无刷直流电机无感矢量控制观测器算法及软件开发

2025 年 第 3 周 周报

日期范围：2025 3 月 17 日到 3 月 22 日

报告人：陈炫润

一、项目进展（具体情况，可以以文字和图片详细说明）

项目/任务（对照上周或计划任务）	进度	关键指标	完成度
将磁传感器测量电角度设置为对比组，验证滑模观测器和锁相环观测器所组成的无位置观测算法的性能。	1. 在罗工的帮助下，通过缓冲数组的方式，实现了特定条件下，同时打印估算电角度和测量电角度。 2. 合理选择数据，使用 excel 整理 Raw Data, 对其进行处理，绘制曲线。	①从 570rpm 切换至 95rpm 运行, 观测器收敛时，测得误差均在 5°～8°之间。	95%
根据 DengFOC 源码，移植成 C 语言库。	1. 观看 DengFOC 全套视频，对 DengFOC 实现的无感 FOC 流程每一个算法细节有了更深刻的理解。 2. 上手，移植修改代码（进行中）。	1. 采用 SVPWM 实现，相较于 SPWM 做功效率提升 15%。 2. 采用 PLL 观测器对电角度进行滤波，相比于反正切法耗时更少，减少了电机运行时的毛刺带来的影响。	70%

项目/任务（对照上周或计划任务）	进度	关键指标	完成度
撰写论文初稿	建立在实验数据以及DengFOC&知乎大佬深入浅出得讲解算法细节原理代码的基础上，牢牢把握论文核心，修改框架。	/	80%

展开说明：

前言注：由于前几天验证估算电角度和测量电角度时，并不真正完全理解DengFOC 的无感 FOC 运行流程，也没有注意到 readme.md 的文档，当时实验做出来效果最好的数据只有一次，之后再也运行不出来比较好的数据效果。如果后续，需要增加新的数据，明确以下几点：

1.基于反电动势来观测电角度、角速度的观测器算法，特别是传统的滑模观测器算法，在低速运行时，因为电动势很小，所以观测的效果很差。

2.如果出现堵转，立即停止本次实验记录，因为堵住后容易导致估算角度发散而无法闭环。或者等效为电机速度太小，估算亦不准确。

3.所以低速时，不能直接让观测器直接估算。总的来说有两大常见的解决方案，一种是高频注入法，另一种是根据力矩输入与频率成比例控制，分为 V/F 和 I/F 启动。延续了 simplefoc，dengfoc 也采用的是对 Uq 控制的 V/F 开环启动。要等待开环启动完毕，再切换至闭环无感 foc 控制。

4.切换速度之后，有时候会出现堵转，有时候会出现反转，或者是调整速度没有反应，这些问题还有待考究。

二、问题与攻关（具体情况，可以以文字和图片详细说明）

1、技术瓶颈

-数据处理能力：

切换不同 ide 版本、不同 ide 的串口监视器，以及使用串口助手来获取直观的曲线图，每一个办法都有缺陷，最后选择使用回 arduino 的 1.8 老版本。添加时间戳，以供作为曲线图的 x 轴时间；采用同一行打印两个电角度的方式，断电后保存合理的数据。

将获取的数据保存至 excel 时，也需要通过分行分列的处理。这里本来是尝试接触 python 的 pandas 库对那一个单元格获取的数据进行正确的提取，结果发现 python 环境的配置上出现很大的问题。最后，通过手动复制粘贴的方式，完成了数据的处理.....

