

东华大学
硕士学位论文
直流无刷伺服电机的DSP控制
姓名： 彭军
申请学位级别： 硕士
专业： 机械设计及理论
指导教师： 管伯良
2003. 1. 1

直流无刷伺服电机的 DSP 控制

摘要

直流无刷伺服电机是一种无换向器的直流电机,随着现代工业的发展,直流无刷伺服电机得到越来越广泛的应用,对直流无刷伺服电机的控制经历了从模拟控制电路到以单片机为核心的数字控制电路的发展过程,目前主要是采用以单片机为主的控制方式。

但模拟电路控制和单片机控制直流无刷伺服电机都存在内在的缺陷,前者由于采用模拟器件容易老化而且对温度变化敏感;后者虽然克服模拟器件的内在缺陷,但单片机运算速度慢,难以实现现代工业对电机实时控制的要求。特别是随着控制理论的发展和现代工业应用中对高性能直流无刷伺服电机控制器的需求,一般的单片机或多片微处理器都不能满足复杂而先进的控制算法,对直流无刷伺服电机的控制性能不能达到比较高的控制精度要求。

本课题针对用单片机控制直流无刷伺服电机精度不高这一难点,提出一种基于 DSP 的直流无刷伺服电机的伺服控制系统,通过对可编程控制器件、专用芯片和单片机控制不同的控制方案对直流无刷伺服电机的控制性能比较分析,通过对不同控制环节的控制算法的比较论证,确立了一套以 DSP (TMS320LF2407) 为核心的直流无刷伺服电机的控制方案。其主要优势在于利用数字信号处理器 DSP 的高速数字运算功能,实现各种高效的控制算法,达到直流无刷伺服电机的精确控制的目的。

本控制系统主要采用三环控制,即电流内环、速度中环和位置外环,在高速有效的数字 PID 算法的控制下输出一定占空比的 PWM 波形,并根据位置信号的反馈,控制电子换向器的高速切换,来实现直流无刷伺服电机的转矩、速度和定位等一系列的控制。

本系统充分利用了 DSP 周边端口丰富,运算速度快的特点,使系统硬件结构简单,扩展性好,软件算法高效,兼容性好。调试结果表明,本系统在控制过程中速度调节范围大,转矩波动小,定位准确,是一种较为理想的控制方案。

关键词: 直流无刷伺服电机、数字信号处理器 (DSP)、脉宽调制 (PWM)、可编程逻辑器件 (CPLD)、闭环控制、数学模型、PID、CC (code composer)、位置传感器

Brushless DC Servo Motor based on DSP control

Abstract:

The brushless DC servo motor is a kind of DC motor, which has no electrical brush. With the development of the modern industry, the application of the brushless DC servo motor has been more and more widely used. The control of the brushless DC servo motor has been developed from analog circuit control to MCU based digital control, which recently is the main control method of the brushless DC servo motor.

Analog and MCU based control of the brushless DC servo motor, however, have their internal shortcomings. The former is sensitive to the temperature due to aging of the analog devices and the later can't meet the real-time-control requirement of the modern industry due to limited speed of the MCU though the problem of the aging of the analog device has been overcome. Especially, with the development of the modern control theory and requirement of the high performance controller of the brushless DC servo motor, MCU or MCUS of common type is unable to follow the high advanced and complicated algorithm so that the result can't reach the requirement of the high accuracy.

Aimed at the accuracy defect of the MCU control, the paper advanced a kind of brushless DC servo motor control system based on DSP. According to the comparison and analysis of the control performance of the PLD, special used chip and MCU and the argumentation of the different control algorithm, we build up the control system with the core of DSP(TMS320LF2407). The system mainly takes advantage of the ability of high-speed digital computing performed by DSP.

Based on the high-efficiency control algorithm of DSP, the aim of high-accuracy control of the brushless DC servo motor can be realized.

The control system mainly adopts three-loop control, which includes current inner loop, speed middle loop and position out loop. According to certain control algorithm, DSP outputs the PWM wave with certain duty, and realize the brushless DC servo motor control by shifting electronic switch with high speed according to the feedback signal of the position sensor.

The system makes full use of the abundance of peripheral port of DSP and its ability of high-speed computing so that the hardware has the easy structure and the flexible expandability as well as the software has high-efficiency algorithm and good compatibility. The result of tests shows that the control system has adjustable speed in wide range, small torque fluctuation and high-accuracy orientation. So it deserves an ideal control system.

The application of the DSP based motor control is a new filed and presently it remains in the stage of research and development. The brushless DC servo motor controller based on DSP has not reached a batch production.

Keywords: Brushless DC Servo Motor、Digital Signal Processor(DSP)、PWM、CPLD、close-loop control、mathematic model、CC (code composer)、position sensor

第一章 前言

1.1 课题的来源、目的与意义

本课题为应用性研究课题, 研究开发以 DSP 为核心的直流无刷伺服电机控制器, 研究成果可直接应用于现代工业各领域的自动定位控制系统, 或作为通用的直流无刷伺服电机的控制器推向市场。

电机控制是 DSP 问世以来主要的应用领域之一, 本人在研究生在读期间, 自主创业, 成立了上海三意电子科技有限公司, 主要应用 DSP 开发电机控制方面的项目和产品。公司致力于在国内推动 DSP 市场, 普及 DSP 技术, 帮助用户尽快将 DSP SOLUTIONS 转化为 DSP 产品, 以推动国内 DSP 产业的发展! 本课题为上海三意电子科技有限公司开发和研究, 具有广阔的市场前景和经济效益。

直流无刷伺服电机控制属于专业性较强的课题, 是对本专业研究生阶段所学知识的较全面训练。通过理论与实践相结合的研究, 大大提高了动手能力和创新能力, 研究成果经现场检验, 测试, 可用于生产实际。

本课题提出的一种基于 DSP 的直流无刷伺服电机控制方法, 将人们从传统的单片机控制领域中解放出来, 转向现代高速数字信号处理 DSP 芯片的数字控制设计, 通过 DSP 本身自带的电机控制的接口功能, 结合 DSP 的高速数字控制算法, 使得直流无刷伺服电机的控制精度达到比较高的层次。随着直流无刷伺服电机应用领域的不断增加, 本研究成果可在各领域的自动定位控制系统中得到广泛的应用。

1.2 直流无刷伺服电机控制的现状

直流无刷伺服电机的控制器虽然在市场上已存在多时, 但市场上的同类产品性能、价格、稳定性等方面往往都难以满足用户的需求。特别是在当今技术开放式的市场环境下, 不同厂商所生产的直流无刷伺服电机的控制模块大多是大同小异, 在技术上几乎没有什么创新与变革。对于一些传统工业, 它们虽然对直流无刷伺服电机控制模块的设计上没有什么特殊的要求, 但是, 随着现代电子技术的飞速发展, 特别是电子产品集成化、模块化的设计思想的迅速崛起, 人们对电子产品的技术性能指标的要求也越来越

高。由于直流无刷伺服电机是一种无换向器的直流电机，可以有效地克服有刷直流电机由于换向火花所引起的转矩波动与电磁干扰等问题，而且还不失有刷直流电机易于控制的特点。所以随着直流无刷伺服电机在相关工业中的推广使用，人们对其控制性能要求也不断提高。因此，研制高性能的直流无刷伺服电机的控制系统成为一项普遍关注的课题。

直流无刷伺服电机的控制经历了从模拟控制电路到以单片机为核心的数字控制电路的发展过程，但都存在内在的缺陷^[1]。前者由于采用模拟器件容易老化而且对温度变化敏感；后者虽然克服模拟器件的内在缺陷，但运算速度慢，难以实现现代工业对电机实时控制的要求。

在 DSP 问世之前，设计者进行传统的电机控制主要使用 2 种组合：用便宜的控制器控制昂贵的直流有刷电机，或用复杂的控制器控制交流电机。随着控制理论的发展，为了研制高性能控制器，一般的单片机或多片微处理器都不能满足复杂而先进的控制算法，使直流无刷伺服电机的控制不能达到比较高的精度要求。自 DSP 出现以来，电机控制成为 DSP 主要应用领域之一^[2]。

目前，国内外对 DSP 在电机控制方面的应用尚处于研发阶段，作为产品，以 DSP 为核心的直流无刷伺服电机的控制器尚未形成成熟的市场，研究与开发以 DSP 为核心的直流无刷伺服电机的控制器是电机控制方面的一个较新的领域。

1.3 课题完成的主要内容

本课题通过对用可编程控制器、专用芯片和单片机几种不同控制方案的性能分析比较，以及对相应控制算法的论证，确立了一套以 DSP（TMS320LF2407）为核心的直流无刷伺服电机控制方案。

本人进行了产品硬件的设计、制作，软件的编程、调试，进行了软件的模拟仿真，来确定 PID 算法的控制参数，最后进行软硬件结合的调试，得出实际的速度、位置、电流参数和波形。

调试结果表明，本系统在控制过程中速度调节范围大，转矩波动小，定位精确，是一种较为理想的控制方案。

第二章 直流无刷伺服电机工作的基本原理

2.1 直流无刷伺服电机的结构原理

直流无刷伺服电机的结构原理图如图 2-1 所示：

直流无刷伺服电机主要由电动机本体、位置传感器和电子开关电路三部分组成。电动机本体在结构上与永磁同步电动机相似，但没有笼型绕组和其他起动装置。其定子绕组一般制成多相（三相、四相、五相不等），转子由永久磁钢按一定极对数（ $2p=2, 4, \dots$ ）组成。图 2-1 中的电动机本体为三相两极，三相定子绕组分别与电子开关线路

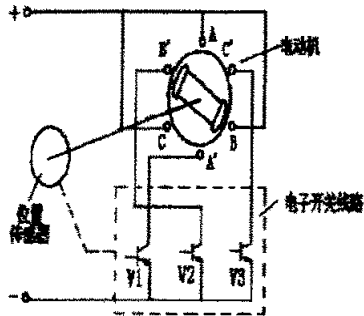


图 2-1 直流无刷伺服电机的结构原理图
A 相、B 相、C 相绕组分别与功率开关管 V1、V2、V3 相接。位置传感器的跟踪转子与电机转轴相联接^[1]。

定子绕组的某一相通电时，该电流与转子永久磁钢的磁极所产生的磁场相互作用而产生转矩，驱动转子旋转，再由位置传感器将转子磁钢位置变换成电信号，去控制电子开关线路，从而使定子各相绕组按一定次序导通，定子相电流随转子位置的变化而按一定的次序换相。由于电子开关线路的导通次序是与转子转角同步的，因而起到了机械换向器的换向作用。

所以，所谓直流无刷伺服电机，就其基本结构而言，可以认为是一台由电子开关线路、永磁式同步电机以及位置传感器三者组成的“电动机系统”。其原理框图如图 2-2 所示。

电机转子的永久磁钢与永磁有刷电机中所使用的永久磁钢的作用相似。均是在电机的气隙中建立足够的磁场，其不同之处在于直流无刷

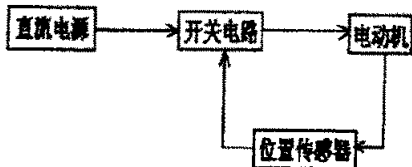


图 2-2 直流无刷伺服电机的原理框图
服电机中永久磁钢装在转子上，而直流有刷电机的磁钢装在定子上，图 2-3 示出了典型直流无刷伺服电机本体基本结构图。

直流无刷伺服电机电子开关线路是用来控制电机定子上各相绕组通电的顺序和时间，主要由功率逻辑开关单元和位置传感器信号处理单元两个部分组成。功率逻辑开关单元是控制电路的核心，其功能是将电源的功率以一定逻辑关系分配给直流无刷伺服电

机定子上各相绕组，以便使电机产生持续不断的转矩。而各相绕组导通的顺序和时间主要取决于来自位置传感器的信号。但位置传感器所产生的信号一般不能直接用来控制功率逻辑开关单元，往往需要经过一定逻辑处理后才能去控制逻辑开关单元。综上所述，组成直流无刷伺服电机各主要部件的框图，如图 2-4 所示。

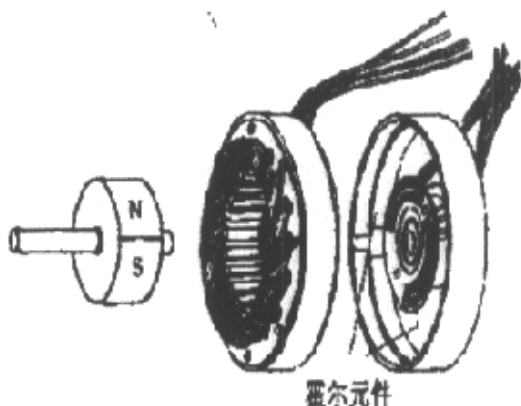


图 2-3 直流无刷伺服电机的基本结构

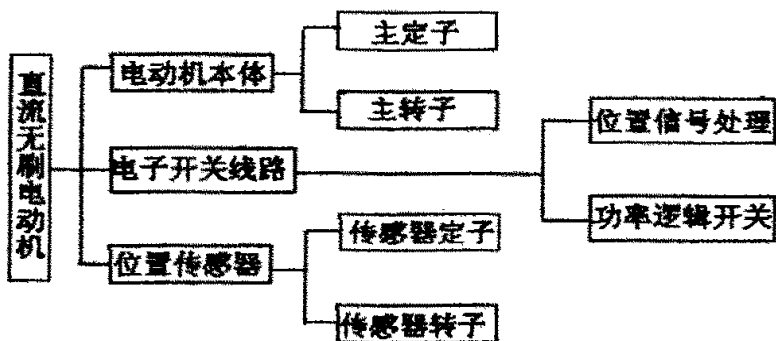


图 2-4 直流无刷伺服电机的组成框图

2.2 直流无刷伺服电机系统的数学模型:

