

基于STM32的 PMSM FOC软件库 培训



Plan

- ❑ 直流无刷马达介绍
- ❑ FOC 基础
- ❑ Clark 及 Parke 坐标系转换
- ❑ Circle limitation
- ❑ 磁链及力矩控制器
- ❑ 电流读取
 - ❑ 3 电阻法
 - ❑ 电流传感器法
- ❑ 转子的速度/位置反馈：
 - ❑ Hall 传感器
 - ❑ 正交编码器

直流无刷马达介绍

■ 两种直流无刷马达

■ **PMSM**：永磁同步马达

■ **BLDC**：直流无刷马达

■ 相似性

■ 同步转矩；

■ 转子为永磁体；

■ AC定子电流

■ 差异性

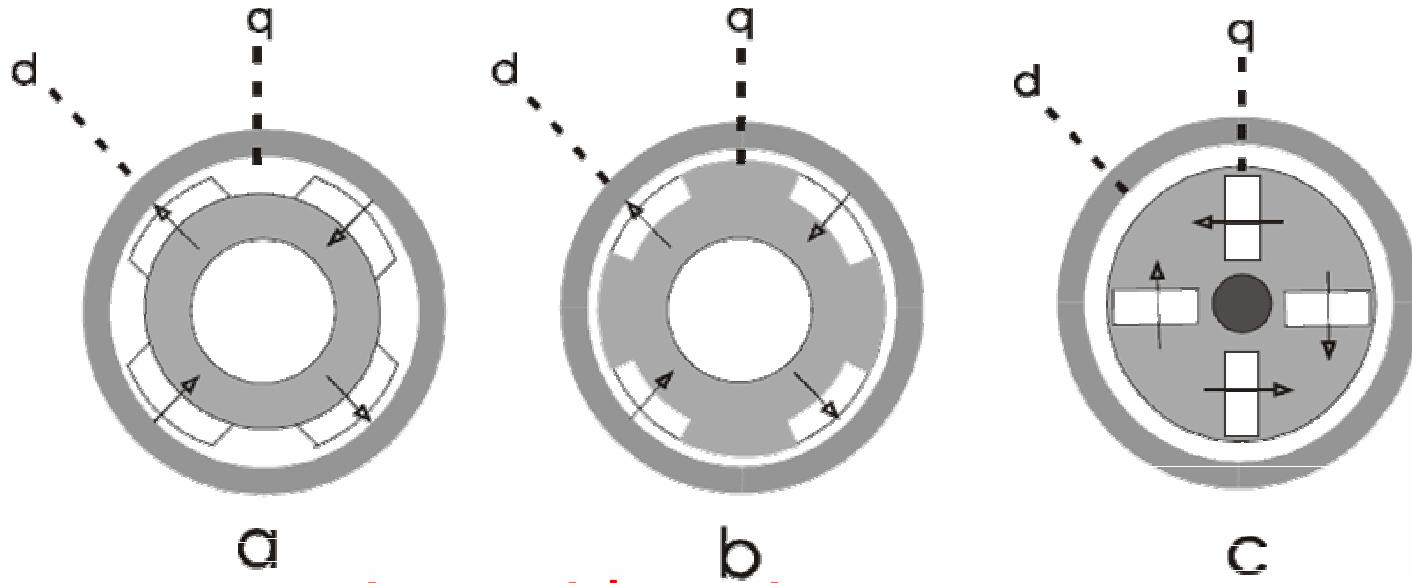
■ PMSM的反电动势为正弦波，而BLDC的为梯形波；

■ BLDC的电感较低；

■ **PMSM需要正弦的定子电流来产生稳定的转矩，而BLDC的定子电流为梯形波**

PMSM: 永磁同步马达

不同的转子结构：



$$L_a < L_b < L_c$$

- a) SM-PMSM: 表面贴式磁钢 ($L_d=L_q$) .
- b) & c) IPM-SM: 内嵌式磁钢($L_d < L_q$).
特别地， b) 内插式; c) 放射状内埋式

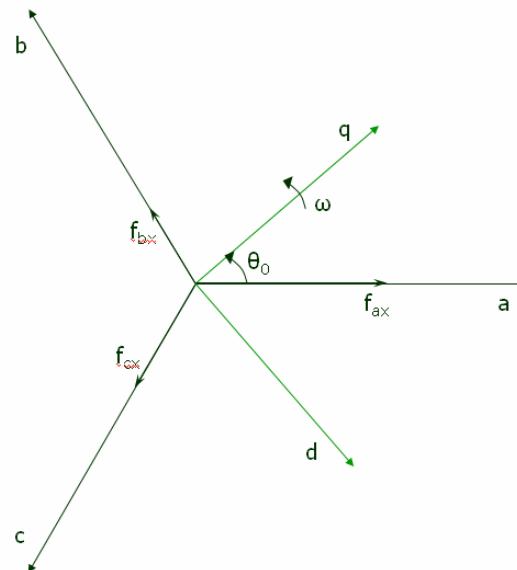
转子磁场定向控制 (FOC): 概述



- 数学理论应用于三相马达磁链及转矩的解耦控制
- 定子电流被分解成：
 - 直轴电流 I_{ds} ：用于产生磁场，与转子的磁场叠加
 - 交轴电流 I_{qs} ：用于控制转矩（其作用等同于直流电机的电枢电流）。
- 益处
 - 马达转速可对负载的变化做出精确而快速的反应
 - 效率优化
 - 可进行位置控制（通过瞬时转矩控制）

坐标变换理论 (1/3)

- 坐标变换理论可以降低马达方程的复杂性，利用坐标变换把定子及转子变量变换到一个旋转坐标系中，该坐标系的转速为角速度 ω 。
- 假设 f_{ax}, f_{bx}, f_{cx} 为三相瞬时变量，位于相移120度的a,b,c坐标上， f_{qx}, f_{dx}, f_{0x} 为其变换变量，位于正交坐标d,q上：



坐标变换理论(2/3)

■ 变换方程为：

$$f_{qdox} = \begin{pmatrix} f_{qx} \\ f_{dx} \\ f_{0x} \end{pmatrix} = \frac{2}{3} \begin{bmatrix} \cos \theta & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ \sin \theta & \sin\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{pmatrix} f_{ax} \\ f_{bx} \\ f_{cx} \end{pmatrix}$$

其中 $\theta = \int_0^t \omega(t) dt + \theta(0)$

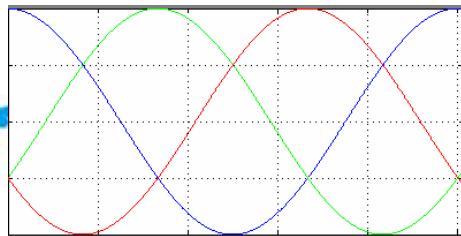
- 3个变换方程用于PMSM矢量控制
 - Clarke: $\omega=0, \theta(0)=0 \rightarrow \theta=0;$
 - Park: $\omega=\omega_r, \theta(0)=\theta_r(0) \rightarrow \theta=\theta_r;$
 - 反Park: $\omega=-\omega_r, \theta(0)=\theta_r(0) \rightarrow \theta=-\theta_r;$

坐标变换理论(3/3)



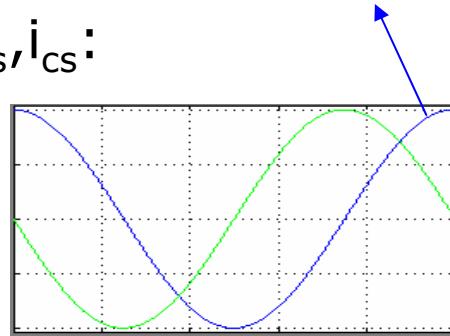
STM32

Clarke变换应用于定子电流： i_{as}, i_{bs}, i_{cs} :

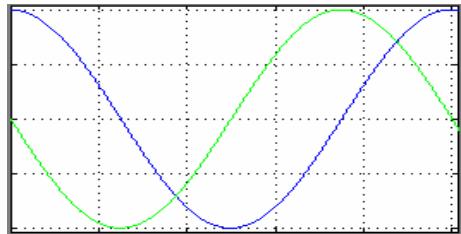


$$i_\alpha = i_{as}$$

$$i_\beta = -\frac{i_{as} + 2i_{bs}}{\sqrt{3}}$$

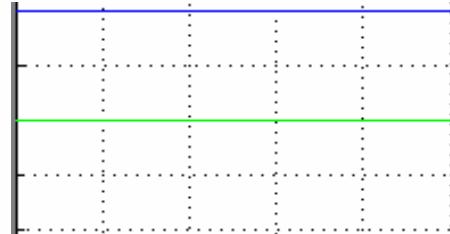


Park变换应用于定子电流： i_a, i_b :

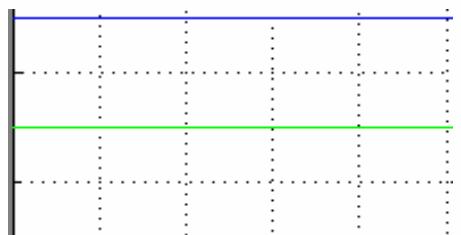


$$i_{qs} = i_\alpha \cos \theta_r - i_\beta \sin \theta_r$$

$$i_{ds} = i_\alpha \sin \theta_r + i_\beta \cos \theta_r$$

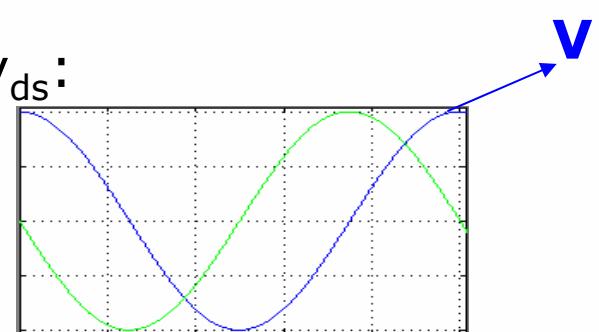


反Park变换应用于定子电压： v_{qs}, v_{ds} :



$$v_\alpha = v_{qs} \cos \theta_r + v_{ds} \sin \theta_r$$

$$v_\beta = -v_{qs} \sin \theta_r + v_{ds} \cos \theta_r$$

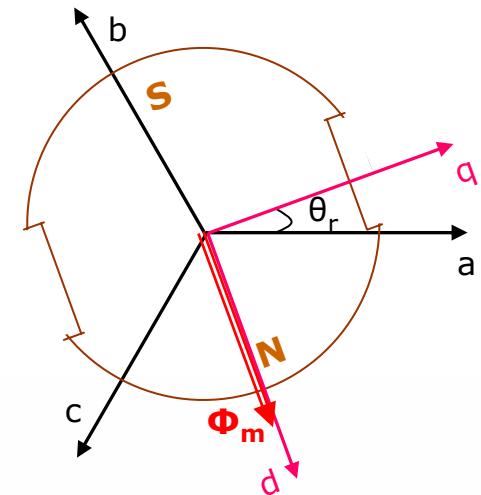


SM-PMSM FOC 1/3

参照右图，SM-PMSM马达的电压和磁链方程为：

$$v_{abc_s} = r_s i_{abc_s} + \frac{d\lambda_{abc_s}}{dt}$$

$$\lambda_{abc_s} = \begin{vmatrix} L_{ls} + L_{ms} & -\frac{1}{2}L_{ms} & -\frac{1}{2}L_{ms} \\ -\frac{1}{2}L_{ms} & L_{ls} + L_{ms} & -\frac{1}{2}L_{ms} \\ -\frac{1}{2}L_{ms} & -\frac{1}{2}L_{ms} & L_{ls} + L_{ms} \end{vmatrix} \begin{vmatrix} i_{abc_s} + \sin\left(\theta_r - \frac{2}{3}\pi\right)\Phi_m \\ \sin\left(\theta_r + \frac{2}{3}\pi\right) \end{vmatrix}$$



SM-PMSM FOC 2/3

把上述的马达方程变换到与转子同步的d,q坐标系中，且使d轴定位在转子磁链轴上，可得到：


$$\begin{cases} v_{q_s} = r_s i_{q_s} + \frac{d\lambda_{q_s}}{dt} + \omega_r \lambda_{d_s} \\ v_{d_s} = r_s i_{d_s} + \frac{d\lambda_{d_s}}{dt} - \omega_r \lambda_{q_s} \end{cases} \quad \begin{cases} \lambda_{q_s} = L_s i_{q_s} \\ \lambda_{d_s} = L_s i_{d_s} + \Phi_m \end{cases}$$

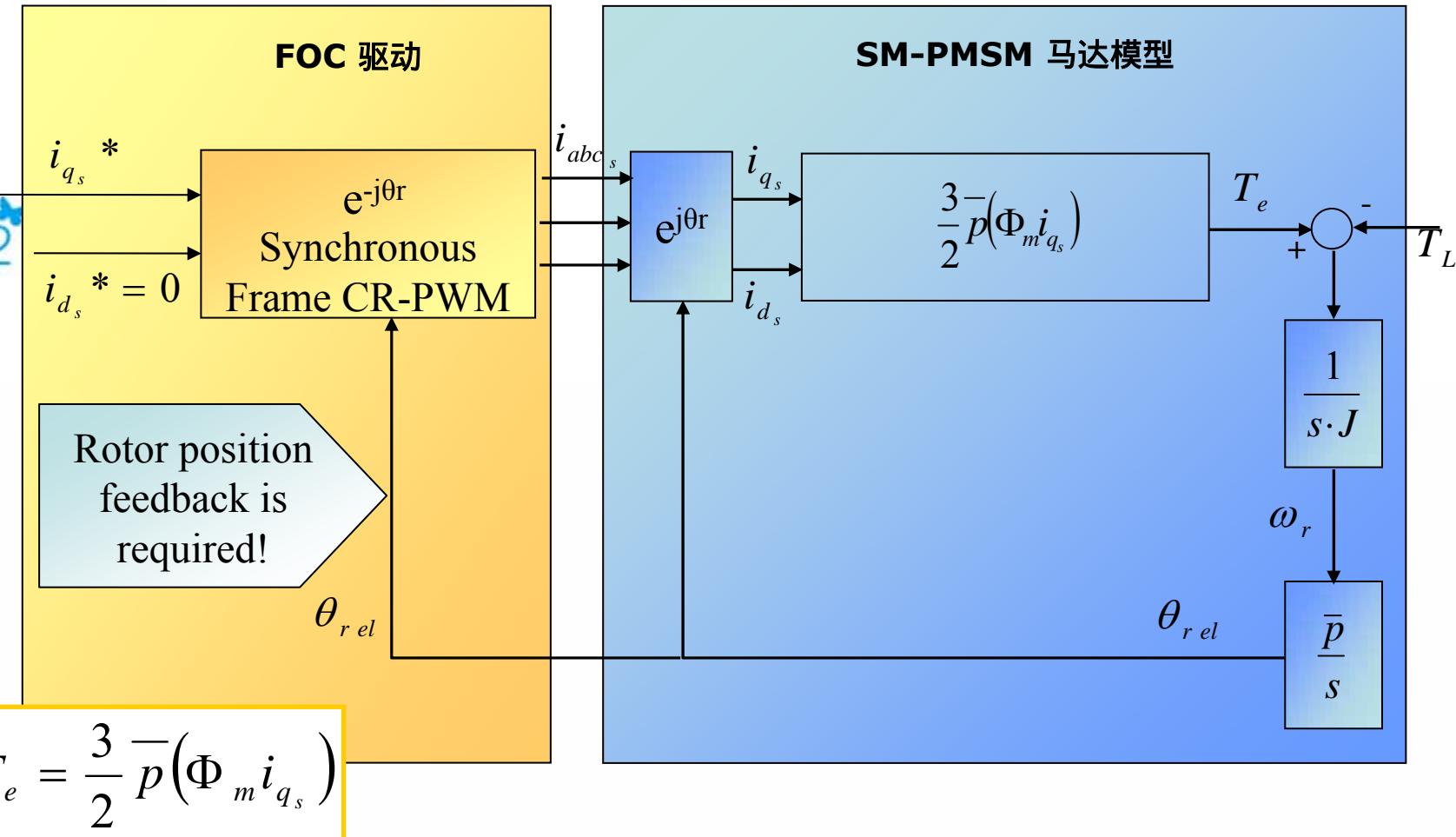
磁链

$$T_e = \frac{3}{2} \bar{p} (\lambda_{d_s} i_{q_s} - \lambda_{q_s} i_{d_s}) = \frac{3}{2} \bar{p} (L_s i_{d_s} i_{q_s} + \Phi_m i_{q_s} - L_s i_{q_s} i_{d_s})$$

