**实验代码**

1. 利用定时器中断实现LED1大约每隔4秒闪烁一次（自由运行模式，模模式，正计数/倒计数模式分别实现）。

2. 定时器1和定时器3分别控制LED1（模模式）和LED2（正计数/倒计数模式）以不同频率闪烁，LED1大约5秒闪烁一次，LED2大约2秒闪烁一次。

#include <ioCC2530.h>

#define uint unsigned int   
#define uchar unsigned char   
#define uint32 unsigned long  
#define led1 P0\_0  
#define led2 P2\_0  
uint counter=0;uint counter2=0;  
void LED\_Init()  
{  
   P0SEL &=~0X01;

P0DIR |=0X01;

P2SEL &=~0X01;

P2DIR |=0X01;  
   led1=1;

led2=1;  
}  
void TIM1\_Init()  
{  
   CLKCONCMD&=~0X7f;

while(CLKCONSTA & 0x40);  
   EA=1;

T1IE=1;  
   T1CTL=0x06;

T1CCTL0 |=0X04;  
   T1CC0H=0xff;

T1CC0L=0xff;  
}  
void TIM3\_Init()  
{  
  EA=1;T3IE=1;  
   T3CTL=0X7f;

T3CCTL0 |=0X04;  
   T3CC0=0xff;  
}  
void main()  
{  
   LED\_Init();

TIM1\_Init();

TIM3\_Init();  
  while(1)  
  {  
  }  
}  
#pragma vector = T1\_VECTOR  
\_\_interrupt void LED1(void)  
{  
    IRCON=0x00;  
    if(counter <300)  
    {  
      counter++;  
    }else{  
      counter=0;  
      led1=!led1;  
    }  
}  
#pragma vector = T3\_VECTOR  
\_\_interrupt void LED2(void)  
{  
    IRCON=0x00;  
    if(counter2 <15625)  
    {  
      counter2++;  
    }else{  
      counter2=0;  
      led2=!led2;  
    }  
}

1. 当串口接收到来自串口调试助手发送来的0x01时改变LED2 状态，并发送"welcome"
2. 通过定时器和串口协调配合，每隔6秒向PC发送 "everything is ok"
3. 实验平台通过串口向PC串口发送字符串“What is your name?”，计算机向平台发送名字，名字的以#号结束，实验平台向串口发送字符串“HELLO”+名字。

\*4. 编程实现，串口如果收到”LED11#”，打开LED1，如果收到”LED12#”，关闭LED1; 串口如果收到”LED21#”，打开LED2，如果收到”LED22#”，关闭LED2。

#include <ioCC2530.h>  
#include <stdlib.h>  
#include <stdio.h>  
#include "string.h"  
#define uint unsigned int   
#define uchar unsigned char   
#define uint32 unsigned long  
#define led1 P0\_0  
#define led2 P2\_0  
#define LEN 6  
uint counter=0;uint counter2=0;  
void LED\_Init()  
{  
   P0SEL &=~0X01;

P0DIR |=0X01;  
  P2SEL &=~0X01;

P2DIR |=0X01;  
   led1=1;

led2=1;  
}  
void delay(int time)  
{  
  for(int i=0;i<time;i++)  
    for(int j=0;j<i;j++);  
}  
/\*\*\*\*\*\*\*\*\*串口实验\*\*\*\*\*\*\*\*\*/  
char Txdata[25]="welcome";  
char Txdata1[20]="everything is ok";  
uchar str1[20]="what is your name?";  
uchar str2[30]="hello,zhuohongjia\n";  
uchar Recdata[LEN];  
uchar temp;  
uchar RXTXflag=1;  
uint datanumber = 0;  
uint stringlen;  
void initUARTSEND()  
{  
  CLKCONCMD&=~0X40;  
  while(CLKCONSTA & 0x40);  
  CLKCONCMD&=~0X47;  
  PERCFG=0x00;  
  P0SEL = 0x3c;  
  U0CSR |=0x80;  
  U0GCR |=11;  
  U0BAUD |=216;  
  UTX0IF=1;  
  U0CSR |=0X40;  
  IEN0 |=0x84;  
}  
void UartTX\_Send\_String(char \*Data,int len)  
{  
  int j;  
  for(j = 0;j < len;j++)  
  {  
    U0DBUF = \*Data++;  
    while(UTX0IF == 0);  
    UTX0IF=0;  
  }  
}  
void main()  
{  
  int a=0;  
  LED\_Init();  
  initUARTSEND();  
  while(1)  
  {  
    if(RXTXflag==1)  
    {  
      if(temp!=0)  
      {  
        if((temp != '#')&&(datanumber < 20))  
        {  
          Recdata[datanumber++]=temp;  
        }  
        else  
        {  
          RXTXflag=3;  
        }  
        temp=0;  
      }  
    }  
  if(RXTXflag==3)  
  {  
    a=strcmp((char const\*)Recdata,"LED11")+11;  
    if(a==11)  
    {  
      led1=0;  
      UartTX\_Send\_String((char \*)Recdata,datanumber);  
    }  
    a=strcmp((char const\*)Recdata,"LED12")+12;  
    if(a==12){  
      led1=1;  
      UartTX\_Send\_String((char \*)Recdata,datanumber);  
    }  
    a=strcmp((char const\*)Recdata,"LED21")+21;  
    if(a==21){  
      led2=0;  
      UartTX\_Send\_String((char \*)Recdata,datanumber);  
    }  
    a=strcmp((char const\*)Recdata,"LED22")+22;  
    if(a==22){  
      led2=1;  
      UartTX\_Send\_String((char \*)Recdata,datanumber);  
    }  
 //  U0CSR &=~0x04;  
//    UartTX\_Send\_String((char \*)str2,sizeof("hello,zhuohongjia\n"));  
 //   UartTX\_Send\_String((char \*)Recdata,datanumber);  
//   U0CSR |=0x04;  
    RXTXflag=1;  
    datanumber=0;  
  }  
}  
}  
  
