ARTERY 雅特力

AT32电机库示范讲解
- FOC 霍尔传感器有感控制

3.6 (四) 13:30-15:30 在线培训

主讲：电机应用软件资深工程师 林明赞 博士

雅特力 | AT32 MCU电机库应用培训

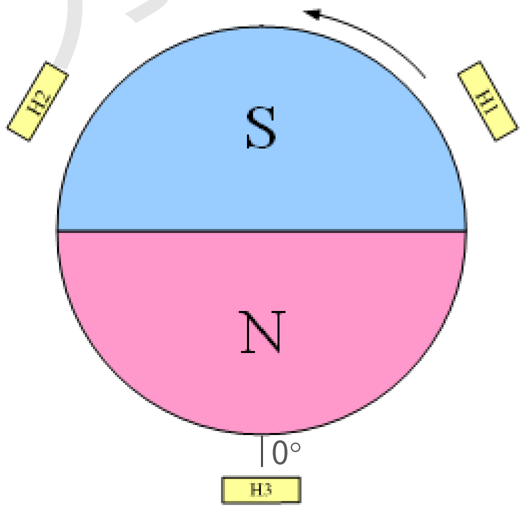
2025.03.06 活动议程

时间	主题	讲师
13:30 -14:30	永磁同步电机FOC霍尔传感器有感控制解说	电机应用软件 资深工程师 林明赞 博士
中场休息		
14:40 -15:30	AT32 电机库FOC霍尔传感器控制快速上手操作解说	电机应用软件 资深工程师 林明赞 博士

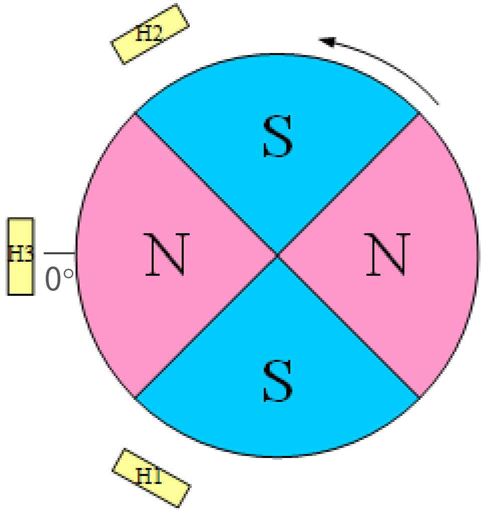
霍尔传感器于FOC控制应用介绍

霍尔传感器感测转子位置

- 三个霍尔传感器安装在定子上，间隔120°电气角，与定子绕组的轴线对齐
- 当转子旋转时，转子磁钢的磁场会通过霍尔传感器，传感器根据磁场的极性输出高电平或低电平
- 霍尔传感器电路通常为开漏型
- 须连接上拉电阻才有输出



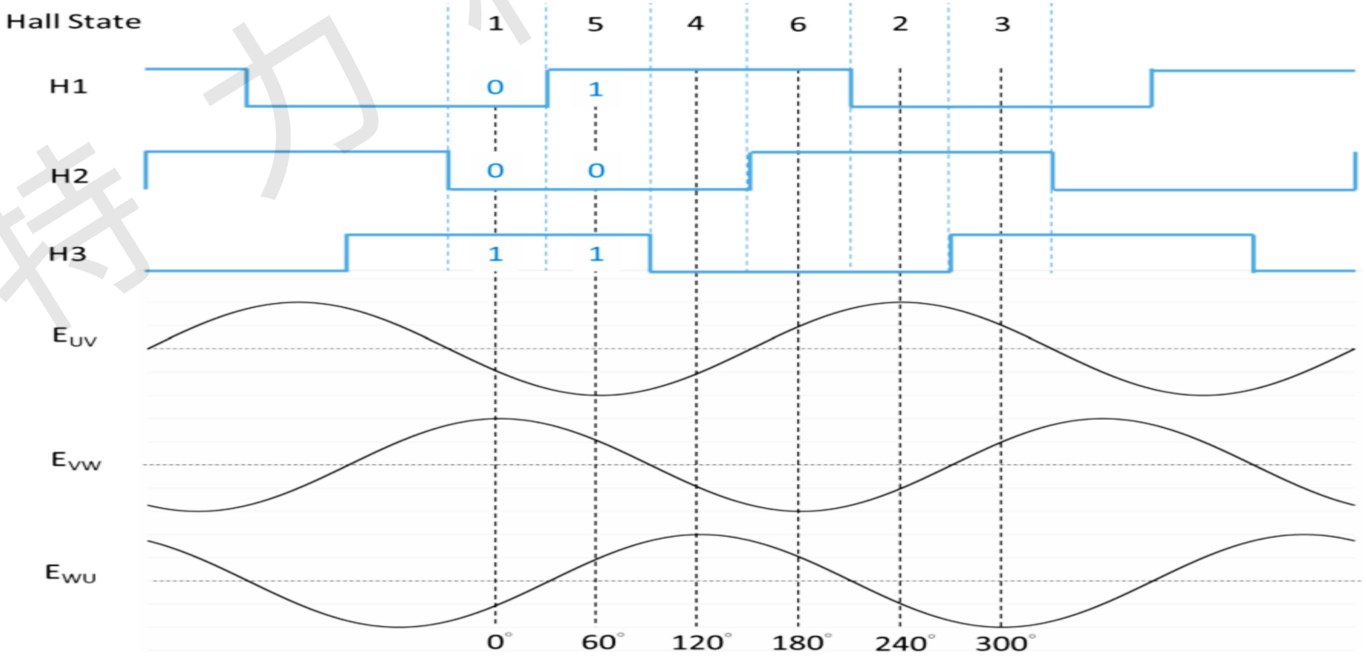
两极转子霍尔传感安装位置



四极转子霍尔传感安装位置

霍尔传感器与反电势线电压和磁极角度关系

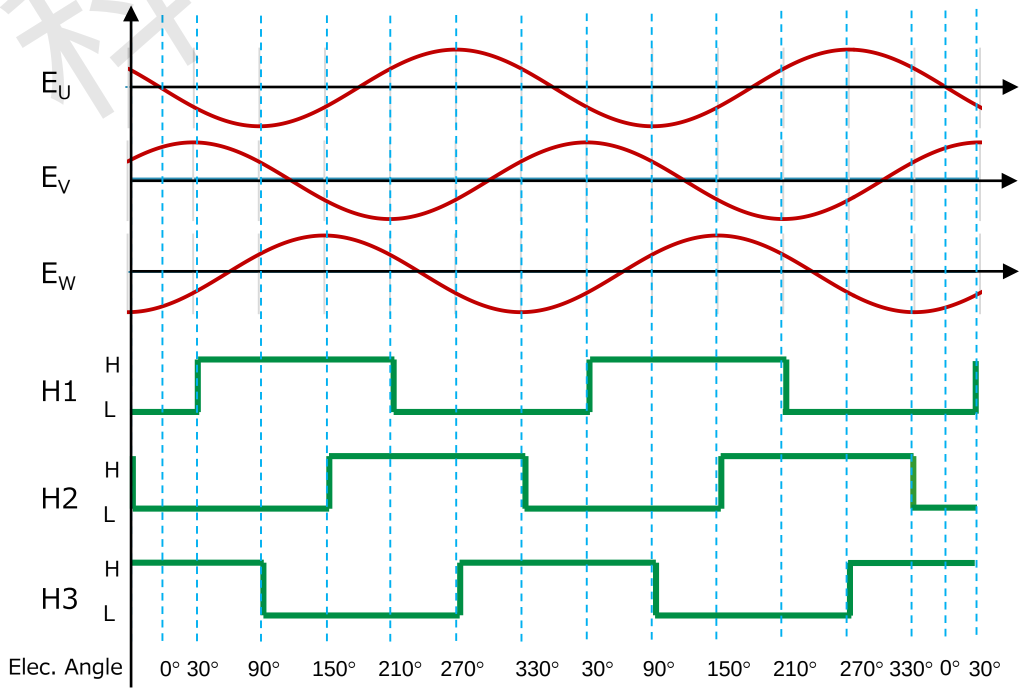
- 三个霍尔传感器的输出组合提供六个独特状态，对应不同的转子位置
- 以每个霍尔正负缘信号点所对应的角度，做为连续转子角度估测误差的修正依据
- 霍尔霍尔正负缘信号变化点，刚好位于反电势线电压的零交越点



PMSM 线电压反电势、霍尔状态与磁极角度的关系图

霍尔传感器与连续磁极角度估测

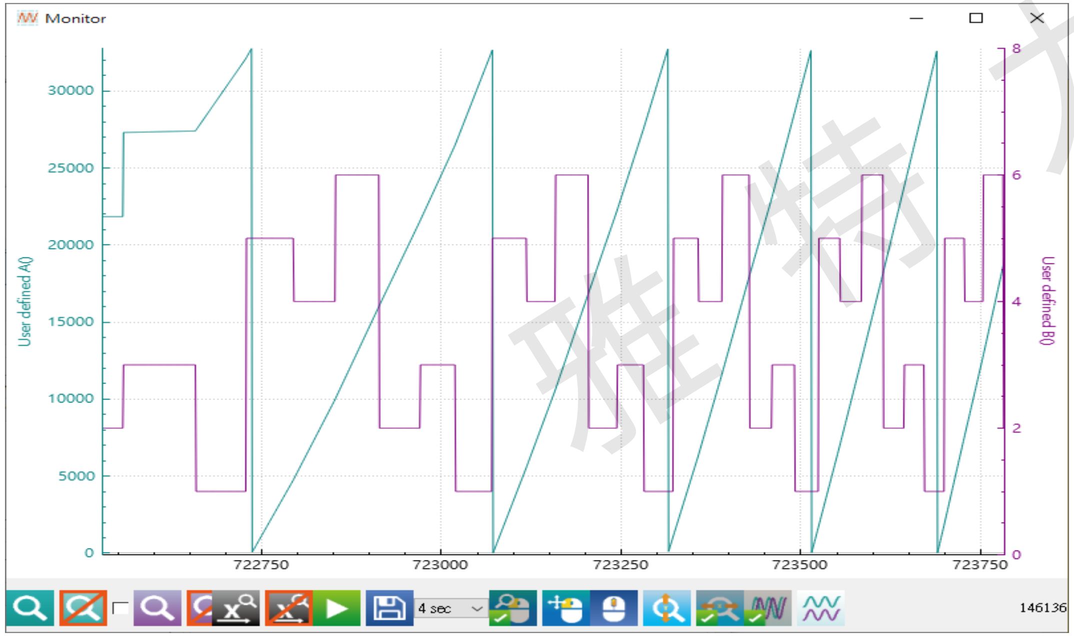
- 采集霍尔信号
  - 读取传感器的输出组合
- 确定初始角度
  - 根据霍尔信号状态确定当前转子电角度
- 估算转速
  - 通过霍尔信号变化周期，估算转速
- 计算连续角度
  - 根据转速和角度更新频率，计算连续角度
- 补偿误差
  - 于霍尔信号变化点，进行角度补偿，消除角度估测误差



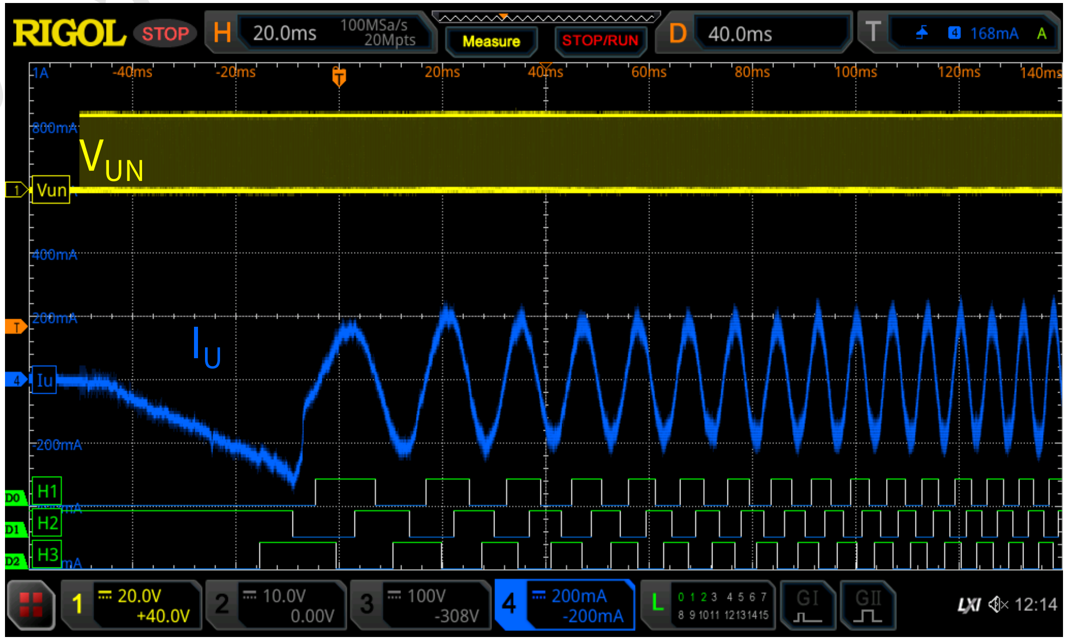
PMSM 相电压反电势、霍尔状态与磁极角度的关系图

基于霍尔传感器的电机磁极角度估测波形

- 根据霍尔信号周期，估算转子速度
- 进而估算下一个霍尔信号转态点的角度递增量
- 于霍尔信号转态点修正估算角度值



霍尔信号与估测转子角度



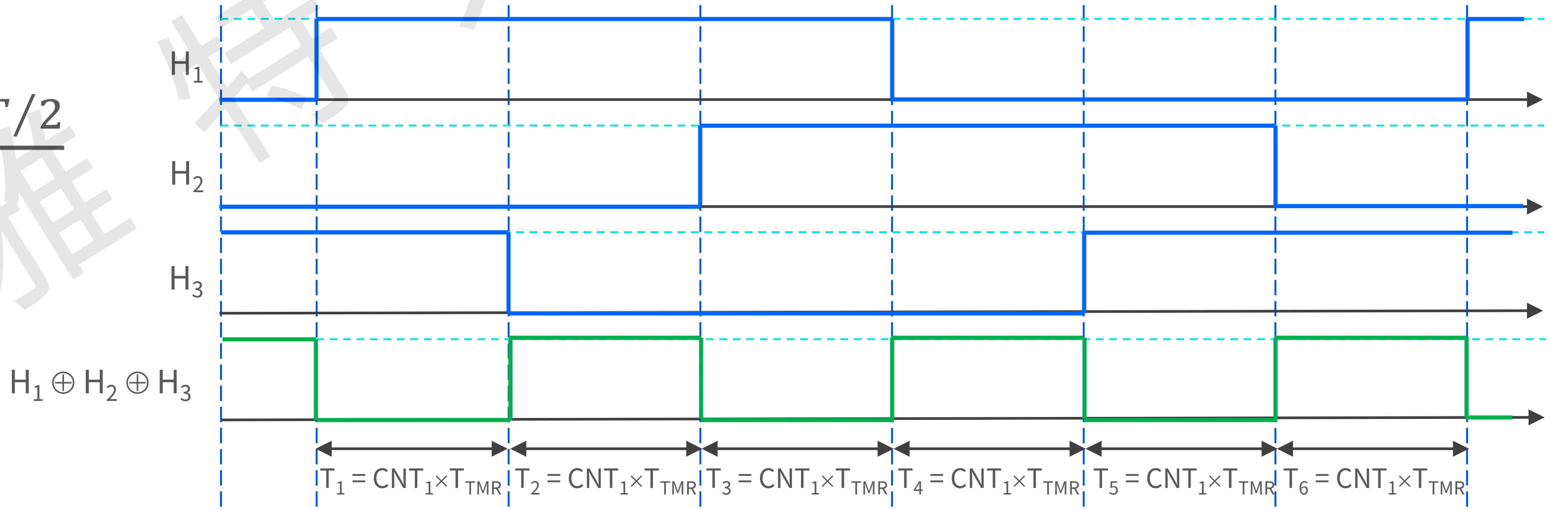
启动时的电流响应波形

转轴速度估测

- 将三相霍尔信号经MCU定时器的斥或功能倍频
- 根据最低转速的霍尔波宽决定补获定时器的计数频率周期T_{TMR}
- 在每个脉波正负缘触发定时器捕获中断同时获取波宽计数值 CNT_x
- 将6次计数值以移动平均计算均值 $CNT_{AVG} = AVERAGE(CNT_1 \sim CNT_6)$
- 计算转速

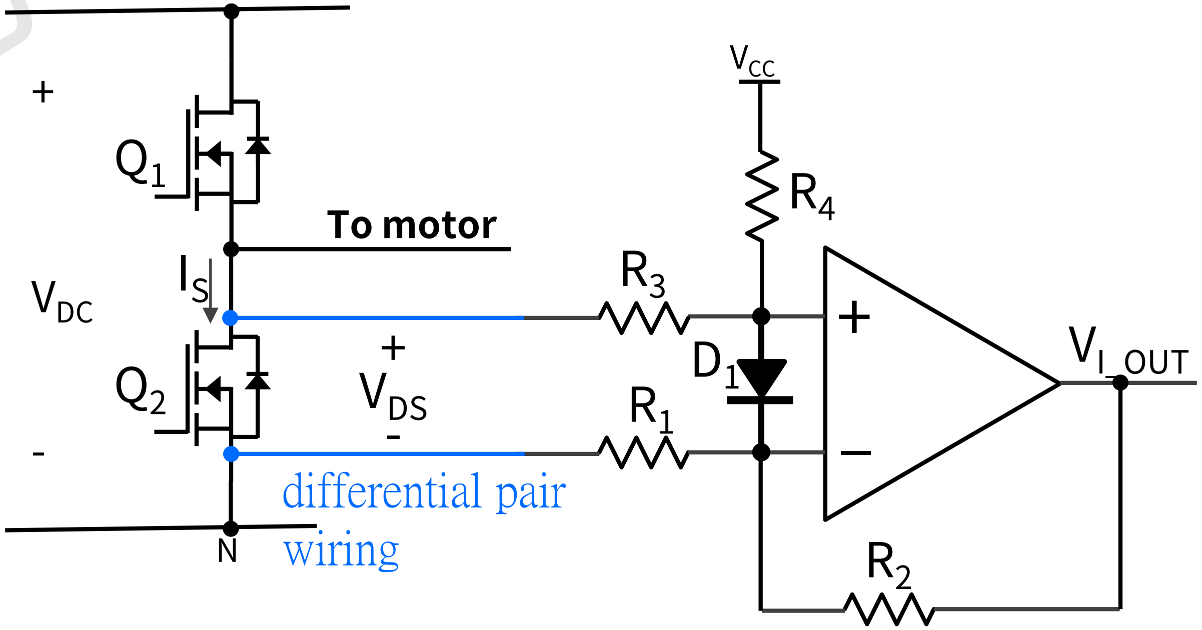
$$SPEED = \frac{MIN\_SPD\_CNT/2}{CNT_{AVG}}$$

