频谱侦测代码分析

作者：滑国青

历史：2023.09.04 v0.1 未完成

# 前言

本文代码分析，是以tracer Rtos版本为准。

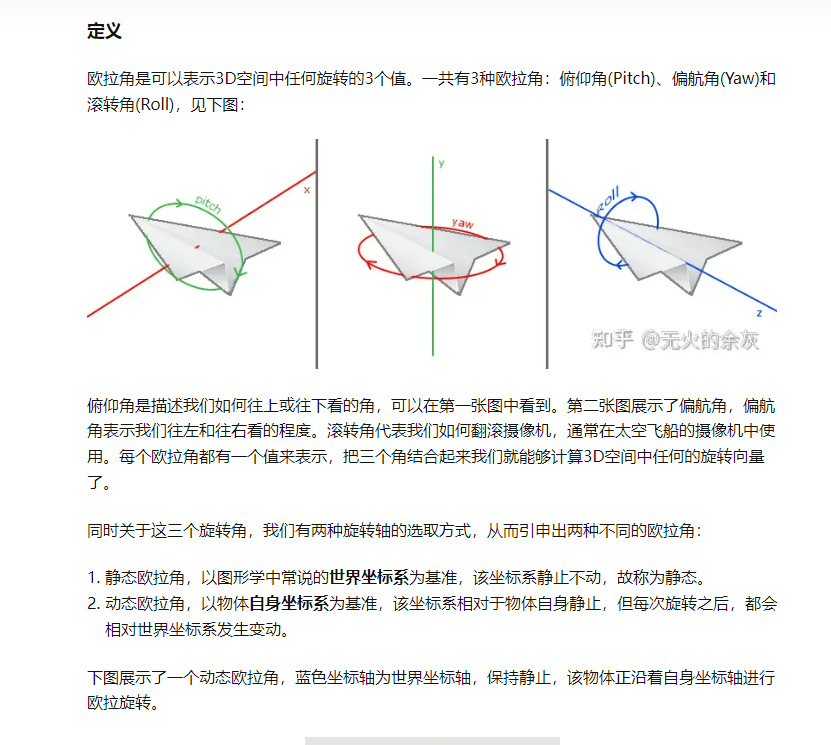
# 概念

## 欧拉角

float roll; float pitch; float yaw;

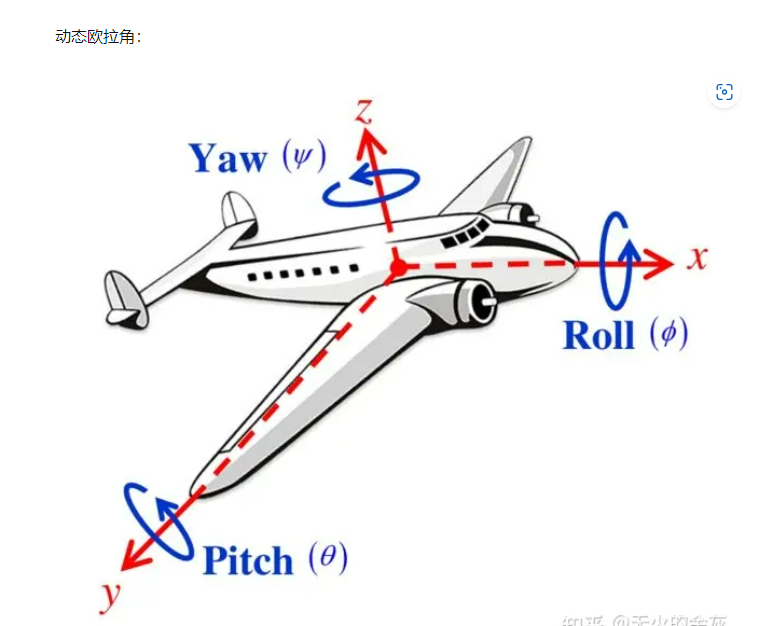
这是定义了三个浮点型变量 roll、pitch 和 yaw 的代码段。这些变量可以用来存储姿态角的值，例如在飞行器或机器人控制中常用的滚转角、俯仰角和偏航角。你可以根据需要使用这些变量来进行相关的计算和操作。

一共有3种**欧拉角**：俯仰角 (Pitch)、偏航角 (Yaw)和滚转角 (Roll)，见下图： 俯仰角是描述我们如何往上或往下看的角，可以在第一张图中看到。 第二张图展示了偏航角，偏航角表示我们往左和往右看的程度。 滚转角代表我们如何翻滚摄像机，通常在太空飞船的摄像机中使用。 每个欧拉角都有一个值来表示，把三个角结合起来我们就能够计算3D空间中任何的旋转向量了。



