无人机控制程序说明

PSoC版

目录

[一、整体工作流程与工程架构 2](#_Toc528337064)

[1、程序工作流程 2](#_Toc528337065)

[2、工程配置 3](#_Toc528337066)

[3、顶层电路图 4](#_Toc528337067)

[4、引脚配置 4](#_Toc528337068)

[二、JY901模块设置与解码程序 5](#_Toc528337069)

[1、模块安装、配置与校准 5](#_Toc528337070)

[2、串口数据解码 7](#_Toc528337071)

[三、蓝牙接收与解码、遥控App 8](#_Toc528337072)

[1、蓝牙模块安装与调试 8](#_Toc528337073)

[2、App使用方法 8](#_Toc528337074)

[3、蓝牙数据解码 9](#_Toc528337075)

[四、PWM控制 11](#_Toc528337076)

[1、程序中的PWM模块 11](#_Toc528337077)

[2、电调的控制 11](#_Toc528337078)

[五、PID控制算法 12](#_Toc528337079)

[1、控制算法原理 12](#_Toc528337080)

[2、控制程序结构 12](#_Toc528337081)

[六、安全包线及相关设定 13](#_Toc528337082)

[1、期望倾角的限制 13](#_Toc528337083)

[2、摇杆控制动力的限制 13](#_Toc528337084)

[3、旋翼加速的限制 13](#_Toc528337085)

[4、连接中断自动停止 13](#_Toc528337086)

[七、程序运行与调试方法 14](#_Toc528337087)

# 一、整体工作流程与工程架构

## 1、程序工作流程

**（1）main.cpp中各函数说明**

|  |  |
| --- | --- |
| CY\_ISR(Timer\_1\_Interrupt\_Handler) | 中断函数，与PWM相关，详见PWM部分 |
| CY\_ISR(Bluetooth\_RX\_Interrupt\_Handler) | 中断函数，与蓝牙解码相关，详见蓝牙部分 |
| CY\_ISR(JY901\_RX\_Interrupt\_Handler) | 中断函数，与JY901串口解码相关，详见JY901部分 |
| CY\_ISR(JY901\_Debug\_Display) | 输出JY901解码得到的信息，仅供调试 |
| CY\_ISR(Launch\_Control) | 中断函数，与控制算法函数有关，详见控制算法部分 |
| int main() | 主函数 |

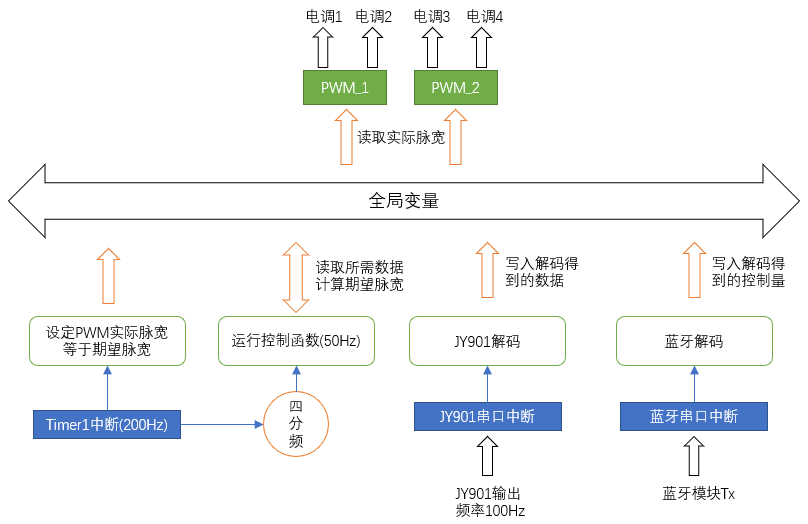
**（2）主函数工作流程说明**

初始化部分，流程为初始化全局中断→初始化LCD显示→初始化定时器1→初始化USBUART→初始化蓝牙串口→初始化JY901串口→初始化控制算法所需变量→初始化电机，使电机停转保持10秒（调试时此时可以接上电池）→电机测试（可选，默认被注释掉）→电机停转

主循环部分，分调试模式和实际运行模式。调试模式下，每0.5秒通过USBUART向上位机发送从JY901得到的姿态、角速度等信息，实际运行模式下，仅在LCD屏上显示当前无人机是受到控制还是强制停止（收不到蓝牙控制信号就会强制停止）。