当 $i_{qs}=I_s$; $i_{ds}=0$ 时，转矩达到最大值

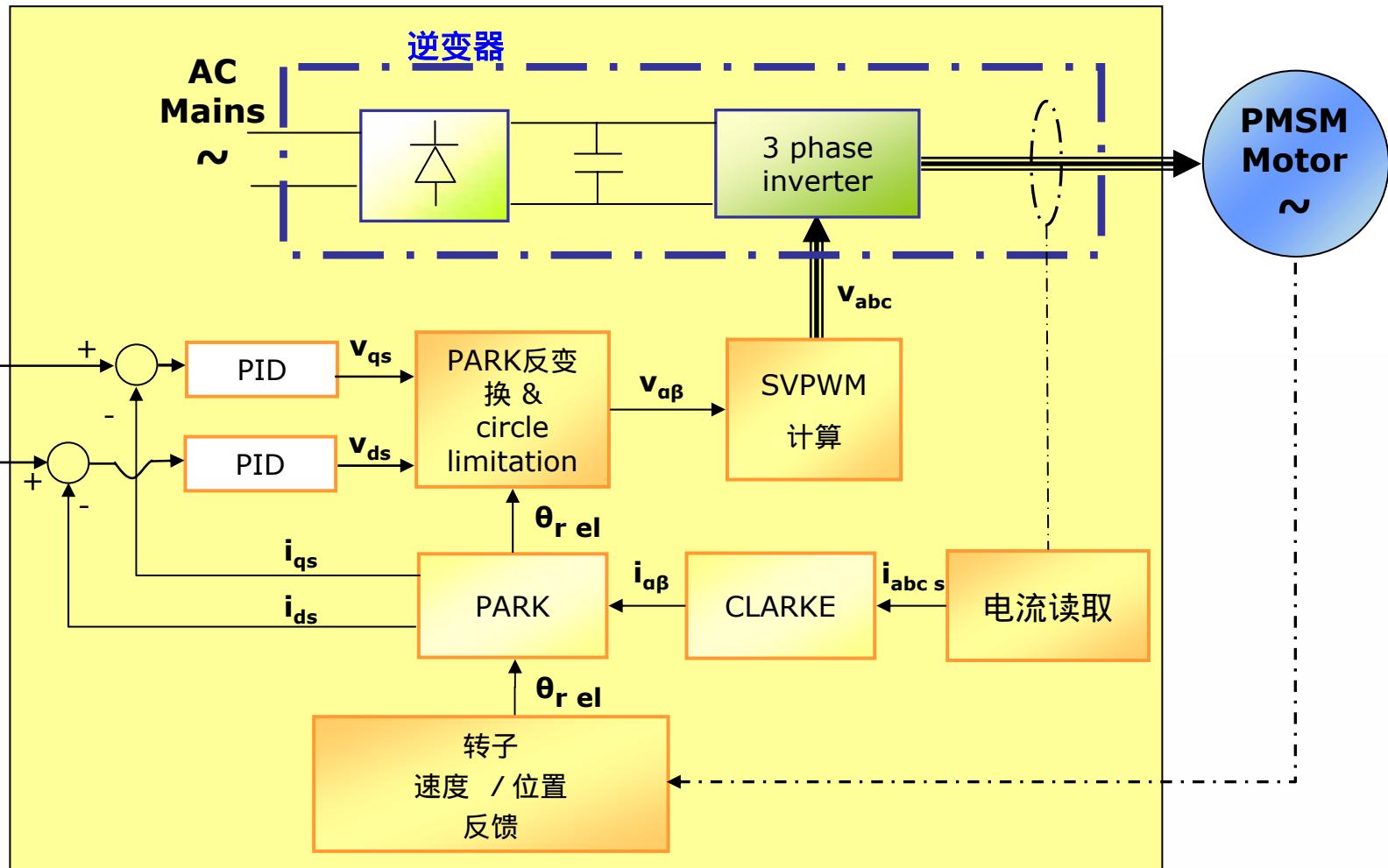
$$T_e = \frac{3}{2} \bar{p} (\Phi_m i_{q_s})$$

SM-PMSM FOC 3/3

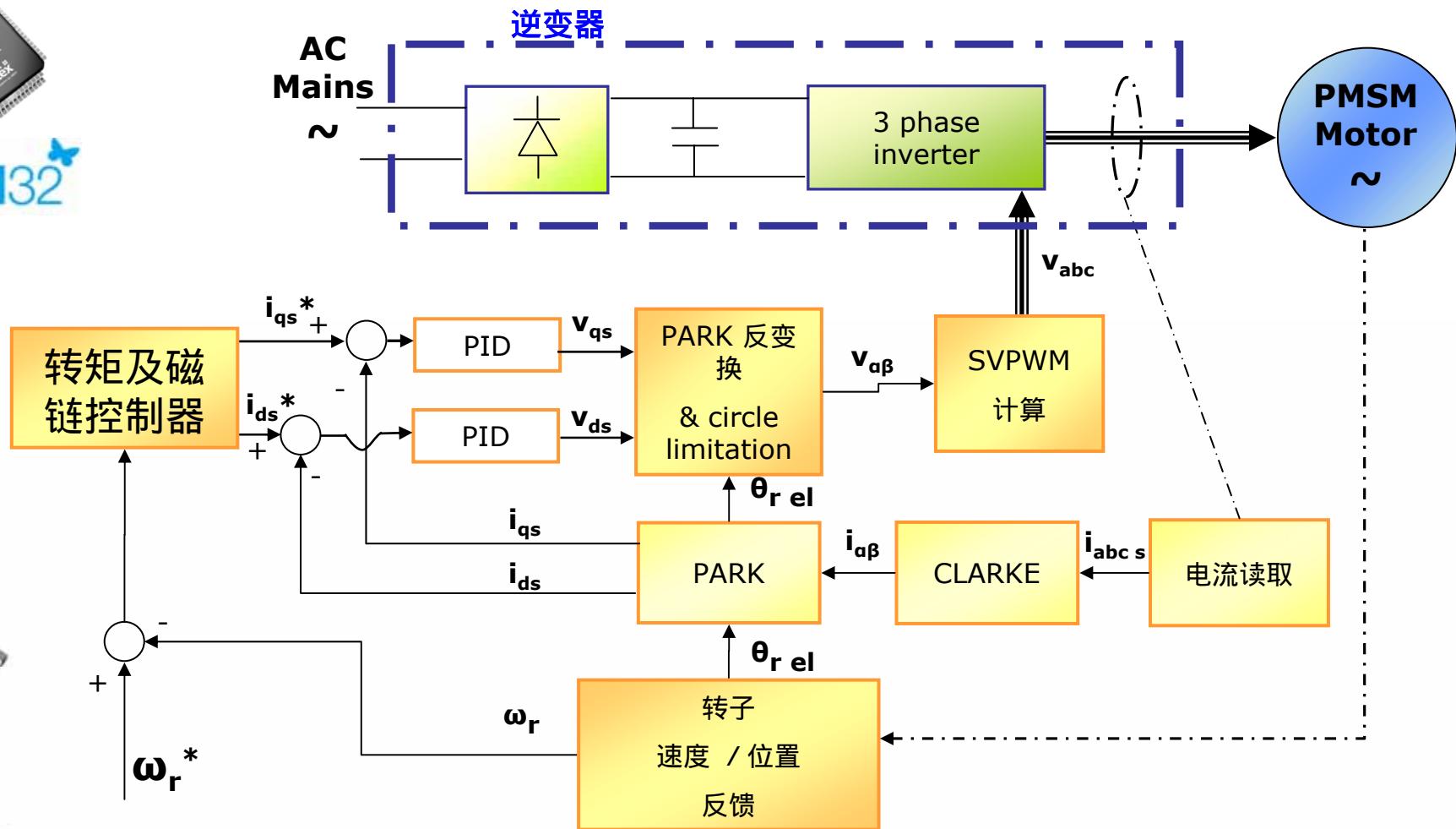


FOC 小结: 把定子电流分解为与转子磁链同相的直轴分量及正交的交轴分量

FOC 转矩控制

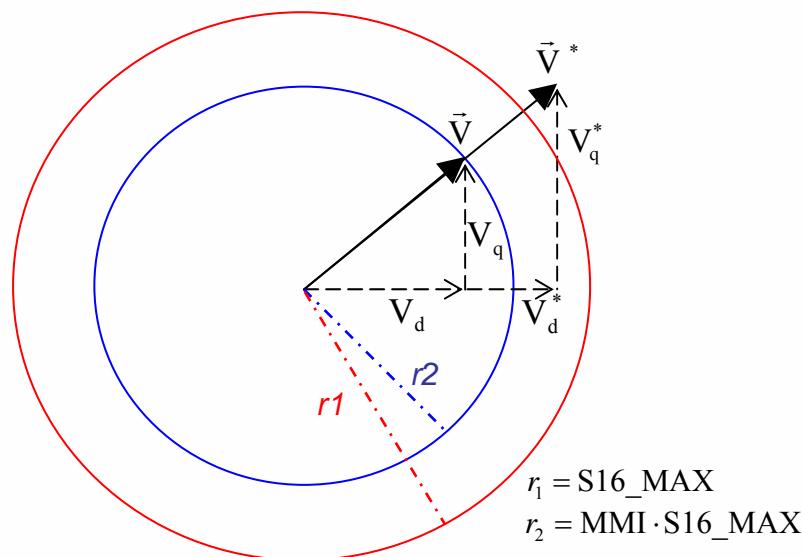


FOC 速度控制



深入： circle limitation

- 从前面的分析可知： v_{qs} 与 v_{ds} 是由2个PID调节器单独计算的，因此需要正确计算电压矢量 \vec{V} ，使之输入到SVPWM模块
- 最大的电压幅值为： $S16_MAX$ （即：32767），若使用3电阻法读取电流，它是PWM频率的函数(详见软件库中的MMI表)



$$V_q = \frac{V_q^* \cdot (\text{MMI} \cdot S16_MAX)}{|\vec{V}^*|}$$

$$V_d = \frac{V_d^* \cdot (\text{MMI} \cdot S16_MAX)}{|\vec{V}^*|}$$

$$\text{Look up table : } \frac{\text{MMI} \cdot S16_MAX^2}{|\vec{V}^*|}$$

深入：弱磁控制 1/3



STM32

- 很多应用需要使马达在额定转速以上运行：这可以通过弱磁控制的方法来实现。
- 额定转速是指马达能始终输出最高转矩的最高转速
- 可通过控制直轴电流 i_d 来降低磁通量，当然，代价是其同样降低了交轴电流 i_q ，最终输出转矩也降低。参见转矩方程：

$$T_e = \frac{3}{2} p (\Phi_m i_{q_s})$$

- 而且，如果需要在弱磁控制区域运行，逆变器必须有刹车电阻或再生发电能力(昂贵的4象限AC-DC,...)

深入：弱磁控制 2/3



STM32

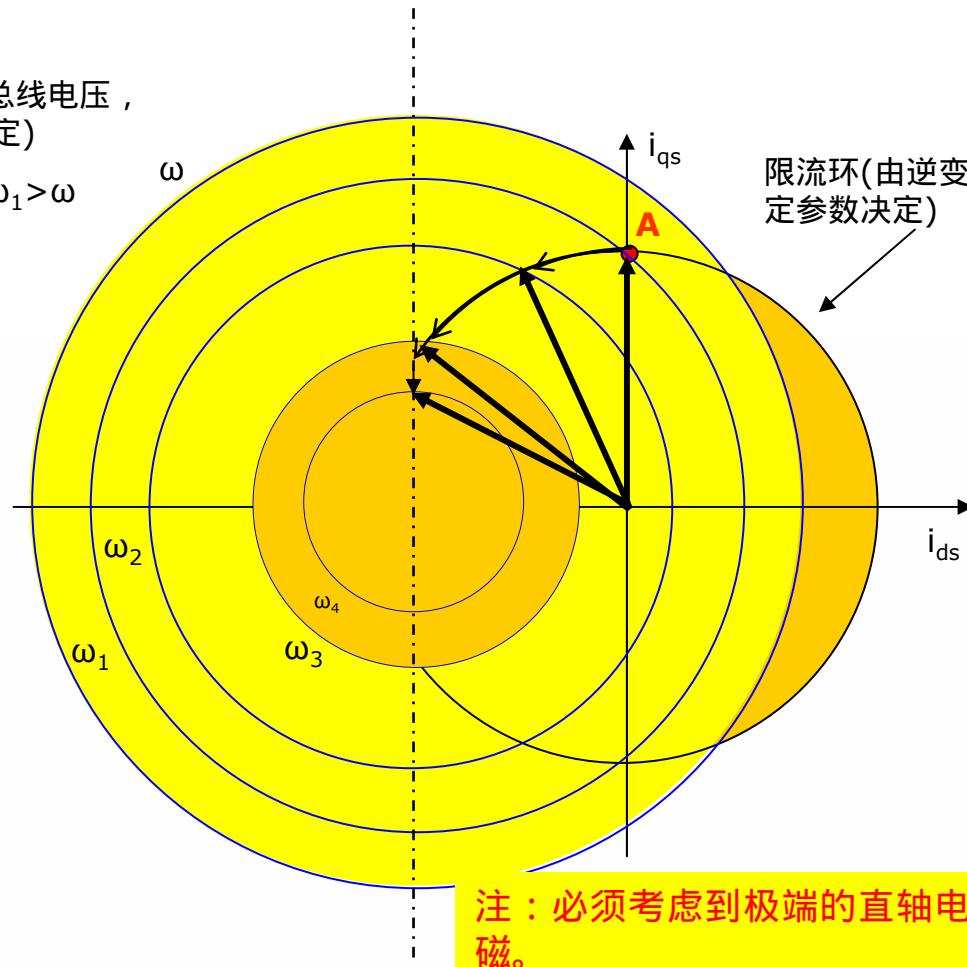
- “转矩及磁链控制器”(FOC速度控制) 实时地提供：目标交轴电流 i_q (通过一个 PID 调节器实现) 及目标直轴电流 i_d (通过查 MC_PMSM_Motor_param.h 中的数据表实现)，以此来逼近目标转速。
- 该头文件由电子表格产生，在该表格中，需要填写如下参数：
 - 逆变器交流输入电压；
 - R_s , 马达绕组电阻；
 - L_s , 马达绕组电感；
 - K_e , 马达反电动势常数；
 - p , 极对数；
 - I_n , 马达额定电流(< 逆变器额定电流)；
 - MMI (它是PWM频率的函数).

