

应用笔记

AN0074

Rev1.0 - 2025-03-18

云途电机调试指南



目录

1	电机控制	3
2	云途无刷电机控制芯片简介	4
2.1	电机控制算法概述	5
2.2	电流采样	5
2.2.1	电流采样的定点化处理	6
2.2.2	电压的定点化处理	6
3	PWM 输出和电流采样	6
3.1	MSPWM 模块简介	6
3.2	单电阻电流采样触发配置	7
3.3	Clarke 变换和 Park 变换	9
3.3.1	Clarke 变换	10
3.3.2	Park 变换	11
4	无刷电机转子位置估算	12
5	编译选项	12
6	电机参数测量	12
6.1	电机相电感	12
6.2	电机相电阻	12
6.3	电机反电动势系数	13
6.4	电机极对数	13
7	电机工程调试步骤	14
7.1	开环验证硬件	14
7.2	电流闭环调整合适的 PI 参数	14
7.3	速度闭环调整合适的 PI 参数	15
8	电机控制程序	15
8.1	电流环控制服务	15
8.2	速度环控制服务	16
8.3	重要的配置参数解释	18
9	电机运行过程中的异常检测	18
9.1	通过硬件保护功能进行异常检测	18
9.2	通过软件进行异常检测	20
10	总结	20
11	参考	20

历史版本

版本号	日期	修改
1.0	2025-03-18	初始版本

1 电机控制

无刷电机和有刷电机在汽车中的应用极为广泛。伴随汽车电子化程度不断提高，电机在汽车里的使用呈现出日益增多的态势，并且这些电机的控制主要依靠 MCU 来达成。

有刷电机通过电刷和换向器来实现电机的换向，控制相对简单。通常使用直流电源供电，通过 PWM 斩波调节电源电压或电流来控制电机的转速和转矩。

无刷电机则采用电子换向器来代替电刷和换向器，控制方法相对复杂。常见的无刷电机控制方法包括：

- 方波控制：通过控制电机的导通和关断时间来实现电机的调速。
- 正弦波控制：通过控制电机的电流和电压的正弦波来实现电机的调速，具有更高的效率和精度。
- 磁场定向控制（FOC）：通过控制电机的磁场方向和大小来实现电机的调速，具有更高的效率和精度。

与有刷电机相比，无刷电机具有以下优点：

- 效率高：无刷电机没有电刷和换向器的摩擦损失，因此效率更高。
- 寿命长：无刷电机没有电刷和换向器的磨损，因此寿命更长。
- 噪音低：无刷电机没有电刷和换向器的摩擦噪音，因此噪音更低。
- 调速范围广：无刷电机可以通过电子换向器实现更广泛的调速范围。

但是，无刷电机的控制方法相对复杂，尤其是现在流行的 FOC 算法，硬件控制需要实现电流环、速度环、位置估算、故障检测等多种计算，这些都是 MCU 的软件算法结合硬件模块实现的，这个过程设计多种控制算法以及调优过程，对于嵌入式开发人员是一个很大的挑战。

如上图所示，无刷电机控制需要生成一个精确可控的旋转磁场，整个控制过程需要通过电流环、速度环、位置估算、故障检测等多种计算，这些都是 MCU 的软件算法结合硬件模块实现的。

为降低电机控制算法的入门难度，云途专门研发了一个参数精简的无刷电机控制算法库。基于此控制库，开发者只需提供部分必要且与硬件相关的参数，就能迅速实现电机控制算法的适配。

2 云途无刷电机控制芯片简介

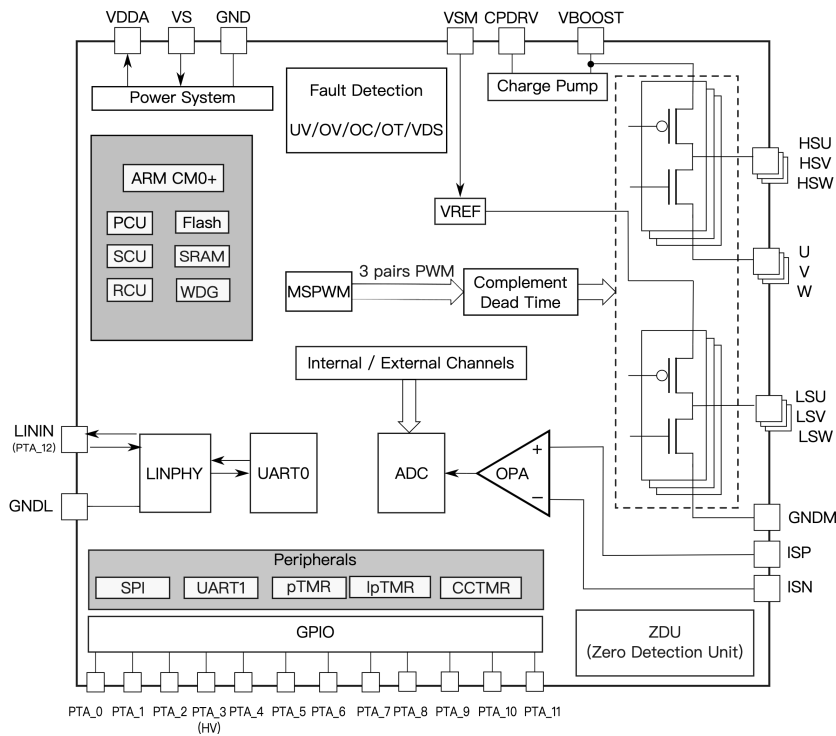


图 1: YTM32Z1MD0x Block Diagram

云途 YTM32Z1MD0x 系列芯片是专为无刷电机应用开发的芯片，芯片内核采用最高运行频率 56MHz 的 Cortex M0+ 内核，内核本身支持硬件 32 位乘法运算，并增加硬件除法和开平方运算，可以满足定点化 FOC 的运算需求。

芯片内部集成高压 LDO，可以实现 12V 电源的直接输入，不需要额外的 LDO 供电。为了满足汽车通信需求，芯片内部集成了 LIN 控制器和 LIN PHY，可以直接连接 LIN 总线。

针对电机控制，芯片内部集成 3 路高边预驱和 3 路低边预驱以及单独的 Boot 升压电路，可以实现 100% 占空比的输出，针对故障检测部分，芯片内部硬件支持过流、过压、欠压、过温以及 VDS 电压异常监控，硬件上可以直接控制电机 PWM 输出，从而实现故障的纯硬件保护。针对电流采样，芯片内部集成一个可编程差分运放，可以实现 4、8、16、32 倍增益调节，更好的满足电机不同运行功率的电流采样需求。

芯片内部 ADC 可以通过 MSPWM 模块设置任意 PWM 位置的触发，可以灵活的实现单电阻电流采样，同时 ADC 模拟通道支持温度、母线电压（内部 12:1 分压），相线电压（内部 12:1 分压）等多种模拟信号输入，可以更为精准的实现电机运行状态的监测。

针对无刷电机的方波控制，芯片内部集成硬件过零检测单元 Zero Detection Unit (ZDU)，并且芯片内部集成虚拟中心点，可以直接检测反电动势过零事件，有效实现 BLDC 电机的方波控制。

2.1 电机控制算法概述

磁场定向控制（Field Oriented Control, FOC）是一种先进的交流电机矢量控制技术，其核心思想是通过坐标变换将定子电流分解为产生磁通的 d 轴分量和产生转矩的 q 轴分量，实现对电机磁链和转矩的解耦控制。FOC 利用 Clarke 变换和 Park 变换将三相静态坐标系转换为同步旋转的 dq 坐标系，在此基础上通过 PI 调节器分别控制 i_d 和 i_q 电流，最终通过逆变换和空间矢量脉宽调制（SVPWM）技术生成驱动信号。这种控制方法使得交流电机的控制性能接近于直流电机，具有动态响应快、调速范围宽、效率高等优点。然而，FOC 控制的实现面临着诸如精确的转子位置检测、复杂的实时计算、对电机参数变化的敏感性等技术挑战。

云途无刷电机也是基于 FOC 算法实现，同样包含传统 FOC 的 Clarke、Park 变换，SVPWM 算法、位置估算算法、PID 控制算法等，如下是一个简单的控制算法框图：

