

**ARFER Y 雅特力**

# AT32电机库示范讲解 - BLDC 无感控制

2.6 (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

2025.02.06 活动议程

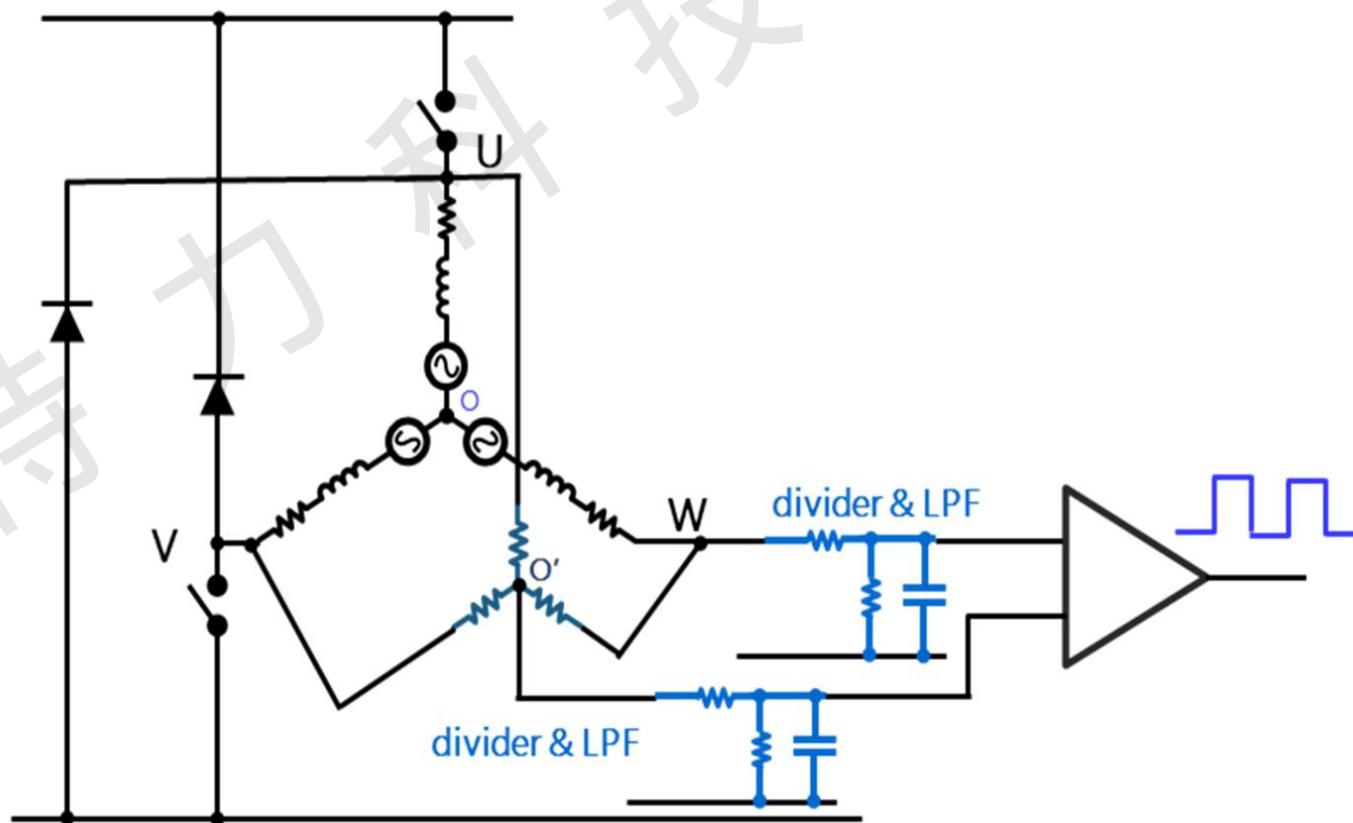
时间	主题	讲师
13:30 -14:30	BLDC无感控制原理与无感电机库架构解说	电机应用软件 资深工程师 林明赞 博士
中场休息		
14:40 -15:30	AT32 电机库BLDC无感控制快速上手操作解说	电机应用软件 资深工程师 林明赞 博士

# 直流无刷电机 无感控制原理

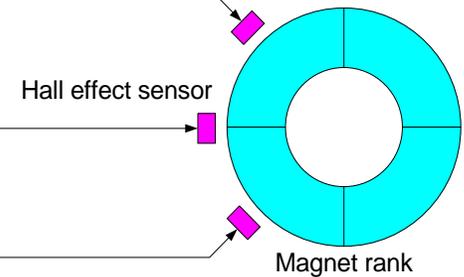
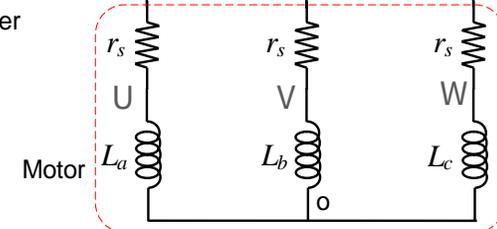
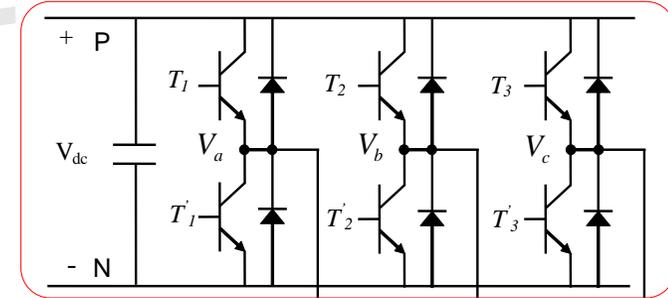
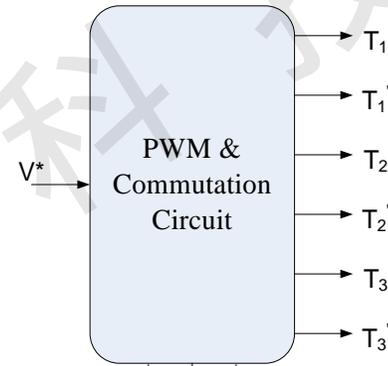
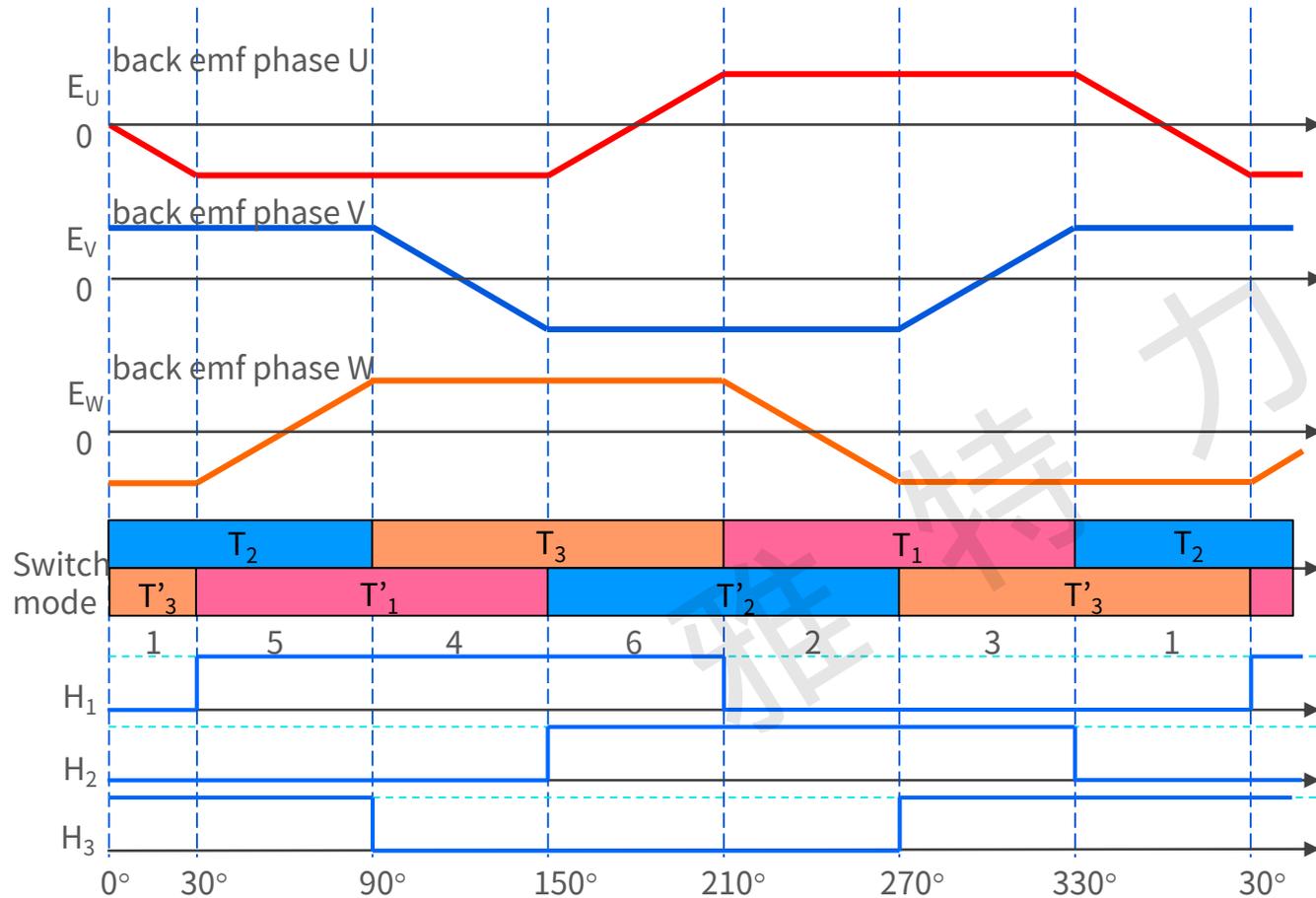


# 直流无刷电机无感控制

- 优点
  - 降低成本与简化接线
  - 提高可靠度
  - 适用于噪声干扰环境
- 缺点
  - 低速控制困难
  - 启动较困难
  - 换相误差较大

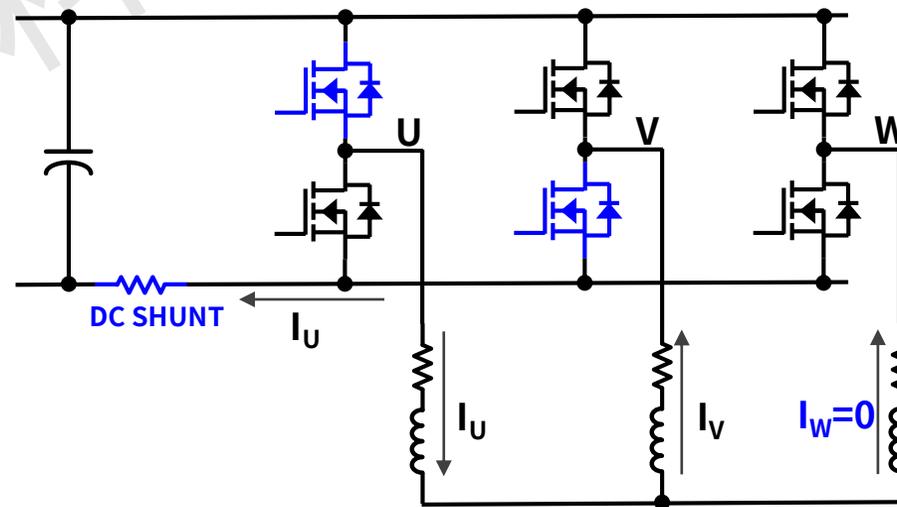


# 直流无刷电机120°导通换相 (6-Step)



# 直流无刷电机端电压

- 当开路相电流为零时可以量测到该相反电势



三相全桥驱动电路

# 直流无刷电机120°驱动电气方程式

- 无刷电机三相电气方程式

$$V_{DC} = R I_U + L \frac{dI_U}{dt} + e_U + V_o$$

$$0 = R I_V + L \frac{dI_V}{dt} + e_V + V_o$$

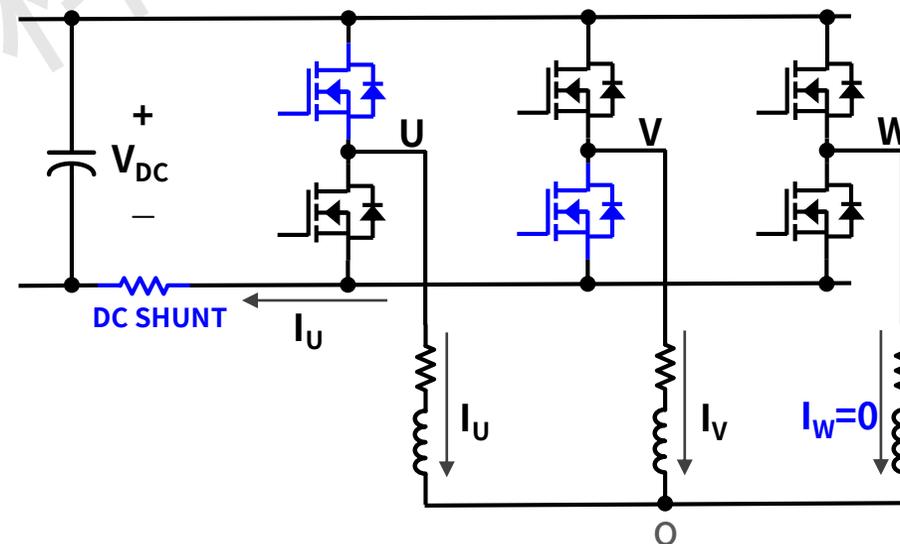
$$V_W = e_W + V_o$$

- 因  $I_U = -I_V$ ,

$$V_{DC} = e_U + e_V + 2V_o$$

$$V_o = (e_W + V_{DC})/2$$

$$V_W = e_W + V_o$$



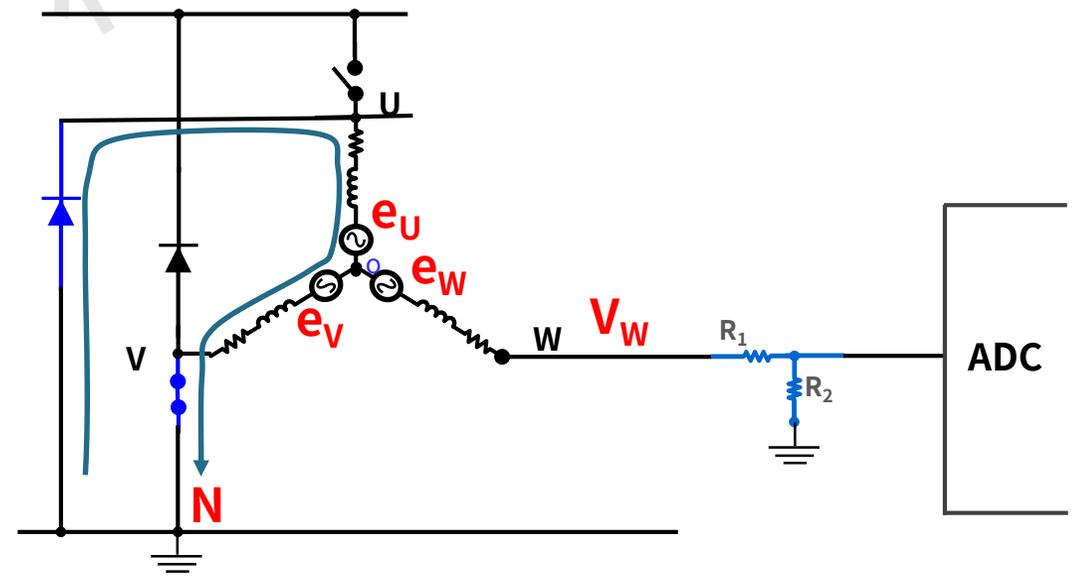
三相全桥驱动电路

# ADC取样：在PWM OFF时侦测反电势

- 在低速时PWM占空比较小，故可选择在PWM OFF时侦测反电势

PWM OFF 时的等效方程式

$$\begin{aligned}0 &= e_U + e_V + 2V_o \\ V_o &= -(e_U + e_V)/2 \\ V_o &= (e_W)/2 \\ V_W &= e_W + V_o = 1.5 e_W\end{aligned}$$



