

深入浅出FOC(Field Oriented Control)

前言:

为什么要学习FOC?

1.电机控制是自动化控制领域重要一环。 2.目前直流无刷电机应用越来越广泛，如无人机、机械臂、云台、仿生机器人等等。 3.电机控制工程师薪水较高。

需要什么基础?

1.C语言，指针，结构体，编程规范。 2.STM32外设使用。 3.原理图阅读。 4.芯片手册阅读。 5.数序坐标系知识

为什么要出本教程?

1.直流无刷电机应用越来越广泛，网上资料比较散落，因此想要出一篇系统性的教程，从头到尾，深入浅出，帮助初学者快速入门直流无刷电机控制。

版本

V1.0.0

联系方式

QQ群：585424959

作者

我家老母猪兴趣小组

第一章 直流无刷电机简介

基本物理知识准备

左手定则

[百度百科](#)

已知电流方向和磁感线方向，判断通电导体在磁场中受力方向，如电动机。伸开左手，让磁感线穿入手心（手心对准N极，手背对准S极），四指指向电流方向，那么大拇指的方向就是导体受力方向。

右手螺旋定则

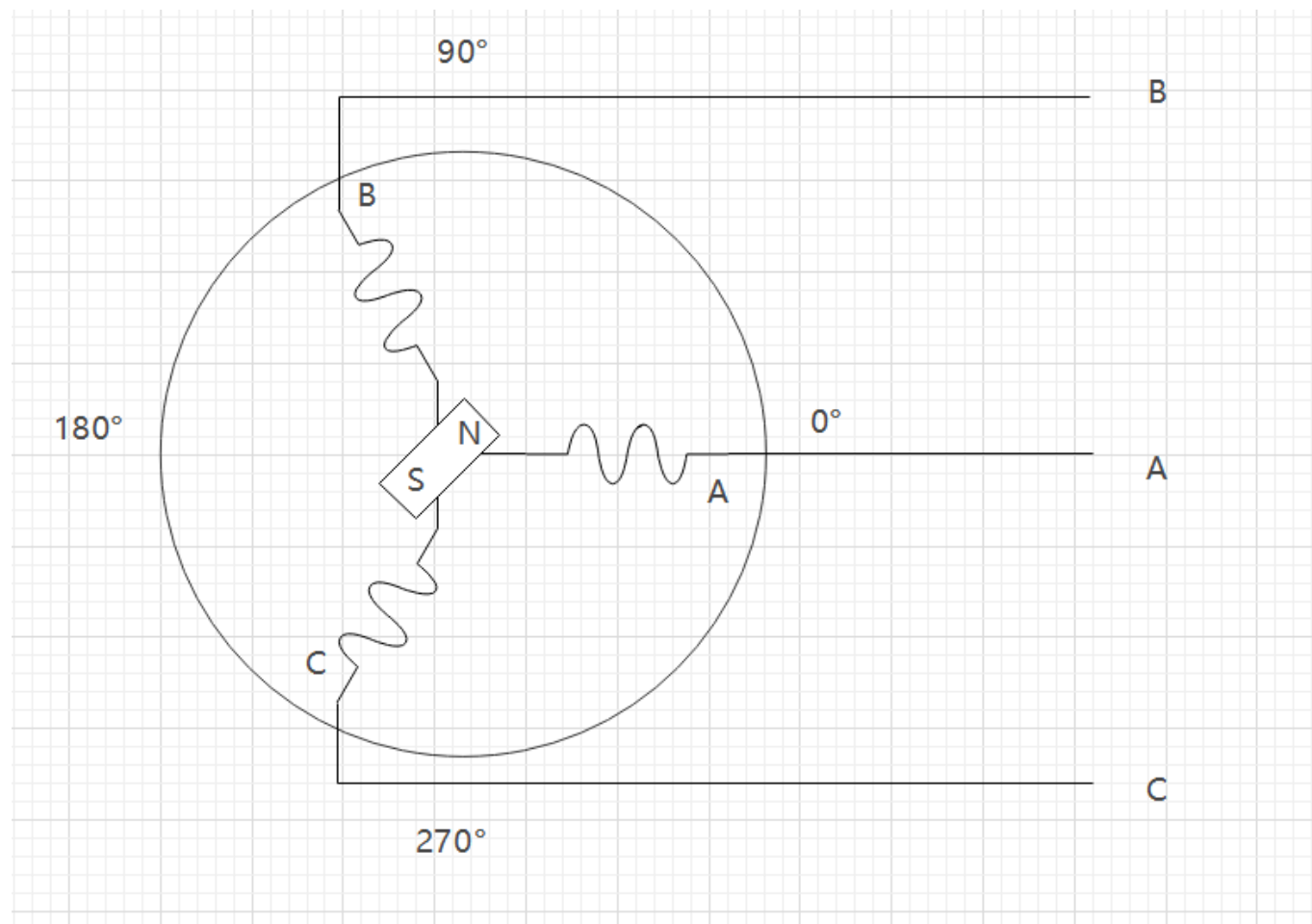
[百度百科](#)

用右手握住通电螺线管，让四指指向电流的方向，那么大拇指所指的那一端是通电螺线管的N极

直流有刷电机简介

直流有刷电机通过换向器来改变电流方向，进而改变绕组的受力方向。由于其是机械换向，因此就带来一系列缺点，例如摩擦大，发热大，效率低等缺点

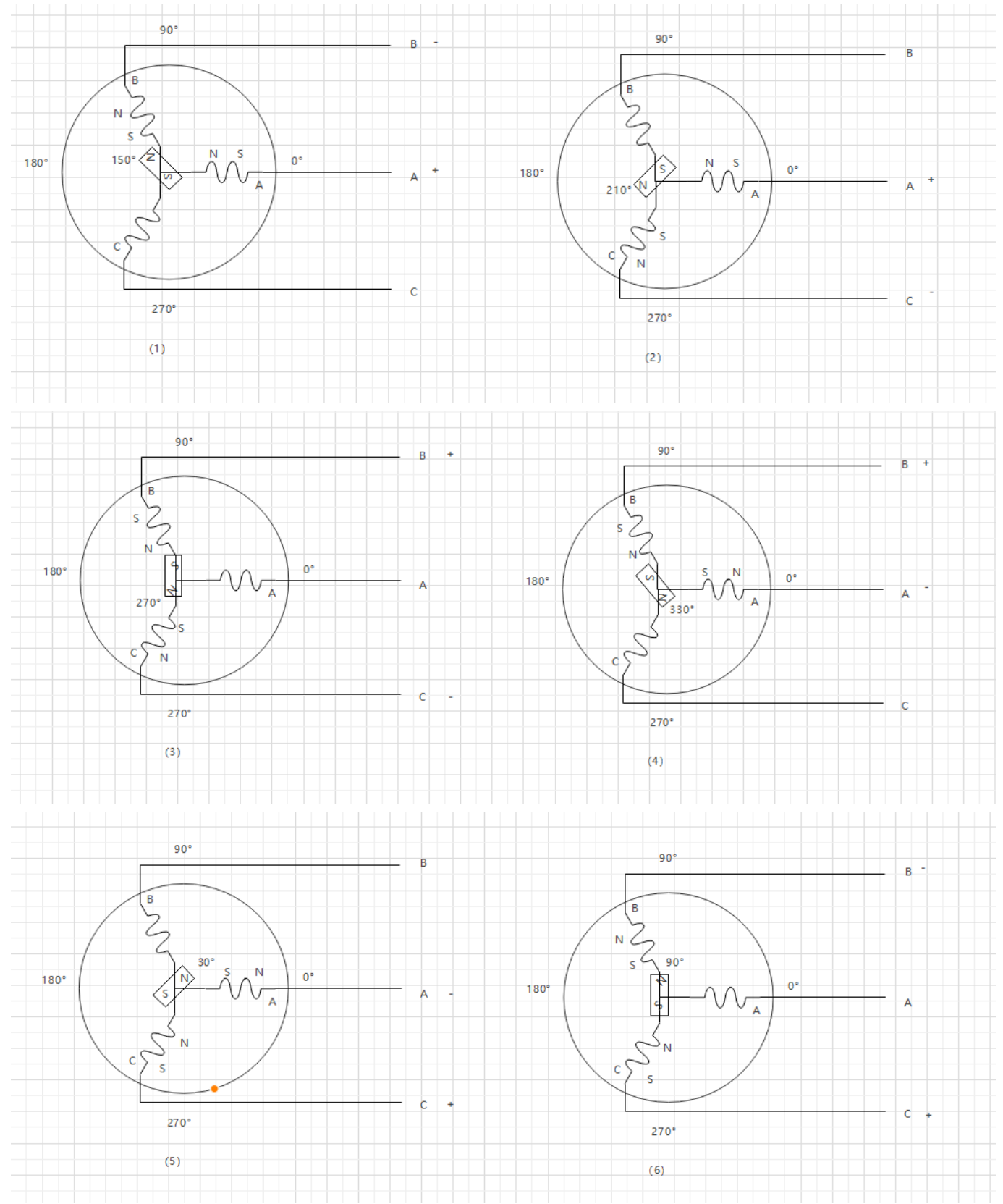
直流无刷电机简介



直流无刷电机通过使用电子器件代替机械换向，解决了直流有刷电机的缺点。为了便于分析我们将直流无刷电机抽象出上图模式，定子由三个线圈组成，转子由一对磁极组成。通过改变ABC三者电流方向来改变定子产生的磁场方向，从而使磁铁转动起来。

第二章 六步换向控制方式

直流无刷电机六步换向



如上图所示，通过控制ABC三相电流方向我们可以控制直流无刷电机旋转，具体步骤如下：

步骤	A相电压	B相电压	C相电压	转子目标角度
1	+	-	悬空	150
2	+	悬空	-	210
3	悬空	+	-	270

步骤	A相电压	B相电压	C相电压	转子目标角度
4	-	+	悬空	330
5	-	悬空	+	30
6	悬空	-	+	90

可以看出，我们只需要像步进电机那样不断的重复这六步换向就可以让BLDC转动起来，甚至会产生一种错觉，是不是我们换向越快电机转的越快呢？答案是:否，这里我们一定要认识到，是当转子处于特定位置时才去触发换向操作，换向是被动换向，想要提高转速一定是要提高电流，让定子产生的磁场更强，让转子更快的达到目标点然后触发换向，具体如下表所示。

正转：

转子实际角度	执行步骤	转子目标角度
90-150	1	150
150-210	2	210
210-270	3	270
270-330	4	330
330-30	5	30
30-90	6	90

反转

转子实际角度	执行步骤	转子目标角度
150-90	6	90
90-30	5	30
30-330	4	330
330-270	3	270
270-230	2	230
230-150	1	150

如何获得转子角度？

我们已经知道了要先检测角度再去换向，那么如何检测当前角度呢？，有以下三种方式。 1.通过安装编码器来计算当前角度。 2.通过安装霍尔元件计算当前角度。 3.通过检测电流来计算当前角度