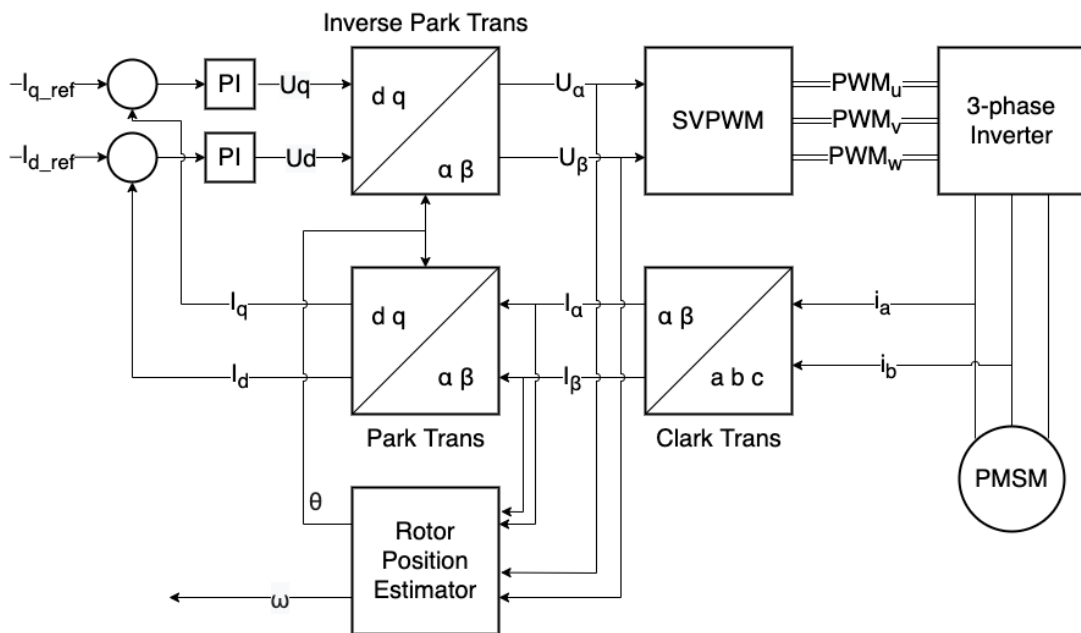


图 2: Motor Control Algorithm

2.2 电流采样

电流采样是 FOC 控制的基础，它为 FOC 控制回路提供了必要的反馈信号。通过精确测量电机相电流，控制器能够实时了解电机的运行状态，进而进行闭环控制。

电机的转矩实际上与电机内部的电流密切相关，无刷电机内部的磁场实际上也是依靠电流产生的，所以对于电机磁场的控制，本质上就是对电机电流的控制。

常用的电机相电流采样方法有分流电阻采样、霍尔电流传感器和磁通门传感器等，一般汽车应用中都是采样分流电阻采样的方式实现，电阻采样根据成本需求以及精度要求等又分为单电阻、双电阻和三电阻采样。因为无刷电机一般采样中心对齐 PWM 方式进行电压控制，所以双电阻和三电阻电流采样点相对固定，一般软件不需要做额外的处理。而单电阻采样因为硬件上只需要一个电子分流器和一个运放电路，总体成本上更有优势，不过单电阻采样的方式需要软件配合动态调整采样点，并通过软件重建三相电流，对于 MCU 有更高的算力要求。

单电阻采样的基本原理是通过在母线上的一个采样电阻采集母线电流，通过无刷电机三个相位的不同开启状态来动态调整采样点采样，并根据相位开启状态恢复无刷电机的三相电流。然而并非所有情况都能满足单电阻采样的需求，电

机电流较小时，此时 PWM 占空比较小，留给单电阻采样的窗口也就对应很小，或者当矢量电压刚好需要跨扇区的时候，此时电机只有一个相位有有效电压，单电阻对于其它两相的电流采样也没有对应的窗口。

为了方便开发者更好的实现低成本的单电阻采样，云途电机控制库中直接集成了单电阻采样的控制算法，支持通过 PWM 相位偏移来扩展单电阻采样所需要的采样窗口，并可以根据采样结果重建三相电流。开发者只需要根据自身硬件特性设置电流采样所需要的窗口。

2.2.1 电流采样的定点化处理

因为 Cortex M0+ 内核内部没有 FPU 单元，电机库中对于 FOC 的运算都是基于定点化处理的方式实现的，电流和电压采样，我们需要对输入信号进行定点化处理。

为了方便对电流进行定点化处理，ADC 支持输出 16 位左对齐的结果，这样相当于直接输出定点化结果。

另外，对于电机母线电流进行采样时候，母线电流会出现电流为负的情况，此时电机一般是减速阶段，为了支持对于这种负电流的采样，芯片内部 PGA 有一个固定的偏置（一般为 V_{ref} 的一半），所以实际采样电流计算可以参考：

$$ADC_{code} = \frac{I_{bus} * R_{sample} * Gain}{\frac{V_{ref}}{2}} * 2^{16}$$

在我们库函数中，开发者需要根据自身硬件配置系统可以运行的最大电流，这个和采样电阻大小、参考电压以及增益配置有关，具体可以根据如下公式计算：

$$I_{max} = \frac{\frac{V_{ref}}{2}}{R_{sample} * Gain}$$

上述公式，单位是 V，单位是欧姆，电流结果单位为 A。

当使用 12 位 ADC 进行电流采样时，电流采样结果减去偏置 Code 之后，需要进行有符号的移位 4 位处理来正确转化为 Q(1.15) 格式的定点化结果。

2.2.2 电压的定点化处理

观测器同时也是依赖母线电压的，所以对于母线电压同样需要进行定点化处理。因为母线电压一般是高于

3 PWM 输出和电流采样

3.1 MSPWM 模块简介

MSPWM 模块是 YTM32Z1MD0x MCU 中专为电机驱动设计的 PWM 模块，是专用的电机 PWM 控制模块，在云途通用 MCU 中 eTMR 模块可以实现同等的功能，PWM 模块主要有如下一些功能：

1. 16 位计数器，工作时钟最高 56MHz，内部支持 7 比特预分频
2. PWM 输出频率占空比可调
3. 支持互补模式、独立模式
4. 支持双缓冲寄存器的硬件加载机制
5. 支持多种触发信号生成
6. 支持溢出、通道匹配等多种中断事件
7. 模块支持 8 个通道，每个通道有两个比较点，可以配置为 PWM 输出比较点和 ADC 触发点

8. 支持左对齐、右对齐、中心对齐以及 PWM 移相等多种 PWM 输出方式
9. MSPWM 模块的 PWM 输出模式支持 VAL0 和 VAL1 两个比较点，可以灵活实现各种 PWM 的输出，以一个通道的 PWM 输出为例：

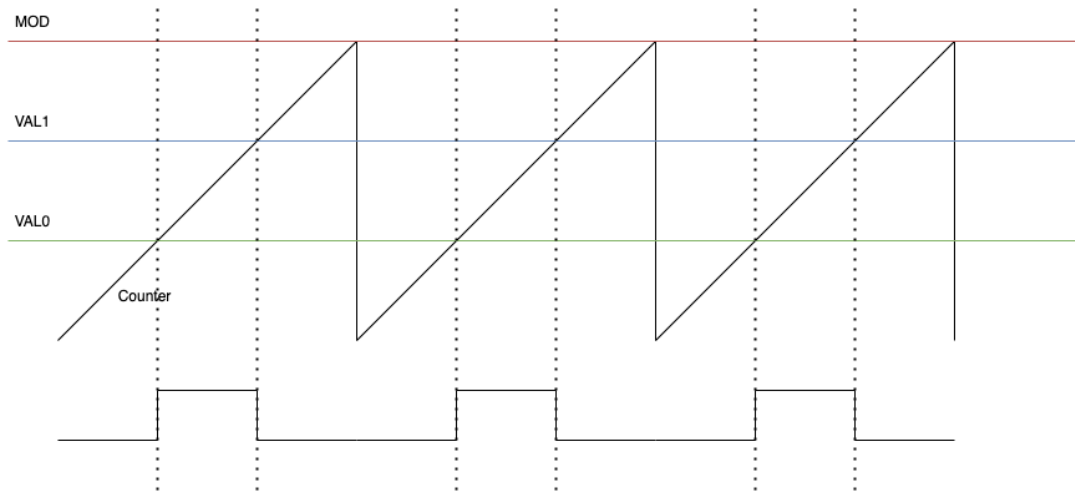


图 3: MSPWM

最简单的 PWM 输出时间就是 VAL0 和 VAL1 与 PWM 内部 Counter 的比较，调整 VAL0 和 VAL1 的值可以实现左对齐、右对齐、中心对齐以及移相等 PWM 波形的输出，这个和一般上下计数与同一个比较值比较的中心模式在实现上有一定差异。