**（3）整体工作流程**

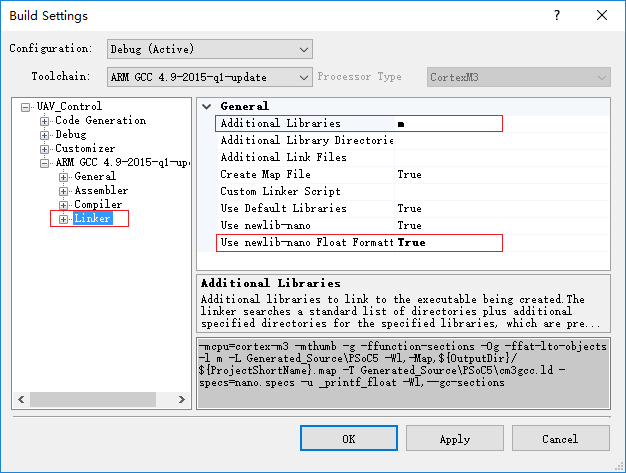
以下流程图是实际运行模式下的工作流程，省略了调试信息输出。调试模式下，仅将控制函数的运行频率降低为2Hz并增加了更多调试信息输出，整体流程类似。



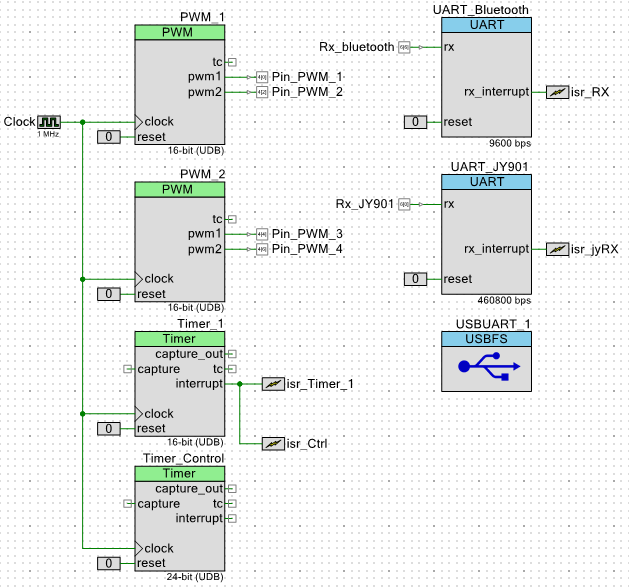
## 2、工程配置

PSoC Creator的工程配置与一般的C或C++语言的IDE有一些区别，本工程总需要使用的浮点数格式化输出和数学库（math.h）需要进行手动配置，否则无法使用。

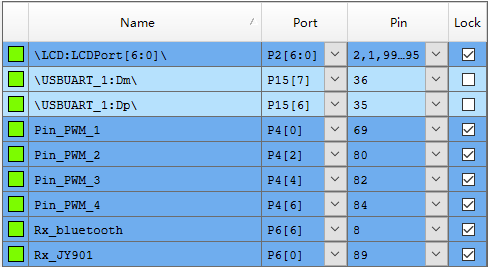
打开菜单栏 Project → Build Settings，展开左边树形目录，展开ARM GCC xxx（型号）项，点击下方的Linker选项，在右侧的Additional Libraries项右边的文本框中填入m，表示使用数学库math.h，否则#include <math.h>会报错。另外，最下方的Use newlib-nano Float Formatting项右侧改为True，这样才能在sprintf函数中使用”%f”格式，否则是无法输出浮点数的。



## 3、顶层电路图



## 4、引脚配置



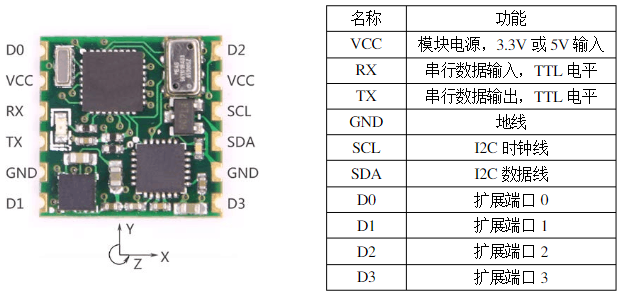
# 二、JY901模块设置与解码程序

无人机使用的JY901惯性导航模块集成了陀螺仪、加速度计、地磁场传感器、微处理器等器件，具有动力学解算与卡尔曼滤波算法，能够通过串口或I2C方式输出时间、加速度、角速度、角度（姿态角）、磁场等数据。具体使用说明及通讯协议细节见

