

基于DSP的异步电动机SVPWM控制技术研究

针对DSP控制异步电动机系统，阐述了系统构成和电压空间矢量SVPWM技术，描述了基本电压空间矢量。详细阐述了SVPWM算法分析和算法实现，描述了Sector值与扇区的对应关系，SVPWM生成的形式有两种：一种是采用七段式SVPWM方法，也称为“软件法”，生成的电压空间矢量PWM波形由七段组成；另一种采用五段式SVPWM方法，也叫做“硬件实现法”，即可以通过F2812内部硬件实现。说明了扇区与跳变时间对应关系表，描述了硬件和软件法对应扇区的三相波形图，说明了基于DSP实现第6扇区PWM的时序图，给出了实验测试的第5扇区和第6扇区三相波形图和经过CCS软件观测到的波形图，验证了基于DSP实现SVPWM控制技术的正确性。

艾红 陆芝杨/北京信息科技大学自动化学院



艾红/副教授

关键词/Keywords

SVPWM ·
扇区 ·
空间矢量 ·
三相波形 ·
矢量控制 ·

空间矢量脉宽调制 (Space Vector Pulse Width Modulation, SVPWM) 技术通过控制三相电压型逆变器功率器件的开关触发顺序和脉宽，产生3个相差电角度且波形失真小的正弦波电流，使电动机获得幅值恒定的圆形磁场，能达到较高的控制性能，而且将逆变器和电动机作为一个整体处理，使模型简单且便于数字化实现。TI公司生产的TMS320F2812 DSP内部具有产生PWM波形的事件管理器，可以通过软件编程和采用DSP内部SVPWM状态机设置空间矢量两种方法实现SVPWM技术。采用DSP控制异步电动机，实现SVPWM是关键。DSP运算速度快、准确度高，可以实时完成SVPWM控制算法^[1-2]。

国家自然科学基金资助 (61273173)，北京市重点学科——检测技术与自动化装置学科建设项目资助 (5111523302)。

DSP控制异步电动机系统构成

DSP控制异步电动机系统构成如图1所示。

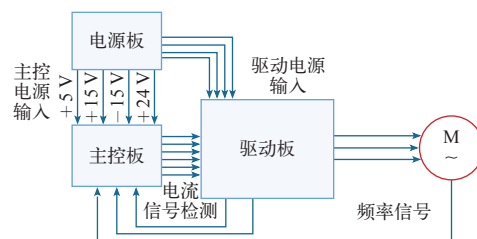


图1 DSP控制异步电动机系统构成

电源板分别为主控板和驱动板提供四路电源，由DSP主控板产生六路PWM波形输入到异步电动机驱动板，交流电变压器主要将220 V电压转换为所需电压输入到异步电动机驱动板的逆变部分，产生使异步电动机旋转所需的电压。驱动板输出三相电压作用于异步电动机，异步电动机驱动板上的电流检测器将异步电动机的两相电流值经过电阻分压输入到DSP控制板中，测得异步电动机的定子电流，电动机端将转速信号输送给主控板。

电压空间矢量SVPWM技术

电压空间矢量SVPWM技术以形成圆形旋转磁场为目标，应用磁力跟踪技术控制逆变器工作，这种方法通过交替使用不同的电压空间矢量来实现。

1. 基本电压空间矢量

图2为三相电压型逆变器模型，通过改变功率开关模块IPM的开关状态、顺序组合以及开关时间，最终使电压空间矢

量形成近似圆形的轨迹，这样得到的电压输出谐波少，能量利用率高。

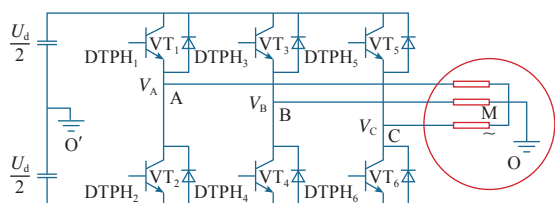


图2 三相电压型逆变器模型

图中, V_A 、 V_B 和 V_C 是电动机的三相电压; 其中的6个功率晶体管有6个DTPH_{*X*} ($X=1,2,3,4,5,6$) 控制开关的状态, 且有上下两路信号, 六路DTPH信号可以计算出最后输出给电动机的空间矢量电压 U_{out} 。当上端DTPH信号为“1”时, 表示上桥臂导通; 当信号为“0”时, 表示下桥臂导通。 V_A 、 V_B 和 V_C 是以O'点为参考点的三相输出电压, 以电动机中性点O为参考点的 V_{AO} 、 V_{BO} 和 V_{CO} 是电动机的三相电压^[3]。