所谓直流无刷伺服电机系统，就是将电机本体、逆变驱动电路和电流反馈控制电路作为一个整体加以考虑。建立直流无刷伺服电机系统的数学模型，就是准确地描述系统的各组成部分，并将它们有机地结合，给出系统的完整描述。

2.2.1 直流无刷伺服电机系统的运行原理

正如前面所说，直流无刷伺服电机系统由三部分组成：永磁同步电机、转子磁极位置传感器及其变换电路、逆变器。图 2-5 示出典型最基本的三相无刷直流力矩电机系统结构。图中以单电流环为例，其基本工作原理是：永磁直流无刷伺服电机工作在自同步状态，其三相绕组所加交流电压是通过逆变器提供的；逆变器所产生的交流电压的频

率和相位是由转子磁极位置传感器及其变换电路决定的,幅值是由输入指令决定的。具体关系可以通过以下分析说明^[4]。

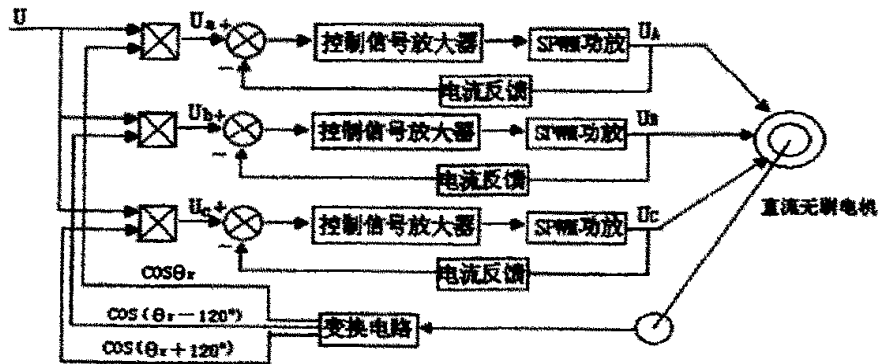


图 2-5 三相直流无刷伺服电机系统框图

转子磁极位置传感器及其转换电路提供一组单位正弦信号

$$\begin{bmatrix} P_a & P_b & P_c \end{bmatrix} = \begin{bmatrix} \cos\theta_r & \cos(\theta_r - 120^\circ) & \cos(\theta_r + 120^\circ) \end{bmatrix} \quad (2-1)$$

式中, θ_r 是转子旋转电角度, P_a 、 P_b 、 P_c 是一组反映转子磁极位置的单位正弦信号。

经逆变器中的乘法器, 实现输入指令与三相单位正弦信号的合成, 有:

$$\begin{bmatrix} u_a & u_b & u_c \end{bmatrix} = u_i \begin{bmatrix} P_a & P_b & P_c \end{bmatrix} \quad (2-2)$$

式中, u_i 为输入指令, u_a 、 u_b 、 u_c 是与 u_i 对应的三相交流信号。在系统没有各相电枢电流反馈的情况下, 该系统的电压平衡方程式为:

$$\begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} = \begin{bmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{bmatrix} \begin{bmatrix} i_A \\ i_B \\ i_C \end{bmatrix} + \begin{bmatrix} p\psi_A \\ p\psi_B \\ p\psi_C \end{bmatrix} \quad (2-3)$$

式中, u_a 、 u_b 、 u_c 为加到三相电枢绕组端电压, p 为微分算子, R_s 是电枢绕组相电阻, i_A 、 i_B 、 i_C 为三相电枢绕组相电流; ψ_A 、 ψ_B 、 ψ_C 为三相磁链。由图 2-5 可见三相电压控制信号到绕组端电压经过了控制信号放大器、功率放大器。即:

$$\begin{bmatrix} u_A & u_B & u_C \end{bmatrix} = k \begin{bmatrix} u_a & u_b & u_c \end{bmatrix} \quad (2-4)$$

式中, k 是逆变器前向通道电压增益, 且 $k=k_1*k_2$, 其中 k_1 是控制信号放大倍数, k_2 是功率放大器放大倍数, $k_2=1/U_{DC}/(2k_1 \times U_i)$, U_{DC} 是桥臂直流电压。当各相加入电枢电流反馈后, 该系统的电压平衡方程式为:

$$\begin{bmatrix} u_A & u_B & u_C \end{bmatrix} = k u_i \begin{bmatrix} u_a & u_b & u_c \end{bmatrix} - k k_f \begin{bmatrix} i_A & i_B & i_C \end{bmatrix} \quad (2-5)$$

式中, k_f 是电流反馈系数。与(2-3)式对比可知:

$$\begin{bmatrix} u'_A \\ u'_B \\ u'_C \end{bmatrix} = \begin{bmatrix} R_s + k k_f & 0 & 0 \\ 0 & R_s + k k_f & 0 \\ 0 & 0 & R_s + k k_f \end{bmatrix} \begin{bmatrix} i_A \\ i_B \\ i_C \end{bmatrix} + \begin{bmatrix} p\psi_A \\ p\psi_B \\ p\psi_C \end{bmatrix} \quad (2-6)$$

式中, u'_A 、 u'_B 、 u'_C 为加入电流负反馈后, 等效外加绕组电压。对比(2-3)与(2-6)式可见, 加入电流负反馈后, 对应所加入信号可以视为绕组电阻增大。

2.2.2 直流无刷伺服电机系统的数学模型^[3]

系统所采用的永磁直流无刷伺服电机转子磁钢采用表面粘贴, 等效励磁线圈产生的空间磁势正弦分布。图 2-6 是一台二极永磁直流无刷伺服电机简图。在 dq 坐标系中, 取永磁体基波磁场的方向为 d 轴, 而 q 轴顺着旋转方向超前 d 轴 90° 电角度。转子参考坐标以 q 轴与固定轴线(A 相绕组轴线)间的电角度 θ_r 来确定。在上述坐标下, 以转子参考坐标表示的电压方程为:

$$u_q = (R_s + k k_f) i_q + p\psi_q + \omega_r \psi_d \quad (2-7)$$

$$u_d = (R_s + k k_f) i_d + p\psi_d + \omega_r \psi_q \quad (2-8)$$

磁链方程为:

$$\psi_q = L_q i_q \quad (2-9) \quad \psi_d = L_d i_d + \psi_f \quad (2-10)$$

以上各式中, u_d 、 u_q 分别是 d、q 轴电压, i_d 、 i_q 分别是 d、q 轴电流, ψ_d 、 ψ_q 分别是 d、q 轴磁链, ω_r 是转子电角速度, ψ_f 是永磁体基波磁势链过定子绕组的磁链, L_q 、 L_d 分别是 q、d 轴电感, 且有 $L_q = L_s + L_{mq}$, $L_d = L_s + L_{md}$ 。

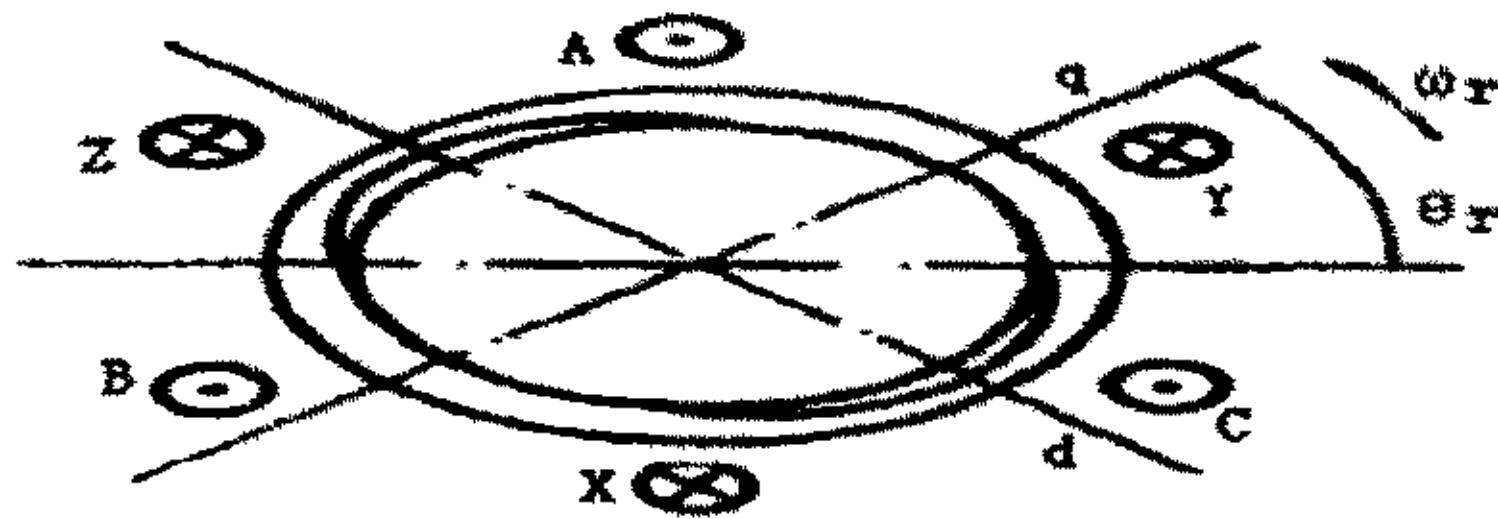


图 2-6 永磁直流无刷伺服电机结构示意图

在转子表面粘贴磁钢的结构中, d、q 轴主电感都趋向零, 即 $L_{d0}=L_{q0} \rightarrow 0$, 所以 d、q 轴电感即是绕组漏感, $L_d=L_q=L_s$ 。将 (2-9)、(2-10) 式代入 (2-7)、(2-8) 式, 且 $p\psi_r=0$, $\omega_r\psi_r=e_r$ (旋转电势), 则有:

$$u_q = (R_s + k k_f) i_q + L_s p i_q + \omega_r L_s i_d + e_f \quad (2-11)$$

$$u_d = (R_s + k k_f) i_d + L_s p i_d + \omega_r L_s i_q \quad (2-12)$$

对应的机械方程式和力矩方程式为:

$$T_e = \frac{J}{P} \frac{d^2 \theta_r}{dt^2} + T_L \quad (2-13)$$

$$T_e = \sqrt{3} k_e i_q \quad (2-14)$$

式中, T_e 是电磁转矩, J 是电机与负载转动惯量, P 是极对数, T_L 是负载转矩。

上述数学模型中的 d、q 坐标系中的参数与 A、B、C 坐标系各量关系可以通过变换矩阵确定。需要说明的是在直流无刷伺服电机系统工作过程中, 为了获得最大电磁力矩和正反向运行的对称性, 一般转子位置传感器及其转换电路所决定的三相交流电压相位应有 $\theta_e = \theta_r$, 即 θ_r 决定的空间相量和 θ_e 决定的时间相量有相同的相位关系。在满足以上条件的情况下, 有:

$$u_q = \sqrt{\frac{2}{3}} U_m [\cos \theta_r \cos \theta_e + \cos(\theta_r - 120^\circ) \cos(\theta_e - 120^\circ) + \cos(\theta_r + 120^\circ) \cos(\theta_e + 120^\circ)] \quad (2-15)$$

$$u_d = \sqrt{\frac{2}{3}} U_m [\sin \theta_r \cos \theta_e + \sin(\theta_r - 120^\circ) \cos(\theta_e - 120^\circ) + \sin(\theta_r + 120^\circ) \cos(\theta_e + 120^\circ)] \quad (2-16)$$

$$i_q = \sqrt{\frac{2}{3}} I_m [\cos \theta_r \cos(\theta_e - \gamma) + \cos(\theta_r - 120^\circ) \cos(\theta_e - 120^\circ - \gamma) + \cos(\theta_r + 120^\circ) \cos(\theta_e + 120^\circ - \gamma)] \quad (2-17)$$

$$i_d = \sqrt{\frac{2}{3}} I_m [\sin \theta_r \cos(\theta_e - \gamma) + \sin(\theta_r - 120^\circ) \cos(\theta_e - 120^\circ - \gamma) + \sin(\theta_r + 120^\circ) \cos(\theta_e + 120^\circ - \gamma)] \quad (2-18)$$

$$E_q = \sqrt{\frac{2}{3}} E_m [\cos \theta_r \cos \theta_e + \cos(\theta_r - 120^\circ) \cos(\theta_e - 120^\circ) + \cos(\theta_r + 120^\circ) \cos(\theta_e + 120^\circ)] \quad (2-19)$$