编码器方式获取电机当前角度

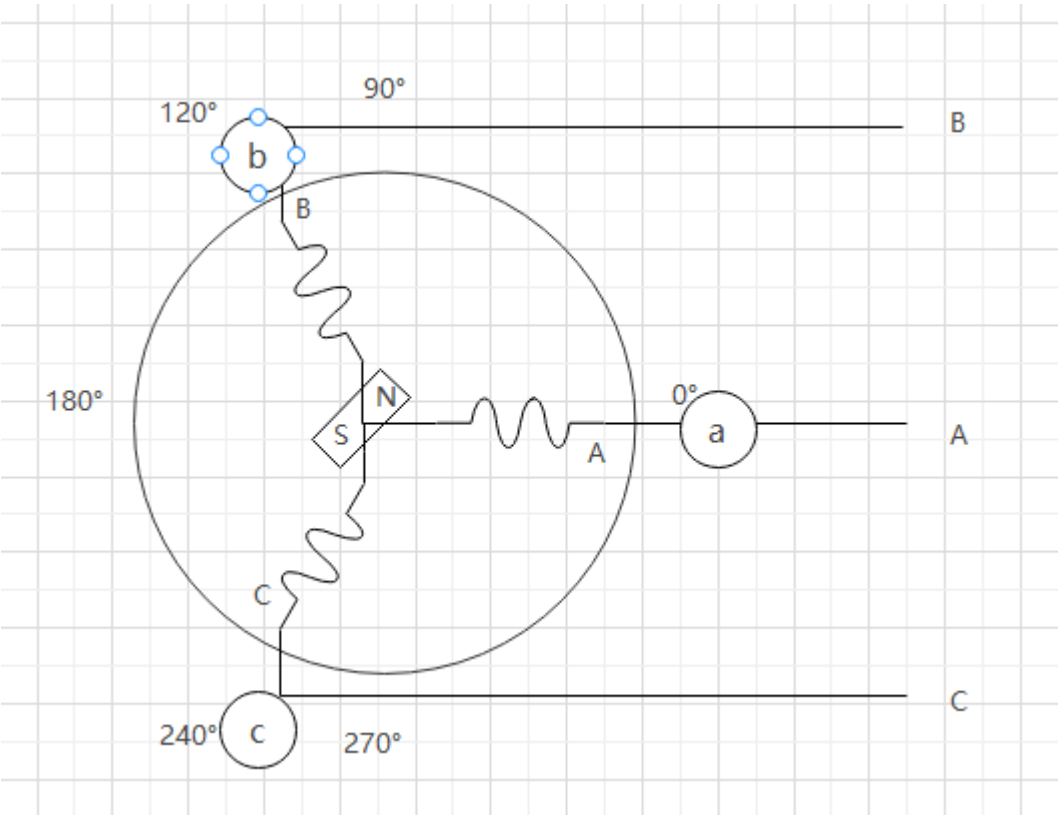
编码器方式分为两种，增量式编码器和绝对式编码器。 增量式编码器： 每次启动之气都需要做一次校准，而且为了防止单片机性能问题导致脉冲丢失，还需要对编码器每圈校准一次。因此经常使用ABZ三轴编码器，AB输

出正交信号，Z轴输出中断。绝对式编码器：只需要在出厂之前做一次校准，之后如果没有拆机便不需要校准，通讯方式一般是SPI和IIC,需要考虑通讯时间对系统的影响。为什么要对编码器进行校准？因为我们无法保证在安装的时候让编码器的0°（机械角度）刚好对应电机绕组的0°（电气角度）

霍尔元件获取电机当前角度

霍尔元件介绍

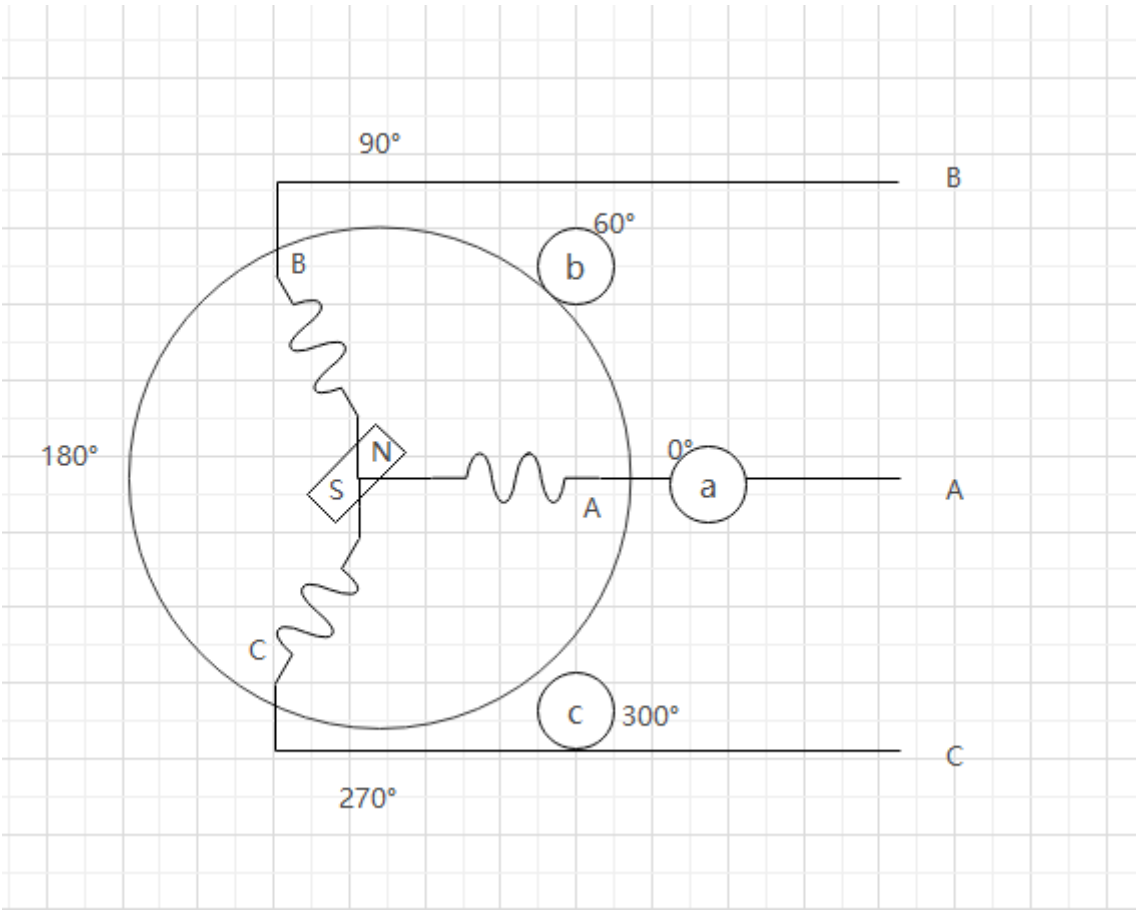
百度百科 霍尔元件有两种安装方式，互差120°和互差60°两种方式，因此在使用之前要判断电机是那种安装方式。假设当有N级指向霍尔元件时霍尔元件输出电平1 霍尔电平和转子角度之间的对应关系如下表所示



【缺少坐标表示、说

明】

转子实际角度（N级指向）	霍尔元件a电平	霍尔元件b电平	霍尔元件c电平
90-150	0	1	0
150-210	0	1	1
210-270	0	0	1
270-330	1	0	1
320-30	1	0	0
30-90	1	1	0

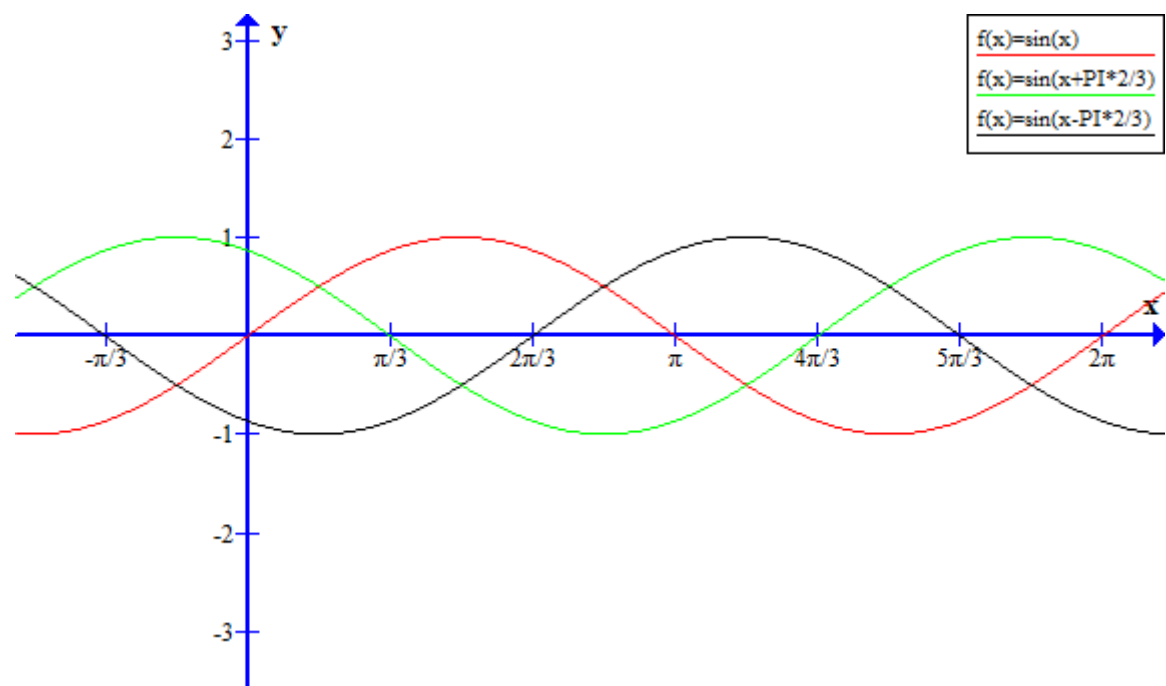


霍尔电平和转子角度之间的对应关系如下表所示

转子实际角度（N级指向）	霍尔元件a电平	霍尔元件b电平	霍尔元件c电平
90-150	0	1	0
150-210	0	0	0
210-270	0	0	1
270-330	1	0	1
330-30	1	1	1
30-90	1	1	0

检测电流获取电机当前角度

由于电机旋转过程中产生的电流理想波形如下图



我们便可以根据采集到的电流就判断出电机当前的角度在哪个范围。 具体分析在FOC控制方法中讲解。

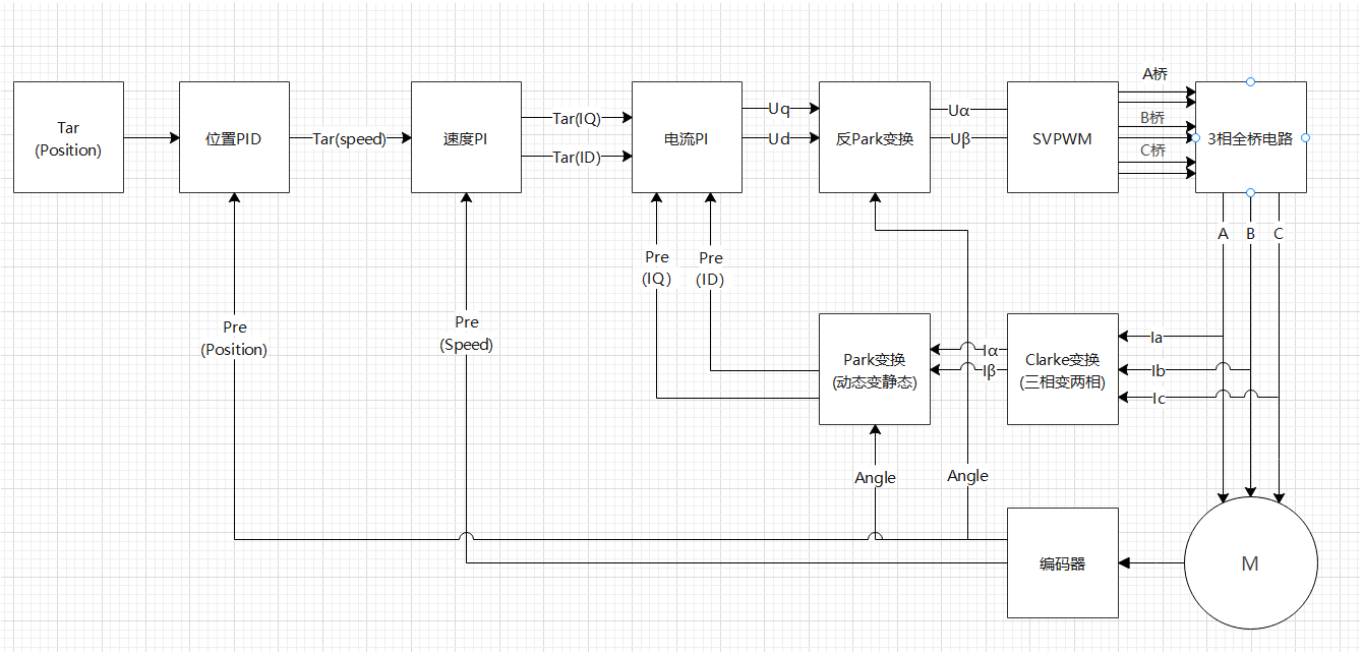
程序编写

引出问题

相信看到这里大家都有一些疑问，就是六步换向的方式只能控制电机正反转，然后通过控制相电压调节转速，那么如何控制电机位置呢，而且这种方式有一部分电流是让电机旋转，有一部分是让电机发热，那么如何做到效率最高呢。通过观察电流发现，电机三相电流并不是正弦波，为了解决以上问题便出现了FOC矢量控制。