- 霍尔信号波宽决定速度取样率



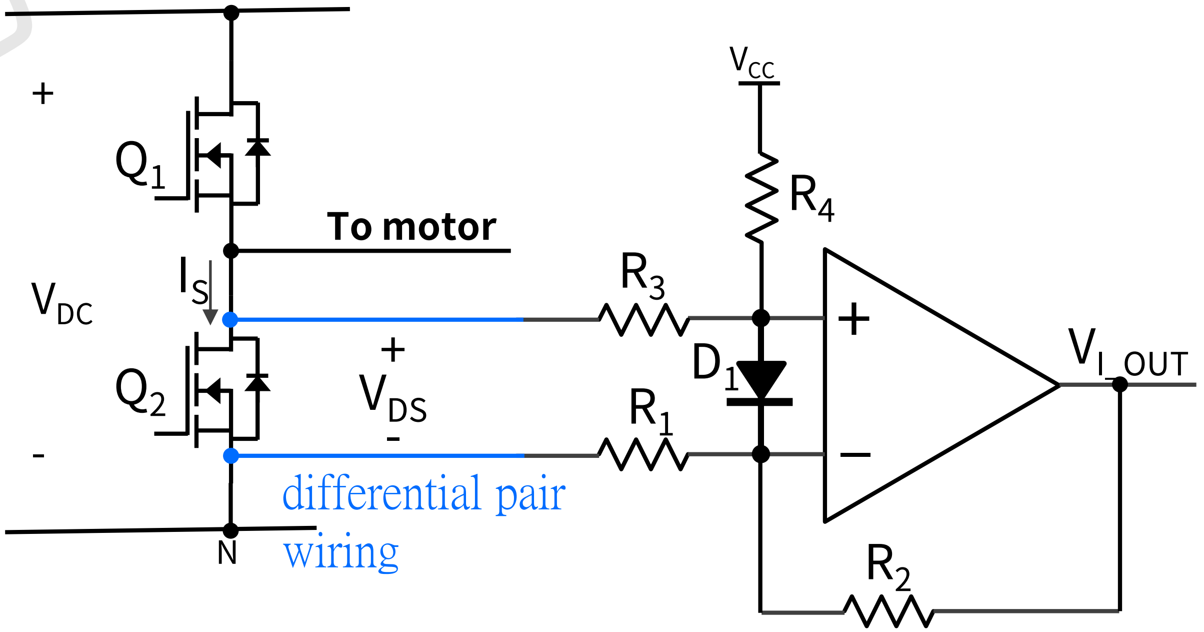
MOSFET下管电流感测

- 当Q_1导通时$V_{DS}= V_{DC}$，
  - 须藉D_1箝位二极管限制OPA输入电压
  - 须选择适当额定功率电阻R_3
  - 分压后的R_1压降须小于OPA最大输入电压
  - 此时OPA输出饱和正电压
- 当Q_1导通时$V_{DS}= I_S \bullet R_{DS_ON}$
  - OPA将V_{DS}电压放大
  - R_4 为偏压电阻，使输出直流准位为$V_{CC}/2$
  - 当Q_2导通时，V_{DS}电压迅速变小
  - OPA的饱和恢复时间须短
  - 选择高带宽与高摆率的OPA



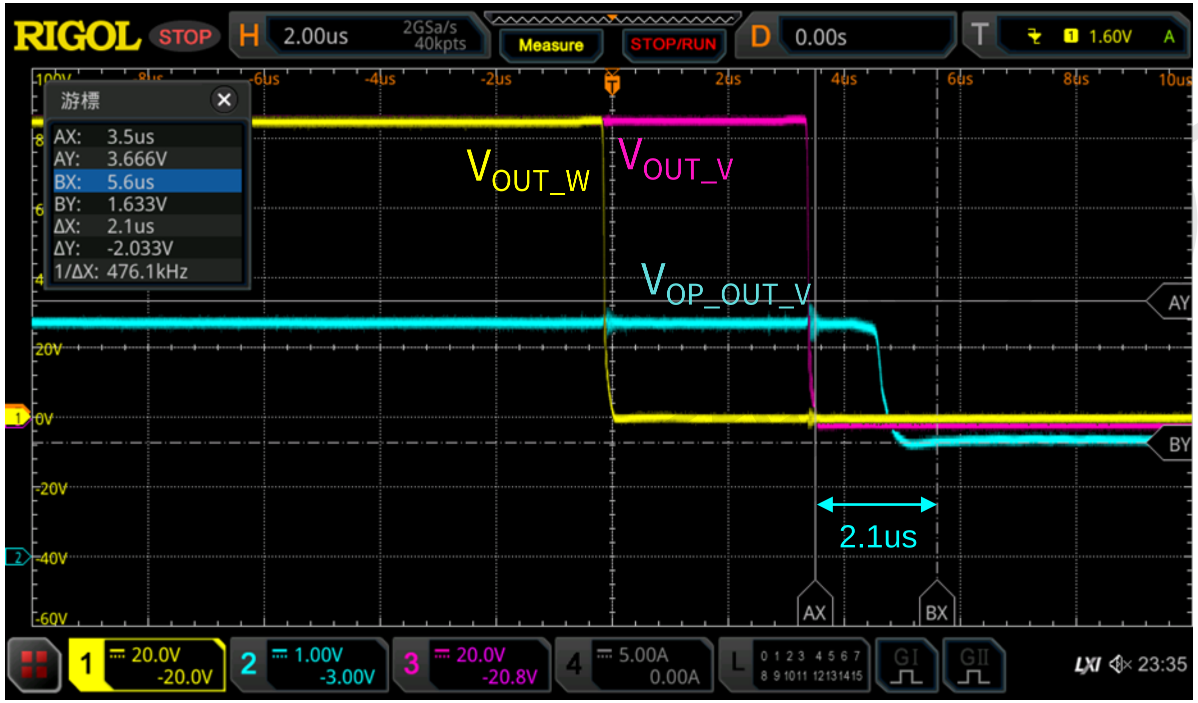
MOSFET下管电流感测

- 优点
  - 成本低廉
  - 容易实现
  - 简化电路
- 缺点
  - MOS内阻易受温升变大，量测失真
  - 下管占空比变窄时取样不易
  - 取样时机准确度要求较高
  - 易受MOSFET切换噪声干扰

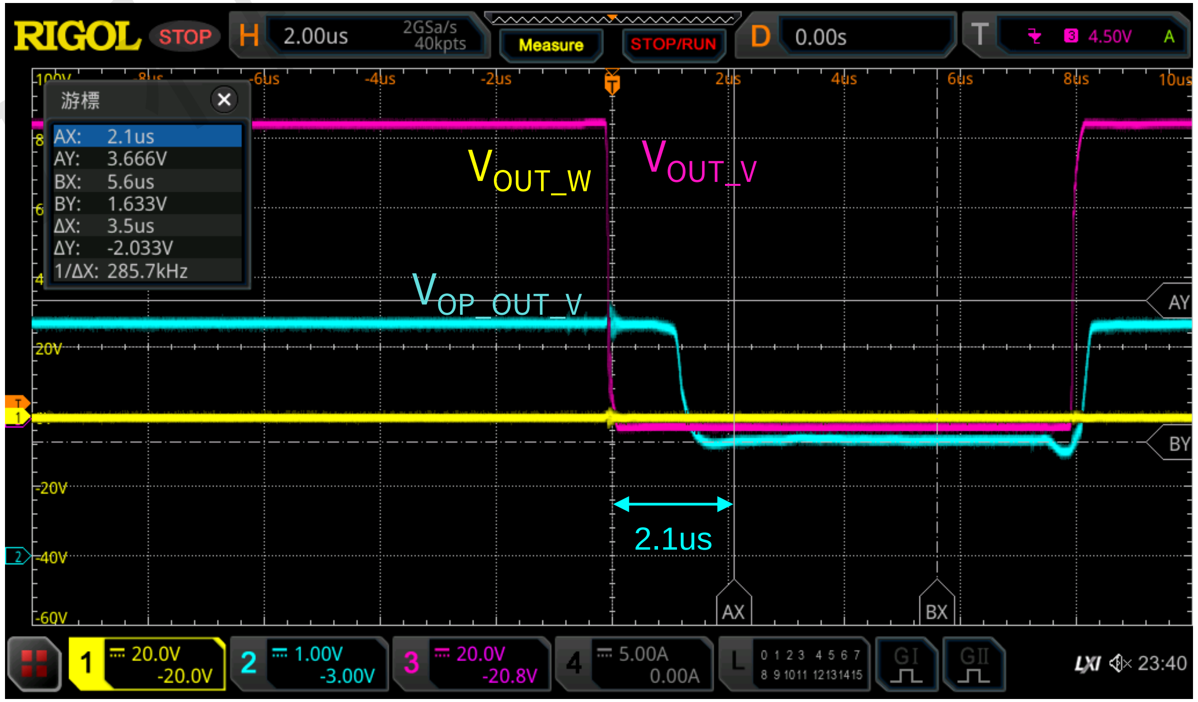


MOSFET下管电流感输出响应波形

- 当MOSFET下管导通时，输出电压为其导通压降
- OPA经饱和恢复时间与稳定时间进入线性放大操作



其它相先关闭时后关相OP的OUT输出电压

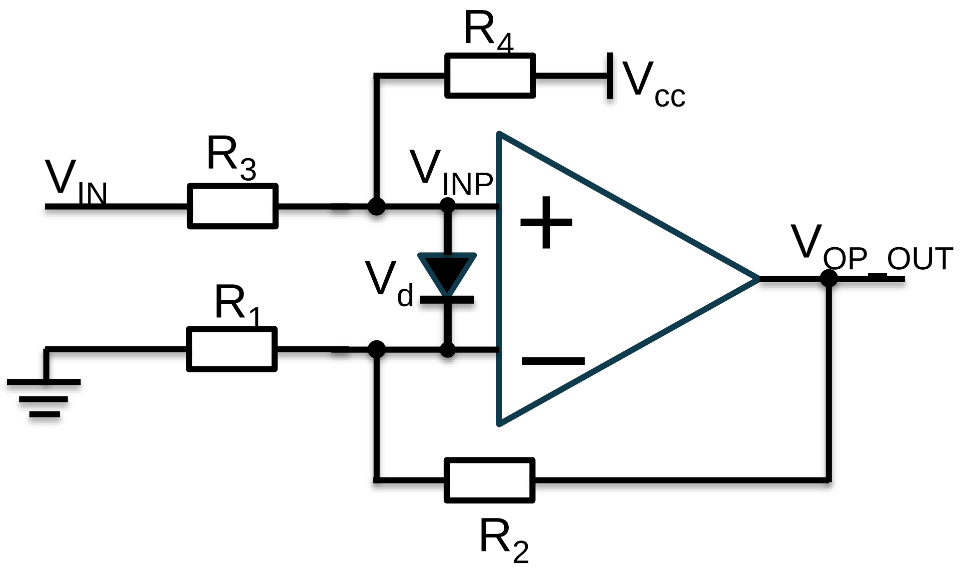


PWM OFF时间很短时OP OUT输出电压

MOS下管电流感测 OP电阻的选定方法

- 决定OP增益，$OP\_Gain = 1 + R_2/R_1$
- 在满足 $R_1 > V_{CC}/\left(OP_{Gain} \cdot I_{OP\_OUT\_LIMIT}\right)$ 条件下，选择R_1
- $R_2 = (OP_{Gain} - 1)R_1$ ；在$V_{INP} < V_{max}$ 限制下选择R_3
- $R_3 = (V_{IN} - V_{INP})R_1R_2/(V_{INP}(R_1 + R_2) - V_d(R_1 + R_2) - V_{CC}R_1)$
- $R_4 = R_3(1 + 2R_2/R_1)$
- 整体电路增益