# ADC取样：在PWM ON时侦测反电势

- 在高速时PWM占空比较大，故可选择在PWM ON时侦测反电势

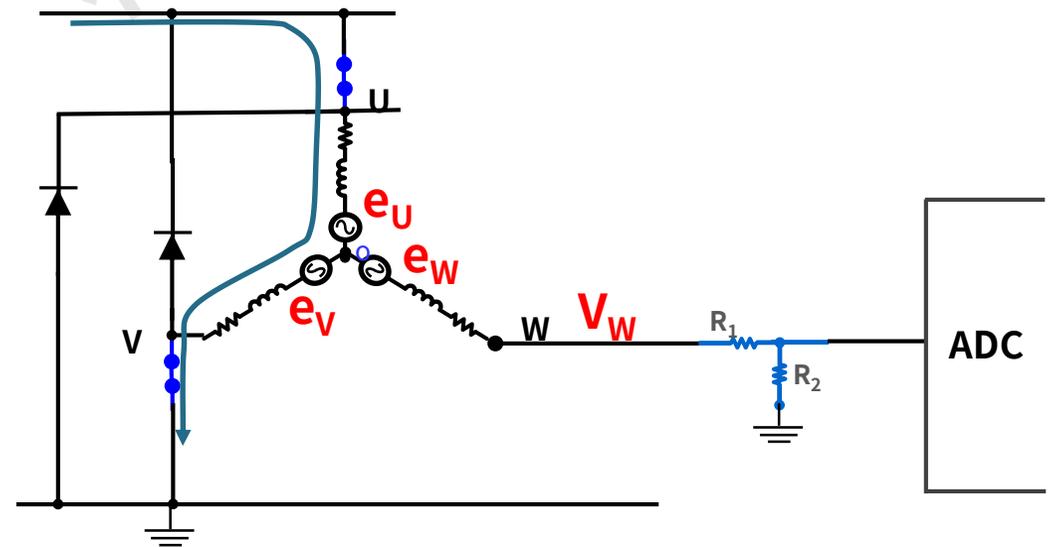
PWM ON 时的等效方程式

$$V_{DC} = e_U + e_V + 2V_o$$

$$V_o = -(e_U + e_V)/2 + V_{DC}/2$$

$$V_o = (e_W)/2 + V_{DC}/2$$

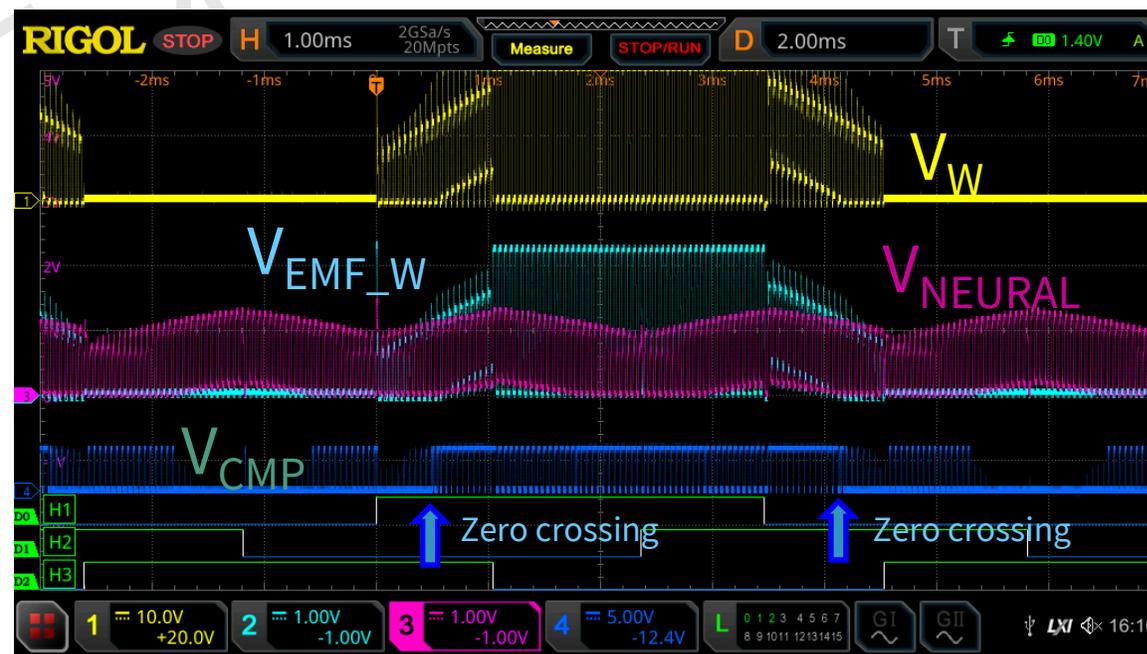
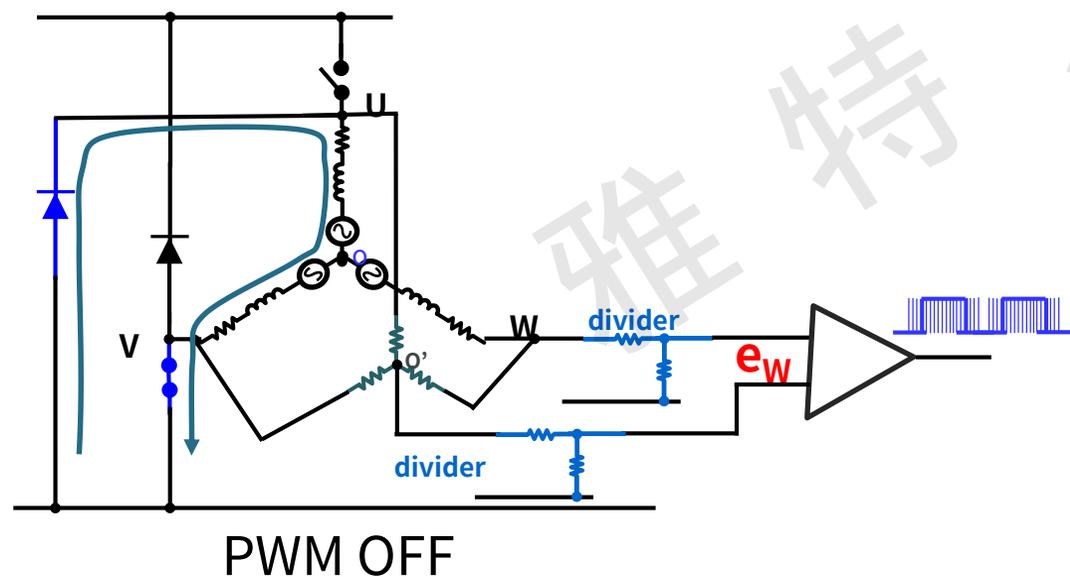
$$V_W = e_W + V_o = 1.5 e_W + V_{DC}/2$$





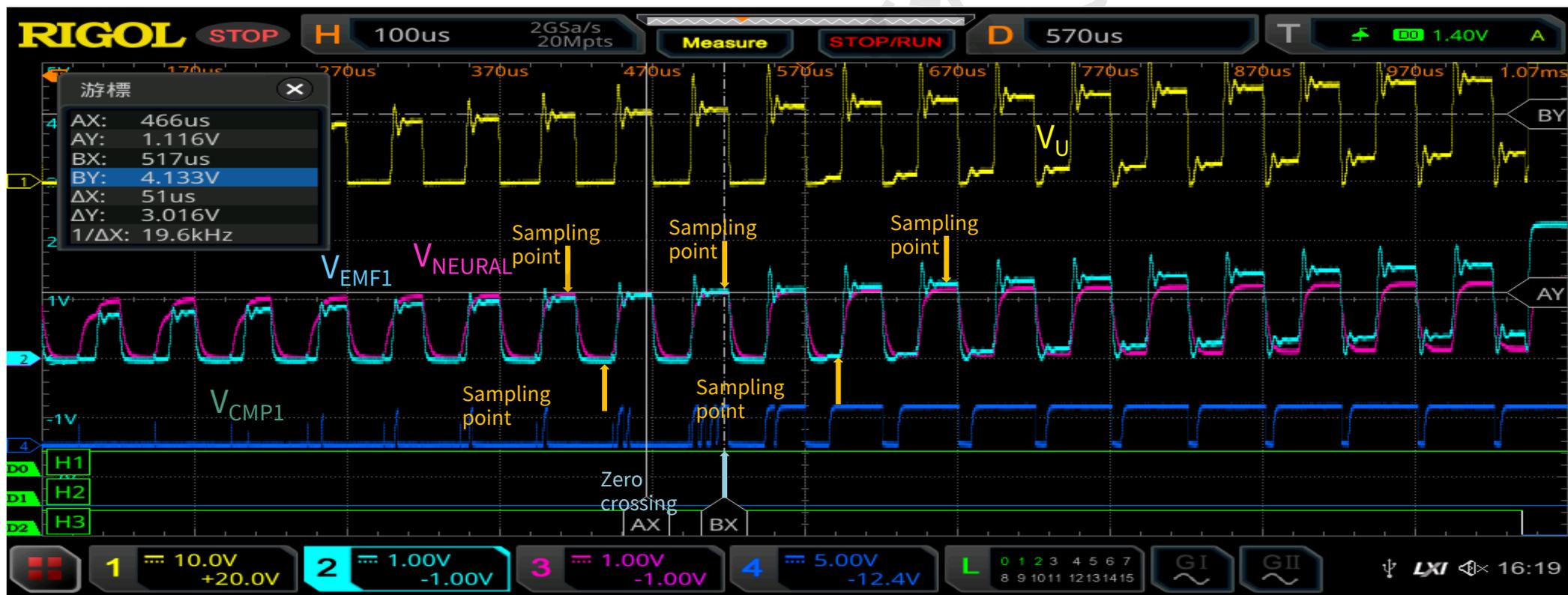
# 无电容滤波的比较器侦测方法

- 因电机线圈与感测电路阻抗差异，浮接相与虚拟中性点间的电压无法同相位
- 比较器输入信号经RC滤波后，输出可得方波换相讯号，但高转速时换相时信号会有相位延迟
- 此时比较器输出带有PWM切换噪声
- 须适当选择取样时机，避开切换噪声



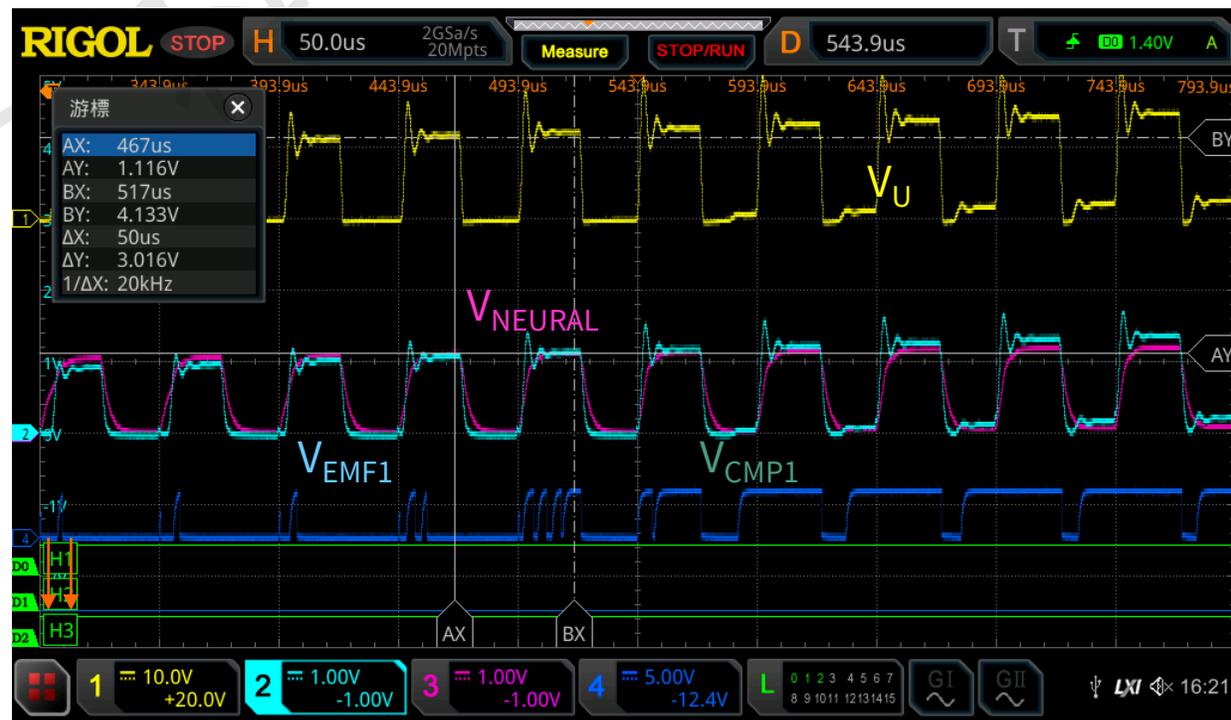
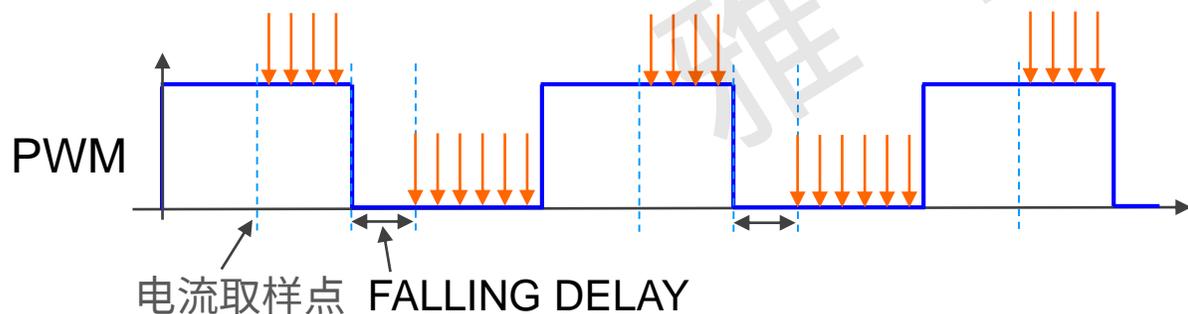
# 比较器侦测时机