通过三相到两相的变换，三组电压值组合状态可以映射到两相坐标系中，得到8个空间矢量，其中包括6个非零矢量和2个零矢量。6个非零矢量构成六边形的轴线，相邻矢量之间的夹角为 60° ；而2个零矢量则位于坐标原点。这8个矢量就是基本空间矢量，用数字表示各种状态分别为100、110、010、011、001、101、000和111。

2. SVPWM算法分析

计算空间矢量关键的一步就是要计算目标矢量所在的扇区。在 $O\alpha\beta$ 坐标系中,仔细观察每一个扇区, U_α 和 U_β 扇区判定的充要条件如表1所示。

表1 扇区判定条件

扇区号	充要条件
1	$U_\alpha > 0, U_\beta > 0 \text{ 且 } U_\beta/U_\alpha < \sqrt{3}$
2	$U_\beta > 0 \text{ 且 } U_\beta/ U_\alpha > \sqrt{3}$
3	$U_\alpha < 0, U_\beta > 0 \text{ 且 } -U_\beta/U_\alpha < \sqrt{3}$
4	$U_\alpha < 0, U_\beta < 0 \text{ 且 } U_\beta/U_\alpha < \sqrt{3}$
5	$U_\beta < 0 \text{ 且 } -U_\beta/U_\alpha > \sqrt{3}$
6	$U_\alpha > 0, U_\beta < 0 \text{ 且 } -U_\beta/U_\alpha < \sqrt{3}$

通过进一步分析得到电压矢量所在的扇区,

可以用3个量的取值来决定。这3个量分别为 B_0 、 B_1 和 B_2 ，其表达式如下

$$\left\{ \begin{array}{l} B_0 = U_\beta \\ B_1 = \frac{\sqrt{3}}{2} U_\alpha - \frac{1}{2} U_\beta \\ B_2 = -\frac{\sqrt{3}}{2} U_\alpha - \frac{1}{2} U_\beta \end{array} \right. \quad (1)$$

当 $B_0 > 0$ 时, 则 $A = 1$, 否则 $A = 0$ 。同理, 当 $B_1 > 0$ 时, 则 $B = 1$; 否则 $B = 0$ 。当 $B_2 > 0$ 时, 则 $C = 1$; 否则 $C = 0$ 。这样 ABC 就可以按顺序有8种组合。当 ABC 全为0或全为1时, 发现实际计算中并不存在这样的情况, 忽略这两种情况, 剩下的6种 ABC 组合分别对应着不同的扇区, 并且是逐一对应的关系。为了判断方便, 设一个变量Sector, 令 $Sector = 4C + 2B + A$, 可以通过判断Sector的值确定所在的扇区, Sector值与扇区的对应关系如图3所示。通过这种方法, 只需经过简单的加减及逻辑运算, 就可以达到确定扇区的效果, 大大降低了扇区的判定难度, 有利于DSP程序的编制。

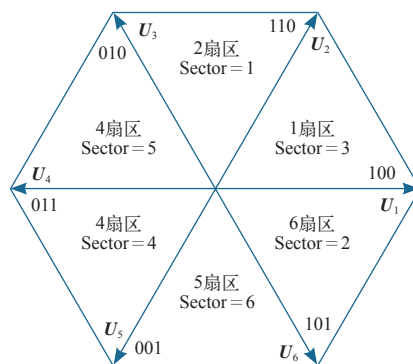


图3 空间矢量扇区对应关系

已知扇区之后, 计算该扇区中两个矢量作用的时间。先假设所在扇区号为2, 作用的基本空间矢量为 $\mathbf{U}_1$ 和 $\mathbf{U}_3$, 得到两个占空比公式为

$$\begin{cases} T_1 = t_2 / T = \frac{1}{2}(-\sqrt{3}U_\alpha + U_\beta) \\ T_2 = t_3 / T = \frac{1}{2}(\sqrt{3}U_\alpha + U_\beta) \end{cases} \quad (2)$$

式中, t_2 、 t_3 为基本空间矢量的作用时间。

引入3个中间变量, 即

$$\begin{cases} X = U_{\beta} \\ Y = \frac{1}{2}(\sqrt{3}U_{\alpha} + U_{\beta}) \\ Z = \frac{1}{2}(-\sqrt{3}U_{\alpha} + U_{\beta}) \end{cases} \quad (3)$$