$$Total\_Gain = (R_4/(R_3 + R_4))(1 + R_2/R_1)$$

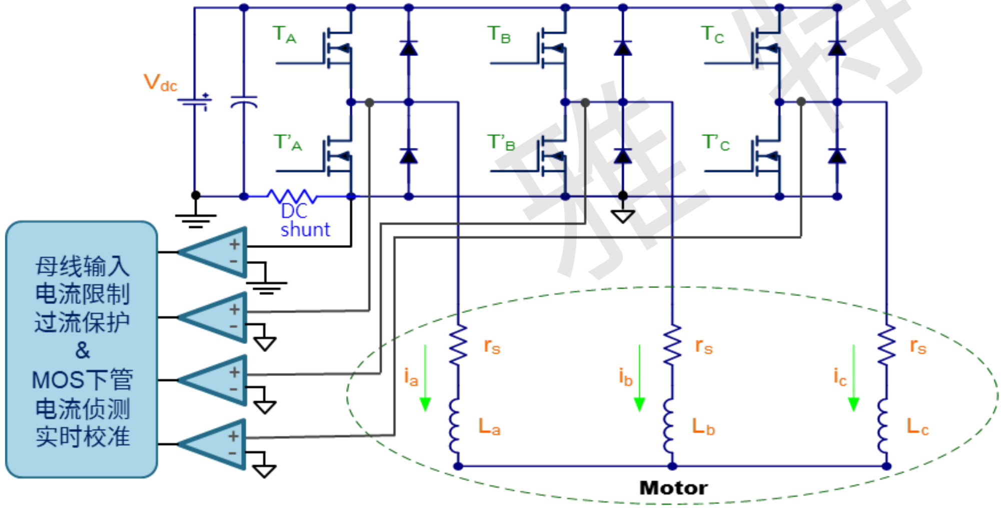


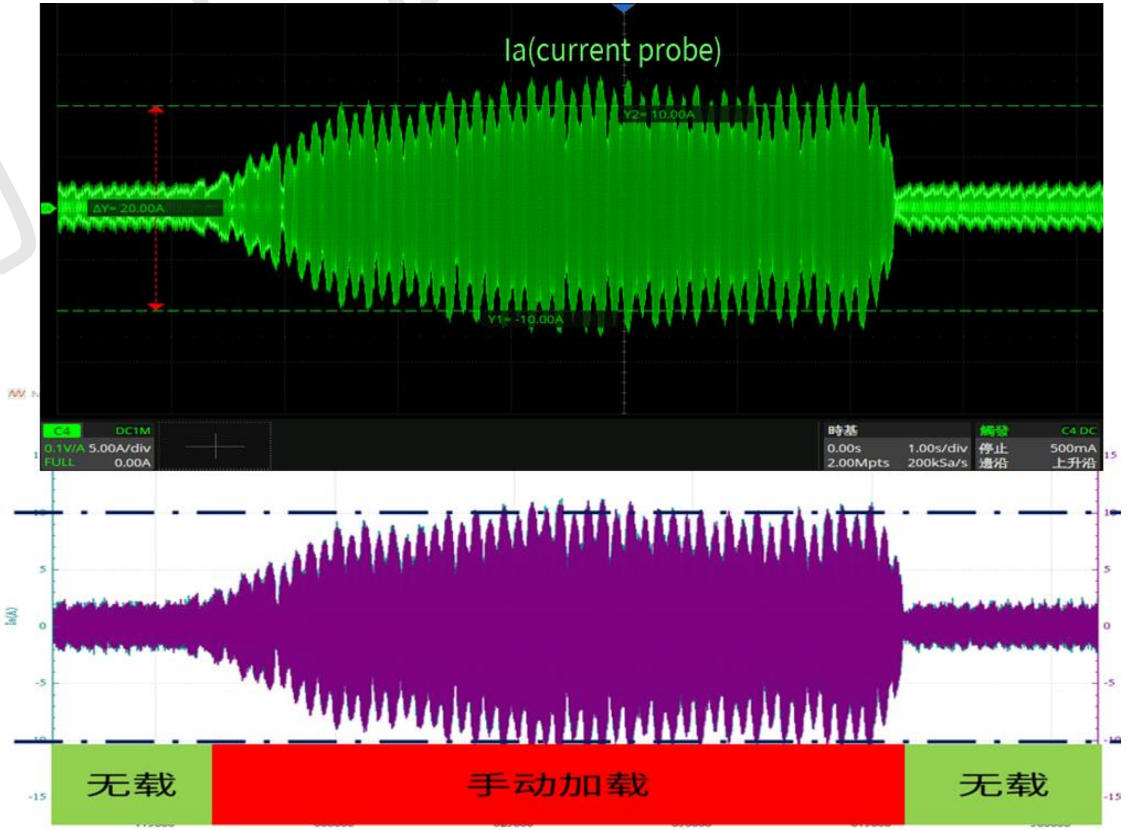
MOSFET下管电流侦测与补偿技术

- 电路简单、低成本
- MOS温升影响电流取样准确度
- 以母线电流感测电阻回授实时校准
- 兼具输入电流限制与过流保护功能

设定初始参数R_{DSON}为 = 4mΩ

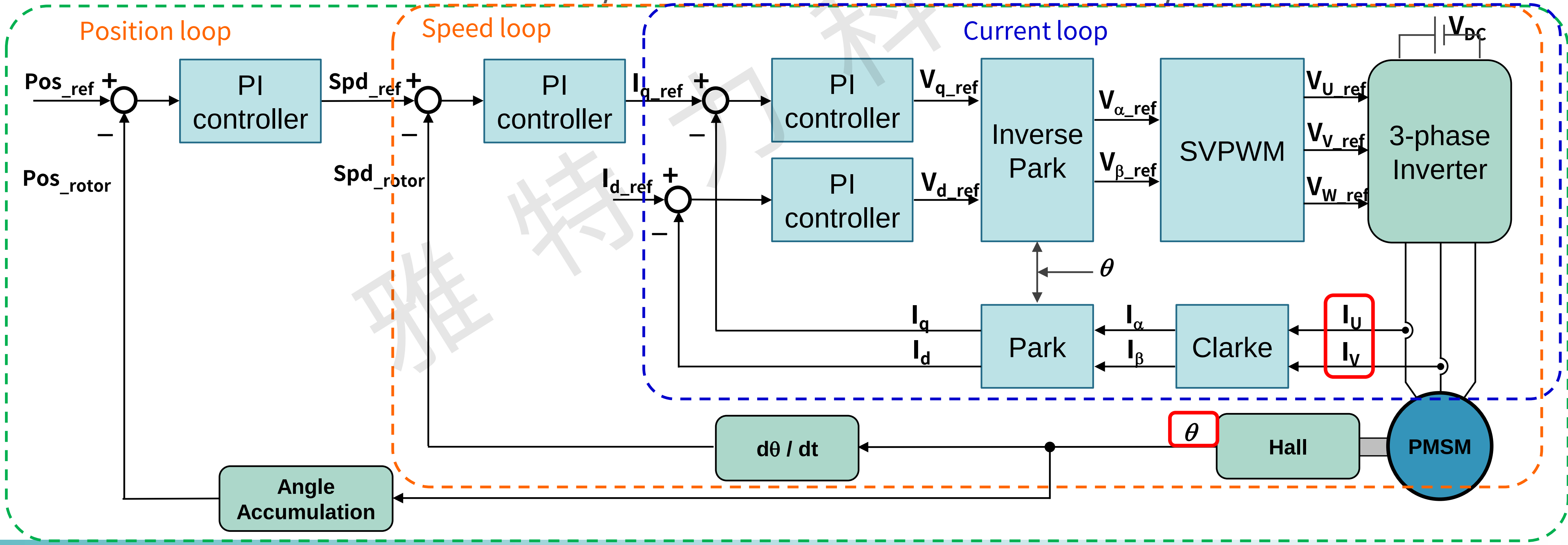
实际MOS R_{DSON}= 8mΩ，自动补偿回归正确阻值





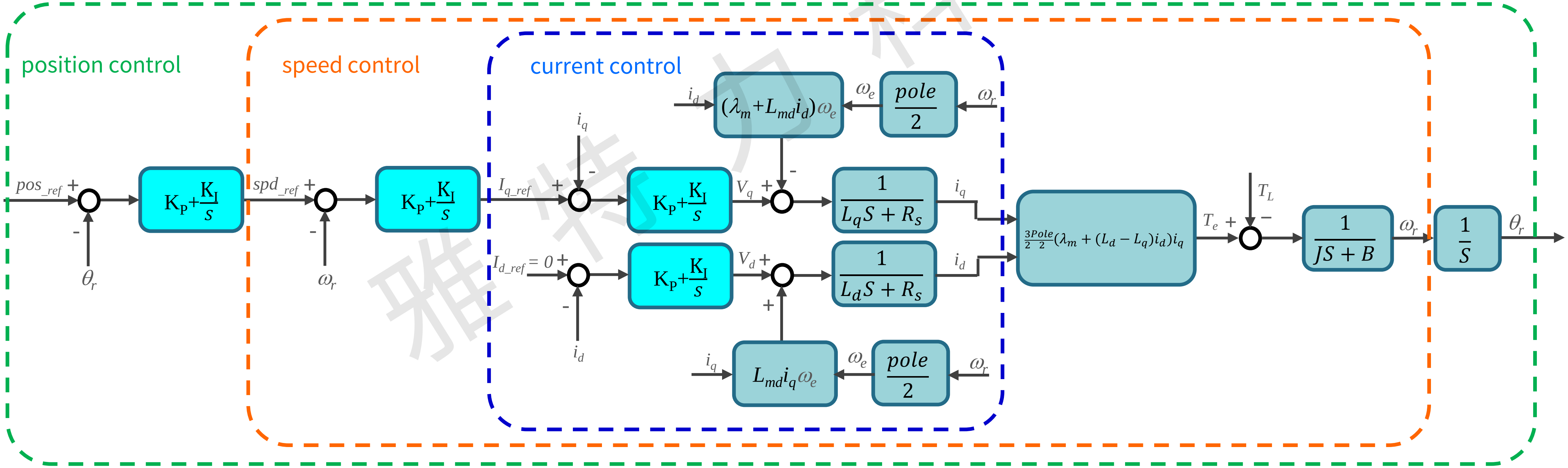
FOC磁场导向控制系统

- 经由霍尔状态估算转子角度，电流感测电阻/MOSFET内阻感测电流
- 构成FOC电流环后，根据速度/角度误差与PI控制器建立速度/位置环



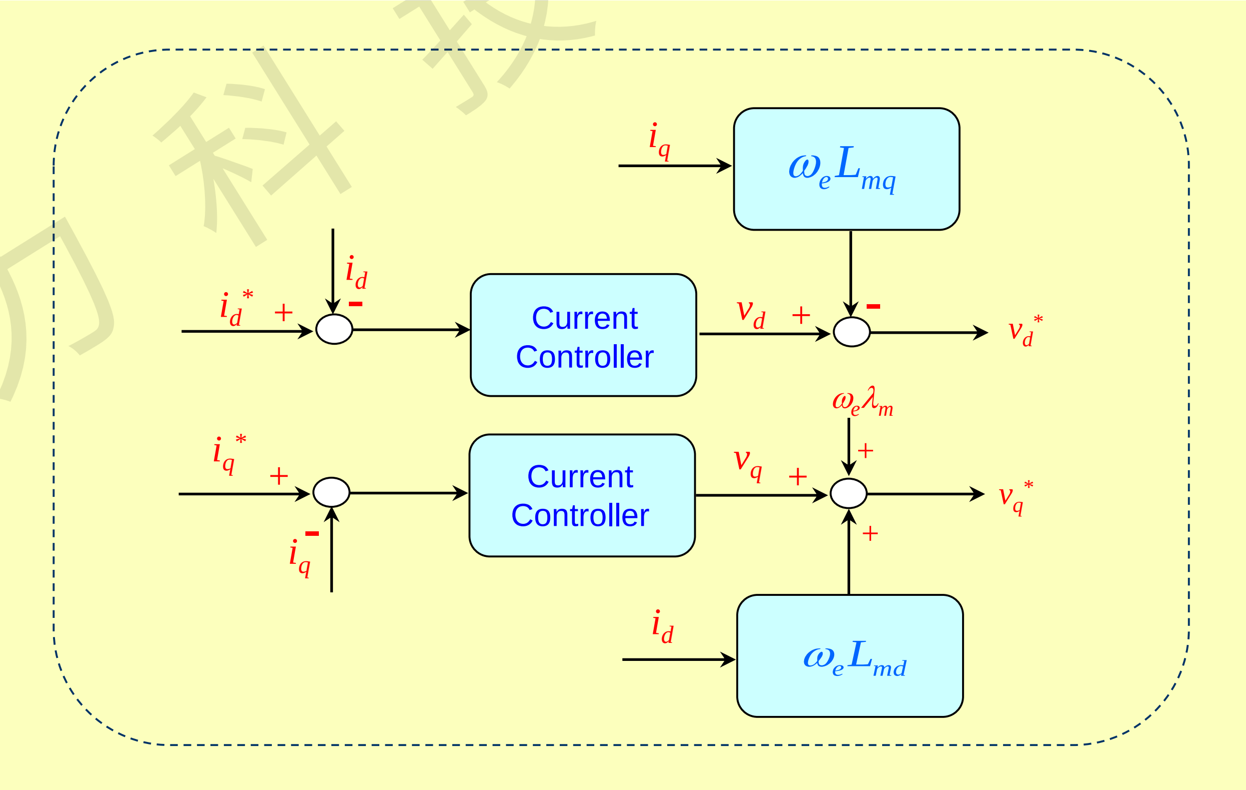
PMSM控制系统转移函数方块图

- 根据永磁同步电机电气方程式与转轴动态方程式可表示如下转移函数方块图
- 参数调整先调整电流环，再调整速度环，最后调整位置环



电流环解耦控制

- 电机d-q轴电压包含线圈电阻与电感压降以及d-q轴耦和的反电势
- 电流控制器输出电压须扣除反电势电压后才能作用于线圈电阻与电感
- 造成电流瞬时响应不实时，在高速时尤为明显
- 可在电流控制器输出电压分别加上d轴与q轴的反电势补偿量
- 让电流控制器的输出，直接作用于线圈阻抗，有利于瞬时响应与参数调整



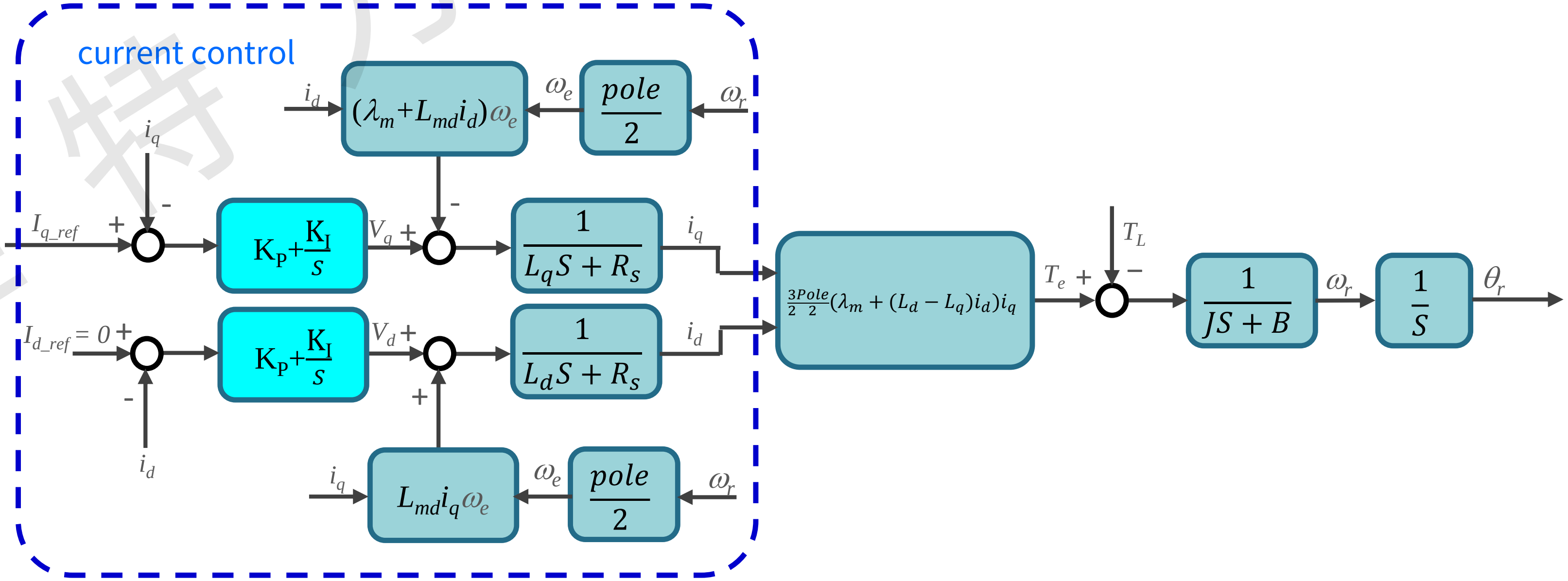
电流环参数调整方法

- 电流环取样频率(PWM切换频率)须为线圈阻抗带宽($\frac{R}{2\pi L}$)的10倍以上
- 电流PI参数自整定的电流带宽参数
  - CURRENT_BANDWIDTH (rad/s) = CURRENT_BANDWIDTH /(2π) Hz
- 当取样频率远高于电流频宽时，取样后的离散系统几乎不影响原有的带宽
- 例如

CURRENT_BANDWIDTH = 3500 rad/s
BW = 3500 / (2π) = 557 Hz

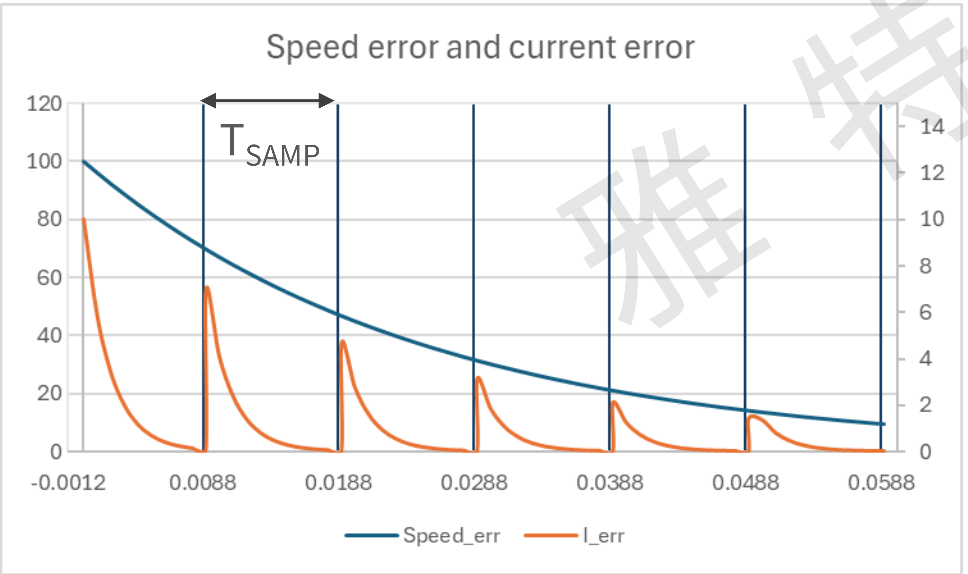
- 电流上升时间 t_r

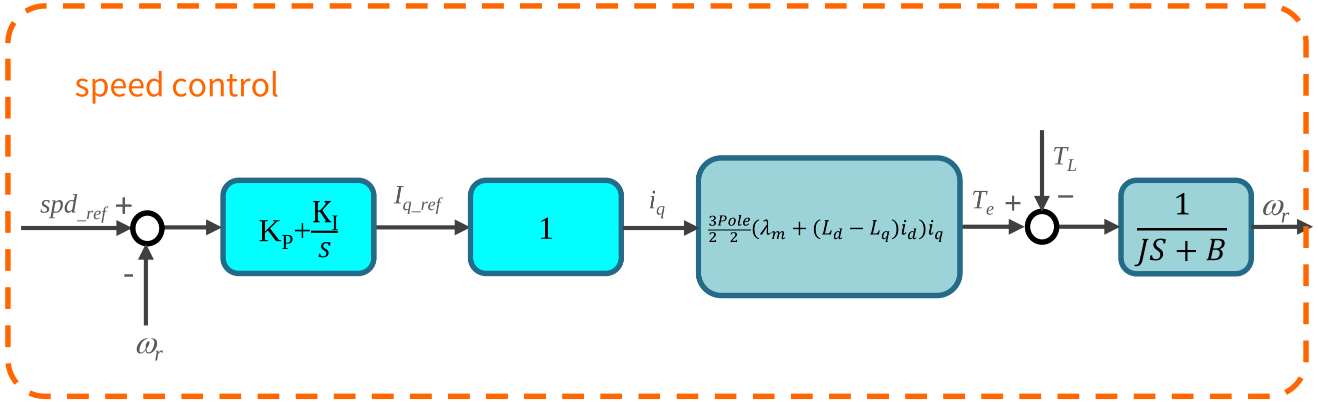
t_r = 0.35 / BW = 0.628ms



速度环参数调整方法

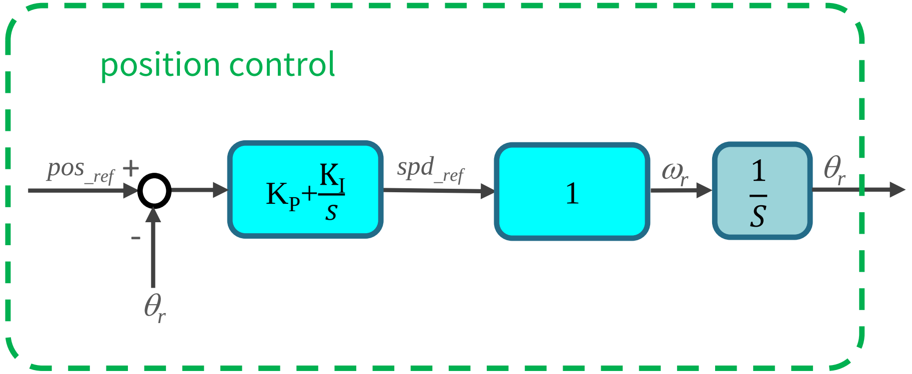
- 电流环带宽须大于速度环带宽5 ~ 10倍
- 对速度环而言，在每次取样点 $I_{d\_ref} \approx i_d, I_{q\_ref} \approx i_q$
- 此时可视电流环增益为 1
- 电机转移函数变成一阶函数，虽然负载转矩可能变动，但仍可容易调整PI参数





位置环参数调整方法

- 速度环带宽须大于位置环带宽5 ~ 10倍
- 在惯量不变无负载情况下，对位置环而言，在每次取样点 $spd_{ref} \approx \omega_r$
- 此时可视电流环增益为 1
- 电机转移函数变成一个积分函数，可容易调整PI参数
- 实际应用时负载可能会变动，简化模型可能失真，PI参数以实际调整为准