第三章 FOC矢量控制

FOC控制框架图



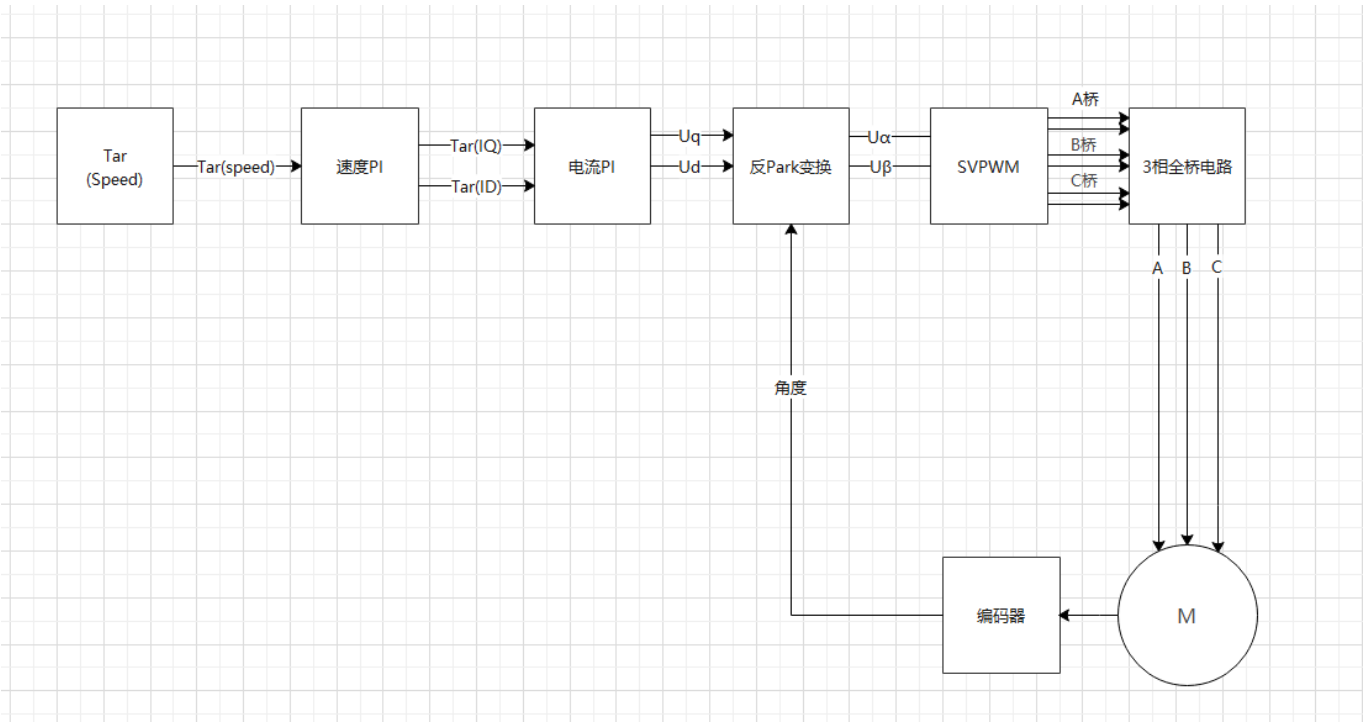
相信很多人在网上都看到过这个控制框图，咋一看还挺复杂，充斥着各种变换和检测，很多资料直接对着这个框图进行讲解，而本教程到这里只是让大家对这个框图有基本印象，接下来我们逐步分析各个模块来源，让大

家都能自己画出来这个控制框图。

第四章 FOC开环控制

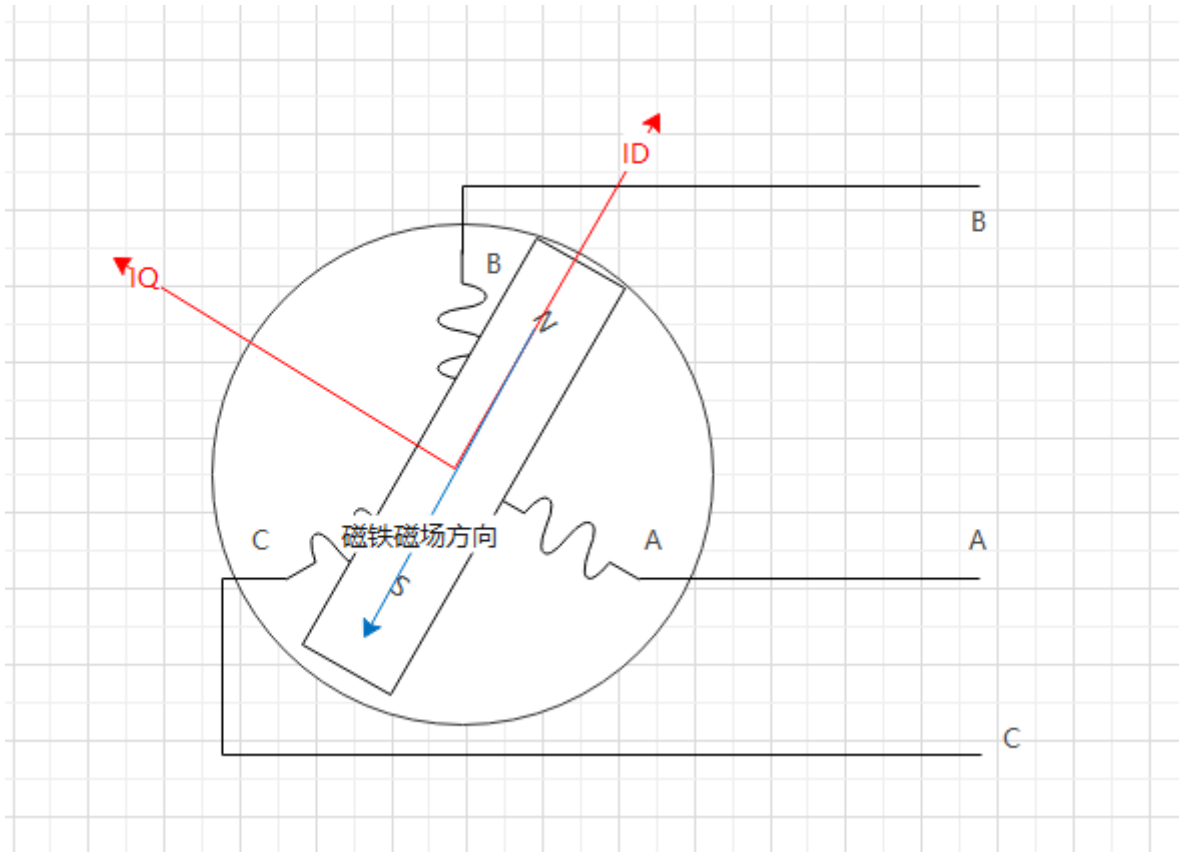
控制框图简化

看着FOC详细控制框图我们发现无从下手，不知道从哪坐起，我们回想一下直流有刷电机的学习过程，首先是让电机转起来，然后进行速度控制，再进一步进行位置控制，同样我们在FOC学习过程中依然可以这样做，我们首先将位置环和速度环去掉，然后就剩下SVPWM，和电流检测部分，既然只是让电机转起来那么电流检测也不需要了，我们就直接给一个目标速度，开环运行，这时候控制框架就能简化成下图所示。备注：(这时候可能会有人有疑问，我们不是用六步换向的方式已经让电机转动起来了么，为什么这里又要学习如何让电机转动？是因为六步换向方式存在一些缺点，而这里使用另一种控制方式（FOC）让电机转动更加平滑，效率更高。)



接下来我们对这部分做详细讲解

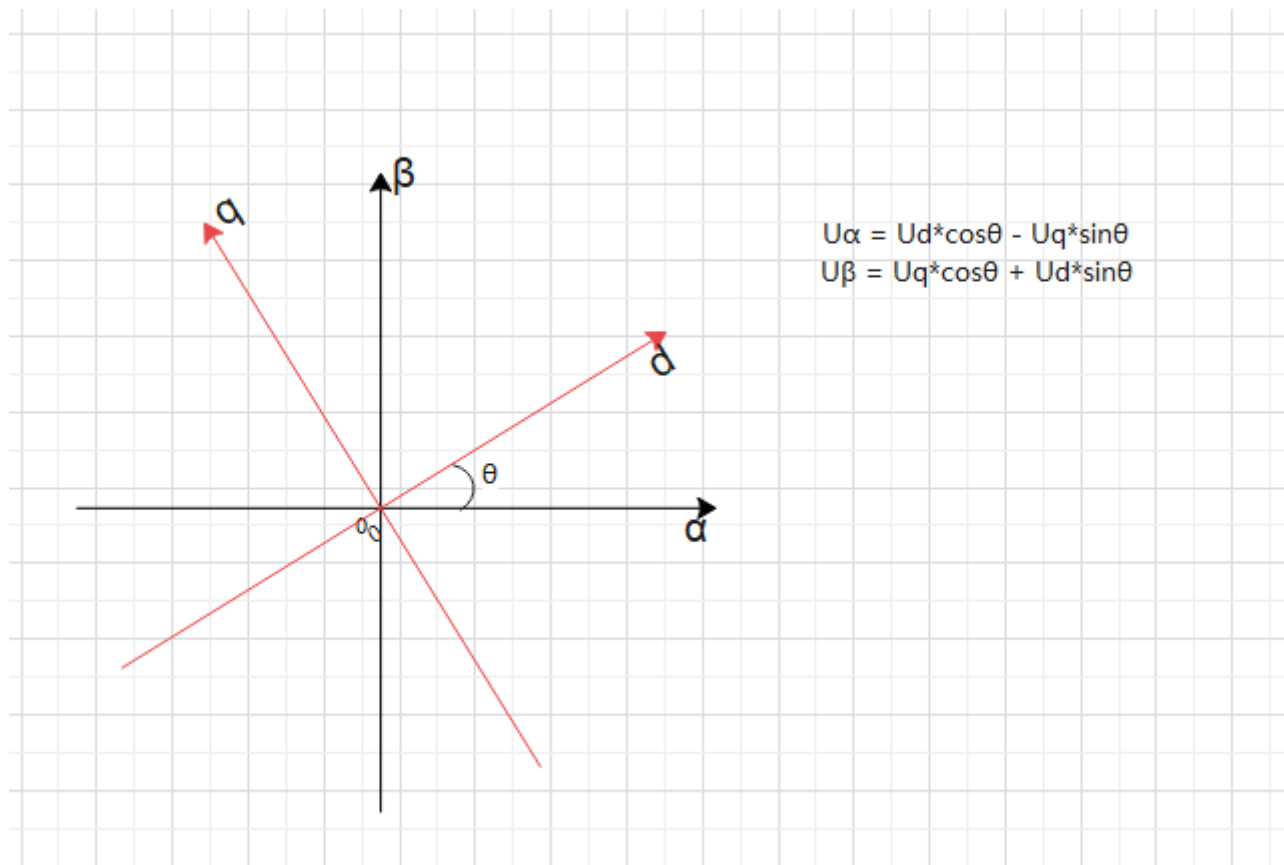
IQ与ID介绍



仔细观察我们可以发现，当三相绕组产生的磁场方向始终与转子磁铁相切的时候最为理想，这样相同电流下产生的旋转力量最大（图中 I_Q ），当我们三相绕组产生的磁场方向与转子磁场方向反向平行时，这时候电机会被吸在原地不动(图中 I_D)，电流都用来产生热量。因此我们以转子磁铁为参考，建立DQ坐标系，DQ坐标系随转子转动。（备注：为了便于理解我对磁铁进行了放大）

