

用STM32F103读取JY62角度传感器的陀螺仪、加速度的数据

<>

文章目录

- 1 准备工作
- 2 接线方式
- 3 程序讲解
 - 3.1 串口程序讲解
 - 3.2 数据处理部分
 - 3.3 UART打印程序
- 4 生成下载文件
- 5 下载程序
- 6 输出结果显示

◇1 准备工作

1. 硬件设备：JY62模块、六合一、USB-TTL、杜邦线、STM32F103C8T6单片机。给大家看下图片。



2) 软件准备：KEIL 5 (5.17版本的)
3) 示例代码

◇2 接线方式

1、使用数据线将STM32F103C8T6与电脑连接通信 注意：给STM32F103C8T6需要用5V才可以
2、JY62与STM32F103C8T6使用杜邦线连接。具体接线如下图所示：
具体接线方式如下：

USB-TTL STM32F103C8T6 JY-62

VCC(5V)	VCC	VCC
TX	RX1(PA10),PA3	TX
RX	TX1(PA9),PA2	RX
GND	GND, GND	GND

◇3 程序讲解

◇3.1 串口程序讲解

从上面的接线可以看出呢。一共用了32单片机的两个UART引脚。一个是UART1这个是把处理过的数据发送到PC端。一个是UART2这个是用来接收JY62传过来的数据。UART2也是使用中断来接收到数据。这些都是在初始化UART2的时候来配置的。前面那部分都是默认的串口配置。设置波特率、数据位8位、停止位1、无奇偶检验、配置串口模式。最后再初始化USART2。一般都是固定的。后面的USART_ITConfig。就是开启串口中断。

```
USART_InitStructure.USART_BaudRate = baudrate;
USART_InitStructure.USART_WordLength = USART_WordLength_8b;
USART_InitStructure.USART_StopBits = USART_StopBits_1;
USART_InitStructure.USART_Parity = USART_Parity_No ;
USART_InitStructure.USART_HardwareFlowControl = USART_HardwareFlowControl_None;
USART_InitStructure.USART_Mode = USART_Mode_Rx | USART_Mode_Tx;
USART_Init(USART2, &USART_InitStructure);
USART_ITConfig(USART2, USART_IT_TXE, DISABLE); //串口2的发送端TX是不使能的
USART_ITConfig(USART2, USART_IT_RXNE, ENABLE); //串口2的接收端的RX使能
使能串口中断后呢。UART2的数据都会一个个字节的往DR寄存器里面写。
void USART2_IRQHandler(void)
{
    if (USART_GetITStatus(USART2, USART_IT_TXE) != RESET) //由于TX中断被禁止了所以这个if是不会进入的。
    {
        USART_SendData(USART2, TxBuffer[TxCounter++]);
        USART_ClearITPendingBit(USART2, USART_IT_TXE);
        if (TxCounter == count) USART_ITConfig(USART2, USART_IT_TXE, DISABLE);
    }
    else if (USART_GetITStatus(USART2, USART_IT_RXNE) != RESET) //直接进入RX的这个中断
    {
        CopeSerial2Data((unsigned char)USART2->DR); //把UART2的数据给到DR寄存器后呢，再由这个CopeSerial2Data函数把数据给到给出去了
        USART_ClearITPendingBit(USART2, USART_IT_RXNE);
    }

    USART_ClearITPendingBit(USART2, USART_IT_ORE);
}
```

当串口2中断接收到数据后会通过上面的CopeSerial2Data () 函数。存在ucRxBuffer[250]。这个数组里面。然后就可以根据商家给的通讯协议去进行数据处理了。

```
void CopeSerial2Data(unsigned char ucData)
{
    static unsigned char ucRxBuffer[250];
    static unsigned char ucRxCnt = 0;

    LED_REVERSE(); //接收到数据，LED灯闪烁一下
    ucRxBuffer[ucRxCnt++] = ucData; //将中断的数据存在了这个数组里
    if (ucRxBuffer[0] != 0x55) //数据头不对，则重新开始寻找0x55数据头
    {

```

```
ucRxCnt=0;
return;
}
```

<>3.2数据处理部分

根据商家的协议由下图所示

7.2.3 角度输出：

数据编号	数据内容	含义
0	0x55	包头
- 23 -		
电话: 0755-33185882 邮箱: wit@wit-motion.com 网站: www.wit-motion.com		