对于数据的处理能力、方法确实有待进一步提升，但好在 excel 的易上手，绘图也便不是难事。以下是一些效果图、表格：

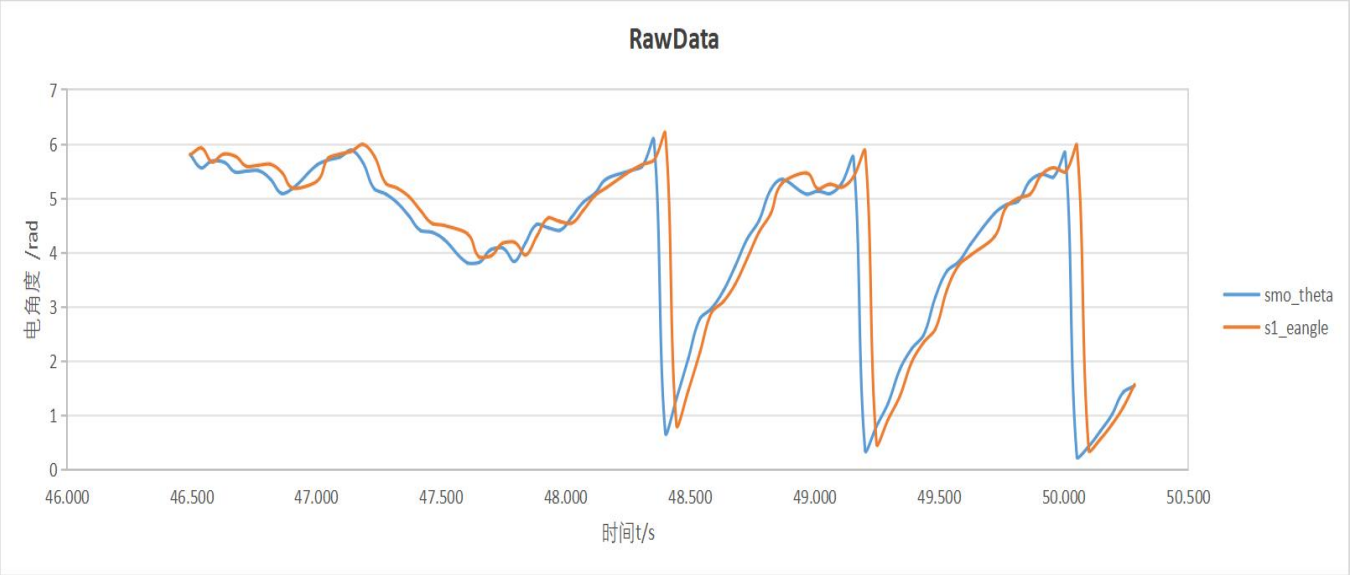


图 1-1

在绘制图 1-1 时，我发现 DengFOC 运行一次 FOC（如果等效于打印间隔）的具体时间均为 43ms~47ms，但是真正运行的计算任务应该是 us 级的。所以 cntt*(1/30000)这个打印间隔与速度应该是有关系的。如果要实现更加好的打印，需要动态调节 cntt，使其与速度关联，这样子也还是非常麻烦。

图 1-1 中存在明显的迟滞，首先这是因为，在速度角度有关的闭环反馈上，都设置了指数衰减型的一阶数字低通滤波器，这引入了一部分延时；而且估算的电角度与磁传感器获取的电角度又存在一定的延迟；而且 PLL 跟踪的目标相位本身就是滞后的，所以 PLL 锁出的相位也是滞后的，需要进行相位补偿。下面是经过处理之后的图 1-2.

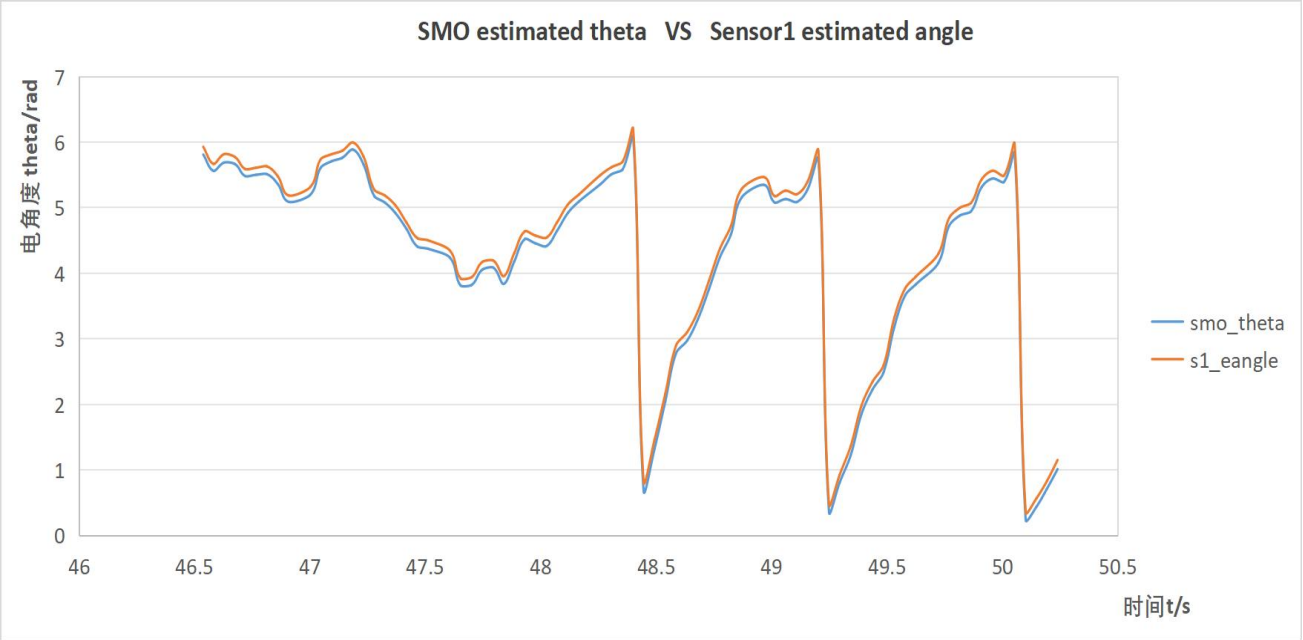
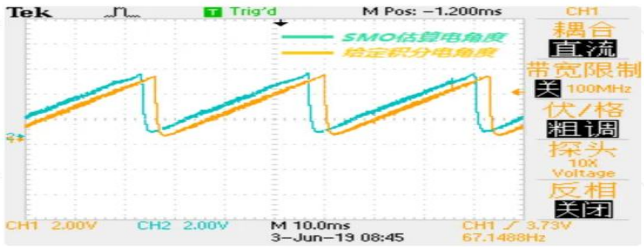