## 空间参考系统：方位角（azimuth）/仰角（elevation）





# 数据接收

## data\_path.c

01\_code\shrd100\_app\src\app\data\_path

### RDMAPDataCallback

从DMA收一个分片数据。

一条完整消息3MB

（3000\*128）点 \* 2路 \*（4字节每点）

(全向侦测图传（3000\*128）点\*4+定向侦测图传（3000\*128）\*4)字节

2路是指全向和定向各一组数据。

一条完整消息被分成60个分片来发送。一个分片是一次DMA数据传输。

每个分片的前一半是全向数据（来自全向天线），后一半是定向数据（来自定向天线）

即每个分片有点数：(3000\*128)/60 = 6400点。

数据是以点来计单位的，一个点是4字节（32位的float）

#define POINT\_PACK\_NUM 2 表示2路数据

#define DMA\_PACK\_NUM 60 表示一条消息分成60分片

#define NET\_MAX\_NUM 30

#define LINE\_NUM 3000 一条消息用3维数组保存 [2路][3000行][128列] ，由于被分成60分片，所以相当于一个分片50行\*128点

#define COLUMN\_NUM 128

#define REV\_DET\_PACK\_LEN (COLUMN\_NUM \* LINE\_NUM \* 4 \* POINT\_PACK\_NUM) 这就是一个消息的buf大小

#define DET\_DMA\_PACKET\_LEN (50 \* 128 \* 4 \* 2) 这就是一个分片的数据长度

#define FFT\_DMA\_PACKET\_LEN (8 \* 2048) 这是droneid共享内存收数据的数据长度。

//#define DET\_DMA\_PACKET\_LEN (6248\*8)