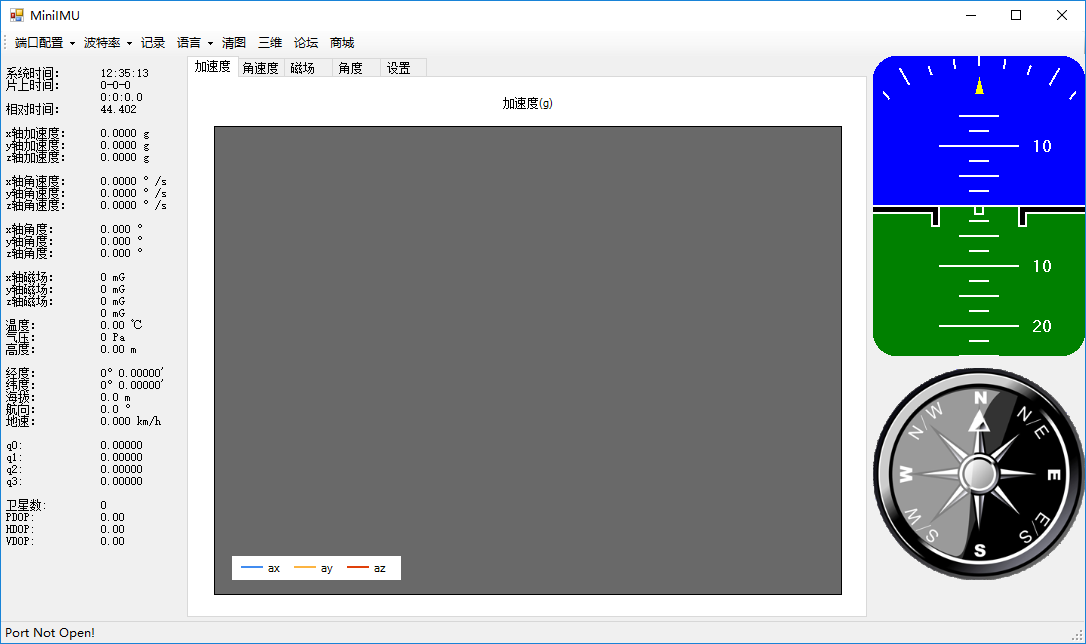
<https://wenku.baidu.com/view/13665ba8b307e87100f69630.html>

## 1、模块安装、配置与校准

因PSoC上I2C配置非常麻烦，在此使用串口方式。JY901的引脚设置和角度约定如右图所示，使用串口通信只需要VCC，RX，TX，GND（TTL电平）。无人机上由于只需接收来自JY901的信息，因此还可以省去RX，即一共只需3根线。



向JY901发送特定格式的信息可以完成配置，但在PSoC上完成这个步骤较为麻烦，因此使用现成的上位机程序完成。将VCC，RX，TX，GND端口连到USB转TTL线上，并连接计算机，运行上位机主程序MiniIMU.exe，程序界面如下。

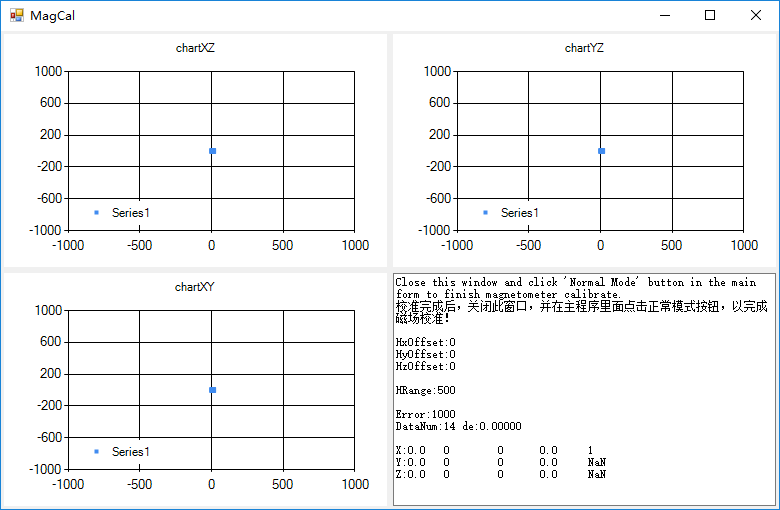


在菜单栏上可以选择端口和波特率。端口在各计算机上都不一样，可以通过设备管理器查看（或一个个尝试），波特率默认是9600。如果发现不行可以短接VCC和D0来重置模块设置。选择正确的配置后即可看到左侧能够显示测量得到的信息，并绘制对应的图表。

打开“设置”选项卡，可以看到如下界面。需要将型号改成JY901（或其他，需要与模块实际型号相符），其中量程、回传内容、波特率、回传速率可以任意设置，设置完成后需点击保存设置。