深入：弱磁控制 3/3



限压环 (由总线电压，
马达参数决定)
 $\omega_3 > \omega_2 > \omega_1 > \omega$

限流环(由逆变器及马达的额
定参数决定)

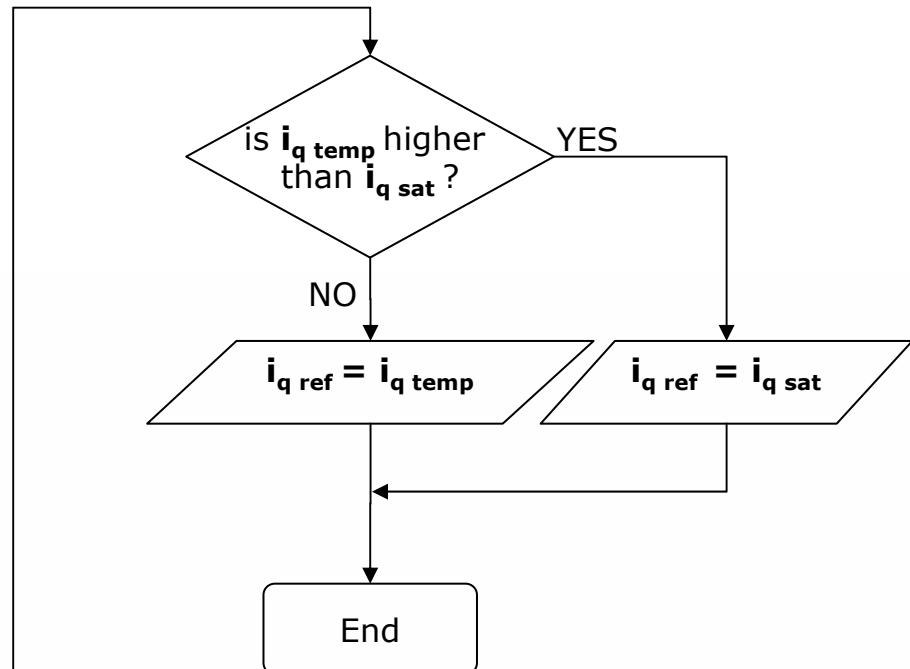
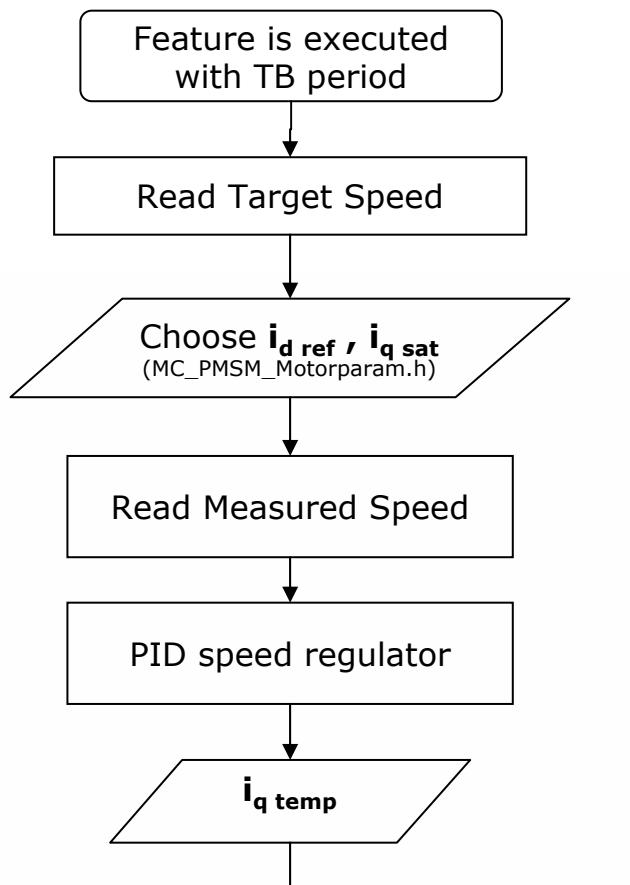


注：必须考虑到极端的直轴电流id会导致永久性退
磁。

深入：速度闭环调节

函数：

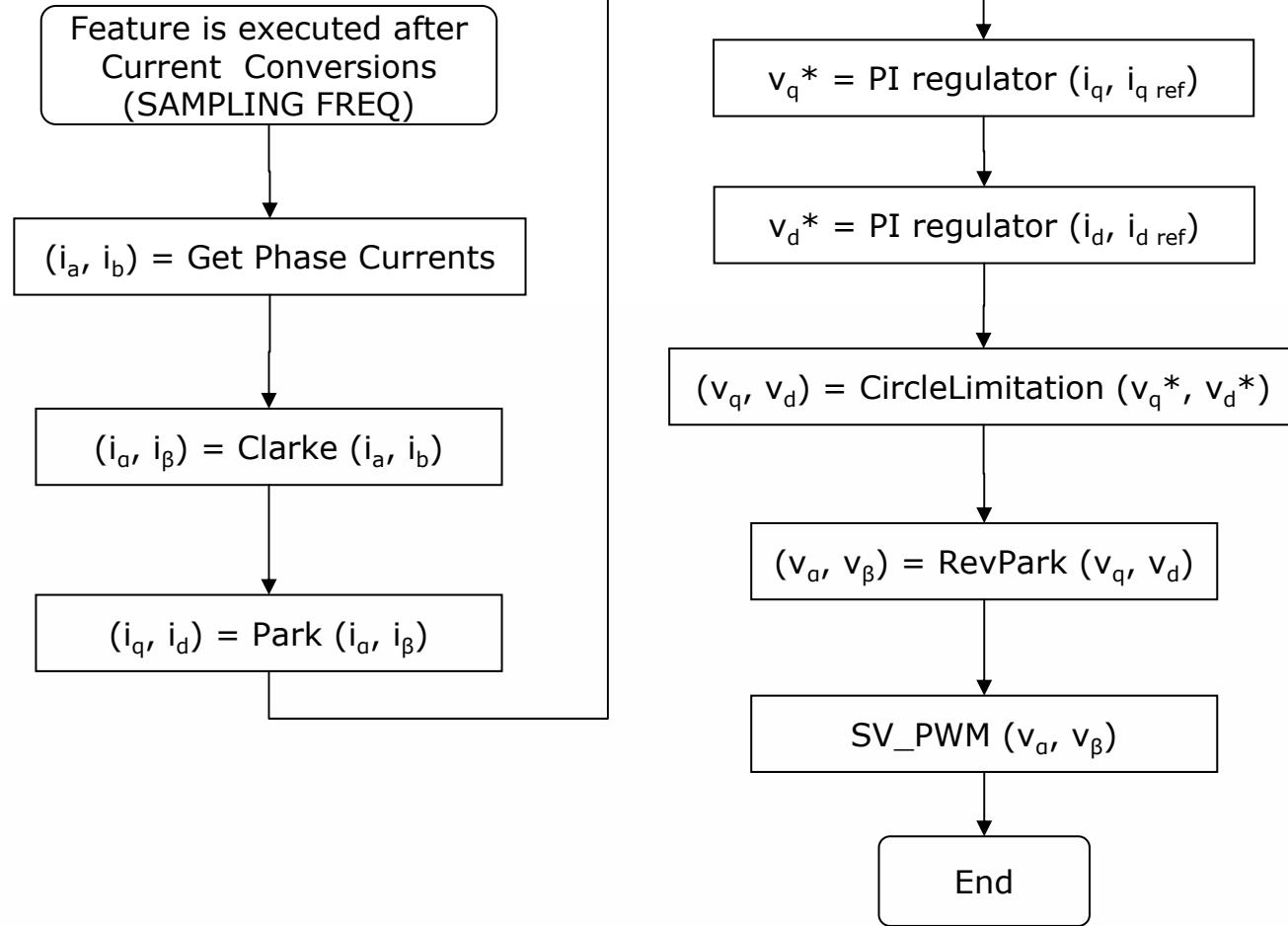
FOC_CalcFluxTorqueRef (MC_FOC_DRIVE.c)



FOC 实现

函数：

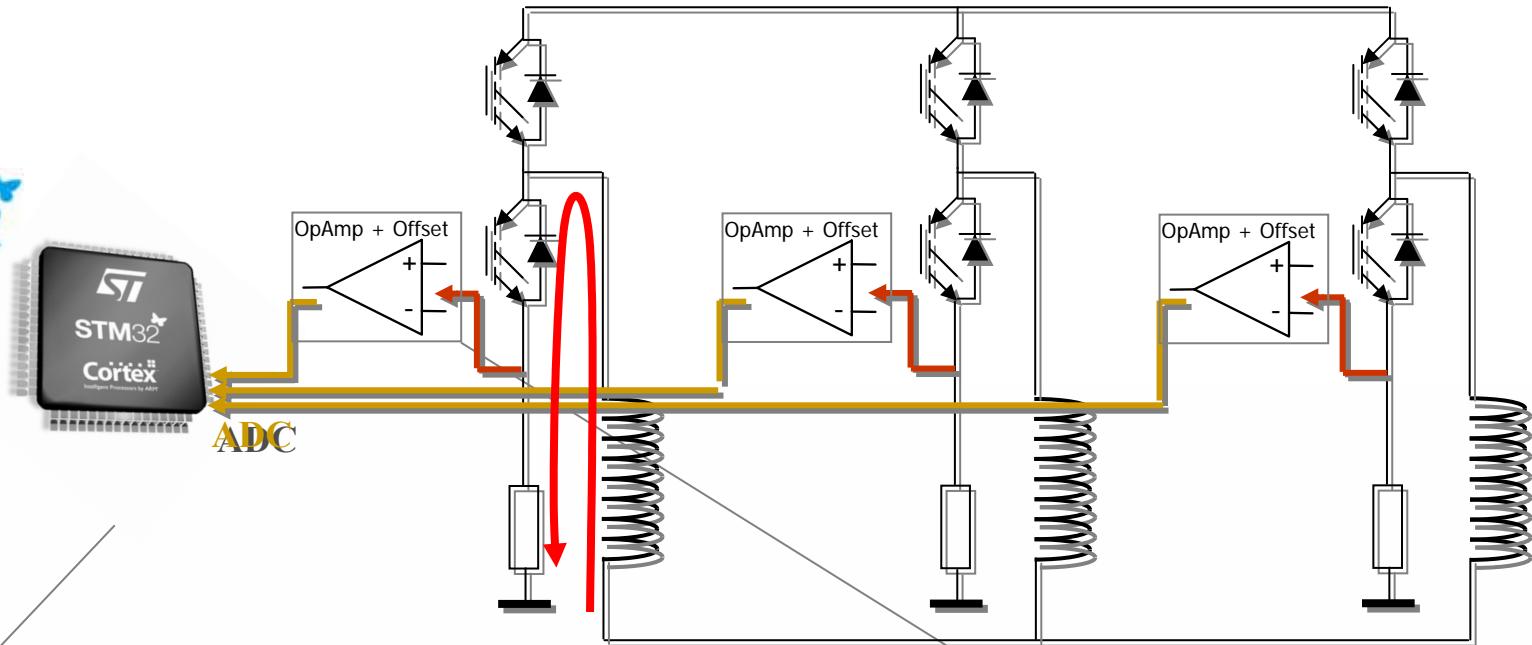
FOC_Model (MC_FOC_DRIVE.c)



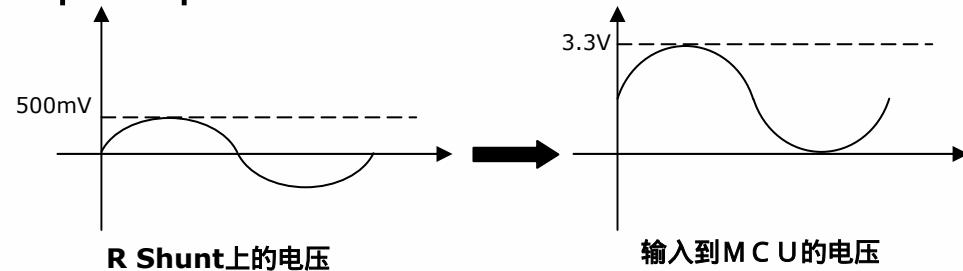
Plan

- ❑ 直流无刷马达介绍
- ❑ FOC 基础
- ❑ Clark 及 Parke 坐标系转换
- ❑ Circle limitation
- ❑ 磁链及力矩控制器
- ❑ 电流读取
 - ❑ 3 电阻法
 - ❑ 电流传感器法
- ❑ 转子的速度/位置反馈：
 - ❑ Hall 传感器
 - ❑ 正交编码器

