

## 5.3 同步电机矢量控制调速仿真

通过坐标变换，把同步电动机等效成直流电动机，再模仿直流电动机的控制方法进行控制。在同步电动机矢量控制系统中，为了准确地定向，需要检测转子位置。因此，同步电动机矢量控制变频调速也可归属于自控变频同步电动机调速系统。

### 5.3.1 可控励磁同步电动机动态数学模型

▲和异步电机类似，作如下假定：

(1) 忽略空间谐波，设定子三相绕组对称，所产生的磁动势沿气隙按正弦规律分布；

(2) 忽略磁路饱和，各绕组的自感和互感都是恒定的；

(3) 忽略铁心损耗；

(4) 不考虑频率变化和温度变化对绕组电阻的影响。

又因为定子三相绕组 $A$ 、 $B$ 、 $C$ 是静止的，转子以角速度 $\omega$ 旋转，转子上的励磁绕组在励磁电压 $U_f$ 供电下流过励磁电流 $I_f$ 。沿励磁磁极的轴线为 $d$ 轴，与 $d$ 轴正交的是 $q$ 轴， $dq$ 坐标系固定在转子上，与转子同步旋转。阻尼绕组是多导条类似笼型的绕组，等效成在 $d$ 轴和 $q$ 轴各自短路的两个独立的绕组。

▲考虑到凸极效应和阻尼绕组，同步电动机的定子电压方程为：

$$\begin{cases} u_A = R_s i_A + \frac{d\psi_A}{dt} \\ u_B = R_s i_B + \frac{d\psi_B}{dt} \\ u_C = R_s i_C + \frac{d\psi_C}{dt} \end{cases}$$

式中： $R_s$ ——定子电阻

$\psi_A$ 、 $\psi_B$ 、 $\psi_C$ ——三相定子磁链

▲转子电压方程为：

$$\begin{cases} U_f = R_f I_f + \frac{d\psi_f}{dt} \\ 0 = R_{rd} i_{rd} + \frac{d\psi_{rd}}{dt} \\ 0 = R_{rq} i_{rq} + \frac{d\psi_{rq}}{dt} \end{cases}$$

按照坐标变换原理，将定子电压方程从 ABC 三相坐标系变换到 dq 两相旋转坐标系。得到新的定子电压方程为：

$$\begin{aligned} u_{sd} &= R_s i_{sd} + \frac{d\psi_{sd}}{dt} - \omega \psi_{sq} \\ u_{sq} &= R_s i_{sq} + \frac{d\psi_{sq}}{dt} + \omega \psi_{sd} \end{aligned}$$

▲在dq两相旋转坐标系上的磁链方程为：

$$\begin{cases} \psi_{sd} = L_{sd} i_{sd} + L_{md} I_f + L_{md} i_{rd} \\ \psi_{sq} = L_{sq} i_{sq} + L_{mq} i_{rq} \\ \psi_f = L_{md} i_{sd} + L_f I_f + L_{md} i_{rd} \\ \psi_{rd} = L_{md} i_{sd} + L_{md} I_f + L_{rd} i_{rd} \\ \psi_{rq} = L_{mq} i_{sq} + L_{rq} i_{rq} \end{cases}$$

▲电磁转矩方程整理后得：

$$T_e = n_p L_{md} I_f i_{sq} + n_p (L_{sd} - L_{sq}) i_{sd} i_{sq} + n_p (L_{md} i_{rd} i_{sq} - L_{mq} i_{rq} i_{sd})$$

### 5.3.2 PMSM 矢量控制系统

正弦波永磁同步电动机具有定子三相分布绕组和永磁转子，在磁路结构和绕组分布上保证定子绕组中的感应电动势具有正弦波形，外施定子电压和电流也应为正弦波。

假想转子由一般导磁材料构成，带有一个虚拟的励磁绕组，通以虚拟的励磁电流，产生的转子磁动势与永磁同步电动机的转子磁动势相等。

永磁同步电动机可以与一般的电励磁同步电动机等效，唯一的差别是虚拟励磁电流恒定且永磁同步电动机具有幅值恒定、方向随转子位置变化的转子磁动势。

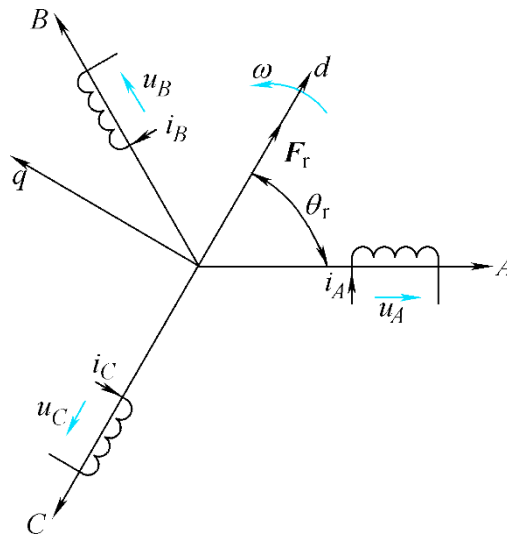


图 5.13 永磁同步电动机物理模型

▲  $I_f = \text{常数}$ ,  $\frac{dI_f}{dt} = 0$ ; 其中  $L_f$  —— 虚拟励磁绕组等效电感

由前面数学模型公式可以知道, 因为  $\psi_f$  保持恒定, 过控制 d、q 轴定子电流分量  $i_d$ 、 $i_q$  就可以有效控制同步电机的电磁转矩  $T_e$ 。电机稳态运行时,  $\psi_d$ 、 $\psi_q$  可看作为常数。在基频以下调速时, 若控制定子电流矢量使之落在 q 轴上, 即  $i_d = 0$ 、 $i_q = i_s$ , 并将定子磁链方程带入定子电压方程可得:

$$\begin{cases} \psi_d = \psi_f \\ \psi_q = L_d i_s \\ \begin{cases} U_d = -\omega L_q i_s \\ U_q = R_s i_s + L_q s i_s + \omega \psi_f \end{cases} \end{cases}$$