- PWM 占空比较小时，令取样点接近于 PWM (PWM OFF)周期结束点
- PWM 占空比较大时，令取样点接近于 PWM ON结束点



# 连续读取比较器输出

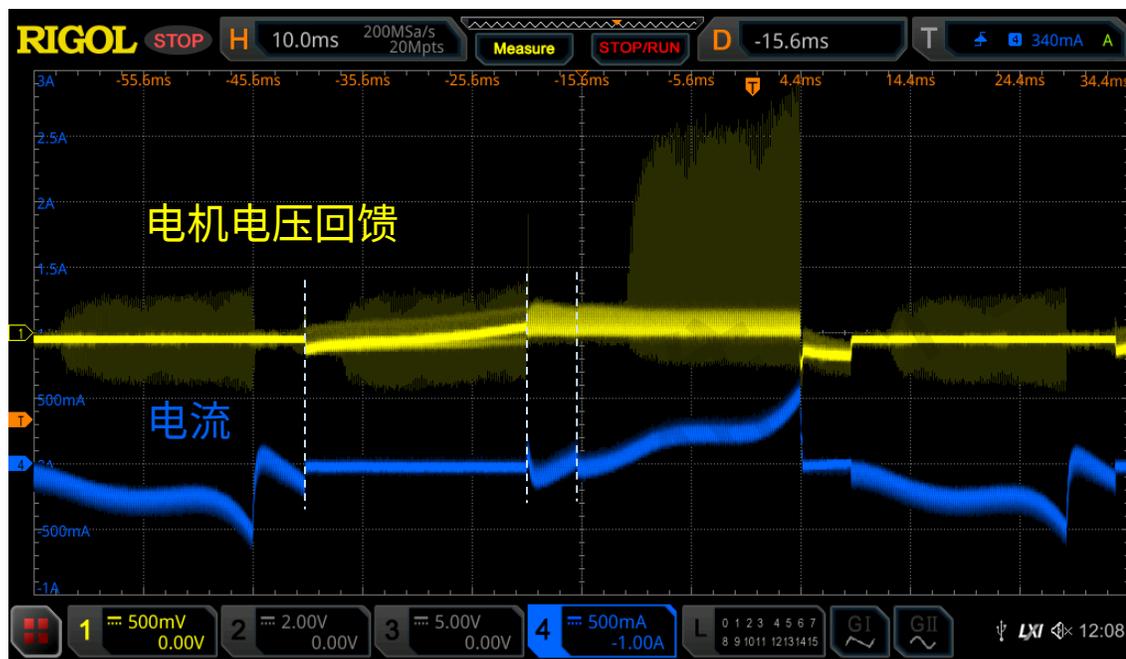
- 针对换相频率较高的高转速电机，可选择连续读取比较器输出
- 读取时机
  - PWM ON：电流取样后
  - PWM OFF：PWM OFF后经一段时间延迟
  - 连续读取间隔时间可设定





# 专利技术 – BLDC 无感ADC取样低速控制

- 传统感测电路因输出的反电势电压小，零交越点侦测不准确，换相不稳定
- 专利电路可提升零交越点电压位准且不衰减反电势信号，换相准确稳定



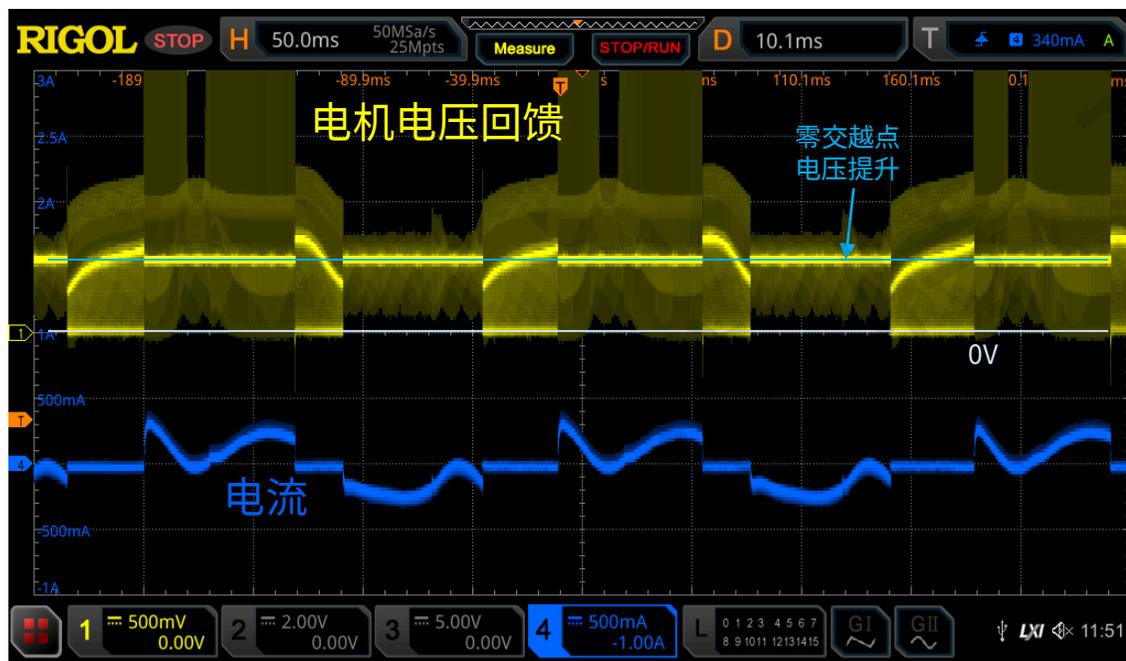
传统感测电路在 200 rpm 下无感控制响应



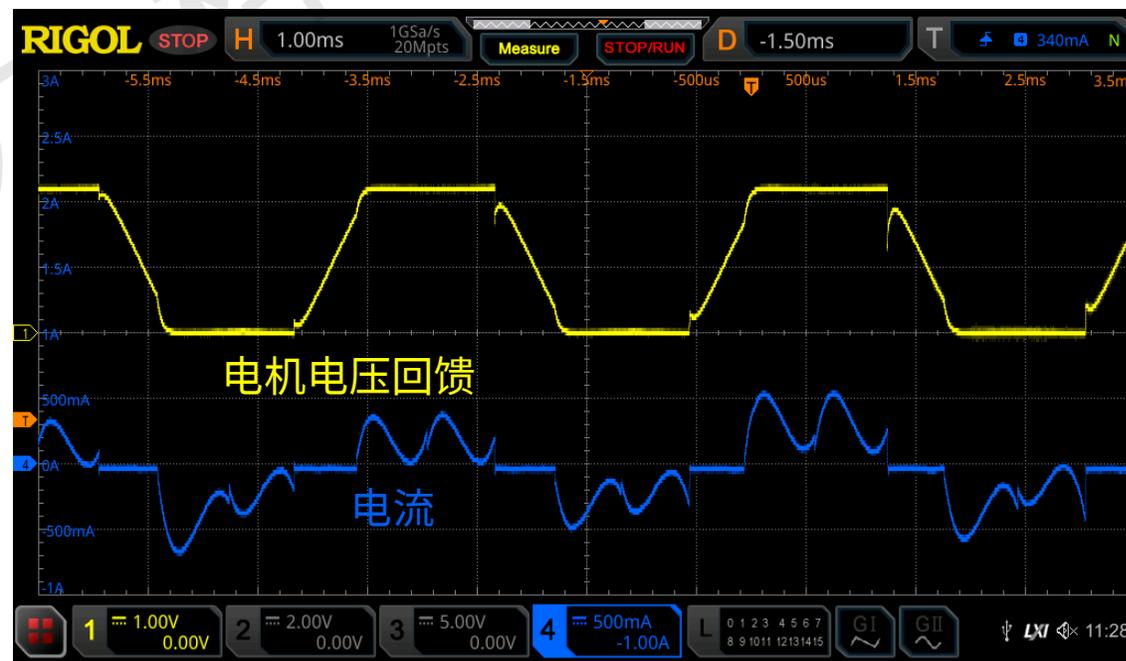
专利感测电路在 200 rpm 下无感控制响应

# 专利技术 – BLDC 无感ADC取样低速控制

- 专利电路可根据PWM占空比状态，决定分压电路连接VCC电压或者接地
- 可适用宽范围的高低速无刷电机转速控制



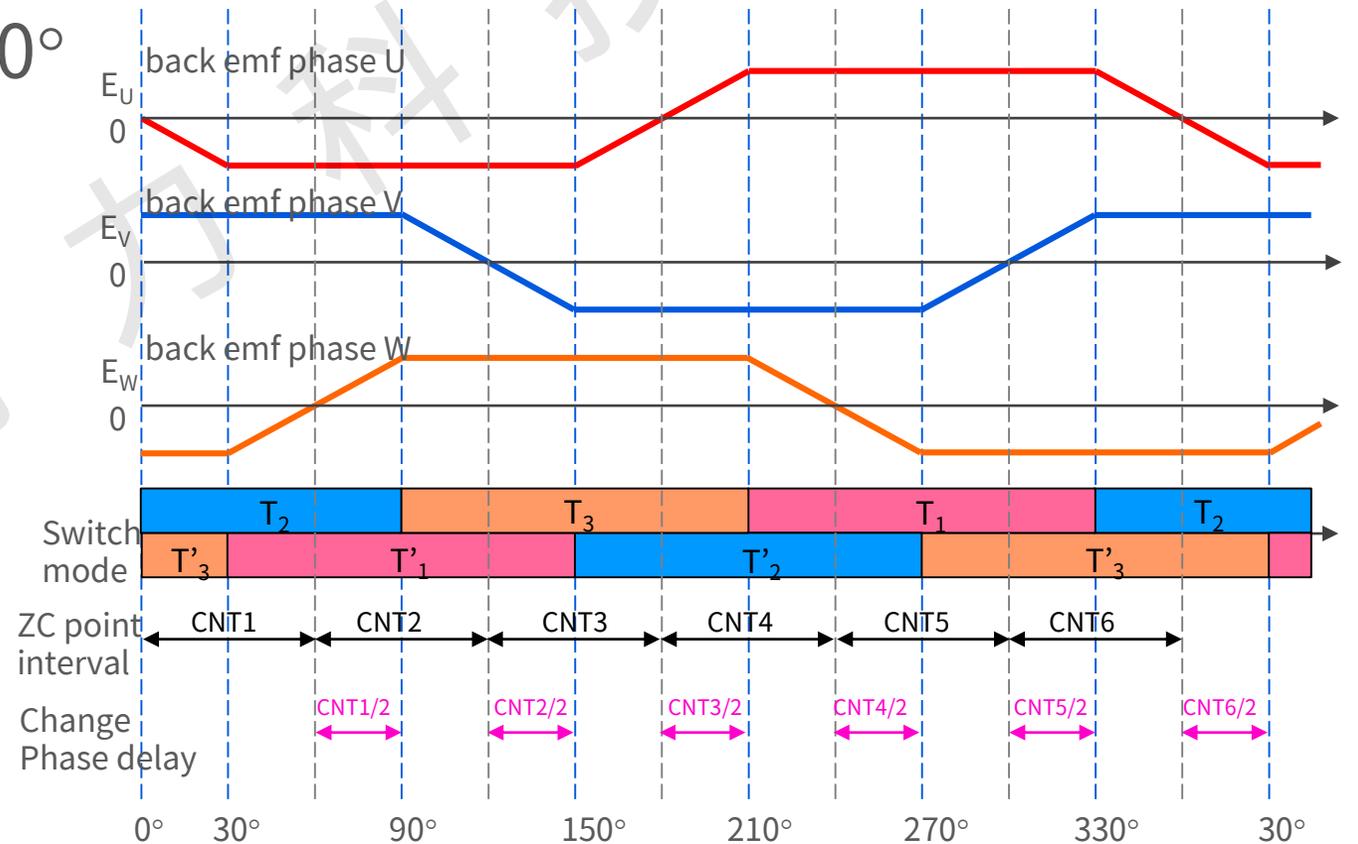
专利感测电路在 80 rpm 下无感控制响应



专利感测电路在 4,200 rpm 下无感控制响应

# 反电势零交越点后30°换相

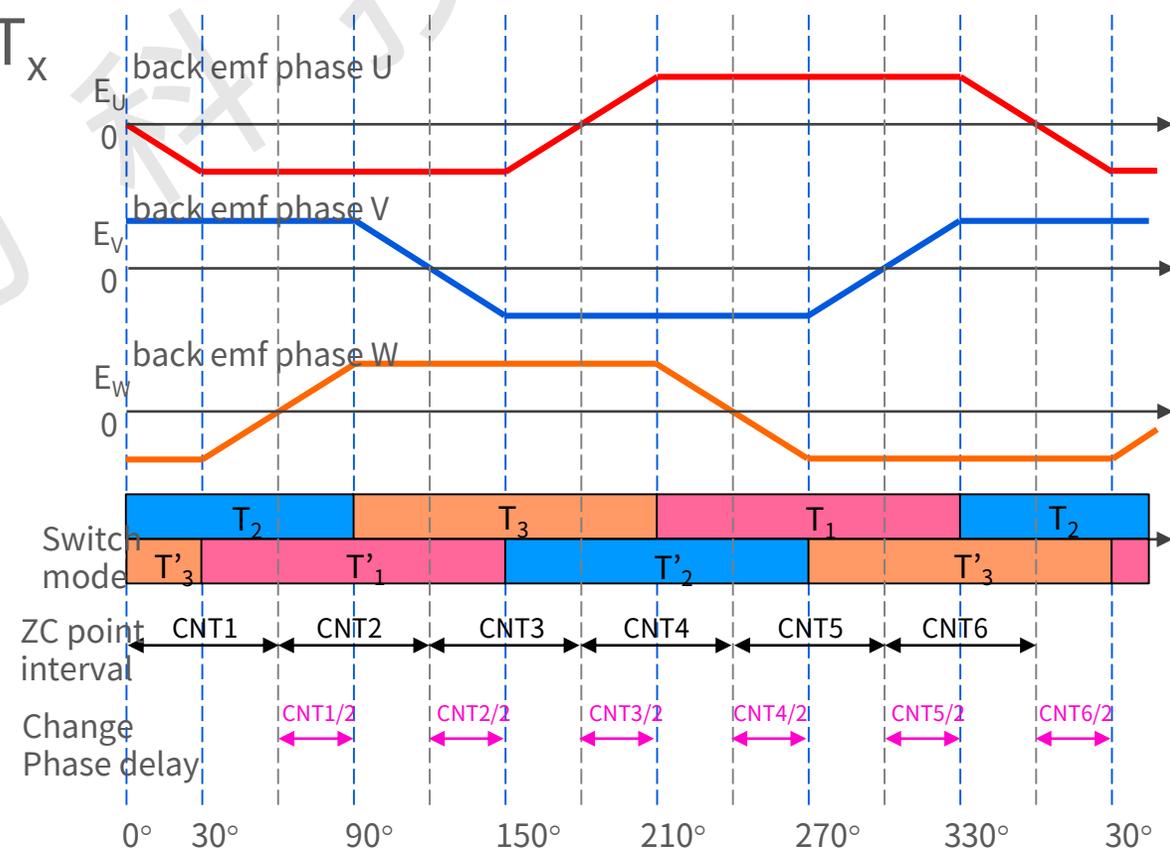
- 因换相点与反电势零交越点相差30°
- 各相反电势零交越点相差60°
- 故可利用定时器延迟最近一次零交越点间隔时间的一半后，进行换相