将模块静止放在水平面上，点击“加计校准”即可开始进行加速度计校准，AxOffset、AyOffset、AzOffset的值会开始变化。等值稳定后，即可认为加速度计校准完毕，点击“正常模型”、“保存设置”即可。



点击“磁场校准”按钮，就会弹出如上图所示的窗口。此时令模块分别绕三个周缓慢旋转一周，当窗口中三个图上都显示完整的椭圆后，磁场校准完成，关闭该窗口，回到主界面点击“正常模式”、“保存设置”即可，此时可以看到主界面中HxOffset、HyOffset、HzOffset已经被设定为了特定的值。旋转时建议将模块用排针固定在面包板上以保持稳定。磁场校准完成后才可以测量z轴角度（Yaw）。

## 2、串口数据解码

在PSoC程序中使用UART模块进行串口通信，只需要Rx端口。串口遵循一定的通讯协议。每个数据帧包含11个字节，以0x55开头，第二位是数据类型，最后一位是校验位，正确情况下等于前10字节的和的最后8位（按位与0xff的结果）。各类数据格式如下

**（1）加速度**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0x55 | 0x51 | AxL | AxH | AyL | AyH | AzL | AzH | TL | TH | SUM |

Ax=((AxH << 8) | AxL) / 32768 \* 16g

Ay=((AyH << 8) | AyL) / 32768 \* 16g

Az=((AzH << 8) | AzL) / 32768 \* 16g

T=((TH << 8) / TL) / 100℃

T是温度，g是重力加速度，16g为量程，是可调的

**（2）角速度**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0x55 | 0x52 | WxL | WxH | WyL | WyH | WzL | WzH | TL | TH | SUM |

Wx=((WxH << 8) | WxL) / 32768 \* 2000(°/s)

Wy=((WyH << 8) | WyL) / 32768 \* 2000(°/s)

Wz=((WzH << 8) | WzL) / 32768 \* 2000(°/s)

T=((TH << 8) / TL) / 100℃

2000°/s为量程，是可调的

**（3）角度（姿态角）**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0x55 | 0x53 | RollL | RollH | PichL | PitchH | YawL | YawH | TL | TH | SUM |

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°

T=((TH << 8) / TL) / 100℃

**（4）磁场**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0x55 | 0x54 | HxL | HxH | HyL | HyH | HzL | HzH | TL | TH | SUM |

x轴磁场Hx=((HxH << 8) | HxL)

y轴磁场Hy=((HyH << 8) | HyL)

x轴磁场Hz=((HzH << 8) | HzL)

T=((TH << 8) / TL) / 100℃

运算时需要注意正负号的处理，两字节数据的类型应当为signed short。

JY901还能够输出其他数据，但暂时用不到，在此不再说明，具体见说明文档。具体解码代码见程序，其他JY系列的模块使用方式类似，基本的数据解码程序可以通用。

# 三、蓝牙接收与解码、遥控App

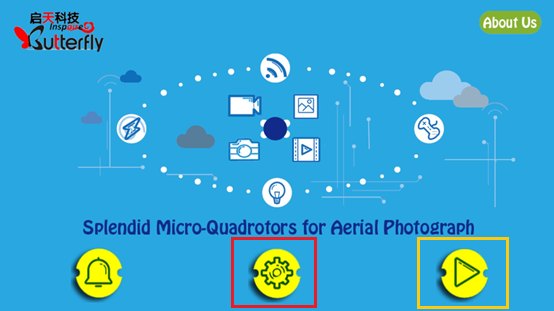
## 1、蓝牙模块安装与调试

蓝牙模块使用串口通讯，连线方式同JY901。由于无人机只负责接收数据，因此蓝牙模块只需要3根线，VCC、GND、TX（TTL电平）。其中TX需要接到PSoC上设置好的接收引脚上，在程序中使用UART模块进行串口通讯。