电磁转矩方程为:

$$T_e = n_p \psi_f i_s$$

运动方程为:

$$T_e - T_L = \frac{J}{n_p} \cdot \frac{d\omega}{dt}$$

由于在控制的过程中始终保持 d 轴定子电流分量  $i_d = 0$  恒定, 而仅对 q 轴定子电流分量  $i_q$  进行控制, 就可以控制电机电磁转矩, 从而获得较好的静、动态, 故也称  $i_d = 0$  控制。

### 1、电流滞环控制

电流滞环跟踪调制矢量控制系统结构如图 5.14 所示。给定转速  $n^*$  与转速实际值  $n$  的偏差值  $n^* - n$  经转速调节器 ASR 的 PI 环节后, 输出与电磁转矩  $T_e$  成正比的 q 轴定子电流分量给定值  $i_q^*$ ,  $i_q^*$  与 d 轴定子电流分量给定值  $i_d^*$  ( $i_d^* = 0$ ) 经过  $2r/3s$  变换后得到三相电流理想给定值  $i_{abc}^*$ , 此时电流滞环比较器根据理想电流给定值  $i_{abc}^*$  与实际电流  $i_{abc}$  进行滞环比较, 得到逆变桥的驱动 PWM 信号, 将信号给逆变桥, 驱动电机运行。

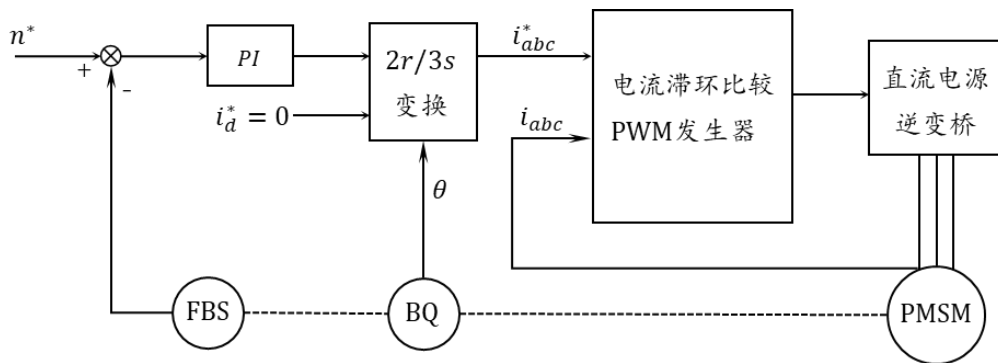


图 5.14 电流滞环跟踪调制矢量控制系统原理

## 2、SVPWM 控制

同步电机的电流滞环跟踪矢量控制系统原理简单，鲁棒性好；但逆变器开关频率不固定，会随电机运行状况发生变化。采用 SVPWM 方式可使逆变器具有固定的开关频率且与异步电机类似，可以减少纹波电流。

另外，定子为三相对称正弦电压时，三相合成矢量幅值： $U_s = \sqrt{\frac{3}{2}} U_m$

由 SVPWM 控制得到  $\begin{cases} \text{基波相电压最大幅值: } U_{mmax} = \sqrt{\frac{2}{3}} U_{smax} = \frac{U_d}{\sqrt{3}} \\ \text{基波线电压最大幅值: } U_{lmmax} = \sqrt{3} U_{mmax} = U_d \end{cases}$

两者之比：

$$\frac{U_{lmmax}}{U'_{lmmax}} \approx 1.15$$

所以 SVPWM 方式的逆变器输出线电压基波最大值为直流侧电压，比 SPWM 逆变器输出电压最多提高了约 15%。

▲因此对于永磁同步电机来说，其矢量控制系统结构如图 5.15 所示：

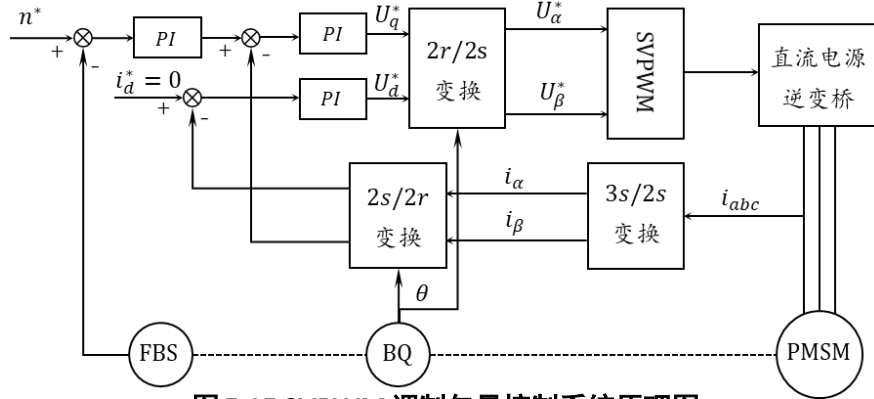


图 5.15 SVPWM 调制矢量控制系统原理图

给定转速  $n^*$  与转速实际值  $n$  的偏差值  $n^* - n$  经转速调节器 ASR 的 PI 环节后，输出 q 轴定子电流分量给定值  $i_q^*$ ，再与实际 q 轴电流分量实际值  $i_q$  作差，偏差值  $i_q^* - i_q$  经电流调节器的 PI 环节后，输出 q 轴分量的给定电压  $U_q^*$ 。d 轴定子电流给定分量  $i_d^* (i_d^* = 0)$  与 d 轴定子电流分量实际值  $i_d$  作差，其偏差值经过电流 PI 调节器后，输出 d 轴定子电压分量的给定值  $U_d^*$ 。