当目标矢量在第1扇区时,有 $T_1 = -Z$, $T_2 = X$;在第2扇区时,有 $T_1 = Z$, $T_2 = Y$ 。采用相同的方法,可以求得各个扇区的两个空间矢量作用时间的占空比值,对应关系如表2所示。

表2 扇区与时间的关系

扇区号	1	2	3	4	5	6
T_1	$-Z$	Z	X	$-X$	$-Y$	Y
T_2	X	Y	$-Y$	Z	$-Z$	$-X$

3. SVPWM算法的实现

SVPWM生成的形式有两种:一种是采用七段式SVPWM方法,也称为“软件法”,即生成的电压空间矢量PWM波形由七段组成;另一种采用五段式SVPWM方法,也称“硬件实现法”,即可以通过F2812内部硬件实现。这两种方法可以产生对称谐波含量小、开关次数少的PWM波形,相对而言,七段式SVPWM产生的波形谐波含量更小,且每个开关周期有6次开关切换;五段式SVPWM采用每相开关在每个扇区状态维持不变的序列安排,使每个开关周期只有3次开关切换,大大减小了开关损耗,同时增大了谐波含量的存在。

七段式SVPWM的出现,最初是为了减少PWM变换时的开关次数,将基本电压矢量作用时间的分配原则选定为在每一次开关状态转变时,只改变其中一相的开关状态。而且为了减小谐波含量,对零矢量进行平均分配,保证产生的PWM波形是对称的。如当100切换到000时,只需改变A相上下开关状态;但如果从100切换到111,则需要改变B、C两相上下开关状态,因此不同的矢量需要与不同的零矢量完成切换,如100、001要配合000,而110、101要配合111矢量^[4]。

$$T_p = T_1 + T_2 + T_0 \quad (4)$$

式中, T_p 为PWM的周期; T_0 为零矢量的作用时间; T_1 为 U_x 矢量电压的作用时间; T_2 为 U_{x+60} 矢量电压的作用时间。

采用对称方式产生PWM,将 T_1 、 T_2 和 T_0 都分成对等的时间,即式(4)变为

$$T_p = \frac{1}{2}T_1 + \frac{1}{2}T_2 + \frac{1}{2}T_0 + \frac{1}{2}T_0 + \frac{1}{2}T_2 + \frac{1}{2}T_1 \quad (5)$$

3个桥臂的上端晶体管必须根据所选择的电压矢量 U_x 或者 U_{x+60} 决定导通或关断,使合成的空间矢量电压施加到电动机上。用零矢量补偿晶体管导通和关断的时间,如 $T_0 = T_p - T_1 - T_2$,有助于晶体管导通与关断时间的转换,降低功率损耗。

以第6扇区为例,其三相波形如图4所示。图中电压矢量出现的先后顺序为 U_0 、 U_1 、 U_6 、 U_7 、 U_7 、 U_6 、 U_1 和 U_0 ,在下一个 T_p 时段, U_s 角度改变,且出现在别的扇区,利用基本电压空间矢量作用时间的计算方法再计算出 T_0 、 T_1 和 T_2 的数值,得到新的三相波,随着角度不断变化, U_s 将依照顺序进入6个扇区。

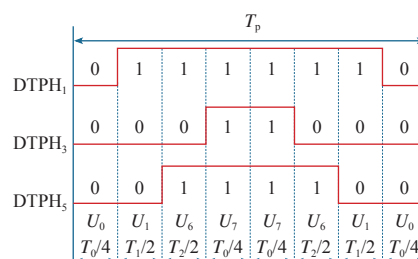


图4 第6扇区三相波形

由DSP生成PWM波形,需要将每一路的跳变时间放入到比较寄存器中,结合6个扇区三相波形图,可以得到6个扇区PWM波形的跳变时间 T_A 、 T_B 和 T_C 。