# 转轴速度估测

- 根据最低转速的霍尔波宽决定捕获定时器的计数频率周期 $T_{TMR}$
- 记录每个零交越点间的时间计数值  $CNT_x$
- 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

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



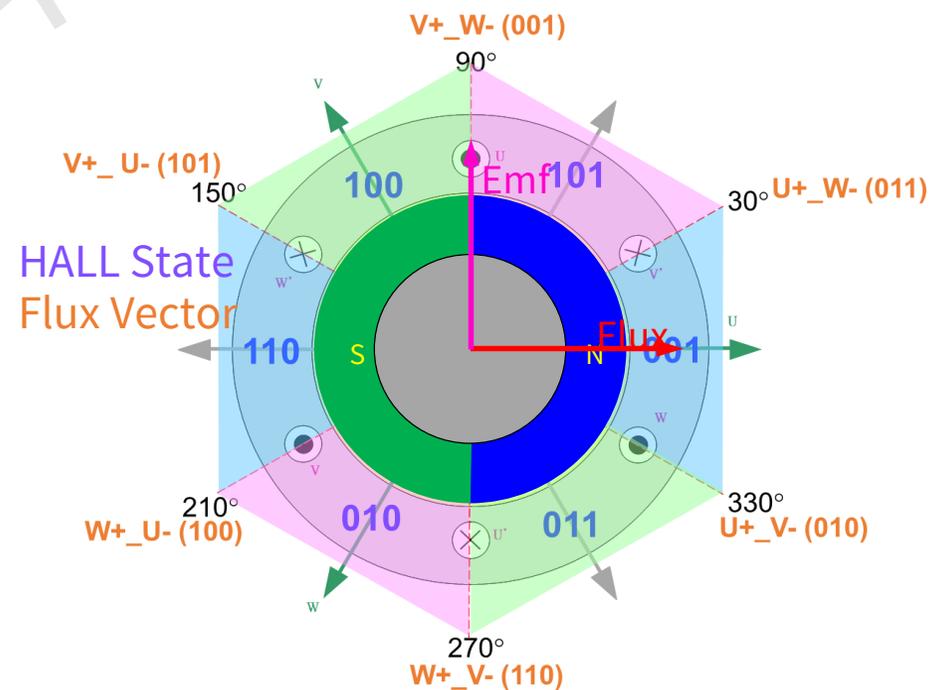
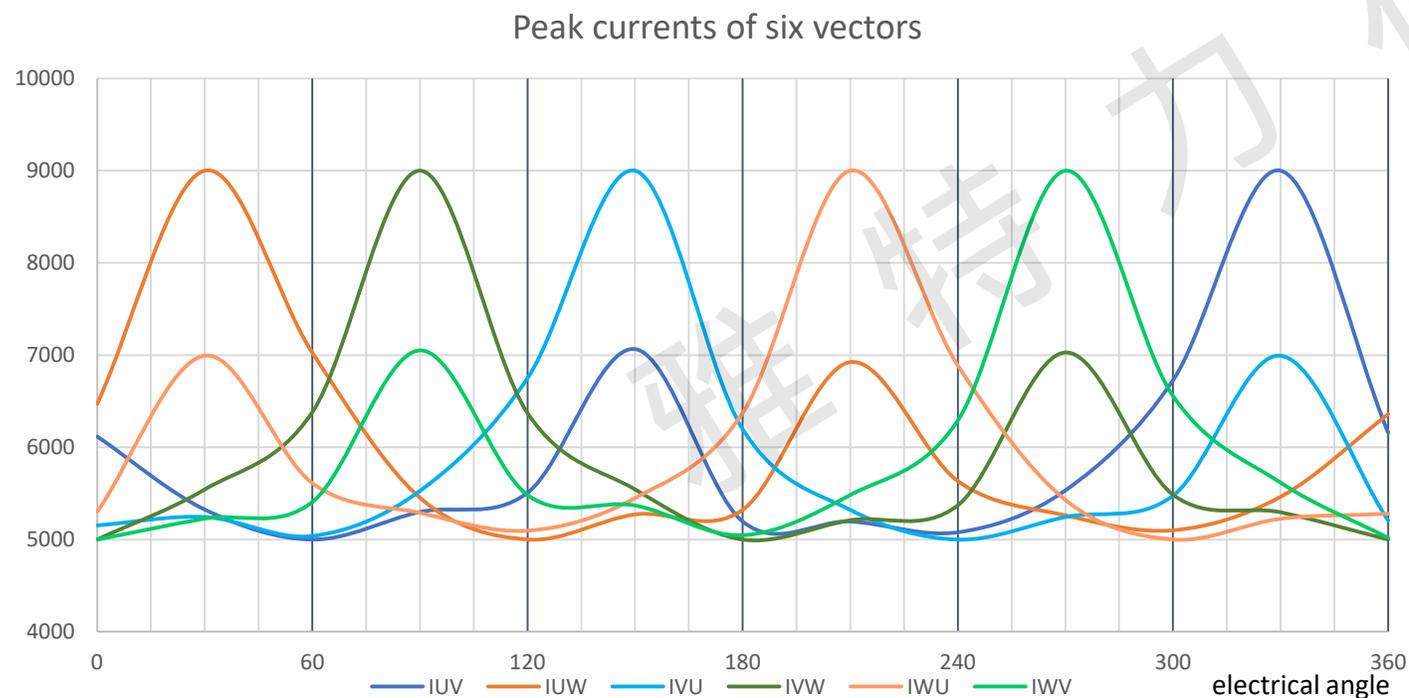
# 无感控制由静止状态启动方式

- 开环启动(强制换相)
  - 依照固定的六步换相顺序，逐步增加换相频率，推动转子加速
  - 当电机转速足够高、反电动势信号可用时，切换为反电动势换相控制
- 转子对齐启动
  - 在特定线圈组施加直流电流，使转子朝向特定角度对齐
  - 于该角度以大电流启动，使电机转速够快而得以侦测反电势
- 转子初始角度侦测
  - 利用转子磁极在不同位置时，测量各矢量对应的电流变化来估算初始位置
  - 于该角度以大电流启动，使电机转速够快而得以侦测反电势



# 转子初始位置侦测

- 依次输入6个两相脉冲电压矢量
- 根据对应的6个电流峰值与邻相电流估算转子位置

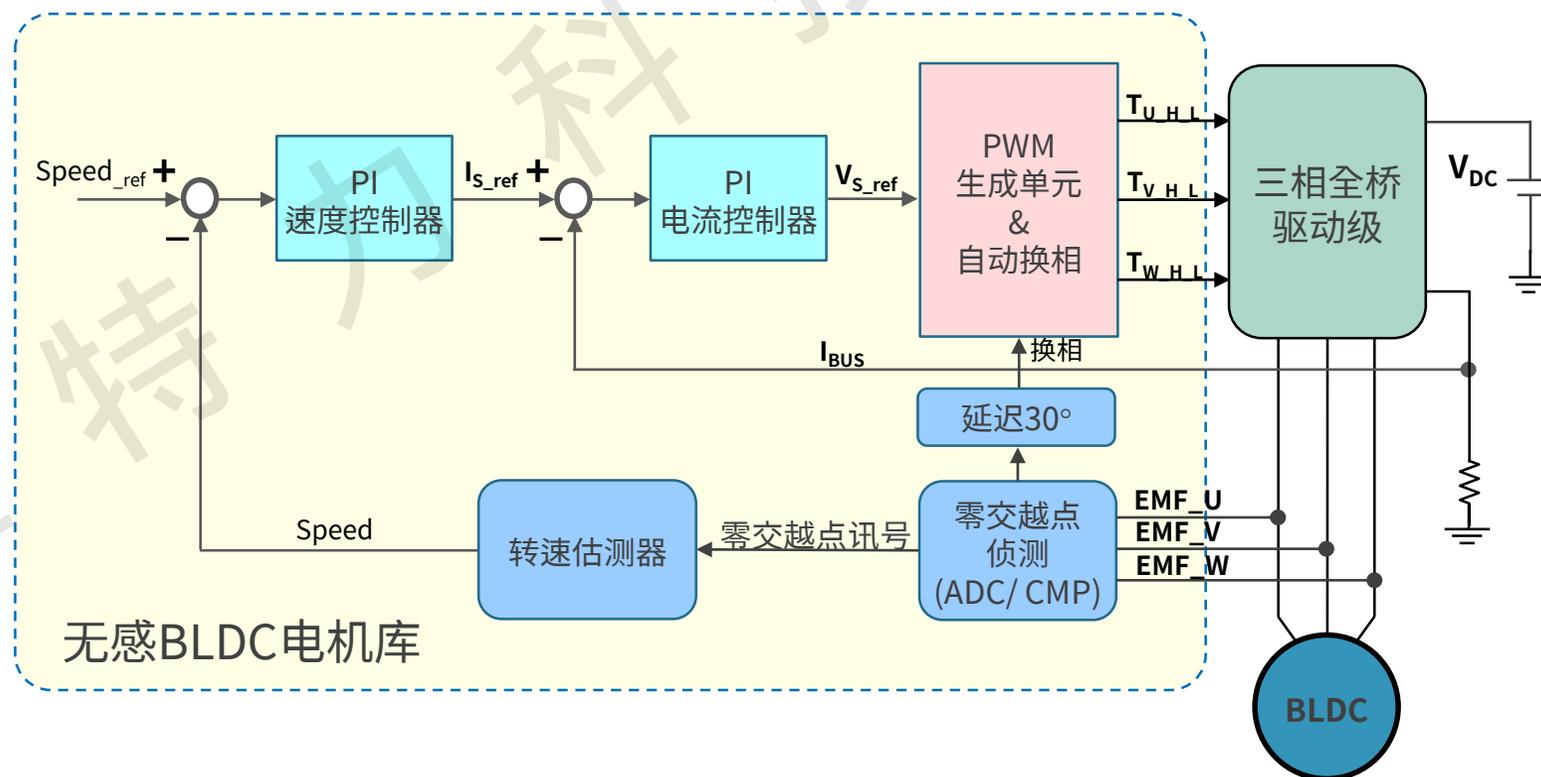


# BLDC无感电机库 架构解说



# 无感 BLDC电机库

- 驱动方法
  - 120°方波控制
- 相电流检测方法
  - 单电阻电流检测
- 初始转子位置估测方法
  - 两相电压矢量转子初始角度估测
- 换相时机估测方法
  - 比较器侦测反电势零交越点
  - ADC回授感测反电势零交越点
- 可实现无传感器控制方法
  - 120°方波电压控制
  - 转矩控制 (120°方波电流控制)
  - 转速控制
  - 回生刹车