其中 d、q 轴定子电流分量的实际值  $i_d$ 、 $i_q$  可以由测量得到的同步电机三相定子电流  $i_{abc}$  经过 Clark 及 Park 变换得到。 $U_d^*$  与  $U_q^*$  经过  $2r/2s$  变换后得到定子电压在  $\alpha$ 、 $\beta$  轴上面的分量给定值  $U_\alpha^*$ 、 $U_\beta^*$ 。再将  $U_\alpha^*$  与  $U_\beta^*$  经过 SVPWM 调制后得到 PWM 信号。从而来控制晶闸管通断，来控制电机运行。

### 5.3.3 Simulink 模型的搭建

电流滞环控制模型如下图 5.16 所示:

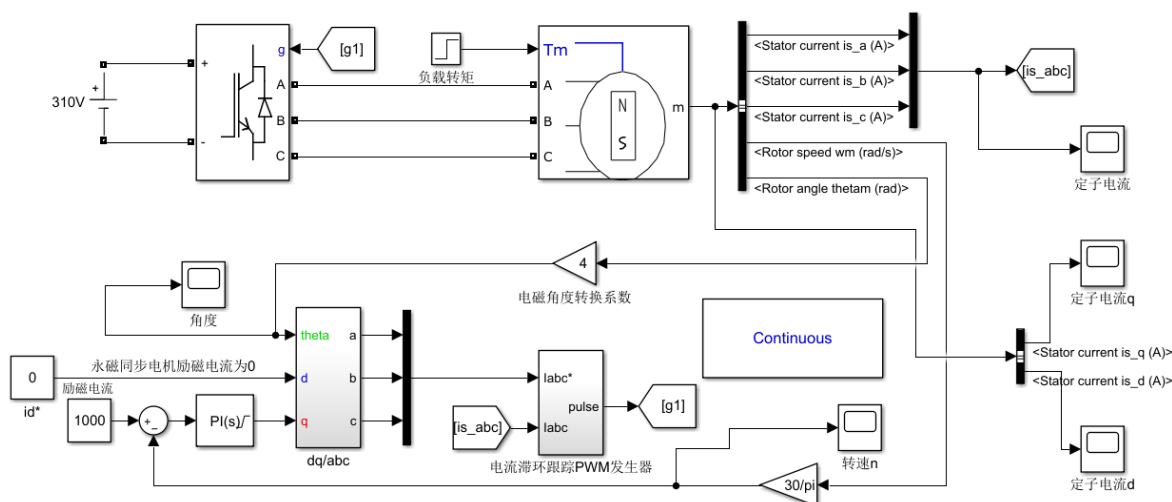


图 5.16 同步电机电流滞环控制模型

其中转速调节器的 $PI$ 设置为:  $K_p = 0.05$ ,  $K_i = 8$ 。原因是该 $PI$ 环节是由设定转速 $n^*$ 与实际反馈转速 $n$ 的差 $n^* - n$ 作为输入, 而输出是给定理想电流的 $q$ 轴分量 $i_q^*$ 。又因为可以通过前面的模拟知道 $i_q^* \approx 10 \sim 30$ 的样子, 而转速输出差一开始就可以达到1000, 作为输入量来说太大了, 因此在设置 $PI$ 的时候, 可以考虑将 $K_p$ 尽量设置小一点, 可以看到 $K_p = 0.05$ 的时候, 我们的开始瞬间输入为 $1000 \times 0.05 = 50$ 左右, 已经很大了, 大于实际电机运行时电流的 $q$ 轴分量 $i_q^*$ , 也不是大很多, 较为合理。在设置 $K_i$ 的时候, 考虑到 $PI$ 调节器的输出理想约为 $i_q^* \approx 10 \sim 30$ 的样子, 那么在合理时间内, 将 $PI$ 环节的输出值(积分与比例环节的和)稳定在 $10 \sim 30$ , 那么综合考虑可将 $K_i$ 设置为8。

在设置同步电机的时候，在其 ‘Advance’ 中 ‘Rotor flux position when theta = 0’ 该选项应该选择 ‘Aligned with phase A axis’，即初始 0 角度应该和 A 轴重合。此外，该角度输出值是机械角度，那么因为在设置同步电机极对数 (Parameters\_p) 的时候，设定为  $n_p = 4$ ，即 4 对磁极，且因为 Park 变换/Park 逆变换都需要用到的是电磁角度，因此这里需要乘以一个磁极对数的系数，才能将机械角度转换为电磁角度（在异步电机中同理：需要将测量的机械角速度  $\omega_m$  转换为电磁角速度  $\omega$  才能参与接下来的角速度—角度计算）。

### 5.3.4 SVPWM 的实现

搭建 Simulink 的 SVPWM 控制模型大致分为： $\alpha\beta$ 变换、 $U_{ref}$ 判断扇区及计算 PWM 占空比、输出 PWM 信号几个流程。

#### (1) $\alpha\beta$ 变换

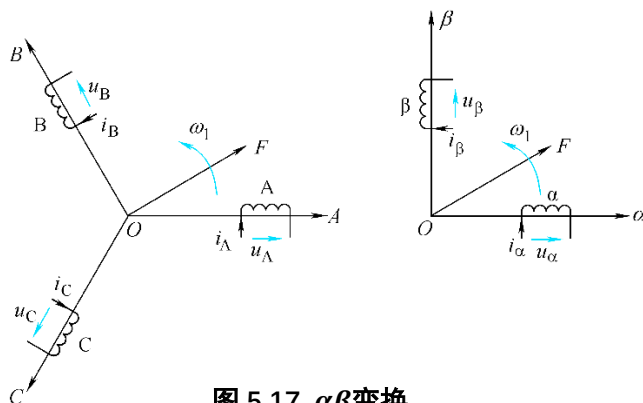


图 5.17  $\alpha\beta$ 变换

三相绕组 A、B、C 和两相绕组之间的变换，称作三相坐标系和两相正交坐标系间的变换，简称3/2变换。按照磁动势相等的等效原则；变换前后总功率不变。

三相坐标系变换到两相正交坐标系（简称 3/2 变换）的变换矩阵：

$$\begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} = \frac{N_3}{N_2} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} i_A \\ i_B \\ i_C \end{bmatrix}$$