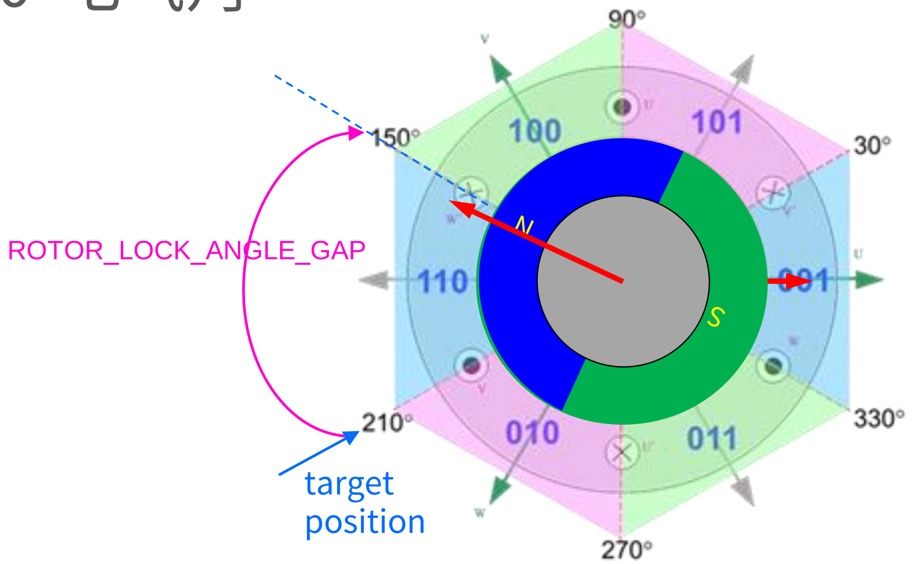


霍尔传感器定位控制

- 霍尔传感器的分辨率为60°电气角，适用于定位精度要求低的场合
- 启动时转子初始角设定于该霍尔区间的中间
- 定位误差小于ROTOR_LOCK_ANGLE_GAP时则三相短路，锁定转子于误差范围内
- ROTOR_LOCK_ANGLE_GAP须大于等于60°电气角
- 须开启低速电压控制设定

```
#define LOW_SPEED_VOLT_CTRL
```

```
/* low speed voltage control parameter */
#define HYSTERESIS_LOW_SPEED               (300)
#define HYSTERESIS_HIGH_SPEED              (400)
```



霍尔有感电机库架构解说

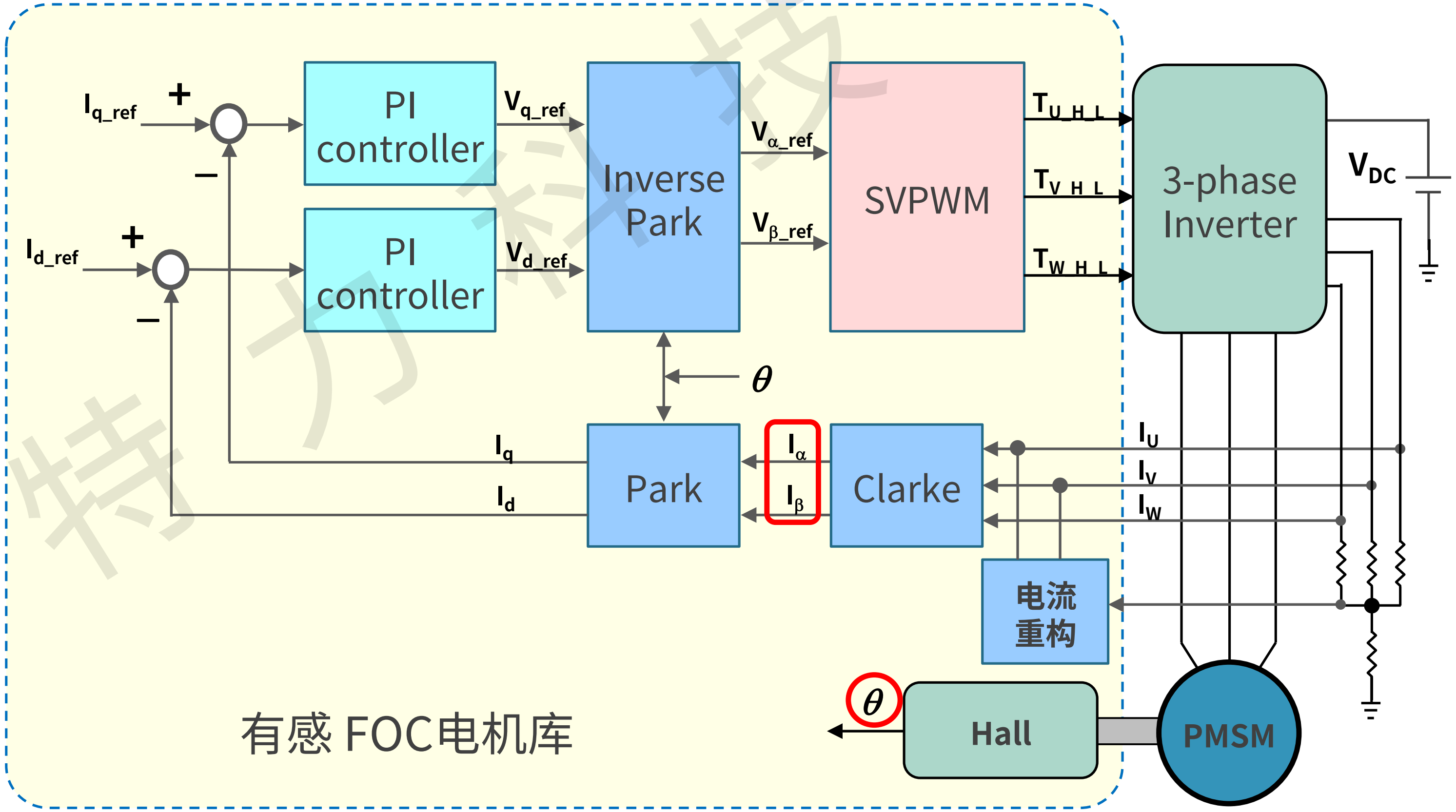
电机库 – E-Bike霍尔传感器有感 FOC

■ 转子位置检测方式
  ➢ 霍尔效应位置传感器

■ 相电流检测方法
  ➢ 三电阻电流检测
  ➢ 双电阻电流检测
  ➢ 单电阻电流检测与电流重构
  ➢ MOS下管电流检测

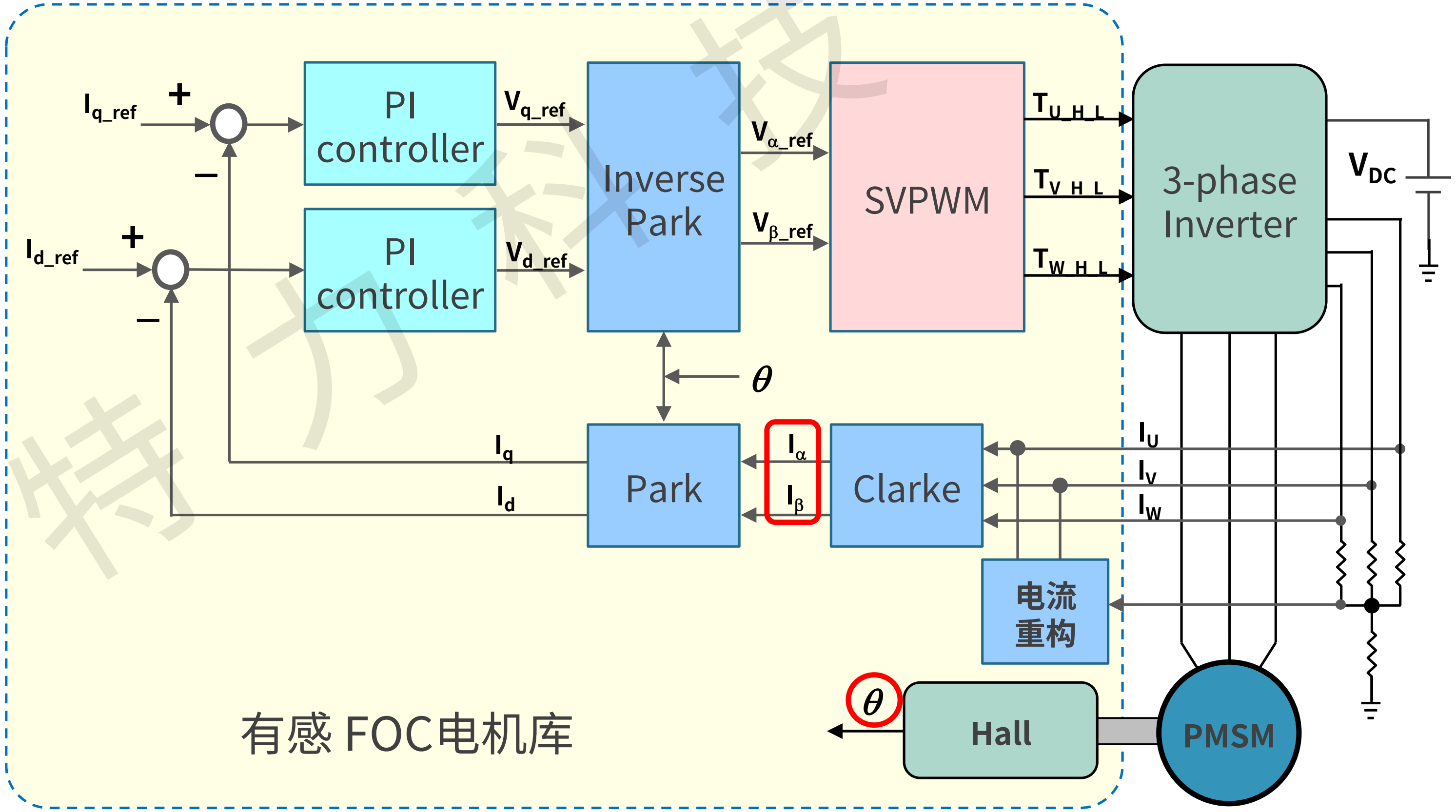
■ 可实现有传感器弦波控制方法
  ➢ 电压矢量控制
  ➢ 转矩控制 (电流矢量控制)
  ➢ 转速控制 (限速)
  ➢ 弱磁控制

■ E-Bike控制功能
  ➢ 母线电流限制
  ➢ MOS内阻自动校准
  ➢ 刹车回充
  ➢ 倒车控制



电机库 – 霍尔传感器有感 FOC

- 转子位置检测方式
  - 霍尔效应位置传感器
- 相电流检测方法
  - 三电阻电流检测
  - 双电阻电流检测
  - 单电阻电流检测与电流重构
- 可实现有传感器弦波控制方法
  - 电压矢量控制
  - 转矩控制 (电流矢量控制)
  - 转速控制
  - 弱磁控制
  - 定位控制
  - 回生刹车



PMSM电机库控制技术与应用方案

应用产品		高速风机	低速风机	压缩机	电动工具	滑板车 电动自行车	电摩	定位云台	轮式机器人
矢量技术	磁场导向控制 FOC	●	●	●	●	●	●	●	●
	弱磁控制			●	●	●	●	●	●
控制回路	转矩/速度控制	●	●	●	●	●	●	●	●
	定位控制							●	●
电流感测	3 or 2电阻电流感测	●	●	●	●	●	●	●	●
	单电阻电流感测	●	●	●	●	●		●	●
有传感器	霍尔传感器	●	●		●	●	●		●
	增量编码器				●		●		●
	磁编码器							●	●
无传感器	转子初始角度侦测	●	●	●	●				
	反电势角度估测	●	●	●	●	●	●		

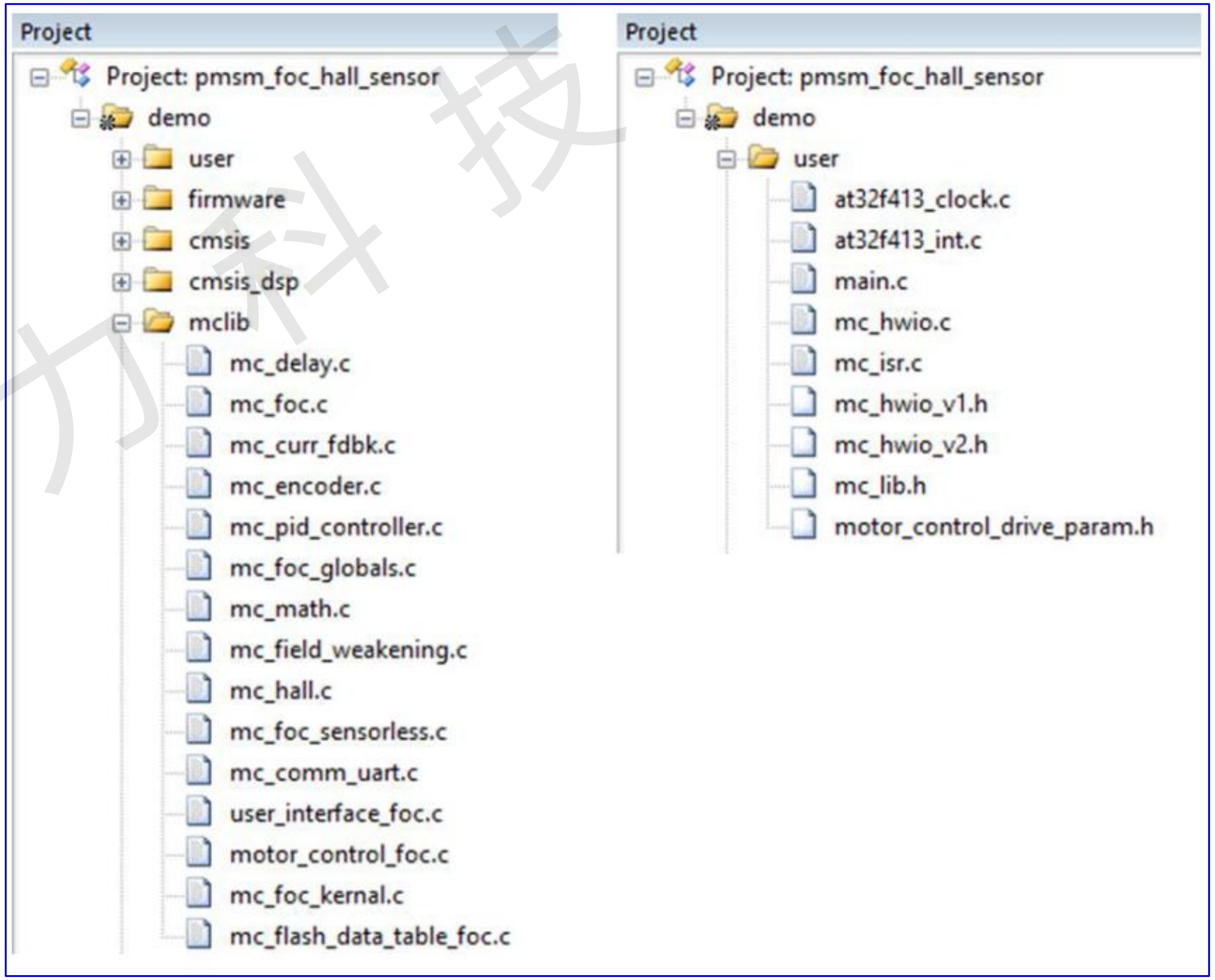
PMSM有感工程专案文件说明

PMSM FOC/BLDC 共通文档	
mc_lib.h	头文件统一管理
motor_control_drive_param.h	用户定义电机驱动架构模式 、电机参数、控制参数、驱动器参数
mc_hwio.c	硬件外设配置
mc_hwio_v1.h	硬件IO接口宏定义配置(电机开发版AT-MOTOR-EVB V1.x)
mc_hwio_v2.h	硬件IO接口宏定义配置(电机开发版AT-MOTOR-EVB V2.x)
mc_isr.c	相关电机控制中断函数
mc_type.h	全局变量类型定义、枚举定义
mc_delay.c	时间延迟相关函数
mc_delay.h	时间延迟相关函数宣告
mc_comm_uart.c	通讯界面相关外设配置
mc_comm_uart.h	通讯uart相关函数宣告、配置
mc_pid_control.c	PID控制器相关函数
mc_pid_control.h	PID控制器相关函数宣告
mc_curr_fdbk.c	电流检测相关函数
mc_curr_fdbk.h	电流检测相关函数宣告
mc_math.c	滤波器相关函数
mc_math.h	滤波器相关函数宣告
mc_hall.c	霍尔传感器相关函数
mc_hall.h	霍尔传感器相关函数宣告

PMSM FOC专用文档	
mc_foc_kernal.lib	电机库核心函数(Keil专用)(适用于芯片有FPU)
mc_foc_kernal_noFPU.lib	电机库核心函数(Keil专用)(适用于芯片无FPU)
libmc_foc_kernal.a	电机库核心函数(AT32IDE专用)(适用于芯片有FPU)
libmc_foc_kernal_noFPU.a	电机库核心函数(AT32IDE专用)(适用于芯片无FPU)
mc_foc_kernal.h	电机库核心函数宣告
motor_control_foc.c	电机控制相关函数
motor_control_foc.h	电机控制相关函数宣告
mc_foc.c	FOC相关函数
mc_foc.h	FOC控制相关函数宣告
mc_hall.c	霍尔传感器相关函数
mc_hall.h	霍尔传感器相关函数宣告
mc_field_weakening.c	弱磁控制相关函数
mc_field_weakening.h	弱磁控制相关函数宣告
mc_foc_globals.c	全局变量定义与默认值设定、全局函数定义
mc_curr_decoupling.c	电流解耦控制相关函数
MTPA_MTPV_table.c	MTPA_MTPV控制查表相关函数
mc_foc_globals.h	全局变量、全局函数宣告、宏定义
user_interface_foc.c	通讯界面相关函数
user_interface_foc.h	通讯界面相关函数宣告
mc_flash_data_table_foc.c	写入flash参数表
mc_flash_data_table_foc.h	写入flash参数表的相关配置

霍尔有感控制专案工程结构

- user 文件夹
  – 主程序、外设规划程序以及参数定义头文件
- firmware 文件夹
  - MCU 外设驱动程序
- cmsis文件夹
  - CMSIS DSP函数程序
- mclib 文件夹
  – 为电机库程序包含PI控制函数、霍尔有感电机库函数、全局变量设定与通讯函数等等