#pragma vector = URX0\_VECTOR  
\_\_interrupt void Receive()  
{  
  URX0IF = 0;  
  temp= U0DBUF;  
}

1. 每隔10秒采集一次VDD/3的电压值并显示出来，并且当串口接收到来自串口调试助手发送来的0x01时，发送VDD/3的电压值。

\*2. 每隔20秒询问一次是否有AD采集任务，若收到回复“temp sensor”则令ADC采集片内温度传感器的值并显示出来，若收到回复“vdd”则令ADC采集VDD/3的值并显示出来。

1. #include <ioCC2530.h>//第一题  
   #include <stdlib.h>  
   #include <stdio.h>  
   #include "string.h"  
   #define uint unsigned int   
   #define uchar unsigned char   
   #define uint32 unsigned long  
   #define LEN 20  
   uint counter=0;  
   uint counter2=0;  
   uchar temp;  
   uchar RXTXflag=1;  
   uint datanumber = 0;  
   uint AvgTemp;  
   char adcdata[20];  
   void ADC\_Init()  
   {  
     ADCH &= 0x00; //EOC 清零  
     ADCCON3 = 0x1f;   
     ADCCON1 &= 0x30; //停止A/D  
     ADCCON1 |= 0x40; //启动A/D  
   }  
   unsigned int getVoltage()  
   {  
     ADC\_Init();  
     unsigned int value = 0;  
     while(!(ADCCON1 & 0x80));  
     value = ADCL >> 2;  
     value |= (((unsigned int)ADCH) << 6);  
     return value;  
   }  
   void TIM1\_Init()//自由模式   
   {  
     CLKCONCMD&=~0X7f;  
     while(CLKCONSTA & 0x40);  
     EA=1;  
     T1IE=1;  
     T1CTL=0x05;  
   }  
   void UartSend\_String(char \*Data,int len)  
   {  
     int j;  
     for(j=0;j<len;j++)  
     {  
       U0DBUF = \*Data++;  
       while(UTX0IF == 0);  
       UTX0IF = 0;  
     }  
   }  
   void delay(int time)  
   {  
     for(int i=0;i<time;i++)  
       for(int j=0;j<i;j++);  
   }  
   void main()  
   {  
     ADC\_Init();  
     initUARTSEND();  
     TIM1\_Init();  
     LED\_Init();  
     P0DIR |= 0x01;  
     while(1);  
   }  
   #pragma vector = T1\_VECTOR  
   \_\_interrupt void LED(void)  
   {  
       char wbuf[20];  
       IRCON=0x00;  
       if(counter <625)//6  
       {  
         counter++;  
       }else{  
         counter=0;  
         AvgTemp=getVoltage()\*3.3/8192;  
         sprintf(wbuf,"VDD/3值为：%d",AvgTemp);  
           led1=~led1;  
         UartSend\_String(wbuf,strlen(wbuf));         
       }  
   }  
   #pragma vector = URX0\_VECTOR  
   \_\_interrupt void Receive()  
   {  
       float num = 0;   // 清中断标志   
       char c = U0DBUF;  
       if(c == 0x01)  
       {  
           memset(adcdata,0,30);  
           num = AvgTemp\*3.3/8192;  
           sprintf(adcdata,"AD转换值为%12.9fv",num);  
           UartSend\_String(adcdata,strlen(adcdata));  
           num=0;  
       }  
       URX0IF = 0;   
   }
2. #include <ioCC2530.h>//第二题  
   #include <stdlib.h>  
   #include <stdio.h>  
   #include "string.h"  
   #define uint unsigned int   
   #define uchar unsigned char   
   #define uint32 unsigned long  
   #define LEN 20  
   uint counter=0;  
   uint counter2=0;  
   static uint AvgTemp;  
   static uint Temp;  
   char adcdata[20];  
   char c;  
   void ADC\_Init()  