由此矩阵搭建如图 5.18 所示 Simulink 模型：

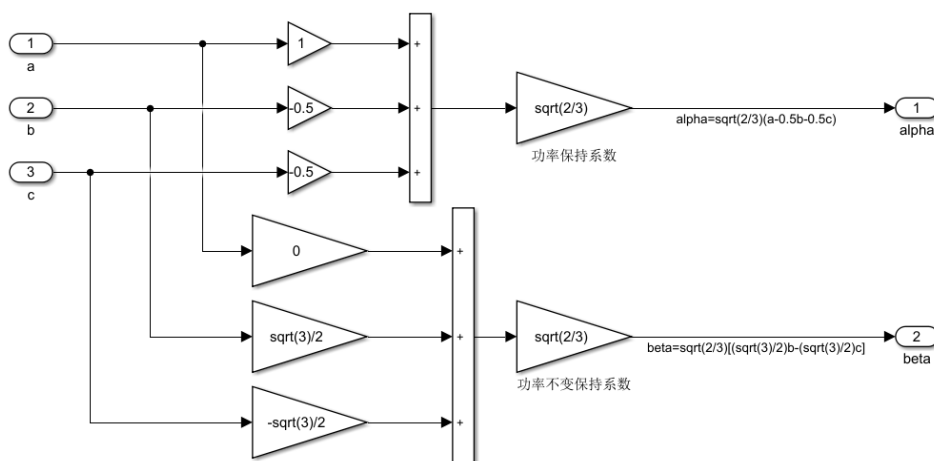


图 5.18 Clark 变换 Simulink 模型

▲也可以得到 Clark 逆变换矩阵：

$$\begin{bmatrix} i_A \\ i_B \\ i_C \end{bmatrix} = \frac{N_3}{N_2} \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ \frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix}$$

由上可以搭建如图 5.19 所示 Clark 逆变换 Simulink 模型：

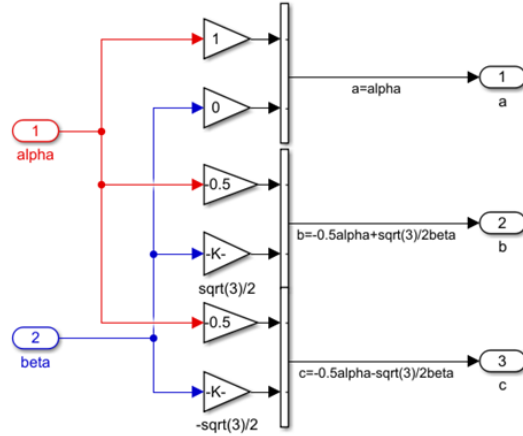


图 5.19 Clark 逆变换 Simulink 模型

## (2) $U_{ref}$ 与扇区判断

如图 5.20 所示，其中 $U_{ref}$ 是期望电压矢量的模，角度 $\theta$ 是它与 $\alpha$ 轴的夹角。

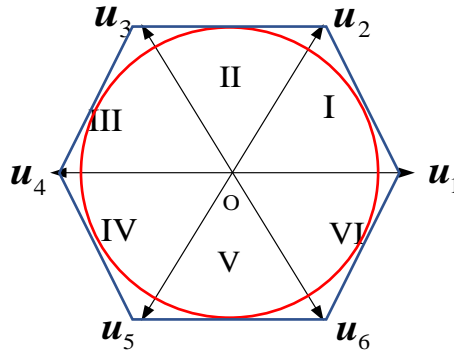


图 5.20 电压空间矢量的 6 个扇区

又因为：

$$U_{ref} = \frac{mU_d}{\sqrt{3}}$$

其中 $m$ 是调制系数， $U_d$ 是逆变器主电路的直流母线电压。当 $m > 1$ 时会过调制，波形失真。此时 $U_{ref}$ 超出了正六边形的内切圆范围。而 $U_{ref}$ 的最大值就是非零电压矢量的值，即：

$$U_{ref} = \frac{2U_d}{3}$$

$\alpha$ 轴即实轴 $Re$ 与 $U_1$ 参考位置重合（如图 5.21 所示）； $\beta$ 轴即虚轴。

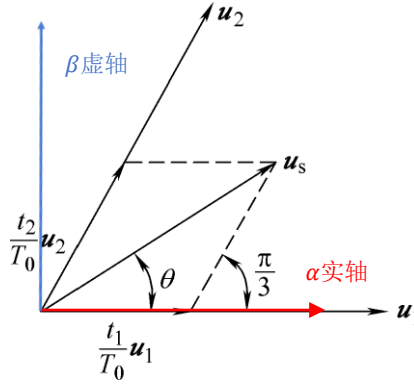


图 5.21 期望输出电压矢量的合成

由此可以定义以下参考变量：

$$\begin{cases} U_{ref1} = U_\beta \\ U_{ref2} = \frac{\sqrt{3}}{2} U_\alpha - \frac{1}{2} U_\beta \\ U_{ref3} = -\frac{\sqrt{3}}{2} U_\alpha - \frac{1}{2} U_\beta \end{cases}$$

再定义：
 
$$\begin{cases} \text{若 } U_{ref1} > 0, \text{ 则 } A = 1, \text{ 否则 } A = 0 \\ \text{若 } U_{ref2} > 0, \text{ 则 } B = 1, \text{ 否则 } B = 0 \\ \text{若 } U_{ref3} > 0, \text{ 则 } C = 1, \text{ 否则 } C = 0 \end{cases}$$
 ABC之间一共有八种组合。

但由判断扇区公式可知道 A、B、C 不会同时为 1 或者 0，所以组合其实是 6 种。A、B、C 取不同的组合对应着不同的扇区，并且是一一对应的，因此完全可以由 A、B、C 的组合判断所在的扇区。为区别扇区的六种状态。令： $N=A+2B+4C$  则 N 可以为 1 至 6 六个整数值。具体对应关系如下表 5.3 所示：