当 PWM 的两个通道配置为互补输出模式的时候，比如 CH0 和 CH1 互补，CH1 输出为 CH0 的反相（还有死区插入逻辑），此时 CH1 的 VAL0 和 VAL1 都不再控制 PWM 输出，但是它们对应的触发输出仍然有效，所以我们可以修改 VAL0 和 VAL1 的值实现 PWM 任意位置对 ADC 采样的触发。

3.2 单电阻电流采样触发配置

因为电机电流是无刷电机控制的基础，一般的硬件设计中电流采样都是放在下桥臂或者母线的地端采样，以成本最低的单电阻采样为例，一般电流采样和功率桥结构如下：

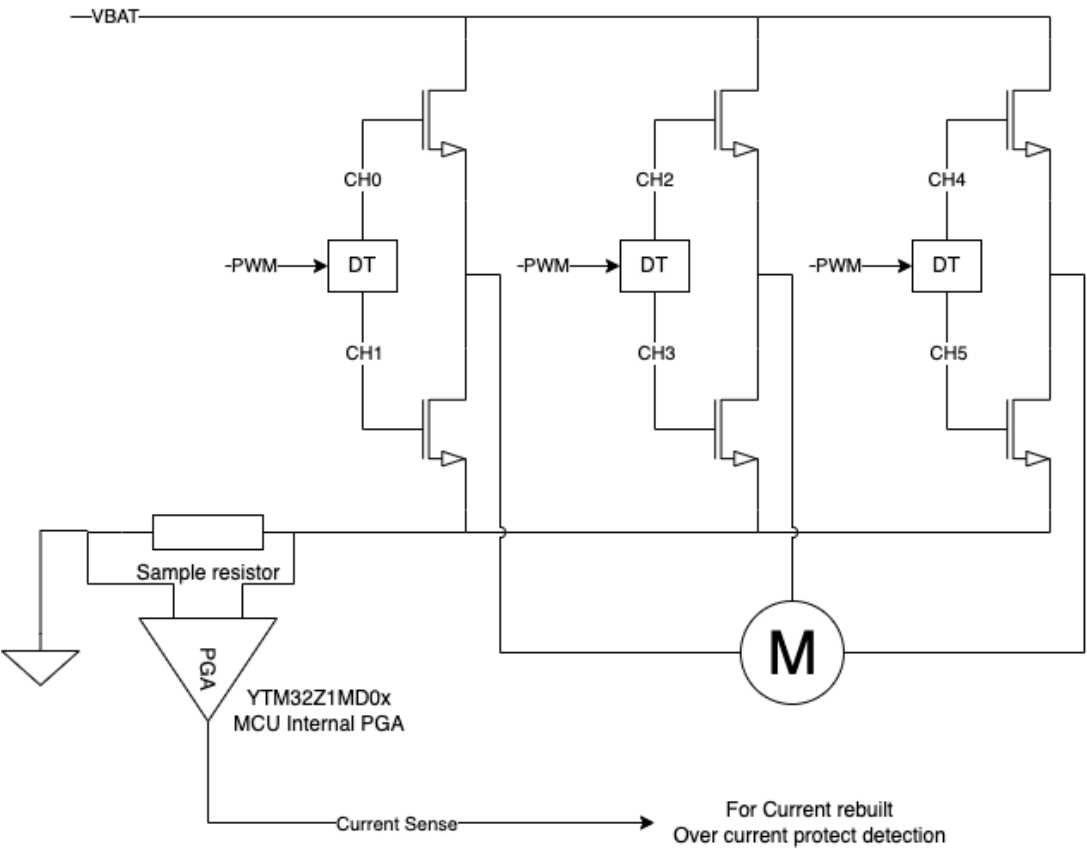


图 4: Power Inverter

YTM32Z1MD0x MCU 内部集成的预驱单元，预驱单元集成互补输出和死区插入集成，同时集成可增益运放单元以及各种硬件异常状态监测，硬件上只需要外部增加 NMOS 功率驱动管和电流采样器即可。

无刷电机的驱动采用中心对齐的 PWM 驱动，YTM32Z1MD0x MCU 中通过专为电机设计的 MSPWM 模块产生互补 PWM 波形并且可以配合 ADC 实现 PWM 周期内多个通道任意位置的采样，特别适用于对于单电阻采样位置不固定的场景。

YTM32Z1MD0x MCU 中单电阻采样方案中 MSPWM 通道使用情况如下表所示:

通道	输出	工作模式	触发输出	说明
CH0	U 相高边控制	PWM 输出	无	VAL0 和 VAL1 配合实现支持移相的中心对齐 PWM
CH1	U 相低边控制	CH0 互补输出	VAL0 VAL1	VAL0 和 VAL1 为单电阻两个采样点触发
CH2	V 相高边控制	PWM 输出	无	VAL0 和 VAL1 配合实现支持移相的中心对齐 PWM
CH3	V 相低边控制	CH2 互补输出	无	VAL0 VAL1 未使用，也支持配置为 ADC 采样触发点
CH4	W 相高边控制	PWM 输出	无	VAL0 和 VAL1 配合实现支持移相的中心对齐 PWM
CH5	W 相低边控制	CH4 互补输出	无	VAL0 VAL1 未使用，也支持配置为 ADC 采样触发点

