电机 PWM 控制与驱动电路实验指导书



哈尔滨工业大学(深圳) 实验与创新实践教育中心

目 录

1. 实验目的	1
2. 实验原理	1
2.1 直流无刷电机六步换相	1
2.1.1 换向逻辑分析	1
2.1.2 程序设计说明	5
2.2 交流伺服电机 Clark/Park 变换	. 7
2.2.1 Clark/Park 变换思想	. 7
2.2.2 程序设计说明	10
3. 实验步骤	12
3.1 直流无刷电机六步换相实验步骤	12
3.1.1 参考实验设计部分	12
3.1.2 六步换向操作步骤	14
3.2 交流伺服电机 Clark/Park 变换实验步骤	21
3.2.1 参考实验设计部分	21
3.2.2 Clark 和 Park 变换操作步骤	23
4. 实验分析	31
附录1系统硬件介绍	32
1.1 系统构成	32
1.2 DSP 核心板	33
1.2.1 接线定义	33
1.2.2 参数	34
1.2.3 功能特点	34
1.3 直流无刷电机	36
1.3.1 接线定义	36
1.3.2 参数	36
1.3.3 结构特点	37
1.4 直流无刷电机	38
1.4.1 接线定义	38
1.4.2 参数	38
1.4.3 结构特点	39
附录 A CCS 工程导入	40
附录 B 上位机实时内核设置	47

1. 实验目的

- (1) 掌握直流无刷电机的控制原理,理解六步换相法;
- (2) 掌握交流伺服电机的控制原理,理解 clark 变换和 park 变换;
- (3) 了解 Dsp 微控制器 CCS 工程创建导入、编译、信号传输通道连接的过程;

2. 实验原理

2.1 直流无刷电机六步换相

2.1.1 换向逻辑分析



图 2-1-1 有霍尔传感器的 BLDC 控制框图

本控制系统的构成包括:控制器(单片机、DSP、PLC等)、IGBT 驱动器、 桥式电路(全桥和半桥)、无刷直流电机、霍尔传感器等。根据无刷直流电机转 子磁极的位置,对定子线圈进行换相通电,形成6步的旋转磁场,进而带动转子 同步转动。

有传感器的控制方式的关键技术在于:一、如何通过安装在电机上的三个霍尔器件 A、B、C来准确快速的获取转子磁极位置信息;二、如何通过 6 个功率器件组成的 3 相桥式电路来控制定子线圈的 6 拍通电方式,从而形成一个旋转磁场。6 拍定子线圈方式如图 2-1-2 所示。



图 2-1-2 6 拍定子线圈通电方式示意图(逆时针旋转)

按照上图 2-1-2 的方式通电时,三相绕组线圈形成的合成磁场会慢慢的逆时针转动,形成一个旋转磁场,从而导致转子的运动。

此外需要说明的是,控制框图里的霍尔传感器是根据霍尔效应制作的一种磁场传感器,它可以有效的反映通过霍尔原件的磁密度,有正向磁场通过霍尔传感器,输出1,有反向磁场通过霍尔,输出0。当霍尔传感器在和电机的转子做相对运动时,根据转子下磁场密度的变化,来产生变化的信号。如下图2-1-3所示。

如果将一只霍尔传感器安装在靠近转子的位置,当N极逐渐靠近霍尔传感器即磁感器达到一定值时,其输出是导通状态;当N极逐渐离开霍尔传感器、磁感应逐渐小时,其输出仍然保持导通状态;只有磁场转变为S极并达到一定值时, 其输出才翻转为截止状态。在S和N交替变化下传感器输出波形占高、低电平各占 50%。如果转子是一对极,则电机旋转一周霍尔传感器输出一个周期的电压波形。



图 2-1-3 霍尔信号变化

当电机按一个(正转或反转)方向转动时,3个霍尔传感器的输出会按照6步的规律变化,如图2-1-4所示。BLDC电机的转子位置检测器件采用的是霍尔传感器,其与微控制器(MCU)连接非常简单。可以使用任一可用的具有电平变化中断功能的MCU的引脚作为其接口。

为了驱动电机运转,必须根据电机转子所处的空间位置,按照一定的换向顺序,对定子上的A、B、C 三个线圈的两相同时进行通电,形成旋转磁场,从而带动转子旋转。一般在其非驱动端上的定子中安装三个霍尔传感器。这样当转子磁极经过霍尔传感器附近时,它们便会发出一个高(低)电平信号,根据这三个霍尔传感器信号的组合,就能得到定子绕组通电换向的精确顺序。

对于典型的三相带传感器的 BLDC 电机,有6个不同的工作区间,每个区间 中对特定的两相绕组通电。每次换向,都有一个绕组连到电源的正极(电流进入绕 组),第二个绕组连到负极(电流从中流出),第三个处于失电状态。转矩是由定子 线圈产生的磁场和转子之间相互作用产生的。其转矩峰值出现在两个磁场正交时, 而在两磁场平行时最弱。为了保持电机转动,由定子绕组产生的磁场应不断变换 位置,因为转子会向着与定子磁场平行的方向旋转。BLDC 电机其三个定子绕组 每换向一轮(1个电周期)需要六步,应按照"六步换向"给定子绕组加电。

转子每转过 60°电角度,定子上其中一个霍尔传感器就会改变状态,因此, 完成电周期需要六步。在同步模式下,每转过 60°电角度相电流切换一次。(一 个电周期可能并不对应于完整的转子机械转动周期,完成一圈机械转动要重复的 电周期数取决于转子磁极的对数。)每对转子磁极换向需要完成一个电周期。因 此,转子的转数=电周期数/转子磁极对数,即提前设定好输出电流或电压信号的 电周期数,就能得到电机转子旋转的转数,实现位置控制。



图 2-1-4 霍尔传感器输出规律变化原理图

电机驱动电路如图 2-1-5 所示。



图 2-1-5 驱动电路示意图

以上为电机转子逆时针旋转控制方法,作为已知,见表 2-1-1。如需控制电机 转子顺时针旋转,请参考表 2-1-2。

表 2-1-1 (MOTOR_DIR==0)逆时针旋转控制规律

霍	尔传感器输	ìλ	相电压			PWM 信号状态
1	2	3	A B C			pwm1.CmtnPointer
0	1	0	DC+	关	DC-	0
0	1	1	DC+	DC-	关	5
0	0	1	关	DC-	DC+	4
1	0	1	DC-	关	DC+	3
1	0	0	DC-	DC+	关	2
1	1	0	关	DC+	DC-	1

霍尔传感器输入			相电压			PWM 信号状态
1	2	3	A B C			pwm1.CmtnPointer
1	0	1	DC+	关	DC-	0
0	0	1	关	DC+	DC-	1
0	1	1	DC-	DC+	关	2
0	1	0	DC-	关	DC+	3
1	1	0	关	DC-	DC+	4
1	0	0	DC+	DC-	关	5

电机 PWM 控制与驱动电路实验指导书

表 2-1-2 (MOTOR DIR==1) 顺时针旋转控制规律

2.1.2 程序设计说明

如下图 2-1-6 所示,此为实验上位机 Matlab 中的程序。

程序设计思想: BLDC 电机采用模拟启动,采用一个递减的斜坡模块控制脉冲 模块的周期,使频率递增,输入到模拟霍尔信号发生模块,产生一个频率逐渐递 增的霍尔信号输入到六步换相模块,电机开始转动。电机模拟启动后,霍尔传感 器模块产生一个标志位,切换到真实霍尔传感器输出。

模式 1 上位机给定参考的电流 REF,使用 rmp1 模块产生一个斜坡到参考的目标值,rmp1 模块产生一个标志位切换到电流 PID 闭环。

模式 2 上位机给定参考的速度 REF,使用 rmp1 模块产生一个斜坡到参考的目标值,rmp1 模块产生一个标志位切换到电流速度 PID 闭环。



图 2-1-6 直流无刷电机 Matlab 控制程序界面



图 2-1-7 程序设计流程图

①模式选择代码块,模式1为电流环控制,模式2是速度环控制。

```
switch (model)
{
case 1:
    SpeedRef=ref_idc;//电流环控制
    break;
case 2:
    SpeedRef=ref_spd;//速度环控制
    break;
}
```

图 2-1-8 模式选择 CCS 程序

②RC3 斜坡代码块,控制 PWM 波形的占空比,当占空比越来越小时,电机速度加快。

```
Connect inputs of the RMP3 module and call the Ramp control 3 macro.下降斜波
rmp3.DesiredInput = CmtnPeriodTarget;
rmp3.Ramp3Delay = RampDelay;
RC3 MACRO(rmp3)
```