表 5.3 扇区值与扇区编号的关系

扇区值N	1	2	3	4	5	6
扇区编号S	II	VI	I	IV	III	V

由上可以搭建如下图 5.22 所示扇区判断的 Simulink 模型：

### (3) 计算 PWM 占空比

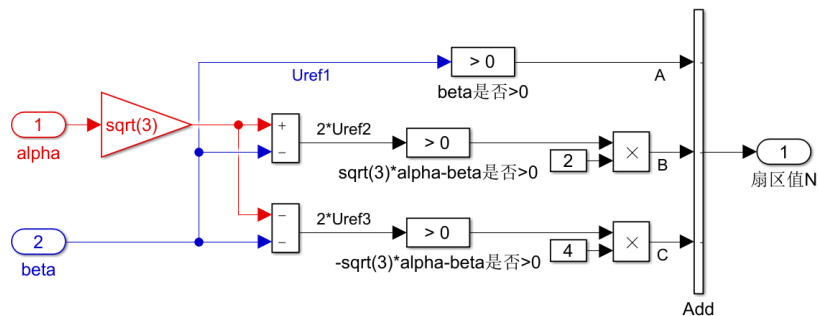


图 5.22 扇区判断 Simulink 模型



SVPWM 的理论基础是平均值等效原理，即在一个开关周期 $T_{PWM} = T_0$ 内通过对基本电压矢量加以组合，使其平均值与给定电压矢量相等。

为了方便判定，可以令：

$$\begin{cases} X = \frac{\sqrt{3}T_0 u_\beta}{U_d} \\ Y = \frac{3T_0 u_\alpha}{2U_d} + \frac{\sqrt{3}T_0 u_\beta}{2U_d} \\ Z = -\frac{3T_0 u_\alpha}{2U_d} + \frac{\sqrt{3}T_0 u_\beta}{2U_d} \end{cases}$$

▲各个扇区 $t_1$ 、 $t_2$ 、 $t_0$ 作用时间如下表 5.4 所示：

表 5.4 各个扇区 $t_1$ 、 $t_2$ 、 $t_0$ 作用时间

扇区值 $N$	1	2	3	4	5	6
$t_1$	Z	Y	-Z	-X	X	-Y
$t_2$	Y	-X	X	Z	-Y	-Z
$t_3$	$T_0 = T_s - t_1 - t_2$					

如果当 $t_1 + t_2 > T_0$ ，必须进行调制（等比例缩小原则）处理，则令：

$$\begin{cases} t_1 = \frac{t_1}{t_1 + t_2} T_0 \\ t_2 = \frac{t_2}{t_1 + t_2} T_0 \end{cases}$$

▲Simulink 实现如下图 5.23 所示：

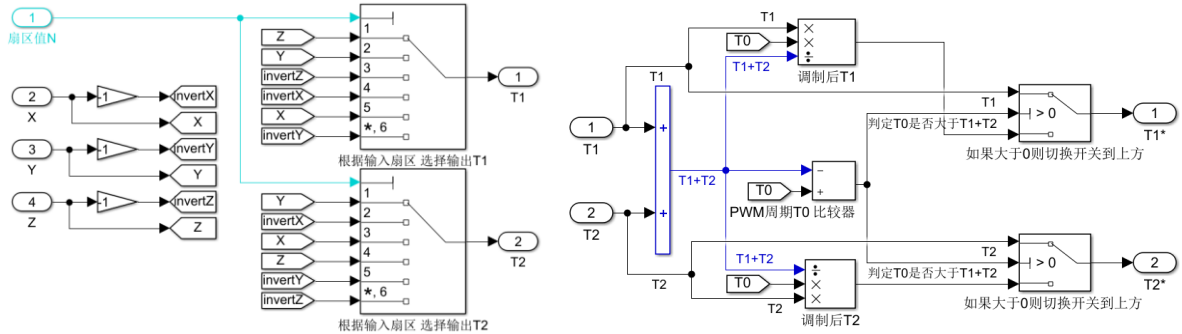


图 5.23 计算各个扇区作用时间  $T1$ 、 $T2$  及判定是否需要调制

#### (4) SVPWM 的控制信号实现

采用零矢量分散法实现，以七步法为例，确定扇区矢量切换点，令：

$$\begin{cases} T_a = (T_0 - t_1 - t_2)/4 \\ T_b = T_a + \frac{t_1}{2} \\ T_c = T_b + \frac{t_2}{2} \end{cases}$$

三相电压开关时间切换点 $T_{cmp1}$ 、 $T_{cmp2}$ 、 $T_{cmp3}$ 与各扇区的关系如下表 5.5 所示：

表 5.5 三相电压开关时间切换点 $T_{cmp1}$ 、 $T_{cmp2}$ 、 $T_{cmp3}$

扇区值N	1	2	3	4	5	6
$T_{cmp1}$	$T_b$	$T_a$	$T_a$	$T_c$	$T_c$	$T_b$
$T_{cmp2}$	$T_a$	$T_c$	$T_b$	$T_b$	$T_a$	$T_c$
$T_{cmp3}$	$T_c$	$T_b$	$T_c$	$T_a$	$T_b$	$T_a$