电流读取硬件结构（3电阻法）

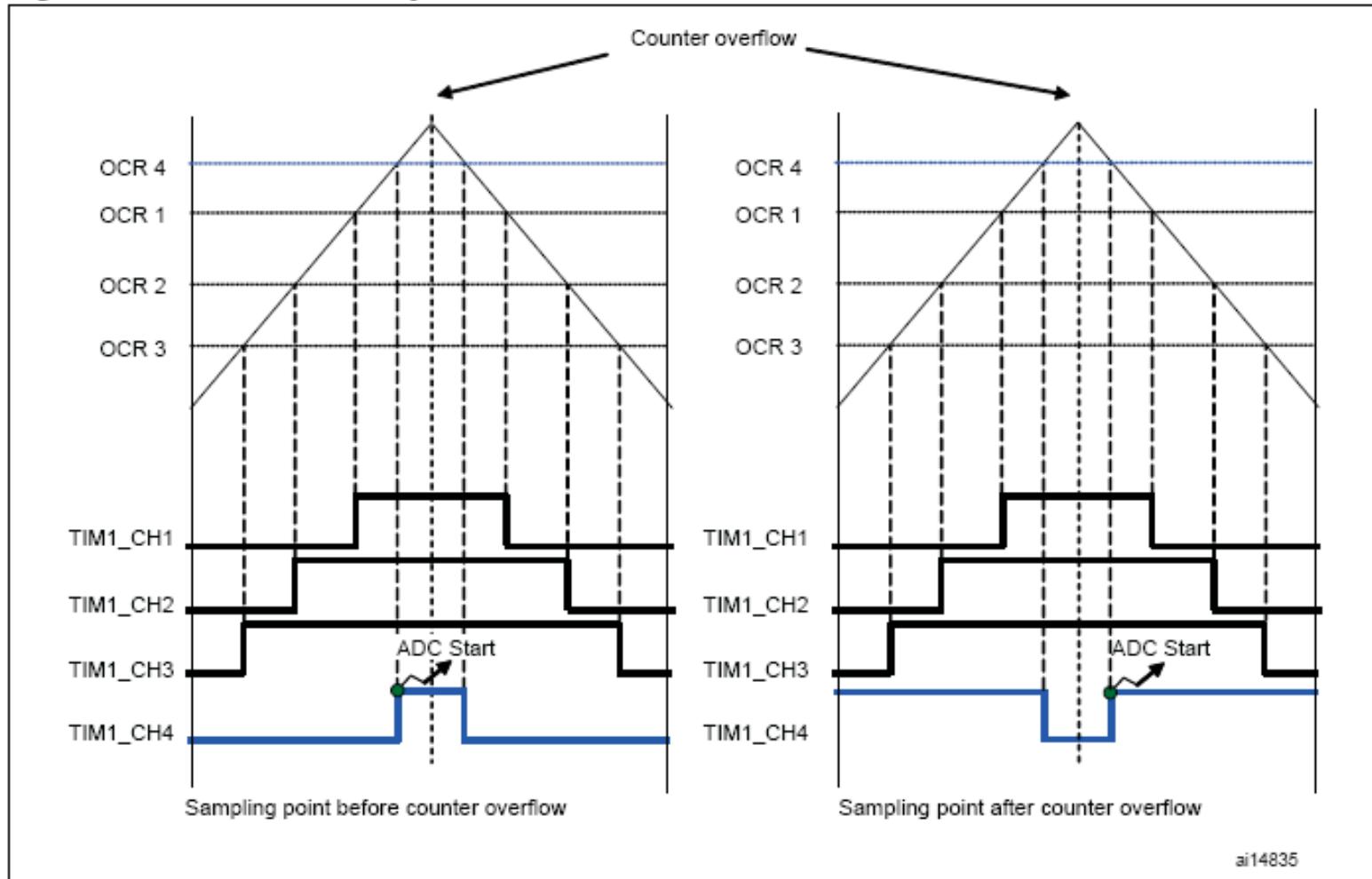


OpAmp + Offset



采样点 由TIM1的CH4输出触发采样

Figure 34. PWM and ADC synchronization

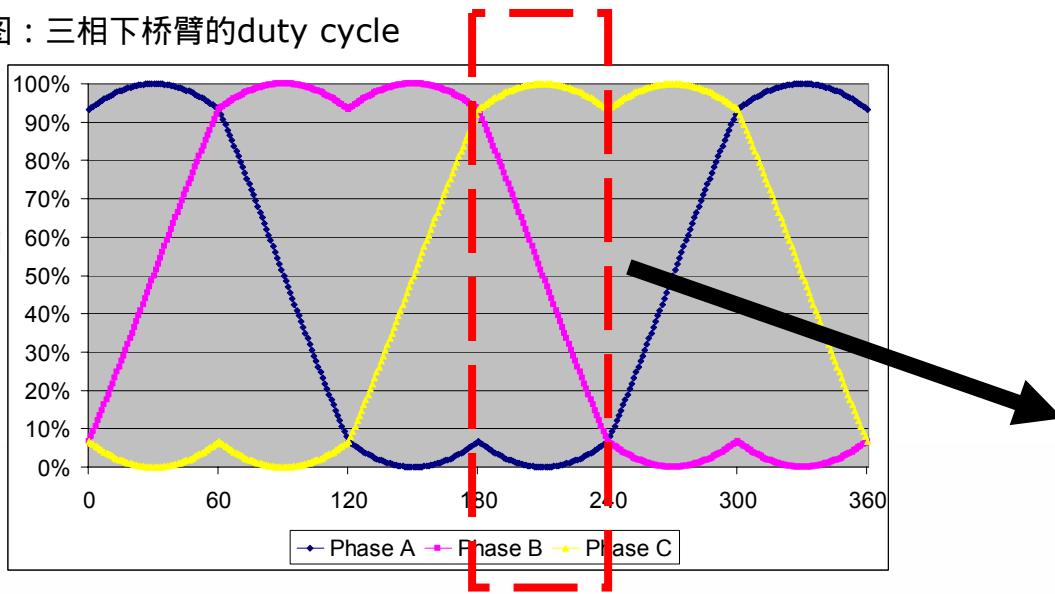


注意事项

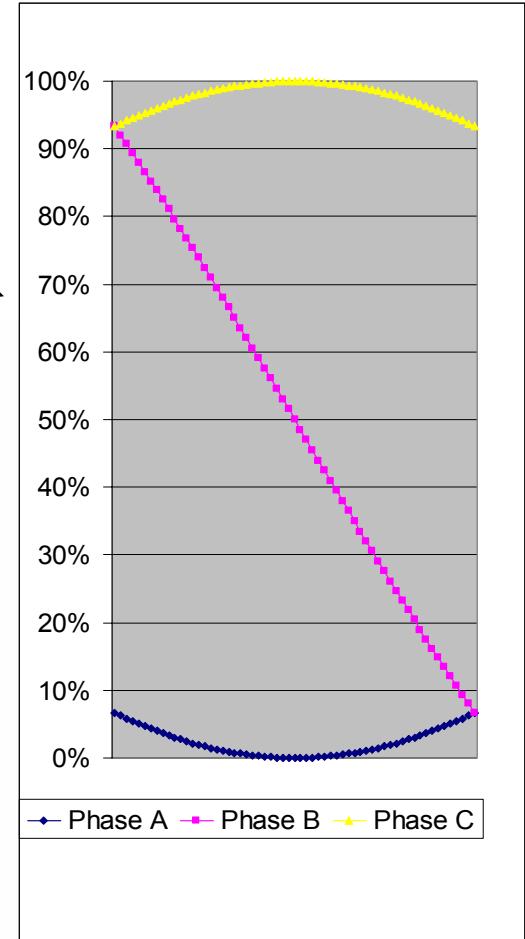
- 必须读取两相的定子电流：由SVPWM的当前扇区决定该读哪两相电流；
- 只有在下桥臂打开时，才能读到该相电流；
- 每次桥臂开关状态有变化时，会在shunt电阻上的电压产生一个电子干扰，假设该干扰的时间长度为 T_{Noise} ；
- 当下桥臂打开后，需要等待一段时间来使shunt电阻上的电压达到稳定值，假设该干扰的时间长度为 T_{Rise} ；
- 在 T_{Noise} 及 T_{Rise} 期间不能读相电流；
- 由于STM32 ADC/TIM1的高性能，我们可以在PWM周期的任意时刻读取电流。当然，我们希望在SVPWM的不同扇区，电流采样点无大的变化。

与SVPWM相关的问题

图：三相下桥臂的duty cycle



- 对于每个SVPWM扇区，总有 2 相桥臂的duty cycle 变化很小而另 1 相的duty cycle变化很大；
- 在上述的介绍中，我们假设C相下桥臂的duty cycle 最大，A相下桥臂的duty cycle最小，而B相下桥臂的duty cycle变化最大，如右图。

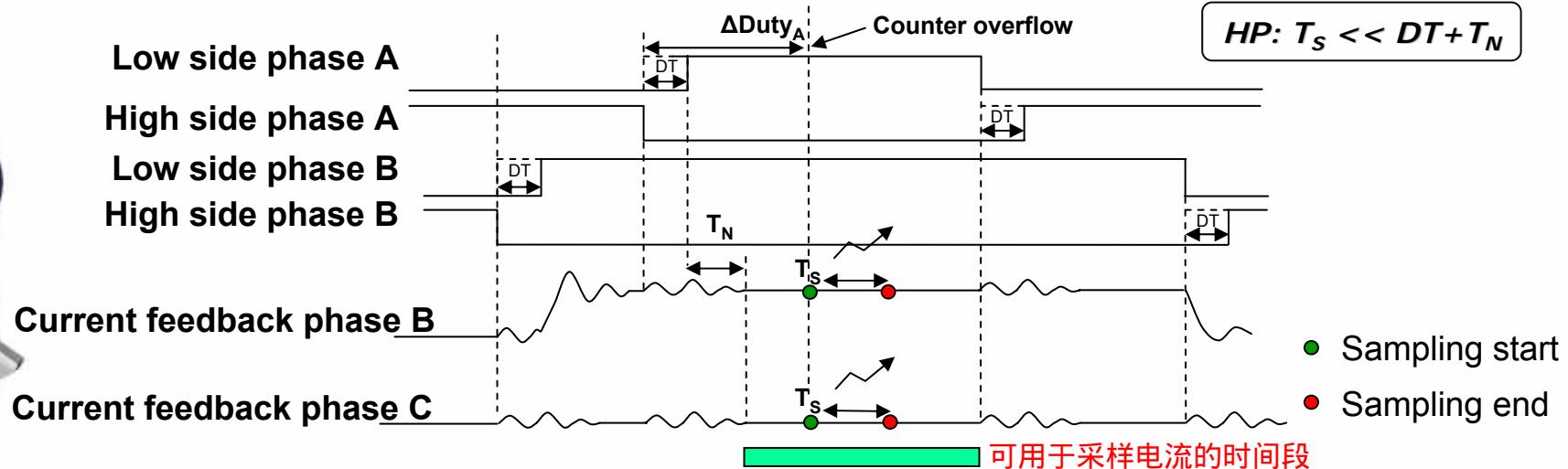


采样点的定义 (1/4) : 低调制系数



若 $\Delta\text{Duty}_A > D_T + T_N$ ：可以把采样点设在定时器的上溢处。

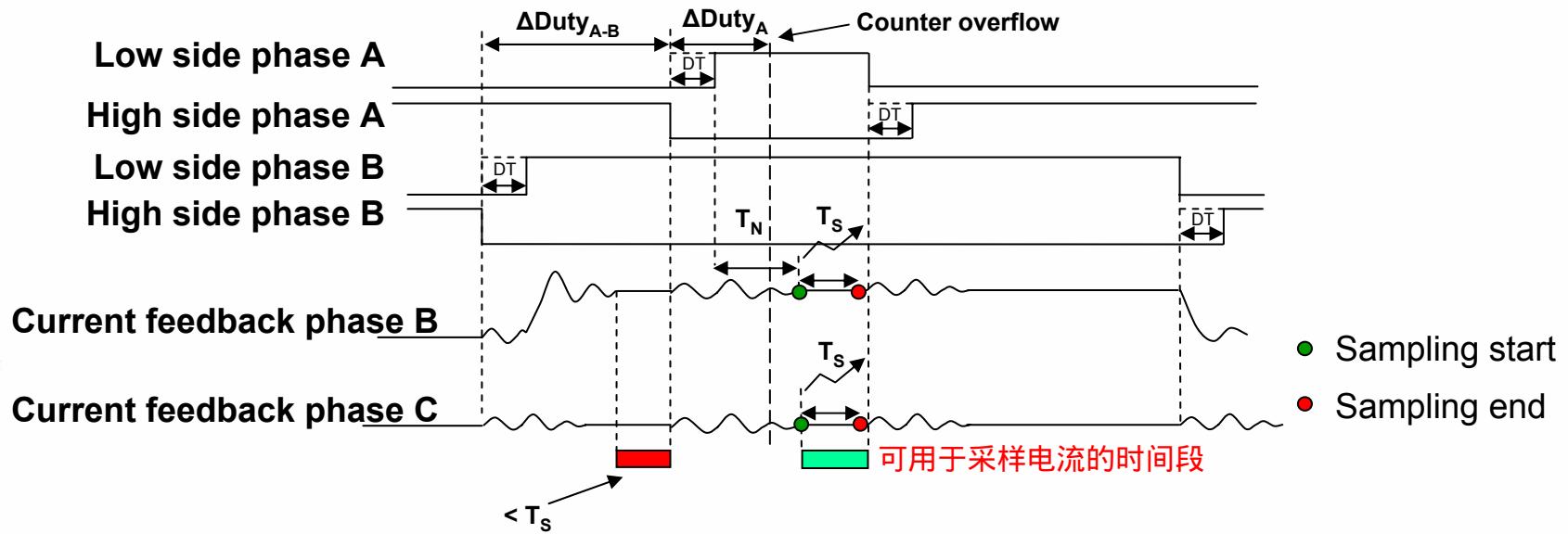
随着调制系数的增大，只要保证 $\Delta\text{Duty}_A > DT + T_N$ ，则始终可以把采样点设在定时器上溢处。



采样点的定义 (2/4) : 高调制系数

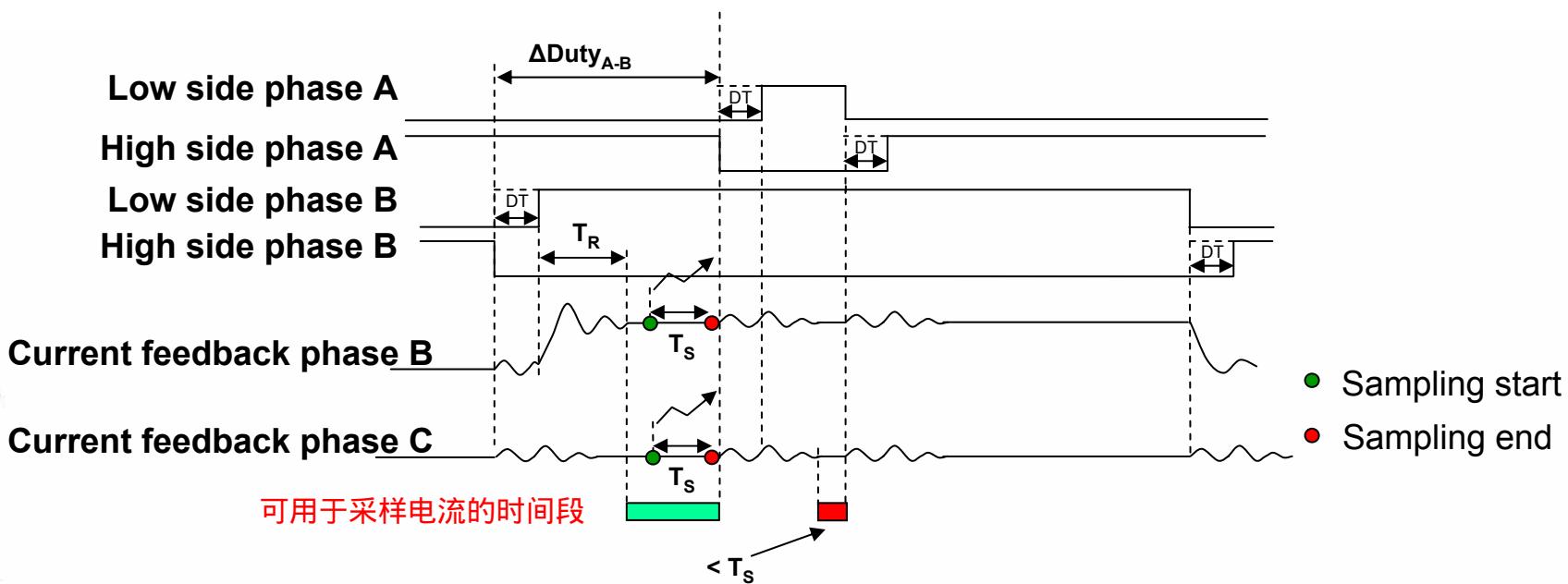


- 若 $(DT + T_N + T_S)/2 < \Delta Duty_A < DT + T_N \& \Delta Duty_{A-B} < DT + T_R + T_S$:
可以把采样点设在A相下桥臂打开后 $T_N \mu s$ 处



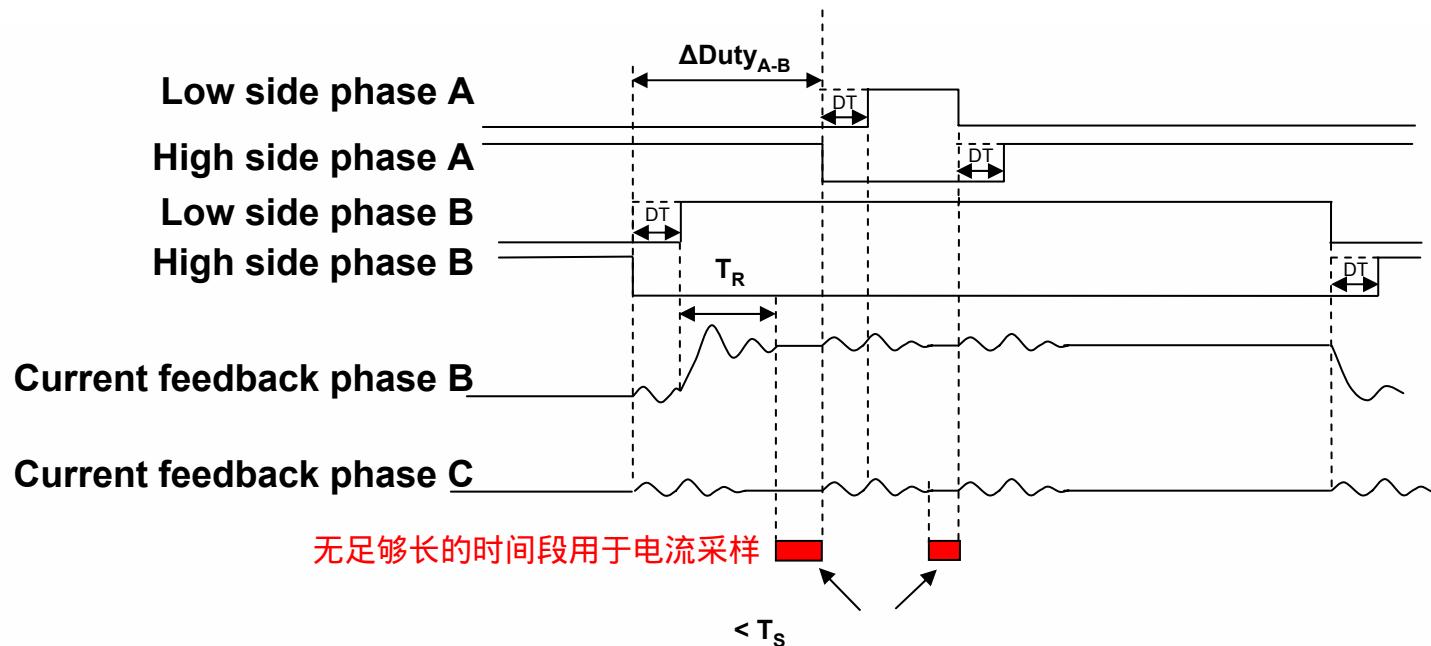
采样点的定义 (3/4) : 高调制系数

- 若 $\Delta\text{Duty}_A < (\text{DT} + \text{T}_N + \text{T}_S)/2$ & $\Delta\text{Duty}_{A-B} > \text{DT} + \text{T}_R + \text{T}_S$: 可以把采样点设在A相上桥臂关断前 $\text{T}_S \mu\text{s}$ 处。