下图是一个 PWM 输出和电流采样以及用户 ADC 通道采样的时序关系图。

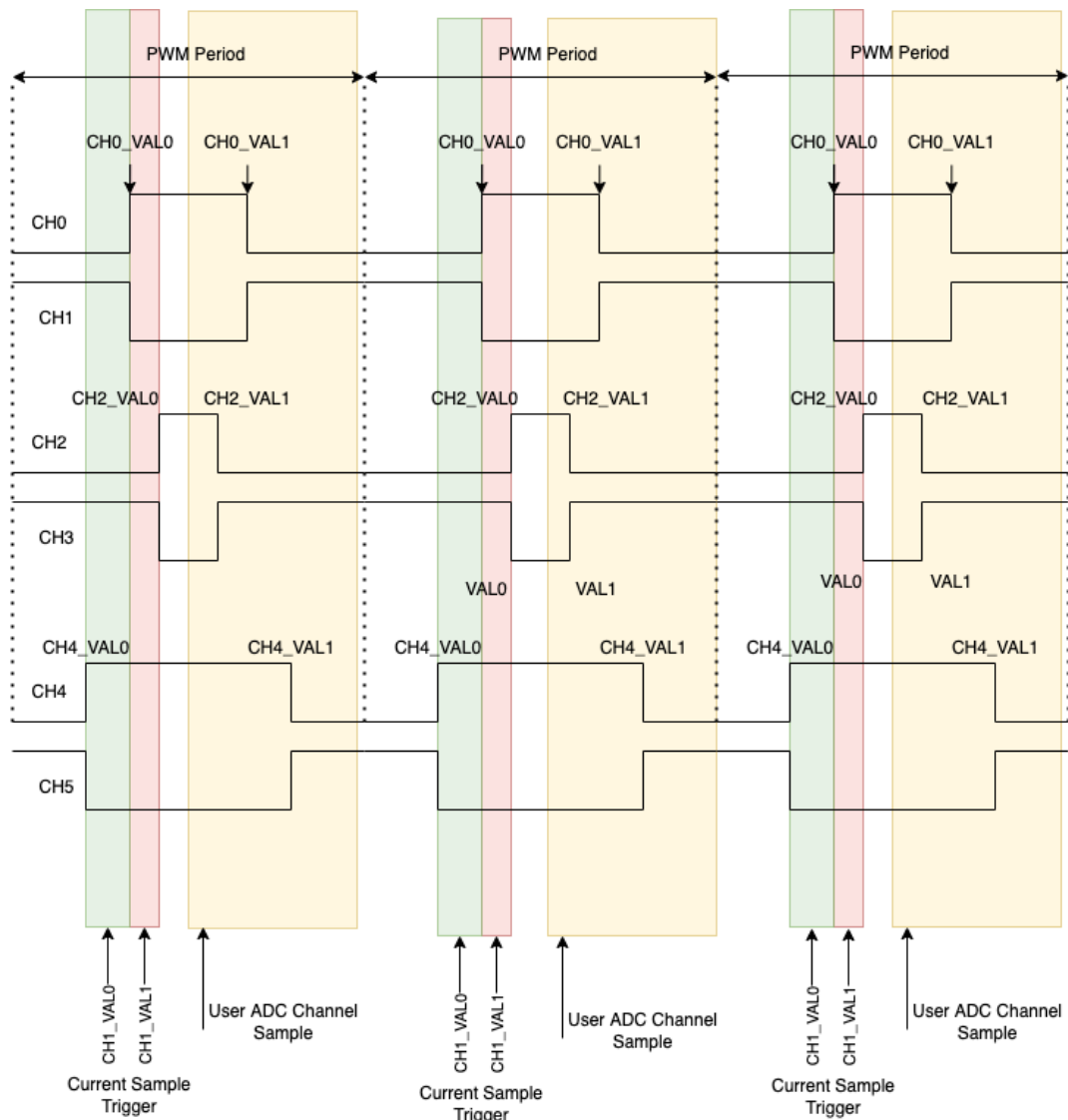


图 5: R1 Current Sample

需要注意的是这里只是一个典型的单电阻采样的 PWM 输出图，并不包含 PWM 移相操作。因为单电阻采样实际采集的是母线电流，所以应避免 PWM 为 0 矢量（PWM 全高或者全低）的时刻采样，比如上图中绿色窗口下 CH1 和 CH3 为开启状态，红色窗口 CH3 打开，两个窗口都可以采集到有效的电流信号，依据基尔霍夫电流定律和三个相位的关系，我们可以通过计算得到实际的三相电流。

这里我们将电流采样集中在 PWM 的前半个周期，当采样结束之后立即开始 FOC 电流环的运算，整个运算需要再后半周期完成，并在周期结束之前更新 PWM 的占空比，这样可以做到整个电流控制的延时最小，提高系统响应。在电流采样完成中断中，可以软件触发 ADC 对其它用户通道的采样，这些通道采样一般对于采样点没有特别要求，主要采集母线电压、环境温度、电机温度、母线平均电流等，这些通道的采集和转换是在黄色采样窗口中完成的。

3.3 Clarke 变换和 Park 变换

在磁场定向控制 (FOC) 中,Clarke 变换和 Park 变换是实现高效电机控制的关键数学工具。这两种变换共同将三相交流系统转换为一个更易于控制的两相直流等效系统。

3.3.1 Clarke 变换

Clarke 变换, 也称为 α - β 变换, 将三相静止坐标系 (a-b-c) 转换为两相正交静止坐标系 (α - β)。

Clarke 变换数学表达:

$$\begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix}$$

根据电流基尔霍夫定律, 电机三相电流满足

$$i_a + i_b + i_c = 0$$

我们可以得到

$$i_c = -(i_a + i_b)$$

带入公式并化简我们得到:

$$i_\alpha = \frac{3}{2} * i_a$$

$$i_\beta = \frac{\sqrt{3}}{2} (i_a + 2 * i_b)$$

这里对于 Clarke 变换, 我们采用幅值不变的原则, 变换结果乘以系数 $2/3$, 于是我们得到:

$$i_\alpha = i_a$$

$$i_\beta = \frac{i_a + 2 * i_b}{\sqrt{3}}$$

Clarke 变换可以将一个三相系统减少一个自由度, 降低控制难度, 保持幅度不变也可以降低后续处理的难度。

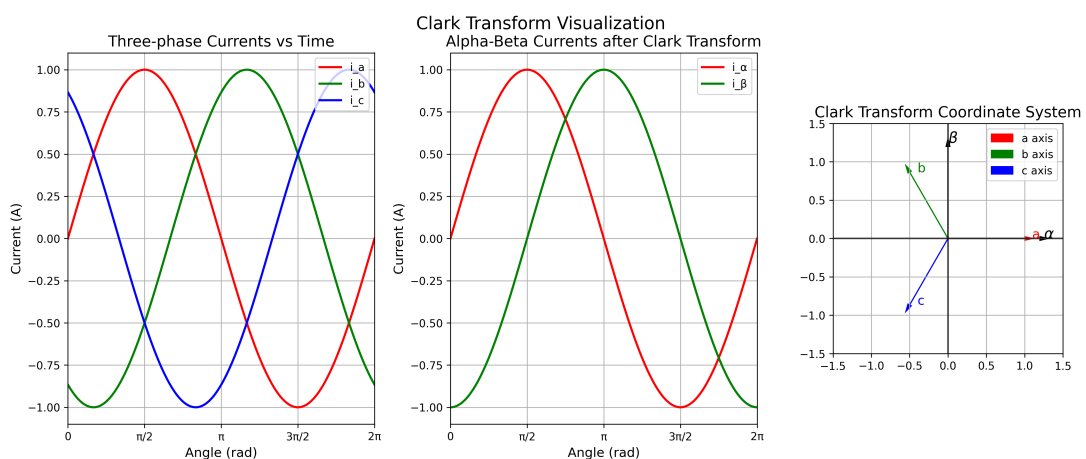


图 6: Clarke Transformation

3.3.2 Park 变换

然而经过 Clarke 变换之后的电流仍然不适合直接进行处理，为了进一步降低控制难度，FOC 算法引入了 Park 变换，也称为 d-q 变换，它的作用是将静止的 α - β 坐标系转换为与转子同步旋转的 d-q 坐标系。

Park 变换的数学表达：

$$\begin{bmatrix} i_d \\ i_q \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix}$$

上述公式中的 θ 是转子电角度，Park 变换通过建立一个旋转坐标系将交流的变换成直流量，这样系统就可以实现对变换后的量进行线性控制，降低系统控制难度。从效果上来看相当于将交流的无刷电机等效成直流电机，从而简化控制系统。

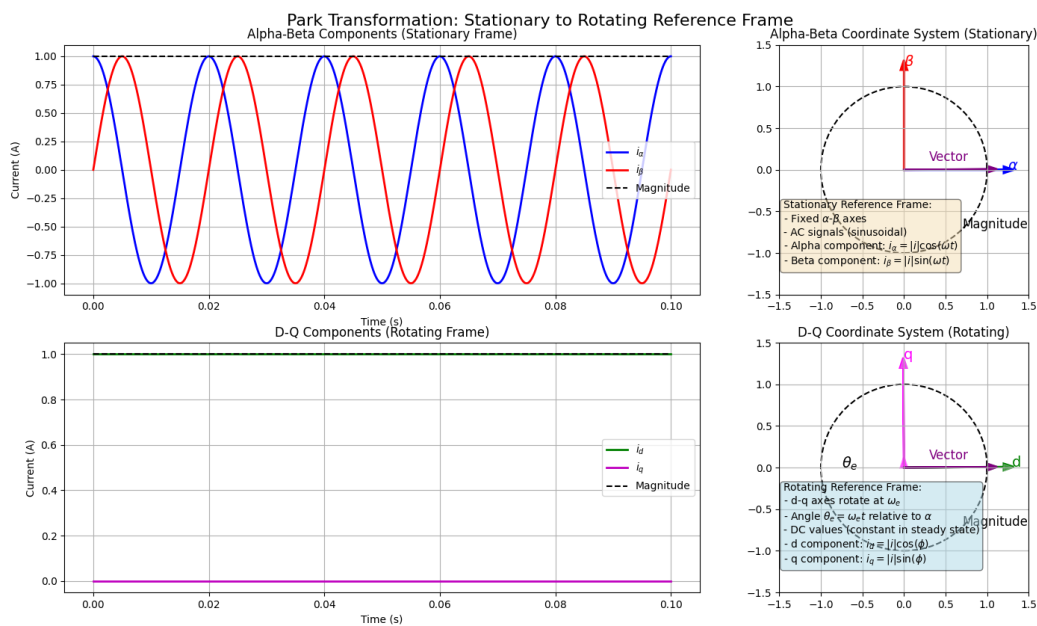


图 7: Park Transform

Park 变换的关键在于通过转子电角度，这个角度是整个 Park 变换的基础，所以转子角度的准确性直接影响到 Park 变换的结果。为了准确得到转子角度，电机控制中通常有有传感器和无传感器两种方式。

基于角度传感器获取转子角度，常用的角度传感器有霍尔传感器、磁编码、光电编码，旋转编码器等。角度传感器的优势是可以准确获取转子位置，软件处理相对简单。不过因为增加了硬件，所以成本上不如无传感器模式，另外也需要考虑各种传感器失效的情况。

基于反电动势的转子位置估计是另外一种应用比较广泛的低成本方案，不过基于反电动势检测转子位置需要 MCU 完成大量运算，对于 MCU 的性能有一定的要求，另外因为在低速或者启动状态，电机的反电动势较小，很难准确估计电机反电动势，所以无传感器的电机启动一般需要依靠开环强拖，当电机电动势足够角度检测时再切换到闭环。