以第6扇区为例, SVPWM产生的原理如图5所示。

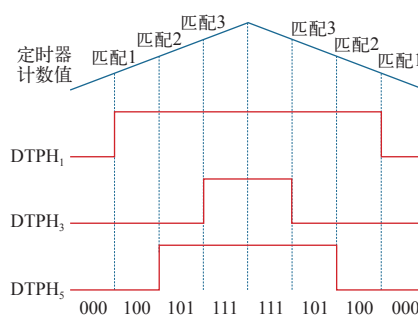


图5 第6扇区PWM产生时序图

为了得到对称的PWM波形，将DSP事件管理器中定时器的计数模式设置为连续增/减模式。定时计数器以初值为0开始增计数，当计数值与比较值相等时，引脚发生跳变；计数到最大值时，计数器开始减计数；当再次与比较值相等时，引脚发生第二次跳变；当计数器一直减到0时，产生下溢中断，完成一个周期。

图5中，计数器从0开始计数，计数到 T_A 时刻，与比较寄存器1的值发生匹配，DTPH₁发生上升沿跳变，接着继续计数；计数到 T_C 时刻，与比较寄存器3的值发生匹配，DTPH₅发生上升沿跳变；计数到 T_B 时刻，与比较寄存器2的值发生匹配，DTPH₃发生上升沿跳变。计数器减计数时，计数到 T_B 时刻，与比较寄存器2的值发生第二次匹配，DTPH₃发生下降沿跳变；计数到 T_C 时刻，与比较寄存器3的值发生第二次匹配，DTPH₅发生下降沿跳变；计数到 T_A 时刻，与比较寄存器1的值发生第二次匹配，DTPH₁发生下降沿跳变；继续减计数到0，产生下溢中断，完成一个周期，同时也是下一个周期的开始。如此往复，实现完整的SVPWM。需要注意的是，比较寄存器中的数值要比周期寄存器中的数值小。

4. 基于DSP内部SVPWM状态机实现SVPWM算法

DSP芯片本身的硬件具有生成SVPWM的功能，通过对比较方式寄存器ACTRA设置空间矢量实现五段式SVPWM，这种方法编程相对简单，但也存在缺点。因为使用了五段式SVPWM，减少了开关次数，但是增大了谐波含量。

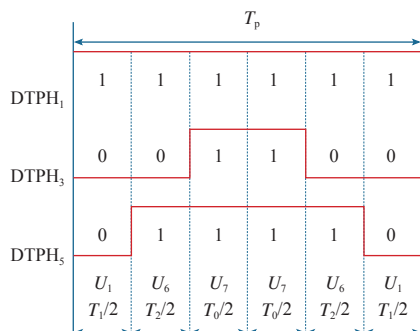


图6 硬件实现第6扇区三相波形

图6为硬件实现第6扇区三相波形，这种方法

的基本特征是在每个PWM周期中都有五段空间矢量作用，而且总有一个通道为常量，表明开关次数少，开关损耗也小。与七段SVPWM相似的地方在于同样需要PWM波形的跳变时间 T_A 、 T_B 和 T_C ；不同点在于程序中要改变比较方式控制寄存器ACTRA，其中ACTRA的位15控制空间矢量旋转方向为“0”时，逆时针旋转；而为“1”时，顺时针旋转。ACTRA位14~位12中写入开启时的空间矢量，后12位是比较输出引脚的输出方式。以第6扇区为例分析说明如图7所示。

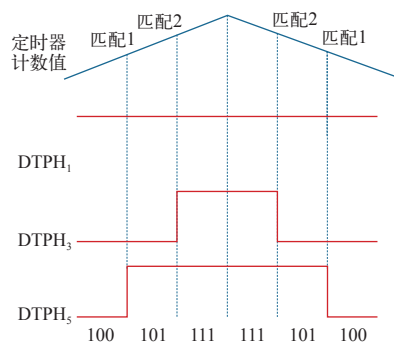


图7 硬件法第6扇区PWM产生时序

控制顺序如下：

1) 写入比较方式控制寄存器ACTRA为1100 0110 0110 0011B，设定ACTRA的位15为“1”，表示顺时针旋转；ACTRA的位14~位12中的空间矢量为100。由100开始旋转到101，比较输出引脚PWM1 输出为强制高；比较输出引脚PWM3和比较输出引脚PWM5的PWM输出为高有效。

2) 将计算得到的 $T_1/2$ 写入比较寄存器3，将 $(T_1 + T_2)/2$ 写入比较寄存器2。

3) 新周期开始，PWM输出为100，即ACTRA的位14~位12中的空间矢量为100。

4) 当计数器计数到与比较寄存器3的值发生第一次匹配时，发生上升沿跳变，由于控制方式中确定旋转方向为顺时针，则PWM输出状态转变为101；当计数增加到与比较寄存器2的值发生匹配时，PWM输出切换至零矢量111。