 **维持智能**
www.wit-motion.com

1	0x53	标识这个包是角度包
2	RollL	X 轴角度低字节
3	RollH	X 轴角度高字节
4	PitchL	Y 轴角度低字节
5	PitchH	Y 轴角度高字节
6	YawL	Z 轴角度低字节
7	YawH	Z 轴角度高字节
8	TL	温度低字节
9	TH	温度高字节
10	Sum	校验和

角速度计算公式：
滚转角（x 轴）Roll=((RollH<<8)|RollL)/32768*180(°)
俯仰角（y 轴）Pitch=((PitchH<<8)|PitchL)/32768*180(°)
偏航角（z 轴）Yaw=((YawH<<8)|YawL)/32768*180(°)
温度计算公式：
T=((TH<<8)|TL)/340+36.53 ℃
校验和：
Sum=0x55+0x53+RollH+RollL+PitchH+PitchL+YawH+YawL+TH+TL

根据这个协议就很好理解我写的数据处理程序。这个JY62串口输出的角度的一共是11位数据。所以，我先判断数据 的长度是不是11位，然后再判断包头是不是55。如果不是55在判断下一位是不是55。直到我找到了55这个包头，然后才进行11位数据的第二位的数据的判断，是不是50、51、52、53、54这种商家定义的代表时间、加速度、角速度、角度、磁场的数。

```
if (ucRxBuffer[0]!=0x55) //数据头不对，则重新开始寻找0x55数据头
{
    ucRxCnt=0;
    return;
}
if (ucRxCnt<11) {return;} //数据不满11个，则返回
else
{
    switch (ucRxBuffer[1]) //判断数据是哪种数据，然后将其拷贝到对应的结构体中，有些数据包需要通过上位机打开对应的输出后，才能接收到这个数据包的数据
    {
        case 0x50: memcpy(&stcTime,&ucRxBuffer[2],8);break; //memcpy为编译器自带的内存拷贝函数，需引用"string.h"，将接收缓冲区的字符拷贝到数据结构体里面，从而实现数据的解析。
        case 0x51: memcpy(&stcAcc,&ucRxBuffer[2],8);break;
        case 0x52: memcpy(&stcGyro,&ucRxBuffer[2],8);break;
        case 0x53: memcpy(&stcAngle,&ucRxBuffer[2],8);break;
        case 0x54: memcpy(&stcMag,&ucRxBuffer[2],8);break;
        case 0x55: memcpy(&stcDStatus,&ucRxBuffer[2],8);break;
        case 0x56: memcpy(&stcPress,&ucRxBuffer[2],8);break;
        case 0x57: memcpy(&stcLonLat,&ucRxBuffer[2],8);break;
        case 0x58: memcpy(&stcGPSV,&ucRxBuffer[2],8);break;
        case 0x59: memcpy(&stcQ,&ucRxBuffer[2],8);break;
    }
    ucRxCnt=0; //清空缓存区
}
}
```