如图当我们让 I_D 为0, I_Q 为1则转子就会逆时针旋转，且转速随着 I_Q 的变大而变大。如图当我们让 I_D 为1, I_Q 为0则转子就会定在原地，且发热量随着 I_D 的变大而变大。因此可以总结一下，要想让转子平稳的转动下去就需要让三相绕组产生的磁场方向一直和转子磁铁的磁场方向垂直，即图中的 I_Q ，同时使 I_D 尽可能为0以减小发热。那么怎么让绕组产生的磁场方向等效为 I_Q I_D 呢由此变引入Park反变换和Clark反变换。

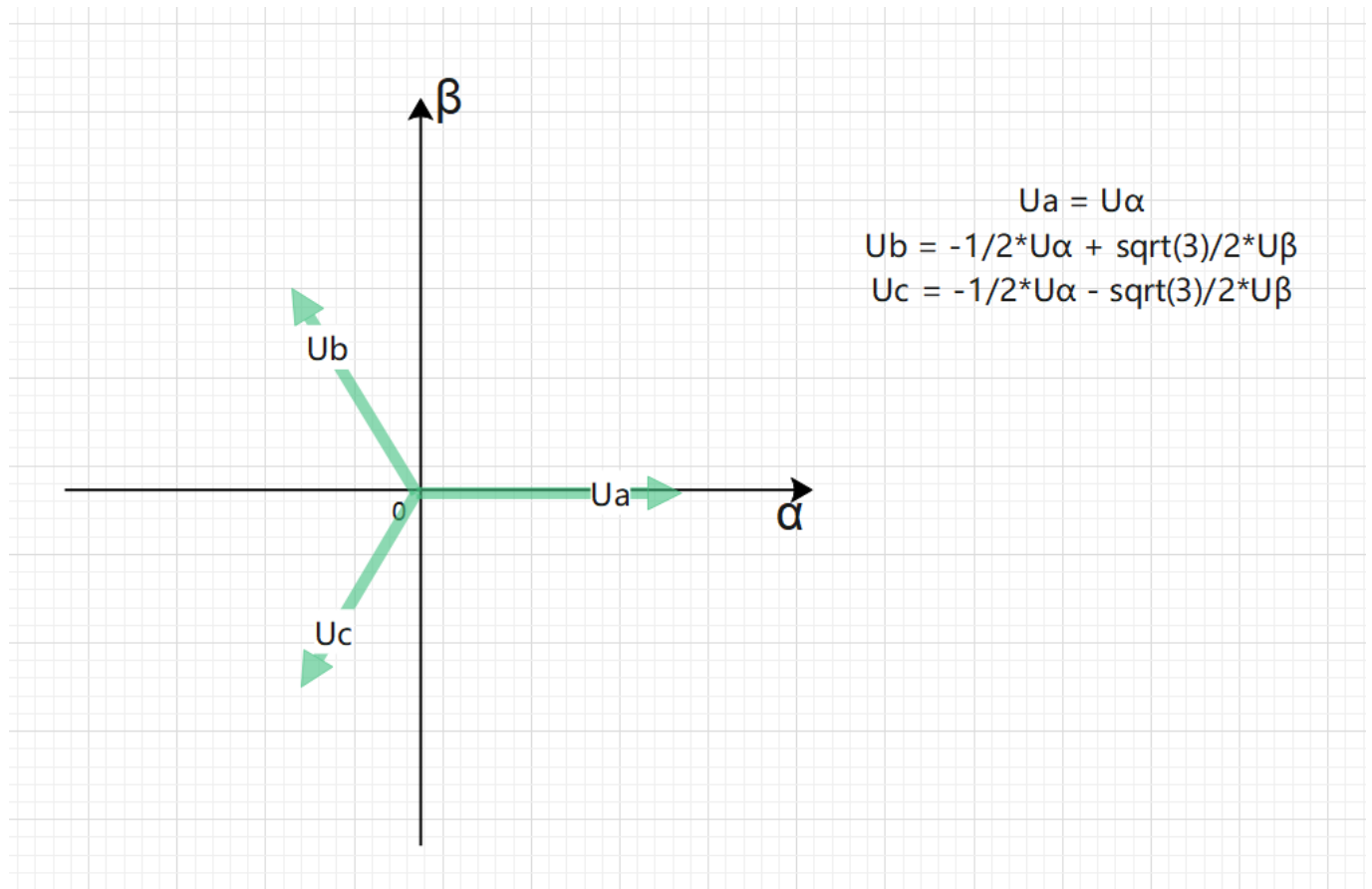
Park反变换



我们知道DQ坐标系是旋转坐标系，但我们的三相绕组是静止的，因此首要任务是将旋转坐标系变换为静止坐标系，即找到一个静止坐标系使其产生的磁场等同于DQ坐标系，我们将这个坐标系命名为 $\alpha\beta$ 坐标系，其中由DQ坐标系向 $\alpha\beta$ 坐标系这一变换过程称为反Park变换。

其中阿尔法轴与a相绕组对齐。 $U_{\alpha} = U_d \cos \theta - U_q \sin \theta$ $U_{\beta} = U_q \cos \theta + U_d \sin \theta$ 经过Park反变换我们将旋转坐标系变成了静止坐标系，接下来便是让静止的两相坐标系向3相绕组坐标系进行变换。

Clark反变换

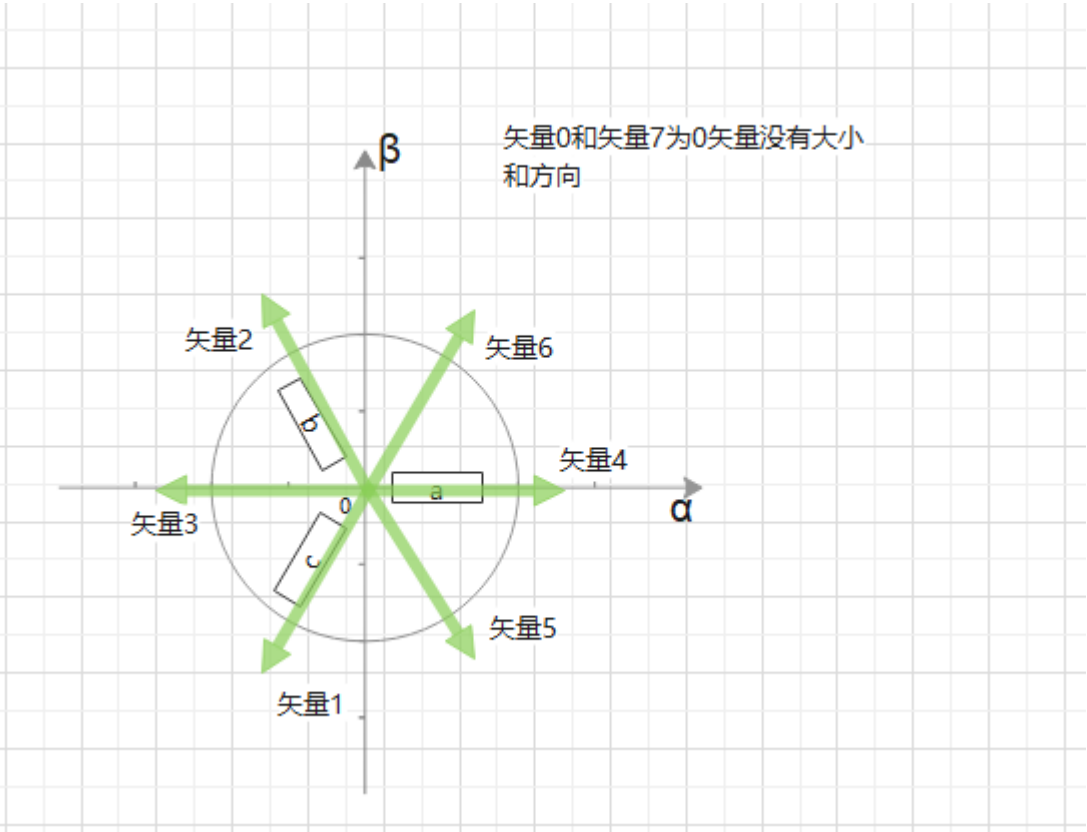


Clark反变换的目的是把两个垂直的坐标系转变为三相绕组坐标系。经过Clark反变换后就可以通过三相绕组去产生ID和IQ了。（备注FOC控制框图中没有Clark变换的原因是通过后边公式的化简发现只需要 I_α 和 I_β 即可）

但是我们最终能控制到的硬件是三相全桥即六个MOS管的导通，因此我们最终要的是MOS管的导通时间，即定时器输出的高电平时间，即PWM的占空比，因此我们接下来要通过 U_a U_b U_c 求出Mos导通时间。

8个矢量介绍

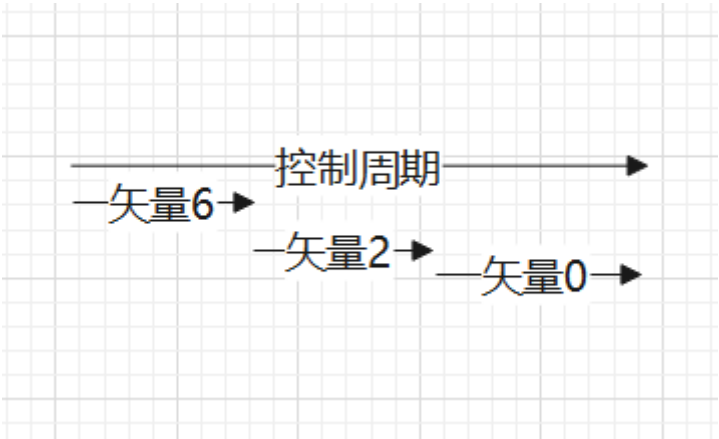
我们已知我们的输入是控制 U_a U_b U_c 导通状态导通时间和转子角度，我们的目标（输出）是ID IQ,我们要解决的问题是如何通过输入产生输出，在解决问题之前我们要先看一看我们的输入是什么，有什么物理含义。由于同一桥臂上下两个MOS不能同时导通(短路)，因此6个MOS总共由8种状态，由于每种状态都会产生一个固定方向的磁场，我们将8种状态称为8种矢量。



矢量	a桥臂状态	b桥臂状态	c桥臂状态
0	0	0	0
4	1	0	0
6	1	1	0
2	0	1	0
3	0	1	1
1	0	0	1
5	1	0	1
7	1	1	1