整理后有 $u_d = 0$,

$$u_q = \sqrt{\frac{3}{2}} U_m = \sqrt{3} U \quad (2-20)$$

$$i_q = \sqrt{\frac{3}{2}} I_m \cos \gamma = \sqrt{3} I \cos \gamma \quad (2-21)$$

$$i_d = \sqrt{\frac{3}{2}} I_m \sin \gamma = \sqrt{3} I \sin \gamma \quad (2-22)$$

$$E_f = \sqrt{\frac{3}{2}} E_m = \sqrt{3} E \quad (2-23)$$

其中, γ 是 u 与 i 的夹角, U_m 是相电压幅值, I_m 是相电流幅值, U 是相电压有效值, I 是相电流有效值, E_m 是相电势幅值, E 是相电势有效值。

2.2.3 直流无刷伺服电机系统的传递函数

将 (2-3) 式进行拉氏变换, 有:

$$u_q(s) = (R_s + k k_f) I_q(s) + L_s S I_q(s) + \omega_r L_s I_d(s) + E_f(s) \quad (2-24)$$

可推出:

$$\frac{I_q(s)}{u_q(s) - E_f(s) - \omega_r L_s I_d(s)} = \frac{1}{(R_s + k k_f) + L_s S} \quad (2-25)$$

由前面分析, $I_d(s) = I_q(s) \tan \gamma$, 根据 (2-5)、(2-23) 式可以画出无刷直流力矩电机系统的传函框图, 如图 2-7 所示。该图是对无刷直流力矩电机系统传函的精确描述, 该模型具有如下特点:

1. 该传函由 dq 坐标电机系统动态数学模型经拉氏变换得到, 因此各量都是与转子转角无关的直流量。

2. 该传函反映出 I_q 引起的交轴电抗压降。

3. 该传函反映出 I_q 与 ω_r 的耦合关系, 同时也说明该传函的非线性。

对于上述无刷直流力矩电机传函的精确描述, 在系统设计中可以利用系统软件进行分析。在满足一定条件下, 也可以将其进行简化。当 γ 很小时, 图 2-7 可以简化成图 2-8, 即忽略电抗压降影响。且当机电时间常数较电气时间常数大很多时, 则电气时间常

数为:

$$T_i = \frac{L}{R_s + k k_f} \quad (2-26)$$

同时在满足 $L_s = R_s + k k_f$ 时, 机电时间常数可以写为:

$$T_m = \frac{J(R + k k_f)}{3 k_e^2} \quad (2-27)$$

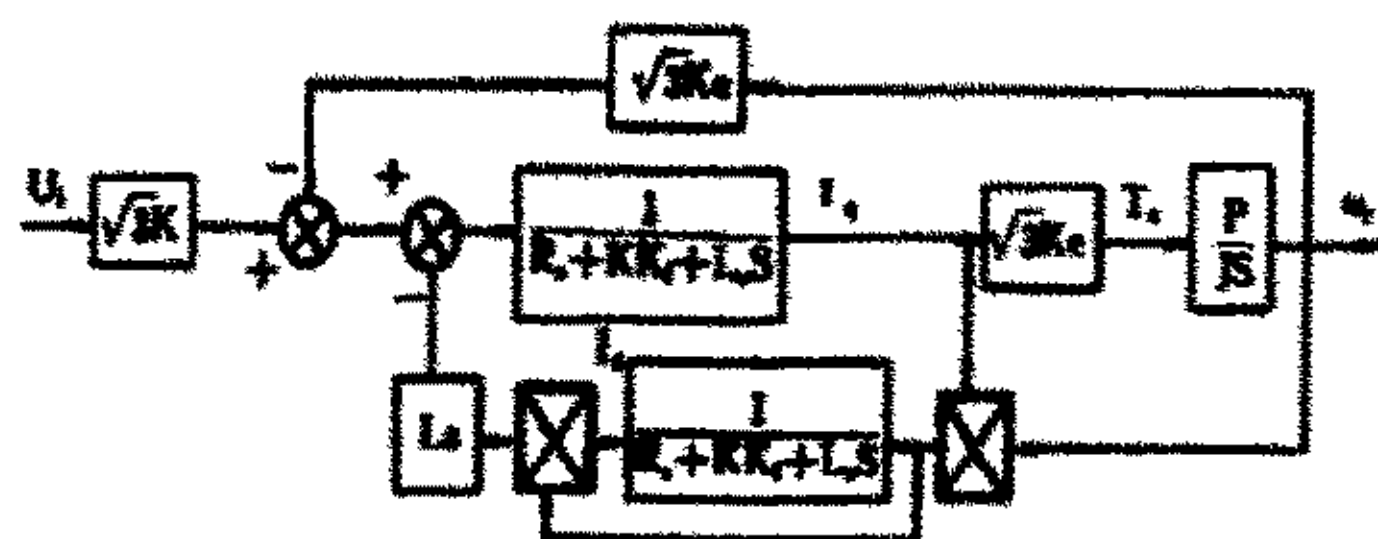


图 2-7 直流无刷伺服电机传递函数框图

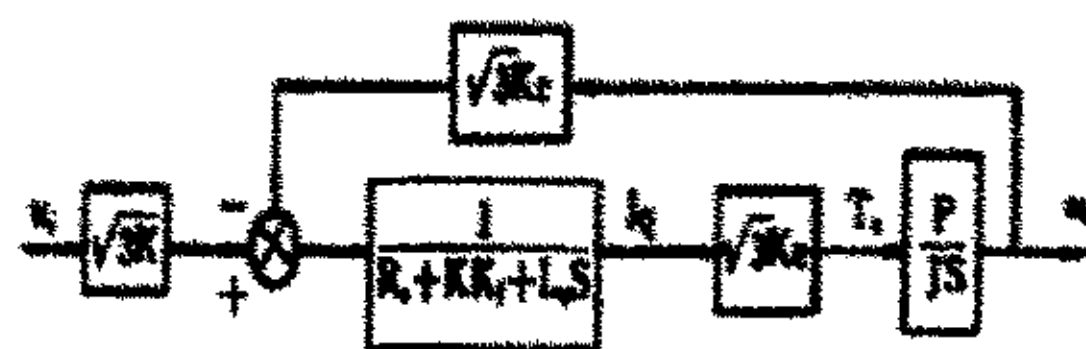


图 2-8 简化的传递函数框图

第三章 控制方案论证

3.1 控制器件的选择

3.1.1 中央控制单元

中央控制单元是直流无刷伺服电机控制的核心,它性能的优劣直接影响到最终系统的控制效果。从实用性角度来说,好的中央控制单元性能稳定,易于维护,可移植性好,而且控制方便,效率高。所以在中央控制器的选择上必需慎重考虑。

3.1.1.1 可编程控制器件,专用芯片和单片机控制

由可编程控制器件能构成时序控制单元,完成对直流无刷伺服电机的全部时序逻辑的控制。但这种控制方式对编程者是极大的考验,这要求编程者对整个直流无刷伺服电机时序控制有非常清楚的了解,甚至最基本最底层的设计都要有编制硬件电路来完成,这无疑加重了设计者的负担

直流无刷伺服电机的专用控制芯片有很多种,如摩托罗拉高性能微控制器(MCU)中的智能化外围模块 TPU^[17]等。但这些专用控制芯片因为是为专门控制而设计,所以从控制范围上来讲受到了很大的限制。它们对不同结构不同容量的直流无刷伺服电机显得无能为力,而在控制精度上由于受到了自身硬件结构的限制,没有更大得改进余地。

单片机的种类很多有 51 系列,96 系列,PIC,ST,MSP430 系列等等,其中以 51 系列单片机作为普遍。但 51 单片机由于受到本身结构的限制(内部 12 分频),对于 12M 的外部晶振只能提供 1MIPS 的指令执行速度,在对直流无刷伺服电机的控制上,特别是对逆变器的开关控制上,单片机控制达不到很高的开关速度,这样对直流无刷伺服电机来说会产生很大的转矩波动,还有就是由其组成的闭环控制系统,由于速度受到了一定的限制,所以在控制精度上达不到很高的要求。

3.1.1.2 数字信号处理器(DSP)控制

数字信号处理器 DSP 由于采用了不同的内部结构,执行速度上相对于单片机有很大

的提高,由于其内部能高速执行各种运算指令,所以在控制效率上远远高于单片机控制。在对直流无刷伺服电机的控制方面,由于有了专用工业控制用的 DSP 的支持,使其在控制精度上达到了一个新的高度,而且此类 DSP 所提供的各种接口(如 PWM 输出, AD 输入, SCI, SPI, I²C 串行通讯接口, CAN, USB 等等)使得其在电机控制领域独领风骚,独占鳌头。DSP 因其强大的运算功能,对直流无刷伺服电机可进行无传感器控制,减少系统元器件的数量,降低成本,其优势很多:

- 可运行高品质的控制算法,减少力矩波动,从而降低振动和噪声
- 减少谐波干扰,从而降低滤波电路的成本;
- 可运行无传感器的算法,省去速度和位置传感器;
- 可执行神经网络和模糊逻辑控制,用于控制多变量和复杂的系统。

——TMS320LF2407^[7]

TMS320 系列得 DSP 的体系结构专为实时信号处理而设计,该系列的 DSP 控制器将实时处理的能力和控制器外设功能集于一身,为控制系统应用提供了理想的解决方案。其灵活的指令集,内在的操作灵活性,高速的运算能力,改进的并行结构和低成本使得该系列成为很多处理应用的理想选择。

TMS320LF2407 是 TI 公司专为电机控制设计的基于 20MIPS 的定点 DSP 芯核的单片控制器,其指令兼容 TMS320C2XX 系列,向上兼容 TMS320C5X 系列,具有良好的可移植性。其主要特点有:

- 采用高性能静态 CMOS 技术,使得供电电压降为 3.3V,减小了控制器的功耗; 30MIPS 的执行速度使得指令周期缩短到 33ns (30MHz),从而提高了控制器的实时控制能力。

- 片能高达 32K 字的 FLASH 程序存储器,高达 1.5K 字的数据/程序 RAM, 544 字双口 RAM (DRAM) 和 2K 的单口 RAM (SARAM)。

- 两个事件管理器模块 EVA 和 EVB,两个 16 位通用定时器; 8 个 16 位的脉宽调制 PWM 通道。它们能够实现:三相反相器控制; PWM 的对称和非对称波形;当外部引脚 PDPINTx 出现低电平时快速关闭 PWM 通道;可编程的 PWM 死区控制以防止上下桥臂同时输出触发脉冲; 3 个捕获单元;片内光电编码器接口电路; 16 通道 A/D 转换器。事件管理器模块适用于控制交流感应电机、直流无刷伺服电机、开关磁阻电机、步进电机、多级电机和逆变器。

- 可扩展的外部存储器总共 192K 字; 64K 字的程序存储器; 64K 字数据存储器;

64K 字 I/O 寻址空间。

- 看门狗定时器模块 (WDT)。
- 10 位 A/D 转换器最小转换时间为 500ns, 可选择由两个事件管理器来触发的两个 8 通道输入 A/D 转换器或一个 16 通道输入的 A/D 转换器。
- 控制器局域网络 (CAN) 2.0B 模块。
- 串行通道接口 (SCI)。
- 16 位的串行外设接口模块 (SPI)。
- 基于锁相环的时钟发生器。
- 高达 40 个可单独编程或复用的通用输入/输出引脚 (GPIO)。
- 5 个外部中断 (电机驱动保护、复位和两个可屏蔽中断)。
- 电源管理包括 3 种低功耗模式, 并且能独立将外设器件转入低功耗模式。

TMS320LF2407 采用哈佛结构, 独立的读写数据总线和地址总线, 支持并行的程序和数据操作数寻址, 50ns 的指令周期, 单周期的乘加指令, 可进行许多复杂的运算。其结构图如图 3-1 所示。