将三相电压开关时间切换点与调制波进行比较，输出 SVPWM 信号，如图 5.24 所示：

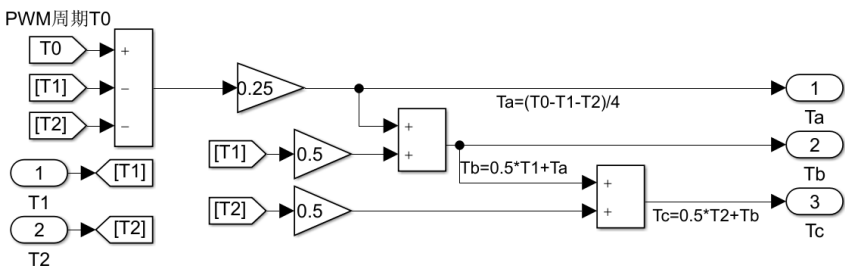


图 5.24 计算各个扇区作用时间 T1、T2 及判定是否需要调制

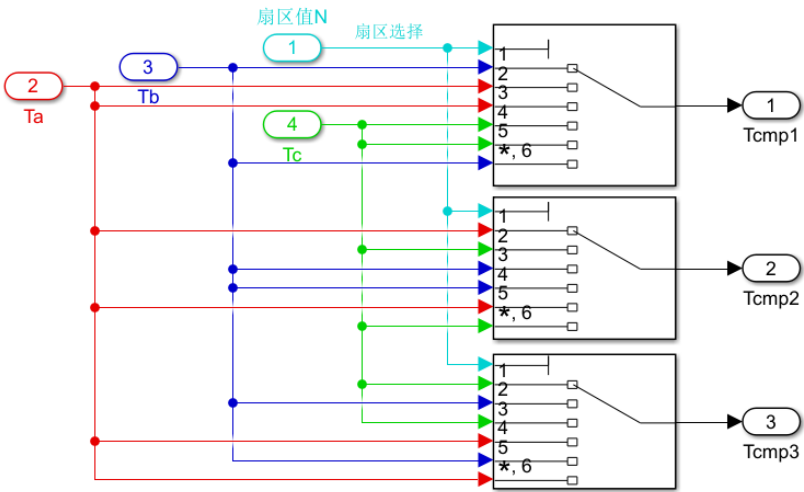


图 5.25 电压开关切换时间信号

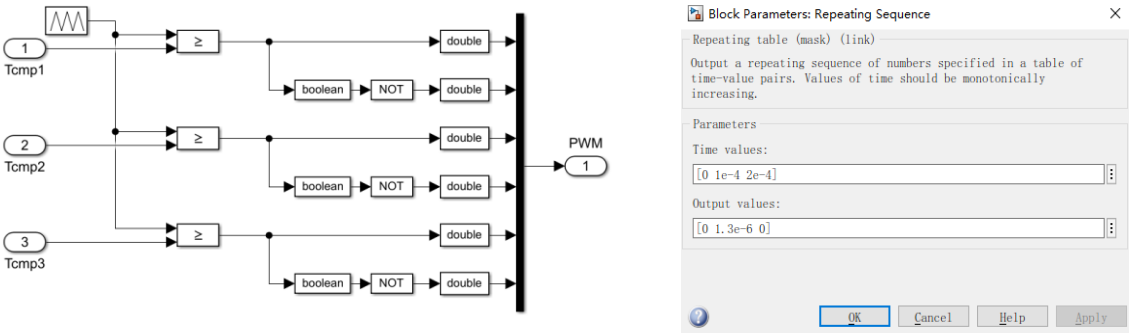


图 5.26 SVPWM 信号的产生

▲总模块如图 5.27 所示：

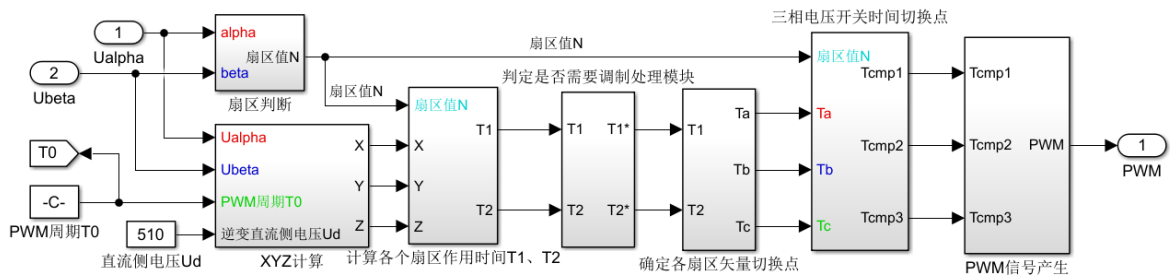


图 5.27 SVPWM 总体模型

### 5.3.5 SVPWM 控制

搭建如图 5.28 所示同步电机 SVPWM 控制模型

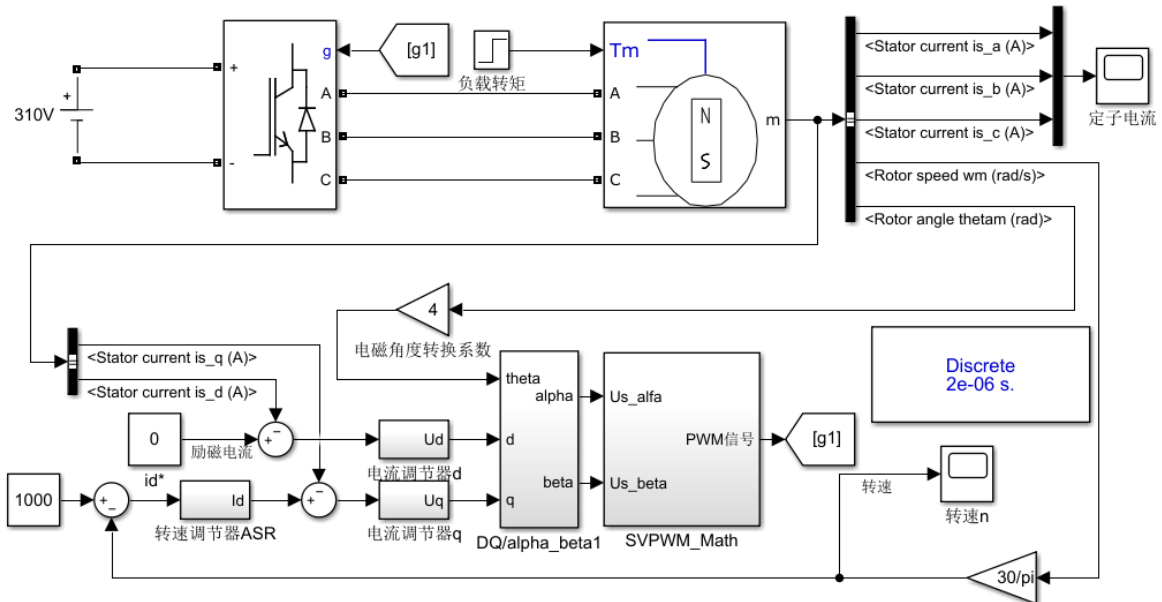


图 5.28 同步电机 SVPWM 控制总体模型

PI 转速调节器参数与电流滞环跟踪调制矢量控制系统相同。2 个 PI 电流调节器参数设计且相同： $K_p = 15$ ， $K_i = 6000$ 。设计思路与前面类似：模型的等效电枢电流  $i_q^*$  的值大约在 10~30 左右，那么因为存在电流负反馈的作用，因此模型的电流调节器的输入值最终归为稳态时应该为 0 左右，而又要保持输出值  $U_d^* \approx 200 \sim 300$  的样子（具体依照负载大小以来确定，可以通过建立开环稳定模型来测量，再反推得到）因此我们的比例环节放大倍数  $K_p$  应该保持在 10~20 左右，另外由于电流可以迅速达到稳定电流，那么意味着我们的动态输入值（ $i_q^* - i_q$ ）会迅速减小，那么为了维持电压输出值  $U_d^* \approx 200 \sim 300$  此时仅能靠积分环节来维持，那么就要求我们的积分环节比较大才行，能够在短时间内积分至我们所需的  $U_d^*$ 。综上可以考虑  $K_i = 6000$ 。

仿真步长设置为：ode23tb。

## 5.3.6 GUI 界面的设计

### 1、主界面的搭建

因为该调速系统有电流滞环控制与 SVPWM 控制之分，因此在 GUI 界面与 Simulink 进行链接时，我们使用了两组反馈（电流滞环控制与 SVPWM 控制作对比）因此可以搭建如图 5.29 所示 GUI 平台界面：

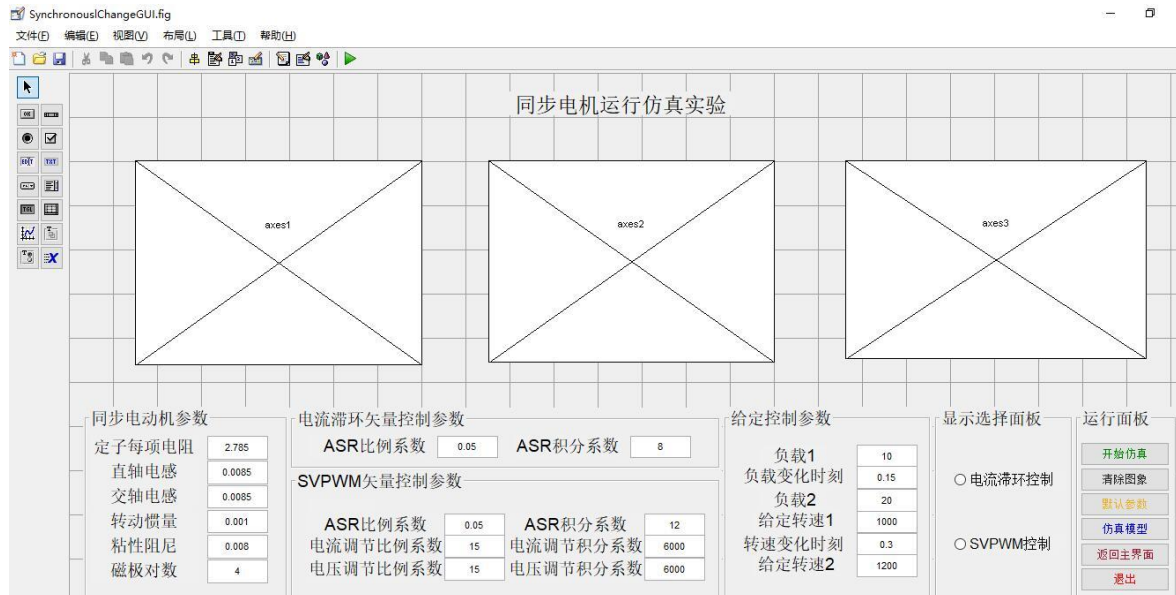


图 5.29 同步电机 GUI 平台界面

上面三张图分别是转速、定子 A 相电流及电磁转矩的变化图。

### 2、回调函数的设置

同理，我们首先完成相似的赋值，再进行运行仿真，因此在 GUI 界面启动模块对应的回调函数中设置如下代码：

```
function pushbutton2_Callback(hObject, eventdata, handles)
if get(hObject,'value')==1
%参数导入
np=str2num(get(handles.np,'String'));%磁极对数
F=str2num(get(handles.F,'String'));%粘性阻尼
J=str2num(get(handles.J,'String'));%转动惯量