图 2-1-9 PC3 斜坡 CCS 程序

③脉冲代码块,是输出一个频率可调的脉冲给 MOD6CNT MACRO 模块。

```
Connect inputs of the IMPULSE module and call the Impulse macro. 脉冲
```

```
impl1.Period = rmp3.Out;
IMPULSE_MACRO(impl1)
```

图 2-1-10 脉冲 CCS 程序

④模拟信号代码块,模拟六步换相信号。

#define MOD6CNT MACRO(v) 1 1 if (v.TrigInput > 0) 1 { if (v.Counter == 0) /* Reset the counter when it is 5 */ v.Counter = 5; else /* Otherwise, increment by 1 */ v.Counter--; } 图 2-1-11 模拟信号 CCS 程序 ⑤采样电流代码块,采集电机电流,做电流闭环。 if (pwm1.CmtnPointer == 0)DCbus current=-(IQ15toIQ((AdcResult.ADCRESULT1<<3)-IDC offset)<<1); 图 2-1-12 电机采样电流 CCS 程序 ⑥信号输出,采集霍尔传感器信号。 **#define** HALL3 READ MACRO(v) v.CmtnTrigHall = 0;/* Reset trigger, it only handshakes with calling program.*/ if (v.EdgeDebounced==0)/* Debounce current position. */ { HALL3 DEBOUNCE MACRO(v) v.CmtnTrigHall = v.EdgeDebounced; /* Set Commutation trigger here*/ }else /* If current position is debounced, find match in table */ HALL3 NEXT STATE MACRO(v) /* and return pointer to current state. Ptr to be incremented*/ /* by MOD6CNT after RET.*/ /* Reset trigger*/ v.EdgeDebounced = 0;图 2-1-13 信号输出 CCS 程序 ⑦计算电机速度。 **#if** MOTOR DIR == 0if ((pwm1.CmtnPointer==2) && (PreviousState==3)&& (hall1.CmtnTrigHall==0x7FFF))//mod_dir1.TrigInput_hall1.CmtnTrigHall==0x7FFF && (mod1.TrigInput) 5 4 { //a=400: speed1.TimeStamp = <u>VirtualTimer;</u> SPEED PR MACRO(speed1)

图 2-1-14 电机速度计算

2.2 交流伺服电机 Clark/Park 变换

2.2.1 Clark/Park 变换思想

在伺服系统中,直流伺服电机能获得优良的动态与静态性能,其根本原因是 被控制量只有电机磁通Φ和电枢电流 Ia,且这两个量是独立的。此外,电磁转矩 (Tm=K_TΦIa)与磁通Φ和电枢电流Ia分别成正比关系。因此,可做线性控制。 如果能够模拟直流电机,求出交流电机与之对应的磁场与电枢电流,独立地加以 控制,即可使交流电机具有与直流电机近似的优良特性。为此,必须将三相交变 量(矢量)转换为与之等效的直流量(标量),建立起交流电机的等效模型,然 后按直流电机的控制方法对其进行控制。

在使用矢量控制方法控制电机时,需建立交流伺服电机的等效模型,使用 Clark 和 Park 变换对矢量进行变换并加以控制。涉及到交流伺服电机中的磁场分布如图 2-2-1 所示。交流伺服电机的磁场方向为正弦磁场。由于 U、V、W 三相的电流分 别为i_a、i_b、i_c。



图 2-2-1 交流伺服电机的磁场

又因为在交流伺服电机中,电流的幅值相等,相位角各相差 120°。所以可得 电机在动态的时候各相电流表达式:

$$\mathbf{i}_{a} = \mathbf{i}_{m} \cdot \sin (\omega t) \qquad (2-2-1)$$

$$i_b = i_m \cdot \sin (\omega t + \frac{2\pi}{3})$$
 (2-2-2)

$$\mathbf{i}_{c} = \mathbf{i}_{m} \cdot \sin \left(\omega t + \frac{4\pi}{3} \right)$$
 (2-2-3)

其中,ω表示电流角频率,i_m表示电流幅值。这三个电流在空间的矢量分布如 图 2-2-2 所示。



图 2-2-2 平面内电流矢量分布图

把平面内的三个电流矢量经过 Clark 变换(将 abc 变换到静止的αβ坐标系下)



图 2-2-3 电流矢量变换原理图

投影到α、β轴上可得出:

$$\begin{bmatrix} i_{\alpha} \\ i_{\beta} \end{bmatrix} = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \cdot \begin{bmatrix} i_{a} \\ i_{b} \\ i_{c} \end{bmatrix}, \quad S_{a} = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}$$
(2-2-4)

$$\exists \vec{x} \not\in (2\text{-}2\text{-}1) \ , \quad (2\text{-}2\text{-}2) \ , \quad (2\text{-}2\text{-}3) \ , \quad (2\text{-}2\text{-}4) \ \text{\mathbb{K}} \land \vec{x} = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}$$
(2-2-4)

$$\begin{cases} i_{\beta} = (2i_{b} + i_{a})/\sqrt{3} \\ i_{\alpha} = i_{a} \end{cases}$$
(2-2-5)

其中 i_{α} , i_{β} 是把平面内的三个矢量用二维坐标来进行表示。为了进一步方便 计算,把二维坐标系建在电机转子上,同时由于电机转子在实时转动,则需要一 个转化规则,将上述二维坐标系转化为转子上的运动坐标系。随时间变化的绕组 电流产生的磁场可等效为旋转磁场,则可以将绕组电流表达在旋转坐标系中。在 二维坐标系进行旋转变化时,也就是进行 Park 变换(把 $\alpha\beta$ 坐标系变换到旋转 dq 坐标系下),如图 2-2-4 所示。



图 2-2-4 dq 变换

经由旋转之后可以得出:

$$\begin{cases} I_D = i_{\alpha} \cdot \cos \theta + i_{\beta} \cdot \sin \theta \\ I_Q = -i_{\alpha} \cdot \sin \theta + i_{\beta} \cdot \cos \theta \end{cases}$$
(2-2-6)

2.2.2 程序设计说明

程序设计思想: 电机开始,旋转一周进行寻相,完成寻相后,通过上位机配置参数,选择模式,模式0表示电流环设置,双击上位机电流环 PID,设置电流环参数,改变不同的参考电流值 iq_ref。模式1表示速度环设置,双击上位机速度环 PID,设置速度环参数,改变不同的参考速度值 spd_ref。模式2表示位置环设置,双击上位机位置环 PID,设置位置环参数,改变不同的参考位置值 pos_ref,程序流程框图如 2-2-5 所示。



图 2-2-5 程序流程框图



图 2-2-6 交流伺服电机三环控制程序界面

①模式选择,1为电流环,2为速度环,3为位置环。

```
switch (model)//模式一电流环,模式二速度环,模式三位置环
{
   case 1:
   {
       SpeedRef=ref_iq;
       break;
   }
   case 2:
   {
       SpeedRef=ref_spd;
       break;
   }
   case 3:
   {
       SpeedRef=ref_pos;
       break;
   }
}
```

图 2-2-7 模式选择 CCS 程序

②PID 程序, PID 参数设置。

```
pid1_id.Kp = _IQ(recbuff[1] / 100.0);//kp参数设置
if(recbuff[2]==0)pid1_id.Ki=_IQ(0);
else pid1_id.Ki = _IQ(T/(recbuff[2] / 200.0));//ki参数设置
pid1_id.Kd = _IQ((recbuff[3] / 10000.0)/T);//kd参数设置
pid1_id.kh = _IQ(recbuff[4] / 1.0);//开关
ref_id=_IQ(recbuff[5] / 100.0);//ID模拟量
```

```
pid1_iq.Kp = _IQ(recbuff[6] / 100.0);//kp参数设置
if(recbuff[7]==0)pid1_iq.Ki=_IQ(0);
else pid1_iq.Ki = _IQ(T/(recbuff[7] / 100.0));//ki参数设置
pid1_iq.Kd = _IQ((recbuff[8] / 10000.0)/T);//kd参数设置
pid1_iq.kh = _IQ(recbuff[9] / 1.0);//ID模拟量
ref_iq=_IQ(recbuff[10] / 100.0);//ID模拟量
```