采样点的定义 (4/4) : 高调制系数

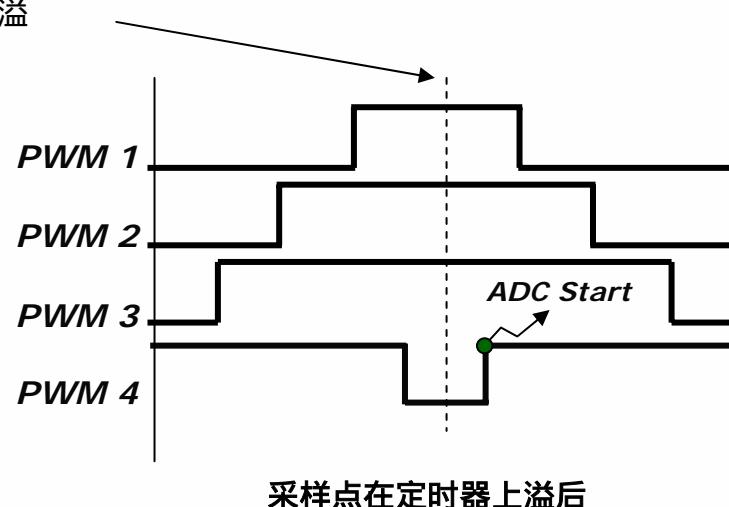
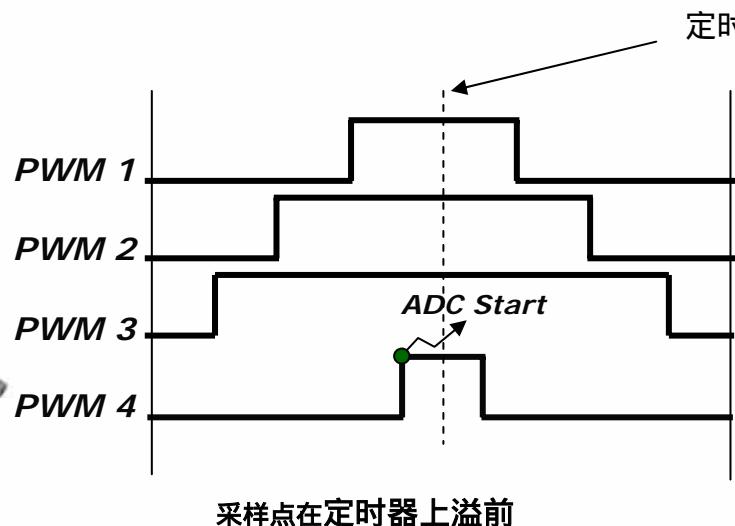
- 若 $\Delta\text{Duty}_A < (\text{DT} + \text{T}_N + \text{T}_S)/2$ & $\Delta\text{Duty}_{A-B} < \text{DT} + \text{T}_R + \text{T}_S$: 则无法采样相电流。为了避免此情况，必须降低调制系数 (MMI) 或 PWM 频率。



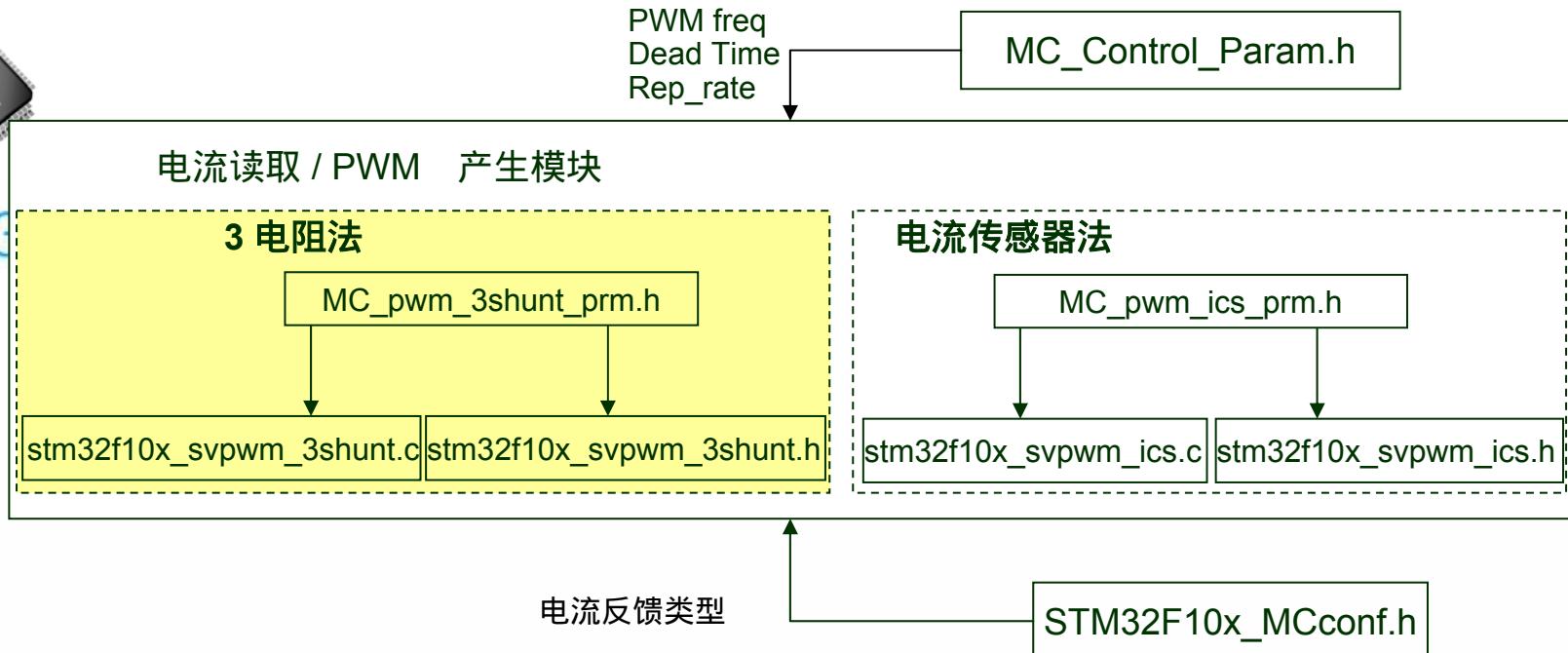
采样点的实现



- TIM1的PWM1, 2,3 用于驱动逆变器；
- 为了设置采样点，需要用到ADC外设的外部触发能力；
- 相电流的ADC转换由PMW4上升沿触发；
- 为了在定时器上溢后采样相电流，要求反转PWM4信号的极性



软件结构



- **STM32F10x_MCconf.h** : 定义用于选定的电流反馈类型；
- **MC_Control_Param.h** : 定义**PWM** 频率 (及MMI- 最大调制系数), 死区时间(ns) 及寄存器更新率**REP_RATE**；
- **MC_pwm_3shunt_prm.h** : 定义采样时间**T_s**, **T_{Noise}**及 **T_{Rise}** (ns)。

定制 1/2



STM32

STM32F10x_MCconf.h

```
/* Current sensing by ICS (Isolated current sensors) */
#ifndef ICS_SENSORS
/* Current sensing by Three Shunt resistors */
#define THREE_SHUNT
```

MC_Control_Param.h

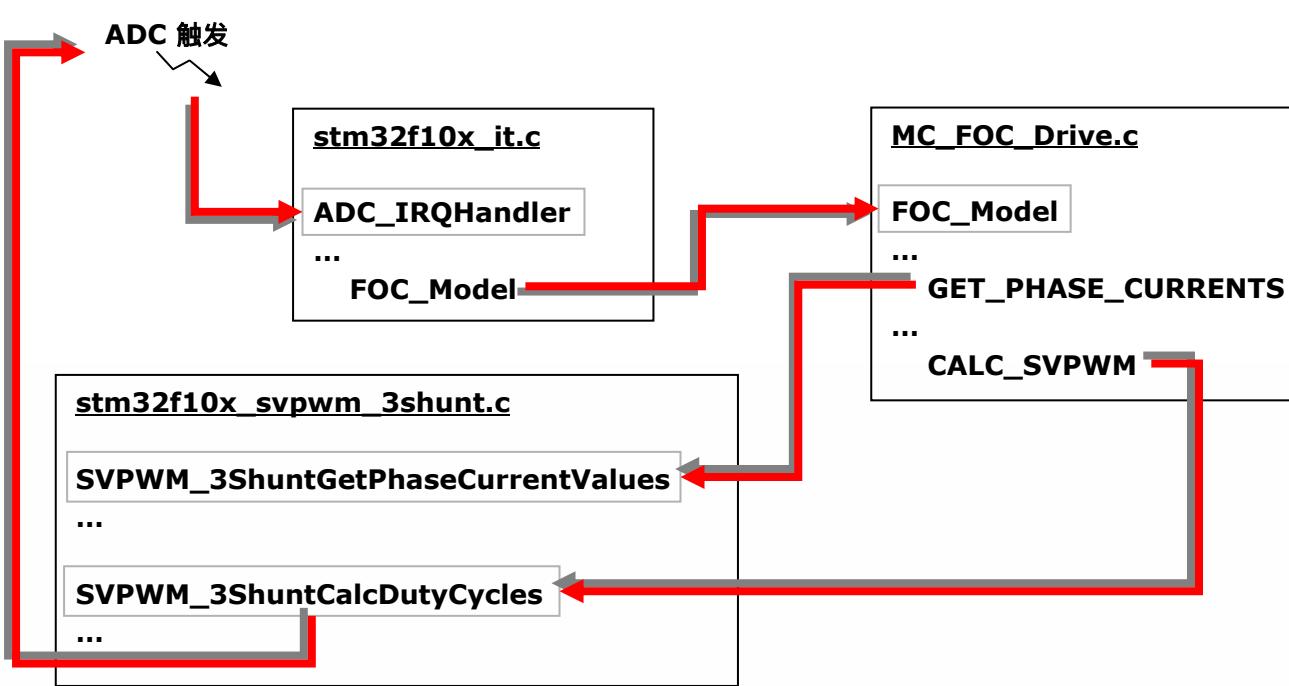
```
***** Power devices switching frequency *****
#define PWM_FREQ ((u16) 12500) // in Hz
***** Deadtime Value *****
#define DEADTIME_NS ((u16) 800) //in nsec; range is [0...3500]
***** Uncomment the Max modulation index *****
***** corresponding to the selected PWM frequency *****
//#define MAX_MODULATION_100_PER_CENT      // up to 11.4 kHz PWM frequency
//#define MAX_MODULATION_99_PER_CENT       // up to 11.8 kHz
//#define MAX_MODULATION_98_PER_CENT       // up to 12.2 kHz
#define MAX_MODULATION_97_PER_CENT        // up to 12.9 kHz
...
***** ADC IRQ-HANDLER frequency, related to PWM *****
#define REP_RATE (1)
```

定制 2/2

MC_pwm_3shunt_prm.h

```
#define SAMPLING_TIME_NS (u16)(700) //0.7usec  
#define TNOISE_NS (u16)(2550) //2.55usec  
#define TRISE_NS (u16)(2550) //2.55usec
```

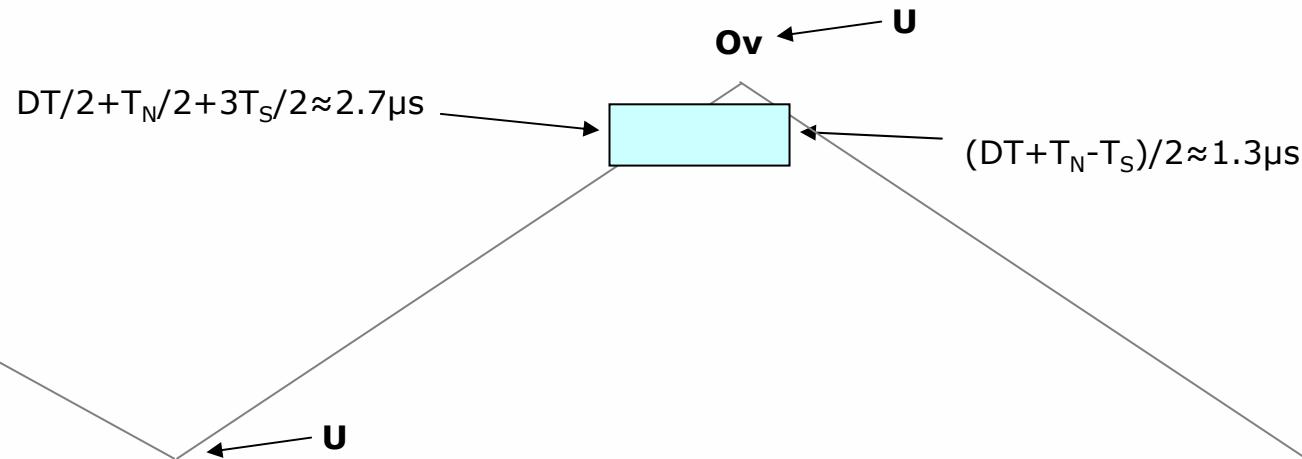
3 电阻法程序流程



触发ADC，并随之调用ADC_IRQ。

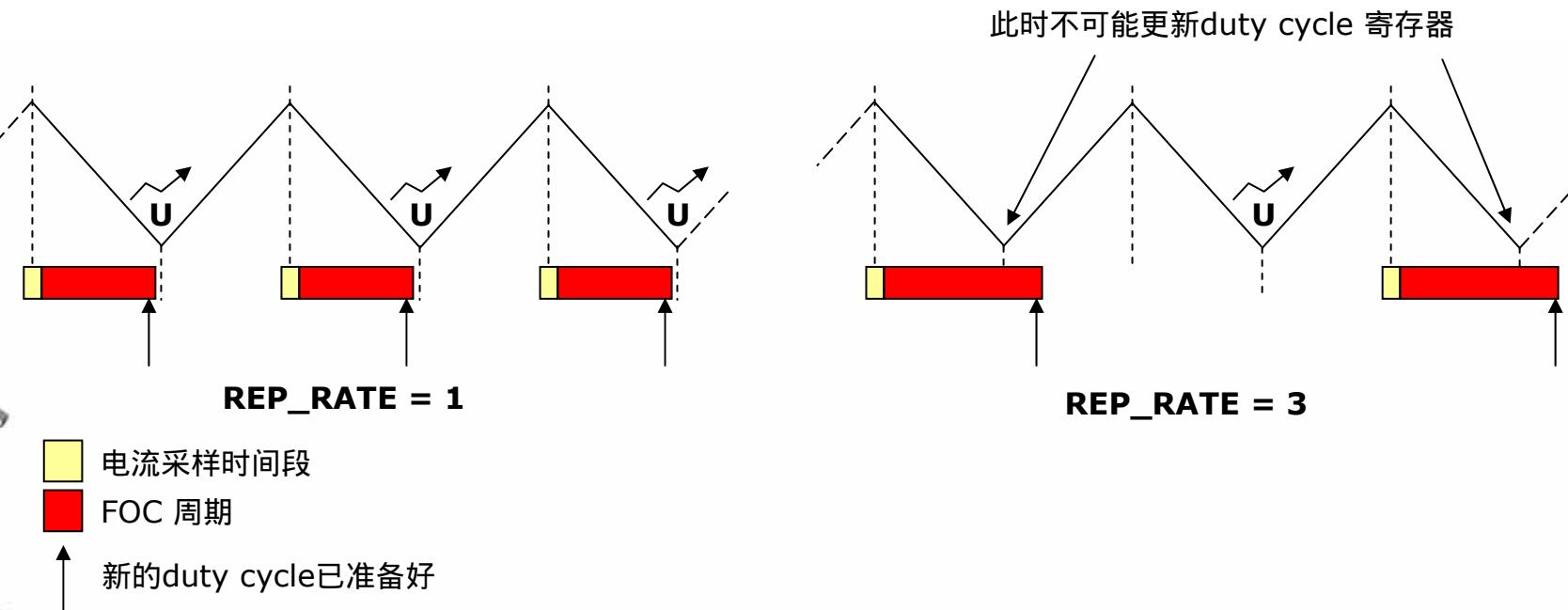
U(Update) 事件

- 从上述可知：ADC转换在定时器上溢的附近被触发；
- ADC转换可在定时器上溢前 $DT/2+T_N/2 + 3T_S/2$ 到定时器上溢后 $(DT+T_N-T_S)/2$ 的时间段触发；
- 如果在定时器上溢处产生U事件，不可能在定时器上溢前触发ADC；
- 因此，必须在定时器下溢处产生U事件。