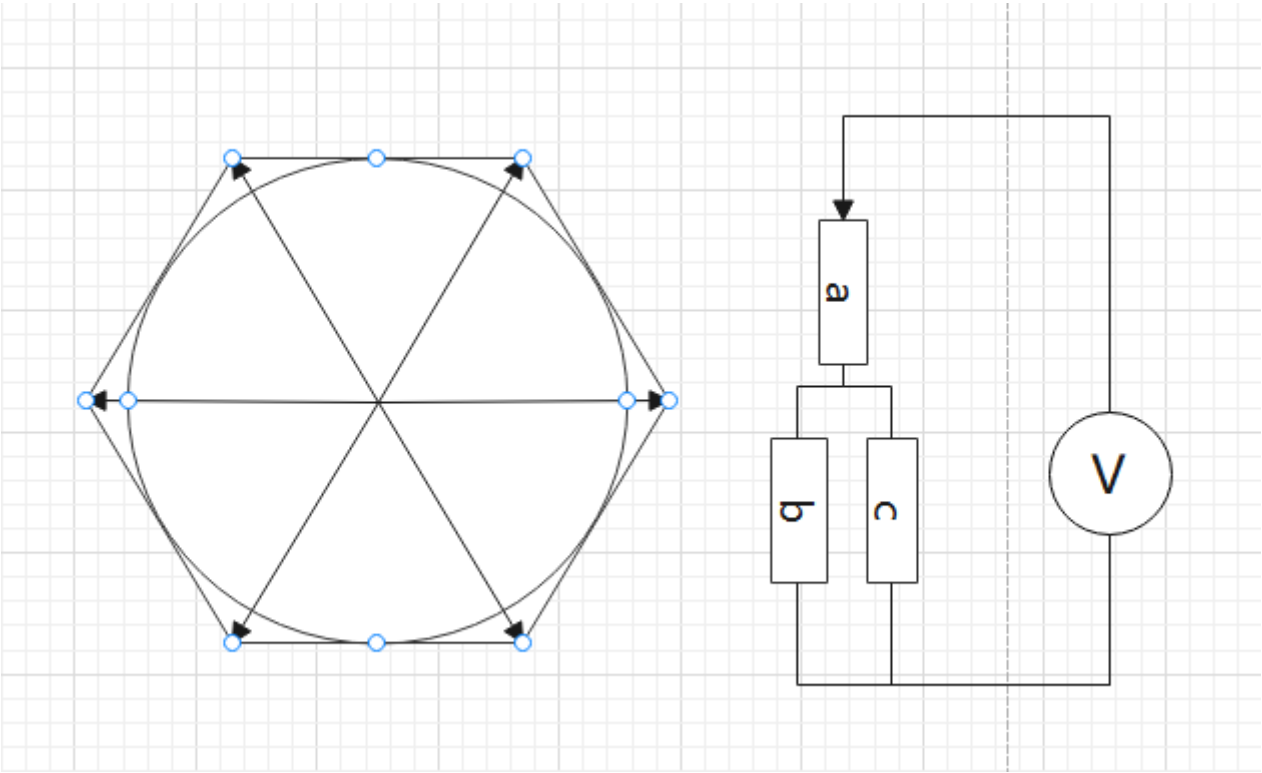
(注：0代表下桥臂打开 1代表上桥臂打开)

假设我们的电机处于0°位置，那么要想让电机转动起来我们只需要产生一个90°的矢量，那么只需要在一个周期内矢量2和矢量6作用的时间各占一半即可产生一个90°的矢量，总结来说就是我们通过控制相邻矢量导通时间比例可以控制矢量的方向，通过插入0矢量可以改变矢量的大小，因此我们在一个控制周期内通过8个矢量的配比来产生任意方向和任意大小的矢量。（只要我们能够产生任意方向的矢量我们就可以先读取当前转子角度，然



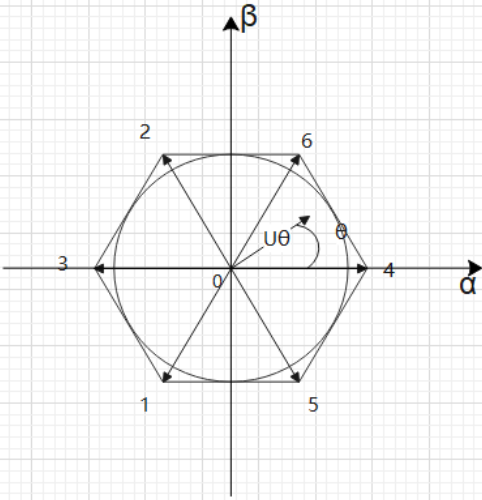
后产生一个超前90度的矢量，这个矢量即是IQ) 。

我们知道矢量方向的范围是0-360度那么矢量的大小是什么呢?由于三相绕组阻值基本一样，因此每个矢量的最大值为 $U_{dc}/3$ ，其中 U_{dc} 为母线电压 (电源电压) ,但我们发现各个方向上矢量最大值并不一样，为了让电机平滑转动，我们的目标矢量的最大值应为上图中六边形的内切圆的半径，即 $U_{dc}/3 \cdot \sqrt{3}/2 = U_{dc}/\sqrt{3}$.具体参考下图。



SVPWM生成

接下来先看一个例子，例如我们想要产生0-60°的任意一个方向的矢量。



设T为控制周期，Tx为矢量x作用时长，
 $|U_4| = |U_6| = \frac{2}{3} * |U_{dc}|$
 $U_{\theta} * T = U_4 * T_4 + U_6 * T_6 + U_0 * T_0$
且 $|U_{\theta}| * \cos\theta = |U_{\alpha}| = |U_4| * \frac{T_4}{T} + |U_6| * \frac{T_6}{T} * \cos(\pi/3)$
 $|U_{\theta}| * \sin\theta = |U_{\beta}| = |U_6| * \frac{T_6}{T} * \sin(\pi/3)$
整理得出：
 $U_{\alpha} * T_s = \frac{2}{3} * U_{dc} * (T_4 + T_6 * \frac{1}{2})$
 $U_{\beta} * T_s = \frac{2}{3} * U_{dc} * T_6 * \frac{\sqrt{3}}{2}$
解算得出
 $T_6 = \frac{\sqrt{3}}{2} * \frac{T_s}{U_{dc}} * U_{\beta}$
 $T_4 = \frac{\sqrt{3}}{2} * \frac{T_s}{U_{dc}} * (\frac{\sqrt{3}}{2} * U_{\alpha} - U_{\beta}/2)$
 $T_0 = T_s - T_6 - T_4$

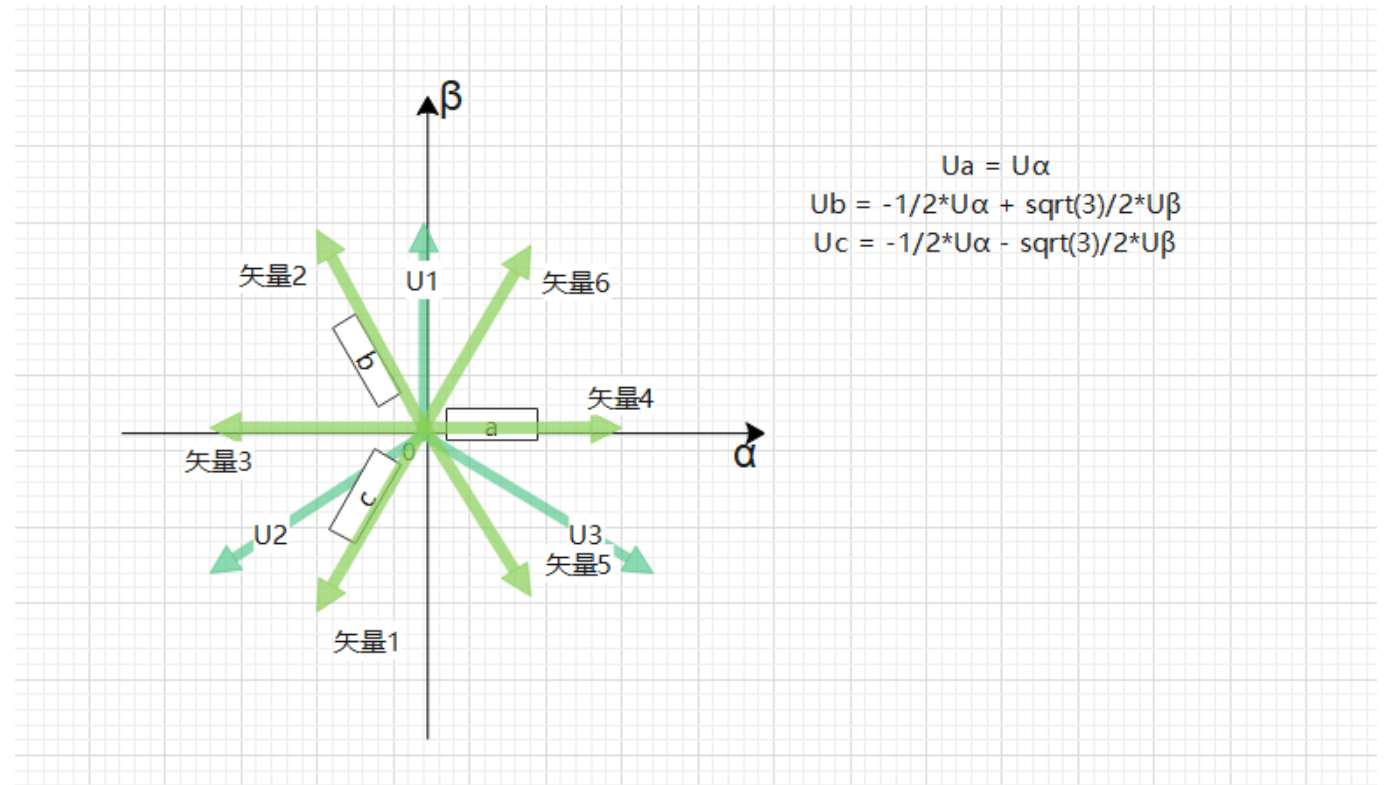
通过计算我们可以得出其他几个扇区的矢量作用时长 由于所有结果格式基本相同因此我们假设: $K = \frac{\sqrt{3}}{2} * \frac{T_s}{U_{dc}}$ $U_1 = U_{\beta}$ $U_2 = -\frac{\sqrt{3}}{2} * U_{\alpha} - U_{\beta}/2$ $U_3 = \frac{\sqrt{3}}{2} * U_{\alpha} - U_{\beta}/2$ 则可以总结出下表:

扇区	T0	T1	T2	T3	T4	T5	T6	T7
1	$(T_s - T_4 - T_6)/2$	0	0	0	$K * U_3$	0	$K * U_1$	$(T_s - T_4 - T_6)/2$
2	$(T_s - T_2 - T_6)/2$	0	$-K * U_3$	0	0	0	$-K * U_3$	$(T_s - T_2 - T_6)/2$
3	$(T_s - T_2 - T_3)/2$	0	$K * U_1$	$K * U_2$	0	0	0	$(T_s - T_2 - T_3)/2$
4	$(T_s - T_1 - T_3)/2$	$-K * U_1$	0	$-K * U_3$	0	0	0	$(T_s - T_1 - T_3)/2$
5	$(T_s - T_1 - T_5)/2$	$K * U_2$	0	0	0	$K * U_3$	0	$(T_s - T_1 - T_5)/2$
6	$(T_s - T_4 - T_5)/2$	0	0	0	$-K * U_2$	$-K * U_1$	0	$(T_s - T_4 - T_5)/2$

得出上表之后我们就可以通过8个矢量来产生任意大小和任意方向的矢量了。通过表格发现不同扇区用到的矢量不一样，因此我们需要做一下扇区判断，之前讲过扇区判断有3种方式，其中电流检测在这里讲解。