4 无刷电机转子位置估算

无刷电机无传感器的位置估算是 FOC 中的重要组成部分，即使在有传感器的系统中一般也需要增加无传感器位置估算，这样可以实现两种角度信息的相互校验。

无传感器的位置估算一般会对 MCU 有较高的算力要求，根据控制周期和对于电流噪声的限制，一般会要求 PWM 在每个周期进行更新，在典型的 16KHz PWM 模式下，MCU 需要在 62.5uS 内完成相关运算，在云途电机库中针对这部分做了算法的优化，可以做到在 30uS 内完成电流环 PID 控制和无传感器的位置估算，这样可以给其他应用处理留出足够的处理时间，避免系统运行负载过高。另外因为整个电流环的计算时间可以在半个 PWM 周期内完成，这样就可以做到周期内的电流采样和 PWM 更新，整个系统控制的时延可以降到最低。

注意：云途 FOC 的位置估算算法为特有算法，所以这部分将以库函数的形式释放，并且仅限于在云途 MCU 上使用，用户不得将该库函数修改和移植到其它芯片上使用。

5 编译选项

因为 FOC 算法对于时间要求非常高，所以使用过程中要注意如下一些事项：

- 编译参数尽量使用最高运行速度 (-Ofast) 优化，云途库函数默认使用最高运行速度优化选项编译和测试，用户应该尽量保持该参数以获得更好的运行效果
- 电机运行过程中不要使用断点，应该使用图形化实时调试的方式调试电机，而不应该使用断点，防止 PWM 在同一个位置一直保持输出而导致电机或者功率桥过流损坏。
- 注意浮点运算和除法的使用，YTM32Z1MD0x 因为采用 Cortex-M0+ 内核，浮点运算是通过软件计算方式实现，这种方式会占用大量的运行时间，所以只能在对时间要求不敏感的任务中使用，FOC 环路中不应有浮点运算。对于除法运算，YTM32Z1MD0x 增加了 DIVSQRT 模块，可以实现硬件除法，用户需要调用该模块实现除法和开方运算。YTM32Z1MD0x 内核支持 32 位的整数乘法运算，所以整个系统都是采用定点数方式实现，乘法是直接通过编译器调用乘法指令实现的。

6 电机参数测量

云途无刷电机库依赖实际的电机参数，为了得到更好的运行效果，需要再算法适配过程中合理的配置电机相关参数。这里跟电机相关的参数主要包括如下参数。

6.1 电机相电感

目前云途电机算法主要支持标贴式的无刷电机，所以有默认 $L_d=L_q$ 的前提，用户在配置电机电感参数时候，可以直接用 LCR 测量电机线电感，通常测试频率选择 1KHz，然后取平均并除二即可得到电机相电感。配置参数中电感的单位为 H，支持浮点数，在系统初始化过程会自动适配为合适的定点数。

6.2 电机相电阻

电机相电阻也是很重要的一个参数，参数的测试可以直接用 LCR 测量线电阻，通常测试频率选择 100Hz，然后取平均除二即可得到。对于某些相电阻比较低的电机，需要用毫欧表进行测量，虽然电机库的运行可以适应一定的参数差异，不过精确的电机参数总会得到更好的运行效果。

6.3 电机反电动势系数

云途电机算法库在无感运行模式下主要通过反电动势信息对转子位置进行估算，所以依赖实际电机的反电动势系数，对于这个系数的测试，比较直接的测试方法就是在外部拖拽电机的时候通过示波器测量相线之间的波形，因为电机观测器实际是基于电机电角度建模的，所以实际并不关心电机具体的极对数。假设我们测量电机相线电压波形如下：

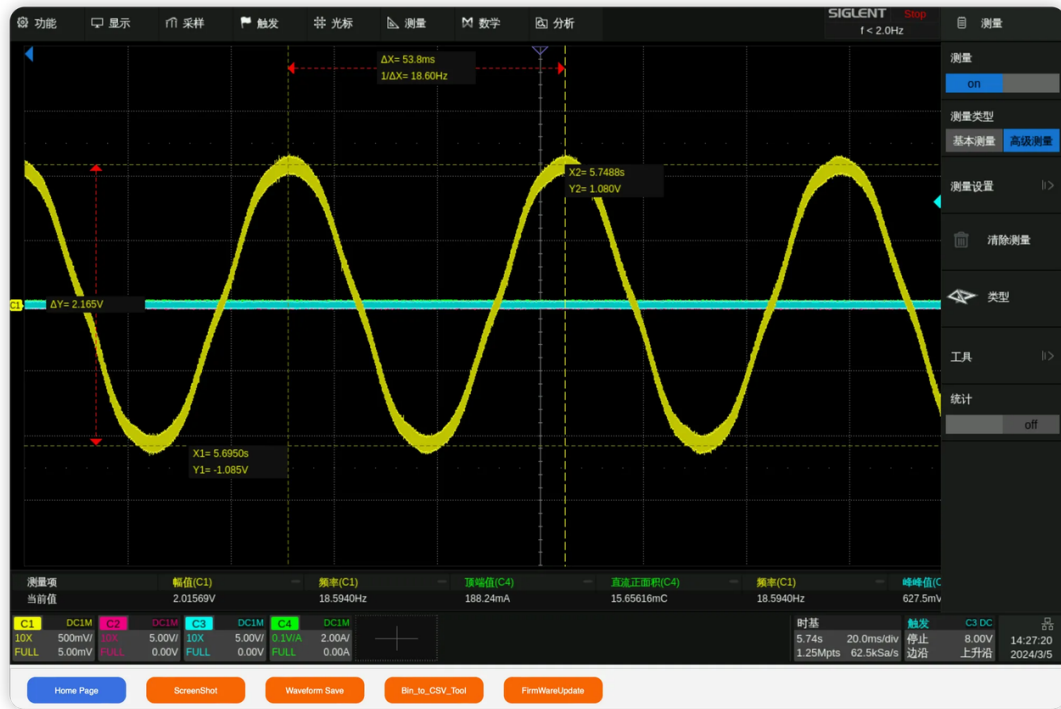


图 8: Motor KFI

选取其中较为均匀的反电势波形 (峰值最大的一段) 我们可以根据如下公式计算：

$$KFI = \frac{V_{pp}}{F * 4 * \pi * \sqrt{3}}$$

其中 F 为正弦波频率，单位是 Hz， V_{pp} 是峰峰值，单位是 V。比如上图中 MOTOR_KFI 可以代入实际数据得到：

$$MOTOR_KFI = \frac{2.165}{18.6 * 4 * 3.14 * 1.732} = 0.006834$$

需要注意的是外部拖拽电机时候尽量保持电机匀速旋转，在波形截取时候也尽量选择正弦波幅度基本稳定的周期进行测量。

6.4 电机极对数

电机极对数并不是电机控制系统必须的一个参数，因为电机控制实际都是按照电角度进行计算的，本质上相当于将电机统一作为单极对电机进行控制的。在实际应用中，电机极对数主要用来做电机电角度和机械角度的相互转化。

对于电机极对数测定，可以采用如下方式进行测试：

1. 将电机两个相线接电源，电源需要根据电机具体功率和工作电流设定合适的限流值，电源工作在恒流模式
2. 如果不确定电机限流可以缓慢增加限流，在电机通电时看电机是否可以停到一个固定角度，转动电机可以感受到电机转过一定角度之后又明显停顿即可

3. 记录电机当前角度，完整转动电机一周，记录电机转动过程中有几个明显停顿角度，这个锁定角度的数量就是电机的极对数

另外一种测试方式还是电机两个相线接示波器，然后手动转动电机完整一周，观察示波器上出现一个完整的正弦周期。

7 电机工程调试步骤

7.1 开环验证硬件

无刷电机的调试过程相对来说比较复杂，需要用户对电机的运行原理有一定的了解，首先我们需要确定硬件是否可以正常工作，为了快速对硬件和用户软件模块配置进行验证，电机工程库支持开环运行模式，这个模式可以通过参数配置开启。

当开启开环运行模式之后，控制器会根据给定的电压和转速发出旋转矢量电压，开环带动电机运行，用户需要根据电机的实际类型和工作电流调整合适的转速以及开环运行电压，当电机运行起来之后，用户可以通过示波器观察电机的电流波形是否为正弦波形。