图 1-2

图 1-2 就可以说明，在 10rad/s 及 90rpm 转速下，滑膜观测器结合 PLL 的滤波估算的电角度与实际磁传感器测得的电角度误差并不大，经过计算，平均误差角度在 6.5 度左右，相对误差为 2.3%。由此可见，中高速运行下以及省去打印的误差，实际的无感 FOC 算法效果较好。而根据内部程序规定，DengFOC 验证版的最高转速为 2005rpm，符合需求。

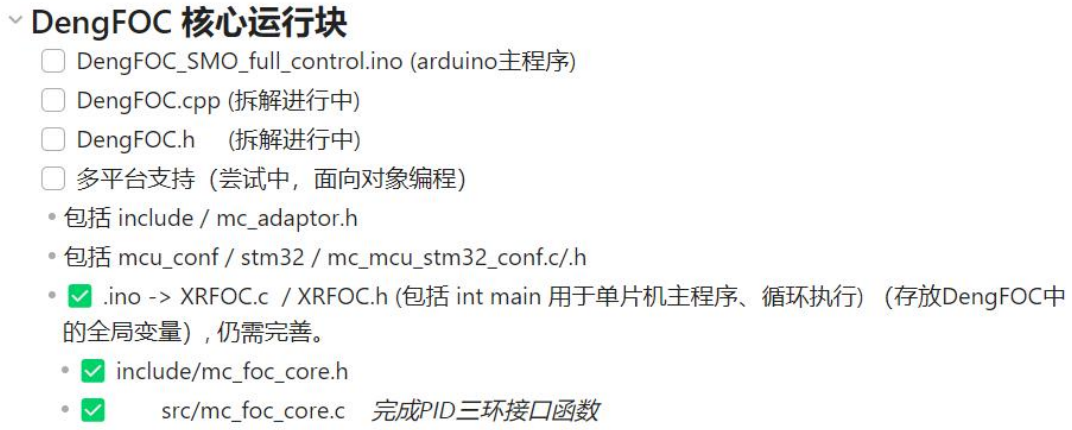
与实际的波形比对也基本吻合。基本是存在较大迟滞。



-编程移植：

就目前来说，移植的还是比较顺利的，除了多平台思想比较棘手抽象之外，所以现在只考虑实现移植成功 DengFoc 对应无感 FOC 必须的算法部分。只需要将 Class 以及类方法，转化为对应 C 语言的结构体，和函数调用接口即可；另一方面，因为 CPP 特性，所以不需要特别注意文件的分层结构，这一点需要考虑；再者，arduino 的运行逻辑与常用的单片机逻辑不同等等，其实移植中还是有一些问题在的，需要斟酌怎么对应两者。

移植进度如下图：



三、下周任务

①继续移植。

②继续整理数据，编写论文。

③继续深入理解 SMO 与 PLL 算法，可以构思 SMO 与 PLL 观测器可以改进的算法部分，比如将饱和函数替代为 sigmoid 函数。