利用电流做扇区判断

观察反Clark变换公式和我们之前 U_1, U_2, U_3 可以发现，这两个公式极其相像。



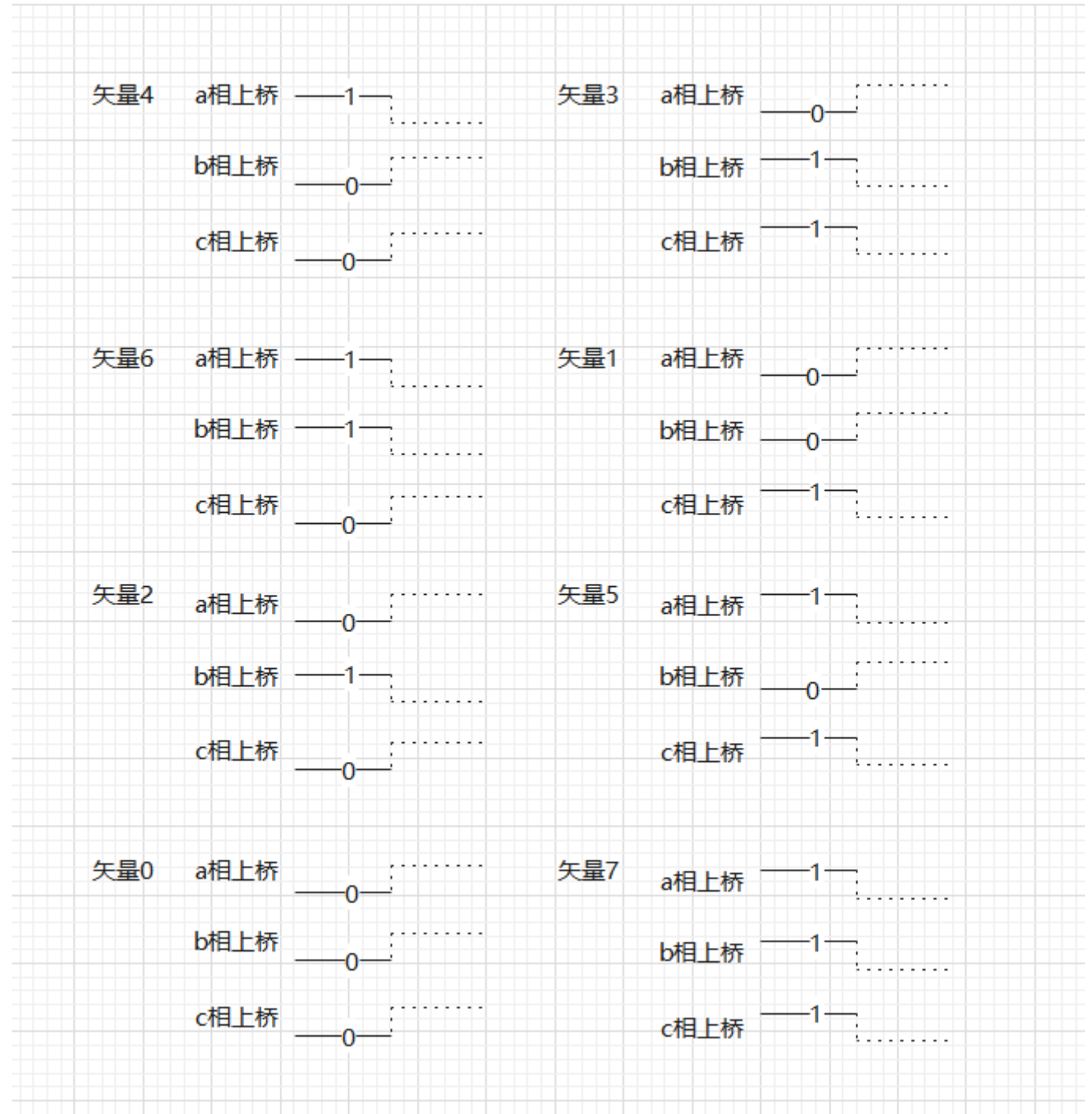
由图可以看出我们能够使用U1 U2 U3的正负来判断当前所处扇区 设 U1>0 时 A=1, 否则 A=0； 设 U2>0 时 B=1, 否则 B=0； 设 U3>0 时 C=1, 否则 C=0； 设 N=4C+2B+A。

扇区	U1>0 (A)	U 2>0 (B)	U3>0 (C)	N
1(0°-60°)	1	0	1	5
2(60°-120°)	1	0	0	4
3(120°-180°)	1	1	0	6
4(180°-240°)	0	1	0	4
5(240°-300°)	0	1	1	3
6(300°-0°)	0	0	1	1

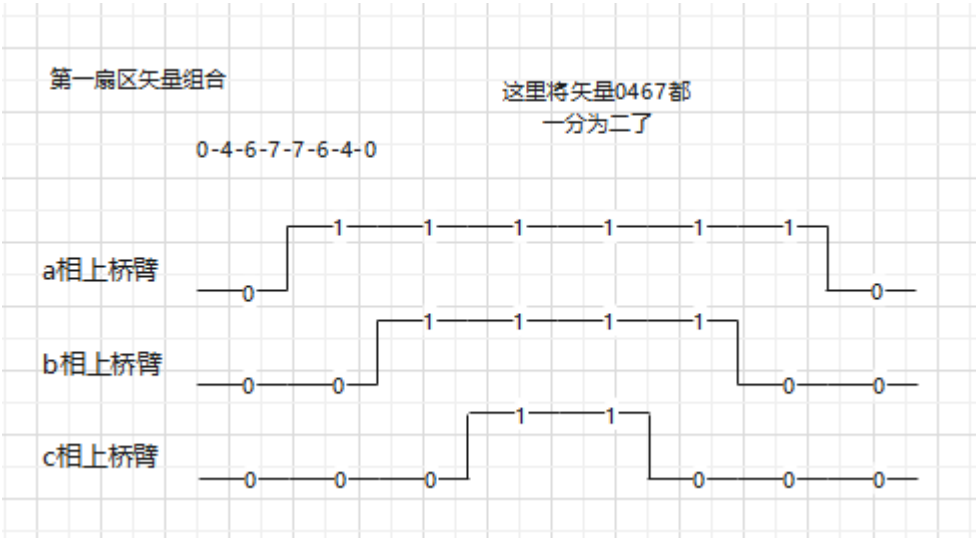
既然我们已经拿到了每个矢量的作用时长，我们便可以求出每个Mos的导通时间，接下来便是把这些导通时间兑换成PWM形式。

七段式SVPWM

我们已知任意一个方向和大小的矢量需要的8个矢量的作用市场，那么如何控制硬件电路实现呢，我们熟悉的硬件电路控制是PWM模式，那么有办法把矢量作用时长换成Mos的导通时间吗，当然可以，首先我们看一下每个矢量对应的MOS导通时间，然后通过组合把这些时间组成PWM模式。



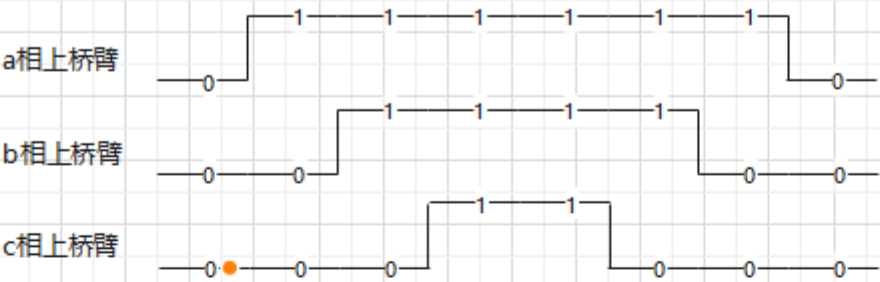
接下来我们要做的是把这些矢量信号排列组合成PWM信号例如在第一个扇区我们用到了矢量T0 T4 T6 T7