在电机电流为正弦波形之后，可以通过 J-Scope 等软件观察电机控制状态变量中的 i_a 和 i_b 的曲线是否正常，如果电流波形为正弦波形，并且 i_a 和 i_b 的曲线也为正弦波形，说明 PWM 模块可以正常触发 ADC 进行电流采样，另外 i_a 和 i_b 的中心点应该保持一致，并且应该在 0 电流附近上下变化，如果中心点异常则说明 ADC 没有正确的校准 OFFSET 值，或者有电机的相线接线异常。

电流可以正常采样之后，我们就可以对观测器的性能进行测试了，因为此时状态变量的角度是开环累加的一个角度，观测器会根据反电动势计算一个电机的实际角度，我们可以通过 J-Scope 将两个角度波形绘制出来进行对比，当两个角度有固定的相位延迟之后，说明观测器可以正常工作。注意，因为电机在开环运行时候，因为实际没有外部负载，所以角度的变化应是线性的，也就是速度应该是稳定的。这个时候观测器的输出角度应该也是线性的，如果中间出现非线性的情况，这说明观测器依赖的输入信号有异常，应该排查 vdc 采样是否正常（包含是否为合适的 Q15 格式），采样点是否正常落在采样开窗中，电机的相电阻、相电感和反电动势系数是否正常。

7.2 电流闭环调整合适的 PI 参数

在观测器正常工作之后，我们就可以对电流闭环进行调整了，这个时候我们需要通过参数调整电机的运行模式，切换到 MOTOR_CURRENT_LOOP 模式，此时电机在开环启动之后，会逐渐切换到观测器角度输出运行，电机的电流环也会开始工作，电流环的 i_d 目标值为 0， i_q 的目标值就是启动电流大小，用户可以根据实际的启动电流大小调整电流环的目标值。调整电流环时候可以通过调试软件实时调整电流环 i_q 的目标值，然后通过示波器电流探头观测电机的相线电流，通过调整电流环的 PI 参数得到合适的电流环响应。

电流环的调整还有一个技巧，因为我们使用的电机大部分都是表贴的无刷电机， L_d 和 L_q 的参数基本是一致的，也就是两个电流环的 PI 参数也是一致的，并且电机的电流环实际是电机内部 LR 电路对电压的响应，所以调整过程并不需要电机处于运转状态，我们首先通过此公式计算出一个大概的 PI 参数，其中阻尼系数 ξ 通常为 0.707， ω_n 为自然截止频率，

$$K_p = (2 \times \xi \times \omega_n \times L_s - R_s) \times \frac{V_{max}}{I_{max}}$$

$$K_i = \omega_n^2 \times L_s \times \frac{V_{max}}{I_{max}}$$

将计算出的 PI 参数乘以 32767 进行标么化处理后, 写到 D 轴的 PI 参数中, 再将 Q 轴的 PI 参数设置为 0, 后将 I_q 设为 0, I_d 根据电机功率给个合适的值, 通过 J-Scope 等软件观察 D 轴电流环的响应情况, 据此调整 D 轴 PI 参数, 这样可以快速的得到一个合适的 D 轴的 PI 参数, 然后将相同的参数复制到 Q 轴, 这样可以快速得到一个合适的电流环参数。

7.3 速度闭环调整合适的 PI 参数

电机的速度环因为很难建立精确的电机运动模型, 整个系统涉及电机的机械结构, 负载类型和负载变化等, 所有速度环的调试目前还是主要靠经验进行调试。

8 电机控制程序

云途电机控制 demo 提供了完整的电机控制状态机, 控制程序主要分为 Motor_FOC_Service 和 Motor_StateSpeedLoopService 两个关键部分, 分别提供电流环的控制和速度环的控制。

注意: 因为电机控制本身实时性要求很高, 针对电机控制程序的调试一般都是通过调试器在线实时读取变量进行数值或者图形的观测, 在调试过程中**不要使用断点**, 这样会导致 PWM 在同一个位置一直保持输出, 导致电机或者功率桥过流损坏。

8.1 电流环控制服务

电机矢量控制的核心是电流环, 电流环的控制服务主要由 Motor_FOC_Service 函数提供, 该函数的主要作用是根据电机的状态变量和设定的目标值, 计算出电流环的输出值, 然后将输出值输出到电机的控制电路中。

Motor_FOC_Service 需要在每个 PWM 周期中执行一次, 对于执行效率有非常高的要求, 为了更为高效的兼容各种应用, 这个服务均以库的形式提供, 库的编译默认开启了最高优化等级选项。Motor_FOC_Service 函数输入为电机的相位电流, 输出为计算后的 PWM 值, 函数内部同时会运行角度观测器, 因此也会更新电机的角度和速度信息。

Motor_FOC_Service 默认为单电阻方案, 所以内部也会包含电机相位电流的重建, 在输出 PWM 的计算中, 服务也会根据用户配置的开窗大小对输出 PWM 进行移相计算, 同时给出对应的电流采样点。对于双电阻和三电阻方案, 该服务的变体为 Motor_FOC_ServiceR23, 此时服务会直接使用用户给定的相位电流进行计算, 输出 PWM 也不再包含移相操作, 计算更为简单。

按照典型单电阻模式, Motor_FOC_Service 主要完成如下工作:

1. 根据当前转子位置和两次单电阻采样值重建电机三相电流
2. Clarke 变换, 将三相静止坐标系转换为两相正交静止坐标系
3. Park 变换, 将两相正交静止坐标系转换为与转子同步旋转的 d-q 坐标系
4. 根据 d-q 坐标系下的电流和目标电流计算出 dq 轴的 PI 输出
5. 输出电压进行圆周限幅操作, 如果是开环模式则直接使用用户给定的输出电压
6. 执行角度观测器, 更新电机的角度和速度信息, 角度观测器在加速和运行状态都会执行
7. 反 Park 变换, 将 d-q 坐标系下的电压转换为两相正交静止坐标系
8. SVPWM 计算, 根据 dq 轴的电压计算出三相电压, 并根据用户给定的开窗大小进行移相计算

电流环的计算可以直接放在 ADC 的中断处理函数中执行, 在完成电流采样之后立即开始执行电流环的计算, 这样可以充分利用有效的电流环计算窗口时间, 计算完成之后应该立即更新 PWM 的占空比, 应确保 PWM 的更新在当前 PWM 结束之前完成, 否则应该进一步优化计算时间或者降低 PWM 的频率, 在完成 PWM 更新之后, 用户可以增加自定义的一些高频任务。

8.2 速度环控制服务

速度环的控制服务主要由 Motor_StateSpeedLoopService 函数提供，该函数的主要作用是根据电机的状态变量和设定的目标值，计算出速度环的输出值，然后将输出值输出到电机的控制电路中。

速度环的典型运行周期是 1ms，但切为速度闭环后速度环响应周期为 PWM 输出周期，速度环除了进行速度控制之外还实现了电机控制状态机的实现，这里涉及的几个状态如下：

1. STATE_STOP: PWM 停止输出，电机处于停机的无驱动状态。
2. STATE_ERROR: 异常状态，用户可以在触发异常之后进行必要的异常处理，然后将电机状态切换为停止状态。
针对有硬件保护的异常，PWM 会直接根据用户配置好的保护逻辑进行保护，这里异常状态主要用于软件异常。
3. STATE_INIT: 初始化状态，在初始化状态下，电机相关的 PI 控制器会进行复位，观测器也会进行复位，控制系统重新恢复用户的初始化速度、电流等参数。ADC 也会重新进行电流采样的 OFFSET 值校准，用户也可以在该状态下对电机的连接和 MOS 等进行校验。所有操作完成之后，状态机会开启 PWM 并将占空比配置为 50%，即 0 矢量和 1 矢量的交替，此时电机并没有有效的电压。不过电流采样和电流环已经开始触发运行，初始化完成之后电机进入 STATE_LOCK 状态。
4. STATE_LOCK: 锁定状态，在锁定状态下，状态机会根据用户配置的锁定电流和锁定时间，在给定的锁定时间内将电流线性增加到锁定电流，整个过程维持 0 角度。锁定完成之后，状态机会进入 STATE_ACCE 状态。
5. STATE_ACCE: 加速状态，在加速状态下，状态机会根据用户配置的加速时间，在给定的加速时间内将电机速度线性增加到启动速度，加速阶段电流保持为启动电流。此时电机位于开环强拖状态，当到达给定的开环速度之后，状态机会结合当前的环路运行模式（开环、电流闭环、速度闭环）以及观测器的输出速度决定下一个状态。
6. STATE_WORK: 运行状态。根据用户配置的运行模式，运行状态分为电流闭环和速度闭环两种模式。电流闭环条件下，电流环的目标值用户可以直接修改，用户可以对目标电流进行修改来观测系统响应。速度环模式下，电流环的输入为速度环的输出值，用户可以调整目标速度来观测系统的响应。
7. STATE_TEST: 测试状态。测试状态下，电机的角度始终为 0，用户可以在该状态下对电流环的响应进行调试，确定合适的电流环参数。