   {  
     ADCH &= 0x00; //EOC 清零  
     ADCCON3 = 0x1f;   
   //  ADCCON1 &= 0x30; //停止A/D  
     ADCCON1 |= 0x40; //启动A/D  
   }  
   unsigned int getVoltage()//测电压  
   {  
     ADC\_Init();  
     unsigned int value = 0;  
     while(!(ADCCON1 & 0x80));  
     value = ADCL >> 2;  
     value |= (((unsigned int)ADCH) << 6);  
     return value;  
   }  
   uint getTemperature(void)//测温度  
   {  
     char i;  
     uint AdcValue;  
     uint value;  
     AdcValue=0;  
     for(i = 0;i < 4; i++)  
     {  
       ADCCON3 |=0x3E;  
       ADCCON1 |=0x40;  
       while(!(ADCCON1 & 0x80));  
       value = ADCL >> 2;  
       value |=(((uint)ADCH)<<6);  
       AdcValue+=value;  
     }  
     value = AdcValue >> 2;  
     value = value\*0.0629-303.3;  
     return value;  
   }  
   void TIM1\_Init()  
   {  
     CLKCONCMD&=~0X7f;  
     while(CLKCONSTA & 0x40);  
     EA=1;  
     T1IE=1;  
     T1CTL=0x05;  
   }  
   void UartSend\_String(char \*Data,int len)  
   {  
     int j;  
     for(j=0;j<len;j++)  
     {  
       U0DBUF = \*Data++;  
       while(UTX0IF == 0);  
       UTX0IF = 0;  
     }  
   }  
   void delay(int time)  
   {  
     for(int i=0;i<time;i++)  
       for(int j=0;j<i;j++);  
   }  
   void main()  
   {  
     initUARTSEND();  
     TIM1\_Init();  
     while(1);  
   }  
   #pragma vector = T1\_VECTOR  
   \_\_interrupt void LED(void)  
   {  
       IRCON=0x00;  
       if(counter <1250)//6  
       {  
         counter++;  
       }else{  
         counter=0;  
         UartSend\_String("请发送指令",sizeof("请发送指令"));  
       }  
   }  
   #pragma vector = URX0\_VECTOR  
   \_\_interrupt void Receive()  
   {  
       char wbuf[20];  
       char temp1[3];  
       c = U0DBUF;  
       if(c == 't')  
       {  
           memset(temp1,0,sizeof(temp1));  
           Temp=0;  
           Temp = getTemperature();  
           temp1[0]=(char)Temp/10+0x30;  
           temp1[1]=(char)Temp%10+0x30;  
           temp1[3]='\0';  
           sprintf(adcdata,"温度转换值为%d",temp1[0]);  
           UartSend\_String(adcdata,strlen(adcd ata));  
       }  
       if(c == 'v')  
       {  
         AvgTemp=getVoltage()\*3.3/8192;  
         sprintf(wbuf,"VDD/3值为：%d",AvgTemp);  
         UartSend\_String(wbuf,strlen(wbuf));  
       }  
       URX0IF = 0; // 清中断标志   
   }

无线射频收发实验

#include <ioCC2530.h>

#include <stdlib.h>

#include <stdio.h>

#include "string.h"

#define LED1 P0\_0

#define LED2 P2\_0

#define RX

char buf[128];

int len=0;

void Delay(int time)

{

for(int i=0;i<time;i++)

for(int j=0;j<i;j++);

}

void initUARTSEND()//串口

{

CLKCONCMD&=~0X40;

while(CLKCONSTA & 0x40);

CLKCONCMD&=~0X47;

SLEEPCMD |= 0x04;

PERCFG=0x00;

P0SEL = 0x3c;

U0CSR |=0x80;

U0GCR |=11;

U0BAUD |=216;

UTX0IF=0;

U0CSR |=0X40;

IEN0 |=0x84;

