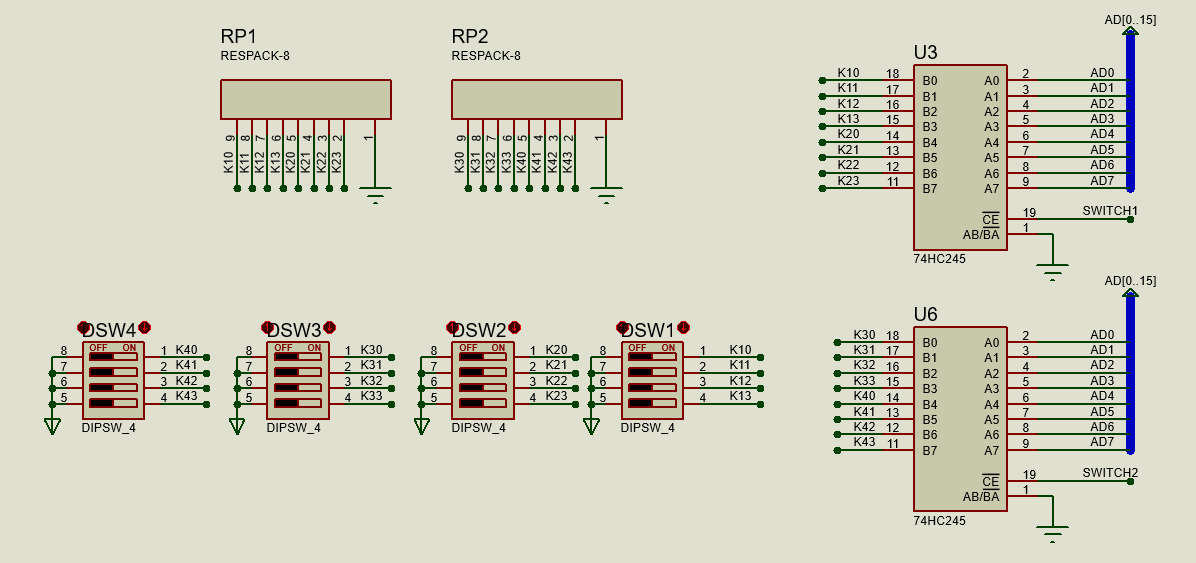
1. 实验目的

熟悉硬件仿真软件proteus界面及相关操作，理解电路设计原理及程序设计思路，重点理解IO端口的输入输出功能及地址译码功能。

1. 实验内容

软件打开所给文件夹 计算机硬件仿真/sim（1）/查询开关.pdspri示例，理解该硬件电路实现原理。本次实验任务需要在此基础上对原理图或程序做出修改，并成功执行，报告中需要演示并说明该硬件电路实现了什么功能以及创新点。

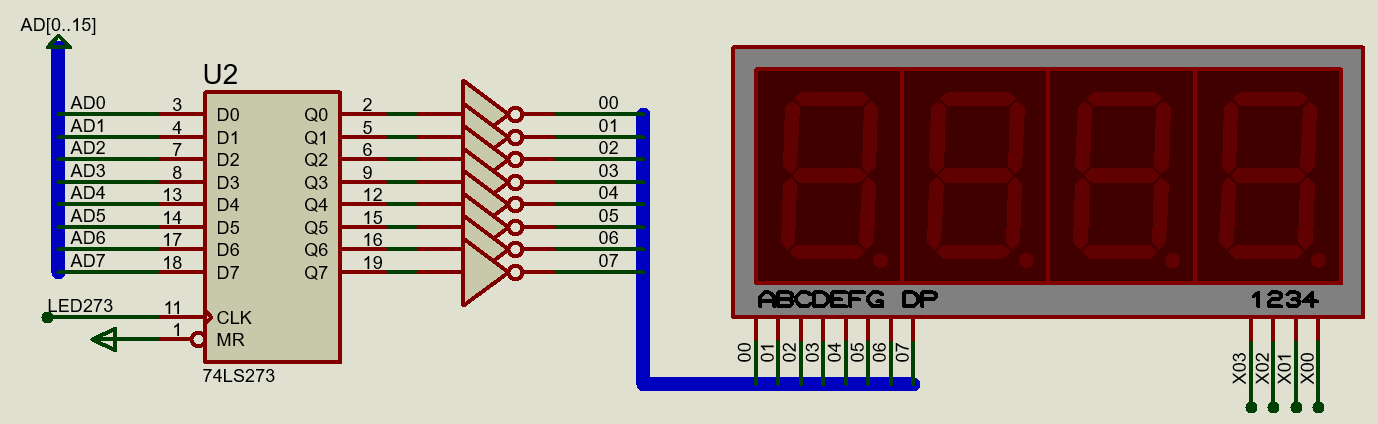
1. 实验过程
2. 目标实现功能
3. 通过四个四通道开关，通过**动态显示**，共同控制一个数码管进行四位十六进制数的显示:页码：P2
4. 用开关和**矩阵键盘**分别实现数码管的段选和位选，从而实现：“选定显示的位——显示特定的数”的功能:页码：P10
5. 硬件电路设计思路
6. 输入接口



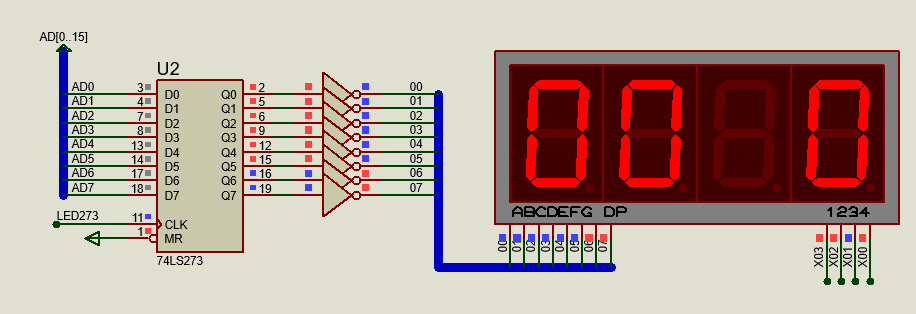
通过下拉电阻控制开关闭合无输入时为低电阻，四个开关分别对应数码管的四个位，第一位对应开关连接到第一片74HC245的低四位（K10-K13），第二位对应开关连接到第一片芯片的高四位（K20-K23），第三位对应开关连接到第二片芯片的低四位（K30-K33），第四位对应开关连接到第二片芯片的高四位（K40-K43）；