#define MAT\_PACKET\_LEN DET\_DMA\_PACKET\_LEN / 2 这是一路的数据长度，是一个分片数据长度的一半。

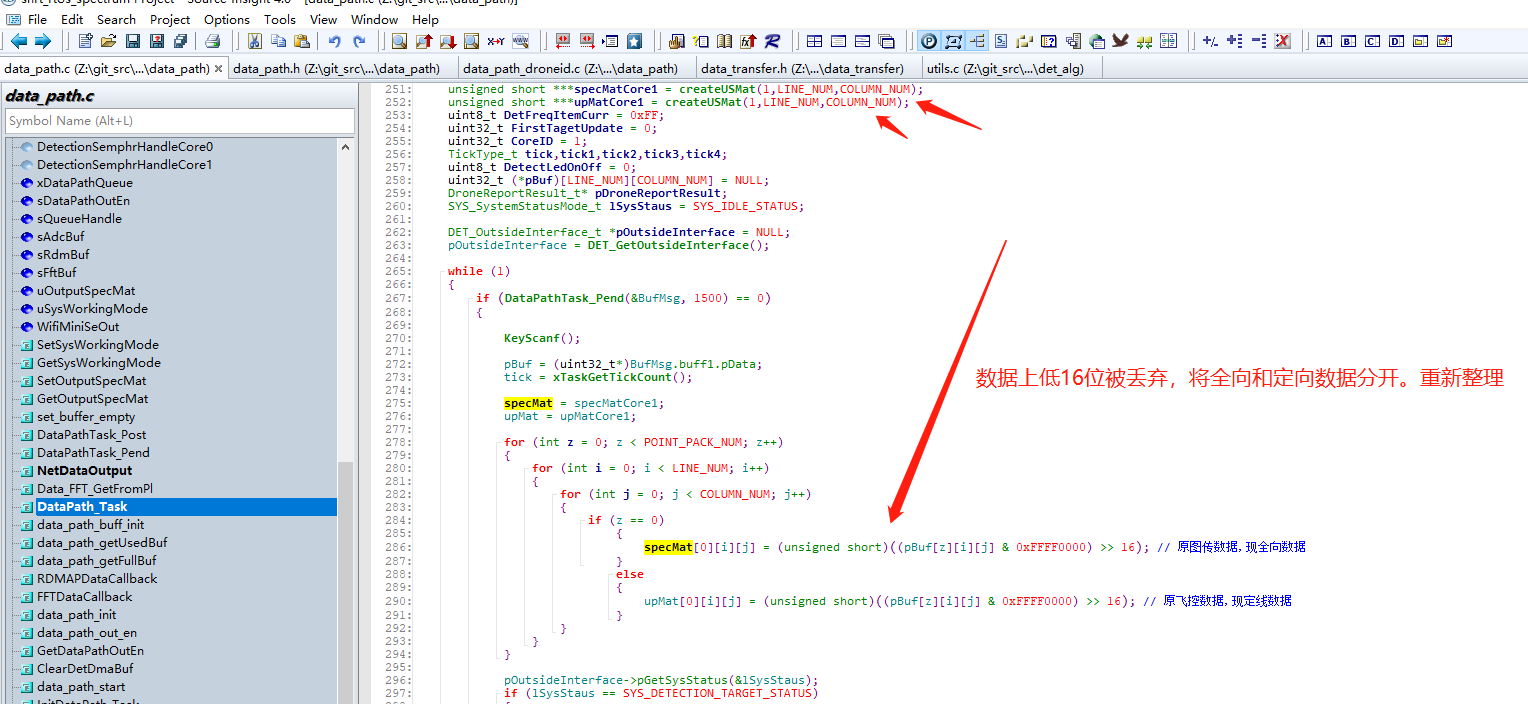
### DataPath\_Task

#### 接收消息并重整数据送算法

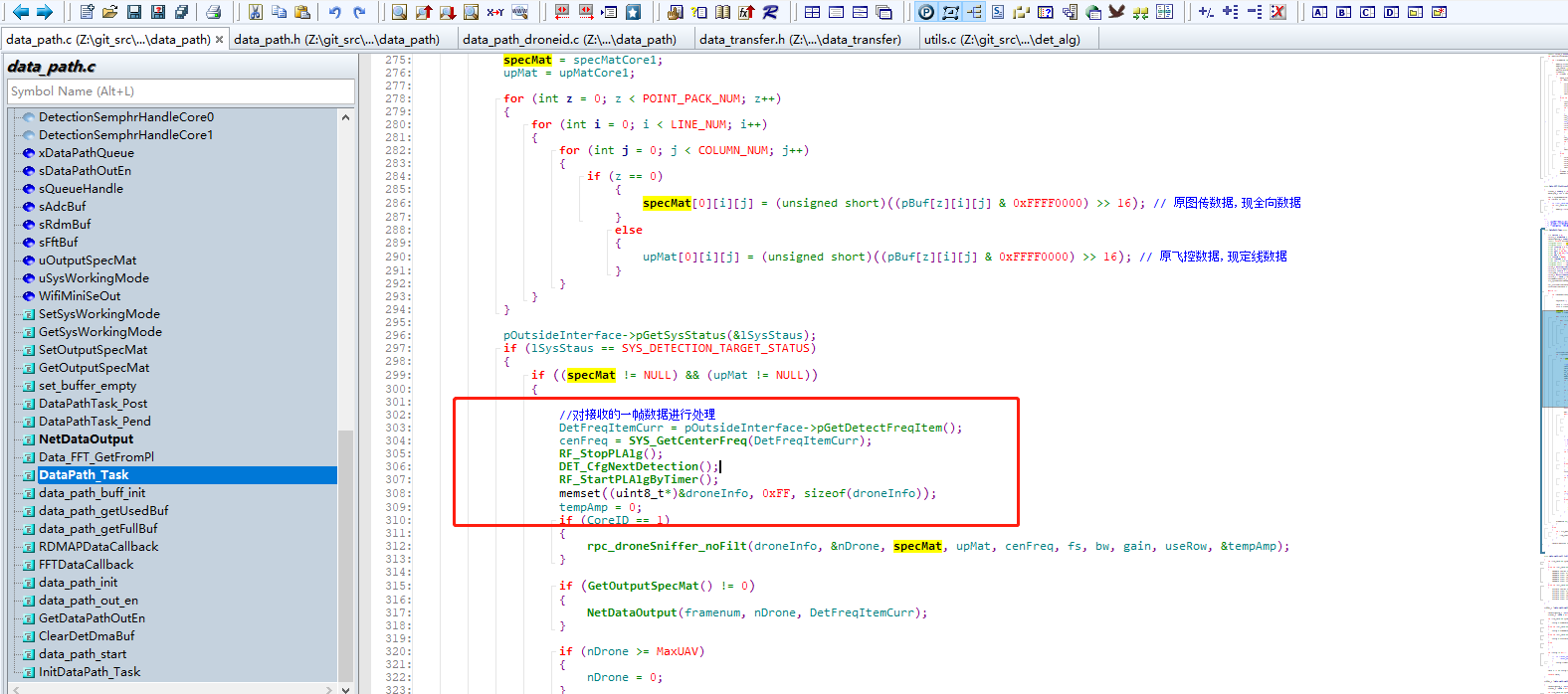