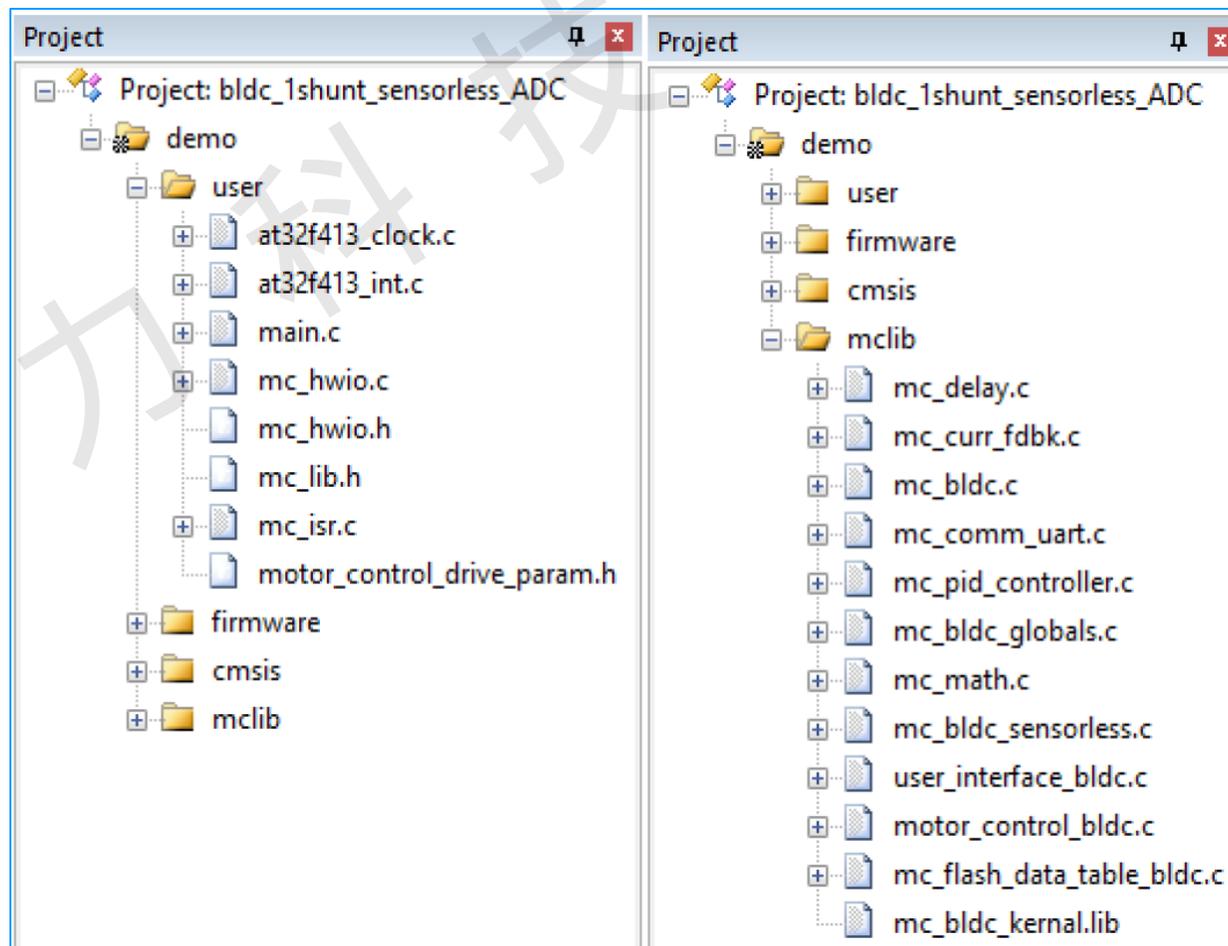


# BLDC电机库控制技术与应用方案

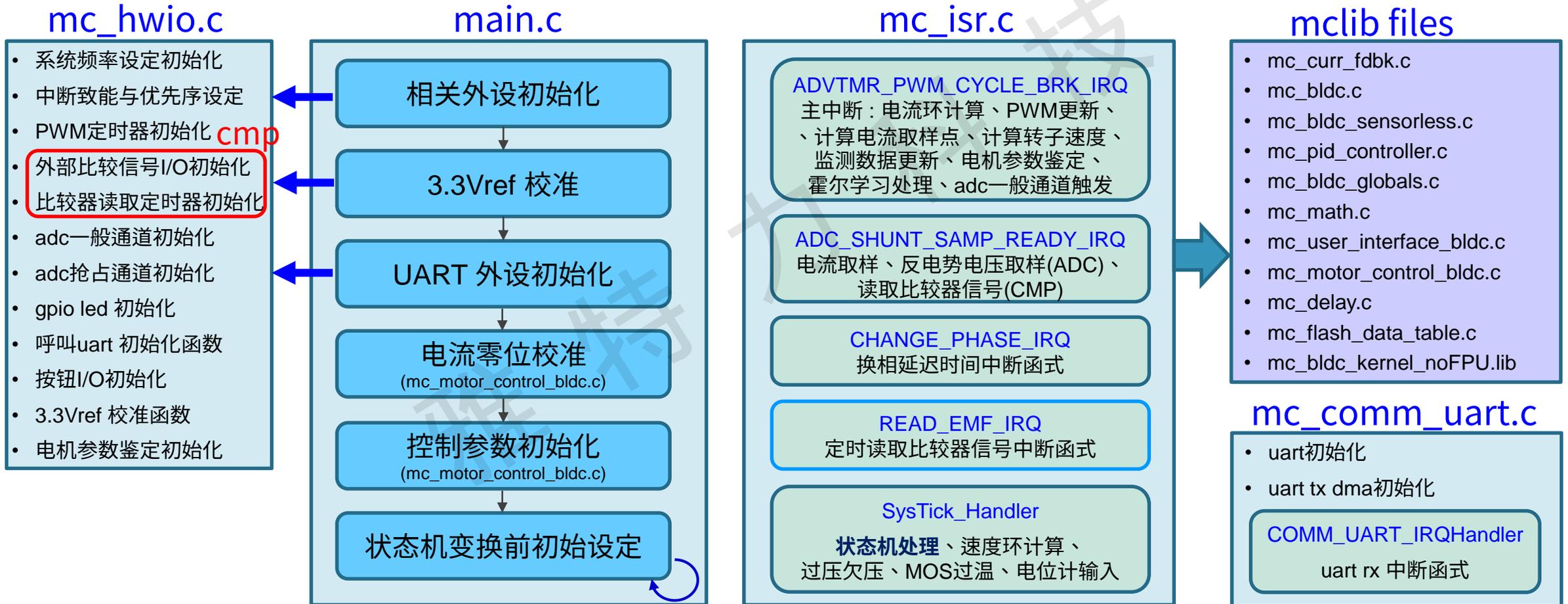
应用产品		高速风机	低速风机	电动工具	链锯	滑板车 电动自行车	电摩	扫地机
两相 驱动	120°导通法	●	●	●	●	●	●	●
	提前激磁弱磁控制			●		●	●	●
控制 回路	转矩控制	●	●	●	●	●	●	●
	速度控制	●	●	●	●			●
电流 感测	单电阻电流感测	●	●	●	●	●	●	●
有传 感器	霍尔传感器	●	●	●	●	●	●	●
无传 感器	转子初始角度侦测	●	●	●	●	●	●	●
	反电势零交越点信号回授	●	●	●	●	●	●	●
	反电势回授侦测零交越点	●	●	●	●	●	●	●

# 无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、无感电机库函数、全局变量设定与通讯函数等等

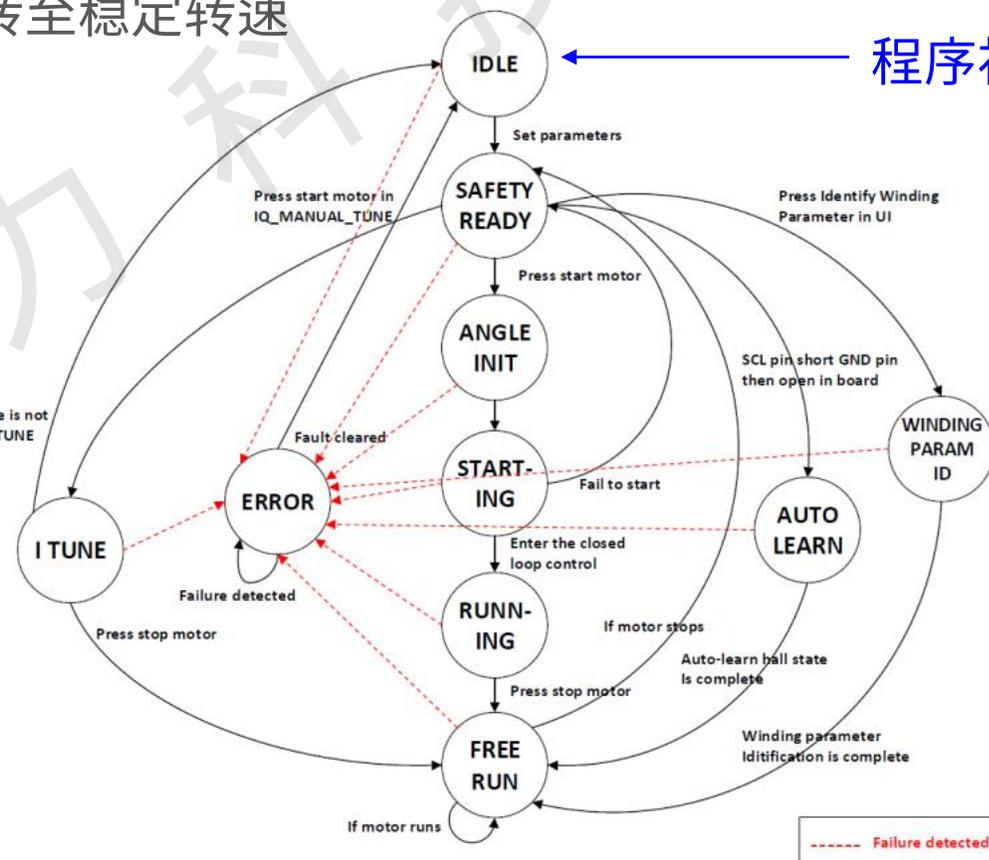


# BLDC无感控制程序流程图



# 程序状态机流程图

- ANGLE\_INIT：由静止状态启动(可使用三种不同启动方法)
- STARTING：可采定电压或定电流方式运转至稳定转速
- RUNNING：进入速度闭环
- 用户可于UI界面实时调整参数
- 或用外部电位计改变命令



# 电机库下载

ARTERY | 雅特力科技

关于雅特力 产品讯息 技术与开发支持 应用方案 新闻中心 联系雅特力

21ic LCES 天猫

应用文件支持

设计资源

电机控制

USB应用

以太网应用

培训中心

电机控制

USB应用

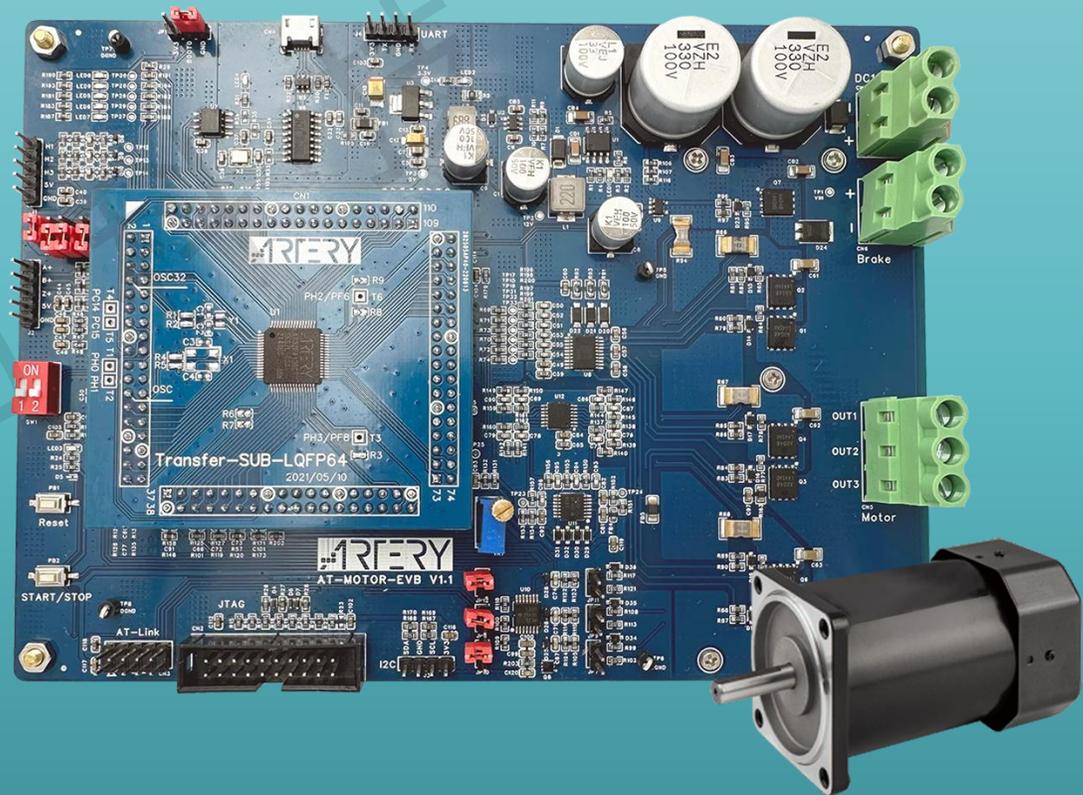
以太网应用

Quick Start Guide(快速入门指南)

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电机库调适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电机库调适不同型号带霍尔传感器的电机运转并应用在电动自行车、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电机库调适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

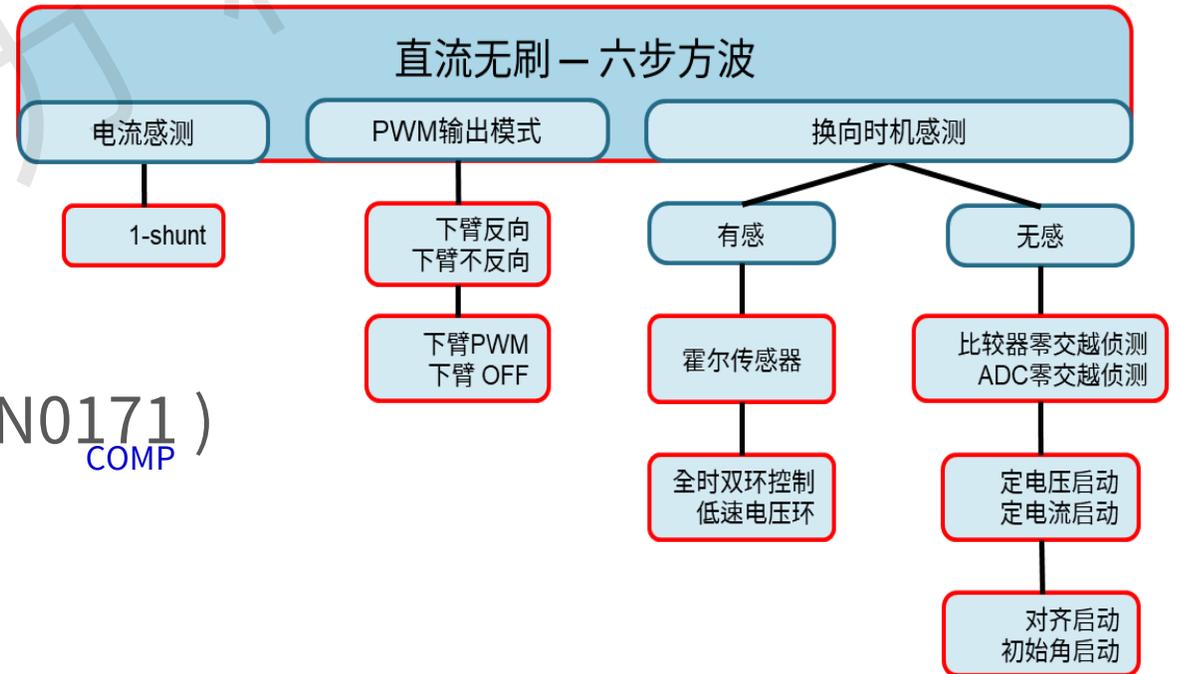
网址：[https://www.arterytek.com/cn/support/motor\\_control.jsp?index=0](https://www.arterytek.com/cn/support/motor_control.jsp?index=0)

# AT32 电机库 BLDC 快速上手 操作讲解



# BLDC电机控制库支持

- 支援BLDC 6-step控制
- 提供AT32F421、AT32F413、AT32F415、AT32F425、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)
- 无感专案的说明可参阅(AN0170, AN0171 )
- BLDC 无感快速入门指南(AN0214)



# 参数头文件 - 电机控制形式 (六步方波)

motor\_control\_drive\_param.h  
控制型式

```

/* internal clock or external crystal */
// #define INTERNAL_CLOCK_SOURCE

/* choose MOTOR_EVB_BOARD version */
#define AT_MOTOR_EVB_V2
// #define AT_MOTOR_EVB_V1

/* mosfet low side complement (high side)
#define COMPLEMENT
/* gate driver low side inverting logic
#define GATE_DRIVER_LOW_SIDE_INVERT

/* EMF compensate function */
// #define EMF_COMPENSATE

/* choose sensor */
// #define HALL_SENSORS
#define SENSORLESS
#define BLDC_SENSORLESS_ADC
// #define BLDC_SENSORLESS_COMP

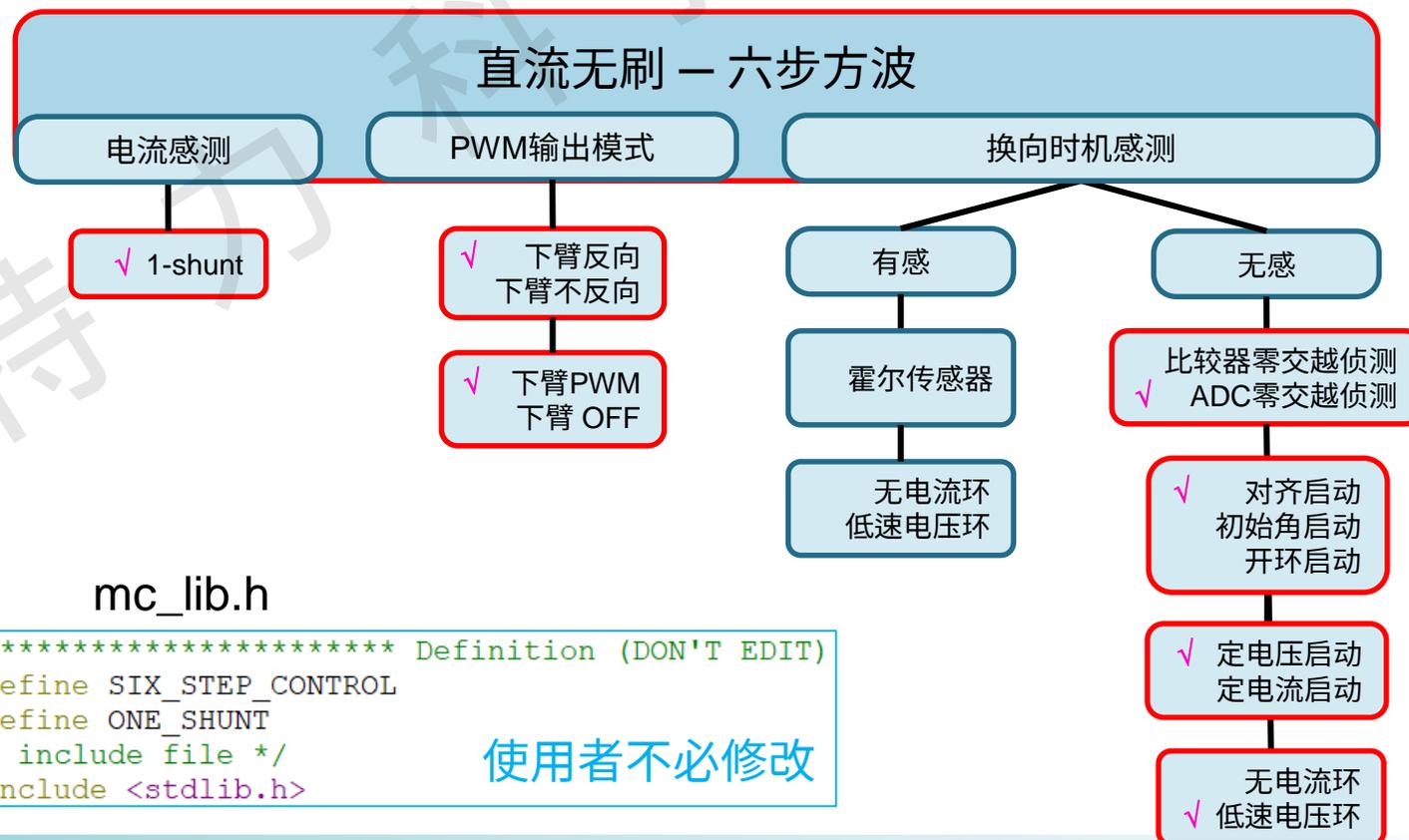
/* choose remove current-loop-control or
// #define WITHOUT_CURRENT_CTRL

/* choose low speed control or not */
#define LOW_SPEED_VOLT_CTRL

#if defined SENSORLESS
/* choose const current/voltage start-up
// #define CONST_CURRENT_START
#define CONST_VOLTAGE_START

/* choose how to start up */
// #define INIT_ANGLE_STARTUP
#define ALIGN_AND_GO_STARTUP
// #define OPENLOOP_STARTUP
    
```