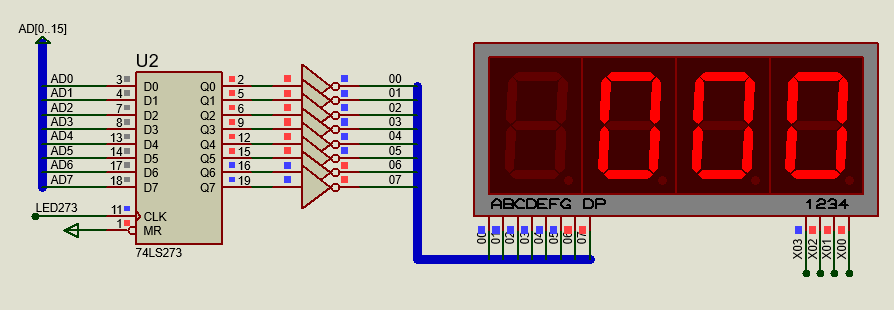
通过地址译码选定输入接口（SWITCH1和SWITCH2）后即可从接口读到开关状态信息（下文软件程序部分详述）；

1. 输出接口

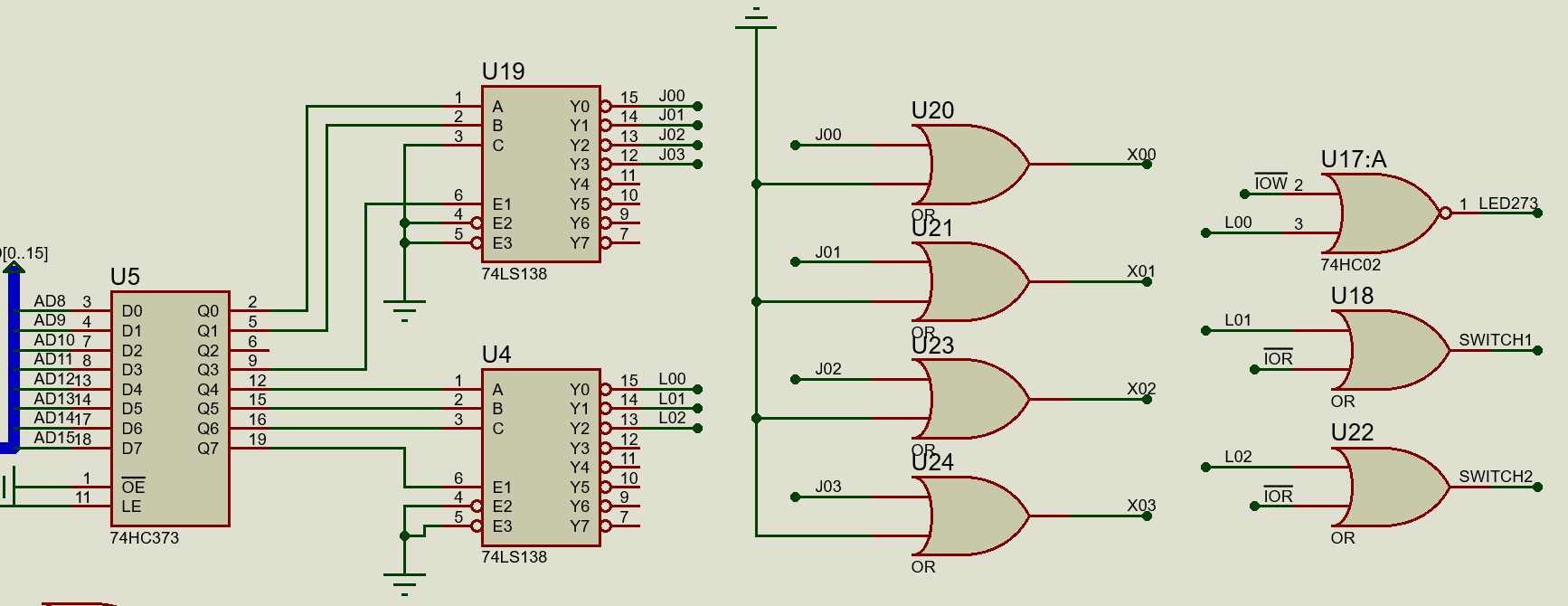


通过地址译码进行位选（X00,X01,X02,X03），选定不同的位后，将数据信号输输到输出接口（LED273），重复上述过程，当显示频率较低时（通过延时子程序实现），可以看到输出有明显的闪烁（如下图所示），当显示频率超过人眼的识别频率时，即看不见闪烁，达到动态显示效果；