Rs=str2num(get(handles.Rs,'String'));%定子每相电阻
Ld=str2num(get(handles.Ld,'String'));Lq=str2num(get(handles.Lq,'String'));
KASR1=str2num(get(handles.KASR1,'String'));
PASR1=str2num(get(handles.PASR1,'String'));
KASR2=str2num(get(handles.KASR2,'String'));
```

```

PASR2=str2num(get(handles.PASR2,'String'));
K1=str2num(get(handles.K1,'String'));P1=str2num(get(handles.P1,'String'));
K2=str2num(get(handles.K2,'String'));P2=str2num(get(handles.P2,'String'));
TL1=str2num(get(handles.TL1,'String'));%转矩起始值
Step_time=str2num(get(handles.Step_time,'String'));%转矩切换时间
TL2=str2num(get(handles.TL2,'String'));%转矩终值
n1=str2num(get(handles.n1,'String'));%给定转速 1
t=str2num(get(handles.t,'String'));%转速切换时间
n2=str2num(get(handles.n2,'String'));%给定转速 2
另外再设置数据的反馈以及结果的显示回调函数，如下代码：
if (get(handles.IM,'value')==1)&&(get(handles.VF,'value')~=1)%判定选择结果
    options = simset('SrcWorkspace','current');
    sim('Synchronous1Change',[],options);%运行仿真文件
    axes(handles.axes1)%图一绘图
    plot(n_out1);grid on;hold on;xlabel('时间 t');
    ylabel('转速 n');title('电流滞环矢量控制转速曲线');
    legend('原参数曲线','改变参数后曲线 1','改变参数后曲线 2');
    axes(handles.axes2)%图二绘图
    plot(is_a1);grid on;hold on;xlabel('时间 t');
    ylabel('电流 Ia');title('电流滞环矢量控制电流曲线');
    legend('原参数曲线','改变参数后曲线 1','改变参数后曲线 2')
    axes(handles.axes3)%图三绘图
    plot(Te_out1);grid on;hold on;xlabel('时间 t');
    ylabel('电磁转矩 Te');title('电流滞环矢量控制转矩曲线');
    legend('原参数曲线','改变参数后曲线 1','改变参数后曲线 2')
end