图 2-2-8 PID 程序

③PID 结构代码块,构建 PID。

#define PID_MACRO(v)	/* PID构造 */	
<pre>v.Err = _IQmpy((v.Ref - v.Fdb),v.kh);</pre>	/* Compute the error */	1
v.Up= _IQmpy(v.Kp,v.Err);	/* Compute the proportional output */	١
v.Ui= v.Ui + _IQmpy(v.Ki,v.Err);	/* Compute the integral output */	١
<pre>v.Ui= _IQsat(v.Ui, v.OutMax, v.OutMin);</pre>		
<pre>\ v.Ud = _IQmpy(v.Kd,(v.Up - v.Up1));</pre>	/*wei fen*/	١
<pre>v.OutPreSat = v.Up + v.Ui + v.Ud;</pre>	/* Compute the pre-saturated output */	١
<pre>v.Out = _IQsat(v.OutPreSat, v.OutMax, v.OutMi</pre>	n);/* Saturate the output */	١
v.Up1 = v.Up; #endif // PIDREG3 H	/st Update the previous proportional output	*/

图 2-2-9 PID 结构程序

④IPark 模块, Park 变换的逆变换。

1

1

#define IPARK_MACRO(v) /*iPARK变换, PARK变换的反变换*/

v.Alpha = _IQmpy(v.Ds,v.Cosine) - _IQmpy(v.Qs,v.Sine); v.Beta = _IQmpy(v.Qs,v.Cosine) + _IQmpy(v.Ds,v.Sine);

#endif // __IPARK_H__

图 2-2-10 CCS 程序中的 IPark 变换

⑤计算 PWM 占空比模块。

图 2-2-11 计算 PWM 占空比 CCS 程序

⑥电机三相输入电压。

3. 实验步骤

3.1 直流无刷电机六步换相实验步骤

3.1.1 参考实验设计部分

1. 首先观察图 3-1-1 所示代码块,变量 pwm1.CmtnPointer 表示当前绕组电流的方向(参数范围 0~5)。

pwm1.CmtnPointer = 0;//State s1: pwm1.CmtnPointer = 1;//State s2: pwm1.CmtnPointer = 2;//State s3: pwm1.CmtnPointer = 3;//State s4: pwm1.CmtnPointer = 4;//State s5: pwm1.CmtnPointer = 5;//State s6:

图 3-1-1 电流方向代码块

2. 观察如图 3-1-2 所示,变量 hall1.HallGpioAccepted 表示此时霍尔传感器的

扇区(参数范围1~6)。

hall1.HallGpioAccepted == 5//当前转子所处位置经由霍尔传感器反馈为第5扇区 hall1.HallGpioAccepted == 1//当前转子所处位置经由霍尔传感器反馈为第1扇区

图 3-1-2 当前霍尔传感器扇区

#if MOTOR_DIR == 0// Comment the following if-else-if statements in case of // non-inverted Hall logics for commutation states. if (hall1. HallGpioAccepted == 2) pwm1. CmtnPointer = 0; else if (hall1. HallGpioAccepted == 6) pwm1. CmtnPointer = 1; else if (hall1. HallGpioAccepted == 4) pwm1. CmtnPointer = 2; else if (hall1. HallGpioAccepted == 5) pwm1. CmtnPointer = 3; else if (hall1. HallGpioAccepted == 1) pwm1. CmtnPointer = 4; else if (hall1. HallGpioAccepted == 3) pwm1. CmtnPointer = 5; 图 3-1-3 送时针旋转换向控制代码

在理解上述 3-1-1、3-1-2、3-1-3 所示代码块之后,加上对六步换相原理的分析,在 CCS 中导入工程 BLDC_SD(详见附录 A)后,打开 main.c 文件,在图 3-1-4 所示空白处填写关于六步换相电机反转的核心代码(项目中可能已经给出):

① 通过 Ctrl+F 搜索 MOTOR_DIR 进行定位。参考在项目工程中已经给出的 实验正转代码 MOTOR DIR==0;

② 通过观察给出的 MOTOR_DIR==0 的代码,结合六步换相的原理,推断并且填写出 MOTOR_DIR==1 的电机反转代码,并在头文件更改 MOTOR_DIR 的赋值,实现电机反转。

947 🖻 #elif MOTOR DIR == 1 948 $\dot{\boxminus}//$ Comment the following if-else-if statements in case of 949 // inverted Hall logics for commutation states. 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 #endif 965 } // Hall1. CmtnTrigHall == 0x7FFF 1 966 // ClosedFlag==TRUE



3.1.2 六	、步换向操 (乍步骤
	计支	先上电再用 USB 接口下载代码。
	注息	CN1-CN6 接口不支持带电插拔,请先切断掉电源。

1. 线路连接: CN3 和 CN5 连接、CN4 和 CN6 连接, USB 与上位机 USB 连接。 在连接 USB 接口前时, 需要先打开**设备管理器**——点击端口; 然后查看现有端口, 再插上 USB 线观察哪一个端口是重新出现的, 并做好实验记录, 为后续操作做好 准备。向上推动断路器开关上电, 实验平台面板框架如图 3-1-5 所示;



图 3-1-5 硬件接线

2. 打开 CCS,在目录"C:\ProgramFiles\MATLAB\R2015b\workspace\pwm \BLDC"下导入工程 BLDC,**工程导入详见附录 A**。

编译完成后点击仿真下载,若点击红框的运行按钮,软件开始运行,若表现 改图表颜色暗淡,则已经处于运行状态,不必再次点击。如图 3-1-6 所示; 注意: 一定要按照附录 A 流程完整的导入 CCS 工程 BLDC,在第三个步骤前保持程序 处于持续运行状态。

电机 PWM 控制与驱动电路实验指导书

CCS Debug - BLDC_SD/main.c - Code Composer Studio				
File Edit View Project Tools Run Scripts Window Help				
i 🗂 🕶 🔄 💽 🌘 🕒 💻 🧟 💀 🖉 🖉 👘 🖉 🌑 🖉 🍝 🕶	🕹 💣 🕶 🔍 👁 🛛	~ ~ ∞ ∞ ∞	A -	
		Qui	ck Access 🕴 😰 🛛 🖪	CCS Edit 🧤 CCS Debug
* Debug ⊠ 这们 · · · □	(x)= Variables 🛛 🙀	'Expressions 1919 Registers		
BLDC_SD [Code Composer Studio - Device Debugging]			🏠 📲 🖻 🏟 🗿	× 🖗 📫 🖻 🔻 👩
Texas Instruments XDS100v2 USB Emulator/C28xx (Suspended)	Name	Туре	Value	Location
main() at main.c:131 0x3F6000				
c int00() at boot28.inc:248 0x3F6C36 (the entry point was re				
🔎 Texas Instruments XDS100v2 USB Emulator/CLA_0 (Disconnecte				
۰ ا				
i main.c ⊠				
▶ 131 {				*
132 133 // Step 1. Initialize System Control:				-
4				F
📮 Console 🔀		🕹 🗘 🔄) 📰 🚮 🥅 🖳 🔜 🖊	' 📮 🕶 📩 🕶 🗖
CDT Build Console [BLDC_SD]				1221
**** Build of configuration Release for project BLDC_SD ****				^
"C:\\ti\\ccsv6\\utils\\bin\\gmake" -k all gmake: Nothing to be done for 'all'.				
**** Build Finished ****				~
<u>&</u>				Free License

图 3-1-6 仿真界面

3. 打开 Matlab, 打开目录"C:\ProgramFiles\MATLAB\R2015b\workspace\ pwm\BLDC"下 speed_control 程序,如图 3-1-7 所示;

📣 MATLAB R20	15b - academic use									X
HOME	PLOTS APPS						9 ¢ 1	🔁 🕐 Search Do	cumentation	۶ 🔍
New New O Script	pen Compare Data Wor	Rew Variable	Analyze Code	Simulink Library	Preferences Set Path Parallel • •	② ➢ Commun s Help ➡ Request	ity Support			
	LE	VARIABLE	CODE	SIMULINK	ENVIRONMENT	RESOURCES				- 0
Current Folder	Onen	Documents P MATLAD P				×	()	Workspace		
🗋 Name 🔺								Name A	Value	
🗄 📕 slprj	G V BLDC +	And and a subscription of the subscription of		▼ * + ∄	健康 BLDC	9		Thunne -	Value	
🗄 📙 speed_co	组织 ▼ 新建文件夹				8≡ ▼ [
ti speed_co ≧ ticcs_c200 È ticcs_c200	 ▶ 计算机 ▲ 本地磁盘 (C:) 本 本地磁盘 (C:) 本 本地磁盘 (C:) ● 系统保留 (F:) ● 可移动磁盘 (G:) ● 可移动磁盘 (G:) ● 可移动磁盘 (G:) 	名称 BLDC_SD common d dnclib user werer	(修改日期 2018/12 2018/12 2018/12 2018/12 2018/12	月 英型 2/23 8:59 文件来 2/23 8:59 文件来 2/23 8:59 文件来 2/23 8:59 文件来 7/23 8:59 文件来 1/23 8:59 文件来	大小 ik Model 4() KB				
Details	Ż#2	(N): speed_control			MATLAB files 打开(0) 取	¥ ĴĂ		٢	11	