霍尔有感控制程序流程图

mc_hwio.c

- 中断致能与优先序设定
- PWM定时器初始化
- 霍尔传感器I/O初始化
- 霍尔传感器定时器初始化
- 速度/位置环定时器初始化
- adc一般通道初始化
- adc抢占通道初始化
- gpio led 初始化
- 呼叫uart 初始化函数
- 按钮I/O初始化
- 霍尔自学习中断初始化
- 3.3Vref 校准函数
- 电机参数鉴定初始化

main.c

相关外设初始化

3.3Vref 校准

UART 外设初始化

电流零位校准
(motor_control_foc.c)

控制参数初始化
(motor_control_foc.c)

状态机变换前初始设定

mc_isr.c

ADVTMR_PWM_CYCLE_BRK_IRQ
主中断：电流环计算、PWM更新、监测数据更新、电机参数鉴定、霍尔传感器角度估算、ADC一般通道触发、PWM BRKIN输入中断处理

SPEED_LOOP_TIMER_IRQ
位置环/速度环控制中断函式

ADC_SHUNT_SAMP_READY_IRQ
电流取样中断函式

HALL_CAPTURE_IRQ
霍尔信号捕获处理中断函式

BUTTON_EXINT_IRQ
启动/停止按钮、霍尔自学习按钮

SysTick_Handler
状态机处理、过压欠压、MOS过温、电位计输入、刹车信号

mclib files

- mc_curr_fdbk.c
- mc_foc.c
- mc_hall.c
- mc_pid_controller.c
- mc_foc_globals.c
- mc_math.c
- mc_user_interface_foc.c
- mc_motor_control_foc.c
- mc_delay.c
- mc_flash_data_table.c
- mc_foc_kernel_noFPU.lib

mc_comm_uart.c

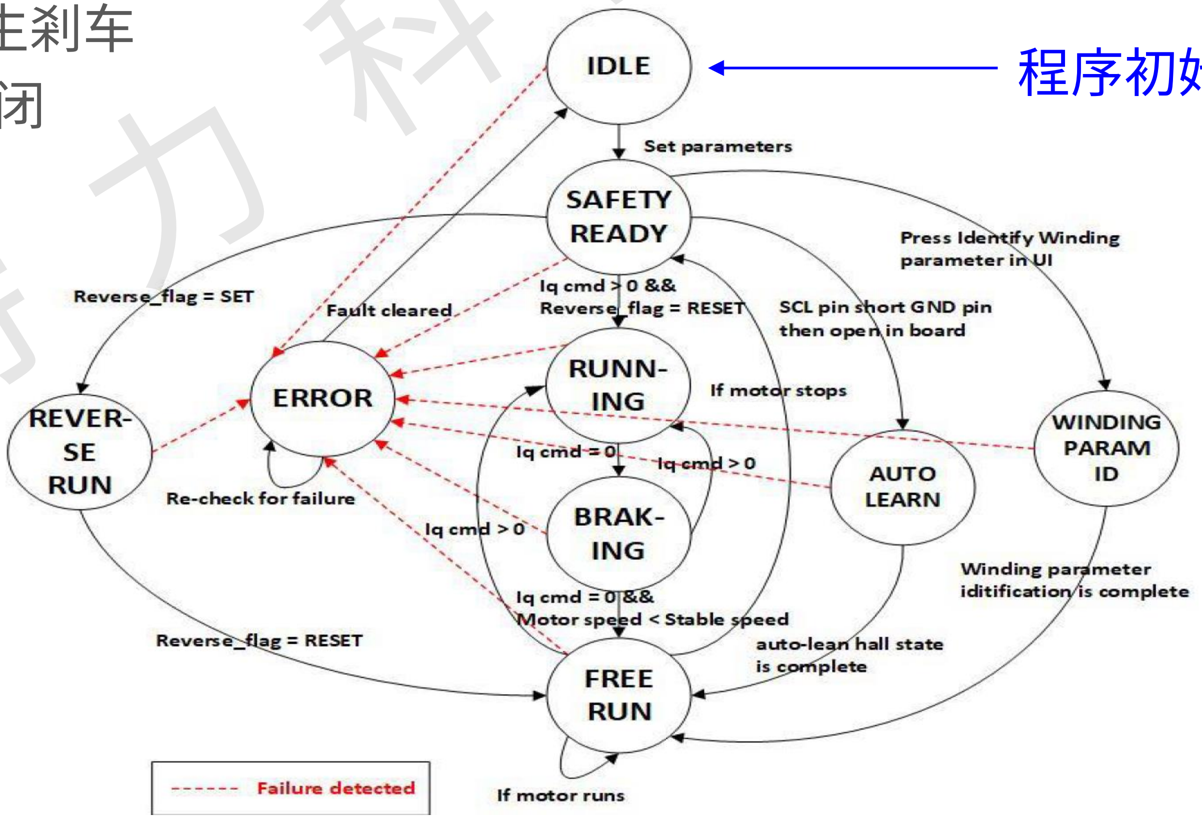
- uart初始化
- uart tx dma初始化

COMM_UART_IRQHandler
uart rx 中断函式

程序状态机流程图

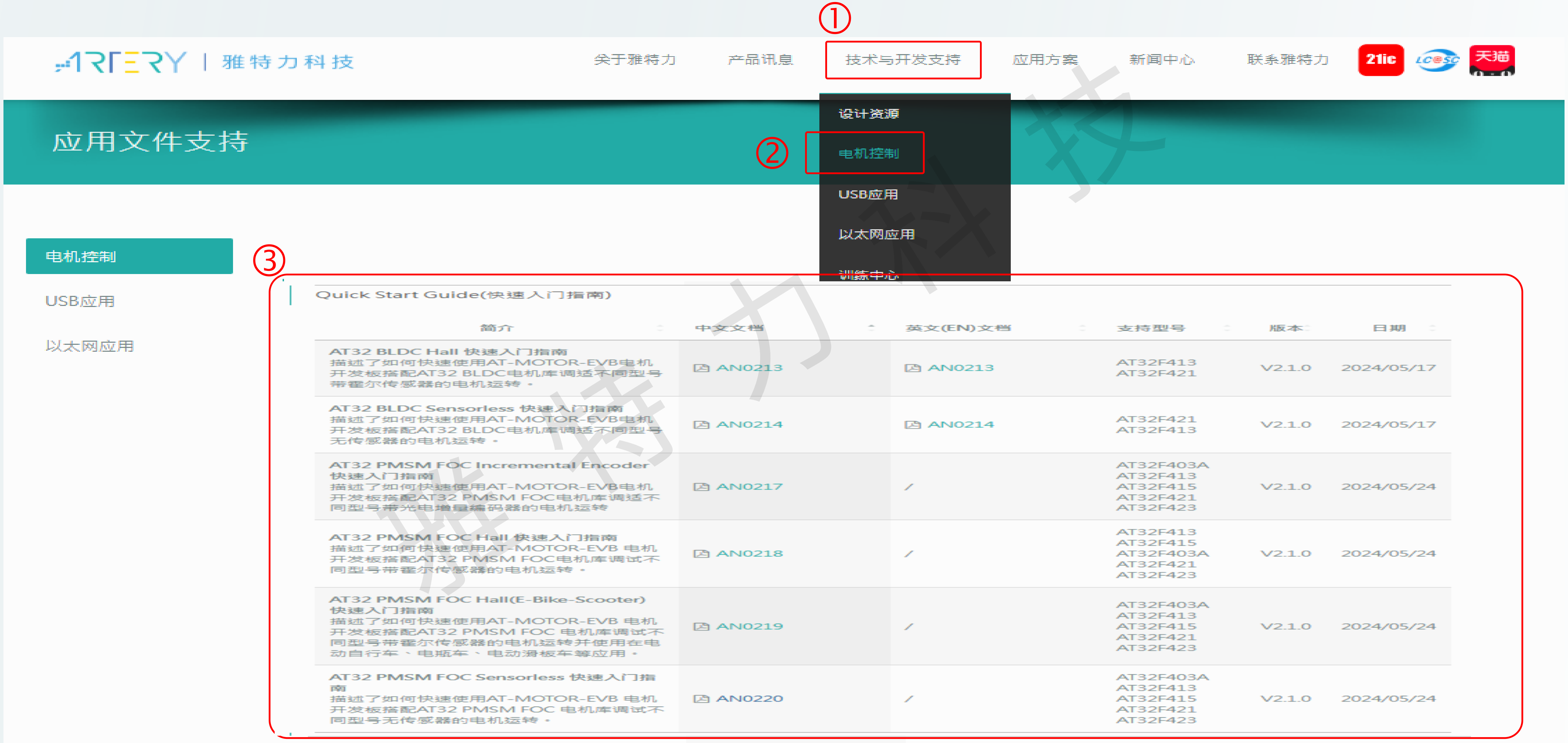
- 于外部命令输入模式，参数初始化及电位计归零后进入SAFETY_READY状态
- 电位计(油门)控制电流(扭力)进入Running运转模式
- 电位计输入零进入BRAKING模式回生刹车
- 转速低于STABLE_SPEED时PWM关闭
  进入FREE_RUN
- 于SAFETY_READY状态，按倒车钮
  电机以固定扭矩倒转并限速

(STABLE_SPEED:可稳定估算角度的速度)



程序初始进入点

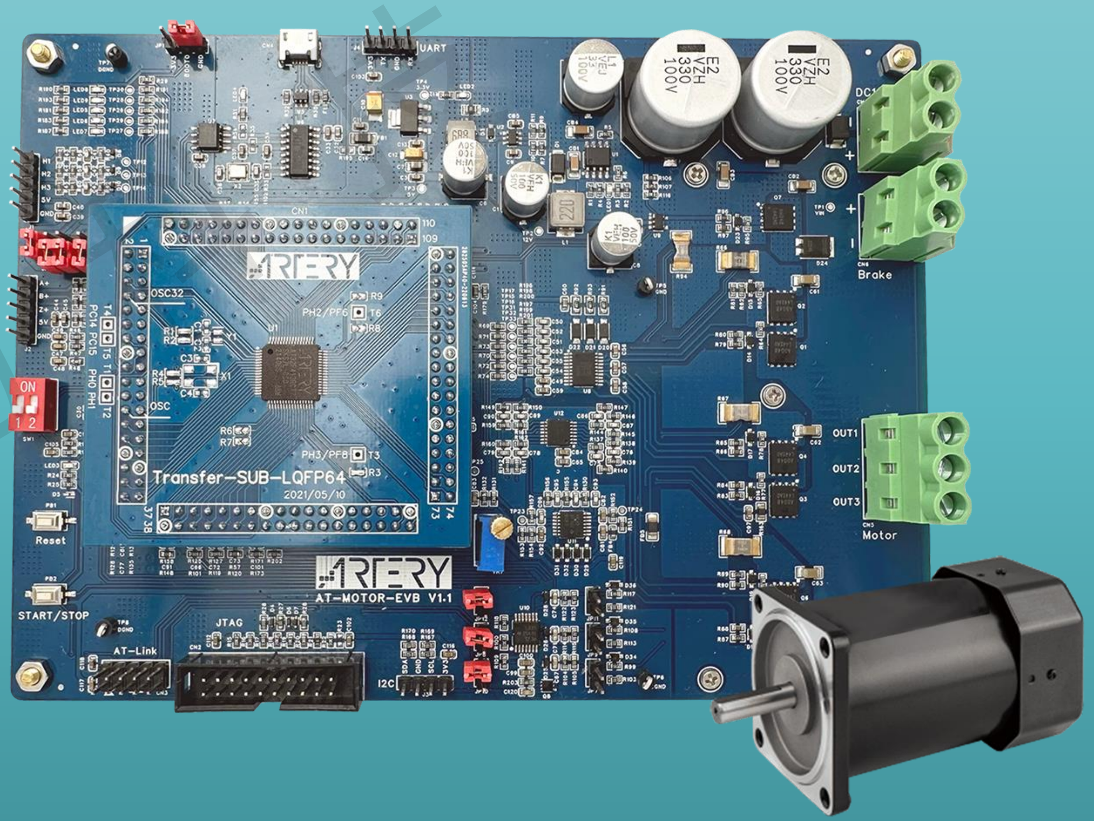
电机库下载



简介	中文文档	英文(EN)文档	支持型号	版本	日期
AT32 BLDC Hall 快速入门指南 描述了如何快速使用AT-MOTOR-EVB电机开发板搭配AT32 BLDC电机库调适不同型号带霍尔传感器的电机运转。	AN0213	AN0213	AT32F413 AT32F421	V2.1.0	2024/05/17
AT32 BLDC Sensorless 快速入门指南 描述了如何快速使用AT-MOTOR-EVB电机开发板搭配AT32 BLDC电机库调适不同型号无传感器的电机运转。	AN0214	AN0214	AT32F421 AT32F413	V2.1.0	2024/05/17
AT32 PMSM FOC Incremental Encoder 快速入门指南 描述了如何快速使用AT-MOTOR-EVB电机开发板搭配AT32 PMSM FOC电机库调适不同型号带光电增量编码器的电机运转	AN0217	/	AT32F403A AT32F413 AT32F415 AT32F421 AT32F423	V2.1.0	2024/05/24
AT32 PMSM FOC Hall 快速入门指南 描述了如何快速使用AT-MOTOR-EVB 电机开发板搭配AT32 PMSM FOC电机库调试不同型号带霍尔传感器的电机运转。	AN0218	/	AT32F413 AT32F415 AT32F403A AT32F421 AT32F423	V2.1.0	2024/05/24
AT32 PMSM FOC Hall(E-Bike-Scooter) 快速入门指南 描述了如何快速使用AT-MOTOR-EVB 电机开发板搭配AT32 PMSM FOC 电机库调试不同型号带霍尔传感器的电机运转并使用在电动自行车、电瓶车、电动滑板车等应用。	AN0219	/	AT32F403A AT32F413 AT32F415 AT32F421 AT32F423	V2.1.0	2024/05/24
AT32 PMSM FOC Sensorless 快速入门指南 描述了如何快速使用AT-MOTOR-EVB 电机开发板搭配AT32 PMSM FOC 电机库调试不同型号无传感器的电机运转。	AN0220	/	AT32F403A AT32F413 AT32F415 AT32F421 AT32F423	V2.1.0	2024/05/24

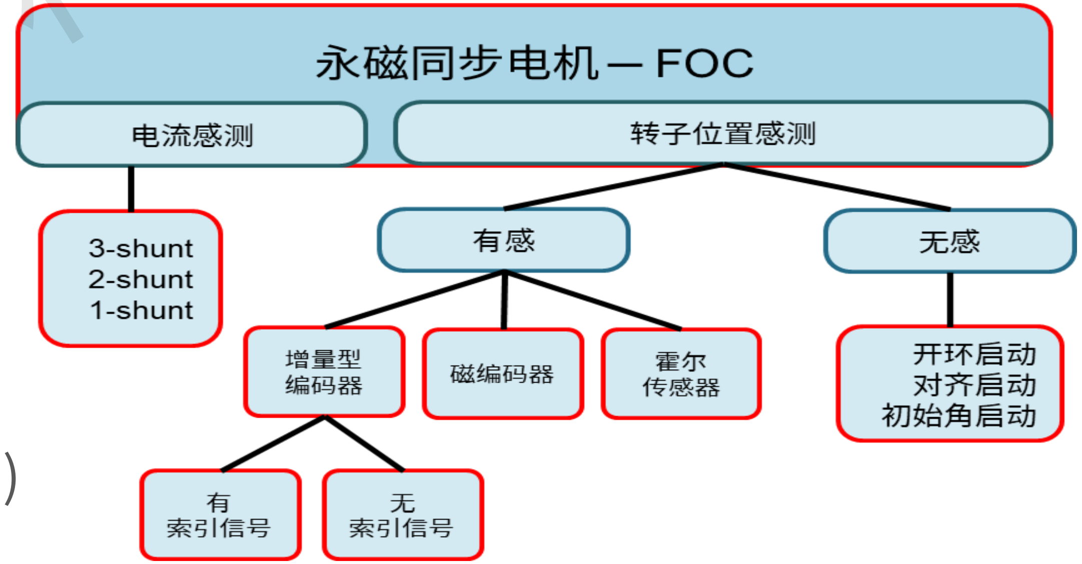
网址：https://www.arterytek.com/cn/support/motor_control.jsp?index=0

AT32 电机库
霍尔传感器FOC
快速上手
操作讲解



PMSM 霍尔传感器FOC电机控制函式库支持

- 支援FOC控制
- 提供AT32F421、AT32F413、AT32F415 、AT32F407 、AT32F403A 、AT32F423 、AT32F435/437 、AT32L021应用范例
- 支持免费AT32IDE编译环境
- 只须设定mc_hwio_v2(v1).h 和 motor_control_drive_param.h头文件
- 宏定变量的说明可参阅 AT32 电机库使用指南(AN0064)
- FOC霍尔传感器专案的说明可参阅(AN0167)
- FOC霍尔传感器E-Bike专案的说明可参阅(AN0194)
- FOC霍尔传感器快速入门指南(AN0218)
- FOC霍尔传感器E-Bike快速入门指南(AN0219)