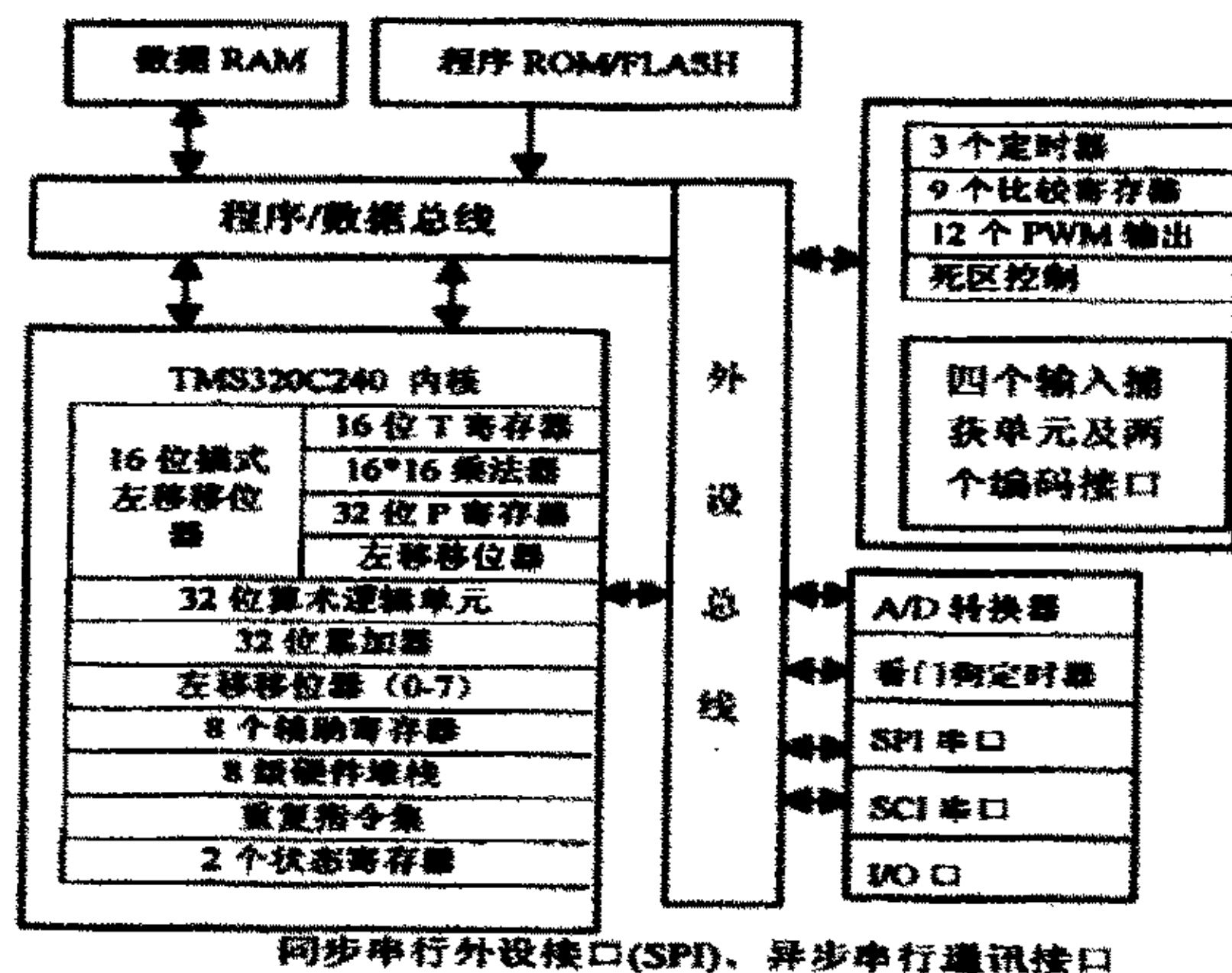


图 3-1 TMS320LF2407 结构图^[7]

TMS320LF2407 可以为高性能传动控制提供先进可靠高效的信号处理与控制的硬件。LF2407 集高速运算与面向电机的高效控制能力于一体, 可以用软件取代模拟器件, 方便地修改控制策略, 修正控制参数, 兼具故障检测、自诊断和上位机管理与通信等功能。LF2407 包含双 10 位模数转换器、同步串行外设接口 SPI, 异步串行通信接口 SCI、PWM 控制的管理器。其 PWM 波形生成单元包含可编程死区控制, 可输出非对称 PWM

波形、对称 PWM 波形和空间矢量 PWM 波形。

TMS320LF2407 也提供通过外围接口进行操作的第三种空间——输入输出。LF24x 使用了一个外围设备总线以支持对大量外设的可选性,外围设备总线用数据总线寻址并通过一个特殊的模块与数据总线接口,所以所有可对数据空间进行操作的指令均可以对所有的外围设备寄存器进行操作。

分离的程序空间和数据空间允许对指令和数据同时操作,譬如当数据正在相乘时,先前的结果可以加到加法器中,同时可以产生一个新的地址,这种并行机制支持在一个机器周期中完成的算法、逻辑和位寻址。LF2407 同时也包括了控制机制、中断管理、重复执行、函数/子程调用等控制机制。

由此可见,采用数字信号处理器(TMS320LF2407)作为中央控制单元是比较理想的选择。

3.1.2 功率放大器件的比较选择

驱动级的功率放大器件有中功率晶体管、大功率晶体管、大功率达林顿晶体管、可控硅、可关断可控硅、场效应功率管、双极型晶体管与场效应管的复合管以及各种功率模块。

3.1.2.1 达林顿晶体管, 可控硅, 场效应功率管, 大功率晶体管 (IGBT) [11]

达林顿晶体管(Dar)是一种复合三极管,一般由二个三极管复合而成。前一个晶体管的发射极深入后一个晶体管的基极,这样的复合管的放大倍数就相当于两个晶体管放大倍数的乘积,所以其电流放大倍数可达千倍以上,即使在开关状态也可达百倍以上。因此,这种晶体管只需很小的基极电流就可以产生很大的输出电流。但这种晶体管的主要缺点是饱和时管压降稍大,因此损耗要大一些。达林顿晶体管在电路中的使用方法与一般晶体管相同,也采用与一般晶体管完全相同的图形符号。

可控硅是一种脉冲触发的开关器件,它的突出优点是输入功率小、输出功率大、耐压高、成本较低。但是,可控硅虽然触发简单,但关断困难,虽然后来又发展了可关断可控硅(GTO),但总的来看线路复杂。同时,也易形成误触发,降低了可靠性。

场效应功率管(MOSFET)是新发展起来的功率器件。这种器件属于电压控制的功率放大器件,有很高的输入阻抗,用小的电压信号就可以控制很大的功率。但 MOSFET 在高

压大电流工作时，导通电阻大，器件发热厉害，输出功率下降，使得目前单只 MOSFET 很难输出 500W 以上的功率，从而限制了其应用范围。

绝缘基极双极型大功率晶体管（IGBT）则是集 MOSFET 的电压控制与双极型大功率晶体管的大电流、低导通电阻的特点于一体的新型复合场控器件，但分立元件的 IGBT 体积大，成本高，而且可靠性也受到很大的限制。

3.1.2.2 逆变功率驱动模块

功率驱动模块是将驱动用大功率晶体管，推动级晶体管及与前级的信号放大、隔离、耦合等功能线路都集成在一起、形成具有较强功能、较大功率输出的复合器件。使用这种器件制造直流无刷伺服电机驱动器，具有结构简单、性能稳定、运行可靠等优点。

比较典型的一种为智能功率驱动模块（IPM），其内部集成了信号放大，信号隔离，功率驱动，逆变桥及相关保护电路。这种功率模块为外设提供了非常方便的接口，可以通过小信号直接驱动大功率器件，非常适合于微机控制。用户无需了解驱动器的内部结构，只要知道相关的输入输出接口的特性，就能直接对目标外设进行控制。由于 IPM 内部集成的丰富的信号转换和驱动保护的功能，所以使用 IPM 模块可以大大简化目标系统的软件设计与硬件设计，这样不但方便了开发人员的设计而且在目标系统的成本上也得到一定程度的降低。

由此可见，前述四种元气件，都是以分离元件为主，分离元件的最大缺点是结构比较复杂，而且稳定性不高，功耗大。而直流无刷伺服电机一般都由多绕组构成，如果是全桥控制的话一般就需要为数不少的功率驱动元件。如果采用分离元件的话，势必造成驱动电路的繁琐复杂，占用体积大，产生的热量不易散发，控制起来就更加困难。基于以上考虑，再加上本系统所选用的是 DSP 控制控制信号驱动能力比较弱，所产生的控制电流不可能很大，如果除控制芯片外，再加上放大、隔离、耦合等电路的话势必会造成电路的复杂性。所以，在这里采用功率驱动模块是比较理想的选择。

3.1.3 驱动方式比较论证

直流无刷伺服电机的驱动方式主要有半桥驱动，电容储能型(c-dump)驱动，全桥驱动，电流型两象限斩波驱动等。

3.1.3.1 半桥驱动

常见的三相半控电路如图 3-2 所示, 图中, L_a 、 L_b 和 L_c 为电机定子 A、B、C 三相绕组, VF1、VF2 和 VF3 为三只 MOSFET 功率管, 主要起开关作用。H1、H2 和 H3 为来自转子位置传感器的信号。在三相半控电路中, 要求位置传感器的输出信号 1/3 周期为高电平, 2/3 周期为低电平, 并要求各传感器信号之间的相位差也是 1/3 周期。

三相半控电路的特点是简单。但电动机本体的利用率很低, 每个绕组只通电 1/3 时间, 另外 2/3 时间处于开断状态, 没有得到充分的利用。使用半桥驱动在运行过程中电机转矩的波动较大, 特别在低速时可能

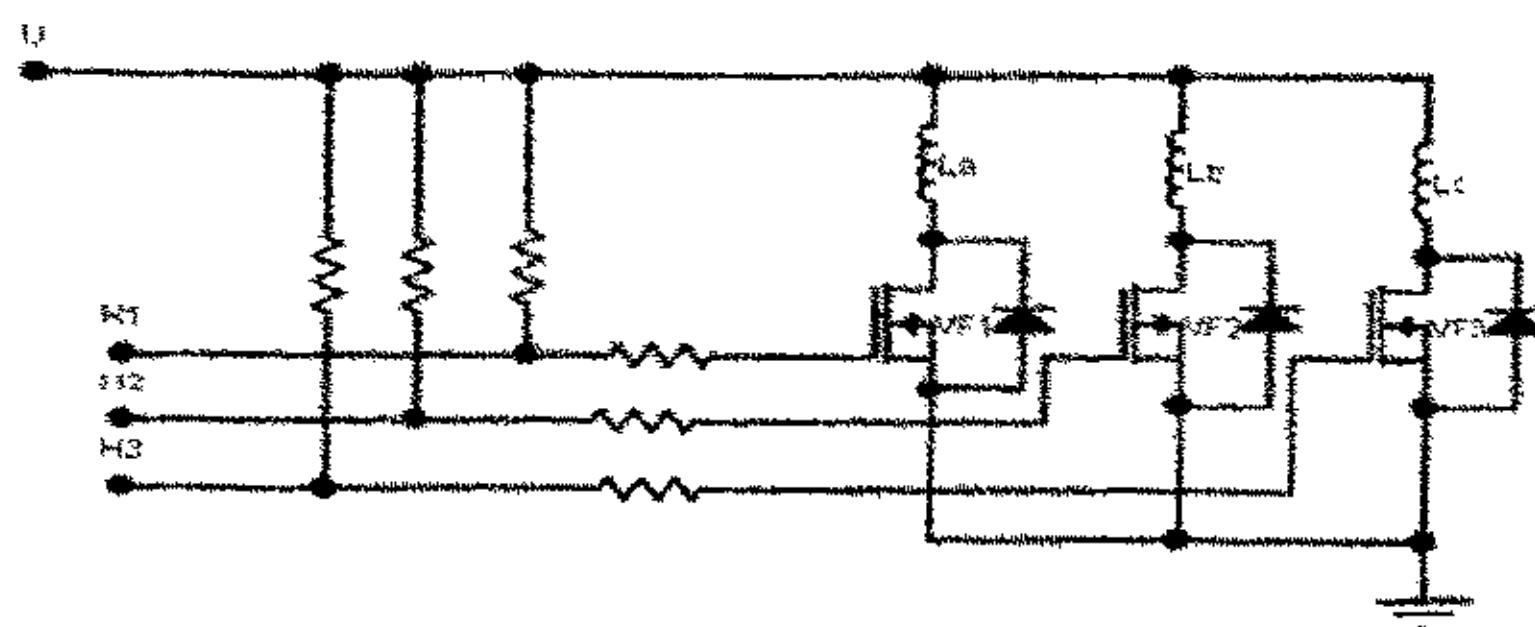


图 3-2 三相半控电路

会有明显的转矩波动, 而且三相半控驱动方式启动特性和低速运行的平稳性都有一定的限制。此外这种三相半控驱动只能使直流无刷伺服电机运行在单一象限, 控制灵活性受到了一定的限制。所以在要求比较高的场合, 一般不采用三相半控电路。

3.1.3.2 电容储能型(c-dump)驱动

电容储能型(c-dump)变换器^[3], 它以 $n+1$ 个功率开关实现了 n 相电机的四象限运行。由于使用了较少的功率开关且不存在桥臂直通的可能性, 它在可靠性和经济性上有相应的优势。

图 3-3 (a) 中 A、B、C 为电机相绕组, 它们分别串联一个主开关 T_a 、 T_b 、 T_c (下面统一用 T_s 表示), 并通过二极管与储能电容 C_0 连接; 斩波开关 T_r 和二极管 D_r 构成 buck 电路使 C_0 上的能量回馈到电源端。假设电机三相绕组对称, 相反电势为波顶宽 120° 电角度的梯形波, 三相主开关管轮流工作 120° , 采用脉宽调制 (PWM) 方式, 则开关过程主电路的等效模型可用图 3-3 (b) 表示。图中 k_c 为 T_s 开关状态变量, k_e 为电机运行状态变量, 反电势与储能电容用等效电压源表示。电动时, $k_e = +1$, 反电势极性如图所示, 每相主开关在相应相反电势为正梯形波波顶时开通。当 T_s 开通时, $k_c = 0$, 电源给相绕组提供能量, 相电流上升; 当 T_s 关断时, $k_c = 1$, 相绕组电流经与本相绕组连接的二极管续流, 给 C_0 充电。 C_0 在主开关关断时增加的能量应在 T_r 开通期间经 buck