1. 译码电路



上文分析到，对四位数码管进行位选只需两位地址码即可，因此将Q0,Q1（AD8,AD9）接到74LS138的A,B端，将C端接地；

根据此译码电路设计，可分析出各输入输出接口地址：

9000H:输入接口SWITCH1

0A000H：输入接口SWITCH2

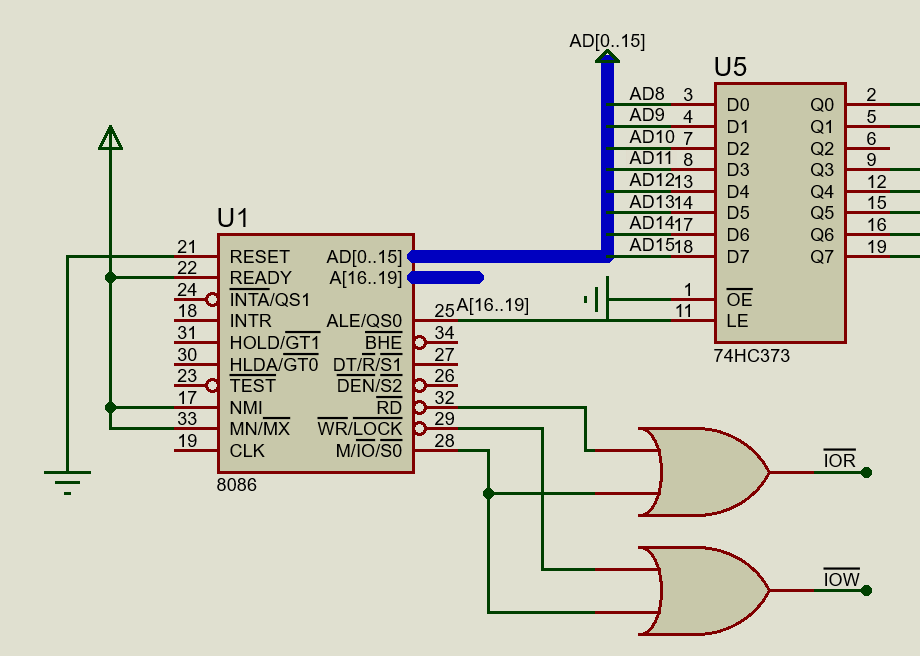
8800H:输出接口1（位选数码管第1位，数据输出到LED273）

8900H:输出接口2（位选数码管第2位，数据输出到LED273）

8A00H:输出接口3（位选数码管第3位，数据输出到LED273）

8B00H:输出接口4（位选数码管第4位，数据输出到LED273）

1. CPU及外围电路（无变化）



1. 软件程序设计思路（不同点已加粗表示）

STACK SEGMENT 'STACK' ;定义堆栈段

STA DB 100 DUP(?) ;定义堆栈大小

TOP EQU LENGTH STA ;栈顶偏移地址

STACK ENDS

DATA SEGMENT 'DATA'

ORG 2H ;数据段起始地址

**LED1 EQU 8800H ;输出端口1地址**

**LED2 EQU 8900H ;输出端口2地址**

**LED3 EQU 8A00H ;输出端口3地址**

**LED4 EQU 8B00H ;输出端口4地址**

**SWITCH1 EQU 9000H ;输入端口SWITCH1地址**

**SWITCH2 EQU 0A000H ;输入端口SWITCH2地址**

SEG7 DB 3fh,06h,5bh,4fh,66h,6dh,7dh,07h ;SEG7储存七段管码表（0-F共十六个数）

DB 7fh,6fh,77h,7ch,39h,5eh,79h,71h

DATA ENDS

CODE SEGMENT 'CODE'

ASSUME CS:CODE,SS:STACK,DS:DATA

START: MOV AX, DATA

MOV DS, AX

MOV AX, STACK

MOV SS, AX

MOV AX, TOP

MOV SP, AX ;初始化段寄存器

LEA BX,SEG7 ;取7段码表基地址

MOV AX,0H ;AX清零

GO:

;低两位

;第一位

**MOV DX,SWITCH1 ;状态接口的地址**

IN AL,DX ;读入开关状态

**AND AL,0FH ;保留低4位**

MOV SI,AX ;作为7段码表的表内位移量

MOV AL,[BX+SI] ;取7段码

**MOV DX, LED4 ;输出接口地址**

OUT DX,AL ;输出到对应接口

;CALL DELAY

;第二位

**MOV DX,SWITCH1 ;状态接口的地址**

IN AL,DX ;读入开关状态

**AND AL,0F0H ;保留高4位**

**MOV CL,4**

**SHR AL,CL ;AL右移四位**

MOV SI,AX ;作为7段码表的表内位移量

MOV AL,[BX+SI] ;取7段码

**MOV DX, LED1 ;输出接口地址**

OUT DX,AL ;输出到对应接口

;CALL DELAY

;高两位

;第三位

**MOV DX,SWITCH2 ;状态接口的地址**

IN AL,DX ;读入开关状态

**AND AL,0FH ;保留低4位**

MOV SI,AX ;作为7段码表的表内位移量

MOV AL,[BX+SI] ;取7段码

**MOV DX, LED2 ;输出接口地址**

OUT DX,AL ;输出到对应接口

;CALL DELAY

;第四位

**MOV DX,SWITCH2 ;状态接口的地址**

IN AL,DX ;读入开关状态

**AND AL,0F0H ;保留高4位**

**MOV CL,4**

**SHR AL,CL ;AL右移四位**

MOV SI,AX ;作为7段码表的表内位移量

MOV AL,[BX+SI] ;取7段码

**MOV DX, LED3 ;输出接口地址**

OUT DX,AL ;输出到对应接口

;CALL DELAY

JMP GO

DELAY PROC ; ----延时子程序----

PUSH CX

MOV CX,3FFH

DELAY1: NOP

LOOP DELAY1

POP CX

RET

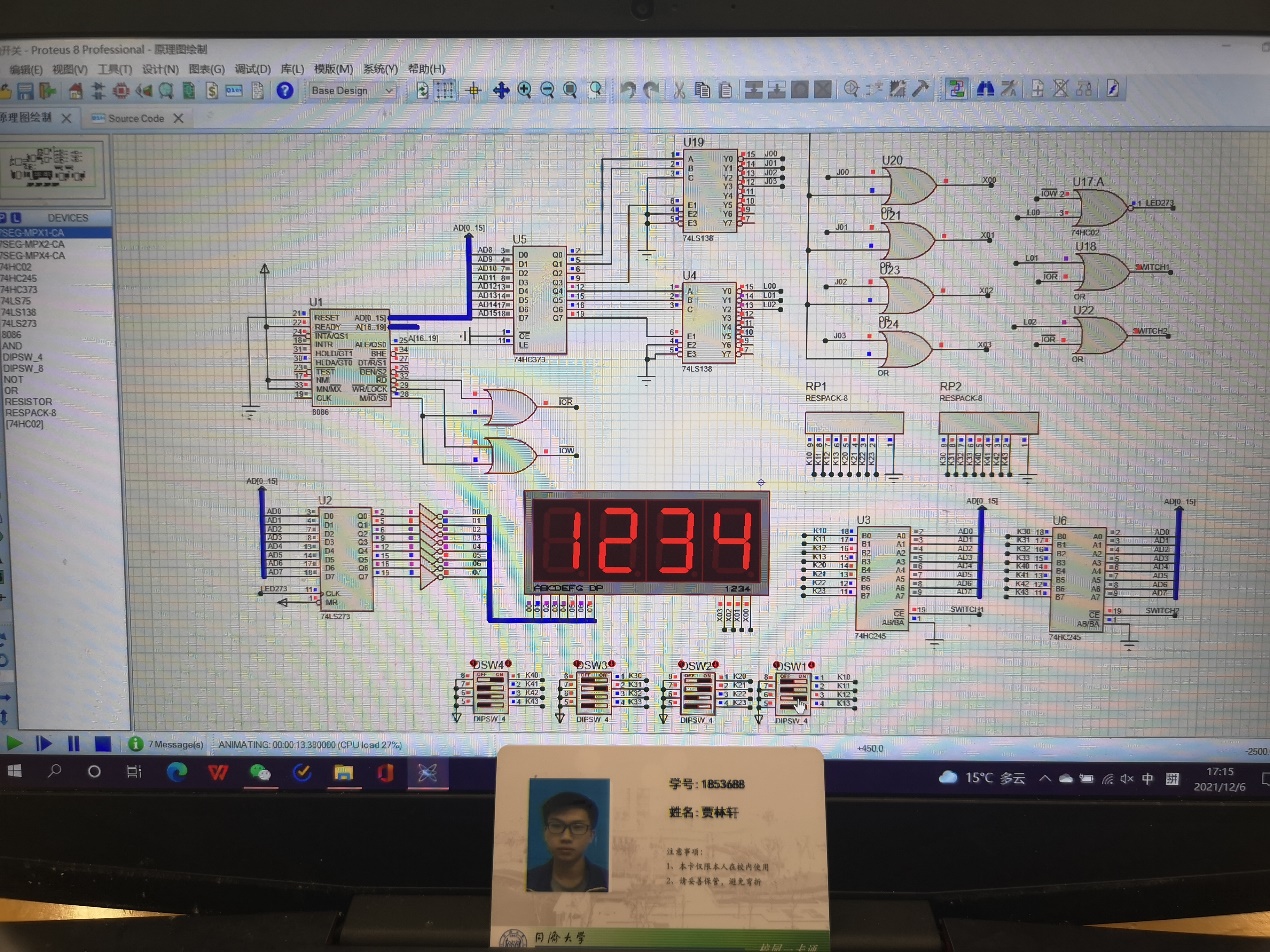
DELAY ENDP

CODE ENDS

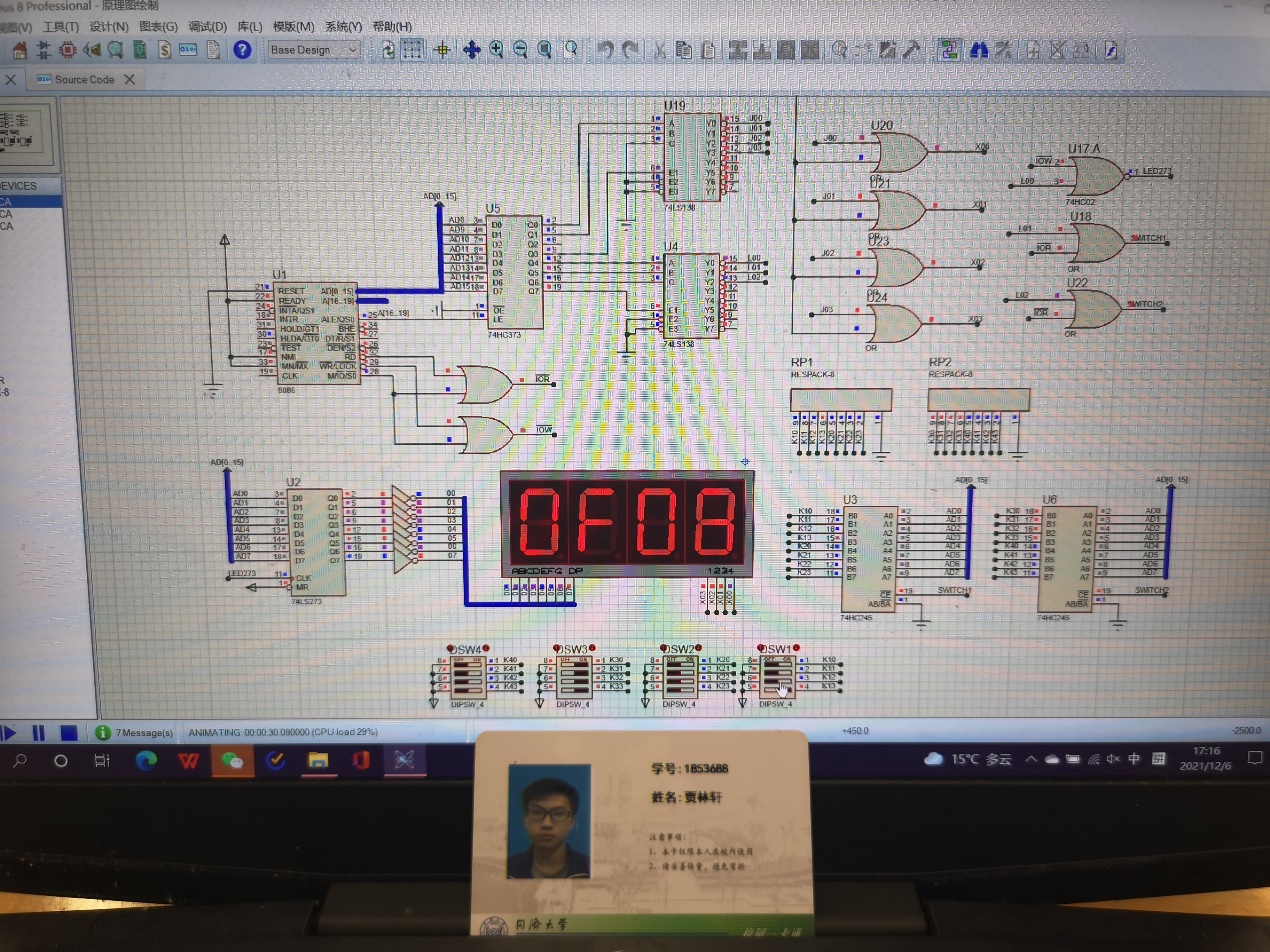
END START

1. 仿真结果