REP_RATE 定义

- 如果PWM频率太高，则有可能在下个U事件到来之前，FOC所计算的新的duty cycle得不到更新。因此，这种情况下必须增大REP_RATE。



性能

PWM frequency	Max Duty Cycle	Max Modulation Index	REP_RATE
Up to 11.4kHz	100%	100%	1
12.2kHz	99%	98%	1
12.9kHz	98.5%	97%	1
13.7kHz	98%	96%	1
14.4kHz	98%	96%	1
15.2kHz	97%	94%	1
16kHz	96.5%	93%	1
16.7kHz	96.5%	93%	3
17.5kHz	95.5%	91%	3

这些参数来自MB459

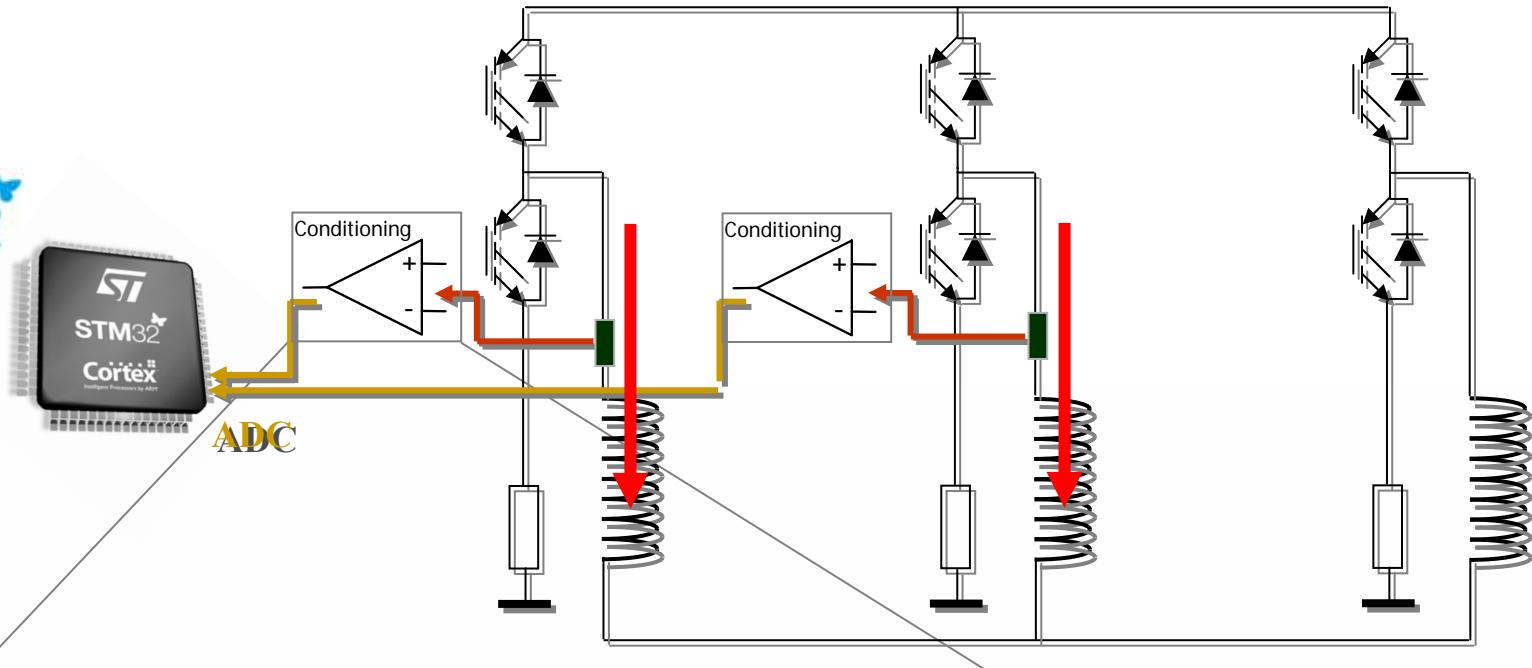
$T_N = T_R = 2.55 \mu s$

$T_S = 0.7 \mu s$

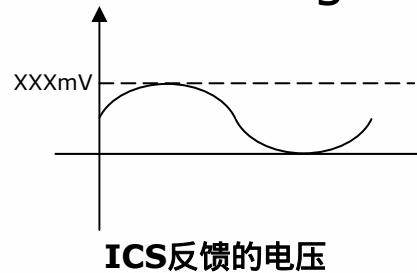
$DT = 0.8 \mu s$

此数据为测试所得

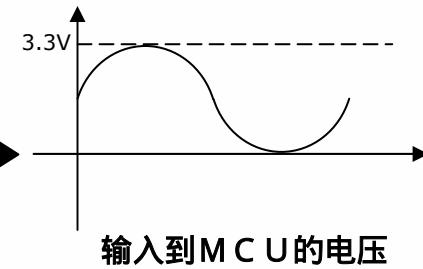
电流读取硬件结构(电流传感器法)



Conditioning Network



ICS反馈的电压

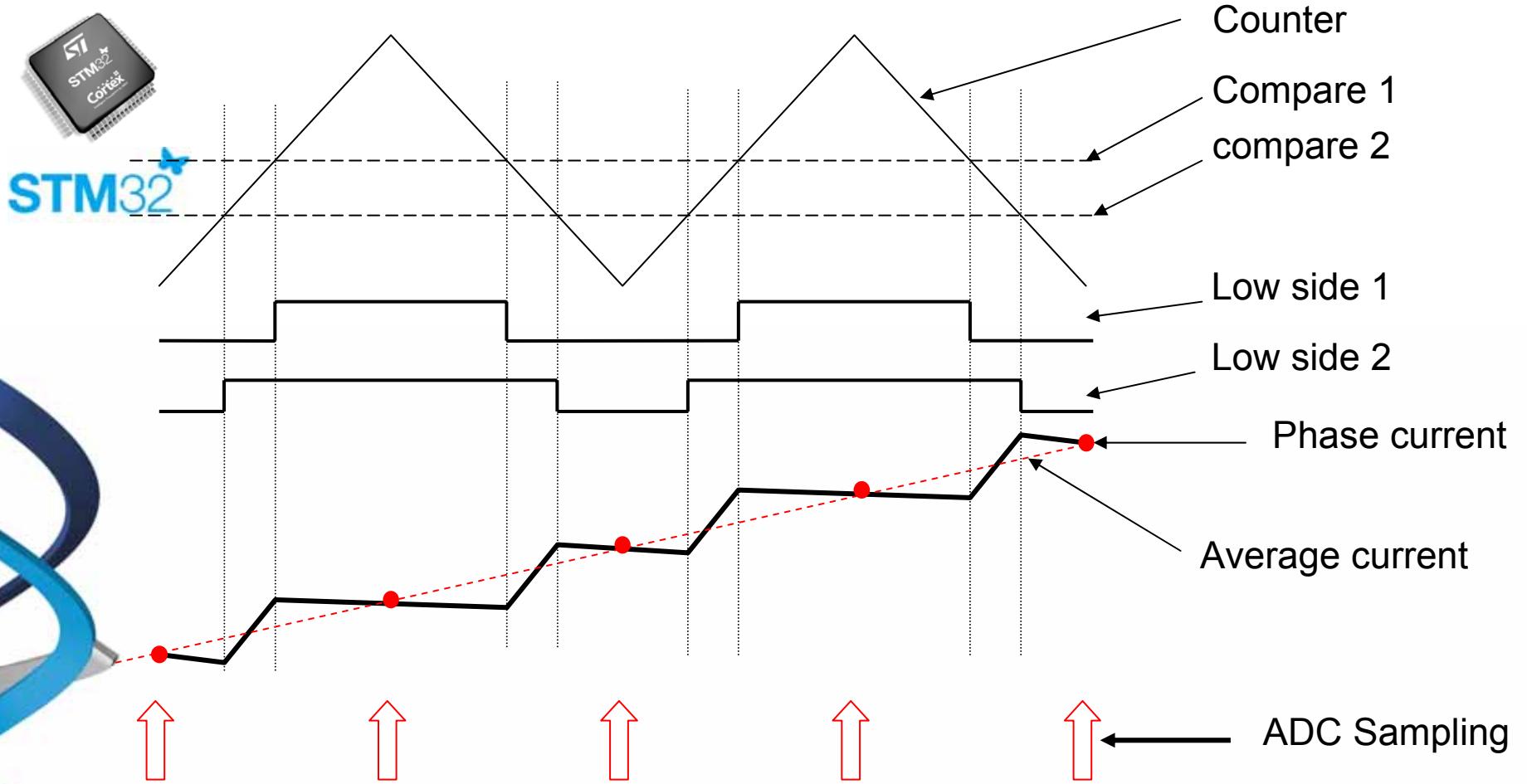


输入到MCU的电压

此硬件网络应根据传感器的特性设计，一般来说，不需要加offset。

必须保证：电流流出逆变器时，ICS产生正的电压

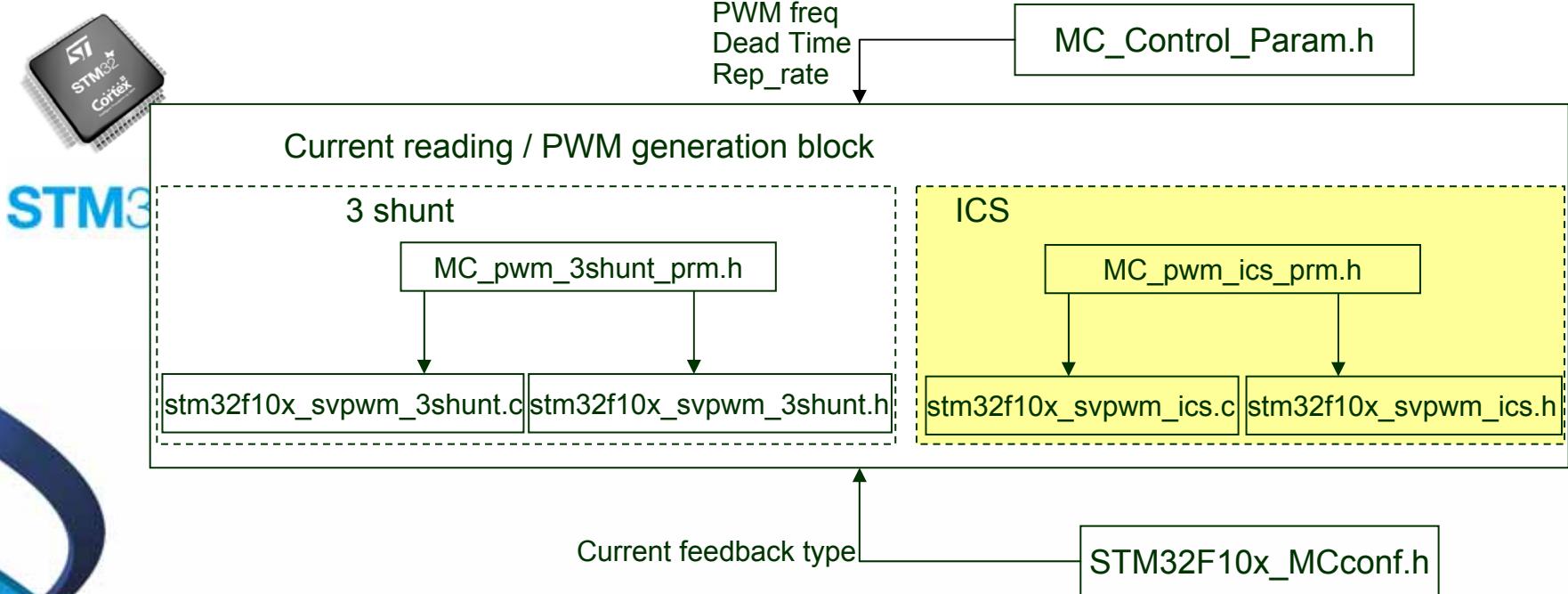
采样点



注意事项

- 只需读取固定的两相定子电流(如：A 相和 B相)；
- 不需要在下桥臂打开时读该相电流；
- 在逆变器开关状态变化时，不需要等待Tnoise或Trise后才采样电流；
- 因此，始终可以用U事件来触发ADC，且duty cycle一直可以达到100%。

软件结构



- **`STM32F10x_MCconf.h`** : 定义用于选定的电流反馈类型；
- **`MC_Control_Param.h`** : 定义**PWM** 频率 (及MMI- 最大调制系数),
死区时间(ns) 及寄存器更新率**REP_RATE**；

定制



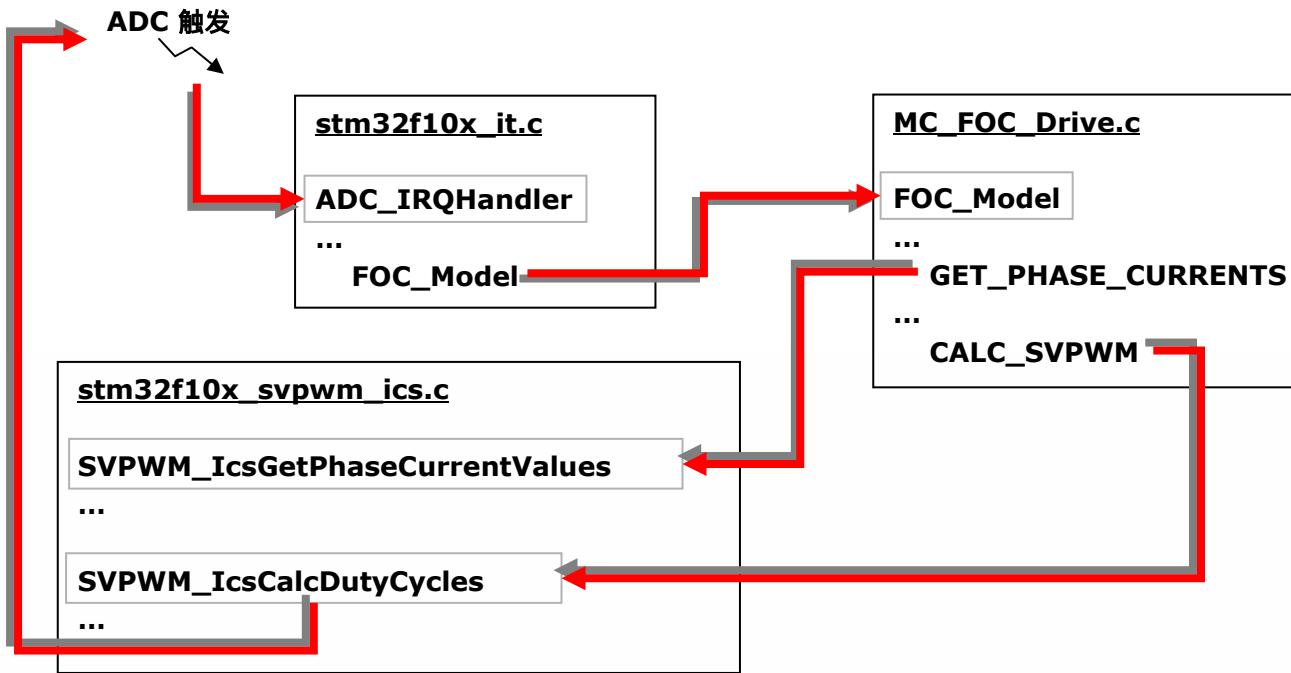
STM32F10x_MCconf.h

```
/* Current sensing by ICS (Isolated current sensors) */
#define ICS_SENSORS
/* Current sensing by Three Shunt resistors */
//#define THREE_SHUNT
```

MC_Control_Param.h

```
***** Power devices switching frequency *****
#define PWM_FREQ ((u16) 14400) // in Hz
***** Deadtime Value *****
#define DEADTIME_NS ((u16) 800) //in nsec; range is [0...3500]
***** Uncomment the Max modulation index *****
***** corresponding to the selected PWM frequency *****
#define MAX_MODULATION_100_PER_CENT // up to 11.4 kHz PWM frequency
//#define MAX_MODULATION_99_PER_CENT // up to 11.8 kHz
//#define MAX_MODULATION_98_PER_CENT // up to 12.2 kHz
//#define MAX_MODULATION_97_PER_CENT // up to 12.9 kHz
...
***** ADC IRQ-HANDLER frequency, related to PWM *****
#define REP_RATE (3)
```