电路回馈到电源端。 C_0 端电压可通过控制 T_r (采用两态调制方式) 使其维持在一定值 E 。

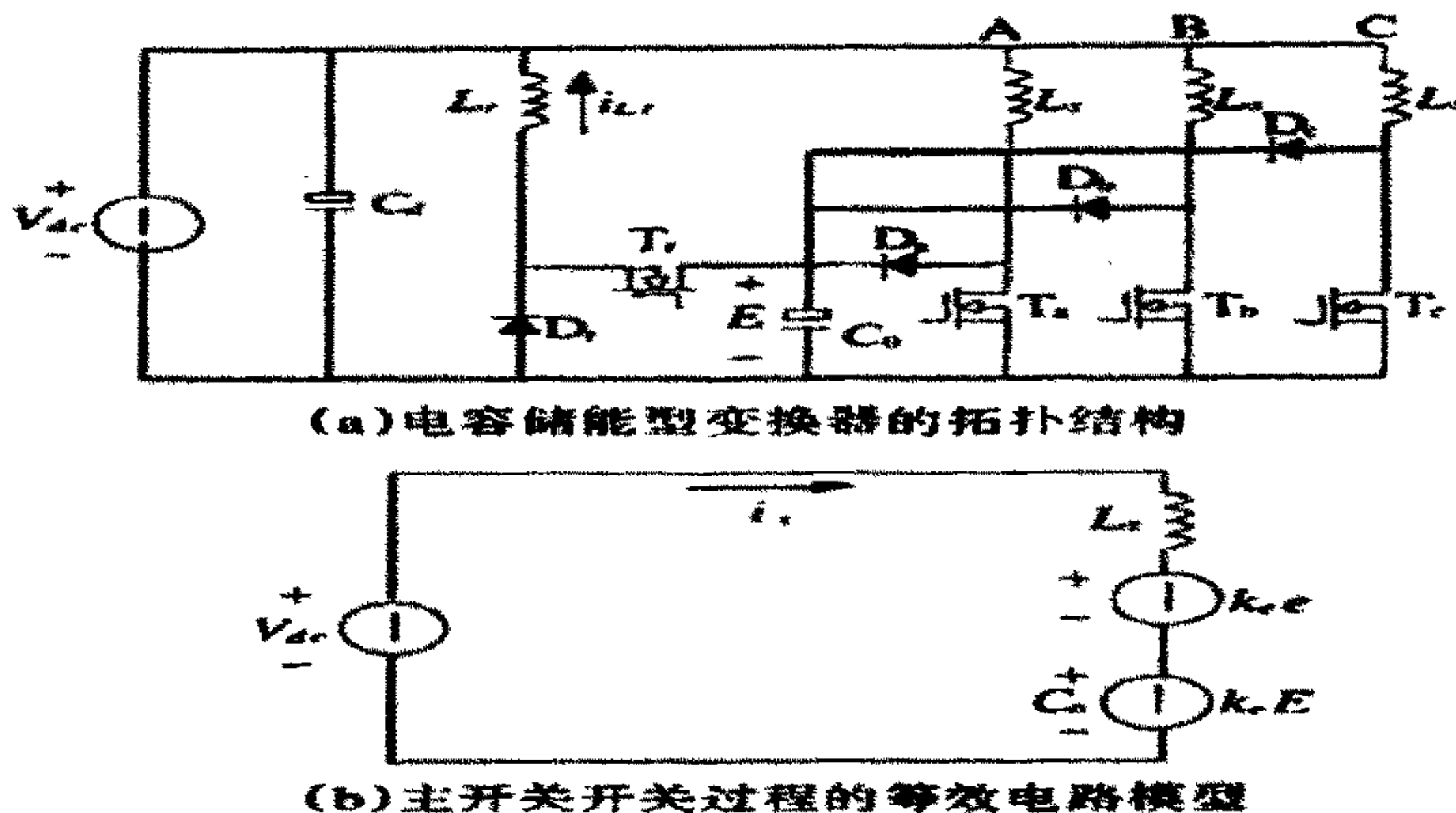


图 3-3 用于直流无刷伺服电机的电容储能型变换器拓扑结构

总的来说, 采用 C-dump 型的优点是能以最少的开关功率器件实现直流无刷伺服电机的四象限运行, 具有颇高的经济性, 但从上述分析来看, 其控制方式比较复杂, 特别在对参数 C_0 的选择方面, 因为储能电容 C_0 作为两级电路能量 (绕组磁能和制动回馈能量) 交换的中介元件, 变换器每相 PWM 工作 120° , 即一相导电方式。为确保一相导电期间, 另两相处于断电状态 (换相除外), 必须选择合适的储能电容 C_0 电压 E , 使之高于主开关端电压 U_T , 从而使关断相续流二极管在反电势负半周期间处于关断状态。若 C_0 电压选择未考虑互感因素, 可能使相绕组在非导通期间产生感应电流, 破坏 C-dump 变换器供电直流无刷伺服电机一相导电的工作条件, 影响系统运行性能。由此可见采用 C-dump 变换器驱动需要考虑多方面因素的, 虽然经济, 但比较复杂, 需有待进一步的研究和试验。

3.1.3.3 三相全控电路驱动

图 3-4 示出了一种三相全控电路, 在该电路中, 电机的三相绕组为 Y 联结。VF1、VF2、…、VF6 为六只 MOSFET 功率管, 起绕组的开关作用。VF1、VF3 和 VF5 为 P 沟道 MOSFET, 低电平时导通, VF6、VF4 和 VF6 为 N 构道 MOSFET, 高电平时导通。它们的通电方式又可分为两两导通方式和三三导通方式两种。

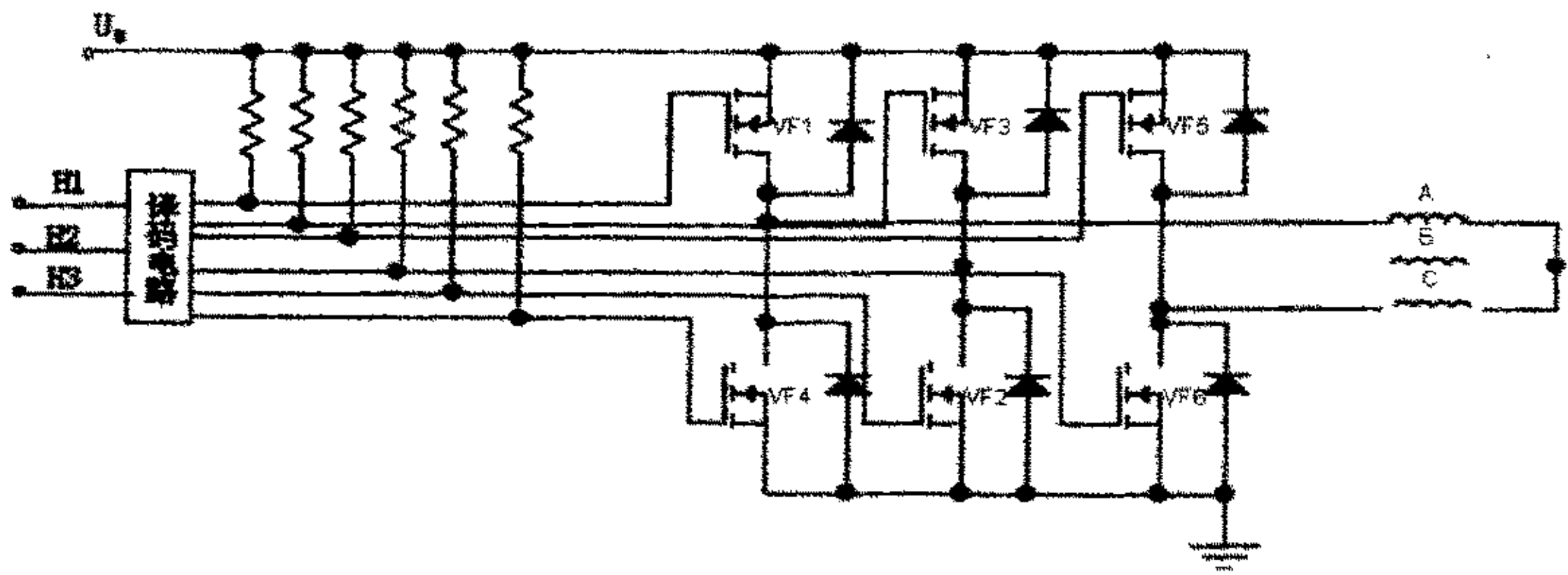


图 3-4 Y 联结绕组三相全控桥式电路

3.1.3.3.1 两两通电方式

所谓两两导通方式是指每一瞬间有两个功率管导通，每隔 $1/6$ 周期（60 电角度）换相一次，每次换相一个功率管，每一功率管导通 120 电角度。各功率管的导通顺序是 VF1VF2、VF2VF3、VF3VF4、VF4VF5、VF5VF6、VF6VF1、…。当功率管 VF1 和 VF2 导通时，电流从 VF1 管流入 A 相绕组，再从 C 相绕组流出，经 VF2 管回到电源。如果认定流入绕组的电流所产生的转矩为正，那么从绕组流出所产生的转矩则为负，它们合成的转矩如图 3-5a 所示，其大小为 $\sqrt{3} T_a$ ，方向在 T_a 和 $-T_c$ 的角平分线上。当电机转过 60 度后，由 VF2VF2 通电换成 VF2 VF3 通电。这时，电流从 VF3 流入 B 相绕组再从 C 相绕组流出，经 VF2 回到电源、此时合成的转矩如图 3-5b 所示，其大小同样为 $\sqrt{3} T_a$ 。但合成转矩 T_{bc} 的方向转过了 60 电角度。而后每次换相一个功率管，合成转矩矢量方向就随着转过 60 电角度，但大小始终保持 $\sqrt{3} T_a$ 不变。图 3-5c 示出了全部合成转矩的方向。

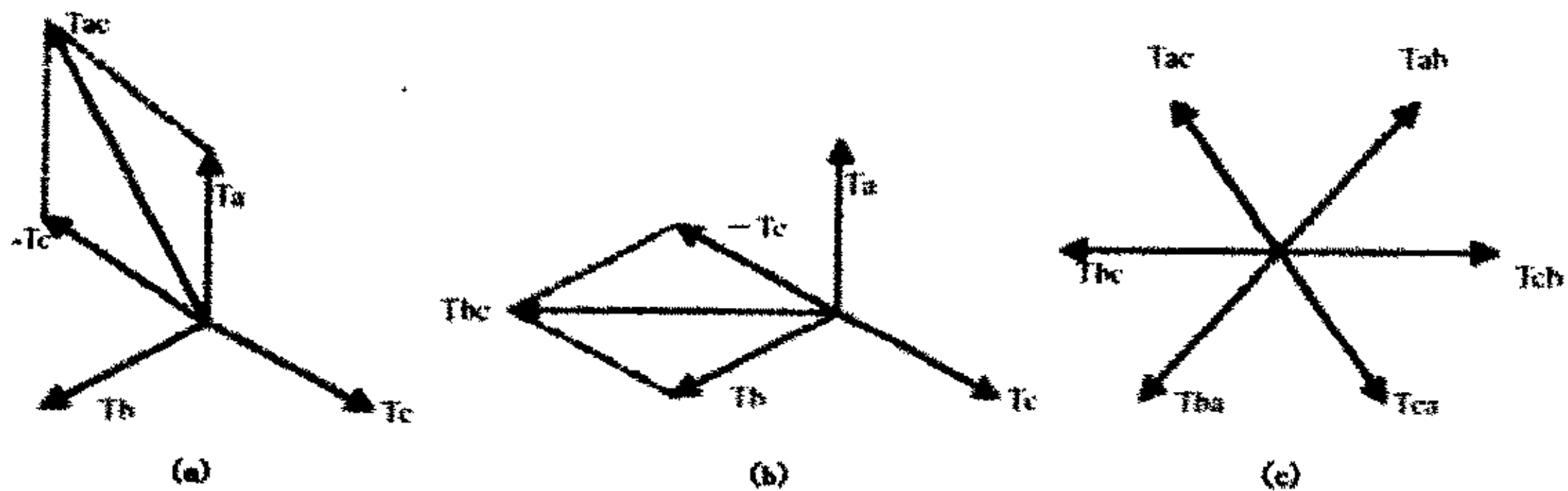


图 3-5 Y 联结绕组两两通电时的合成转矩矢量图

a)VF1、VF2 导通时合成转矩 b)VF3、VF4 导通时合成力矩 c)两两通电时合成转矩矢量图

所以, 同样一台直流无刷伺服电机, 每相绕组通过与三相半控电路同样的电流时, 采用三相 Y 联结全控电路, 在两两换相的情况下, 其合成转矩增加了 $\sqrt{3}$ 倍。每隔 60 电角度换向一次, 每个功率管通电 120 度, 每个绕组通电 240 度, 其中正向通电和反向通电各 120 度, 其输出转矩波形如图 3-6 所示。由图 3-6 可以看出, 三相全控时的转矩波动比三相半控时小得多, 仅从 $0.877T_m$ 到 T_m 。