- 图 3-1-7 Matlab 软件打开项目
- 4. 打开程序的路径必须是全英文路径,如图 3-1-8 所示;



Paspeed_control - Simulink academic use File Edit View Display Diagram Simulation Analysis Code Tools Help 🖎 • 📲 🧇 🔶 🛔 👹 • 🧮 • 🕑 🗉 🖉 • inf External • 🕢 • 🛗 • speed_control 🛞 🎦 speed_control 🕨 Q Paradox BLDC Control Motor - PC software K N K Note Packet How to run this demo: 1. Connect the Hardware correctly. 2. Build all the demo. ⇒ Out1 1 1 Input AE MODEL Goto Packet Inpu 3. Start real-time code 0.4 双击设置COM印 rt [4h] PWM 1A PWM 1E PWM 2A (1)0 0 In1
 In1
 In2
 In3
 In4
 In5 Out Count mtnPointer CmtnPeriod Target -0 IMPULSE RC3 PWM28 PWM3A PWM38 Manual Switch MOD6_CNT DutyFunc id In2 3-Phase Inverter BLDC MOTOR PWMDRV FB_idc PID_SPD Ou PID_idc Out FB idc pid spd pid do speed HALL_DRV HALL_A 0 HALL E Ē. lose flag HALL_C 11 View 1 warning 129% Ready ode3

图 3-1-9 双击 Packet Input 模块示意图

6. 在弹出的窗口中选择 Board setup 选项。在弹出的对话框中设置 COM 口参数。在第一个步骤时已经对新出现的端口做了记录,此时的端口需要与记录的端口保持一致,波特率是 115200。点击 Test 以检查系统是否能够正常通信,出现绿色 √则表示通信正常。如图 3-1-10 所示;



电机 PWM 控制与驱动电路实验指导书

- 图 3-1-10 COM 口参数设置界面
- 7. 双击 single-out 模块, 如图 3-1-11 所示;



图 3-1-11 双击 single-out 模块示意图

8. 双击 Packet Output 模块设置 Packet Input 的 COM 口, 如图 3-1-12 所示;

电机 PWM 控制与驱动电路实验指导书



图 3-1-12 双击 Packet Output 模块示意图

9. 在弹出的窗口中选择 Board setup 选项,在弹出的对话框中设置 COM 口参数。在第一个步骤时已经对新出现的端口做了记录,此时的端口仍然需要与记录的端口保持一致,波特率是 115200。点击 Test 以检查系统是否能够正常通信,出现绿色 √则表示通信正常。如图 3-1-13 所示;



图 3-1-13 设置 COM 口参数界面

10. 把电机模式改为1(电流环运动模式),双击 MODEL 模块,在弹出的对 话框中 Constant Value 填入1(电流环运动模式),单击 OK,如图 3-1-14 所示;



电机 PWM 控制与驱动电路实验指导书

图 3-1-14 设置为电流环运动模式(力矩模式)界面 11. 启动运行程序,打开 PID 模块(力矩模式 PID),如图 3-1-15 所示;



图 3-1-15 力矩模式 PID 模式参数设置面板

12. 速度显示模块会自动弹出,若未出现,则打开 Speed (速度) 模块,如图 3-1-16 所示;

a speed_contr	trol * - Simulink academic use	- 0 %
File Edit V	view Display Diagram Simulation Analysis Code Tools Help	
因•副•		
speed_control		
🕒 🎦 speed_		•
•		
8	www.waw.husy.们块给定多数。 speed	
→	Paraneters are hie loois view simulated rep 21 10 5	
	0.0 1.000 Sceed 1800	
	0.090	
	pid,kp	
	0.0 2.0	
	0.350 al Switch NOCO	
	pid,ki	
	0.0 0.100	
	PID_dio Out	
	pra_ka 460 460	
	OK Cancel Help Apply 0 10 20 30 40 50 60 70 80 90 100	
6	Running Offset=100 T=123.910	
(B)		
»		
Running the m	nodel on 'Simulink Desktop Real-Time' View diagnostics 100% T=123.148	FixedStepDiscrete

电机 PWM 控制与驱动电路实验指导书

图 3-1-16 力矩模式速度实时监控

13. 在对话框中拖动滑块给定 pid_ref 参考电流,拖动滑块给定 PID 参数,电 机开始运动(参考电流为0时电机不动)。运行结束拖动滑块使参考电流为0, 电机停转,再点击停止按钮,打开 speed 示波器模块,点击缩放按钮,可以看到 完整的运行过程的曲线图。(注意,第一次滑动滑块时,驱动板上自举电容需要 充电,所以电机不会运动,把滑块置零后,再次滑动时电机开始运动)

通过打开 speed 或 pos 模块观察速度位置变化情况,并保存几张速度或位置变 化曲线。其中可以通过 view-style-调整图像背景颜色。

如果程序设计有误, 电机可能会出现以下现象:

- 1. 在滑动上位机 PID 模块时, 电机开始会动作一下, 但不会连续运动;
- 2. 在 CCS 中编译时程序会报错;
- 3. 电机完全无动作或有异响。

注意事项:

(1)在按下停止按钮前,若没有将 pid_ref 滑块置零,电机依 然会继续运转,请一定要将 pid_ref 置零,再点击停止按 钮。

(2)程序只需改动关键词 MOTOR_DIR==1 处的代码,以及头 文件中 MOTOR DIR 的赋值(0或1),其余部分勿改动。

3.2 交流伺服电机 Clark/Park 变换实验步骤

3.2.1 参考实验设计部分

1.根据对 Clark 变换原理的理解,以及给出的变量的意义,设计出 Clark 变换 核心代码。

变量	含义	注释
		应用中, v.Alpha 就相当
v.Alpha	i _α	于公式中的
		i _α
		应用中, v.Beta 就相当
v.Beta	i _β	于公式中的
		i _β
		应用中, v.As 就相当于
v.As	i _a	公式中的
		i _a
		_IQmpy(A,B)是一个乘
_IQmpy(A,B)	兵中, A,D 刀 加代衣网	法函数,表示 A,B 两个
		乘数相乘
	甘山 ٨ 韦元仁音一个	_IQmpy2(A)是一个 2
_IQmpy2(A)		倍关系,表示当前 A 的
	<u> </u>	二倍
		_IQ 表示数据格式,例
_IQ		如像 int, double 之类的
		数据格式
		应用中,v.Bs 就相当于
v.Bs	i _b	公式中的
		i _b

表 3-2-1 关于 Clark 代码的变量含义

2.打开 CCS,在目录"C:\ProgramFiles\MATLAB\R2015b\workspace\pwm \PMSM"下导入工程 PMSM_SD(详见附录 A),打开 main.c 文件,找到 main.h 按 下 Ctrl 再点击 main.h,找到 clarke.h 按下 Ctrl 再点击 clarke.h.,如图 3-2-1 所示,搜索 CLARKE_MACRO 到指定位置。请在下面的空白处,结合对 Clark 变换原理的理 解以及应用表 3-2-1 中给出的变量填写核心代码。此实验为验证实验,所以程序中 代码已经提前详细给出,直接按后续实验步骤操作即可。(每行代码后都需要加 上"\",不得有空行,否则编译会出错)。

```
37
38 #define CLARKE_MACRO(v)
39
40
41
42
43
44
45 // 1/sqrt(3) = 0.57735026918963
46
```