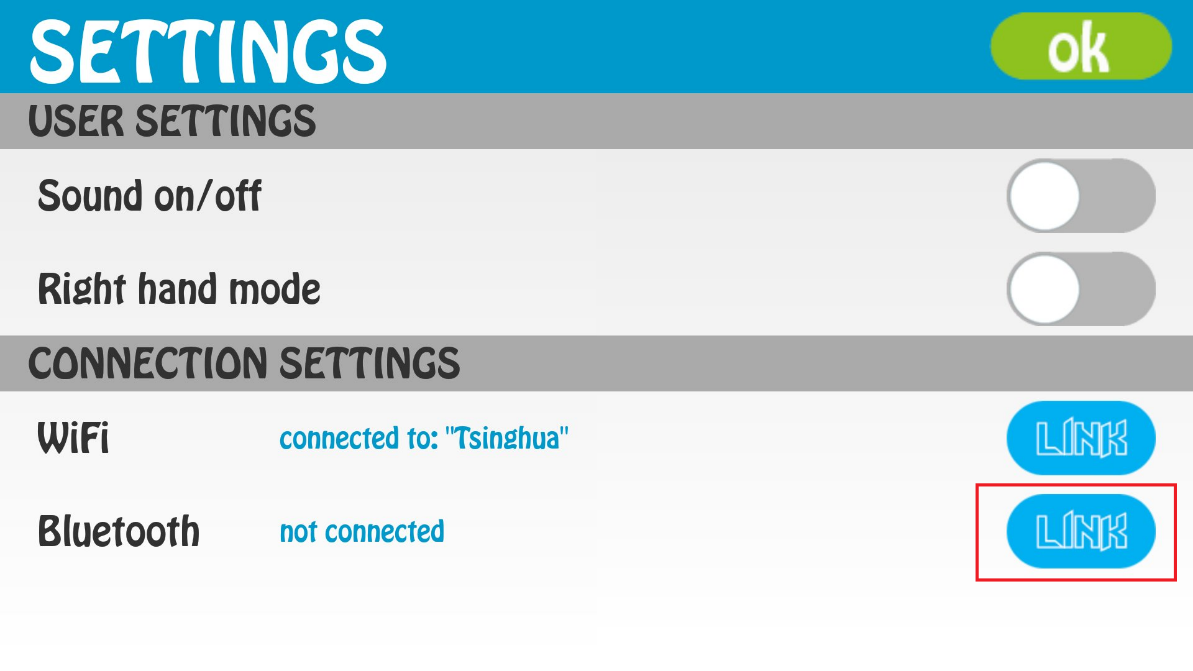
蓝牙模块通电后，即可被手机或笔记本搜索到。蓝牙模块可以被替换，且对应的名称和PIN码都可以修改。在开始使用前需要确保手机能够稳定地连接蓝牙模块。

## 2、App使用方法

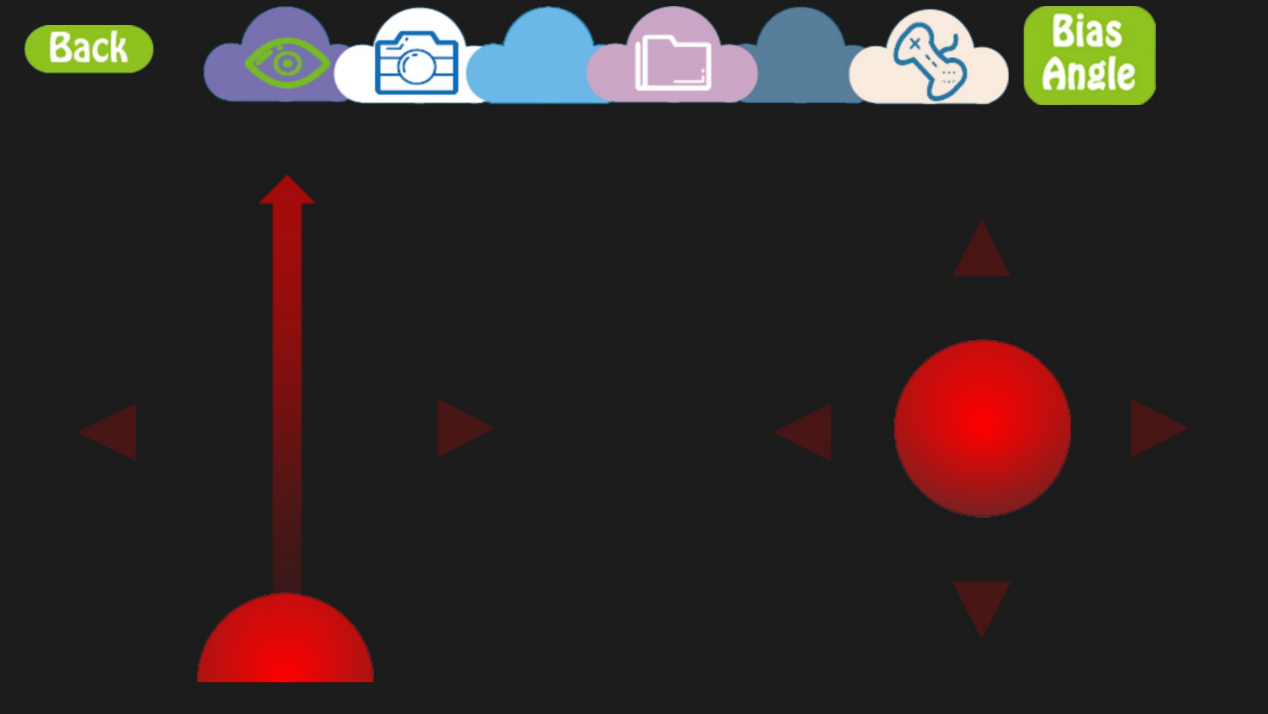
无人机使用的是BUTTERFLY微型无人机的遥控程序。在打开App前，需要确保手机已经与蓝牙模块配对并连接。打开App，即可看到以下界面：