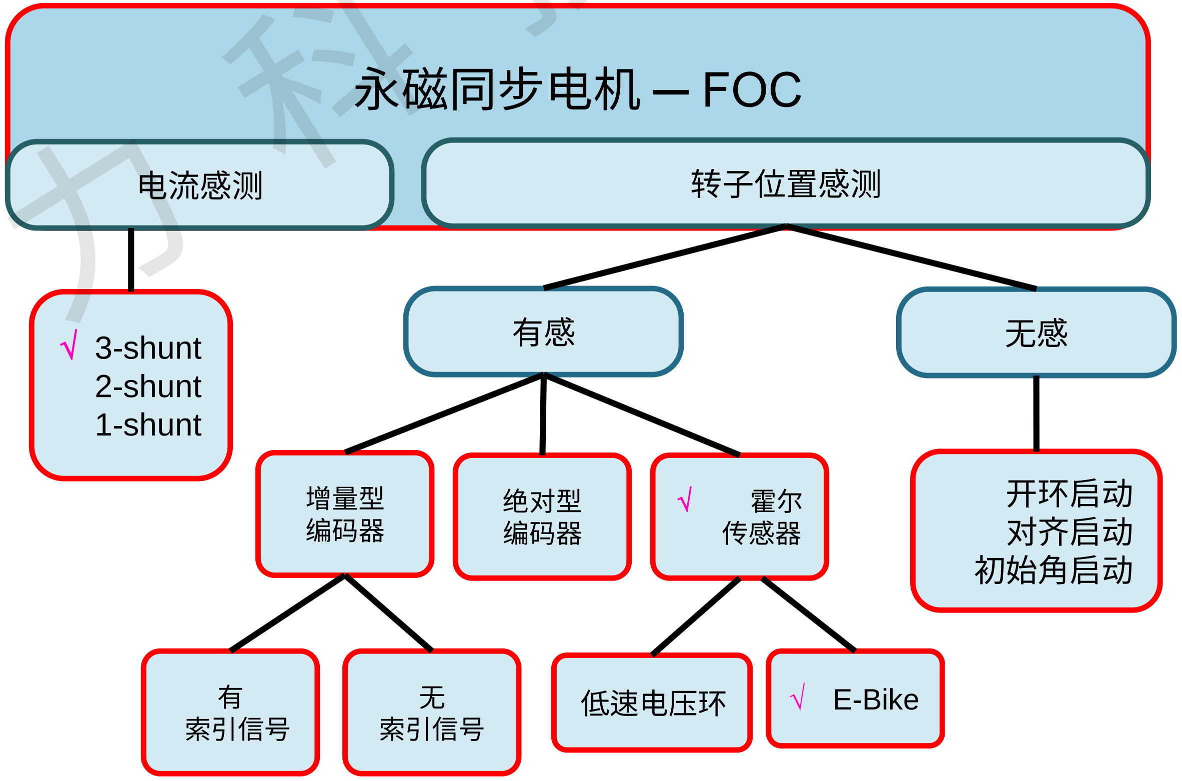
参数头文件 – 电机控制形式 (FOC)

motor_control_drive_param.h

控制型式

```
#define AT_MOTOR_EVB_V2
//#define AT_MOTOR_EVB_V1

/* gate driver low side inverting logi
#define GATE_DRIVER_LOW_SIDE_INVERT

#define FOC_CONTROL

#define THREE_SHUNT
//#define TWO_SHUNT
//#define ONE_SHUNT

#ifdef TWO_SHUNT
#define U_V_SHUNT
//#define V_W_SHUNT
//#define U_W_SHUNT
#endif

#define HALL_SENSORS

#ifdef HALL_SENSORS
#define LOW_SPEED_VOLT_CTRL          /* if
#endif

/* Application */
#define E_BIKE_SCOOTER

#ifdef E_BIKE_SCOOTER
#define DC_CURRENT_LIMIT          0
#define RDS_AUTO_CALIBRATION      0
#endif
```

- **选择宏定义宣告项目，完成电机控制形式设定**



参数头文件 – 电机参数

motor_control_drive_param.h

电机参数 (Motor-related parameter)

```
/********************************** Motor-related par
#define POLE_PAIRS                    (8/2)
#define RS_LL                         (1.89f)
#define LS_LL                         (0.002387f)
#define LD_LQ_RATIO                   (1.0f)
#define KE                            (0.003437f)
#define NOMINAL_CURRENT               (1.7f)

/*** Quadrature encoder ***/
#define ENCODER_PPR                   (1000)
#define ENC_IDX_COUNT                 (11)
#define ENC_STALL_TIME                (1000)


/* hall learn table */
#define HALL_LEARN_DIR                (0)
#define HALL_LEARN_0_STATE            (1)
#define HALL_LEARN_1_STATE            (5)
#define HALL_LEARN_2_STATE            (4)
#define HALL_LEARN_3_STATE            (6)
#define HALL_LEARN_4_STATE            (2)
#define HALL_LEARN_5_STATE            (3)
```

- POLE_PAIRS : 电机极对数
- RS_LL : 绕组线对线电阻值
- LS_LL : 绕组线对线电感值
- NOMINAL_CURRENT : 额定电流
- HALL_LEARN_DIR : 霍尔自学习转向
- HALL_LEARN_X_STATE : 霍尔状态对应的矢量

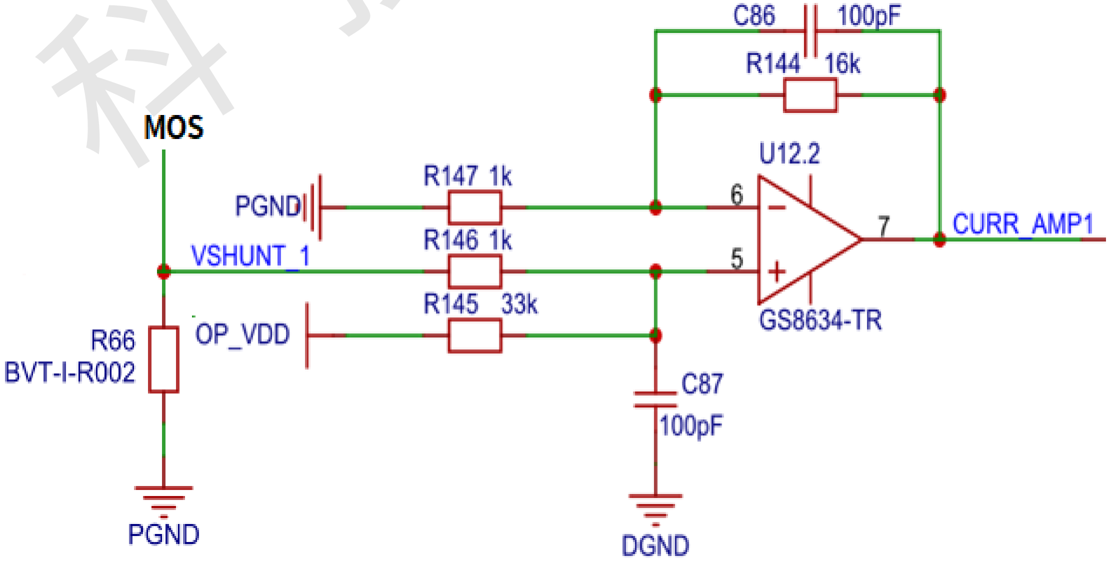
参数头文件 – 驱动器相关参数

motor_control_drive_param.h

驱动器参数 (Drive-related parameter)

```
#define VDC_RATED                        (24.0f)
#define BAT_LOW_VOLT                     (12.0f)           /*!< minimum allowable ba
#define V_SENSE_GAIN                     (10/(3.9f+180+10)) // 0.05157
#define ADC_REFERENCE_VOLT               (3.3f)
#define ADC_DIGITAL_SCALE_12BITS         (4095.0f)
/* Clock */
#if defined AT32F413xx
#define SYSTEM_CORE_CLOCK                (200000000)
#elif defined AT32F421xx
#define SYSTEM_CORE_CLOCK                (120000000)
#endif

#define TMR_CLK                          (SYSTEM_CORE_CLOCK)      //system_core_clock
#define DEADTIME_CLK_SFT_BITS            ((tmr_clock_division_type) 2)
#define DEADTIME_NS                      ((uint16_t)300)    /* in nsec; check the MOS
#define MIN_INTERVAL_TIME                (2500)     /* ns; pwm shift time for one shunt
#define ADC_TRIG_DELAY_TIME              (0.0)      /* us; adc trigger delay time for ga
/* Current */
#define MAX_CURRENT                      (5.0f)
#define MIN_CURRENT                      (-MAX_CURRENT)
#define DC_MAX_CURRENT                   (10.0f)
#define CURRENT_SPAN_SHIFT               ((uint8_t) 1)
#ifdef ONE_SHUNT
#define R_SHUNT                          (0.005f)
#define OP_GAIN                          (39.0f/(1+39)*(1+9.1/1))
#define CURR_OFFSET_VOLT                 (ADC_REFERENCE_VOLT*(1.0f/(1+39)*(1+9.1f/1)))
#else
#define R_SHUNT                          (0.002f)
#define OP_GAIN                          (33.0f/(1+33)*(1+16/1))
#define CURR_OFFSET_VOLT                 (ADC_REFERENCE_VOLT*(1.0f/(1+33)*(1+16/1)))
#endif
```

- 根据驱动器电路，设定相关电气参数



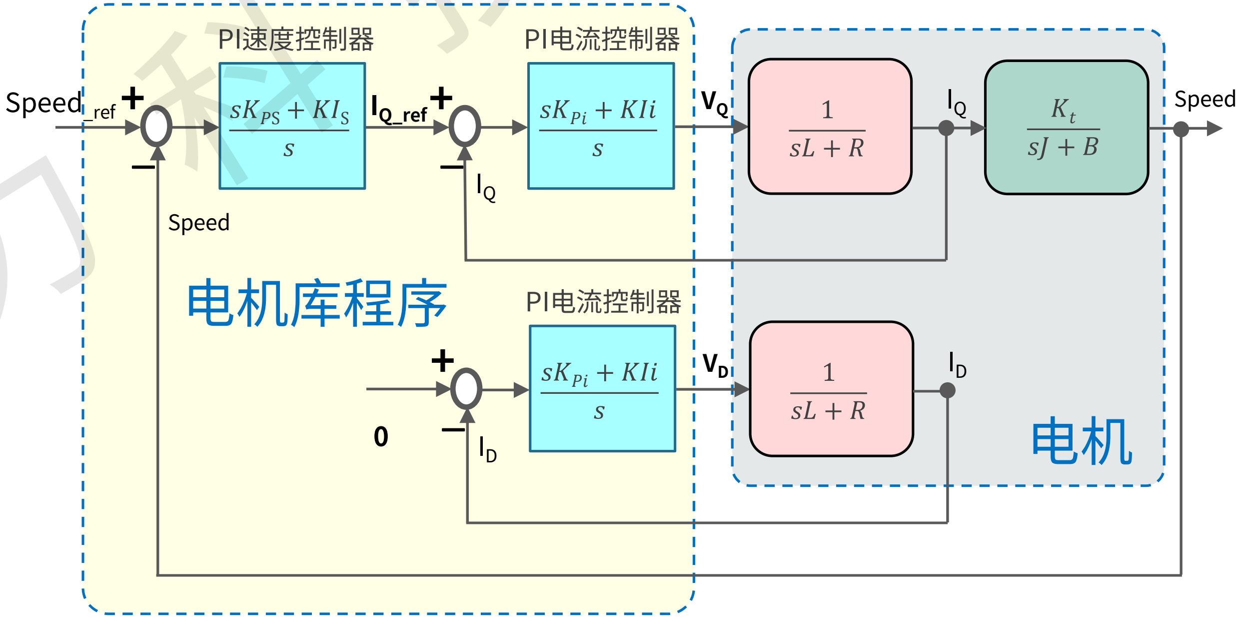
3-shunt/
2-shunt参数设定

参数头文件 – 控制环相关参数

motor_control_drive_param.h

控制参数 (Control-related parameter)

```
/*********************************** Control-related parameter ***********
#define PWM_FREQ                        (16000)     /* Hz */
#define MOTOR_CONTROL_MODE              (OPEN_LOOP_CTRL)
#define CTRL_SOURCE                     (CTRL_SOURCE_SOFTWARE)
#define UI_UART_BAUDRATE                (1500000UL)

/* current tuning parameter */
#define TUNE_TARGET_CURRENT             (1.0f)
#define TUNE_CURRENT_TOTAL_PERIOD       (100)
#define TUNE_CURRENT_STEP_PERIOD        (2)

/* SPEED */
#define SPEED_LOOP_FREQ                 (1000)  /* Hz */
#define MIN_SPEED_RPM                   (10)
#define MAX_SPEED_RPM                   (6000)
#define STABLE_SPEED_RPM                (50)
#define SLICK_SPEED_RPM                 (50)
#define MIN_POSCTL_SPD                  (100)
#define MIN_CONTROL_SPEED               (200)
#define ACC_SPD_SLOPE                   (15)     /* (rpm/ms) */
#define DEC_SPD_SLOPE                   (15)

/* POSITION */
#define POSITION_LOOP_FREQ              (200)      /* Hz */
#define MAX_POSITION_ANGLE              (360000)    /* Degree */
#define MIN_POSITION_ANGLE              (-MAX_POSITION_ANGLE)
#ifdef HALL_SENSORS
#define CMD_TO_VAL_GAP                  (20000)     /* hall eletric angle
#else
#define CMD_TO_VAL_GAP                  (1000)      /* counts (ex: encode
#endif
#define SMALL_POS_CMD_GAP               (100)
#define ROTOR_LOCK_ANGLE_GAP            (60.0f)

/* open loop control */
#define OLC_ANGLE_INC                   (0)
#define OLC_VOLT                        (0.0f)          /* (V) */

/* hall learn */
#define LEARN_OLC_VOLT                  (0.8f)        /* (V) */
#define LEARN_OLC_ANGLE_INC             (5)
#define LEARN_TIME                      (10000)       /* (ms) */
#define LEARN_ALIGN_TIME                (500)         /* (ms) */

/* Id/Iq pid parameter */
#define PID_ID_KP_DEFUALT               (7500)
#define PID_ID_KI_DEFUALT               (6000)
#define PID_ID_KP_GAIN_DIV              (512)
#define PID_ID_KP_GAIN_DIV_LOG          (LOG2(PID_ID_KP_GAIN_DIV))
#define PID_ID_KI_GAIN_DIV              (8192)
#define PID_ID_KI_GAIN_DIV_LOG          (LOG2(PID_ID_KI_GAIN_DIV))
```



ID_Kp = PID_ID_KP_DEFUALT / PID_ID_KP_DIV

ID_Ki = PID_ID_KI_DEFUALT / PID_ID_KI_DIV

外设定义头文件 – 外设参数定义

外设定义参数 (mc_hwio.h)

```c
/* adc reading pin definition */
#define ADC_CONVERTER                 ADC1
#define ADC_CONVERTER_CRM_CLK         CRM_ADC1_PERIPH_CLOCK
#define ADC_CONVERTER_CRM_CLK_DIV     CRM_ADC_DIV_8
#define ADC_SHUNT_SAMP_READY_IRQ      ADC1_2_IRQHandler
#define ADC_SHUNT_SAMP_READY_IRQn     ADC1_2_IRQn
#define ADC_ORDINARY_CH_LEN           6

/* dma1 ch1 for adc ordinary conversion */
#define ADC_ORDINARY_DMA_CRM_CLK      CRM_DMA1_PERIPH_CLOCK
#define ADC_ORDINARY_DMA_CHANNEL      DMA1_CHANNEL1
#define ADC_ORDINARY_DMA              DMA1
#define ADC_ORDINARY_DMA_FLEX         DMA_FLEXIBLE_ADC1
#define ADC_ORDINARY_DMA_FLEX_CH      FLEX_CHANNEL1

#define CURR_PHASE_A_ADC_CH           ADC_CHANNEL_0
#define CURR_PHASE_A_ADC_GPIO_CRM_CLK CRM_GPIOA_PERIPH_CLOCK
#define CURR_PHASE_A_ADC_PORT         GPIOA
#define CURR_PHASE_A_ADC_GPIO_PIN     GPIO_PINS_0

#define CURR_PHASE_B_ADC_CH           ADC_CHANNEL_1
#define CURR_PHASE_B_ADC_GPIO_CRM_CLK CRM_GPIOA_PERIPH_CLOCK
#define CURR_PHASE_B_ADC_PORT         GPIOA
#define CURR_PHASE_B_ADC_GPIO_PIN     GPIO_PINS_1

#define CURR_PHASE_C_ADC_CH           ADC_CHANNEL_2
#define CURR_PHASE_C_ADC_GPIO_CRM_CLK CRM_GPIOA_PERIPH_CLOCK
#define CURR_PHASE_C_ADC_PORT         GPIOA
#define CURR_PHASE_C_ADC_GPIO_PIN     GPIO_PINS_2
```

控制器MCU pin map

Peripheral function	AT32F421C8T6		AT32F413RCT7	
VBAT	1	VBAT	1	VBAT
STATUS_LED1	2	PC13	2	PC13
STATUS_LED2	3	PC14	3	PC14
STATUS_LED3	4	PC15	4	PC15
ENCODER_A+			18	TMR5_CH1(PF4)
ENCODER_B+			19	TMR5_CH2(PF5)
MODE_SW1	35	PF6	47	PF6
MODE_SW2	36	PF7	48	PF7
OSC_IN	5	OSC_IN	5	OSC_IN
OSC_OUT	6	OSC_OUT	6	OSC_OUT
RESET	7	NRST	7	NRST
SPEED_VR(F413)			8	ADC12_IN10(PC0)
			9	ADC12_IN11(PC1)
			10	ADC12_IN12(PC2)
			11	ADC12_IN13(PC3)
VSSA	8	VSSA	12	VSSA
VDDA	9	VDDA	13	VDDA
CURR_FDBK1	10	ADC_IN0(PA0)	14	ADC_IN0(PA0)
CURR_FDBK2	11	ADC_IN1(PA1)	15	ADC_IN1(PA1)
CURR_FDBK3	12	ADC_IN2(PA2)	16	ADC_IN2(PA2)
IBUS_FDBK	13	ADC_IN3(PA3)	17	ADC_IN3(PA3)
BEMF1_LF	14	ADC_IN4(PA4)	20	ADC_IN4(PA4)
BEMF2_LF	15	ADC_IN5(PA5)	21	ADC_IN5(PA5)
BEMF3_LF	16	ADC_IN6(PA6)	22	ADC_IN6(PA6)
VBUS	17	ADC_IN7(PA7)	23	ADC_IN7(PA7)
ENCODER_Z+			24	PC4
			25	ADC_IN15(PC5)

UI程序与电机控制调校

定义头档相关参数

I/O mapping

Board param.

Motor param.

Control param.