5) 当计数到最大值时，计数器开始减计数；当与比较值寄存器2发生第二次匹配时，空间矢量状态转变为101；当与比较值寄存器3发生第二次匹

配时,空间矢量状态转变为100,回到了初始状态。

6) 继续减计数到0,产生PWM下溢中断,完成一个PWM周期,并开启下一个周期。如此往复循环,最终实现SVPWM。

矢量控制

交流电动机的三相模型先等效成 $O\alpha\beta$ 两相静止模型,再经过旋转变换为 Odq 两相旋转模型,空间矢量 i 分解为与 d 轴平行的 i_d 和与 q 轴平行的 i_q ,分别可以看作是励磁电流和电枢电流,只要像控制直流电动机那样控制这两个信号就可以得到控制交流电动机的效果。矢量控制理论框图如图8所示。

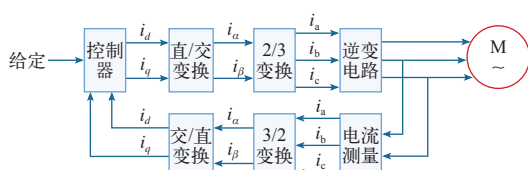


图8 矢量控制理论框图

主要用到的等效变换有三相/两相Clarke变换(3S/2S变换)和两相静止/两相旋转Park变换(2S/2R变换)。Clarke变换是将三相坐标系 $OABC$ 变换到两相坐标系 $O\alpha\beta$; Park变换是将两相静止坐标系 $O\alpha\beta$ 变换到两相旋转坐标系 Odq 中。完成模型变换后,在矢量控制中就可以像控制直流电动机一样控制交流电动机^[5]。

空间矢量模块

由坐标变换得到的两相静止坐标电压值,经由空间矢量计算得到三相PWM开关跳变时间。空间矢量模块输入为 U_α 、 U_β ,输出为 T_a 、 T_b 和 T_c 。空间矢量模块实现的主要程序如下:

```
void svgendq_calc (SVGENDQ *v)
{
    _iq V_a, V_b, V_c, t1, t2;
    long Sector = 0;
    V_a = v->U_beta;
    V_b = _IQmpy (_IQ (-0.5), v->U_beta) + _IQmpy
        (_IQ (0.8660254), v->U_alfa);
    V_c = _IQmpy (_IQ (-0.5), v->U_beta) - _IQmpy
        (_IQ (0.8660254), v->U_alfa);
```

//作逆Clark变换完成两相到三相转换,由 U_α 、 U_β 求得 V_a 、 V_b 和 V_c 三相电压值。

```
if (V_a > _IQ (0))
    Sector = 1;
if (V_b > _IQ (0))
    Sector = sector + 2;
if (V_c > _IQ (0))
    Sector = sector + 4;
```

由 V_a 、 V_b 和 V_c 值计算得到Sector值确定扇区。Sector值与扇区号的对应关系如表4所示。

表4 Sector值与扇区号的对应关系

Sector	1	2	3	4	5	6
扇区号	2	6	1	4	3	5

```
V_a = v->U_beta;
V_b = _IQmpy (_IQ (0.5), v->U_beta) + _IQmpy
    (_IQ (0.8660254), v->U_alfa);
V_c = _IQmpy (_IQ (0.5), v->U_beta) - _IQmpy
    (_IQ (0.8660254), v->U_alfa);
```

设 $X = V_a$ 、 $Y = V_b$ 和 $Z = V_c$,分扇区求解对应的跳变时间。

t_1 、 t_2 为相邻矢量的作用时间,在不同扇区取不同对应值,如表5所示。

表5 扇区的作用时间对应

扇区号	1	2	3	4	5	6
t_1	$-V_c$	V_c	V_a	$-V_a$	$-V_b$	V_b
t_2	V_a	V_b	$-V_b$	V_c	$-V_c$	$-V_a$