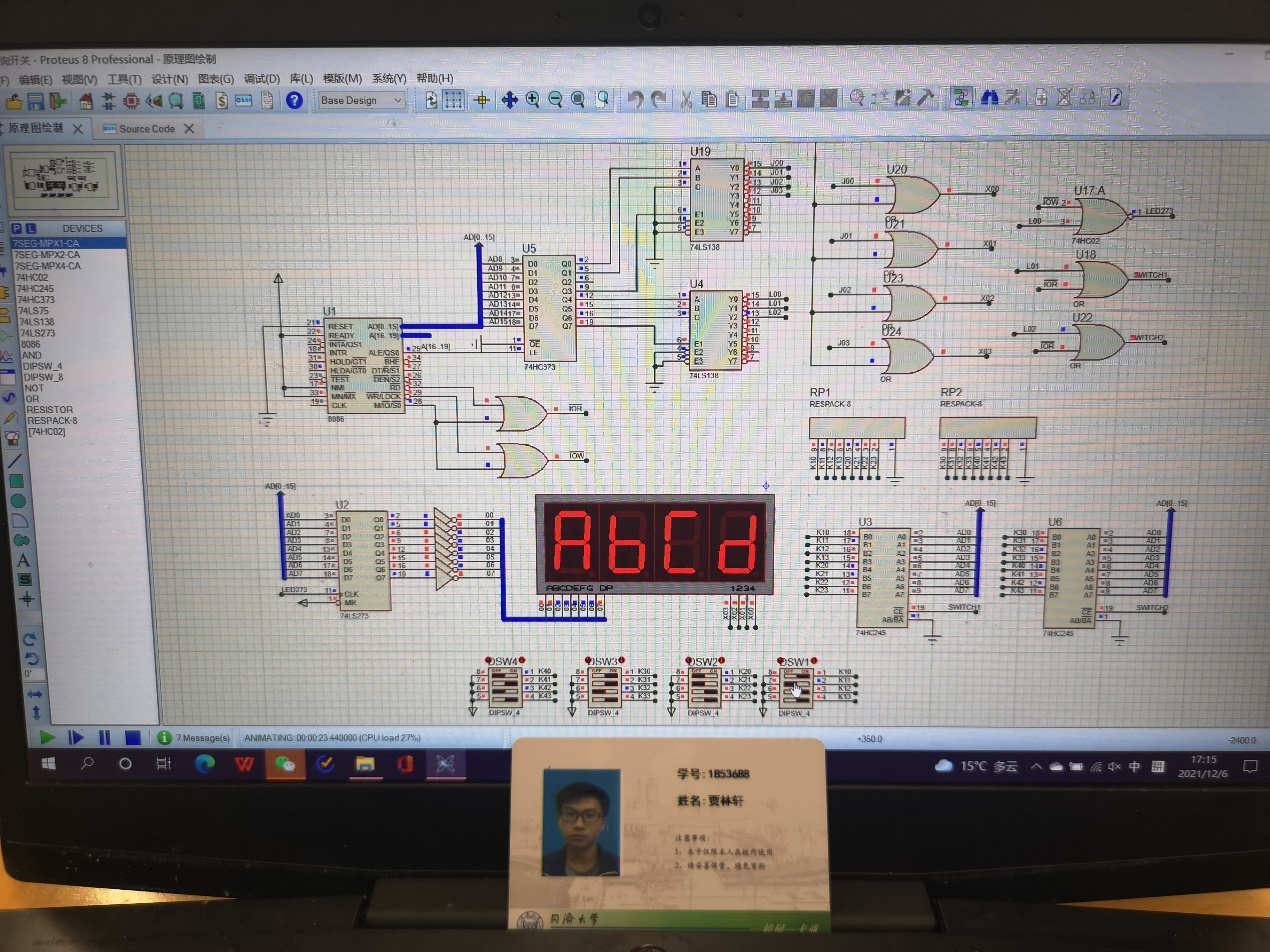
显示数字1234



显示数字0F08



显示数字ABCD

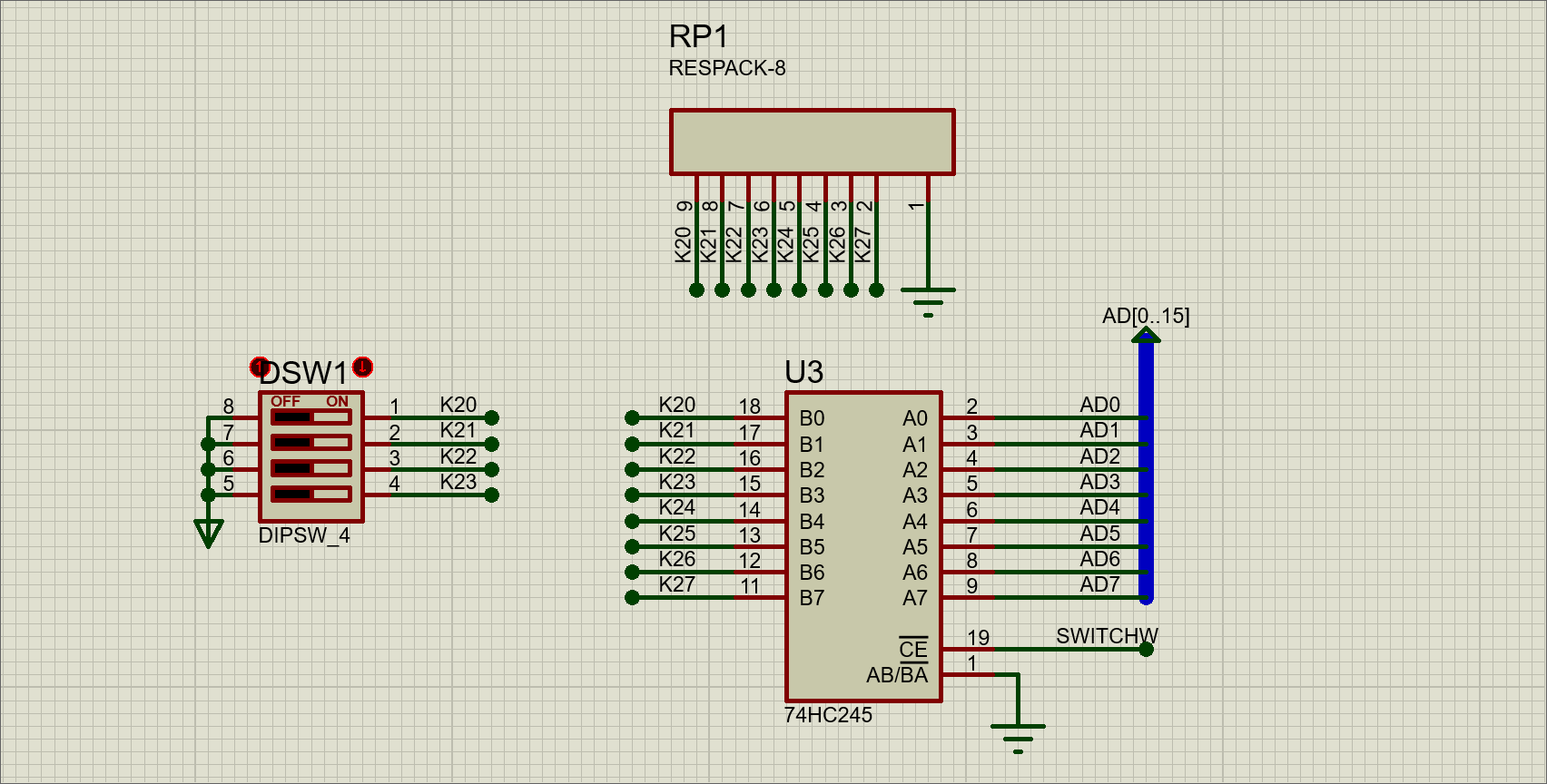


功能2：

一．硬件电路设计

* 1. 四并联开关位选电路设计（输入接口）

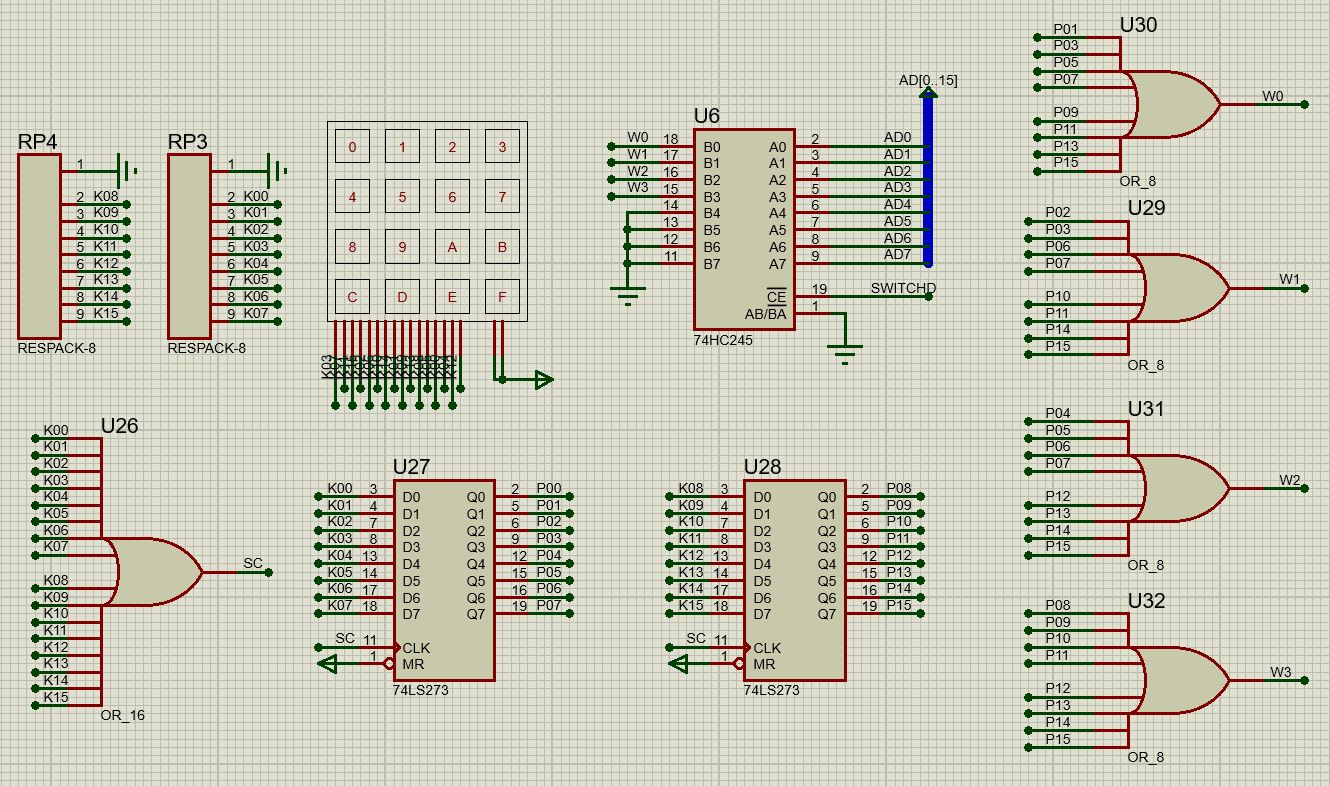
作为位选输入，接口地址为SWITCHW



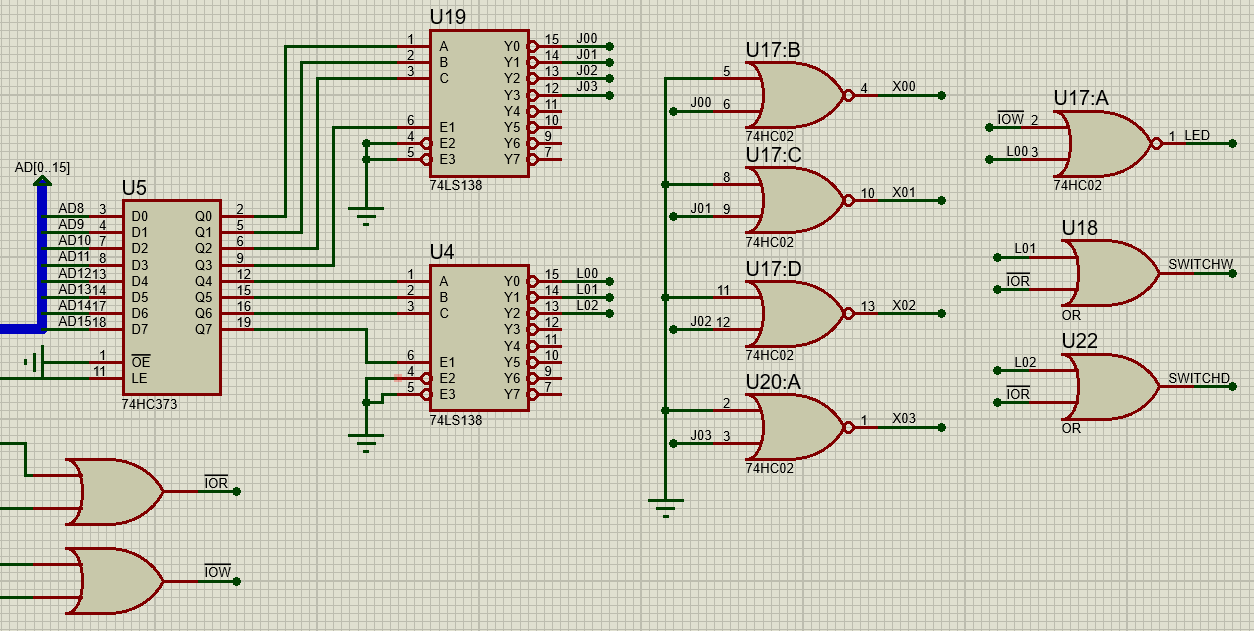
* 1. 4×4矩阵键盘段选电路设计（输入接口）

作为段选输入，接口地址为SWITCHD

过程中发现矩阵键盘的输入不能保持（松开键后数据消失）因此需要将信号用锁存器锁存，原件U26为一个16输入或门电路，用以生成一个上升沿SC输入到74LS273的脉冲端，当K00-K15任何一位为1时（即有输入），SC为1，同时此上升沿可以将K00-K15的电平信息锁存在Q端（P00-P15），U29-U30组成16-4编码电路，将P00-P15的电平信息编码为W0-W3的四位二进制数，用以从7段管码表中进行段选；