状态机的切换可以参考如下流程图：

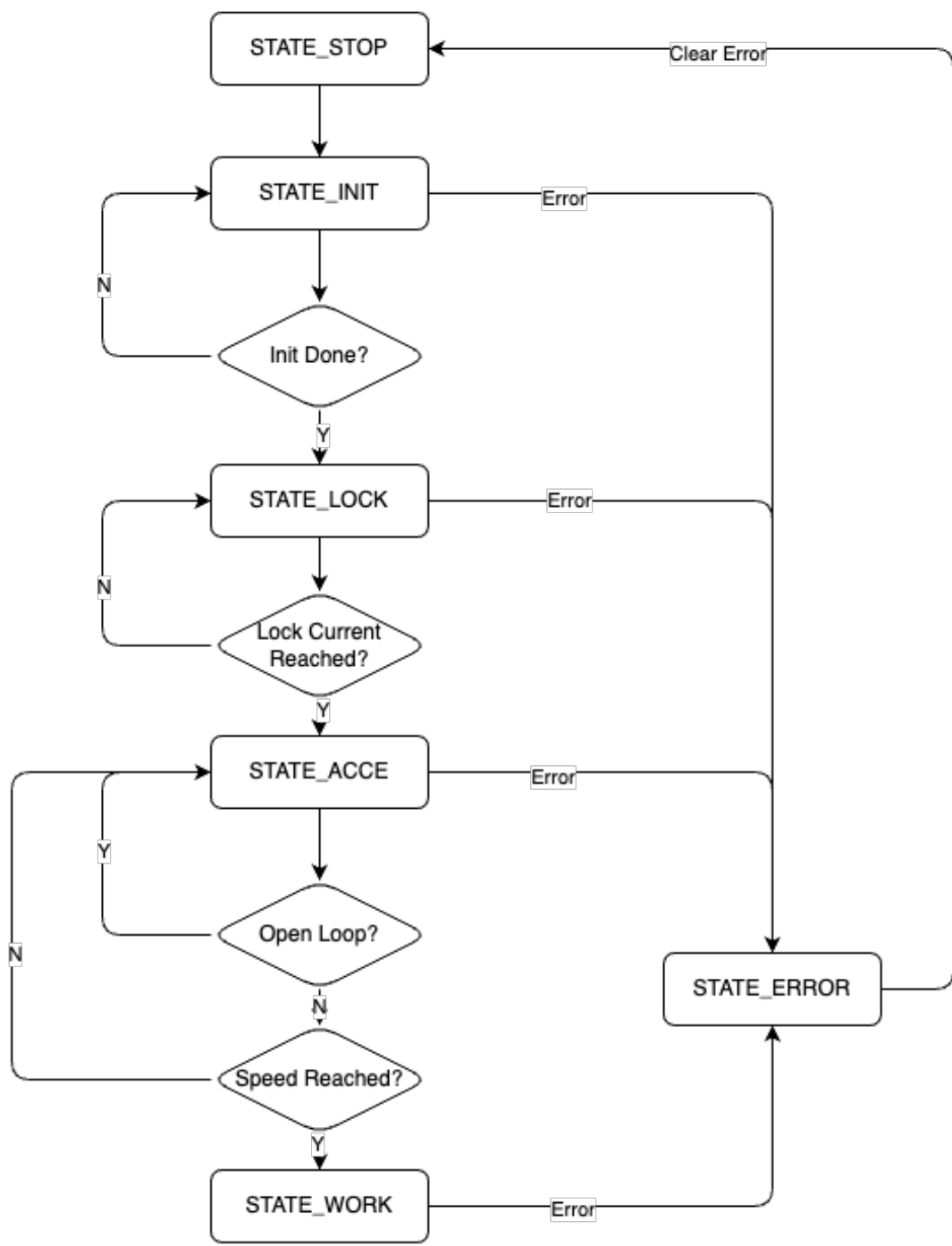


图 9: Speed Loop State Machine

上图并没有包含 STATE_TEST 等特殊状态，默认演示程序也没有使用这些状态，用户可以根据自己的需求进行修改。

因为 Motor_StateSpeedLoopService 函数对于执行时间要求相对较低，并且涉及用户针对电机的处理逻辑，所以这部分状态机的代码是以源码形式进行释放，用户可以根据自己的需求进行修改。结合之前提到过的电机调试步骤，用户可以自行调整状态机的实现，并结合已有的三种闭环模式进行电机的参数调试。

上面状态机的切换中，在 STATE_ACCE 状态支持开环的状态保持和闭环的状态切换，用户在硬件调试状态可以保持在 STATE_ACCE 状态实现开环运行，这种状态下就可以对电流采样和 PWM 的发波以及 ADC 的触发进行测试，开环状态最重要的是保持电机电流环运行周期和 PWM 周期同频，并且执行时间不能超过 PWM 当前周期。确保电流环的

执行频率是稳定的。另外针对 ADC 和硬件电路部分，需要保证在开环状态 ADC 可以准确采集到电机的相位电流，电机的相位电压的 PWM 控制也应该是准确的。

在 STATE_ACCE 状态，观测器已经开始工作，用户可以通过对比开环角度 $gs.theta$ 和观测器 $gs.theta$ 的波形来确保观测器是准确的，如果观测器角度有畸变或者角度异常，则应该检查电机 R_s 、 L_s 和 KFI 参数设置是否准确，电机的母线电压采样是否准确，电压也应该维持 Q15 格式，同时参数中 V_{peak} 和 I_{peak} 应该和电路硬件相匹配，当参数设置无误且观测器输入电流准确时，观测器角度输出应和开环累加的角度输出有一个比较固定的相位差，此时可以开始进行电流闭环的调试。

8.3 重要的配置参数解释

当使用云途电机 Demo 程序适配新的电机或者新的硬件平台时，需要对一些重要的配置参数进行调整，这些参数包括：

1. PWM 频率相关参数，包含系统的主频（实际需要填写 eTMR 模块的时钟频率），PWM 采样窗口，PWM 死区时间等。其中采样窗口在双电阻和三电阻中是无效的，可以直接配置为 0，死区时间则应该和硬件电路相匹配。针对自动运算参数原则上应该不需要做改动。
2. 最大电流和最大电压，这两个参数主要实现对电机电压和电流的标幺化处理，用户应该严格按照实际的电路参数进行配置，否则可能产生数据溢出和观测器运行不准确的现象。
3. MOS 内阻，这个参数可以根据实际 MOS 参数进行配置，这个参数会影响观测器的输出。
4. R_s 和 L_s ，这两个参数主要用于观测器的计算， R_s 和 L_s 应该和实际电路相匹配，否则观测器的输出可能会出现异常。
5. KFI，这个参数主要用于观测器的计算，KFI 通过实际电机测量进行调整，本文档中对该参数的测量也有明确说明，用户可以参考测试方法进行测试，新的 Demo 中也加入了计算公式，只需要对应给出 V_{pp} 和频率信息即可，该参数不准确会影响观测器对角度的观测。
6. 观测器滤波系数，这个一般不需要特别调整，一般默认即可。
7. 电流环和速度环的 PI 参数，这两个参数主要用于控制电机的运行，用户需要根据实际的电机和硬件电路进行调整，这个参数调整的难度比较大，需要对电机的运行原理有一定的了解。
8. 电机的启动速度，这里用的是 RPS 值，如果是 RPM 值，这需要除以 60。
9. 电机的锁定电流，这里用的是 A 值，支持浮点数。
10. 电机的锁定时间，这里用的是 ms 值。
11. 电机的加速时间，这里用的是 ms 值。
12. 电机的运行速度，这里用的是 RPM 值。
13. 电机的运行模式，包含开环、电流闭环和速度闭环三种模式。
14. 电机开环运行电压，这里用的是 Q15 格式的电压，满幅值为实际母线电压，用户在开环模式时应该配合启动速度进行调整。

9 电机运行过程中的异常检测

电机在运行过程中应该程序保持对各种运行异常的检测，比如过流、过压、欠压等。在 YTM32Z1MD0x MCU 中实际已经集成了很多硬件异常的硬件监测，本章将会针对这些异常进行详细说明。