按照同样方法便可分析出所有扇区对应的PWM信号

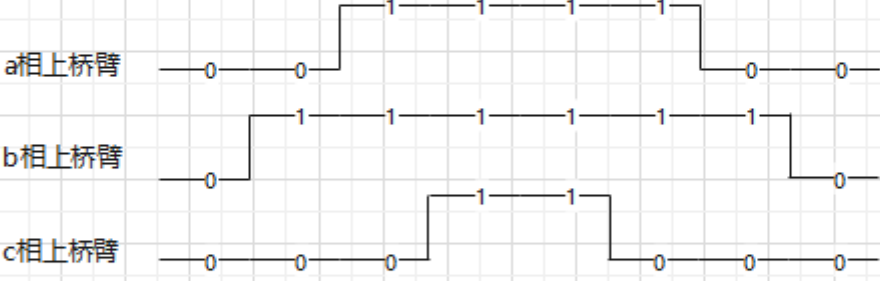
第一扇区矢量组合

0-4-6-7-7-6-4-0



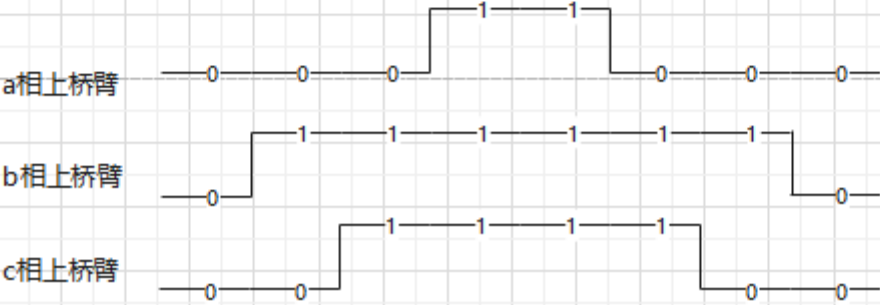
第二扇区矢量组合

0-2-6-7-7-6-2-0



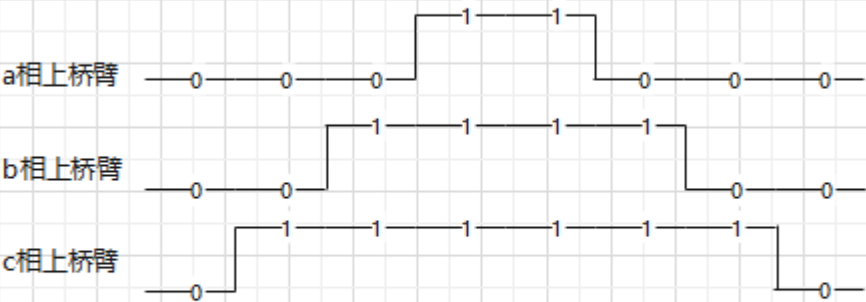
第三扇区矢量组合

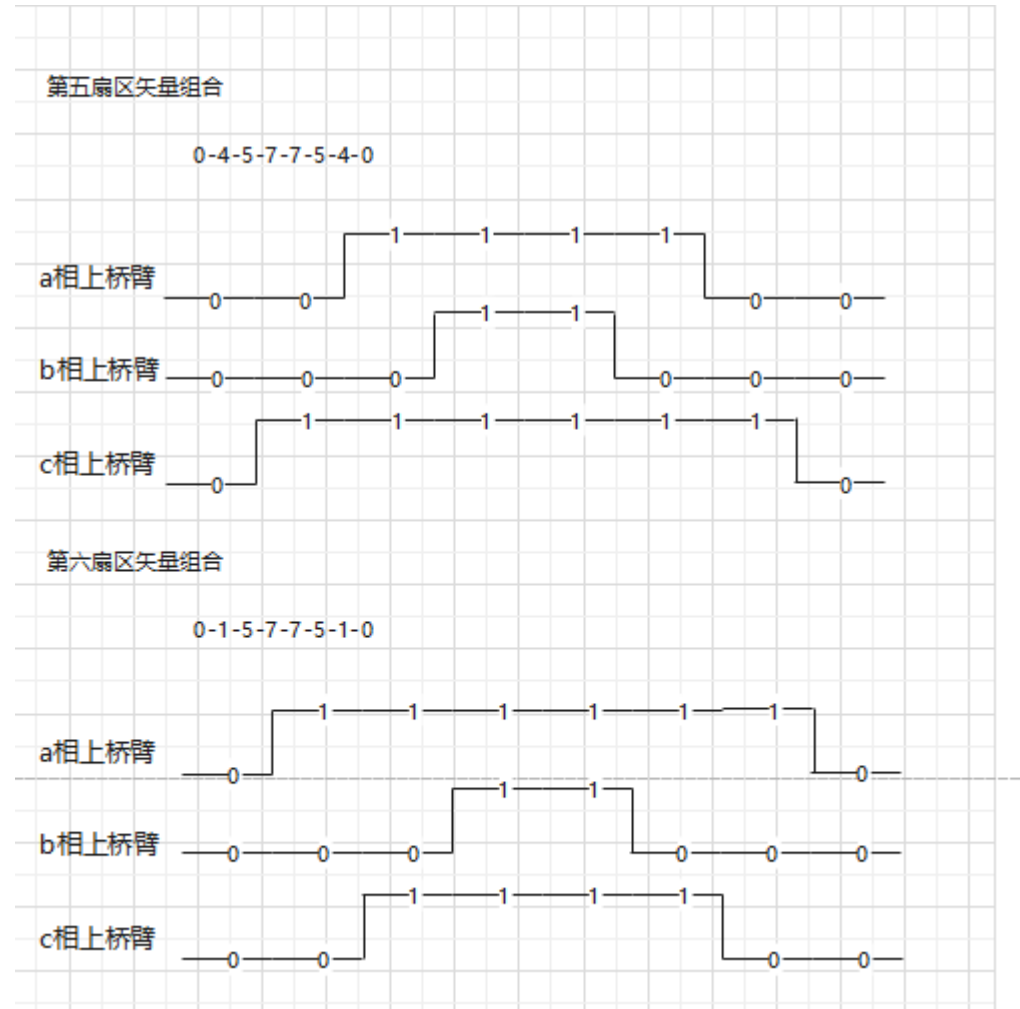
0-2-3-7-7-3-2-0



第四扇区矢量组合

0-1-3-7-7-3-1-0





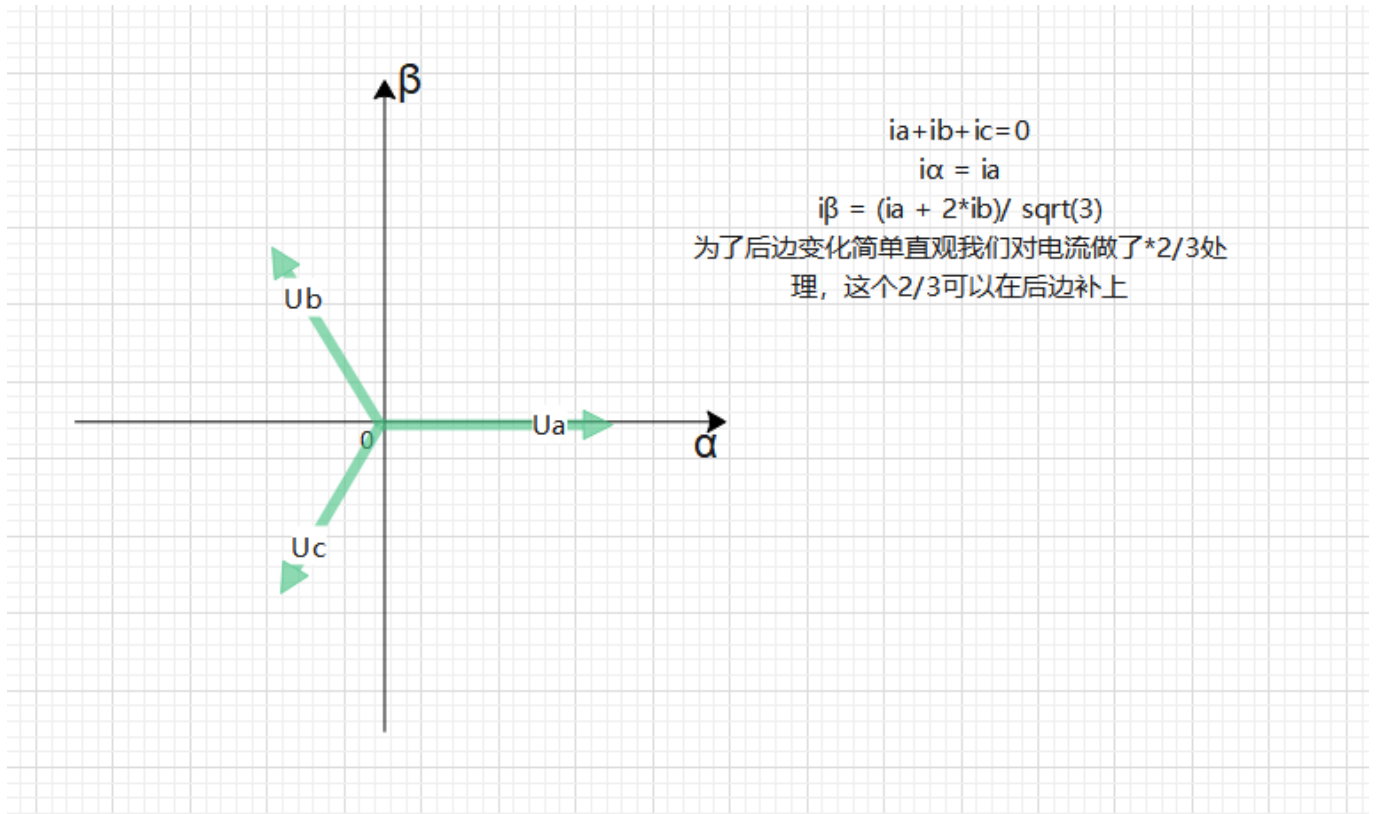
这样的波形不就是我们熟悉的中央对其方式的PWM波吗。到此我们已经完成了FOC开环运行。

程序编写

第五章 FOC电流闭环

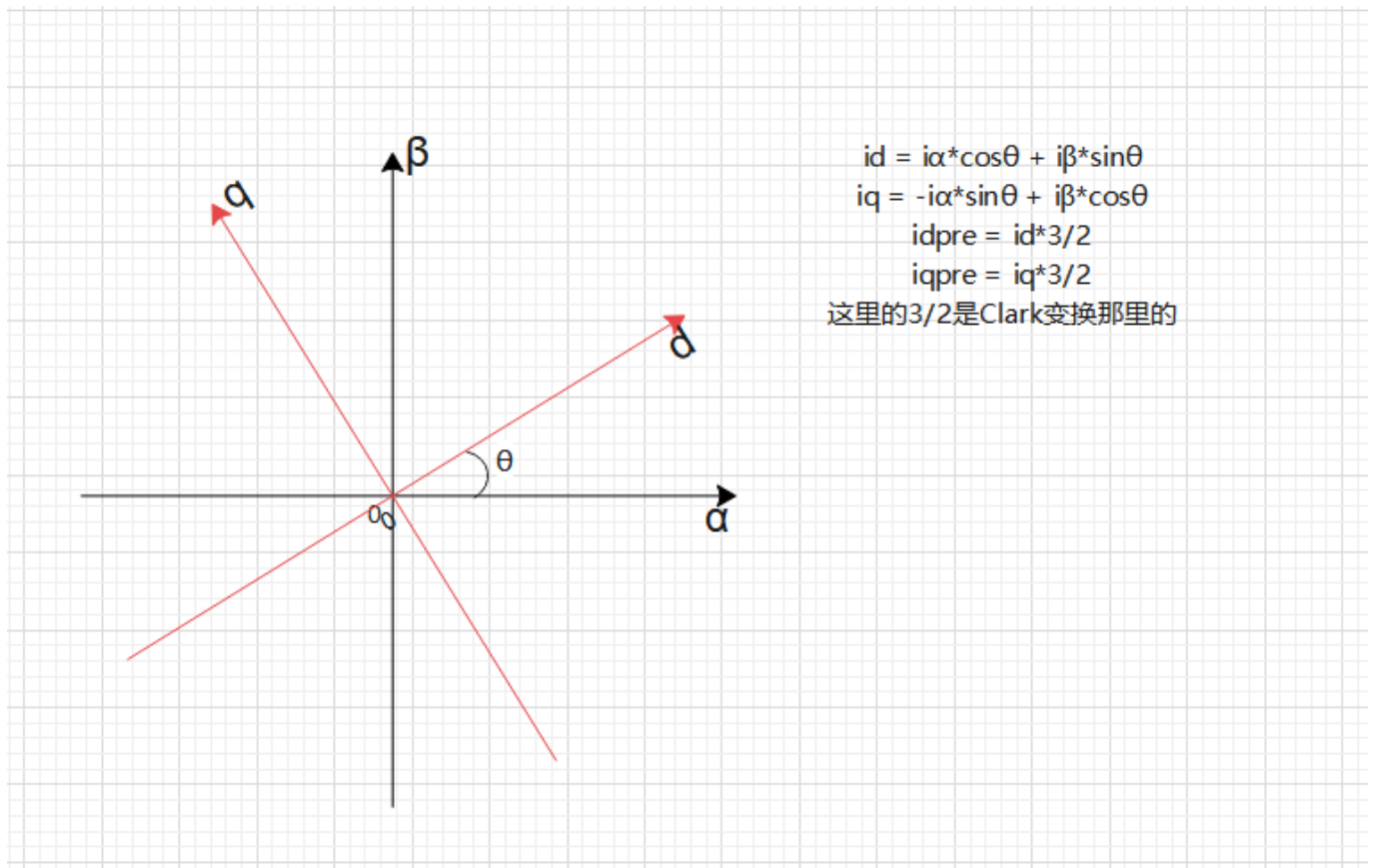
其实到这里我想大家应该已经明白Clark变换和Park变换的作用了，我们拉开视角再看我们刚刚开环做了什么。
1.给出目标电流 2.通过定时器控制三相全桥电路输出这个目标电流。（不用纠结电压电流，因为它们之间只是个系数问题）那么我们要闭环岂不是只需要检测出实际电流然后做个PID闭环即可。那么为什么要用到Clark变换和Park变换呢，因为我们如果不用这些变换那么我们的目标电流和实际电流都是正弦波，这对我们PID控制很不利，因为我们熟悉的PID是目标值不变，让实际值去跟随目标值。而且我们拿到的三相电流 I_a I_b I_c 不直观， I_α 和 I_β 也不直观，而 I_q 和 I_d 就直观的多， I_d 用于发热， I_q 用于旋转。对 I_q 和 I_d 进行闭环控制通俗易懂还能配合速度环，因为想要转的快就增加 I_q 呗。