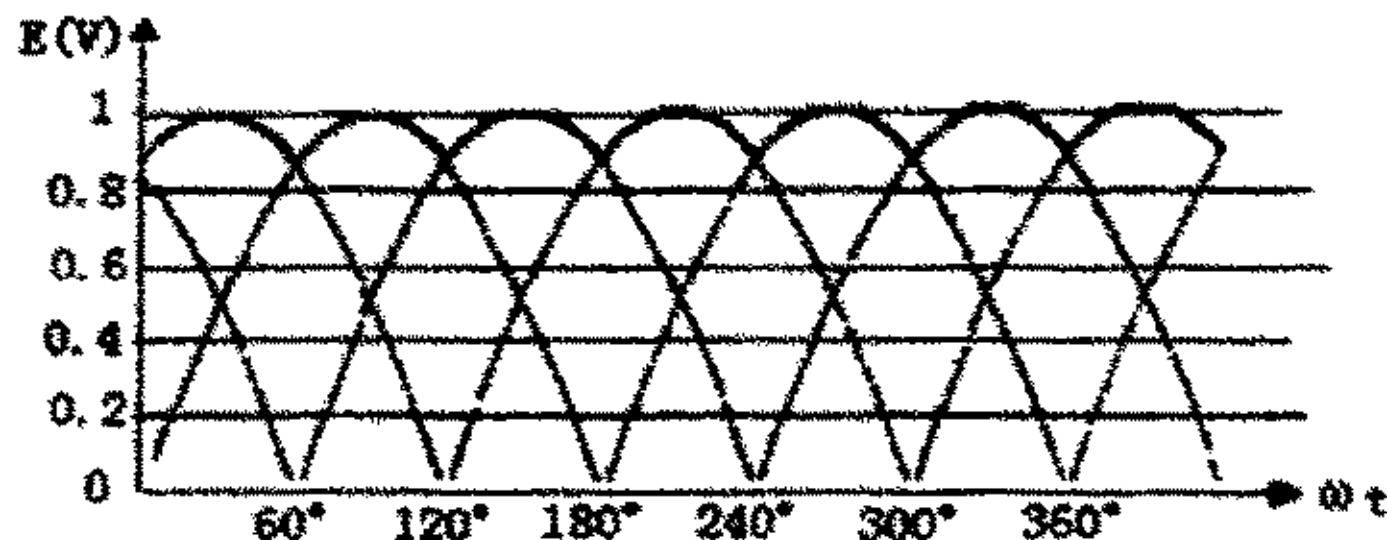


图 3-6 全控桥输出波形图

3.1.3.3.2 三三通电方式

所谓三三通电方式, 是指每一瞬间均有三只功率管同时通电, 每隔 60 度换相一次, 每个功率管通电 180 度。它们的通电次序是 VF1VF2VF3、VF2VF3VF4、VF3VF4VF5、VF4VF5VF6、VF5VF6VF1、VF6VF1VF2、VF1VF2VF3。当 VF6VF1VF2 导通时, 电流从 VF1 管流入 A 相绕组, 经 B 相和 C 相绕组(这时 B、C 两相绕组为并联)分别从 VF6 和 VF2 流出。这时流过 B 相和 C 相绕组的电流分别为流过 A 相绕组的一半, 其合成转矩如图 3-7 所示, 方向同 A 相, 而大小为 $1.5T_a$ 。经过 60 电角度后, 换相到 VF1VF2VF3 通电, 即先关断 VF6 而后导通 VF3(注意, 一定要先关 VF6 而后通 VF3, 否则就会出现 VF5 和 VF3 同时通电, 则电源被 VF3VF6 短路, 这是绝对不允许的)。这时电流分别从 VF1 和 VF3 流入, 经 A 相和 B 相绕组(相当 A 相和 B 相并联)再流入 C 相绕组, 经 VF2 流出, 合成转矩如图 3-7b 所示。其方向与 C 相向, 转过了 60 度, 大小仍然是 $1.5T_a$ 。再经过 60 电角度后, 换相到 VF1VF2VF3 通电, 而后依次类推。它们的合成转矩矢量图如图 3-7c 所示。原理图如图 3-14 所示。

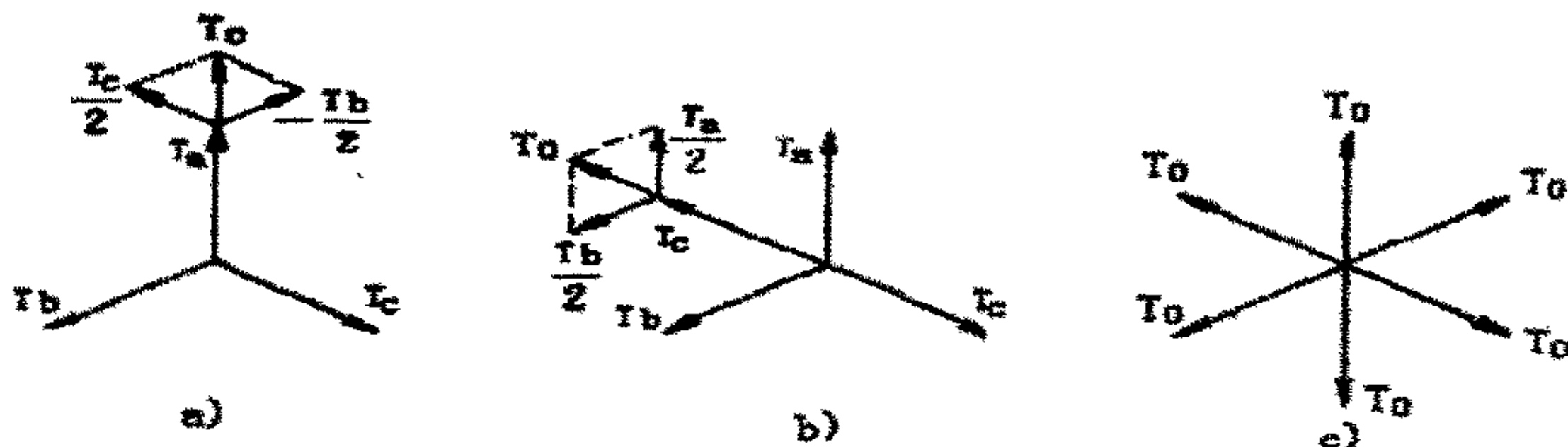
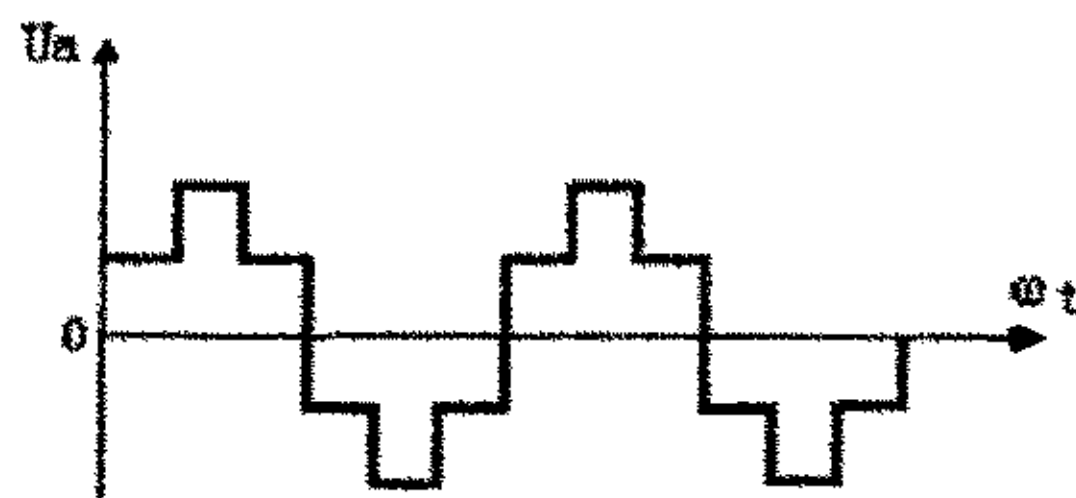


图 3-7 三三通电时的合成转矩矢量图 a) VF6VF1VF2 导通时的合成转矩 (b) VF1VF2VF3 导通的合成转矩 c) 三三通电时的合成转矩

从某一相上看、它们的电压波形如图 3-8 所示。



从以上分析可以看出三相全桥逆变驱动虽然增加了开关管的数量,但增大了转矩输出且转矩波动明显小于三相半桥驱动,而且启动特性和低速平稳性都较好。而且三相全桥驱动方式还能实现电机的四象限驱动,又便于模块化的设计,特别适合于 DSP 的全数字化控制,可见三相全桥驱动相对于三相半桥和 C-dump 驱动方式来说,是比较理想的驱动方案。

图 3-8 Y 联结三三相通电方式一相电压波形

3.2 位置传感器的选择

位置传感器在直流无刷伺服电机中起着测定转子磁极位置的作用,为逻辑开关电路提供正确的换相信息,即将转子磁钢磁极的位置信号转换成电信号,然后去控制定子绕组换相。位置传感器种类较多,且各具特点。日前在直流无刷伺服电机中常用的位置传感器有下叙几种形式。

3.2.1 电磁式与光电式位置传感器

电磁式位置传感器是利用电磁效应来实现其位置测量作用的,有开口变压器、铁磁谐振电路、接近开关等多种类型。其主要原理是通过电机内部扇形与各相绕组的电磁感应来确定电机运行的当前位置。

电磁式位置传感器虽然具有输出信号大、工作可靠、寿命长、使用环境要求不高、适应性强、结构简单和紧凑等优点,但这种传感器信噪比较低,体积较大。同时其输出波形为交流,一般需经整流、滤波后方可应用,控制起来也相当复杂。

光电式位置传感器是利用光电效应制成的,主要由跟随电动机转子一起旋转的遮光板和固定不动的光源及光电管等部件组成。

光电式位置传感器虽然性能较稳定,但存在输出信号信噪比较大、光源灯泡寿命短、使用环境要求较高等缺陷。

3.2.2 磁敏式位置传感器

磁敏式位置传感路是指它的某些电参数按一定规律随周期磁场变化的半导体敏感

元件。其基本原理为霍尔元件的磁阻效应。目前,常见的磁敏传感器有霍尔元件或霍尔集成电路、磁敏电阻器及磁敏二极管等多种。其中以霍尔元件最为常见,霍尔元件是通过磁感应强度效果产生微弱的电压信号通过信号调理和放大转换成 DSP 可以接受的采样信号。直流无刷伺服电机转子能在不同位置由永磁转子和定子的电枢反应生成的合成磁场所感应出不同磁感应强度,通过霍尔元件将信号传递给 DSP。其工作原理图如图 3-9 所示,图中示出了直流无刷伺服电机在不同转子位置时霍尔元件的工作情况。

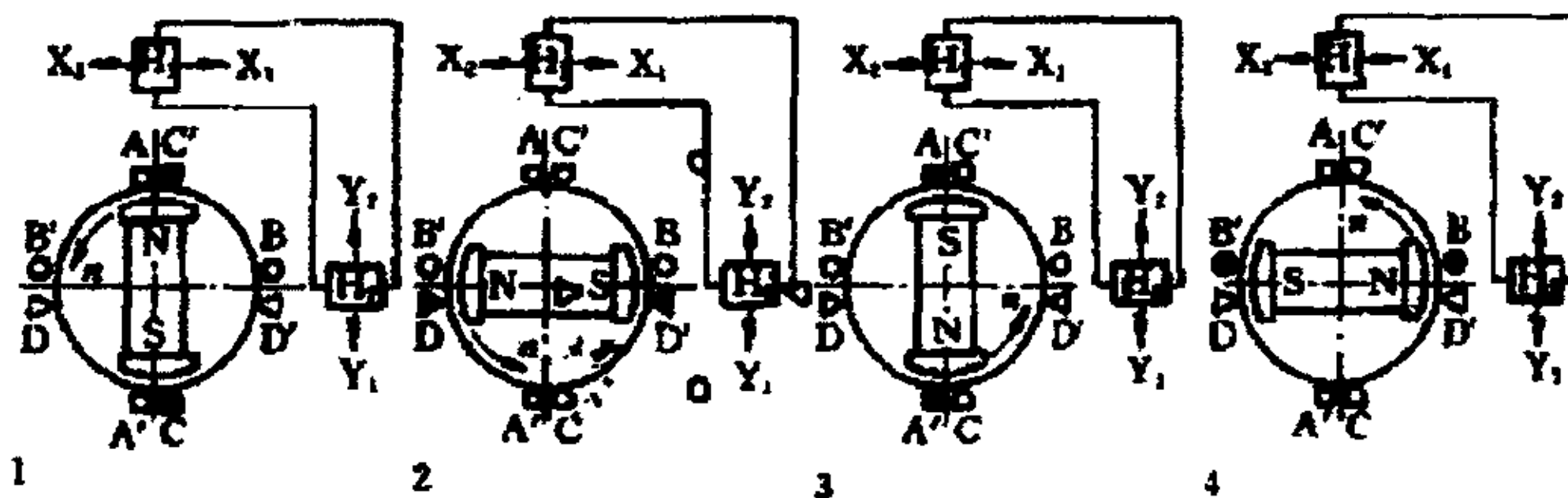


图 3-9 霍尔直流无刷伺服电机的原理图^[4]

在众多位置传感器中,霍尔元件从性能上讲,其效果是最好的^[4],虽然霍尔元件价格比较高,但其体积小,而且大多都集成在被控电机的内部,具有很好的输入输出特性。

此外还有一种无位置传感器的位置检测方法,主要是通过测量其他绕组的反电动势,根据反电动势与转子位置的一定规律,来判断转子的当前位置,从而控制电子开关换向。当然,这是一种最为经济的方案,有一定的实用价值,是一个比较好的位置检测的发展方向。