## ● 选择宏定义宣告项目，完成电机控制形式设定



mc\_lib.h

```

/***** Definition (DON'T EDIT) *****/
#define SIX_STEP_CONTROL
#define ONE_SHUNT
/* include file */
#include <stdlib.h>
    
```

使用者不必修改

# 参数头文件 - 电机参数

motor\_control\_drive\_param.h

电机参数 (Motor-related parameter)

```
/* ***** Motor-related parameter ***** */
/* Motor parameters */
#define RS_LL (1.89) /* Stator resistance */
#define LS_LL (0.002387) /* Stator inductance */
#define POLE_PAIRS (8/2)
#define KE (0.003437f) /* Back EMF constant */
#define NOMINAL_CURRENT (1.7)

/* angle detect duty */
#define ANGLE_INIT_DETECT_DUTY ((int16_t) (0.15*ANGLE_INIT_PERIOD))
#define ANGLE_INIT_I_DIFF ((int16_t) 500)

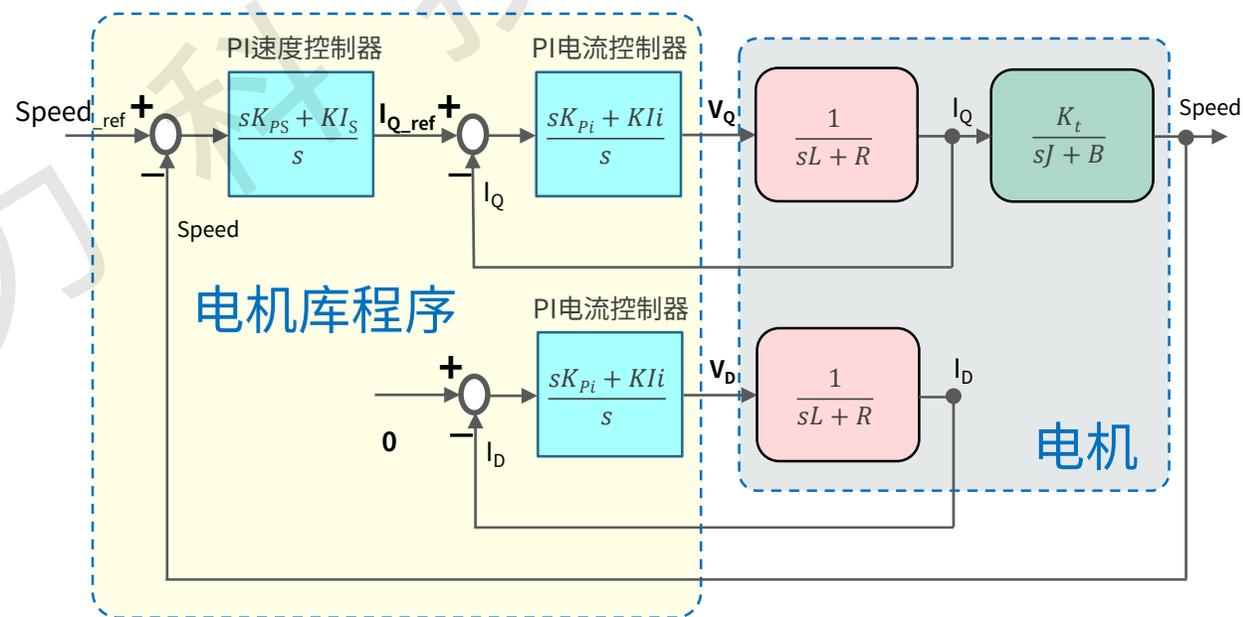
/* hall learn table */
#if defined (SENSORLESS)
#define HALL_LEARN_DIR (0) /* Polarity, 0 or 1 */
#define HALL_LEARN_0_STATE (2) /* BH-CL */
#define HALL_LEARN_1_STATE (3) /* BH-AL */
#define HALL_LEARN_2_STATE (1) /* CH-AL */
#define HALL_LEARN_3_STATE (5) /* CH-BL */
#define HALL_LEARN_4_STATE (4) /* AH-BL */
#define HALL_LEARN_5_STATE (6) /* AH-CL */
#else
#define HALL_LEARN_DIR (0) /* Polarity, 0 or 1 */
#define HALL_LEARN_0_STATE (1) /* BH-CL */
#define HALL_LEARN_1_STATE (5) /* BH-AL */
#define HALL_LEARN_2_STATE (4) /* CH-AL */
#define HALL_LEARN_3_STATE (6) /* CH-BL */
#define HALL_LEARN_4_STATE (2) /* AH-BL */
#define HALL_LEARN_5_STATE (3) /* AH-CL */
#endif
#endif
```

- 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 – 控制环相关参数

motor\_control\_drive\_param.h  
控制参数 (Control-related parameter)

```
#define PWM_FREQ (15000) /*!< Hz */
#define MOTOR_CONTROL_MODE (motor_control_mode)(SPEED_CTRL)
#define CTRL_SOURCE (ctrl_source_type)(CTRL_SOURCE_SOFTWARE)
#define UI_UART_BAUDRATE_ (1500000UL) /*!< bit/s */
/* I-SAMPLE PARAMETER */
#define I_SAMP_MIN_TIME (180) /*!< nsec */
#define I_SAMP_DELAY (1000) /*!< nsec */
/* SPEED */
#define SPEED_FILTER_TIMES (6)
#define SPD_LP_BANDWIDTH (300.0f) /* 2*pi*freq */
#define PWM_IN_FILTER_TIMES (5)
#define SPD_CMD_FILTER_TIMES (5)
/* SPEED PARAMETER */
#define MIN_SPEED_RPM (10)
#define MAX_SPEED_RPM (7200) /*!< rpm */
#define MAX_CCW_SPEED_RPM (7200) /*!< rpm */
#define ACC_SPD_SLOPE (5) /*!< rpm/ms */
#define DEC_SPD_SLOPE (5)
/* pi parameter */
#define PID_IS_KP_DEFAULT 6000
#define PID_IS_KI_DEFAULT 11300
#define PID_IS_KP_DIV 1024
#define PID_IS_KP_DIV_LOG LOG2(PID_IS_KP_DIV)
#define PID_IS_KI_DIV 32768
#define PID_IS_KI_DIV_LOG LOG2(PID_IS_KI_DIV)
```



$$IS\_Kp = PID\_IS\_KP\_DEFAULT / PID\_IS\_KP\_DIV$$
$$IS\_Ki = PID\_IS\_KI\_DEFAULT / PID\_IS\_KI\_DIV$$

# 外设定义头文件 - 外设参数定义

## 外设定义参数 (mc\_hwio.h)

```

/* 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	
<b>VBAT</b>	1	<b>VBAT</b>	1	<b>VBAT</b>
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)
<b>VSSA</b>	8	<b>VSSA</b>	12	<b>VSSA</b>
<b>VDDA</b>	9	<b>VDDA</b>	13	<b>VDDA</b>
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

a. 在线调整控制参数

b. 实时观察输出响应

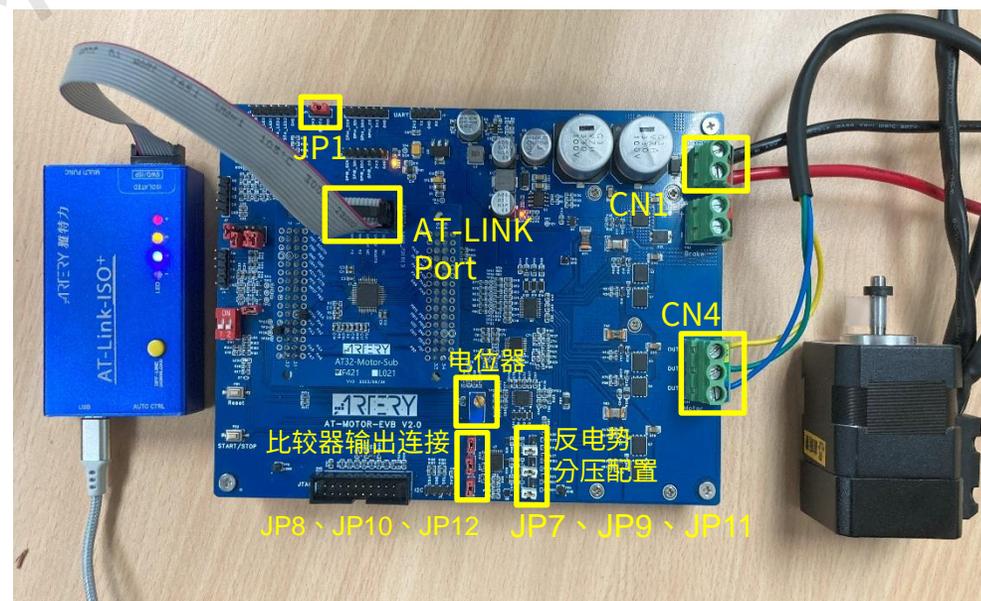
c. 控制参数写入Flash

实时监控看响应特性与数据

实时监控输出响应

# 控制器接线

- 检查JP1 Boot0是否接地(GND)
- 根据母线电压选择是否短接JP7、JP9、JP11 ( $V_{BUS} > 34V$ )
- 于比较器模式须将JP8、JP10、JP12短接，将比较器信号连接至MCU 接脚
- 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CN1 (依标示)
- 设定电源供应器电压24V / 3A
- 开启电源供应器输出
- PC端执行上位机程序后建立联机  
( ArteryMotorMonitor.exe)



电机与开发板接线图

# 监控界面软件运行操作 - 自整定功能

The screenshot displays the Artery Motor Monitor V2.1.18 software interface. The window title is "Artery Motor Monitor V2.1.18" and the menu bar includes "文件", "设置", and "帮助". The main interface is divided into several sections:

- 程序状态 (Program Status):** Located at the top left, it shows the connection "COM37: ATLink-USART" and "AT32F421\_ML\_V2.1.3". Below this is a "状态" (Status) section with a list of error states, including "ESC\_STATE\_SAFETY\_READY" and "no error".
- 线圈参数识别 (Winding Parameter Identification):** Located in the middle left, it shows a table with identified process state and parameters like RS\_LL and LS\_LL.
- PI参数自整定 (PI Parameter Auto-tuning):** Located in the middle right, it shows a table with auto-tuned PI parameters for the current controller, such as Torque KP, Torque KI, and Torque KI DIV.
- 启动/停止电机 (Start/Stop Motor):** Located at the bottom left, it contains buttons for "启动电机" (Start Motor) and "停止电机" (Stop Motor).
- 错误状态解除 (Error State Release):** Located at the bottom left, it contains a button for "错误解除" (Error Release).
- 写入闪存 (Write Flash):** Located at the bottom left, it contains a button for "写闪存" (Write Flash).
- 控制模式 (Control Mode):** Located at the bottom left, it shows a dropdown menu for "BLDC - sensorless (ADC) 控制型式 (read)" and "Speed Control".
- 控制来源 (Control Source):** Located at the bottom left, it shows a dropdown menu for "software control".
- 运行日志 (Operation Log):** Located at the bottom right, it shows a table with log entries including timestamps and messages like "Exec Fault ACK...ok" and "Set REG 10 = 3000 ok".

Red boxes and labels highlight these key features: "程序状态", "线圈参数识别", "PI参数自整定", "启动/停止电机", "错误状态解除", "写入闪存", "控制模式", "控制来源", and "传输状态记录".

# 监控界面软件运行操作 - 电流环参数实时调整

The screenshot displays the Artery Motor Monitor V2.1.18 software interface, which is used for real-time monitoring and adjustment of motor parameters. The interface is divided into several functional areas:

- Program Status (程序状态):** Located at the top left, it shows the current state as `ESC_STATE_SAFETY_READY`.
- Error Status (错误状态):** A list of error conditions is shown, with `no error` selected. Other errors include over voltage, under voltage, over temperature, over current, encoder, hall, parameter identify, hall learn, and startup errors.
- Command (命令):** Buttons for `启动电机` (Start Motor) and `停止电机` (Stop Motor) are present.
- Parameter Adjustment (参数调整):** The central area contains several configuration tables:
  - Unit step config (步阶电流命令参数):**

Current Tune target current	0.500	A
Current Tune total period	50	ms
Current Tune step period	5	ms
  - Speed Voltage Controller (Without Current control loop) (电流控制器PI 参数):**

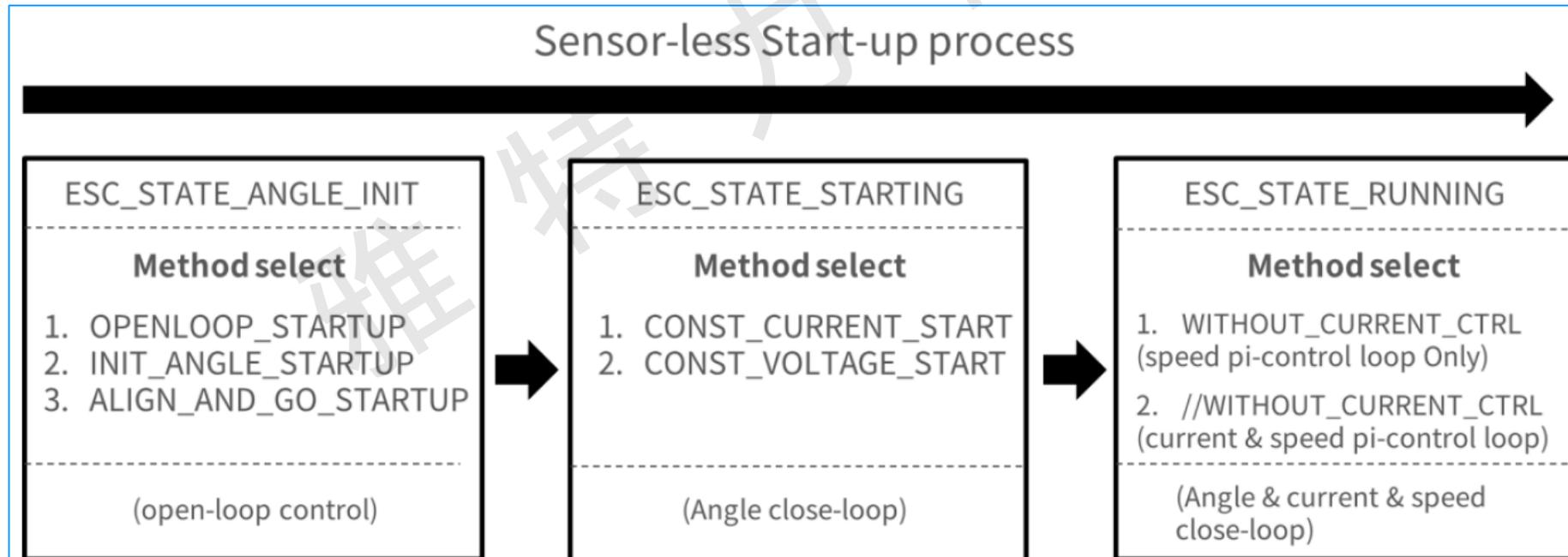
Speed Voltage KP	3000
Speed Voltage KI	100
Speed Voltage KP DIV	1024
Speed Voltage KI DIV	1024
  - Sensor-less control (six-step) (IQ Current control):**

Torque KP	6000
Torque KI	11300
Torque KP DIV	1024
Torque KI DIV	32768
- Run Log (运行日志):** A table at the bottom right records system events:

1	08:46:10	Load ok
2	08:46:10	Set REG 8 = 3 ok
3	08:39:50	Exec Fault ACK...ok
4	18:46:11	Exec Stop Motor...ok
- Control Mode (控制模式):** A dropdown menu at the bottom left is set to `BLDC - sensorless (ADC) 控制型式 (read)`.
- Real-time Monitoring (实时监看):** A graph on the right side shows the `Torque(current) reference (Iq)(A)` and `Torque(current) measured (Iq)(A)` over time. The reference is a step function, and the measured current follows it closely, demonstrating the current loop's response.

# 直流无刷电机无感启动流程

- ANGLE\_INIT：选择开环、初始角侦测或对齐转子方式启动
- STARTING：角度闭环，可选定电压或定电流方式运转进入速度闭环
- RUNNING：进入速度闭环，可选择是否使用电流环



# 启动参数介绍

- 对齐启动 (ALIGN\_AND\_GO\_STARTUP)