9.1 通过硬件保护功能进行异常检测

YTM32Z1MD0x MCU 中异常检测集中在 PCU 和 PREDRV 两个模块。

其中 PCU 主要监测 MCU 本身运行的一些异常：

1. VDDA 电源的过压和欠压保护，对应标志位 PCU_SR_VDDA_UV 和 PCU_SR_VDDA_OV, VDDA 是芯片内部的 LDO，也是芯片正常运行的基础，PCU 模块可以持续检测 VDDA 的电压，当电压过高或者过低会触发对应的中断事件。
2. VBOOST 的过压和欠压保护，对应标志位 PCU_SR_VBOOST_UV 和 PCU_SR_VBOOST_OV, VBOOST 用于给预驱提供高边开启电压，当 VBOOST 电压过高或者过低会影响高边 MOS 的开启速度，PCU 模块可以持续检测 VBOOST 的电压，当电压过高或者过低会触发对应的中断事件。
3. Over Current 过流保护事件 PCU_SR_OC_DRV 标志，过流保护是通过检测 ILS 电阻上的电压来实现过流异常的监测的，具体的阈值通过 PREDRV_ILS_OVCFG 进行配置，注意这里过流保护对应的实际电流和外部采样电阻大小有关，和芯片内部运放增益大小是无关的。
4. Over Temperature 过温保护事件，对应标志位 PCU_SR_OVT_DEB。过温保护是通过芯片内部温度传感器实现的，支持 85℃ 和 165℃ 两个阈值，当用户需要自定义的过温阈值时，也可以通过 ADC 采集芯片内部温度传感器然后换算成实际温度实现自定义过温阈值的保护。因为温度变化是一个相对缓慢的过程，为了避免误触发，温度传感器的输出应该进行适当的滤波才能用户过温保护。
5. VSM 电源的过压和欠压保护。对 VSM 电源的检测实际也间接实现了对 VS 电源的监测，因为实际应用中 VS 一般是 VSM 电源经过一个二极管提供的。VSM 电源主要用于给芯片内部预驱部分电路的供电，涉及 VBOOST 电源的产生和高边预驱的产生。而 VS 电源则主要给芯片数字控制电路供电，比如产生芯片内核电压的 LDO 供电，ADC 供电等。VSM 电源的过压和欠压阈值支持多个档位，可以通过 PCU_VSMCR_VSM_UV_CFG 和 PCU_VSMCR_VSM_OV_CFG 进行调整，对应寄存还支持对过压欠压信号的滤波和使能进行控制。

PCU 模块内部的这些异常保护都支持产生对应的中断事件和标志查询，这些异常默认并不会直接影响 PREDRV 和 PWM 的输出，用户需要在对应中断中进行异常处理，并决定是否要对电机的工作状态进行调整。如果需要硬件对这些异常事件进行保护，则需要配合 PREDRV 模块的 PREDRV_FDRV 寄存器定义针对各种异常下 PREDRV 的保护措施。针对每一种异常事件，用户都可以选择 PREDRV 模块试下如下几种不同的保护措施：

1. PREDRV 控制所有低边 MOS 开启，电机相当于相线接地，这个时候电机处于刹车状态，电机迅速停止转动，此时低边 MOS 中会有瞬时的大电流流过。
2. PREDRV 控制所有高边 MOS 开启，电机相当于相线接电源，这个时候电机同样处于刹车状态，电机迅速停止转动，此时高边 MOS 中会有瞬时的大电流流过。
3. PREDRV 控制所有高边和地板 MOS 处于关闭状态，电机相线处于高阻状态，此时电机通过 MOS 的续流二极管进行续流，电机表现为缓慢停止转动，MOS 中不会有电流突变。不过在开启弱磁控制时，电机相线电压会超过电源电压，此时会对供电系统和 MOS 形成冲击。
4. PREDRV 本身处于三态状态，此时所有高边和地板 MOS 处于关闭状态，电机相线处于高阻状态，此时电机通过 MOS 的续流二极管进行续流，电机表现为缓慢停止转动，MOS 中不会有电流突变。不过在开启弱磁控制时，电机相线电压会超过电源电压，此时会对供电系统和 MOS 形成冲击。

因为不同的保护策略各有优缺点，用户需要根据实际情况进行选择，或者直接从软件层面进行异常处理，比如缓慢控制电机降速直到系统可以正常停止。

PREDRV 模块还支持对 MOS 工作状态的监测，可以实现对 MOS 开启状态的 VDS 电压进行监测，确保 MOS 本身开启状态的正常工作，VDS 监测的基本原理是在开启 MOS 的时候测量 MOS 漏极和源极之间的电压，正常情况下 MOS 的开启电阻都是很小的，对应的漏源极电压也比较小，当 MOS 没有完全开启，或者工作异常时，MOS 导通电阻变大，此时漏源极电压也变大，因此可以通过漏极和源极之间的电压来判断 MOS 的工作状态。PREDRV 模块支持对所有 MOS 的 VDS 的电压进行监测，默认情况下软件可以通过开启对应中断来监测 VDS 异常事件。当需要进行硬件级别的保护时，则需要 PREDRV 的 PREDRV_FAULT0 中进行开启，并在 PREDRV_CTRL 寄存器定义具体的保护行为。

注意：VDS 电压会受到 MOS 工作的噪声影响，因此除非特别必要，建议 `PREDRV_FAULT0_TVDS` 保持最大值，`PREDRV_FAULT0_VDSOV` 也应配置为最大电压，已实现更为可靠的 VDS 异常监测。

9.2 通过软件进行异常检测

对于通用系列 MCU，因为本身不集成预驱和运放等电路，所以相关的异常检测都需要通过软件方式实现。根据实际应用的需要，相关的检测可以放在电流环的高频任务里面，也可以放在速度环的低频任务里面。一般来说针对电流过流和 MOS 的异常检测放在电流环的高频任务里面，针对电压过压和欠压以及过温事件的检测放在速度环的低频任务里面。因为实际系统设计中还需要考虑噪声的影响，所以针对异常的监测需要进行适当的滤波处理。

对于电机控制来说，过流保护是一个对时间要求比较高的保护，这里可以采用芯片内部 ACMP 模块配合实现过流的保护，通过设定比较器的阈值并结合相电流或母线电流采样通道来实现过流的保护。在云途 MCU 中都会有几个通道同时支持 ADC 和 ACMP 功能，用户可以用 ADC 进行电流的采样，用 ACMP 进行过流的监测。实际使用中还可以实现电机弦波控制和方波的无感切换。ACMP 模块还支持通过 TMU 模块直接触发 eTMR 的 Fault 保护，eTMR 模块则支持定义在 Fault 事件产生之后的保护行为（具体参考 eTMR 模块 FAULT 寄存器）。

针对供电电压和温度的保护，一般会采用 ADC 进行采样，然后通过软件进行滤波和比较，当电压或者温度超过阈值时，触发异常事件，用户可以在异常事件中进行处理。如果需要进行硬件保护，则可以通过 ADC 的 WDG 功能实现，可以将对应通道开启 WDG 比较功能，当电压或者温度超过阈值时，会触发 WDG 事件，用户可以在 WDG 事件中进行处理。

10 总结

云途电机库提供了基础的 FOC 控制算法，不过对于电机控制来说，实际应用中还需要根据实际应用情况进行参数的调整，以实现更好的控制效果。另外对于电机控制中的异常监测，用户还需要根据实际电机和硬件情况进行调整，以及针对不同的异常响应需求进行调整，电机控制库中相关的配置仅供参考。

11 参考

文章中部分图片来自 AI 生成内容，仅供演示参考。

版权与联系方式

云途半导体及其子公司保留随时对云途半导体产品或者本文档进行更改，更正，增强，修改和改进的权利，恕不另行通知。购买者在下单前应了解产品的最新相关信息。云途的产品根据云途的团队和订单确认时的销售条件进行销售。

购买者对云途产品的选择权，挑选以及使用负全部责任，云途不对购买者的产品设计和应用协助承担任何责任。

云途在此未授予任何知识产权的任何明示或暗示许可。

转售与本文所述信息不同的云途产品将使云途对此类产品的任何保证失效。

云途半导体和云途半导体标志使云途半导体的商标。所有其他产品或服务名称均为其各自所有者的财产。

本文档中的信息将取代并替换之前在本文档的任何先前版本中提供的信息。

©2020 - 2025 江苏云途半导体版权所有

联系我们:

主页: www.ytmicro.com