新买的IMU的特性总结下大概分为这几点：

1. 静态精度高：IMU内置结算算法，可以直接输出当前角度，将IMU旋转50圈，原始数据的误差大概在十几度左右，经过代码处理后的数据大概能优化到2-3度左右，也就是转一圈偏大概0.05度左右，个人认为精度还是相当可以的，而且回程误差大概在1度左右甚至更小，最起码拿IMU重新标陀螺仪是可行的。
2. 漂移很大：IMU静态是有着不小的漂移的，而且这个漂移并不是固定的，而是一个与时间有关的且随机的量，之前测试将陀螺仪静置，大概3分钟后角度将由0度漂移到1度左右，不过后面优化了算法后好了不少。此外，当陀螺仪开机时有小概率产生连续且极大的漂移，角度将以肉眼可见的速度不断增加，目前程序上并没有就这一点做处理，主要是这种情况概率比较小，很难测试，如果要解决，可以在主控启动的时候加一个报警程序，比如启动的前3秒内如果角度超过5度就报警，然后重启电源
3. 误差来源很多：IMU的误差来源主要有：转过的角度累计、时间、转动速度、转动方向。其中比较好矫正的是转动角度累计和转动方向，这个可以通过标定来解决大部分误差。转动速度也会对角度测量有一定影响，不过并不多，其实要是要解决的话也可以通过标定来修正，但问题主要在于标定时角速度变化比较大，不好观测，而且考虑角速度的影响会大大加剧标定的工作量。至于时间，这个是最玄学的，前面提到IMU的漂移和时间有关，此外，由角度累计映射到角度修正值的系数也会随时间变化，举个例子，假如上电后立马自转50圈，IMU的漂移可能只有8-9度左右，然而如果上电后5分钟重启单片机再自转的话（这里提到重启单片机是为了通过程序消除这五分钟的角度漂移），那么误差就有可能来到12-13度左右，这就很玄学了，其实严格来讲，这个也可以通过标定来解决，但这样一是会导致程序复杂度大大增加，同时标定的工作量也会极大的增加。
4. 动态精度差：这里的动态精度是指旋转时的角速度测量值。目前测量时IMU的动态精度并不高，定角速度旋转过程中角速度变化会比较大，这也导致IMU采用与光纤陀螺仪相同的角速度积分算法时精度并不高，正因此我才会改用IMU的原始角度数据，不过我也怀疑角速度变化大也可能是标定架的elmo角速度不稳定，因为在标定过程中我发现elmo上位机中显示的角速度也不是固定不变的，这点存疑。

接下来是标定的流程，其实这个到不难，大致流程就是让IMU在标定架上正向和反向自转5、10、15……50圈，记录误差然后和角度累计做一下拟合，我选择了简单的线性拟合，原因有两点：1.下面是一次拟合和三次拟合结果对比，可以发现三次项和二次项的数量级已经非常非常小了，三次和一次之间差距并不大，而五位数的乘方运算显然对单片机来说十个不小的负担，此外，前面也说了，误差影响的因素非常多，而且也很难定量描述，所以没必要追求公式的严谨性，差不多就行了。2.其实在代码中我并不是将角度积分带入多项式求解误差修正值的，而是先将角度变化量进行修正之后再积分，之所以这样做，是因为这种算法会更好的考虑到陀螺仪的正反转，求出来的精度更高，而一次多项式相较于二次三次式来说计算步骤会简化很多

最后标定出来的正反误差系数再Imu.h里修改就行了，改完之后还是要再调一调，看看误差能不能更小

2024.10.15：

经过两周的优化，IMU的大部分性能已经被优化的很完全了，静态性能薄纱了老陀螺仪，但有一个很致命的缺点，就是IMU的数据相较于陀螺仪有着大约150ms的滞后，表现在角度上则是同一时刻IMU的数据较陀螺仪能差十几二十度，为了修正这一点，我加了基于陀螺仪历史数据的修正算法，但动态精度只能维持在0-2度左右，此外，关于陀螺仪数据标定，后面我又尝试标了一次，效果很差，不如直接像pid调参一样调两个修正参数。。。

剧冠男

2024．10 . 5