* 1. 输入输出端口地址译码电路



根据此译码电路设计，可分析出各输入输出接口地址：

9000H:位选输入接口SWITCHW

0A000H：段选输入接口SWITCHD

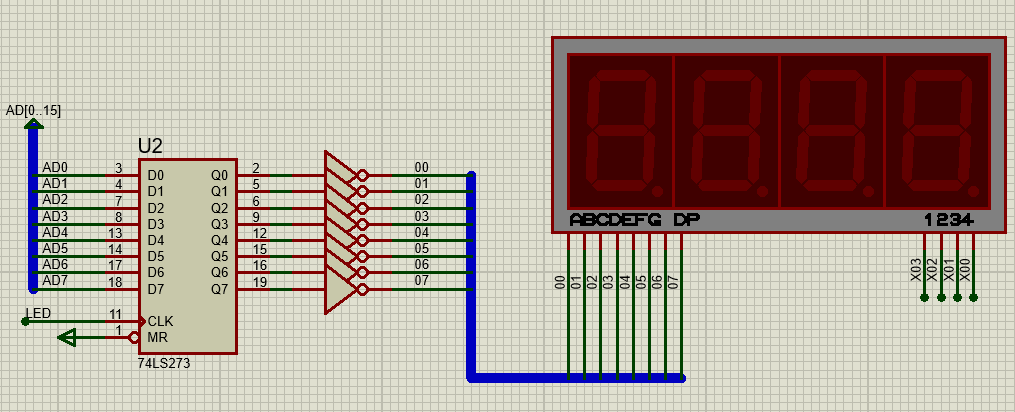
8800H:输出接口1（位选数码管第1位，数据输出到LED）

8900H:输出接口2（位选数码管第2位，数据输出到LED）

8A00H:输出接口3（位选数码管第3位，数据输出到LED）

8B00H:输出接口4（位选数码管第4位，数据输出到LED）

* 1. 数码管显示电路（输出接口）



二．软件程序设计

STACK SEGMENT 'STACK' ;定义堆栈段

STA DB 100 DUP(?) ;定义堆栈大小

TOP EQU LENGTH STA ;栈顶偏移地址

STACK ENDS

DATA SEGMENT 'DATA'

ORG 2H ;数据段起始地址

**LED1 EQU 8800H ;输出端口1地址**

**LED2 EQU 8900H ;输出端口2地址**

**LED3 EQU 8A00H ;输出端口3地址**

**LED4 EQU 8B00H ;输出端口4地址**

**LED DW LED1,LED2,LED3,LED4 ;将输出端口制表方便取用**

**SWITCHW EQU 9000H ;位选输入端口SWITCHW地址**

**SWITCHD EQU 0A000H ;段选输入端口SWITCHD地址**

SEG7 DB 3fh,06h,5bh,4fh,66h,6dh,7dh,07h

DB 7fh,6fh,77h,7ch,39h,5eh,79h,71h

DATA ENDS

CODE SEGMENT 'CODE'

ASSUME CS:CODE,SS:STACK,DS:DATA

START: MOV AX, DATA

MOV DS, AX

MOV AX, STACK

MOV SS, AX

MOV AX, TOP

MOV SP, AX ;初始化段寄存器

MOV AX,0H ;AX清零

GO:

MOV DX,SWITCHW ;取位选接口地址

IN AX,DX ;从位选接口读位选信息

MOV SI,AX ;位选信息作为偏移地址

LEA BX,LED ;取输出接口表

MOV CX,[BX+SI] ;暂存输出接口地址

MOV DX,SWITCHD ;取段选接口地址

IN AX,DX ;从端选接口读段选信息

MOV SI,AX ;段选信息作为偏移地址

LEA BX,SEG7 ;取七段管码表

MOV AL,[BX+SI] ;取管码

MOV DX,CX ;取输出接口地址

OUT DX,AL ;输出

CALL DELAY ;延时

JMP GO

DELAY PROC ; ----延时子程序----

PUSH CX

MOV CX,03FFH

DELAY1: NOP

LOOP DELAY1

POP CX

RET

DELAY ENDP

CODE ENDS

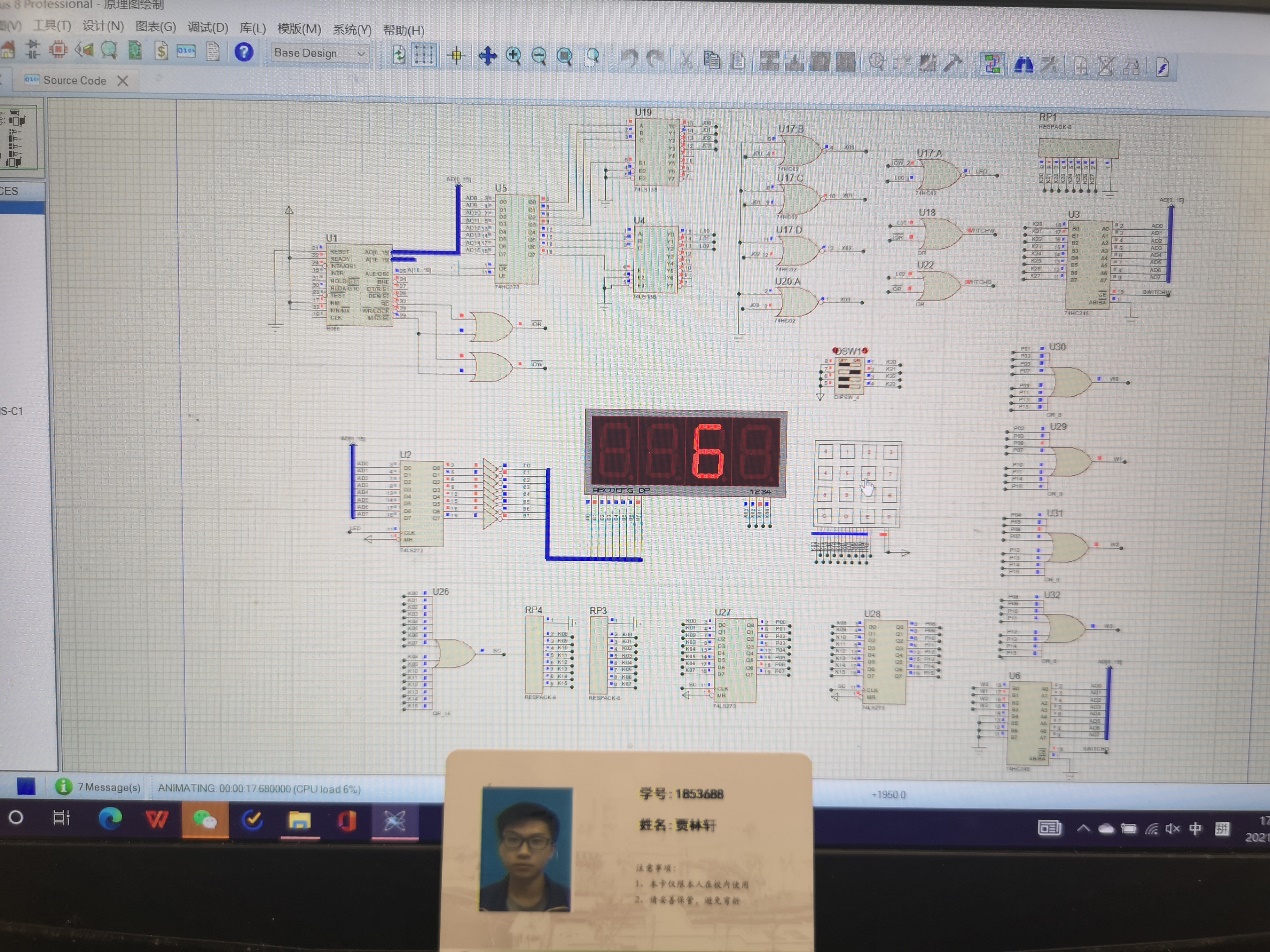
END START

三.仿真结果

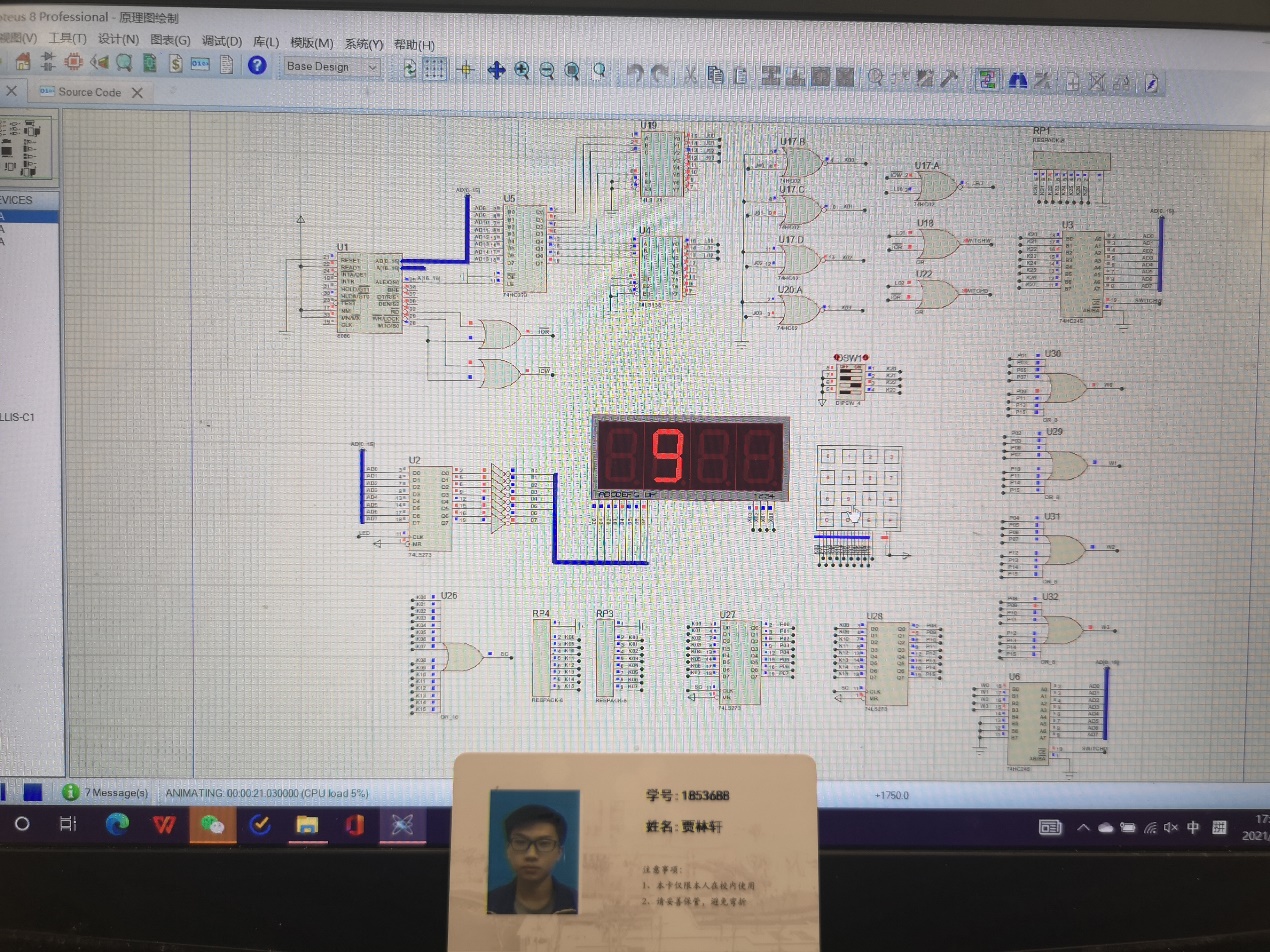
1. 在第1位显示3



2.在第2位显示6



3.在第3位显示9



4.在第4位显示12