图 3-2-1 填写 Clark 变换代码

3. 根据对 Park 变换原理的理解,以及给出的变量的意义,设计出 Park 变换的核心代码。项目程序工程中已经默认给出。

表 3-2-2 关于 Park 代码的变量含义

变量	含义	注释
v Da	T	应用中, v.Ds 相当于公式中的
v.Ds	ID	I _D
N Os	L	应用中, v.Qs 相当于公式中的
v.Qs	١Q	I _Q
IOmmy(A D)	其山 AD 公别代表西个乖粉	_IQmpy(A,B)是一个乘法函
_IQIIIpy(A,D)	兵中,AD 分别代衣两个来数	数,表示A,B两个乘数相乘
v Alaba	;	应用中, v.Alpha 就相当于公
v.Alpna	1_{α}	式中的i _a
v Data	÷	应用中, v.Beta 就相当于公式
v.Beta	$^{1}\beta$	中的i _β
v Casina		应用中, v.Cosine 就相当于公
v.Cosine	COS Ø	式中的cos 0
v Sin a	ain 0	应用中, v.Sine 就相当于公式
v.sine	Sin Ø	中的sin θ

4. 打开 main.c 文件, 按住键盘上 Ctrl 键, Ctrl+f 搜索 Park_MARCRO,按下 Ctrl 使用鼠标点击 PARK_MACRO, 进入到 park.h 文件中, 如图 3-2-2 所示, 请在 下面空白处,结合对 Park 变换原理的理解以及应用表 3-2-2 中给出的变量填写所 设计的核心代码。(每行代码后都需要加上"\",不得有空行,否则编译会出错)。(由于为验证实验,程序代码已经给出。)

44 #define PARK_MACRO(v)	X
45	
46	
47	
48	
49	
50	
51 #endif //PARK_H	
52	

图 3-2-2 填写 Park 变换代码

5. Clark 和 Park 变换代码设计完成后,点击保存,编译下载代码,并根据后续步骤验证代码。代码设计若无异常,经过实验验证,电机可正常运行,则在上位机中电流环、速度环、位置环都可正常运行。

3.2.2 Clark 和 Park 变换操作步骤

1. 线路连接: CN1 和 CN5 连接、CN2 和 CN6 连接, USB 与上位机 USB 连接, 连接电源, 开关上电, 如图 3-2-3 所示;



图 3-2-3 硬件接线

2. 打开 CCS,在目录"C:\ProgramFiles\MATLAB\R2015b\workspace\pwm \PMSM"下导入工程 PMSM_SD(工程导入详见附录一)。

在程序编译过程中,出现关键字错误,不存在 kh,打开 main.c,将 kh 所在的 代码行前出现红色小×号的那一行代码给注释掉,如果 kh 前不存在红色小×号不 要注释掉这一行代码。

顺利编译完成后点击仿真下载,点击运行按钮,同时观察电机圆盘是否旋转 一圈寻相完成。如果未出现转一圈自动寻相,则重新启动编译程序直到完成为止, 不然后续流程不能完成。如图 3-2-4 所示;

电机 PWM 控制与驱动电路实验指导书

File Edit View Project Tools Run Scripts View	a Ø ♥ • ♥ • ♥ ■ ■ O• Variables 23 ♥ Name	"Epressions IIII Registers Type	Value 	Quick Access II 램 I 목 CCS E	idik (Pg. CCS Debug)
Construction C	k ♥ ■ □ Or Variables 21 M Name	Expressions III Registers Type	Value	Quick Access : 함 이 등 CCS E 소 4는 이 속 가 또 높 다 1 Location	dik (<u>Pg. CCS Debug</u>) 1월 - 7 - 1 - 1 1월 - 19 19 19 19 19 19 19 19 19 19 19 19 19 1
Debug 22 Debug 22 Debug 22 Debug 24	kg ♥ ■ □ 00 Variables ⊠ %g Name	Epressions III Registers Type	Value	松 化 P 化 X K K P C C	d ∨ • □ <i>s</i>
OF PMINA OF PMINA	Name	Type	Value	Location	
A Tray and the second sec					
 main() at main(-12/6 0/354094) arg, ramin() at ang main(-12/6 0/354096) cint00() at boot25.inc:248 0/354096 (the entry point was reached) Texas Instruments XD5180/2 US8 Emulator/CLA_0 (Disconnected : Unknown) 					
m cymool ar doocaancew aonodor (pre emy ponn was reacted) x [®] Texas Instruments XDS100v2 USB Emulator/CLA_0 (Disconnected : Unknown)					
*					
📓 main.c 🛙					- 0
• 176 (
177 178 // Step 1. Initialize System Control:					
179 // PLL, WatchDog, enable Peripheral Clocks					-
Console II			÷.	9 🔞 🗊 🖬 🗐 🔍 🕣 •	.
CDT Build Console [PMSM_SD]					
**** Build of configuration Balance for project DMCM CD ****					*
barra of contrage screet to project that ap					
"C:\\ti\\ccsv6\\utils\\bin\\gmake" -k all					
Bundet Hotstill to be done for base i					
Build Finished ****					
					*
	6	0		○ 英 *	× 🛛 🕯 🖬 🗘

图 3-2-4 仿真启动界面

3. 打开 Matlab 软件,单击 Open,在目录"C:\ProgramFiles\MATLAB\R2015b \workspace\pwm\PMSM"下打开 PMSM 项目,如图 3-2-5 所示;

MATLAB R2015b - academic use				12	. 0 >
HOME PLOTS APPS		BULD		Search Documentation	n P
w New Open Character Data	Variate Analyze Code III OPeterences @ Peterences @ Community n Variate * Print Simulation Layout Peterences @ Peterences @ Peterences @ Community r Workspace @ Cherr Commands * Literary III Perelet Add-Omm Meb @ Request Support				
FILE VARIABLE	CODE SIMULIAK ENVIRONMENT RESOURCES				
🔥 Open		0	Wedgeses		•
()	 + + 度度 pwm 	U	workspace	M.L.	
			Name +	Value	
A AND A					
★ 牧童夫 ○○○○	体成日期 英国 大小				
ik 下戦 ik common	2018/12/21 10:40 文件夹				
	2018/12/21 10:40 文件央 2018/12/21 10:40 文件央				
MI MOZTO PERSEZIA	2018/12/21 11:09 文件夹				
🚌 💼	64 2018/12/21 11:09 文件夹				
2045 User	2018/12/21 10:40 文件夹				
■ Bb (speed_control)					
文档					
▲ 音乐					
➡ 家庭组					
X14-6(N):	All MAILAB hies				
	[1开(O)] 取油				
ils 🔨					
Ready				O 🕈 🕉 🔏 🛙	-

图 3-2-5 Matlab 软件打开项目

4. 打开程序的路径必须是全英文,如图 3-2-6 所示;



电机 PWM 控制与驱动电路实验指导书

图 3-2-6 全英文程序路径示意图

5. 双击 Packet Input 模块设置 Packet Input 的 COM 口, 如图 3-2-7 所示;



图 3-2-7 双击 Packet Input 模块示意图

6. 在弹出的窗口中电机 Board setup 选项, 在弹出的对话框中设置 COM 口参数, 如图 3-2-8 所示;



电机 PWM 控制与驱动电路实验指导书



7. 打开 single-out 模块, 如图 3-2-9 所示;



图 3-2-9 打开 single-out 模块

8. 打开 Packet Output 模块,设置 Packet Input 的 COM 口,如图 3-2-10 所示;



电机 PWM 控制与驱动电路实验指导书

图 3-2-10 双击 Packet Output 模块

9. 在弹出的窗口中电机 Board setup 选项,在弹出的对话框中设置 COM 口参数,如图 3-2-11 所示;

a speed_control/single	g speed_control/single_out * - Simulink academic use			
File Edit View D	isplay Diagram Simulation Analysis Code Tools Help			
		(Petama)		
	Block Parameters: Packet Output			
speed_control ×	Simulink Desktop Real-Time Packet Output (mask) (link)			
😁 🎦 speed_control	Write binary data or a CAN message to a communication channel.		•	
	Data acquisition board		*	
53	Install new board Delete current board			
=	Standard Devices Serial Port [4h]	Standard Devices Serial Port		
AI	Timing			
100	Sample time:	Senal port: Data bits:		
	0.004			
	Maximum missed ticks:	lone -		
		Baud rate:		
	Show "Missed Ticks" port			
	Thes CP0 when washing			
	- Input/Output	XON/XOFF flow control		
	1	RTS/CTS flow control	E	
	Output packet size:			
	23	Direct port access		
	Output packet field data types:	OK Test Baued Connel		
	'23*uint8'	OK TEST Revent Californ		
	Output packet field byte order: Little Endian		-	
	Show "Data Ready" port			
	Show "Data Error" port			
1001	Initial value:			
	Final value:			
>			-	
Ready		View 68 warnings 125%	ode3	
	Un Cancer neip Apply			