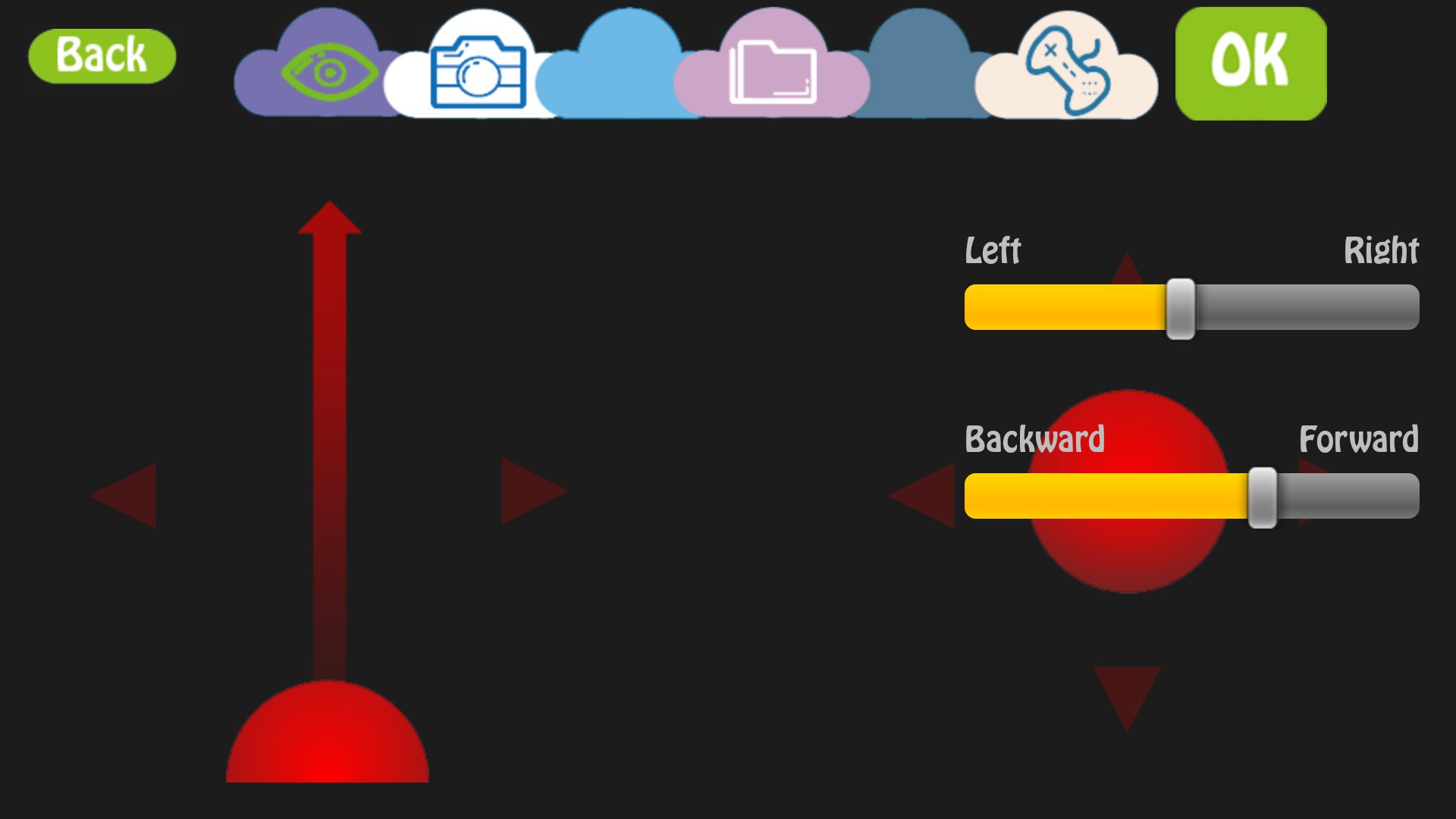
点击红框内的按钮进入设置界面，点击黄框内的按钮进入操作界面。设置界面如下图所示



点击右下角的LINK（红框中），连接已经与手机配对的蓝牙模块。成功连接后即可进入操作界面，如下图所示



左侧摇杆前后推控制油门大小，左右移动控制无人机自转，右侧摇杆控制无人机前后左右移动（倾斜）。右上角的Bias Angle可以加入角度补偿，经手动调节后可以有效地补偿陀螺仪的角度漂移，调整界面如下图所示：