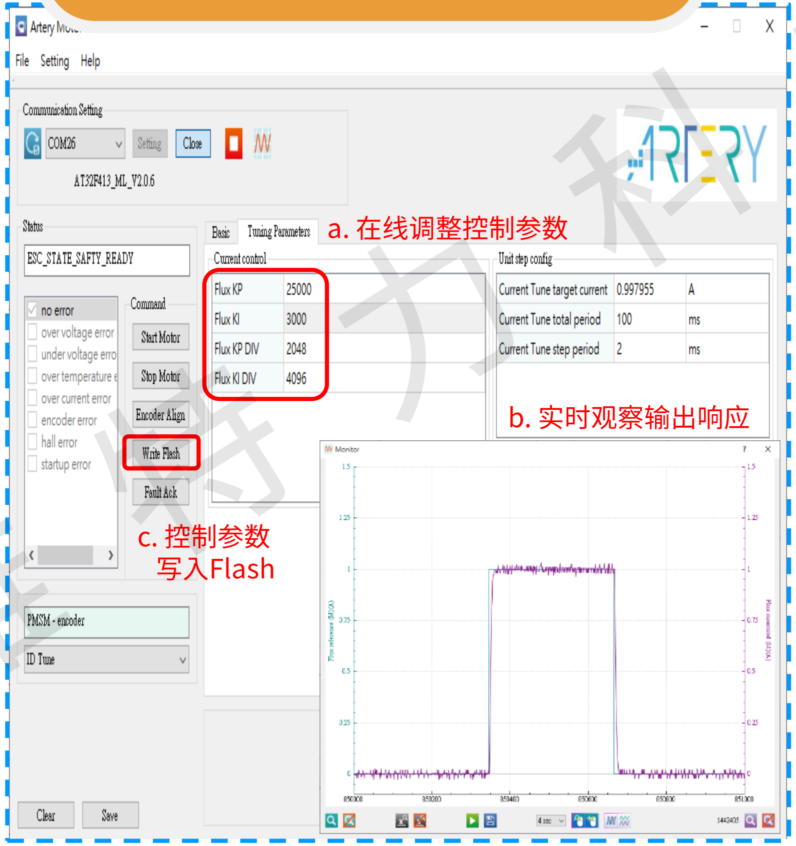
Drive mode

- FOC_CONTROL
- ONE_SHUNT
- SENSORLESS

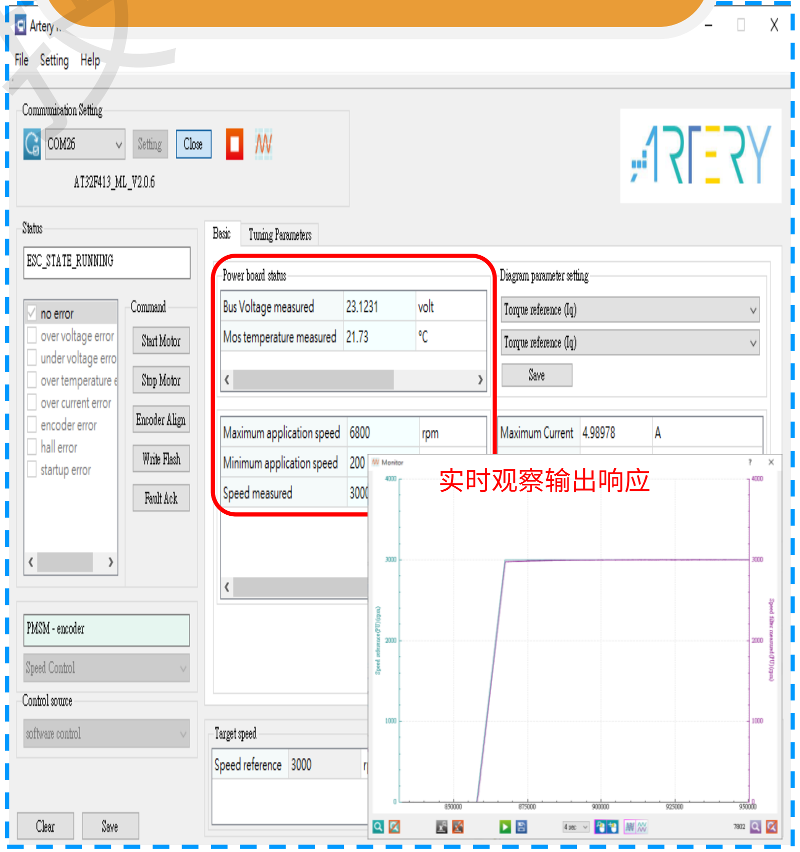
电机控制程序

Motor control project

UI程序调校PID控制参数

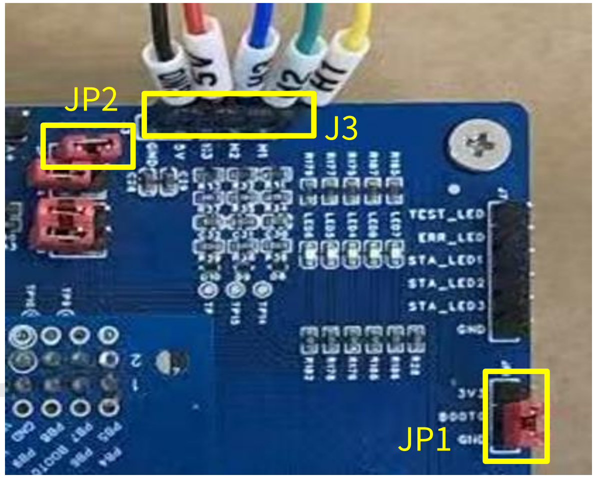


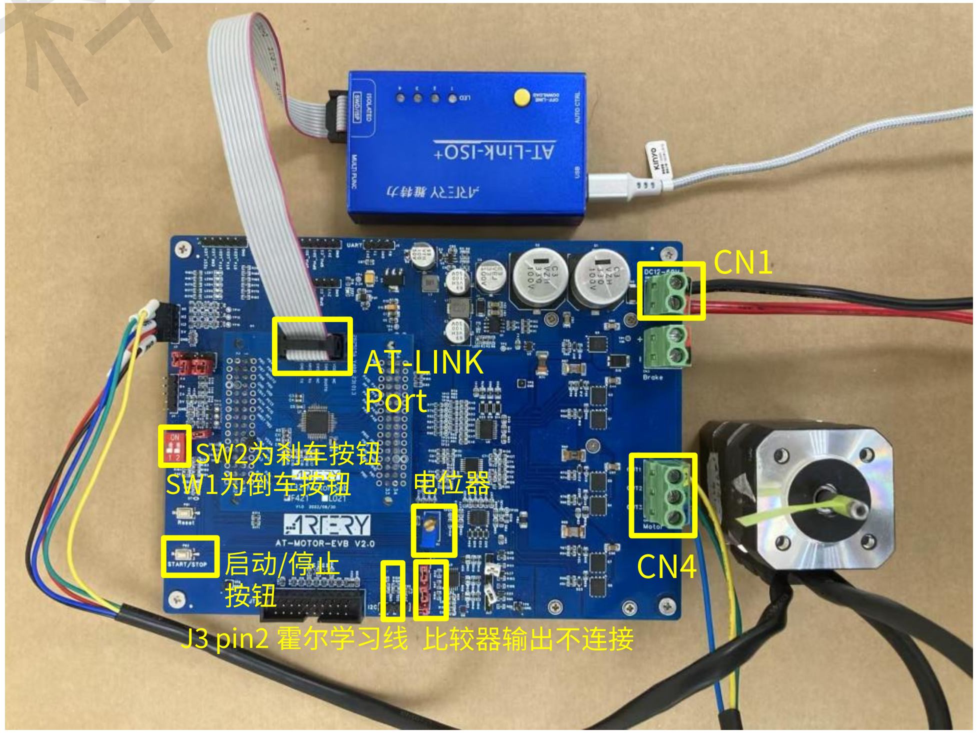
实时监看响应特性与数据



控制器接线

- 将霍尔线连接到J3 (依线标)
- 检查JP1 Boot0是否接地
- 检查JP2是否短接
- 将AT-LINK连接到小板
- 连接电机线到CN4 (依线色)
- 连接电源供应器电源接线到CN4 (依标示)
- 设定电源供应器电压24V / 3A
- 开启电源供应器输出
- SW1为倒车按钮
- SW2为刹车按钮
- J3 pin2 为霍尔学习输入



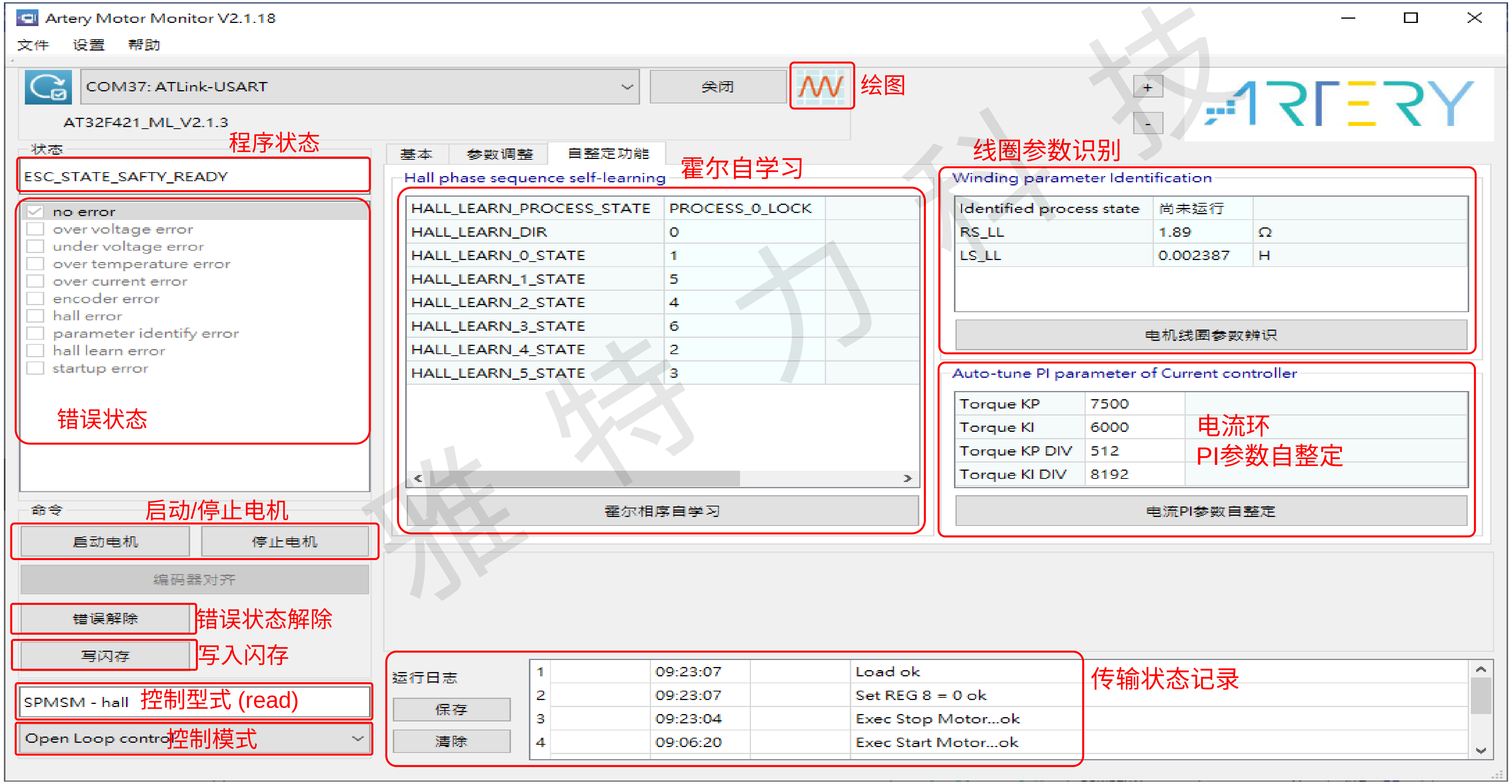


电机与开发板接线图

监控界面软件运行操作 –开环控制

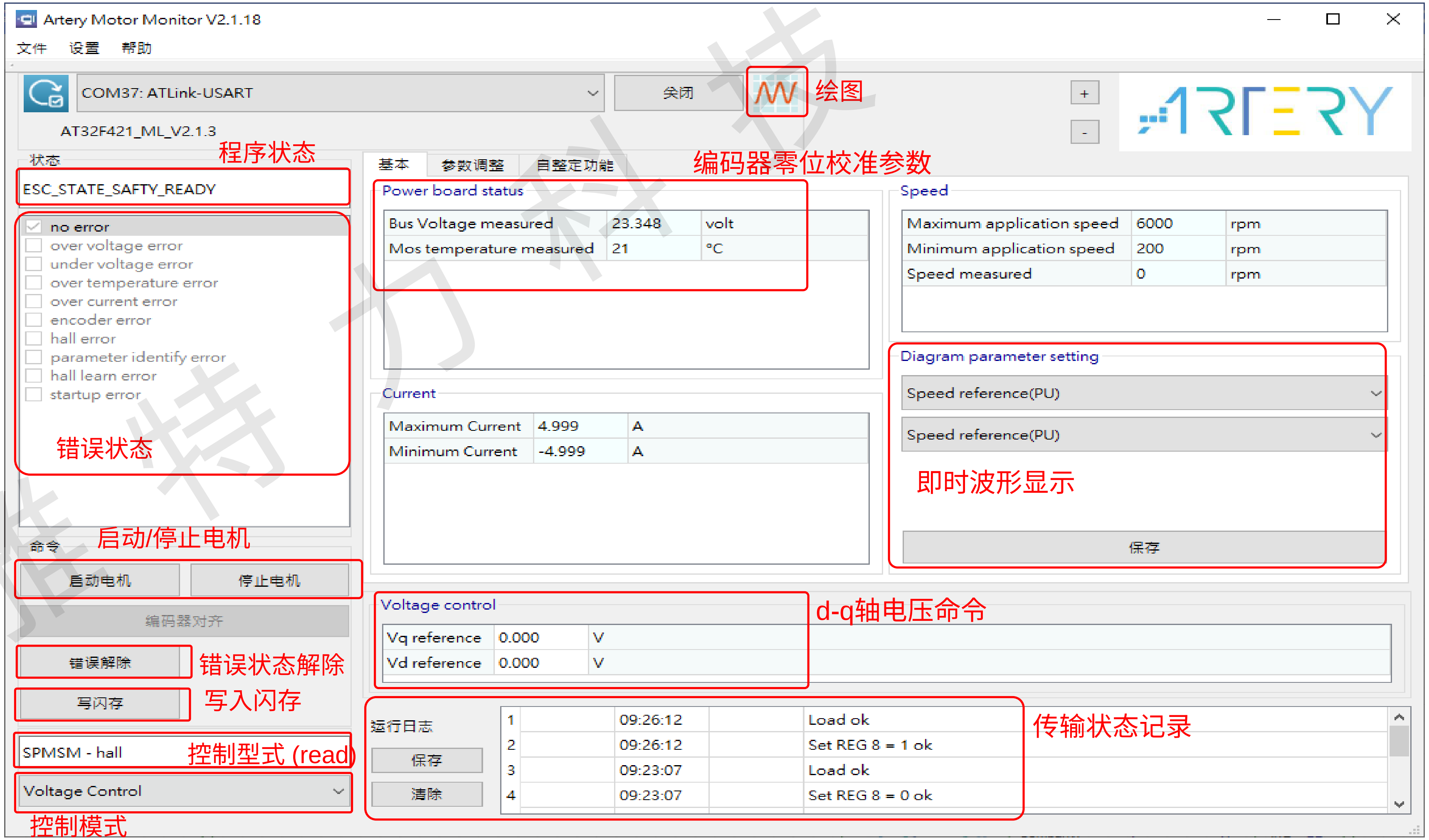
- 设定初始开环电压
- 设定初始开环角度递增值
- 增加一点开环电压
- 增加一点角度递增值
- 依次递增电压与角度递增值，逐步拉高转速