ICS 电流读取程序流程



在 FOC 模块之后，调用宏 **CALC_SVPWM(SVPWM_IcsCalcDutyCycles)**，其功能为：计算下个更新周期的 duty cycle

Plan

- ❑ 直流无刷马达介绍
- ❑ FOC 基础
- ❑ Clark 及 Parke 坐标系转换
- ❑ Circle limitation
- ❑ 磁链及力矩控制器
- ❑ 电流读取
 - ❑ 3 电阻法
 - ❑ 电流传感器法
- ❑ 转子的速度 / 位置反馈：
 - ❑ Hall 传感器
 - ❑ 正交编码器

转子的速度/位置反馈



- 如前所述，Park变换的执行需要知道转子的位置
- “转子的速度/位置反馈”模块为FOC算法提供了转子的角度信号并且为速度外环提供了速度信号（在速度控制模式下）
- 当前的库文件支持三种转子的速度/位置反馈：
 - 3 Hall 传感器 (60° or 120°)
 - 正交编码器
 - 无传感器

速度变量格式

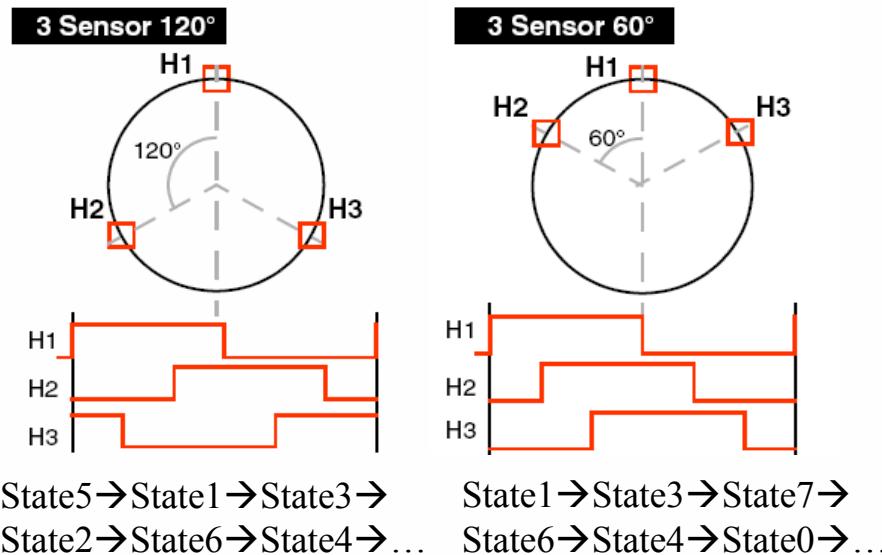
- 在库文件中使用了两种格式：
 - $F_{0.1Hz}$ ，该格式供速度环PID（在速度控制模式下）及用户界面层使用
 - Digit Per PWM (F_{dpp})，其表示每个PWM周期转子角度的变化量 (s16)。该格式可直接累加从而得到转子的位置。 $F_{0.1Hz}$ 与 F_{dpp} 的关系为：

$$F_{dpp} = F_{0.1Hz} \frac{65536}{10 \cdot F_{PWM}(Hz)}$$

Hall 传感器

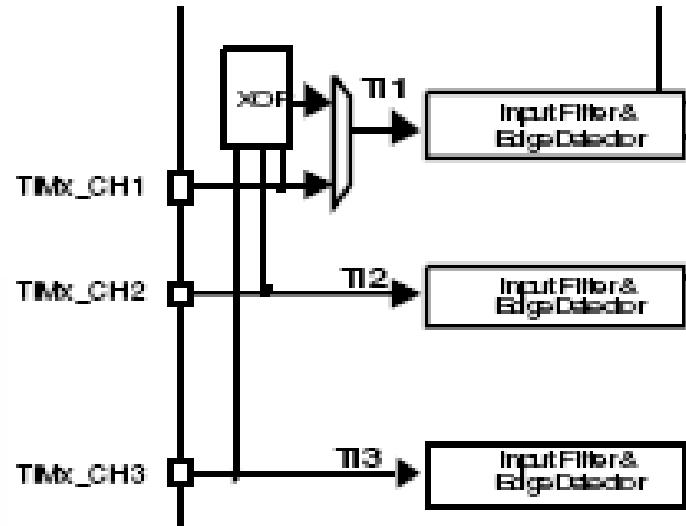


- Hall 传感器可感知转子磁场的极性
- 一般地，在三相永磁马达中：
 - 三个Hall 传感器用于反馈转子的位置信号；
 - 该三个Hall 传感器可以是120°或60°放置的。



STM32的Hall 传感器接口

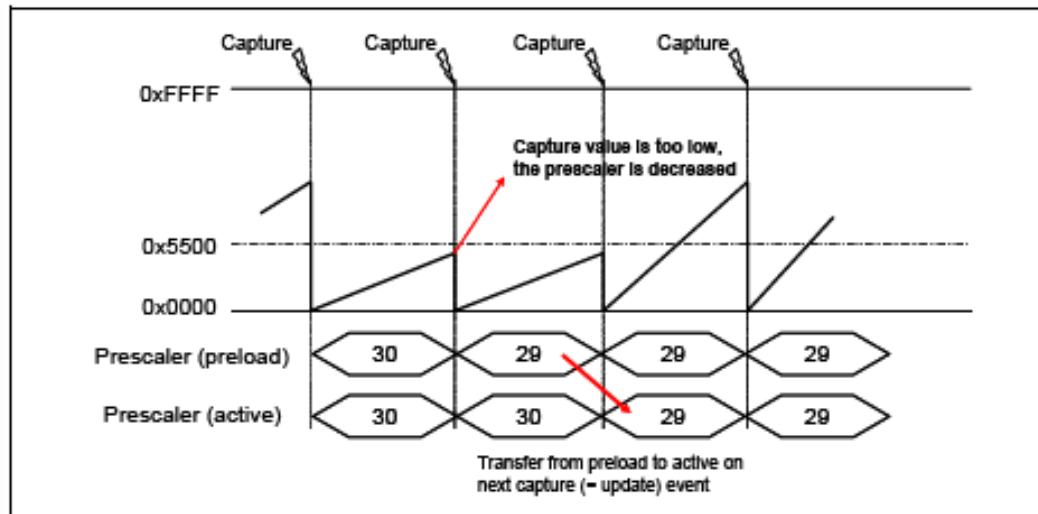
- STM32的定时器在发送Hall信号到输入捕捉逻辑模块之前对这三个信号进行异或；



- 当输入捕捉发生时，计数器的值先保存而后计数器被硬件清0 (clear-on-capture)；
- 注意：库文件中的输入捕捉极性是异或信号的下降沿。

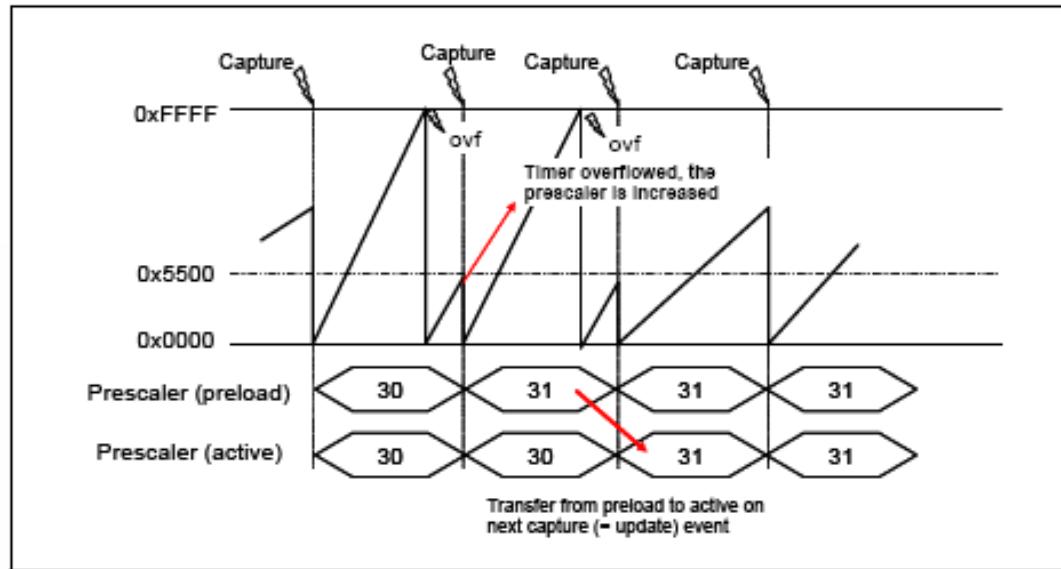
速度检测 1/2

- 转动方向的确定：基于Hall信号的当前及前一次状态；速度决对值的确定：基于输入捕捉寄存器及预分频器的值。
- 为了保持测量精度，定时器的预分频器须根据捕捉值不断地进行调整
- 若计数器值太小，预分频器(为预装载寄存器)被减小：



速度检测 2/2

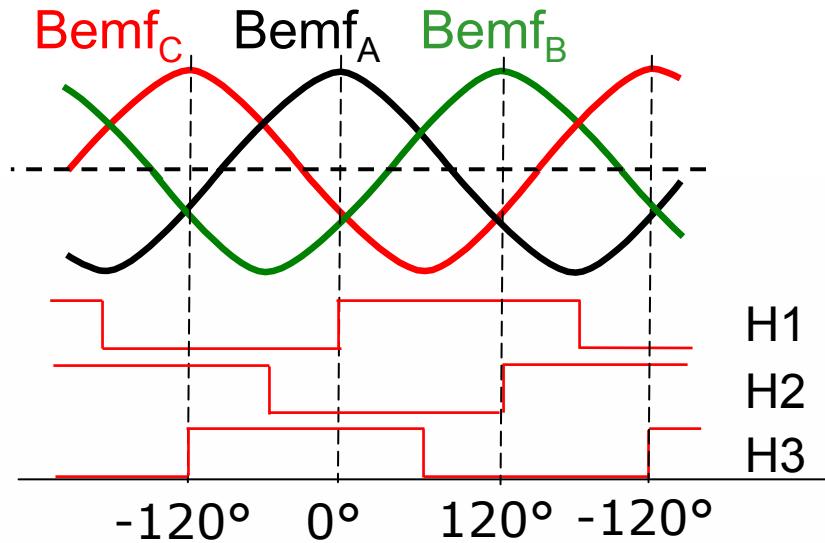
- 相反，若计数器值上溢，预分频器(为预装载寄存器)被增大：



- 由于预分频器寄存器为预装载的，其新的值仅在下一个捕捉事件时起作用，这样就不会干扰测量。

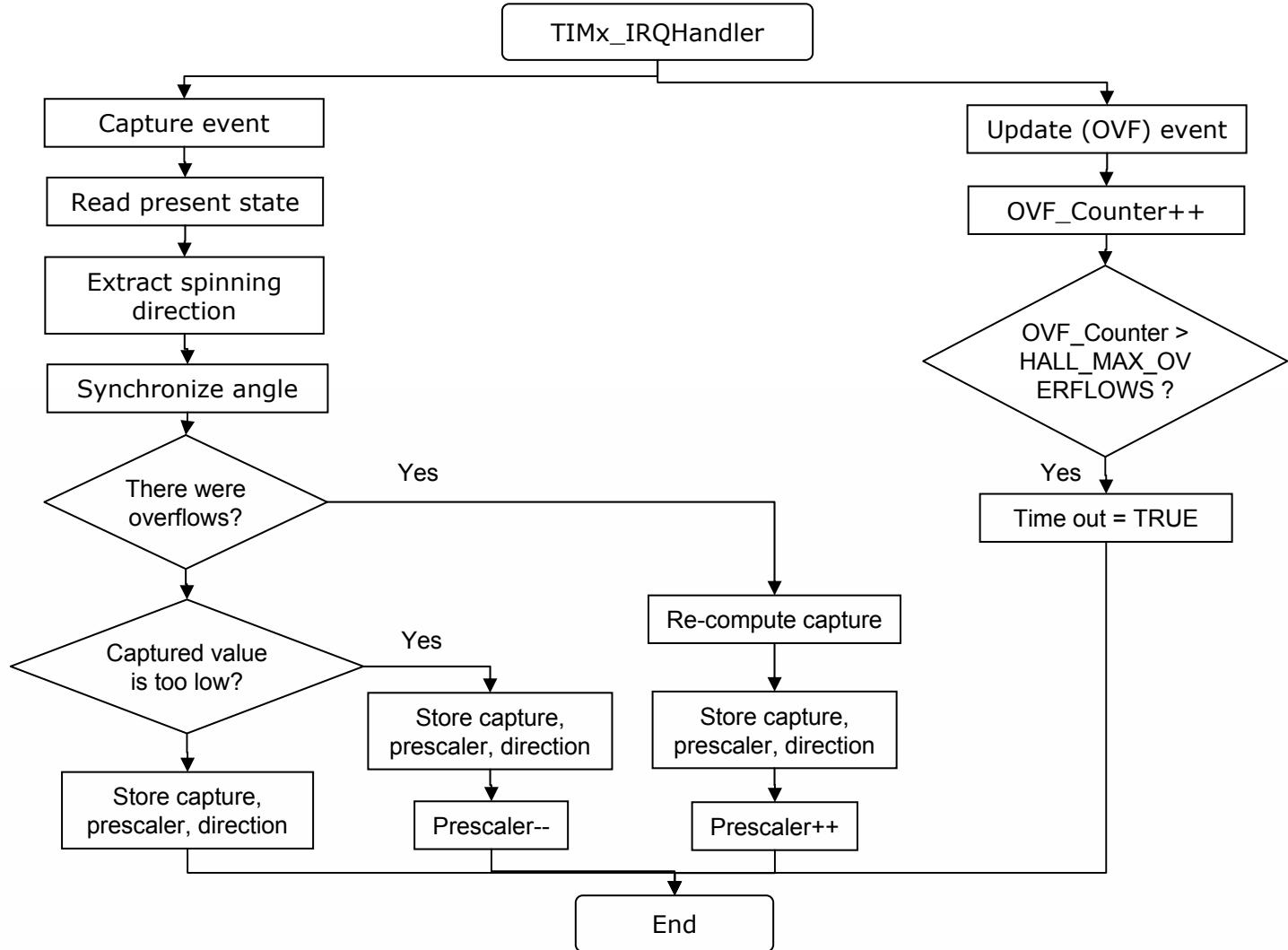
电角度的计算

- 根据Hall的当前状态及转动方向，每次输入捕捉发生时，电角度被同步：



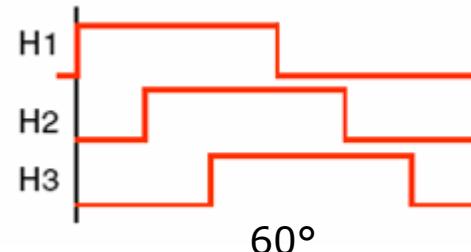
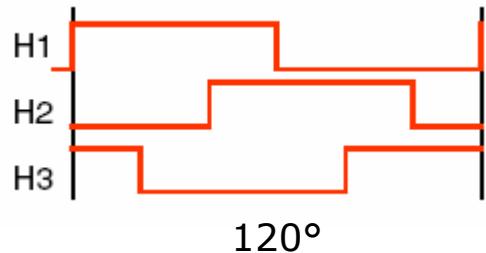
- 每次FOC算法执行时，转子电角度被更新：仅需要对 F_{dpp} 进行积分