再结合表5求出切换时间 T_a 、 T_b 和 T_c 的值。

事件管理器初始化程序如下:

```
void F28X_EV1_PWM_Init (PWMGEN *p)
//PWM初始化
{
    EvaRegs.T1PR = p->n_period; //初始化周期寄存器数值
    EvaRegs.T1CON.all = PWM_INIT_STATE;
    //初始化通用寄存器1控制寄存器
    EvaRegs.DBTCONA.all = DBTCON_INIT_STATE; //初始化死区控制寄存器
    EvaRegs.ACTRA.all = 0; //初始化比较方式控制寄存器
```

```
EvaRegs.COMCONA.all = 0; //初始化比较控制寄存器
```

```
EvaRegs.CMPR1 = p->n_period; //将周期值n_period赋值给比较寄存器1
```

```
EvaRegs.CMPR2 = p->n_period; //将周期值n_period赋值给比较寄存器2
```

```
EvaRegs.CMPR3 = p->n_period; //将周期值n_period赋值给比较寄存器3
```

```
}
```

初始化PWM, 将周期值n_period赋值给定时器1周期寄存器EvaRegs.T1PR。

将PWM_INIT_STATE的值赋值给定时器1控制寄存器EvaRegs.T1CON.all, 其值为1000 1000 0100 0000, 使能计数器, 设定定时器工作在连续增/减计数模式。

设定死区控制寄存器为DBTCON_INIT_STATE初始值: 0000 1111 1111 0100B, 使能3个死区定时器, 周期值为15, 定标因子为1/32。

比较方式控制寄存器为全0。利用DSP产生PWM波形一般有两种方法: 一种是编程法实现; 另一种是空间矢量PWM模式。将需要的波形对应数值赋值给比较方式控制寄存器的位14~位12, 由DSP内部自动生成, 这里选用编程法, 所以将比较方式控制寄存器的位14~位12赋值为“000”。

比较控制寄存器EvaRegs.COMCONA.all值为全0。禁止比较单元操作, 禁止空间矢量PWM模式。3个比较寄存器赋初值为n_period。

在文件f28xpwm.h中

```
#define PWM_INIT_STATE (FREE_RUN_FLAG + \
    TIMER_CONT_UPDN + \
    TIMER_CLK_PRESCALE_X_1 + \
    TIMER_ENABLE_BY_OWN + \
    TIMER_ENABLE)
```

给定PWM初始状态 1000 1000 0100 0000B

```
#define DBTCON_INIT_STATE ( DBT_VAL_15 + \
    EDBT3_EN + \
    EDBT2_EN + \
    EDBT1_EN + \
    DBTPS_X32 )
```

死区定时器控制寄存器初始值 0000 1111 1111 0100B。

经过DSP程序运行截取的第5扇区和第6扇区PWM输出波形如图9、图10所示。

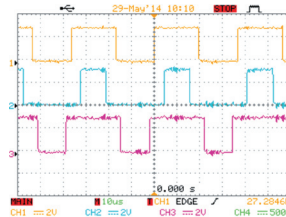


图9 第5扇区三相波形

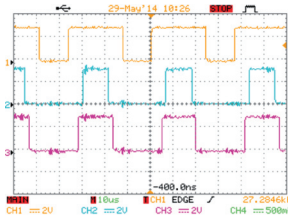


图10 第6扇区三相波形

结束语

矢量控制作为一种先进的控制方法, 被广泛应用于电动机的控制领域。其主要是将异步电动机的模型转换成直流电动机的模型。在矢量控制方法中, SVPWM具有电流谐波少、转矩脉动小及系统可靠性高等优点, 能够达到很好的控制准确度。TMS320F2812 DSP以集成内核、最优化电动机控制事件管理器以及多通道PWM成为电动机的最佳控制芯片。从SVPWM的基本原理出发, 以TI公司生产的TMS320F2812 DSP为控制核心, 实现了数字化SVPWM变频调速控制系统的研究。

参考文献

- [1] 张家龙. 基于DSP的模糊自适应异步电机矢量控制的研究[J]. 采矿技术, 2013, 13(4): 43-47.
- [2] 于国庆, 张颖, 李永伟, 等. 基于DSP的异步电机SVPWM控制系统及优化研究[J]. 河北科技大学学报, 2012, 33(5): 259-262.
- [3] 廖华平, 程小华. 基于DSP的SVPWM矢量控制变频系统研究[J]. 微电机, 2006, 39(9): 70-72.
- [4] 黄少瑞, 郝润科, 朱军, 等. 基于DSP的异步电机SVPWM控制技术实现[J]. 电气自动化, 2010, 32(5): 20-23.
- [5] 蔡焱钢, 南余荣. 一种用于无速度传感器异步电机控制的自适应滑膜观测器[J]. 电机与控制应用, 2008, 35(1): 32-36.

(收稿日期: 2015-02-13) EA