}

void UartSend\_String(char \*Data,int length)

{

int j;

for(j=0;j<length;j++)

{

U0DBUF = \*Data++;

while(UTX0IF == 0);

UTX0IF = 0;

}

}

void initrf()

{

//FRMCTRL0 |=(0x20|0x40);

TXFILTCFG=0x09;

AGCCTRL1=0x15;

FSCAL1=0x00;

RFIRQM0 |=(1<<6);

IEN2 |=(1<<0);

EA=1;

FREQCTRL=0x0b;

SHORT\_ADDR0=0x05;

SHORT\_ADDR1=0x00;

PAN\_ID0=0x22;

PAN\_ID1=0x00;

RFST=0xed;

RFST=0xe3;

FRMFILT0 &=~(1<<0);

}

void tx(void)

{

char i;

char mac[]={"hncu"};

RFST=0xe3;

while(FSMSTAT1 &((1<<1)|(1<<5)));

RFIRQM0 &=~(1<<6);

IEN2 &=~(1<<0);

RFST =0xee;

RFIRQF1=~(1<<1);

RFD=7;

for(i=0;i<6;i++)

{

RFD=mac[i];

}

RFIRQM0 |=(1<<6);

IEN2 |=(1<<0);

RFST =0xe9;

while(!(RFIRQF1 &(1<<1)));

RFIRQF1=~(1<<1);

LED2=~LED2;

Delay(200);

}

void main(void)

{

initUARTSEND();

P0SEL &=~0x01;

P2SEL &=~0x01；

P0DIR |=0x01；

P2DIR |=0x01;

LED1=1;

LED2=1;

EA=0;

SLEEPCMD &= ~0x04;

while(!(SLEEPSTA & 0x40));

CLKCONCMD &= ~0x47;

SLEEPCMD |= 0x04;

initrf();//初始化RF

EA = 1;//中断使能

//发送或等待接收中断

while(1)

{

#ifndef rx

tx();

Delay(200);

Delay(200);

# else

//UartSend\_String(buf,len);

#endif

}

}

#pragma vector=RF\_VECTOR

\_\_interrupt void rf\_isr(void)

{

char i;

EA=0;

if(RFIRQF0&(1<<6))

{

len =RFD-2;

len &=0x7f;

for(i = 0; i < len; i++)

{

buf[i] = RFD;

Delay(200);

}

RFST=0xED;

UartSend\_String(buf,len);

S1CON =0;

RFIRQF0 &= ~(1<<6);

LED1 = ~LED1;

//Uart0Send\_String(buf);

}

EA=1;

}

（1）节点1通过红外传感器采集信息

（2）人体进入测量范围时，节点1通过无线传输通知中心节点2

（3）节点2蜂鸣器产生提示警报并显示警报信息

#include <ioCC2530.h>  
#define uint unsigned int   
#define uchar unsigned char   
#define uint32 unsigned long  
#define led1 P0\_0  
#define led2 P2\_0  
#define Red P1\_0 //高电平   
#define Beeper P0\_5  
unsigned int len;  
unsigned char buf[4];  
unsigned char Buff[4]="QST";  
void delay(int n)  
{  
  unsigned int i,j;  
  for(i=0;i < n;i++)  
  {  
    for(j=0;j < n;j++)  
    {  
      asm("NOP");  
      asm("NOP");  
      asm("NOP");  
    }  
  }  
}  
void Delay(int n)  
{  
  for(int i = 0;i < n ;i++)  
    for(int j = i; j>0;j--);  
}  
void LED\_Init()  
{  
  P0SEL &=~0X01;  
  P0DIR |=0X01;  
  P2SEL &=~0X01;  
  P2DIR |=0X01;  
  led1=0;  
  led2=0;  
}  
void RedInit()  
{  
  P1SEL &=~0x01;  
  P1DIR &=~0x01;  
}  
void Beep\_Init()  
{  
  P0SEL &=~0x20;  
  P0DIR |=0x20;  
  Beeper = 0;  
}  
void initUARTSEND()  
{  
  CLKCONCMD&=~0X40;  
  while(CLKCONSTA & 0x40);  
  CLKCONCMD&=~0X47;  
    