Clark变换



Clark变换的目的在于三相变两相。

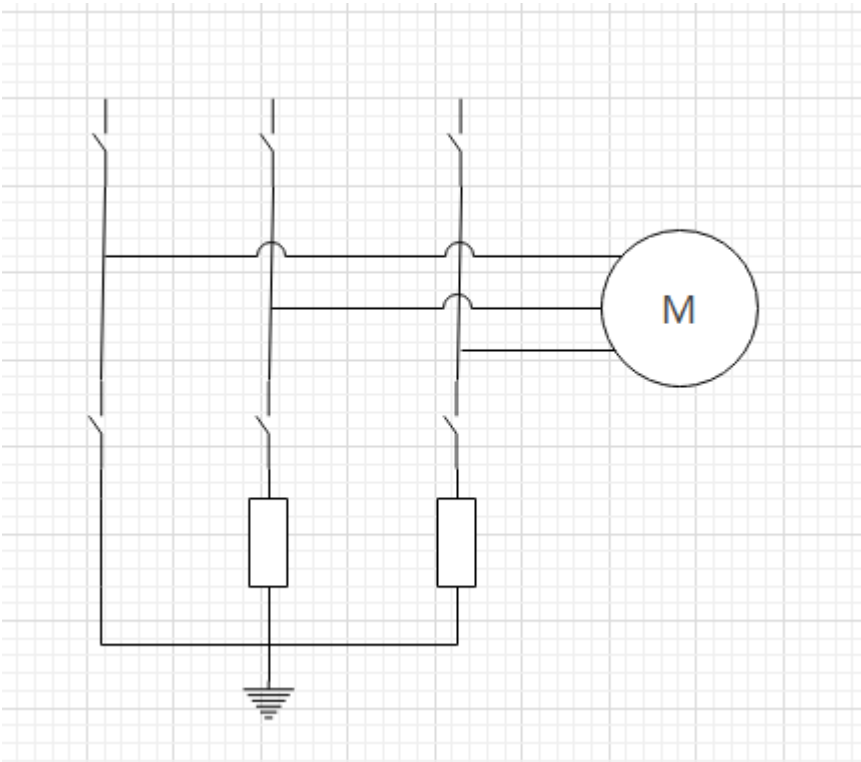
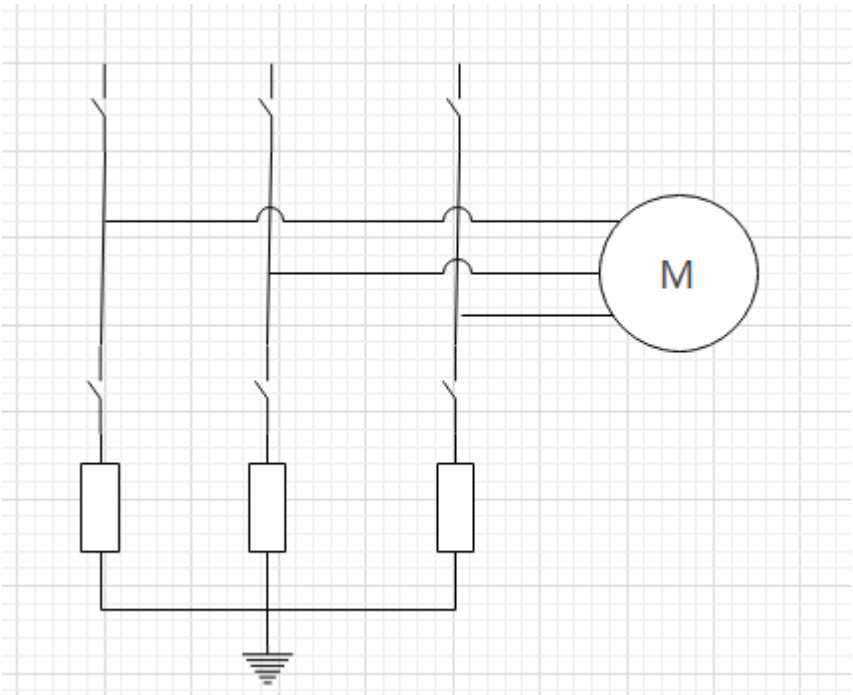
Park变换

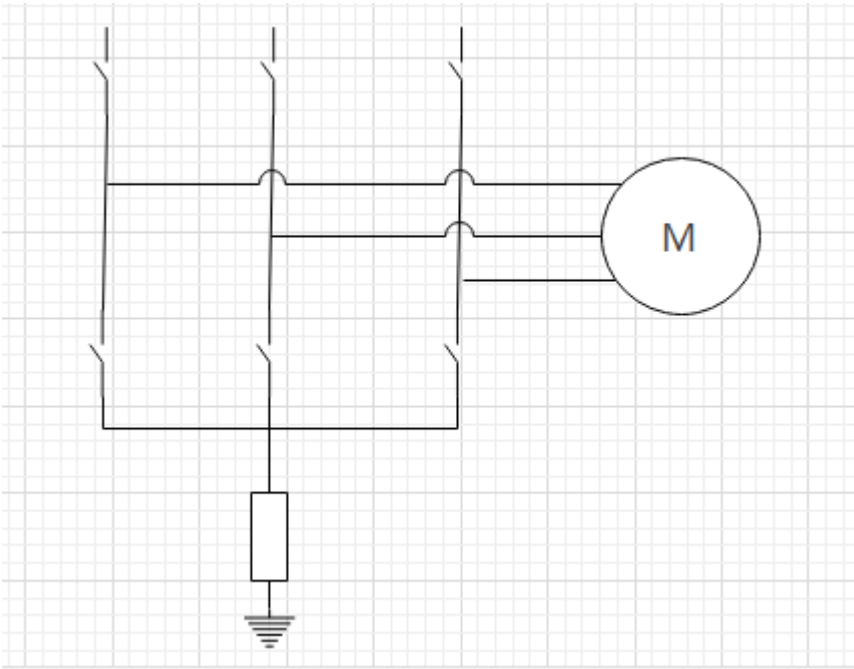


Park变换的目的在于静止变旋转。

电流采样

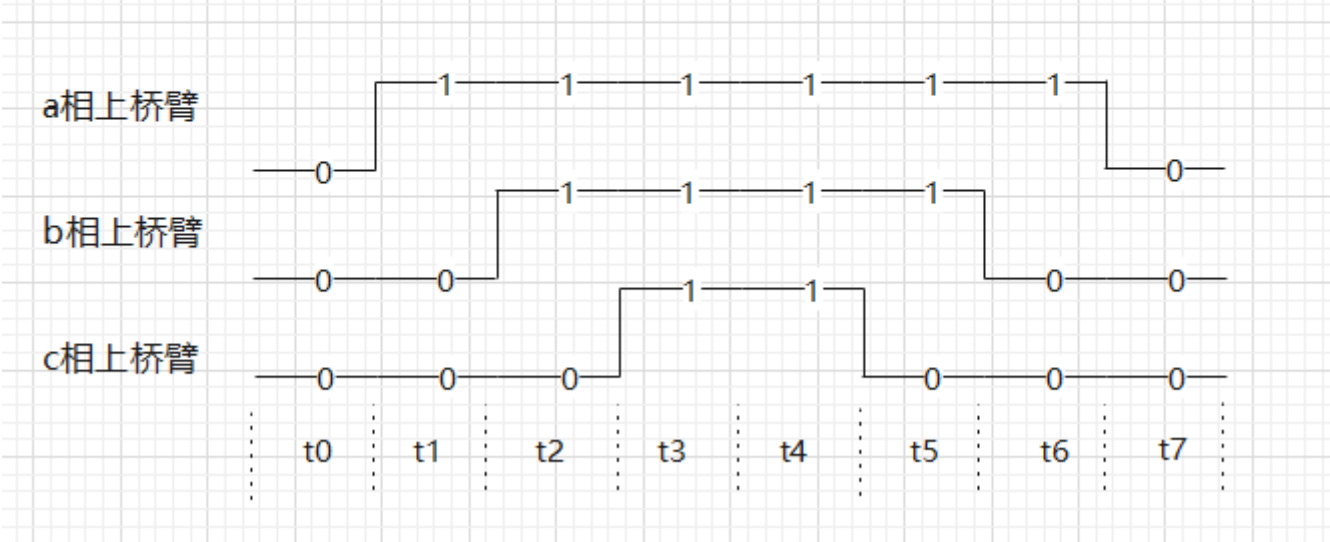
经过变换之后我们就要面临电流采样问题了。通过上一章对7段式SVPWM的分析我们发现在一个周期内，不同时刻Mos的导通是不一样的，一般采样电阻都是接在下桥臂，因此只能在相关桥壁的下MOS管打开时刻进行该相的电流采样。



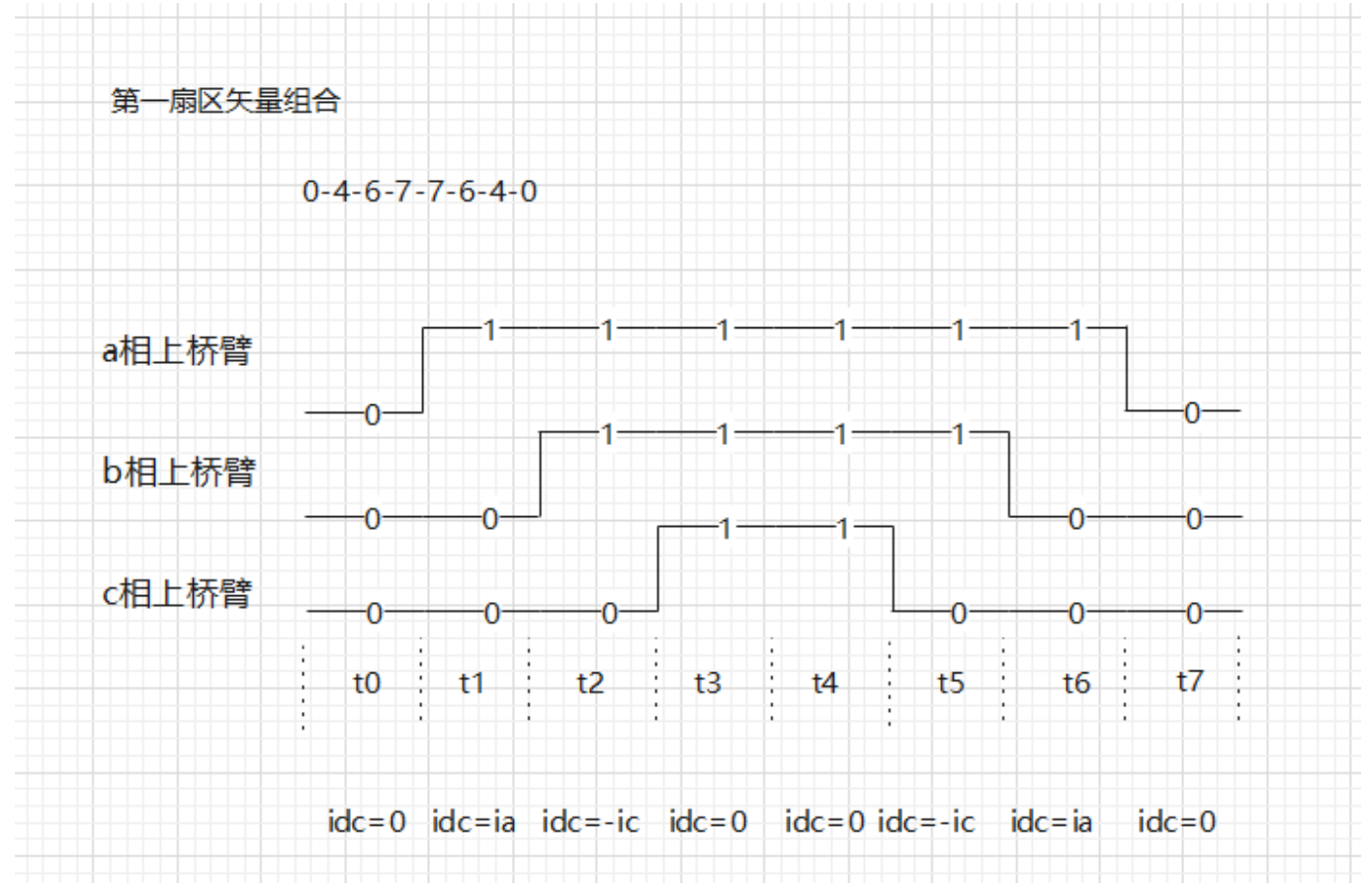


第一扇区矢量组合

0-4-6-7-7-6-4-0



由电路图可知 1.必须在三相全桥的下桥臂打开的时候才能进行该相电流采样。 2.Mos的开关瞬间会对相电流产生干扰，因此采样时间避开Mos导通状态切换的时刻。 3.电阻可以在任意时刻进行电流采样 4.根据SVPWM波形发现每个扇区都有两相电流变化较小，一相电流变化较大，因此三电阻采样时要采集电流变化较小的两相。 5.由于当目标电流较大时，可能会存在一相的占空比接近100%，则导致该相电流无法采样，因此双电阻采样要对最大输出电流做限制，流出电流采样时间。 6.单电阻要在一个周期内采样两次电流。



我们可以在t5和t6时刻采样到 $-i_c$ 和 i_a 电流完成电流采样

程序编写

第六章 速度闭环

相信大家到这里已经较为清楚了，速度环，目标是给定速度，反馈是实际速度，输出是 I_d I_q ，其中 I_d 为0， I_q 越大转的越快。

程序编写

第七章 位置闭环

位置环同理，目标是给定位置，反馈是实际位置，输出是目标速度。

程序编写