根据以上各方案比较,鉴于被控电机本身提供霍尔位置传感器,所以本系统采用霍尔位置传感作为位置检测基准,这种检测方式不但对位置检测提供了很好的保障,而且位置信号和 DSP 之间都提供了非常方便的接口,从而达到了对电子开关高可靠性的控制。

3.3 速度检测方案的比较选择

转速控制是实现高性能伺服驱动的关键环节,由于直流无刷伺服电机作为一种典型的机电一体化装置,其结构上的特殊性及应用场合的特殊性,其转速信号的获取及处理,有别于传统的 AC 和 DC。为此,如何得以获取其精确的转速信号以实现对其转速的高精度控制;同时满足直流无刷伺服电机伺服驱动系统日益数字化发展的需要,也就成了直流无刷伺服电机应用的一个突出问题之一。

速度测量的最简单的方法是在转子同轴上安装一各个直流的测速发电机,由于直流

测速发电机能将速度信号很好的转换成电压信号,所以控制起来比较方便。但是直流测速发电机体积比较庞大,只有在大容量场合下使用,而在配备有位置传感器的直流无刷伺服电机的控制场合下,根本没有必要。

当采用光电编码盘或霍尔元件作为位置传感器时,可以通过对位置脉冲信号的处理计算得出当前直流无刷伺服电机的转速,同时还能得出其转向。比较常用的脉冲测速发有 M 法和 T 法及 M/T 法^[1],M 法也就是我们通常所说的测频法,其基本原理是在特定时间内测量编码脉冲数,比较适用于电机转速较高的速度测量;而 T 法是我们通常所说测周法,其基本原理是在指定的一个速度脉冲周期里测量和基准时间的脉冲数,比较适用于电机转速较低的速度测量。这两种速度测量方案的速度测量范围都不是很大,当超出其测量范围是将会产生比较大的误差。结合两者测量方法,采用 M/T 法,M/T 法综合了 M 法和 T 法的优点,其基本原理,如图 3-10 所示,是在稍大于规定时间 T_c 的某一时间 T_d 内,分别对测速脉冲个数 m_1 (即图中 PLG)和时钟脉冲个数 m_2 进行计数,从而可求出电动机转速,即: $n=60m_1f_0/pm_2$ (式中: f_0 —时钟脉冲的频率; p —电动机旋转一周测速脉冲发生器发出的脉冲个数)。

结合上述分析,对于直流无刷伺服电机伺服驱动系统,其转子位置传感器输出脉冲经过适当整形处理之后可以作为 M/T 法的测速脉冲(低速场合可以先经过倍频处理以提高其测速精度)。因此,采用 M/T 法实现直流无刷伺服电机数字式测速是可行的。

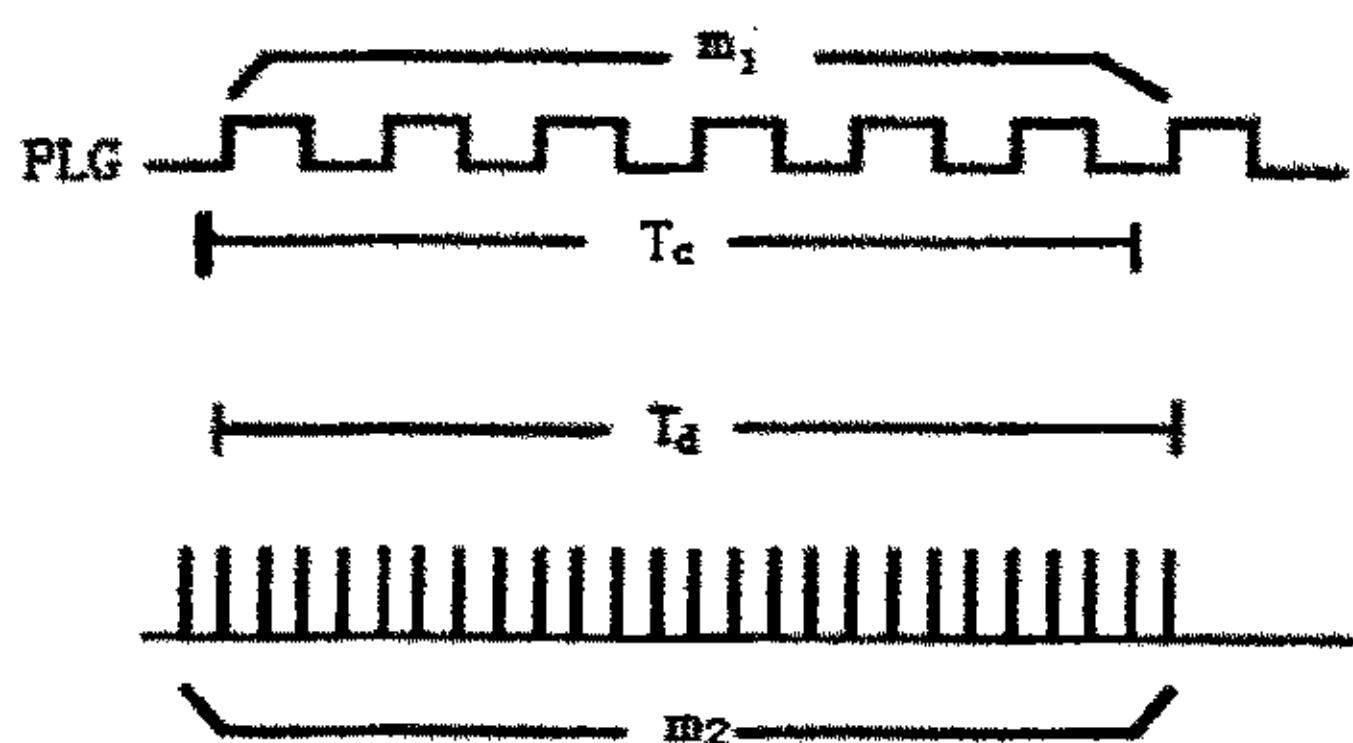


图 3-10 M/T 法测速原示意图

当采用反电势法的无位置传感器的控制时,可对检测到的反电动势进行数字处理,运用特殊算法如模糊算法,卡尔曼递推等,前向预测估计出当前速度。

在本系统中,由于在位置检测中结合使用了霍尔传感和光电正交编码器为速度检测提供了很好的检测方案,所以在速度检测中同样采用这种方式,再结合 M/T 的测速软件设计,使得本系统的速度测量能达到一个比较高的精度。

3.4 电流采样方式比较与选择

对于全桥 PWM 逆变器来说,直流无刷伺服电机的输出转矩与相电流成正比,因此

需要以相电流作为反馈来构成转矩闭环。获得相电流信号最有效的方法便是采用 3 个电流传感器直接测量逆变器的三相输出。对于三相星型连接的电机来说,电流采样通常的方法是在电机三相绕组的任意两相上安置两个霍尔元件常用磁平衡式霍尔元件 LEM 模块,将所得的小电流信号经精密放大处理来检测相电流信号,过载及短路保护功能则通过直流侧的电流传感器实现。这一方案可以提供关于电动机相电流的全部信息,所构成的转矩闭环控制系统对由于电流换相引起的转矩波动具有较强的抑制作用,适用于对电动机输出转矩平稳性要求较高的场合,但系统的成本较高。

两相电流的测量也可以用过采样电阻取值,再经信号隔离放大获取电流信号。这种采用方法的最大优点是成本较低,但存在功耗较大的缺点。此外由于采用大功率电阻,所以电阻的体积较大,系统显得比较笨重,再加上电阻阻值的非线性性,使得测量的精度也难以得到保证。

文献网^[7]提出了一种单电流传感器实现相电流检测的方法,其主要思想是通过 PWM 控制信号与逆变器所提供的电流通路之间的相互关系,在 PWM 有效的期间来采样电流值,由于采用时分控制的电流采样方案,可以实现单相电流传感器获取三相电流信号。这种方案简单经济,具有颇高的性价比。

考虑但电阻采样的功耗与精度问题,在结合单相电流检测的方法,采用单相霍尔电流检测(原理将 4.3 节中讨论)在在精度控制上和在成本考虑方面是一个比较好的折中。由于 DSP 在信号处理方面具有很大的优势,通过一定的算法,可以将一相电流检测信号转换成三相电流信号,这样通过软件来实现单相向三相的转换;同时霍尔元件的电流采样在精度上能达到一个比较高的标准,本身提供了强弱电的隔离,通过简单的信号放大,信号能直接被 DSP 采样获取。

3.5 控制算法的比较论证

由于本系统是以 DSP 为控制核心来实现直流无刷伺服电机的各种控制,且 DSP 具有很强大的数据运算处理能力,所以好的控制算法可以充分发挥出 DSP 的控制优势,达到最佳控制的目的。

现在比较流行的是非线性控制,但由于非线性控制系统本身比较复杂,而且缺少比较可靠的理论基础,或者说对于可能出现现象根本无法解释,还可能出现振荡,甚至混沌现象,即非稳定的极限环,又非无限制的发散。总而言之,非线性系统在本系统难以实现。

模糊控制也是现在被广泛使用的控制方法之一，它由神经网络与模糊逻辑结合而成，直接控制输入输出关系。然而，模糊控制在构建神经网络时，需要获取大量的信息，这就不得不进行大量的数据采集与参数估计。系统在运行过程中还必须反复更新自身的网络结构，不断学习，以适应新的环境。此外模糊神经网络控制一般很难达到最优控制，且其最终学习的结果有时不太可靠，所以，此控制方案在本系统控制中还有待于研究。

最经典的控制算法莫过于 PID 算法，特别是现在被广泛使用的数字 PID 算法^[13]，再配合 DSP 的高速运算能力，使其在各种算法中颇具有一定的优势。众所周知 PID 算法由三部分组成，即比例环节、积分环节和微分环节组成。在要求不高的情况下，有时只需比例和微分环节即可达到一般控制的目的。数字 PID 算法是将原来经典的模拟 PID 信号数字化，离散化，一般采用增量式数字 PID 算法，其优势在于便于软件实现，控制效率较高。PID 比起现在比较流行的模糊神经网络的控制，在控制效果上来讲，PID 控制丝毫不亚于神经网络控制，有时它的调节能力还强于神经网络控制。由于 PID 控制模型非常好建立，其控制原理在经典的自动控制原理中都能得以成熟的验证，操作简单，只需对比例、积分和微分三个参数设置即可完成一般的控制目的。当然，为了达到比较高的控制精度要求，克服一些经典 PID 算法的缺点，如积分饱和等，本系统将采用改进型的数字 PID 算法，这种改进性的数字 PID 算法能有效的克服如积分饱和等经典 PID 算法的固有缺点，而且控制起来也非常方便，其结果无论在原理上和是在软件仿真中都得以很好的验证。其基本构成原理框图如 3-11 所示。该方法的具体实现将在下面几章具体阐述。

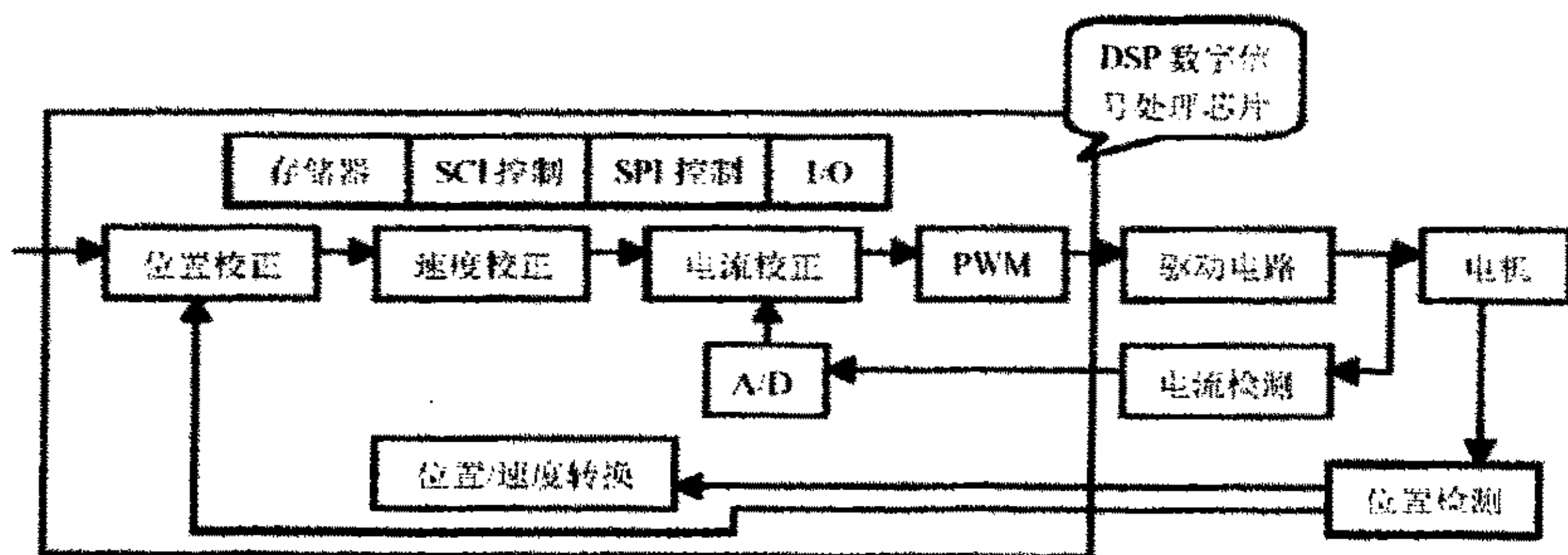


图 3-11 采用 PID 控制的结构框图

第四章 控制系统的总体设计

本系统的控制电路主要由功率驱动、三相逆变和逻辑控制电路组成，当然还包括 DSP 接口电路，以及必要的保护电路。其控制系统硬件的构成，如图 4-1 所示。本系统采用 PWM 方式实现对直流无刷伺服电机的控制。其基本原理是交流输入经过整流、稳压并为逆变电路提供直流电源。驱动部分采用富士公司的智能驱动模块 (IPM)，控制由位置、转速、电流三闭环组成，从而有效地提高了直流无刷伺服电机的控制精度。位置给定由给定直流无刷伺服电机自带的霍尔元件给出，并由 LF2407 的 CAP 端进行捕捉定位。转速及其方向给定由光电编码器形成正交编码信号，由 LF2407 的 QEP 端进行速度与转向的鉴定。根据给定的速度信号，DSP 产生一定的 PWM 波。通过调整 PWM 波宽度控制功率管的开关时间，实现对直流无刷伺服电机的控制。电流给定由单电流霍尔传感器在 LF2407 的控制下传达三相电流信号并由 LF2407 自带的 A/D 进行采用，由 LF2407 根据电流采样值来调整 PWM 波的占空比，从而改变相电流幅值。人机界面接口采用键盘液晶显示，并提供上位机 (SCI 方式) 与现场总线 (CAN 总线) 的接口。在保护方面一旦产生故障，通过故障保护电路，封锁 PWM 的输出直至故障取消。