  PERCFG=0x00;  
  P0SEL = 0x3c;  
  P2DIR &= ~0XC0;  
  U0CSR |=0x80;  
  U0GCR |=11;  
  U0BAUD |=216;  
  UTX0IF=0;  
  U0CSR |=0X40;  
  IEN0 |=0x84;  
}  
void UartTX\_Send\_String(char \*Data,int len)  
{  
  int j;  
  for(j=0;j<len;j++)  
  {  
    U0DBUF = \*Data++;     //将数据写入DBUF  
    while(UTX0IF == 0);   //等待发送完毕  
    UTX0IF = 0;  
  }  
}  
void rf\_init()  
{  
  //硬件CRC以及AUTO\_ACK使能  
FRMCTRL0 |= (0x20 | 0x40); /\* AUTO\_ACK | AUTO\_CRC \*/  
        //设置TX抗混叠过滤器以获得合适的带宽  
TXFILTCFG = 0x09;  
        //调整AGC目标值  
AGCCTRL1 = 0x15;  
        //获得最佳的EVM  
FSCAL1 = 0x00;  
// RXPKTDONE 中断位使能  
RFIRQM0 |= (1<<6);  
//  RF 中断使能  
IEN2 |= (1<<0);  
        //开中断  
        EA = 1;  
        //信道选择，选择12信道  
FREQCTRL = 0x0c;   
        //目标地址过滤期间使用的短地址  
SHORT\_ADDR0 = 0x05;  
SHORT\_ADDR1 = 0x00;  
        //目标地址过滤期间使用的PANID  
PAN\_ID0 = 0x22;   
PAN\_ID1 = 0x00;  
        //清除RXFIFO缓冲区并复位解调器    
RFST = 0xed;   
        //为RX使能并校准频率合成器  
RFST = 0xe3;   
        //禁止帧过滤  
        FRMFILT0 &= ~(1<<0);  
}  
void tx(char \* mac,int len)  
{  
     unsigned char i;       
    //为RX使能并校准频率合成器  
RFST = 0xe3;   
 //wait for SFD not active and TX\_Active not active   
        // TX\_ACTIVE | SFD   
while (FSMSTAT1 & ((1<<1) | (1<<5)));   
        //禁止RXPKTDONE中断  
RFIRQM0 &= ~(1<<6);   
        //禁止RF中断  
IEN2 &= ~(1<<0);   
        // ISFLUSHTX   
RFST = 0xee;   
        // 清除 TXDONE 中断   
RFIRQF1 = ~(1<<1);  
        // 传输的帧长度     
        RFD = len+2;  
        //将mac的内容写到RFD中  
        for(i=0;i<len;i++)  
        {  
          RFD = mac[i]；  
        }   
// 打开RX中断   
RFIRQM0 |= (1<<6);  
//打开RF中断  
        IEN2 |= (1<<0);  
        //ISTXON   
RFST = 0xe9;   
 //等待传输结束  
        while (!(RFIRQF1 & (1<<1)));  
        //清除 TXDONE状态    
       RFIRQF1 = ~(1<<1);  
       //LED1灯状态改变  
       led2=~led2;  
       //延时  
       delay(200);  
}  
void main()  
{  
  LED\_Init();  
  RedInit();  
  initUARTSEND();  
  CLKCONCMD &=~0x7f;    
  while(!(SLEEPSTA & 0x40));//CLKCONSTA  
  rf\_init();  
  EA = 1;  
  Beep\_Init();  
  //Beep=1;  
  while(1)  
  {  
    #ifndef RX  
    led1=1;  
    if(Red==1)  
    {  
      tx("QST",sizeof("QST"));  
      //led2=!led2;  
      delay(200);  
      delay(200);  
    }  
    #else  
   // UartTX\_Send\_String("have people",sizeof("have people"));  
    led2=1;  
    #endif  
  }  
}  
#pragma vector = URX0\_VECTOR  
\_\_interrupt void Receive()  
{  
    URX0IF = 0;  
 //temp= U0DBUF;//第四题  
}  
#pragma vector=RF\_VECTOR  
\_\_interrupt void rf\_isr(void)   
{  
  unsigned char i;  
  EA=0;  
  //UartTX\_Send\_String("OK",sizeof("OK"));  
  if(RFIRQF0 & (1<<6))  
  {  
    len=RFD-2;  
    for(i = 0; i < len; i++)  
    {  
      buf[i]=RFD;  
      delay(200);  
    }  
    RFST=0xed;  
    S1CON = 0;  
    RFIRQF0 &=~(1<<6);  
    led1=~led1;  
    if(buf[0]=='Q')  
    {  
      Beeper=1;  
      Delay(200);  
      Beeper=0;  
      Delay(200);  
      UartTX\_Send\_String("have people",sizeof("have people"));  
    }  
    //UartTX\_Send\_String(buf,strlen(buf));  
    memset(buf,'\0',4);  
  }  
  EA=1；  
}