```
/* lock start-up */
```

```
#define LOCK_VOLT (2.5) /*!< V */ 转子对齐时间
```

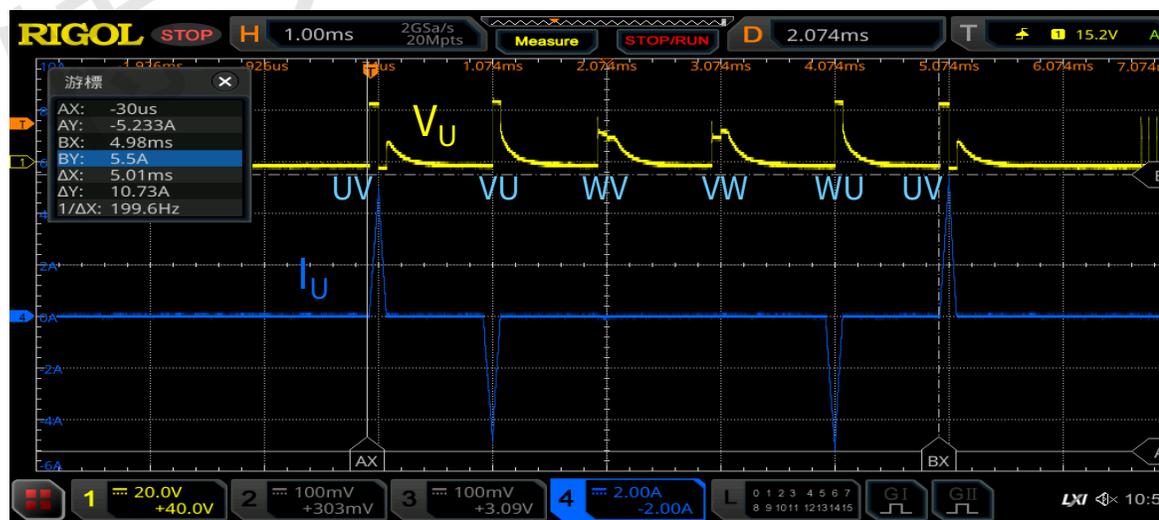
```
#define LOCK_PERIOD (500) /*!< msec */ 转子对齐时间
```

- 初始角度侦测 (INIT\_ANGLE\_STARTUP)

```
/* angle detect duty */ (0xFFFF)
```

```
#define ANGLE_INIT_DETECT_DUTY ((int16_t) (0.15*ANGLE_INIT_PERIOD))
```

```
#define ANGLE_INIT_I_DIFF ((int16_t) 500)
```



# STARTING\_STATE 角度闭环参数设定

- 启动过程控制模式

```
#define START_CURRENT (0.6) /*!< A */  
#define START_VOLTAGE (1.5) /*!< V */
```

- 进入RUNNING前换相次数

```
#define SENSE_HALL_TIMES (8)  
#define REBOOT_PERIOD_MS (1000) /*!< msec */ 判定启动失败延迟时间
```



定电压启动模式



定电流启动模式

# 监控界面软件运行操作 - 开环控制

The screenshot shows the Artery Motor Monitor V2.1.18 interface with several key areas highlighted:

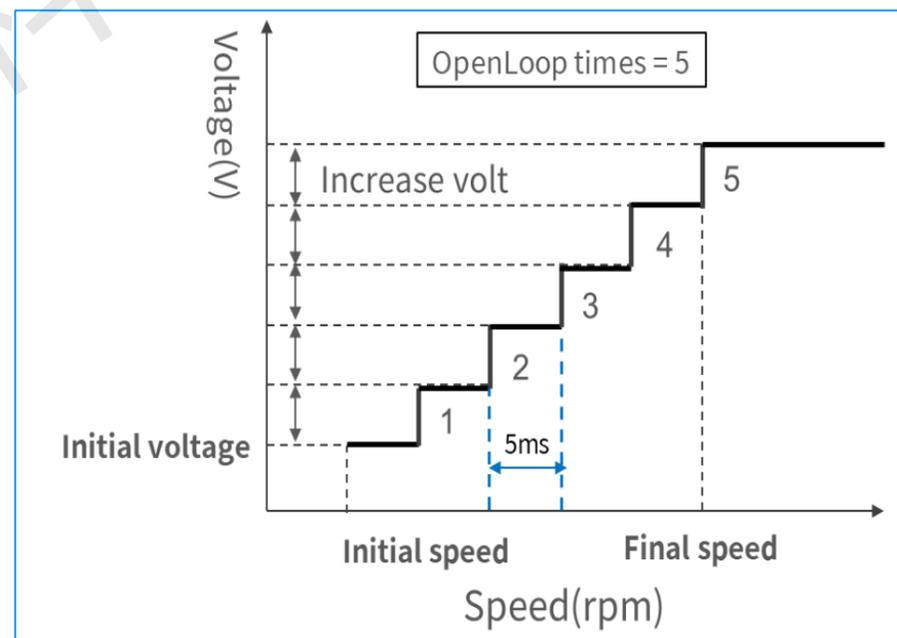
- 程序状态 (Program Status):** ESC\_STATE\_SAFETY\_READY
- 错误状态 (Error Status):** A list of error types with checkboxes, currently showing 'no error' selected.
- 启动/停止电机 (Start/Stop Motor):** Buttons for '启动电机' (Start Motor) and '停止电机' (Stop Motor).
- 错误解除 (Error Clear):** Button to clear error status.
- 写入闪存 (Write Flash):** Button to write to flash memory.
- 控制模式 (Control Mode):** Dropdown menu set to 'BLDC - sensorless (ADC) 控制型式 (read)'.
- 开环控制参数 (Open Loop Control Parameters):**

OpenLoop initial voltage	0.712	V
OpenLoop initial speed	100	rpm
OpenLoop increase volt	0.003	V
OpenLoop final speed	350	rpm
OpenLoop times	90	
- Speed Voltage Controller (Without Current control loop):**

Speed Voltage KP	3000
Speed Voltage KI	100
Speed Voltage KP DIV	1024
Speed Voltage KI DIV	1024
- Sensor-less control (six-step):**

Start Current	0.199	A
Start Voltage	1.422	V
EMF low speed offset (Rising)	35	
EMF low speed offset (Falling)	35	
EMF high speed offset (Rising)	-100	
EMF high speed offset (Falling)	-100	
- 传输状态记录 (Transmission Status Record):**

1	08:56:22	Load ok
2	08:56:22	Set REG 8 = 0 ok
3	08:51:40	Load ok
4	08:51:40	Set REG 8 = 5 ok



开环控制参数示意图

# ADC零交越点侦测参数

- EMF low speed offset (rising, falling) : PWM OFF 取样
- EMF high speed offset (rising, falling) : PWM ON 取样
- 取样点切换磁滞设定

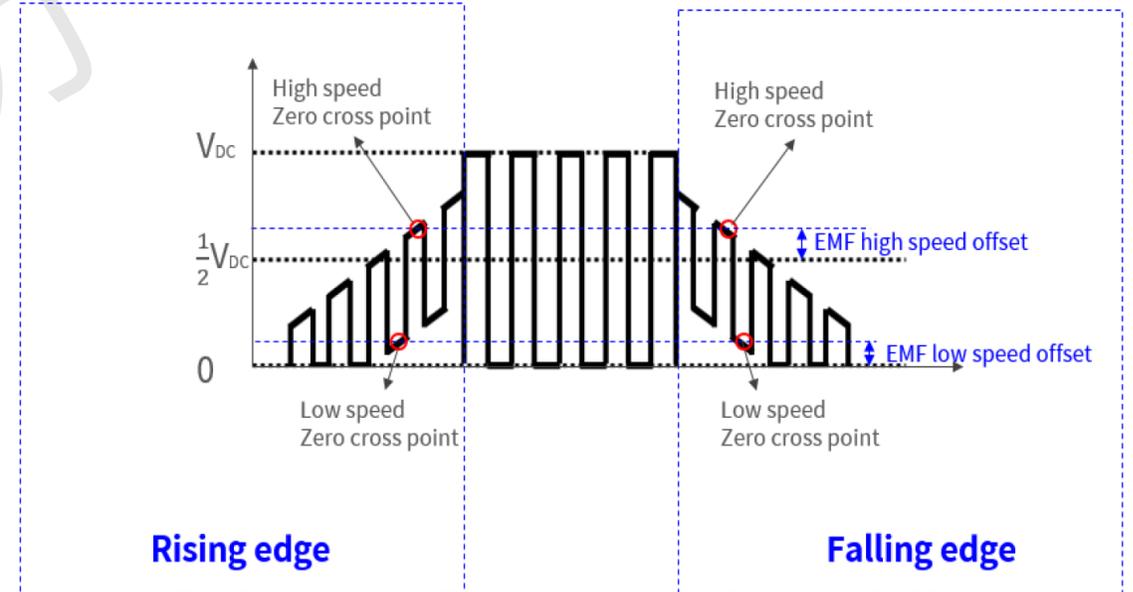
```
#define EMF_CHANGE_PERCENT_H (55) /*!< pwm duty(%) */  
#define EMF_CHANGE_PERCENT_L (35) /*!< pwm duty(%) */
```

- 低速电压控制磁滞切换设定

