|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 佛山科学技术学院  实验报告  实验名称 实验5 计数器  实验项目 由计数器触发LED绿色和红色灯闪烁  专业班级 22物联网工程2 姓名 学号  指导教师 罗平 成绩 日期 2024/5/3     1. 实验目的   1．学习80C51单片机计数器的使用和编程。  2．熟悉计数器中断处理程序的编程。   1. 实验电路（元件清单）   1、实验电路原理图    2、元件清单   |  |  |  | | --- | --- | --- | | **元器件编号** | **元器件名称** | **说明** | | U1 | AT89C52 | AT89C52 单片机 | | R9~R16 | RES | 电阻 | | D9~D16 | LED-RED  LED-GREEN | 红色 LED  绿色 LED | |  | 7SEG-BCD-GRN | 7 段 BCD 数码管 | |  | LOGICTOGGLE | 单脉冲触发器 |  1. 实验内容 2. 绘制原理图。      1. 编写程序。   #include <REGX52.H>  unsigned char T0\_count = 0;  unsigned char T1\_count = 0;  unsigned char total\_count = 0;  unsigned char HEXToBCD(unsigned char x)  {  return (x / 10) \* 16 + (x % 10);    }  void Delayxms(unsigned char xms) //@12.000MHz  {  while(xms--)  {  unsigned char i, j;  i = 2;  j = 239;  do  {  while (--j);  } while (--i);    }    }  void Init\_Counter()  {      //计数器初始化  TF0 = 0;  TF1 = 0;  TR0 = 1;  TR1 = 1;    TMOD = 0x55;    IE=0x8a;  //中断初始化  // IT0 = 1; // 1 = 下降沿触发，0 = 低电平触发  // IT1 = 1; // 1 = 下降沿触发，0 = 低电平触发  // EA = 1; //允许中断  // EX0 = 1; //允许外部中断0  // EX1 = 1; //允许外部中断1  //  //译码管初始化  P1 = HEXToBCD(0);    //计数初始化  TH0 = 0xff;  TL0 = 0xfd;      TH1 = 0xff;  TL1 = 0xfa;      }  void main()  {  Init\_Counter();      while(1)  {      }  }  void Timer0(void) interrupt 1 //  {  unsigned char i;    TH0 = 0xff;  TL0 = 0xfd;  for(i = 1; i <= 3; i++)  {  P0 = ~P0;  Delayxms(500);  P0 = ~P0;  Delayxms(500);  }    total\_count++;    P1 = HEXToBCD(total\_count);        }  void Timer1(void) interrupt 3 //外部中断1  {  unsigned char i;    TH1 = 0xff;  TL1 = 0xfa;    for(i = 1; i <= 6; i++)  {  P2 = ~P2;  Delayxms(500);  P2 = ~P2;  Delayxms(500);  }    total\_count++;    P1 = HEXToBCD(total\_count);    }  3、主要模块程序流程图。     1. 实验数据调试及讨论   外部输入3个计数脉冲后,接在 P0口的绿色 LED 闪烁3次  外部输入6个计数脉冲后,，接在 P2口的红色 LED 闪烁6次.  TO 和T1 总的中断次数在 P1口7段 BCD 数码管显示。       1. 软件清单   Main.c  Project.hex |
|  |