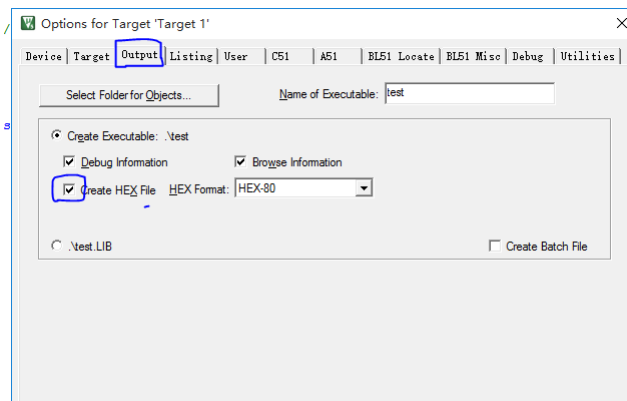
<>3.3 UART打印程序

把上面得到的正确的格式的数据输出到电脑上。

```
printf("Time:20%d-%d-%d %d:%d:%.3f\r\n",stcTime.ucYear,stcTime.ucMonth,stcTime.ucDay,stcTime.ucHour,stcTime.ucMinute,(float)stcTime.ucSecond+(float)stcTime.usMiliSecond/1000);
    delay_ms(10);
    //输出加速度
    //串口接受到的数据已经拷贝到对应的结构体的变量中了，根据说明书的协议，以加速度为例 stcAcc.a[0]/32768*16就是X轴的加速度。
    printf("Acc: %.3f %.3f %.3f\r\n", (float)stcAcc.a[0]/32768*16, (float)stcAcc.a[1]/32768*16, (float)stcAcc.a[2]/32768*16);
    delay_ms(10);
    //输出角速度
    printf("Gyro: %.3f %.3f %.3f\r\n", (float)stcGyro.w[0]/32768*2000, (float)stcGyro.w[1]/32768*2000, (float)stcGyro.w[2]/32768*2000);
    delay_ms(10);
    //输出角度
    printf("Angle: %.3f %.3f %.3f\r\n", (float)stcAngle.Angle[0]/32768*180, (float)stcAngle.Angle[1]/32768*180, (float)stcAngle.Angle[2]/32768*180);
    delay_ms(10);
    //输出磁场
    printf("Mag: %d %d %d\r\n",stcMag.h[0],stcMag.h[1],stcMag.h[2]);
    delay_ms(10);
    //输出气压、高度
    printf("Pressure: %ld Height: %.2f\r\n",stcPress.lPressure, (float)stcPress.lAltitude/100);
    delay_ms(10);
    //输出端口状态
    printf("DStatus: %d %d %d %d\r\n",stcDStatus.sDStatus[0],stcDStatus.sDStatus[1],stcDStatus.sDStatus[2],stcDStatus.sDStatus[3]);
    delay_ms(10);
    //输出经纬度
    printf("Longitude: %ldDeg %.5fm Latitude: %ldDeg %.5fm\r\n",stcLonLat.lLon/10000000, (double) (stcLonLat.lLon % 10000000)/1e5, stcLonLat.lLat/10000000, (double) (stcLonLat.lLat % 10000000)/1e5);
    delay_ms(10);
    //输出地速
    printf("GPSHeight: %.1fm GPSTYaw: %.1fDeg GPSV: %.3fkm/h\r\n", (float)stcGPSV.sGPSHeight/10, (float)stcGPSV.sGPSYaw/10, (float)stcGPSV.lGPSVelocity/1000);
    delay_ms(10);
    //输出四元素
    printf("Four elements: %.5f %.5f %.5f %.5f\r\n\r\n", (float)stcQ.q[0]/32768, (float)stcQ.q[1]/32768, (float)stcQ.q[2]/32768, (float)stcQ.q[3]/32768);
    delay_ms(10); //等待传输完成
```

<>4 生成下载文件

1、在OUTUT勾选Creat HEX File



https://blog.csdn.net/Fred_1986

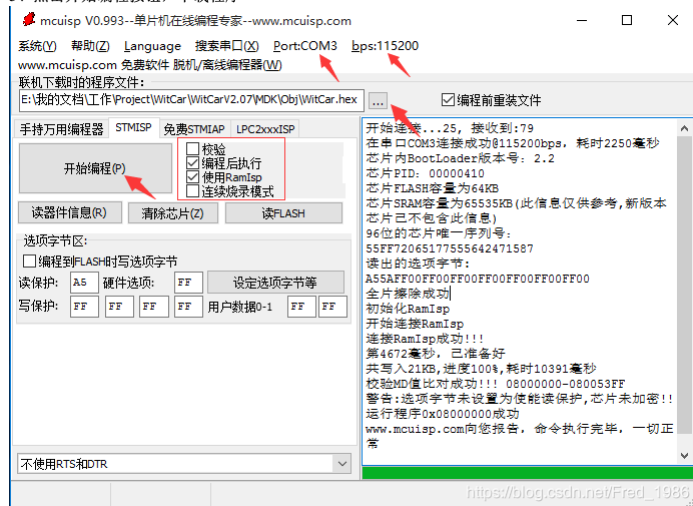
2、找到生成的HEX文件



https://blog.csdn.net/Fred_1986

◇5 下载程序

- 1、用数据线连接好STM32F103C8T6单片机和PC端。
- 2、按照“软件配置.png”图示设置好软件。
- 3、找到所需的hex文件。
- 4、按STM32F103C8T6单片机的红色的Bootloader 键
- 5、点击开始编程按钮，下载程序



https://blog.csdn.net/Fred_1986

◇6 输出结果显示

在电脑上正确连接好板子，首先打开串口调试助手，找到相应的端口，然后打开串口，注意这里波特率设置为9600，然后就可以观察到左边的窗口有数据输出了。如图所示：