监控界面软件运行操作 –自整定功能

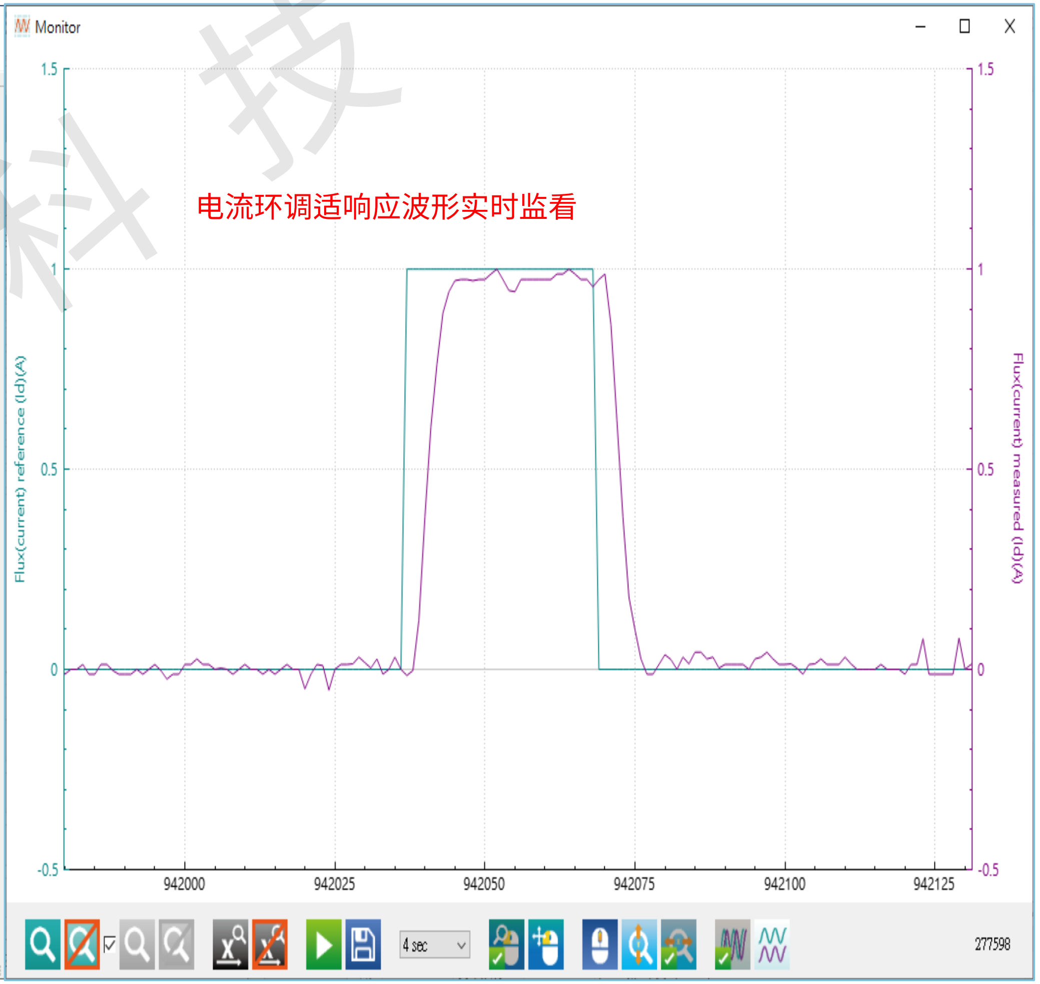


监控界面软件运行操作 – 电压控制

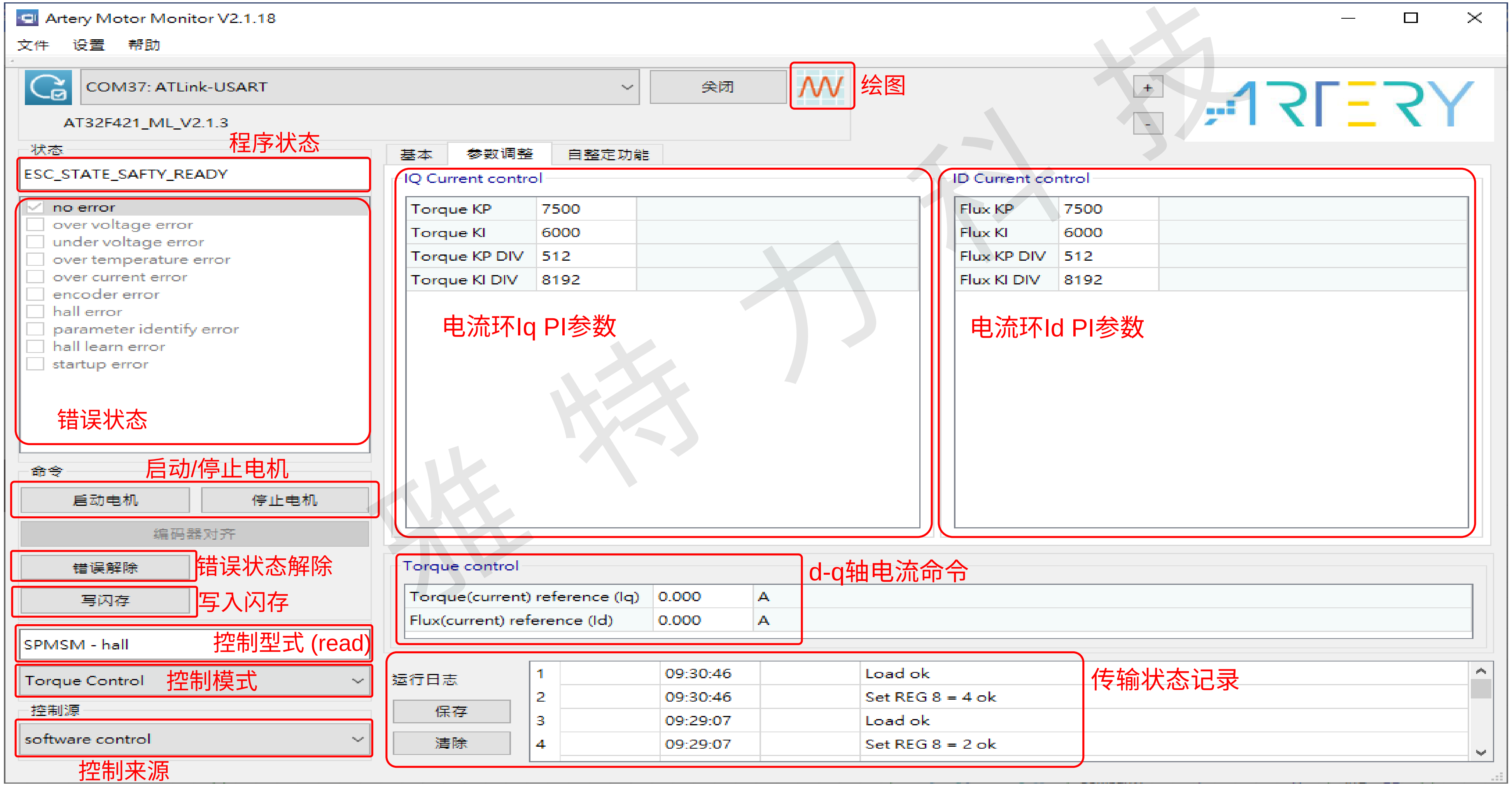
- 设定q轴电压Vq
- 启动电机即可运转
- 逐步增加Vq电压
- 可观测Id与Iq电流
- 适当调整d轴电压Vd使Id电流接近于0
- 在电流环未适当调整时，可在电压模式观察电流回授状态



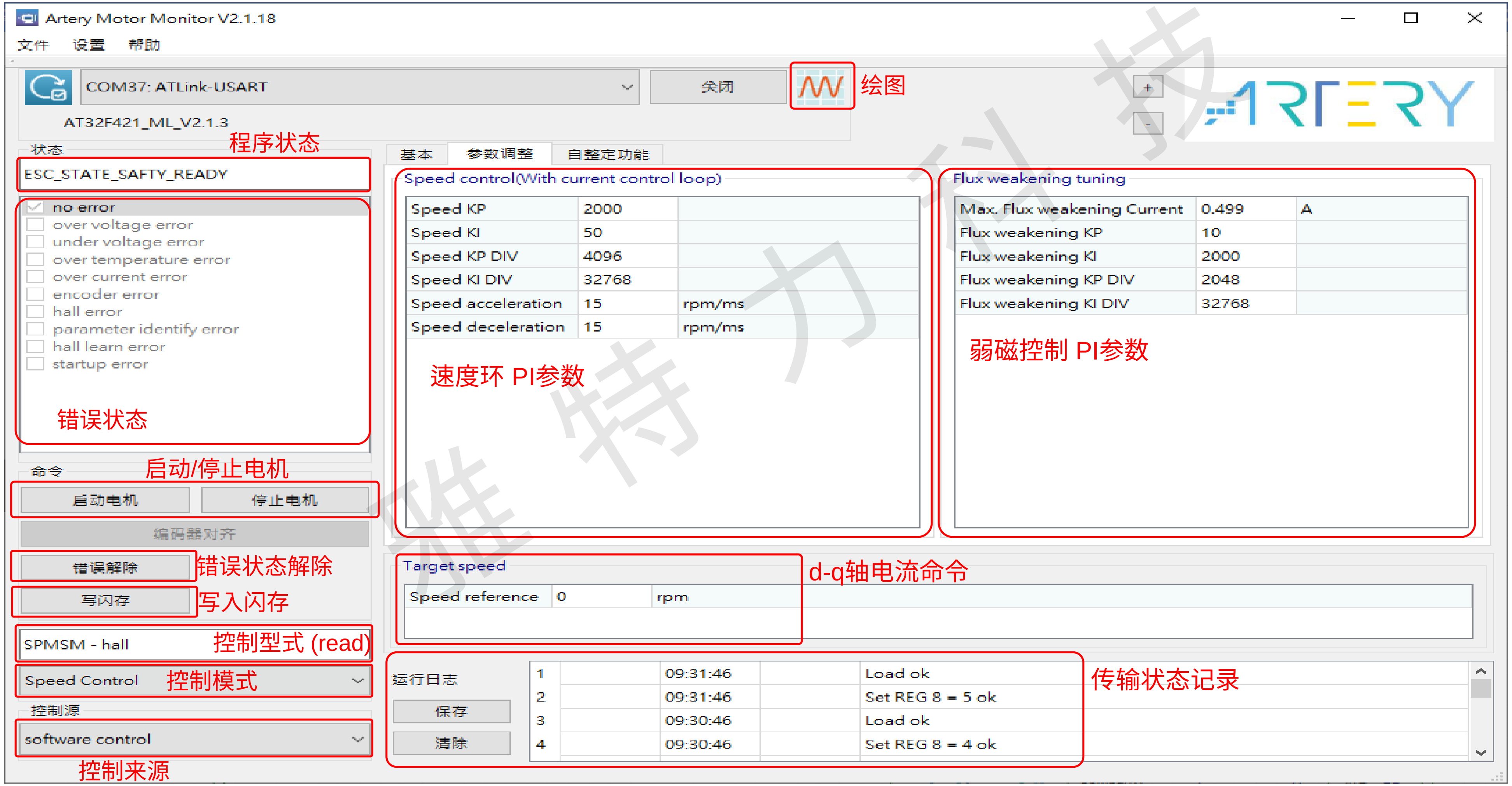
监控界面软件运行操作 – 电流环参数实时调整



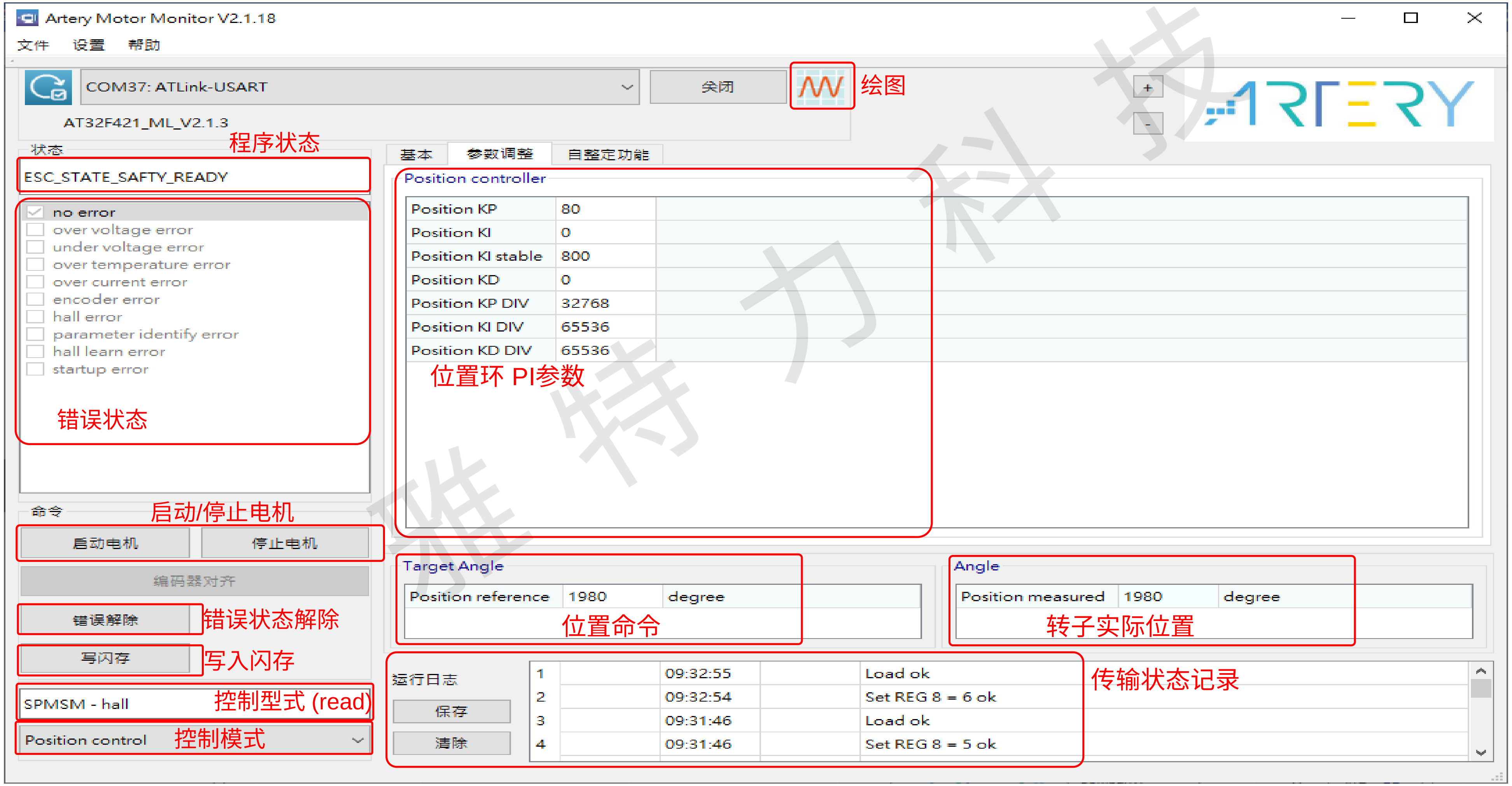
监控界面软件运行操作 – 转矩控制



监控界面软件运行操作 – 速度控制

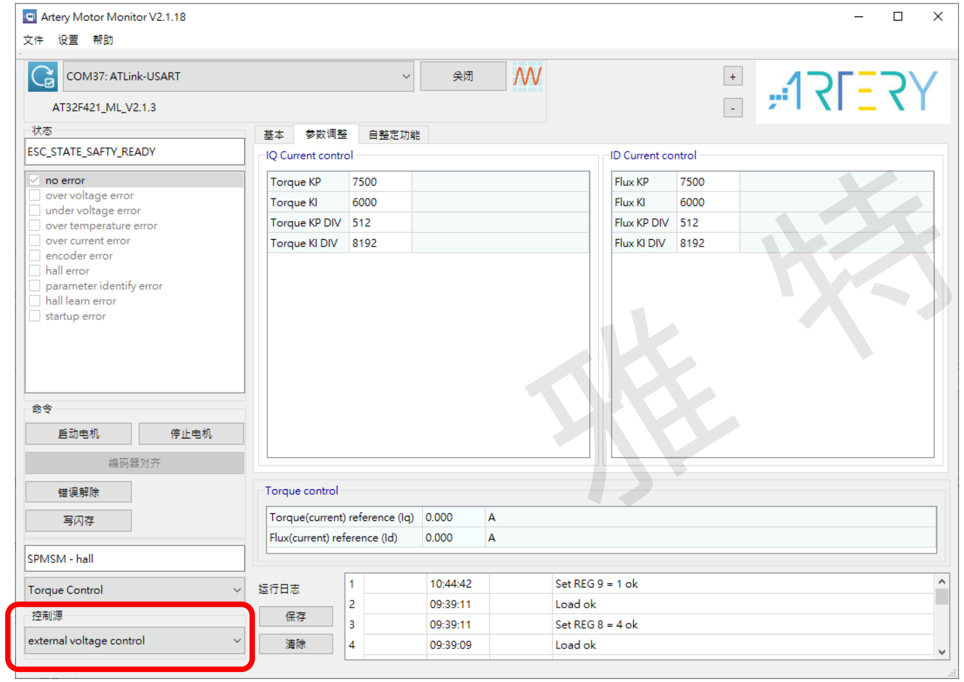


监控界面软件运行操作 – 定位控制

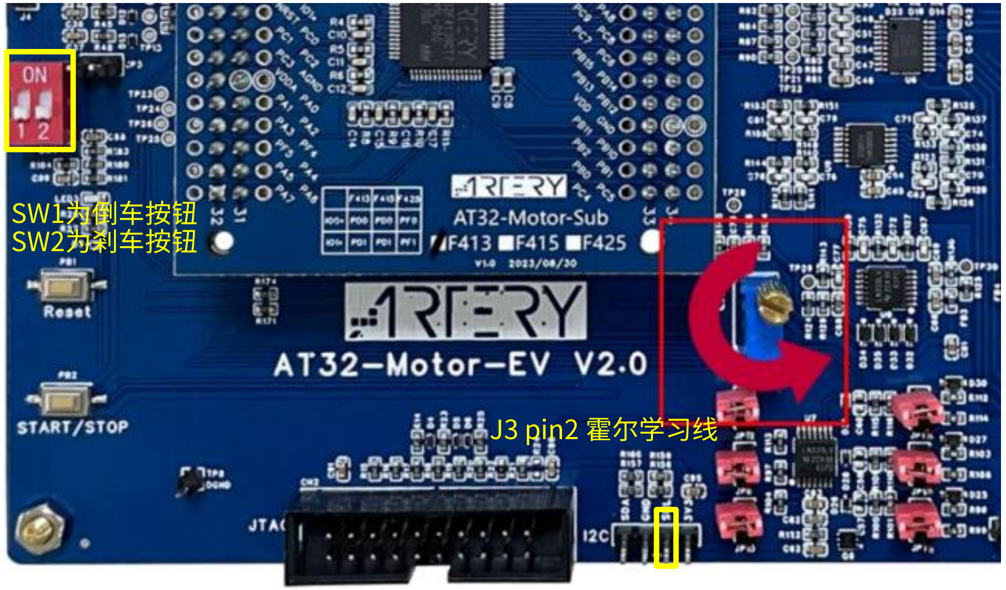


外部命令控制 (E-Bike)

- 或在motor_control_drive_param.h头文件中定义’CTRL_SOURCE’为’CTRL_SOURCE_EXTERNAL’
- 复归电位计后，开始启动 ；在上位机切换为外部控制源，若电位计不为零，会直接启动



外部控制源



E-Bike 控制参数

- 倒车控制参数

```
#define REVERSE_MAX_SPEED_RPM              (-500)      倒车速限
#define REVERSE_CURRENT                    (-0.3f)     倒车电流
```

- 刹车控制参数

```
#define BRAKING_CURRENT                    (-0.3f)     刹车回生电流
```

- 输入直流电流限制

```
#define DC_MAX_CURRENT                     (10.0f)
```

Q&A

提问时间

10分钟

联系我们

中文官网

English Website

微信公众号

Linkedin

- 销售咨询 sales_artery@arterytek.com
- 技术服务 support_artery@arterytek.com

ARTERY

Thank you!