## 3、蓝牙数据解码

连接成功并开始控制后，手机App会循环给蓝牙发送数据，一个数据帧包括15个字节，其含义如下表所示

|  |  |
| --- | --- |
| **位数** | **含义** |
| 1 | 起始位，0xA5 |
| 2 | 无效位 |
| 3 | 油门大小，低位 |
| 4 | 油门大小，高位 |
| 5 | 目标偏航角速度，低位 |
| 6 | 目标偏航角速度，高位 |
| 7 | 目标滚转角，低位 |
| 8 | 目标滚转角，高位 |
| 9 | 目标俯仰角，低位 |
| 10 | 目标俯仰角，高位 |
| 11 | 无效位 |
| 12 | 俯仰角偏置 |
| 13 | 滚转角偏置 |
| 14 | 校验位 |
| 15 | 无效位 |

经高低位拼接后，其中油门大小的范围是0-1024，0为最小，1024为最大；偏航角速度的范围是0-1024，摇杆在中间是512，偏左是0-512，偏右是512-1024；目标滚转角、俯仰角的范围均为0-1024，摇杆在中间是512，摇杆偏左、偏后是0-512，偏右、偏前是512-1024。

至于各个数字对应多少的角度可以自行设置，范围大则反应更灵敏，但容易过度调整，需要经过试验并找到平衡。具体的解码程序见源码。

# 四、PWM控制

## 1、程序中的PWM模块

在顶层电路图中使用了两个PWM模块，通过WriteCompare函数即可设置脉宽。程序中已经写好了调节的函数，输入数值0-1000对应电机的停止和最大转速（在之后会说明）。

程序中使用了同步更新的方法，即各种算法计算得到的期望电机速度储存在一组变量中，通过一个由定时器驱动的函数来不断调用WriteCompare函数，当发现设定速度有变化时就将其更新。

## 2、电调的控制

输入PWM波至电调，可以控制电机的转速，转速由脉宽，即高电平的持续时间决定。0.5ms对应停止，2.5ms对应最高速度，输出的频率应控制在400Hz以内。

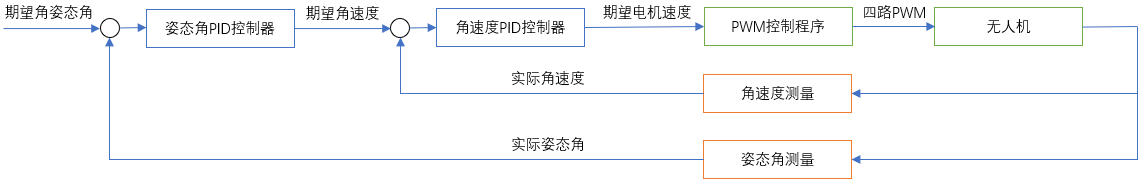
电调校准的步骤为：先输入0.5ms脉宽，保持一小段时间后输入2.5ms脉宽，再保持一小段时间后回到0.5ms脉宽。电调上电时需要确保输入的是0.5ms脉宽（会发出嘀的一声，之后就可以正常使用了），否则电调无法工作且会持续发出响声。

# 五、PID控制算法

## 1、控制算法原理

控制程序主要控制滚转角Roll、俯仰角Pitch和偏航角速度AccelZ（因为偏航角Yaw的测量精度相对较差）尽量跟随去期望值，而期望值由遥控程序给出。

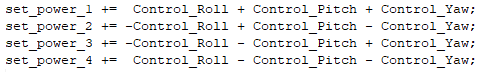
滚转角Roll、俯仰角Pitch由两层闭环控制，系统框图如下



控制偏航角速度AccelZ的系统框图类似，只是只有一个角速度闭环。具体代码见程序。

为控制无人机在空间中稳定悬停，必须有姿态闭环和位置闭环才能够实现，否则陀螺仪模块的漂移无法避免且不能修正。因无人机上还没有安装足够精度的空间位置测量模块，因此目前只能实现姿态闭环。为了控制无人机稳定飞行，可以采用设定合适的角度偏移来抵消陀螺仪的漂移，并使用遥控器手动控制的方法。在测试中逐渐降低遥控器的灵敏度（修改蓝牙解码），即可实现越来越精细和稳定的控制。