```
#define HYSTERESIS_LOW_SPEED (800) /*!< rpm */  
#define HYSTERESIS_HIGH_SPEED (1000) /*!< rpm */
```

Sensor-less control(six-step)		
Start Current	0.599	A
Start Voltage	1.429	V
EMF low speed offset(Rising)	35	
EMF low speed offset(Falling)	35	
EMF high speed offset(Rising)	-100	
EMF high speed offset(Falling)	-100	

六步方波反电势调试参数界面



六步方波反电势调试参数示意图

# 监控界面软件运行操作 - 速度环参数实时调整

Artery Motor Monitor V2.1.18

文件 设置 帮助

COM37: ATLink-USART 关闭  绘图

AT32F421\_ML\_V2.1.3

状态 **程序状态**

ESC\_STATE\_SAFETY\_READY

no error  
 over voltage error  
 under voltage error  
 over temperature error  
 over current error  
 encoder error  
 hall error  
 parameter identify error  
 hall learn error  
 startup error

**错误状态**

命令 **启动/停止电机**

编码器对齐

**错误状态解除**

**写入闪存**

BLDC - sensorless (ADC) **控制型式 (read)**

Speed Control **控制模式**

控制源  
software control

**控制模式**

基本 参数调整 自整定功能

Speed control(With current control loop)

Speed KP	1000	
Speed KI	20	
Speed KP DIV	4096	
Speed KI DIV	32768	
Speed acceleration	5	rpm/ms
Speed deceleration	5	rpm/ms

**有电流环速度控制  
PI参数**

Speed Voltage Controller(Without Current control loop)

Speed Voltage KP	3000	
Speed Voltage KI	100	
Speed Voltage KP DIV	1024	
Speed Voltage KI DIV	1024	

**无电流环速度控制  
PI参数**

Sensor-less control(six-step)

Start Current	0.199	A
Start Voltage	1.422	V
EMF low speed offset(Rising)	35	
EMF low speed offset(Falling)	35	
EMF high speed offset(Rising)	-100	
EMF high speed offset(Falling)	-100	

Target speed

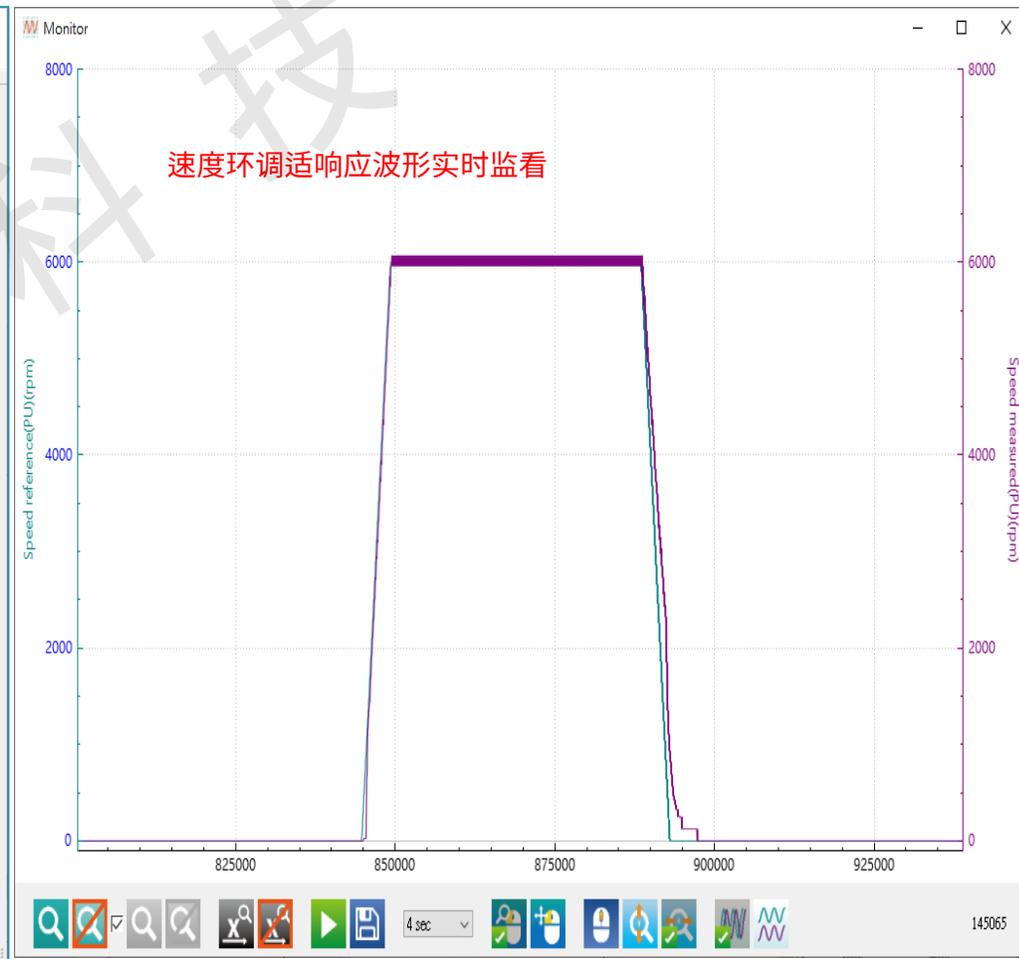
Speed reference 0 rpm

**运行目标**

运行日志

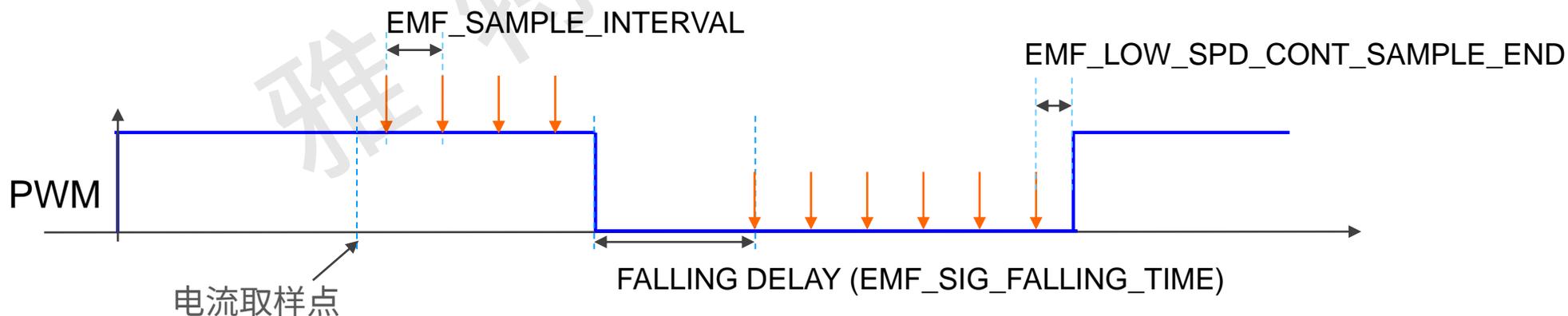
1	08:51:40	Load ok
2	08:51:40	Set REG 8 = 5 ok
3	08:49:45	Load ok
4	08:49:45	Set REG 8 = 0 ok

**传输状态记录**



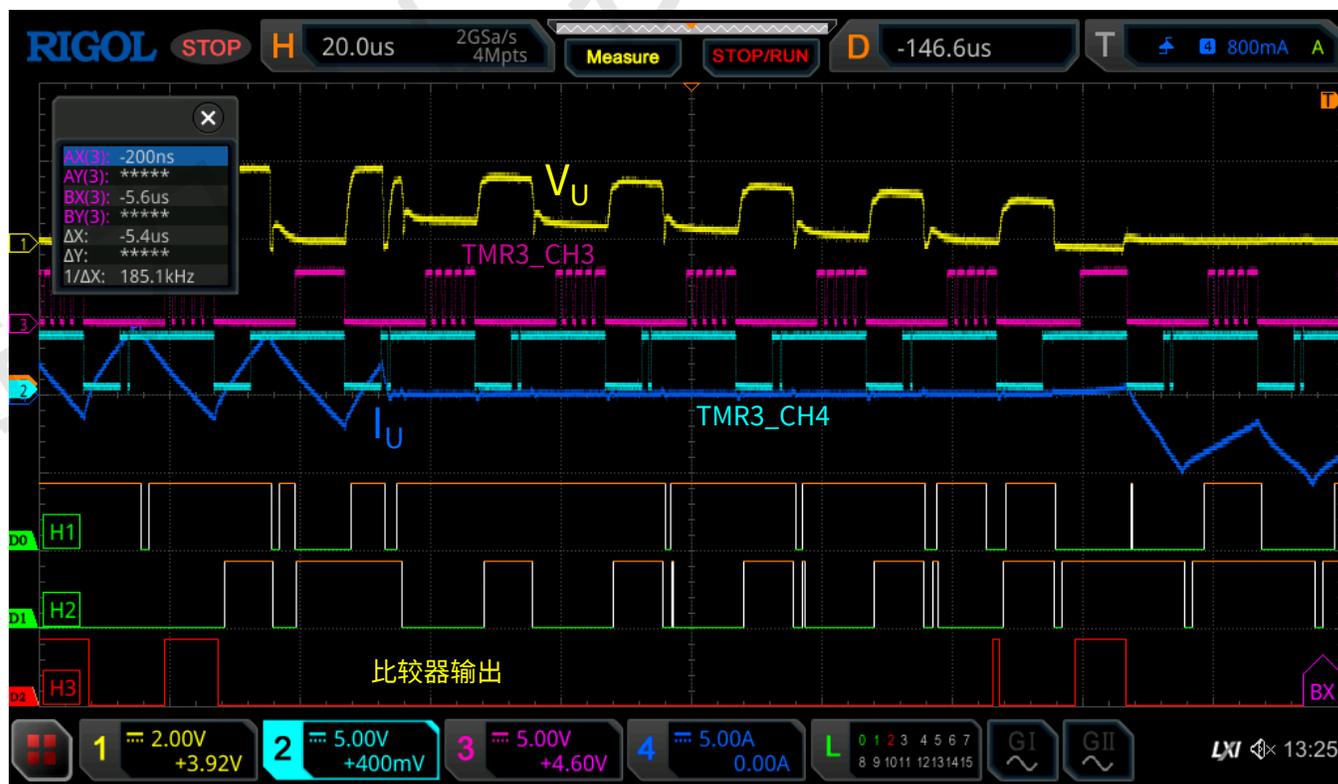
# 比较器连续读取模式参数设定

```
#ifndef EMF_CONTINUOUS_SAMPLE
#define EMF_SAMPLE_INTERVAL      (0.05 * PWM_PERIOD)  /*!< 5% pwm duty*/
#define SENSE_GPIN_DELAY        (100)                /*!< the delay from ovf of emf timer to sense
#define EMF_LOW_SPD_CONT_SAMPLE_END (PWM_PERIOD - EMF_SAMPLE_INTERVAL - SENSE_GPIN_DELAY)
#define EMF_AVOID_NOISE_TIMES    (5)
#else
#define EMF_AVOID_NOISE_TIMES    (1)
#endif
#define EMF_SIG_FALLING_TIME      (21.5)              /*!< usec */
```



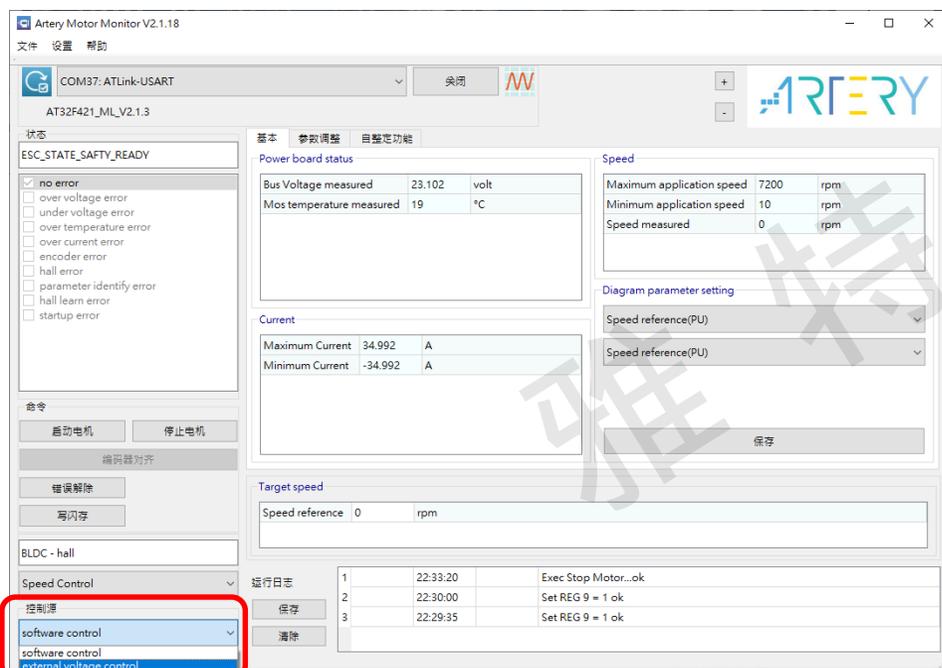
# 比较器连续读取实测图

- 使用定时器根据占空比决定连续取样时机
- 本例中使用TMR3为定时器
- TMR3\_CH3触发PWM OFF区间连续读取
- TMR3\_CH4触发PWM ON区间连续读取



# 外部命令控制

- 在控制源选择外部电压控制，SW1-1可设定转向
- 或在motor\_control\_drive\_param.h头文件中定义'CTRL\_SOURCE'



外部控制源



提问时间

10分钟

Q & A

# ARTERY 联系我们



## 重庆办公室

重庆市九龙坡区科城路60号康田西锦荟1栋10F  
+86-23-6868 8899

## 苏州办公室

苏州市工业园区通园路699号港华大厦1501  
+86-0512-6835 6375

## 上海办公室

上海市长宁区仙霞路317号B栋512室  
+86-0512-6835 6375

## 深圳办公室

深圳市福田区天安数码城创新科技广场二期西座603室  
+86-0755-8390 0669 分机62568

## 台湾办公室

新竹市金山八街1号五楼  
+886-3-577 8788



中文官网



微信公众号



English Website

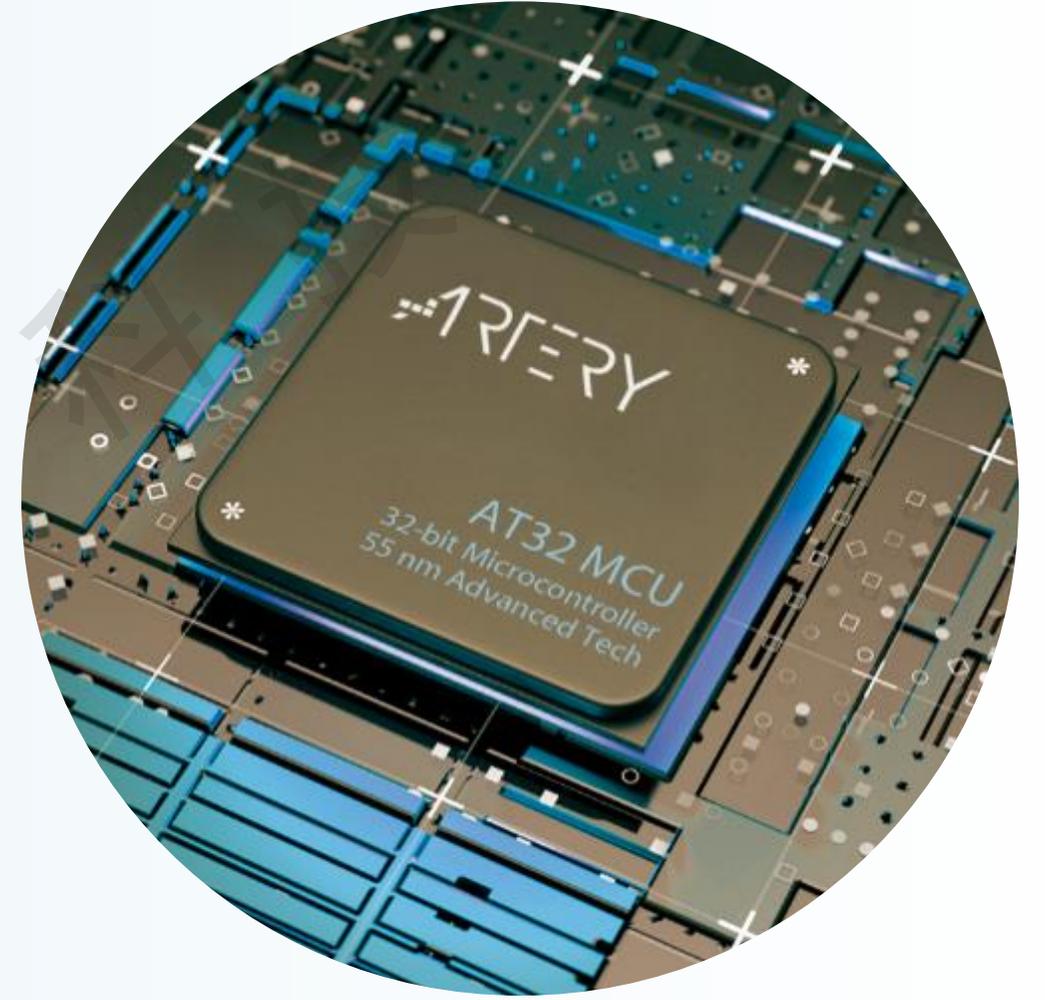


Linkedin

销售咨询 [sales\\_artery@arterytek.com](mailto:sales_artery@arterytek.com)

技术服务 [support\\_artery@arterytek.com](mailto:support_artery@arterytek.com)

**Thank you!**



雅特力