```

### 5.3.7 仿真结果及分析

完成上述一些列操作后，点击开始仿真，可以得到如下图 5.30 所示运行结果：

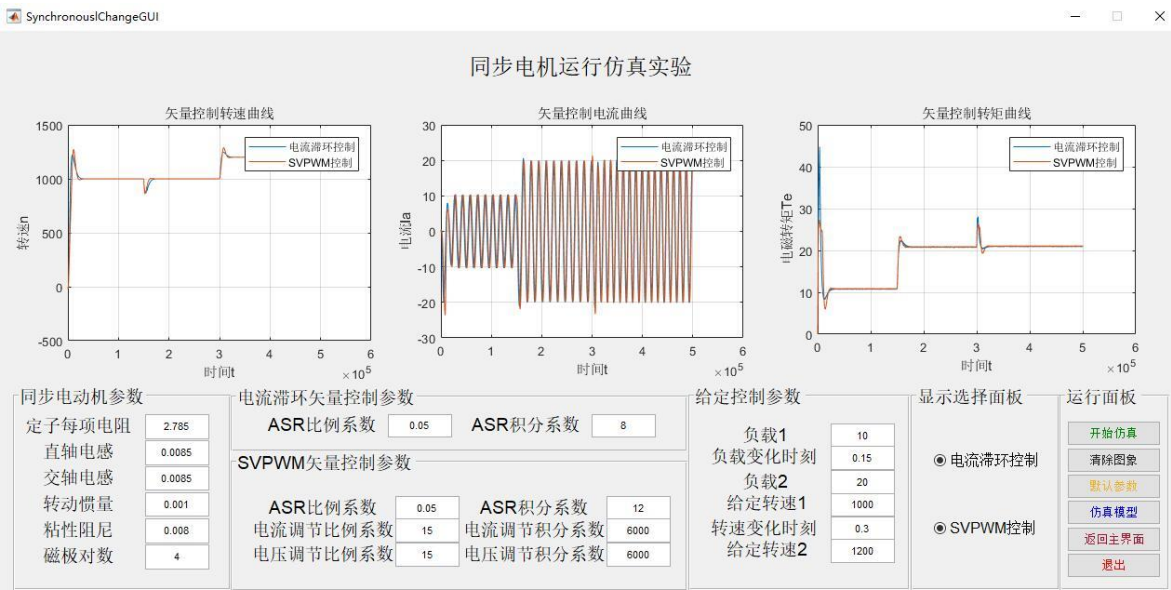


图 5.30 同步电机 GUI 平台运行结果

对于运行结果来说，从图中可以看到，电流滞环控制的动态响应速度更快，但是由于是滞环控制，因此开关频率不一定，所以容易造成电流的纹波比较大，而 SVPWM 控制的电流纹波要小。另外，在模拟启动，起始转矩为  $10N \cdot m$ ，终止转矩为  $20N \cdot m$ ，转矩切换时间为  $0.2s$ ，可以得到图 5.31 所示转速仿真图：

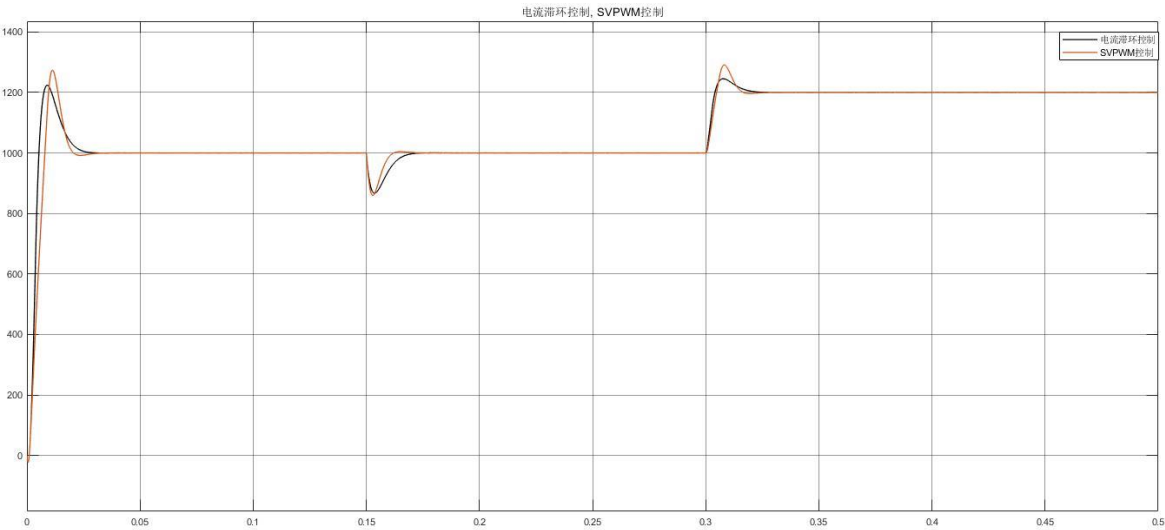


图 5.31 同步电机转速调整结果

由图，其结果与我们预期设想结果相符合。