stm32f10x_hall.c

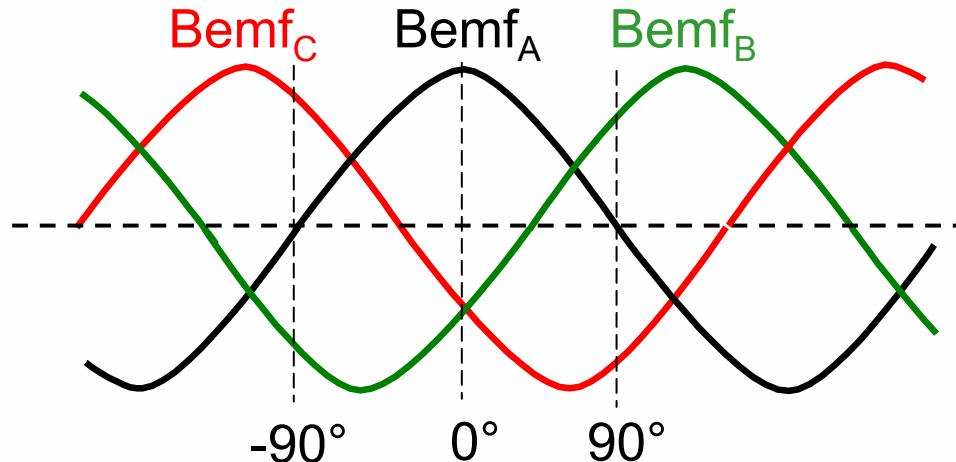


规定

- 库文件要求：H1，H2及H3信号在正转时的时序如下：



- 在库文件中，角度0° 的位置解释为A相的反电动势最大处：



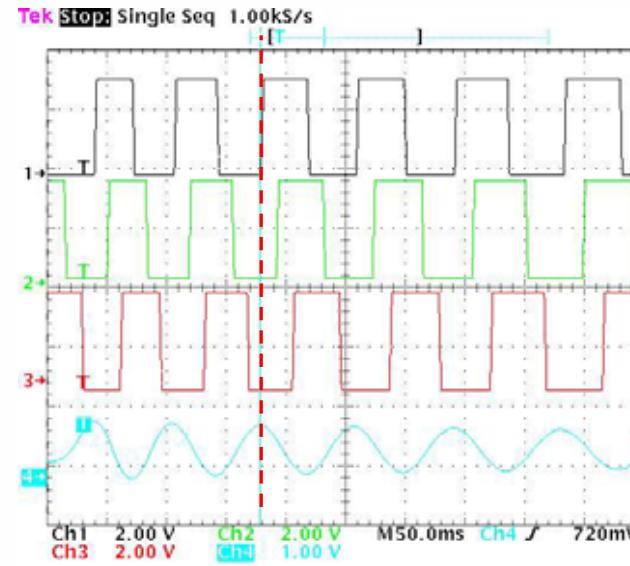
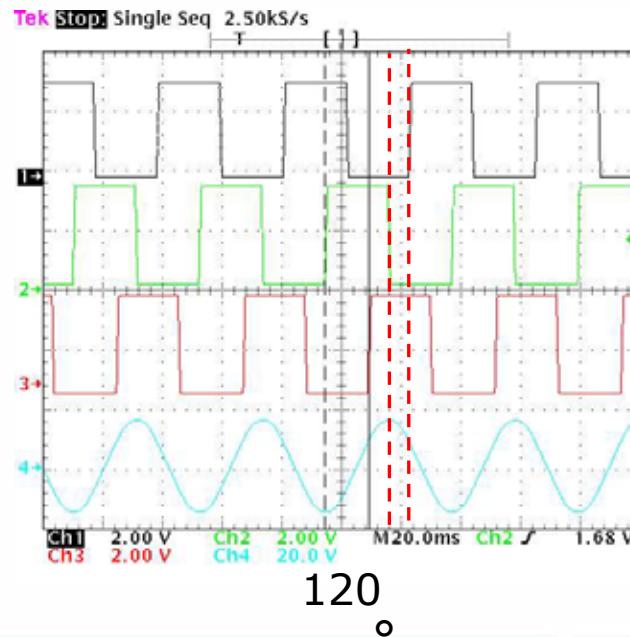
怎样连接Hall传感器到MB459板 1/2

- 任选一个马达的相作为参考，把它插入A相插槽(J5-2)；
- 用手朝你规定的正方向转动马达，用示波器测量三个相端与虚拟中心点间的反电动势波形；
- 与A相的反电动势相差 $+120^{\circ}$ 的相为B相(J5-3)，另一相则为C相 (J5-1)。

怎样连接Hall传感器到MB459板 2/2



- 检测三个Hall信号的输出(如有可能，使用三个上拉电阻) 及A相的反电动势(使用差分探头)；
- 按照H1，H2及H3信号的规定，测量A相反电动势的最大处与H1上升沿的相移：



MC_hall_prm.h

- 当“stm32f10x_MCconf.h”中定义了“HALL_SENSORS”或“VIEW_HALL_FEEDBACK”，须填写下列内容：
 - **TIMERx_HANDLES_HALL**: 定义用于检测HALL的定时器；
 - **HALL_SENSORS_PLACEMENT**: 120° 或60° (DEGREES_120 或 DEGREES_60)
 - **HALL_PHASE_SHIFT**: 电角度相移(degrees)，即：0°与TIMERx 输入信号CC1上升沿之间的电角度(STM32 MC-KIT上的H1信号)；
 - **HALL_MAX_SPEED_FDBK_RPM**: 转子机械转速(rpm)，当大于它时，认为速度反馈错误；
 - **HALL_MAX_SPEED**: 检测到速度大于 HALL_MAX_SPEED_FDBK_RPM时，由函数HALL_GetSpeed返回的值(单位：0.1Hz)；
 - **HALL_MAX_PSEUDO_SPEED**: 检测到速度大于 HALL_MAX_SPEED_FDBK_RPM时，由函数HALL_GetRotorFreq返回的值(单位：dpp)；
 - **HALL_MIN_SPEED_FDBK_RPM**: 转子机械转速(rpm)，当小于它时，认为速度反馈错误；
 - **HALL_MAX_RATIO**: 定义了最大的预分频系数(可避免在马达停机时，过多增大预分频系数)，其限制了最小的可检测转速；
 - **HALL_MAX_OVERFLOW**: 定时器连续溢出的次数，当达到该值时，报告“Speed feedback”故障；
 - **HALL_SPEED_FIFO_SIZE**: 速度FIFO的长度，该FIFO用于平均检测的转速。

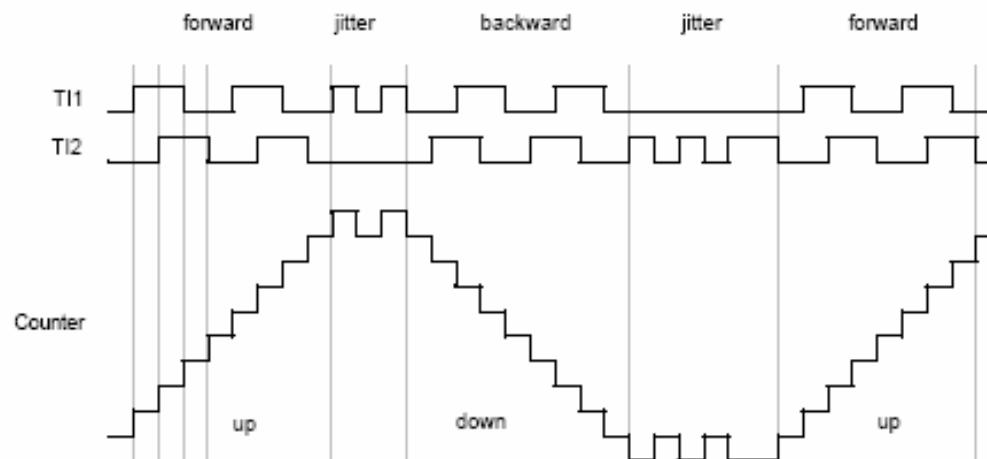
正交编码器

- ❑ 正交编码器产生两个方波信号，它们相差 $+/-90^\circ$ ，其符号由转动方向决定；
- ❑ 它提供了转子的位置信号，其值非常精确，但它是**个相对位置值**；
- ❑ 由于STM32具有正交编码器接口，在转子位置与正交编码器接口的定时器之间可以建立一对一的关系；
- ❑ 为了同步上述两者的关系，在第一次上电的起动前需要定位，而以后的起动则不再需要，除非MCU再次上电或发生故障。

STM32的正交编码器传感器接口



- 定时器的两个输入TI1和TI2用于与增量式正交编码器接口；
- 这两个信号的边沿作为计数器的时钟($4x$)，而ARR寄存器初始化为：
 $(4 * \text{PULSE_PER_REVOLUTION}) - 1$ ；
- 根据这两个信号时序的关系，计数器向上或向下计数

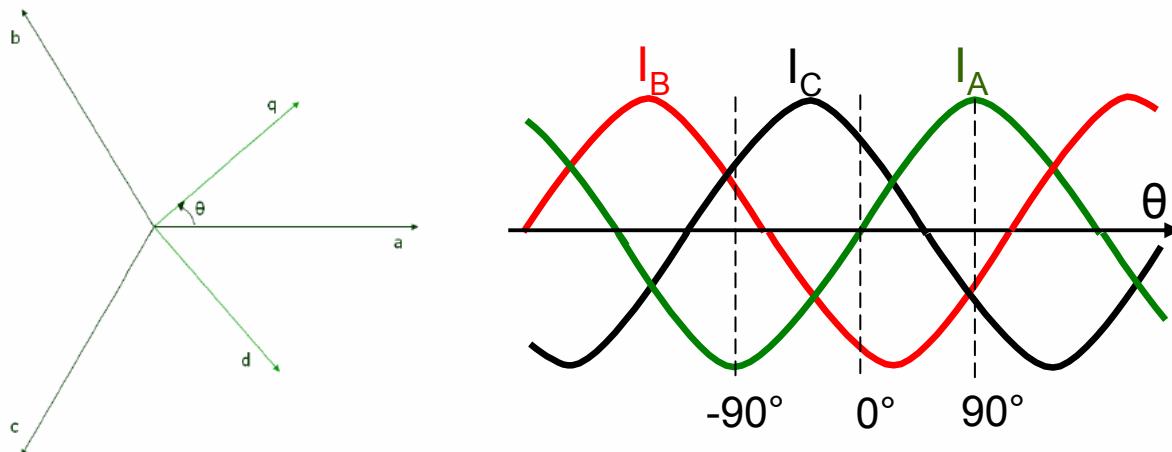


转子定位规律



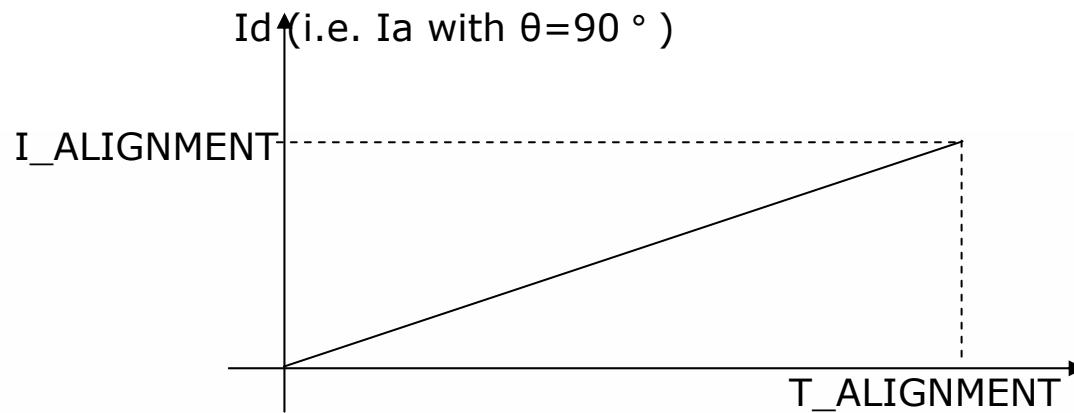
STM32

- 在MCU复位或发生故障后的第一次起动，才需要转子定位；
- 通过在定子中加入一个以下的磁链来实现：
 - 幅值呈线性增加且受控的(I_d)；
 - 选择一个转子角度。
- I_a , I_b , I_c 及磁链定向的关系如下图，它是 θ 角的函数：



定位参数

- “I_ALIGNMENT”及“T_ALIGNMENT”必须选择合适的值，以至于当计数器的初始化结束时，转子无任何震荡：



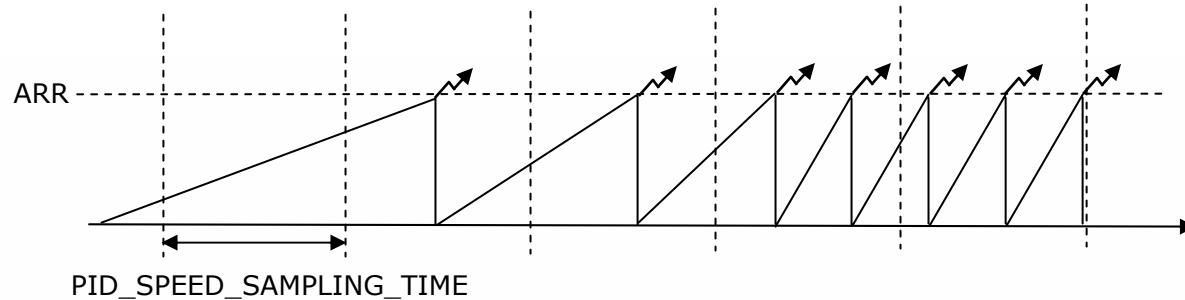
$$I(\text{digit}) = \frac{I(A) \cdot R_{\text{shunt}} \cdot A_V}{3.3} \cdot 65536$$

$A_V = 2.57$ on MB459B

速度检测



- 相对于Hall传感器信号是检测“周期”，此时应检测编码器信号的频率；
- 在每个“PID_SPEED_SAMPLING_TIME”周期后，检测定时器计数器值的变化；



- 由于该时基由SysTick定时器提供(低优先级)，可能会存在最大30us的延时；
- 如果每2ms检测一次速度，其最大延时可小于1.5%

电角度的计算

- 在定位后，计数器的值给出了确切的转子电角度；
- 因此，提供给FOC算法的电角度由下式给出：

$$\text{El angle} = \frac{65536 \cdot \text{Timer counter}}{4 \cdot \text{ENCODER_PPR} * \text{POLE_PAIRS_NUM}}$$

它由函数“*ENC_Get_Electrical_Angle()*”给出。

怎样连接正交编码器到MB459板

- 当输入正向的三相电流到马达时，库文件希望读到的速度值也为正的(即：定时器是向上计数的)；
- 手动地转动马达，使之向着假定的正方向运行，检测反电动势，以确定马达的相序连接正确：J5 2-3-1对应于相序A，B，C；
- 连接编码器的输出到J8 H1-H2，跳线W17及W18应装上，库文件运行在编码器模式下。手动地转动马达，使之向着假定的正方向运行，检查测量到的速度值：
 - 若其 > 0 ：说明连接OK (即：TI1领先于TI2)
 - 若其 < 0 ：你需要交换编码器的信号线(以保证TI1领先于TI2) 或修改代码stm32f10x_encoder.c 的164行：

把：TIM_ICPolarity_Rising, TIM_ICPolarity_Rising);

改为：

TIM_ICPolarity_Rising, TIM_ICPolarity_Falling);

MC_encoder_param.h

- 当“stm32f10x_MCconf.h”定义了“ENCODER”或“VIEW_ENCODER_FEEDBACK”，应填写以下内容：
- **TIMERx_HANDLES_ENCODER**: 定义了接口编码器的定时器；
- **ENCODER_PPR**: 编码器每转一周，单个通道产生的脉冲数；
- **MINIMUM_MECHANICAL_SPEED_RPM**: 转子机械转速(rpm)，当小于它时，认为速度反馈错误。此时，错误计数器会加1；
- **MAXIMUM_MECHANICAL_SPEED_RPM**: 转子机械转速(rpm)，当大于它时，认为速度反馈错误。此时，错误计数器会加1；
- **MAXIMUM_ERROR_NUMBER**: 连续速度错误的次数，当达到该值时，报告“Speed feedback”故障；
- **SPEED_BUFFER_SIZE**: 速度FIFO的长度，该FIFO用于平均检测的转速；
- **T_ALIGNMENT**: 定位周期(单位：ms)；
- **ALIGNMENT_ANGLE**: 定子磁链的定位角(单位：度)
- **I_ALIGNMENT**, 最终的Id幅值(digit)；