图 3-2-11 设置 COM 口参数

10. 把电机模式改为1(电流环运动模式),打开 MODEL 模块,在弹出的对 话框中 Constant Value 选项中填入1(电流环运动模式),单击 OK,如图 3-2-12 所示;

电机 PWM 控制与驱动电路实验指导书



图 3-2-12 设置为电流环模式(力矩模式)示意图

11. 启动运行程序, 打开 PID_IQ 模块(力矩模式 PID), 如图 3-2-13 所示;



电机 PWM 控制与驱动电路实验指导书



12. 在对话框中拖动滑块给定 pid_ref 参考电流,拖动滑块给定 PID 参数,电 机开始运动(参考电流为0时电机不动);



图 3-2-14 力矩模式参数设置面板

13. 打开 Speed 和 pos 模块可以监控力矩模式的速度和位置,如图 3-2-15 所示;



电机 PWM 控制与驱动电路实验指导书







图 3-2-16 力矩模式速度实时监控

通过打开 speed 或 pos 模块观察速度位置变化情况,并保存几张速度或位置变 化曲线。其中通过 view-style-调整图像背景颜色。

注意事项:在按下停止按钮前,若没有将 pid_ref 滑块置零,电机依然会继续运转,请一定要将 pid_ref 置零,再点击停止按钮。

4. 实验分析

4.1 直流无刷电机六步换向

 1、理解和写出无刷直流电机换相的逻辑(逆时针/顺时针旋转时霍尔器件的 状态,以及相应三相通电的方式)。

2、通过对 CCS 程序的工程导入、程序编译烧录、信号传输通道设置、电机 工作模式选择,了解直流无刷电机的整个控制操作流程,实现对电机的运动控制, 特别是运动速度大小的控制。该实验属于验证性实验,因此为体现实验效果的完 成程度,需要记录保存几张不同 pid_ref 参考电流下的速度变化图像。pid_ref 可以 动态持续的变动,相应的也能查看到对应速度的变化,因此每个实验组的同学得 到图像应当不同,以此体现实验工作差异和实验完成程度。

3、实验涉及到部分源代码的设计,根据分析理解六步换向原理和现已给出的 源代码,编写程序关键词 MOTOR_DIR==1 处的代码(项目可能已经给出),实 现电机的相对反方向转动。

4.2 交流伺服电机 Clark 变换和 Park 变换

1. 根据 Clark 变换和 Park 变换的原理,写出永磁同步电机的 Park 和 Clark 变换 公式。

2. 通过对 CCS 程序的工程导入、程序编译烧录、信号传输通道设置、电机工 作模式选择了解伺服电机的整个控制操作流程,实现对电机的运动控制,特别是 运动速度大小控制。该实验为验证性实验,通过实验操作了解交流伺服电机是如 何实现位置的精确控制,以及速度的改变调整,能够对交流伺服电机的控制原理 和技术平台有一定了解。为体现实验顺利的完成程度,通过速度 speed 或者位置 pos 观察框观察电机速度/位置随着 PID_ref 滑动模块的变化曲线,并在停止后保存 图像,每组实验得到的曲线图应当不同,以此体现实验工作差异和实验完成程度。

附录1 系统硬件介绍

1.1 系统构成

该实验装置由交流伺服电机、直流无刷电机、DSP 驱动板及驱动器等主要部件组成,结构如图 1-1、1-2 所示。



图 1-1 电机调试平台内部结构示意图



图 1-2 电机调试平台外部结构



图 1-3 硬件连接原理框图

1.2 DSP 核心板

1.2.1 接线定义



图 1-4 DSP 驱动板接线定义

1.2.2 参数

表 1-1 DSP 核心板参数

1				
28035 核心板				
	32 位中央处理单元(CPU)			
	60MHz 主频			
	45 个复用通用输入输出(GPIO)引脚			
	三个 32 位 CPU 定时器			
	串行端口外设(一路 SCI/两路 SPI/一路 I2C/一路 LIN/一			
4土 上	路 eCAN)			
村品	14 路增强型脉宽调制器(ePWM),可选高分辨率			
	PWM(HRPWM)			
	一路增强型捕捉(eCAP)			
	两路高分辨率输入捕获(HRCAP)			
	一组增强型正交编码器脉冲(eQEP)			
	16 路模数转换器(ADC)12 位 AD			
备注	可编程控制律加速器(CLA), 32 位浮点算术加速器独			
	立于主 CPU 之外的代码执行			

表 1-2 驱动单元参数

	8301 驱动板
	工作电源电压范围为 8V-60V
	2.3A 沉电流和 1.7A 源门驱动电流能力
	具有可调节增益和抵消补偿的集成双重并联电流放大器。可选放
	大 10、20 倍、40 倍以及 80 倍
特点	3 或 6 路 PWM 输入的独立控制
	支持 100%的占空比的引导门驱动器
	可编程空载时间以避免外部 FET 直通
	外部 MOSFETs 的可编程过流保护
	能减少电磁干扰(EMI)的转换速率控制
	驱动器是由 28035 核心板和 8301 驱动板组合而成, 驱动板能接
友让	一组霍尔或一组编码器,可以驱动 500W 以下的无刷直流电机
莆任	和永磁同步电机。驱动板供电电压 12-60V,目前最大电流 5A,可
	以搪锡改变最大电流到 20A。

1.2.3 功能特点

DSP 芯片,也称数字信号处理器,是特别适用于数字信号处理运算的一种微处理器,其主要应用是实时快速地实现各种数字信号处理算法。根据数字信号处理的要求,DSP 芯片一般具有如下主要特点:

- 1. 在一个指令周期内可完成一次乘法和一次加法;
- 2. 程序和数据空间分开,可以同时访问指令和数据;
- 3. 片内具有快速 RAM, 通常可通过独立的数据总线在两块中同时访问;

- 4. 具有低开销或无开销循环及跳转的硬件支持;
- 5. 快速的中断处理和硬件 I/O 支持;
- 6. 具有在单周期内操作的多个硬件地址产生器;
- 7. 可以并行执行多个操作;

8. 支持流水线操作,使取指、译码和执行等操作可以重叠执行。当然,与通 用微处理器相比,DSP 芯片的其他通用功能相对较弱。

本系统采用的为 DSP-F28035 核心板,如图 1-5 所示。



图 1-5 DSP 核心驱动板示意图

该系列微控制器为 C28x[™]内核和控制律加速器(CLA)供电,此内核和 CLA 与 低引脚数量器件中的高集成控制外设向耦合。该系列的代码与以往基于 C28x 的代 码相兼容,并且提供了很高的模拟集成度。一个内部电压稳压器允许单一电源轨 运行。对 HRPWM 模块实施了改进,以提供双边缘控制(调频)。增设了具有内 部 10 位基准的模拟比较器,并可直接对其进行路由以控制 PWM 输出。ADC 可 在 0V 至 3.3V 固定全标度范围内进行转换操作,并支持公制比例 VREFHI /VREFLO 基准。ADC 接口专门针对低开销/低延迟进行了优化。

1.3 直流无刷电机

1.3.1 接线定义



图 1-6 直流无刷电机接线定义

1.3.2 参数

表 1-3 直流无刷电机参数

电机型号	DT60BL100-230
额定功率(W)	200
额定扭矩(N·m)	0.65
瞬时最大扭矩(N·m)	1.75
额定/最大转速(r/min)	3000/4000
电机转动惯量(kg.cm ²)	0.48
每秒最大功率(kw/s)	21.5
电阻(Ω)	0.22
电感 (mh)	0.29
额定电流(A)	10.5
瞬时最大电流(A)	31
极对数(对)	4
反电动势常数(V/Krpm)	4.2

1.3.3 结构特点



无刷电机

图 1-7 直流无刷电机模型

直流无刷电机由电动机主体和驱动器组成,是一种典型的机电一体化产品。 电动机的定子绕组多做成三相对称星形接法,同三相异步电动机十分相似。电动 机的转子上粘有已充磁的永磁体,为了检测电动机转子的极性,在电动机内装有 位置传感器。驱动器由功率电子器件和集成电路等构成,其功能是:接受电动机 的启动、停止、制动信号,以控制电动机的启动、停止和制动;接受位置传感器 信号和正反转信号,用来控制逆变桥各功率管的通断,产生连续转矩;接受速度 指令和速度反馈信号,用来控制和调整转速,提供保护和显示等等。

直流无刷电机的特点:

- 1. 可替代直流电机调速、变频器+变频电机调速、异步电机+减速机调速;
- 2. 具有传统直流电机的优点,同时又取消了碳刷、滑环结构;
- 3. 可以低速大功率运行,可以省去减速机直接驱动大的负载;
- 4. 体积小、重量轻、出力大;
- 5. 转矩特性优异,中、低速转矩性能好,启动转矩大,启动电流小;
- 6. 无级调速,调速范围广,过载能力强;

1.4 直流无刷电机

1.4.1 接线定义



图 1-8 电机接线定义

1.4.2 参数

表 1-4 交流伺服电机参数

电机型号	TS4607 N1680 E200
额定功率(W)	200
额定扭矩(N.m)	0.64
瞬时最大扭矩(N.m)	1.91
额定/最大转速(r/min)	3000/5000
Encoder 电机转子惯量(kg.m ²)	0.19×10 ⁻⁴
Resolver 电机转子惯量(kg.m ²)	0.2×10 ⁻⁴
每秒最大功率(kw/s)	21.5
机械常数(ms)	0.9
额定电流(A)	1.7
瞬时最大电流(A)	5
极对数(对)	4
绝缘等级	F 级
使用温度(℃)	0—40
保存温度(℃)	-10—85
耐震性	5G 以下
使用湿度	85%RH以下(不结露)
IP 等级	标准 IP42 (可支持 IP65)

1.4.3 结构特点



图 1-9 交流伺服电机基本结构图

如上图 1-9 所示,交流伺服电机主要由定子和转子构成。定子铁心通常用硅钢片叠压而成。定子铁心表面的槽内嵌有两相绕组,其中一相绕组是励磁绕组, 另一相绕组是控制绕组,两相绕组在空间位置上互差 90°电角度。工作时励磁绕组 与交流励磁电源相连,绕组上电,电机运动。

输入轴上装有玻璃制编码器圆盘,圆盘上印刷有能够遮住光的黑色条纹。圆 盘两侧有一对光源与受光元件,中间有一个分度尺元件。圆盘转动时,遇到玻璃 透明的地方光就会通过,遇到黑色条纹光就会被遮住。受光元件将光的有无转变 为电信号之后就成为脉冲(反馈脉冲)。"圆盘上条纹的密度=伺服电机的分辨率" 亦即"每转脉冲数",根据条纹可以掌握圆盘的转动量。

附录 A CCS 工程导入

1. 打开电脑桌面 DSP 编译软件 CCS,出现位置空间选择,直接电机确定。 选择 Project,点击 Import CCS Projects,如下图 A-1 所示;

CCS Edit - Code Composer Studio				
File Edit View Navigare	Project Run Scripts	Window Help		
📑 👻 🔛 🐚 🗐 🦂 🚽	 New CCS Project New Energia Sket Examples 		Quick Access	CCS Edit 🏷 CCS Debug
	 Build All Build All Build Configuratic Build Working Set Clean Build Automatical Show Build Setting Add Files RTSC Tools Import CCS Project Import Energia SI Import Energia SI Properties 	Ctrl+B ns + y 15 Sv3.3 Projects sv3.3 Projects etch oraries		
			🐥 🔶 🔀 📰 📰 📰	\$↓ ♂ ₽ • ○ • □
		CDT Build Console [PMSM_SD]		~

图 A-1 打开添加选项

14. 点击 Browse,在目录 C:\ProgramFiles\MATLAB\R2015b\workspace\pwm 下中先导入第一个小实验工程 BLDC,第二个小实验导入 PMSM。选择 pwm(提 供直流无刷电机实验 BLDC,提供交流伺服电机实验 PMSM)文件夹中的 BLDC_SD/PMSM_SD,然后点击确定,如图 A-2 所示。注意该案例是导入第二个 小实验的过程,第一个小实验与此只是名字不同,其他过程都相同。

	Import CCS Eclipse Projects		- • ×	
CCS Edit - Code Composer Studio	Select CCS Projects to Impor Select a directory to search f	t or existing CCS Eclipse projects.		X
□ ・ □ 10 0 1	 Select search-directory: Select archive file: Discovered projects: 		Browse	🗄 🖻 📴 CCS Edit 🦫 CCS Debug
		浏览文件夹	Select All	
	Copy projects into worksp	Select root directory of the p Common Commo	projects to import	
	?	Finish	Cancel	
				•
		<u></u>		Free License

图 A-2 添加工程示意图

2. 点击 Finish, 完成工程导入, 如图 A-3 所示;

	V Import CCS Eclipse Projects		
CCS Edit - Code Composer Studio	Select CCS Projects to Import Select a directory to search for existing CCS Eclipse projects.		×
File Edit View Navigate Project	Select a directory to search for existing Cos Eclipse projects: Select archive file: Discovered projects: PMSM_SD [C:\Users\HITSZ\Desktop\pwm\PMSM_SD] Automatically import referenced projects found in same search-ce Copy projects into workspace Open the Resource Explorer to browse available example projects. Finish	Browse Browse Select All Deselect All Refresh	
			-
	<u>e</u>		Free License

- 图 A-3 添加工程完成示意图
- 3. 工程导入成功,如图 A-4 所示;

😚 CCS Edit - Code Composer Studio		
File Edit View Navigate Project Run Scripts	Window Help	
🖻 • 🗟 🕤 🥵 • 🖉 💠 • 🖉 •		Quick Access
🖕 🕞 Project Explorer 💥 📄 🧐 🗸 🖓 🗖		
PMSM_SD [Active - Debug]		
	⊑ Console ⊠	÷ ↔ 🗐 💷 🔐 🖻 🛼 📑 📮 ▾ 😁 ▾ ▫ ▫
	CDT Build Console [PMSM_SD]	
د ۳MSM_SD	<u></u>	+ Free License

图 A-4 工程导入成功界面 4. 选择 Release 模式,如图 A-5 所示;

电机 PWM 控制与驱动电路实验指导书



图 A-5 选择 Release 模式

5. 进入 Release 模式后会报错,可以忽略报错继续操作,右键单击导入的工程文件,选择 Properties 选项,如 A-6 所示;

CCS Edit - Code Composer Studio						
File Edit View Na	wig	ate Project Run Scripts Wi	indow Help			
i 📑 🕶 🖾 👘 i 📮 i 4	5	• Ø 🏘 • 🛷 • 🔟 🍤	⇔ - ⇔ -		Quick Access 🔡 🔀 CCS Edit	CCS Debug
Project Explore	er S		鼠标右击	面目夕称		
PMSM S		Active Debug				
		New	+			
		Show In	۰.			
		Add Files				
0	P	Сору	Ctrl+C			
	13	Paste	Ctrl+V			
3	×	Delete	Delete			
		Refactor	•			
		Source				
		Move				
		Rename	F2			
		Import	•			
a c	Su	Export				
		Build Designt				
		Clean Project				
		Rebuild Project				
4	2	Refresh	E5			
	2	Close Project	13			
		Make largets	F		🔶 V 🖄 🔤 🖬 🔚 🖛 🖬 🖬 🕇	
		Index	e P	MSM_SD]		*
		build Configurations				
		Debug As	•			
		Compare With	•			
		Restore from Local History				
		Team	•			-
🚝 PMSM_SD 🛛 🧹		Properties	Alt+Enter	8.	Free Lice	ense

图 A-6 选择 Properties 选项 6. 选择 Include Options 选项,如图 A-7 所示;

Properties for PMSM_SD	
type filter text	Include Options 🗢 🔹 🗢
 Resource General Build C2000 Compiler Processor Ontions 	Configuration: Debug [Active]
 Optimization Debug Options Include Options Advanced Options C2000 Linker Debug 	Specify a preinclude file (preinclude) 🕢 🔊 중 🖓 🖓
	添加 删除
最	Add dir to #include search path (include_path, -I) DAPMSM\/common\/include* DAPMSM\/domib\/drviib* DAPMSM\/domib\/mathilb* DAPMSM\/mathilb* DAPMSM\/domib\/mathilb*
Show advanced settings	OK Cancel

电机 PWM 控制与驱动电路实验指导书

图 A-7 配置系统文件

8. 删除前四个默认路径(注意最后一个文件不能删除),如图 A-8 所示;

