1. 滤波和不滤波的差别
2. 相关性如何表现
3. 滤波参数的确定
4. 滤波合适稳定，如何去截取

程序将同时分别发送滤波和未滤波的z轴角速度和温度，以保证实时性，从而更准确地说明两种处理方法的影响。另外要注意的是，卡尔曼滤波的R参数根据之前测的方差，将z轴卡尔曼参数的R设为0.0002（之前0.0012），温度则为0.9. 将z轴滤波函数的P\_mid参数发出，结论是现在的P\_mid较之前收敛更快。

实验一：图像分析

滤波与未滤波的z角速度：



滤波与未滤波的温度：



结论：

1.通过滤波，z轴角速度和温度值变得连续。

2.温度的幅值减小。

实验二：各自分析统计参数

由于温度一直有变化，所以分析角速度均值和方差都无意义。实际上，两者均值相差无几。

现将未滤波的角速度和温度分成一组，另外将滤波的分为一组进行温度零飘表构建。

从表的数据来看，由于未滤波数据为ADC采集的数字信号，滤波信号则为数学公式得出，因此滤波后建的表远远大于未滤波的。下图红色为滤波后的。



除了开始的那一点，可以看出为滤波得出的表震荡比滤波之后的表大得多。

至于开始为什么那么抖，我们从原始数据

可以看出开始的10000（50秒）个点是抖动很大的。去除之后重新拟合



这就比较准了

为了统一标准，避免表的温度间隔不一，我们采用0.01℃的步长建表，如下



0.1℃步长，比较符合实际情况



从此图就可以看出，为什么滤波与否的结构相差不大了。因为当0.1℃步长时，两者相差也不大了。

实验三：衡量

之所以建立温度零飘表是因为温度和角速度具有一定的相关关系。现在，我们求一下四组对应的相关系数。（虽然有两组是滤波和未滤波混搭，理论上没有意义）

之前是直接用现有的数据，减去温度零飘表的数之后再积分，看哪个角度偏移小。这是没有道理的。一是，“现有的数据”要和建立温度零飘表对应起来，如果是滤波之后建，“现有的数据”就是指滤波之后的数据。二是，像一那样对应起来之后，再用“现有的数据”检验就失去了可靠性，因为被检验物正是由“现有数据”得出的。

正确应该写入程序中，用实际去检验。但是这又涉及到一个问题，使用滤波之后建立的温度零飘表时，是在滤波前减去，还是滤波后减去。从公式上来说，卡尔曼滤波并不对偏置做处理，因此在滤波前减去还是滤波后，都是可以的。当然最保险的方法是，在用滤波之后的温度零飘表时，对角速度不进行滤波，只对温度进行滤波（为了找到表中相应的数据）。这样角速度的噪声都是一样的了，而偏置大小就看哪个温度零飘表准一点。

当然为了保持变量统一，我们也不要减去上电十秒钟的误差的平均数了。

最后还要强调，一定要同时测量两个表的结果，保证实时性，可比较性。

插曲：发现前面一些点不稳定，只出现一次。不能发现原因。（之前考虑过的，刚写完寄存器那会儿读数会不会异常，第一个数好像一直不对，不知道是不是发数问题）



再多等一些时间看还有没有这种情况

大概第400个点稳定