本程序中使用的是X形控制，无人机的前方设定为PSoC板上安装LCD屏的位置。由于一个方向的动作需要操控不止一个电机，例如向前倾斜（俯）需要后两个电机（3，4号）加速，前两个电机（1，2号）减速，因此PID算法先生成三个轴上的控制量Control\_Roll、Control\_Pitch、Control\_Yaw，最后根据正负号和它们对各电机转速的影响来调整电机转速，代码片段如下图所示。



另外，由于旋翼产生的升力与转速的平方成正比，因此在程序中首先计算的是期望的动力，之后用开方加比例的方式换算成旋翼转速。这样可以使控制更加准确，但另一方面也使得摇杆操控有些不便。经测试达到约60%转速时无人机可以起飞，但换算成动力仅有36%，因此改成动力控制后摇杆推到约36%无人机就会起飞，控制时需小心。

## 2、控制程序结构

接收陀螺仪信息的函数和负责控制的函数（根据目前的姿态、角速度等信息给出期望的电机转速）是相互独立的，分别由不同的中断控制。两个函数之间使用一组全局变量来传递信息，各变量的功能说明见程序代码。

控制的准确度和灵敏度受到控制算法更新的频率影响，建议陀螺仪传输数据的频率不低于算法更新的频率，算法更新频率在运算流畅的前提下尽可能提高，实际飞行时需要至少20Hz。

# 六、安全包线及相关设定

## 1、期望倾角的限制

由于无人机目前没有位置闭环，且控制算法更新频率低（不超过200Hz），而无人机系统是不稳定且具有强非线性的，因此在大角度倾斜的情况下会难以操控，需要对期望的倾斜角进行一定的限制。

蓝牙解码程序得到的数据为0-1024，其对应的最大期望角是可以自行设定的，一般设置为20-30°。另外，由于控制程序有两层闭环，内部的角速度闭环输入的期望角速度也可以进行限制。

## 2、摇杆控制动力的限制

在操控无人机的姿态时，每个轴上角度的控制都需要两个电机加速、另外两个电机减速。因此油门推到顶时电机不能达到最大转速，必须留出余量来控制无人机的姿态。程序设定油门推到顶时对应90%的动力，这个数值可以更改。无人机到40%动力即可起飞。

## 3、旋翼加速的限制

无人机上安装的电机动力非常强大，如果因程序的Bug等原因导致一个或两个电机的转速突然达到最大，无人机会瞬间翻身，不利于控制，也增加了危险性。因此在程序中加入了旋翼加速的限制，即每次算法更新，动力相比于上一回合更新时的增量是有上限的，这样就能够使得动力相对慢地增加。相反，动力减小的速度是不受限制的，使得无人机可以在紧急情况下瞬间停下来。

如果修改了控制算法的更新频率，每回合动力增加的上限也需要相应地更改，以保证安全性。

## 4、连接中断自动停止

遥控程序将会通过蓝牙不断给无人机发送控制数据，具体实现方法为，设置一全局变量，在每回合算法更新时将其加1，而收到蓝牙发来的数据时就将该变量清零。如果该变量的值达到一定的大小，说明过长时间没有收到蓝牙发来的信息，即可认为连接中断，强制控制所有电机停转。

如果修改了控制算法的更新频率，连接中断后等待的回合数也需要相应地更改，使得中断连接后能够及时停下来，而有连接时也不至于因更新频率太快导致误判断连接中断。

# 七、程序运行与调试方法

对于调试模式，将变量Debug\_Mode设为1，用数据线连接PSoC和计算机，烧写程序后运行。程序初始化完成后会有一个10秒的等待时间（时间可以更改），此时LCD屏上显示“Start”。如果需要启动电机，需要在这10秒之内将电池接入。开始运行后，LCD屏上会显示调试信息，默认是第一行显示3个轴向的控制量，第二行显示设定的动力（0-1000）。如果没有与手机App连接，则设定的动力全为零。在调试模式下控制算法函数的更新频率不能过快，否则LCD屏的显示会出现问题。

对于实际运行的模型，需要将变量Debug\_Mode设为0，在烧写程序后将PSoC与计算机断开，使用无人机自带的电源（与电池相连的电源转换模块）连接PSoC的电源和地引脚为开发板供电。此时开机初始化后仅等待一秒即可开始控制。

目前无人机仅能使用蓝牙控制，蓝牙信号可来自遥控App或者使用同种通讯协议的上位机，具体格式见之前说明。在连接断开后所有电机自动停转且无法控制。