消息组包（60分片）完成后，收完整的消息后的处理。

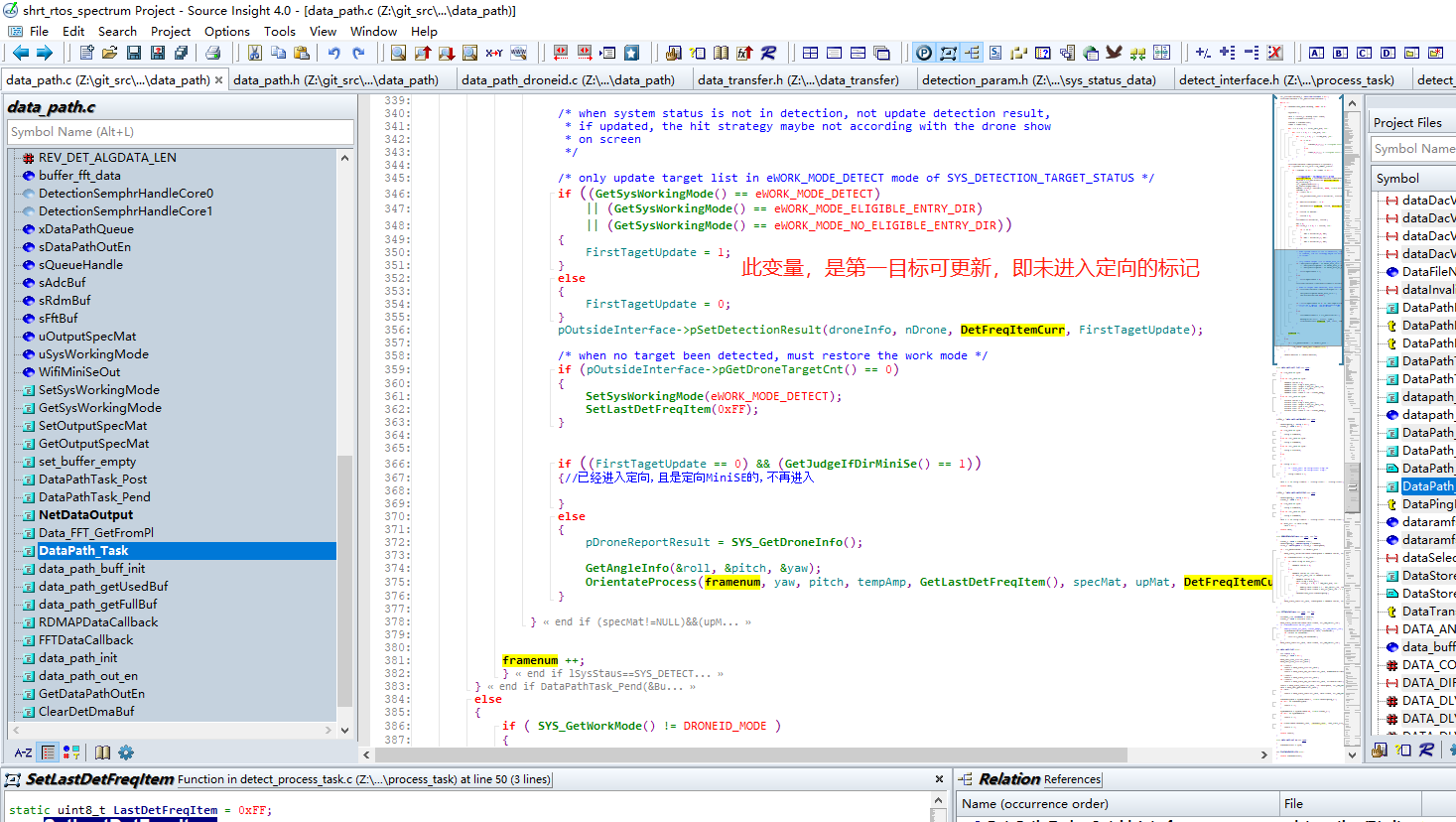
1个点数据32位，实际上只高16位有效，低16位被丢弃，所以当前实际上一个点数据是2个字节有效。

将全向和定向数据分离开，将低16位去掉



#### 切频





# 结束