😳 CCS Edit - Code Compose	er Studio		×
Properties for PMSM_SD	a Sugari, Nav. Sulpti, Mindow State		
type filter text	Include Options		CS Edit 🏷 CCS Debug
 Resource General Build C2000 Compiler 	Configuration: Debug [Active]	▼ Manage Configurations	
Processor Options Optimization Debug Options Include Options Advanced Options C2000 Linker Debug	Specify a preinclude file (preinclude)	ହ ୟ ଅ ୫୮ ୫୮	
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Add dir to #include search path (include_path, -I) \$(CG_TOOL_ROOT)/include* 不可交件不能删除!!!!	ର ଛ ଇ ଚା ହା	
			oject.h"
Show advanced settings Show 20128	U35.ccxmi (Active)	OK Cancel	
	**** Build Finished ****		
PMSM_SD			Free License

图 A-8 删除 4 个默认路径

9. 添加新路,添加目录 C: \ProgramFiles\ MATLAB \R2015b \workspace \pwm\BLDC (第一个实验是 BLDC,第二个小实验是 PMSM)文件夹下面的四个路 径: \user\include、\common\include、\dmclib\drvlib、\dmclib\mathlib,如图 A-9 所

示;注意是在 pwm 文件下的 BLDC 或 PMSM 中的子文件中进一步添加。

😵 CCS Edit - Code Composer	r Studio			- x
Properties for PMSM_SD	August Nav. Scotte Wildow Mile-			
type filter text	Include Options		(→ → → →	CS Edit 🏷 CCS Debug
▷ Resource				
General	Configuration: Debug [Active]		Manage Configurations	
Build A C2000 Compiler	Configuration.		Wanage Configurations	
Processor Options				
Optimization 🛛	Add directory path		A A A A A A A A A A A A A A A A A	
Debug Options	Directory			
Advanced Options	Directory.			
⊳ C2000 Linker	() <u>-</u>			
Debug	Workspace	Variables Browse		
(刘顺文/#本	×			
Select a fo	lder from file system:			
		OK Cancel	<u>ଜ</u> ିଇ ଇ ନା ନା	
pwn				
4 1 co	mmon		出土东市北北西	
	nclude	Ile the life	里山称加延坝	
Þ 🌺 dn	nclib			
文件夹 (F): i	include			
新建文件表	夹(M) 确定 取消			oject.h" 🔺
(?) Show advanced			OK Cancel	
K 1MS320F280	35.ccxml [Active]			,
	**** Build Fin	nished ****		
PMSM_SD		<u>@</u>		Free License

图 A-9 添加 PMSM 文件

10. 添加完成后,单击 OK(注: 必须是全英文路径),如图 A-10 所示;



图 A-10 添加完成

11. 点击编译工具图标旁边的下三角图标 5. ,选择 Release 模式,等待编译

完成,如图 A-11 所示;

CCS Edit - Code Composer Studio		
File Edit View Navigate Project Run Scripts	Window Help	
· · · · · · · · · · · · · · · · · · ·		Quick Access 🗄 🖻 🕞 CCS Edit 🎭 CCS Debug
Project Explo		
更次选择Release	衰式,等待编译完成。)
	CDT Build Console (PMSM SD)	
	concerte framework	
	**** Build of configuration Release for project P	MSM_SD ****
	"C:\\ti\\ccsv6\\utils\\bin\\gmake" -k all gmake: Nothing to be done for 'all'.	
	**** Build Finished ****	
/05		
PMSM_SD		Free License

图 A-11 重新编译

12. 编译完成后,单击仿真图标,进入仿真模式,如图 A-12 所示;



图 A-12 进入仿真模式

13. 编译完成后,自动进入 Debug 测试界面,其三个符号分别表示运行-停止-退出功能,如图 A-13 所示。此时点击运行按钮,让程序持续处于运转状态,才能 进行接下来的操作。回到第一个实验的实验步骤 3,或者是第二个实验的步骤 3, 根据实际实验顺序操作。注意不要拔掉 USB 线,如果程序如下警告,重新按照指 导操作一次。

CCS Debug - PMSM_SD/main.c - Code Composer Studio					x
File Edit View Project Tools Run Scripts Window Help					
	2 - 2 3 4	• * Ø Ø * •	A +		
安行 、 退出		Qu	ick Access	CCS Edit 🧏 CCS De	bug
☆ Debug ⊠ 停止 ★ ▽ □	(x)= Variables 🛛 🙀	Expressions 1010 Register	s	- 0	
A SPMSM_SD [Code Composer Studio - Device Debugging]			約 🕫 🖻 🏟 🗿	***	
a 🧬 Texas Instruments XDS100v2 USB Emulator/C28xx (Suspended -	Name	Туре	Value	Location	8.7
main() at main.c:176 0x3F6084					
_args_main() at args_main.c:81 0x3F6F56					
c_int00() at boot28.inc:248 0x3F6D96 (the entry point was re					
Texas Instruments XDS100v2 USB Emulator/CLA_0 (Disconnected)					
۲. III کې او					
5 man.c 2					
177					
178 // Step 1. Initialize System Control:				-	
*				۴.	
E Console 🔀		÷ 🖓	s 📰 🚮 🥅 🖳 🔍 🖻	1 📮 🕶 🔂 🕶 🗖 🗖	
CDT Build Console [PMSM_SD]					
**** Build of configuration Release for project PMSM_SD ****				^	
"C:\\ti\\ccsv6\\utils\\bin\\gmake" -k all gmake: Nothing to be done for 'all'.					
**** Build Finished ****					
ê.	:			Free License	

图 A-13 Debug 仿真界面

附录 B 上位机实时内核设置

1. 点击 Simulation, 在下拉菜单中选择 Model Configuration Parameters (Simulation 参数设置),如图 B-1 所示。



图 B-1

2. 点击左侧菜单 Solver,设置仿真开始时间 Start time 为 0,结束时间 Stop time 为 inf; Solver options 菜单下, Type 选 Fixed-step, 求解器 Slover 选 odel (Euler) 算法,设置 Fixed-step size 为 0.01,如图 B-2 所示。

S Configuration Parameters: SMExp1	L/Configuration (Active)		
Category List			*
Select Solver Data Lipot/Export Diagnostics Hardware Implementation Model Referencing Simulation Target Code Generation	Simulation time Start time: 0.0 Solver options Type: Fixed-step	Stop time: inf Solver: odei (Euler) D. 01 Unconstrained SingleTasking	
<	ш	OK Cancel Help	, Apply

图 B-2

3. 点击左侧 Code Generation, 点击 Browse, 选择 sldrt.tlc (配置实时内核), 点击 OK 完成设置, 如图 B-3 所示。

Configuration Parameters: SMExp1	I/Configuration (Active)	
Category List		
Select:	Target selection	
Solver Data Import/Export	System target file: sldrt.tlc	Browse
> Optimization		
Hardware Implementation	Description: Simulink Deskton Real-Time	
Model Keferencing Simulation Target	Build susses	
Code Generation	Makefile configuration	
	V Generate makefile	
	Wake command: make rtw	
	Temnlate makefile: sldyt tmf	
	Code Generation Advisor	
	Select objective:	- Г
		Chash Madal
	Check model before generating code: Ull	Check model
	E Generate code only	Build
	Package code and artifacts Zip fil	.e name:
	m	•
0		mcel Heln Annly
🔁 S	ystem Target File Browser: SMExp1 📃	
Syst	tem Target File: Description:	
ide	link grt.tlc IDE Link GRT	*
rea	ltime.tlc Run on Target Hardware	
rsi	m.tlc Rapid Simulation Target	
The second se	efen tic S-Function Target	
	rt tla Simuliak Deakton Deal Tire	
510	simulink besktop kear-lime	
SID	Intert.tic Simulink Desktop Real-Time (ERT	
slr	t.tlc Simulink Real-Time	
slr	tert.tlc Simulink Real-Time (ERT)	-
sys	temverilog_dpi_ert.tlc SystemVerilog DPI Component Gen	=
sys	temverilog dpi grt.tlc SystemVerilog DPI Component Ger	
tlm	generator ert.tlc SystemC TLM Component Generator	
tlm	generator grt.tlc SystemC TLM Component Generator	
Full	Name: C:\ProgramFiles\MATLAB\R2015b\toolbox\sldrt\rtw\sldrt.	tlc
Full	Name: C:\ProgramFiles\MATLAB\R2015b\toolbox\sldrt\rtw\sldrt.	tlc
Full	Name: C:\ProgramFiles\MATLAB\R2015b\toolbox\sldrt\rtw\sldrt.	tlc

图 B-3

电机 PWM 控制与驱动电路实验指导书



图 B-4