|  |
| --- |
| 实验目的：  1、了解数码管显示原理；  2、掌握读表程序的编写；  3、熟练Proteus仿真软件的使用；  4、熟练单片机实验箱的实验操作。 |
| 实验内容及实验器材：  1、利用IO口实现动态扫描数码管，数码管显示“12345678”。  LED数码显示原理：七段LED显示器内部由七个条形发光二极管和一个小圆点发光二极管组成，根据各管的极管的接线形式，可分成共阴极型和共阳极型。LED数码管的g~a七个发光二极管因加正电压而发亮，因加零电压而不以发亮，不同亮暗的组合就能形成不同的字形，这种组合称之为字形码。  2、由于显示的数字0－9的字形码没有规律可循，只能采用查表的方式来完成我们所需的要求了。这样我们按着数字0－9的顺序，把每个数字的笔段代码按顺序排好！建立的表格如下所示：  code dis\_code[11]={0xc0,0xf9,0xa4,0xb0, 0x99,0x92,0x82,0xf8,0x80,0x90, 0xff};  // 0，1，2，3，4，5，6，7，8，9，off  3.23.3 实验设备和器件  PC机 一台  PROTEUS仿真软件 一套  实验箱 一台  PLink下载器 一个 |
| 实验过程：（实验步骤、数据记录及处理）  1、Keil工程创建  参考附录1文档，在Keil4开发环境中创建一个Keil工程，并进行相关的工程设置，也可以直接使用资料提供的Keil工程进行实验。  2、仿真实验  1）包含代码工程和调试功能的仿真：  参考实验1-3或附录2文档，在Proteus仿真软件中新建原理图工程、代码工程，并完成相关的工程设置，也可直接使用资料提供的Proteus模板工程文件进行实验。  根据实验要求设计电路原理图和修改/增加代码，工程编译成功后即可在原理图界面进行仿真实验，并观察仿真实验现象。  2）纯仿真电路的仿真  参考附录2文档，新建原理图工程，并绘制当前实验的电路原理图，在控制器的属性中，关联到Keil编译生成的.hex文件路径，即可直接进行仿真，观察实验现象。  3、硬件实验  1）通过Proteus烧录  参考实验1-3或附录3文档，将Plink下载器一端通过USB线连接到电脑USB口，另一端通过10P线连接到核心板  在Proteus工程的快捷工具栏中的下拉列菜单Debug /Release/配置选项中，选择Release后，点击左侧工程设置选项，并按照实验1-3或附录3设置，设置完成后点击确定；点击上传图标，等待上传成功。  2）通过P-Link烧录  参考附录3文档，写好代码并生成HEX文件后，打开P-Link软件；点击“自动识别”选项，识别出对应芯片型号后，左下角将显示初始化完成提示；点击“打开文件(Flash)”选项，找到Keil生成的HEX文件，并按照下图所示设置，设置完毕后直接点击“自动编程”按键，烧录完成后左下角会提示“写Flash完成”。  代码如下：  #include <reg52.h>  #include <intrins.h>  unsigned char data dis\_digit;  unsigned char code dis\_code[11] = {0xc0, 0xf9, 0xa4, 0xb0, // 0, 1, 2, 3  0x99, 0x92, 0x82, 0xf8, 0x80, 0x90, 0xff}; // 4, 5, 6, 7, 8, 9, off  unsigned char data dis\_buf[8];  unsigned char data dis\_index;  void main()  {  P0 = 0xff;  P2 = 0x00;  TMOD = 0x01;  TH0 = 0xFC;  TL0 = 0x17;  IE = 0x82;  dis\_buf[0] = dis\_code[0x1];  dis\_buf[1] = dis\_code[0x2];  dis\_buf[2] = dis\_code[0x3];  dis\_buf[3] = dis\_code[0x4];  dis\_buf[4] = dis\_code[0x5];  dis\_buf[5] = dis\_code[0x6];  dis\_buf[6] = dis\_code[0x7];  dis\_buf[7] = dis\_code[0x8];    dis\_digit = 0x01;  dis\_index = 0;    TR0 = 1;  while(1);  }  void timer0() interrupt 1  {  TH0 = 0xFA;  TL0 = 0x17;    P2 = 0xFF;  P0 = dis\_buf[dis\_index];  P2 =~ dis\_digit;  dis\_digit = \_crol\_(dis\_digit, 1);  dis\_index++;    dis\_index &= 0x07;  } |
| **实验结论：** |