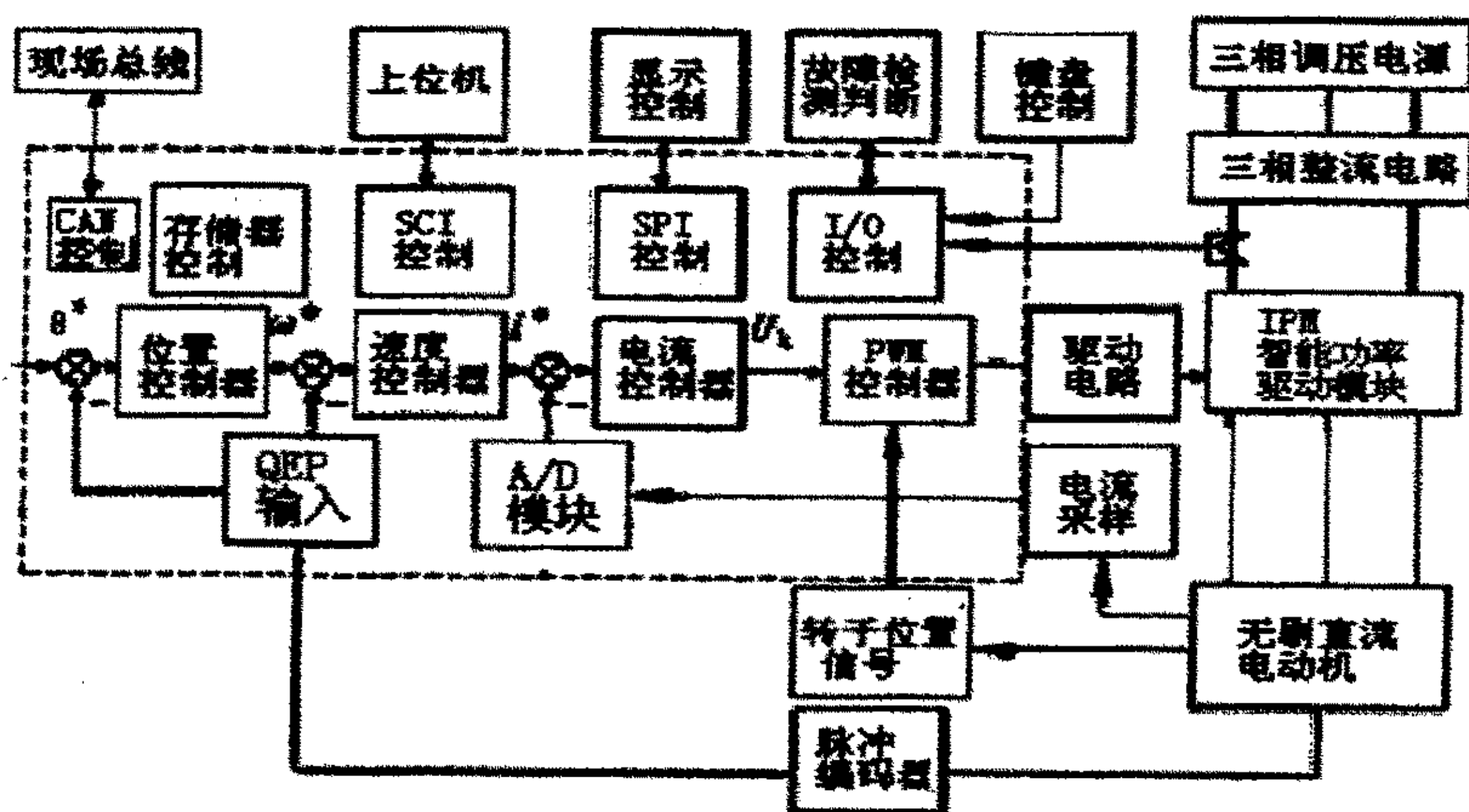


图4-1 系统结构框图

4.1 电源（功率输入整流回路设计）

电源提供主要有两种，一个是控制回路的电源，另一个是驱动部分的电源。前者主要是DSP的供电电源+3.3V，主要用TI公司的电源管理芯片TPS767D301来解决5V输入3.3V输出的电源转换。TPS767D301可以提供两路输出，一路为固定的3.3V输出，主要是供给LF2407；另一路是可调输出，由两电阻通过一定比值来决定输出电压的大小，可作某些器件的基准电源。还有运算放大器的供电电压+12V与-12V，是由220V变压器的抽头引出，经全桥整流滤波由三端线性稳压器件7812和7912提供。驱动部分主要采用常用的交-直-交变换供电，交直变换采用不可控三相全桥整流，简单经济。直交变换部分采用富士公司生产的智能功率模块（IPM）。系统运行方式选用两相导通三相六拍工作方式，即每次只能两个管导通，每个IGBT导通角为120电角度。采用这种运行方式可以最好地利用准方波气隙磁场的平顶部分，使电机出力最大，平稳性最好。

4.2 功率变换器与驱动电路

功率驱动部分主要采用IPM智能功率模块，它不仅把功率开关器件和驱动电路集成在一起，而且还内藏过电压、过电流和过热等故障检测电路，并可将检测信号送CPU。即使发生负载事故或使用不当，也可保证IPM自身不受损坏。目前的IPM一般采用IGBT作为功率开关器件，并内藏电流传感器及驱动电路的集成结构。

IPM驱动板上，采用软启动以减少对主电路的冲击。使用不可控整流，330 μ F \times 6电容滤波，RDC电路吸收电网电压的冲击。为了安全，选择IPM。它自带保护信号输出，可与过压过流信号经或门后输出至DSP保护中断，其电源由8路独立供给，分别给IPM和控制电路供电。图4-2为IPM驱动板的系统方框图。

其中，驱动部分的作用是进行阻抗匹配和整形；死区电压产生部分为PWM的波形产生5 μ s的死区，以防止上下桥臂的开关管同时导通损坏开关器件。当然LF2407也提供带死区输出的PWM

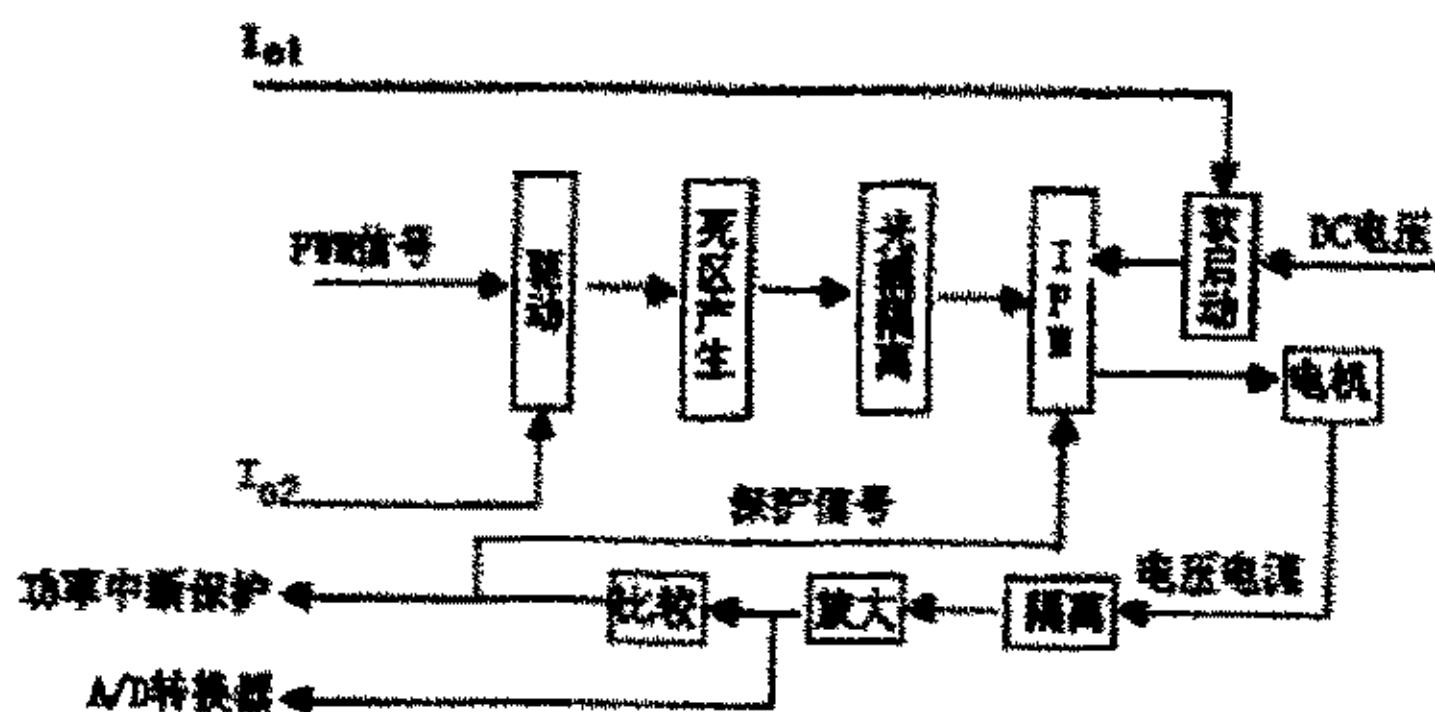


图 4-2 IPM 驱动板的系统方框图

控制，这样更有效地防止了电源短路事故的发生；光耦隔离用来将强电部分和弱电部

分隔离；直流电机的电压和电流返回至控制回路，供 DSP 监测；一旦有异常发生，DSP 将通过 I_{o2} 将死区电压产生电路封锁，以隔离 SVPWM 信号，同时由特殊中断使 PWM 端处于高阻状态，达到双保险。

4.3 相电流检测与调整

直流无刷伺服电机的电枢电流有重要意义，由于其正比于轴上转矩，往往作为电机转矩的代表，对指示、控制有重要作用。电枢电流流过绕组与电路器件会发热，因此它又是电机与装置安全运行的重要参数。

对于三相星型连接的电机来说，正如方案论证中所讲的，电流采样通常的方法是在电机三相绕组的任意两相上安置两个霍尔元件，来检测相电流信号。为了达到性价比高的相电流检测，本系统采用一种更为简单有效的单电流传感器检测相电流的方法^[6]。

本系统采用两相导通三相六拍运行方式，同时设计 PWM 调制方式采用半桥型 PWM 调制（即采用上桥调制下桥恒通的调制方式），一方面可以降低逆变器的开关损耗，另一方面是为本系统采用的单电流传感器检测方案而设计的。假设 AB 两相导通，如图 4-3 (a)、(b) 中所示虚线分别对应 PWM 调制信号有效期间和无效期间各自的电流通路。

位于在桥式整流电路的低电压端与地之间的采样电阻用于检测主回路上的电流。采样电阻上的电压经过 DSP 上的 ADC 单元，变为数字的电流信号。为了实现 20kHz 的电流环，每 $50\mu s$ 电流必须采样一次。在新 PWM 波产生之初，载入电流检测值，与给定的参考电流值一起来控制 PWM 波宽度，从而产生新的 PWM 波。同时，当电流检测值超过所允许的最大值即主电路过电流的时候，发出中断信号，产生中断，执行相应的中断处理程序就可以启动过流保护程序，封锁所有驱动信号的输出，直至故障解除。

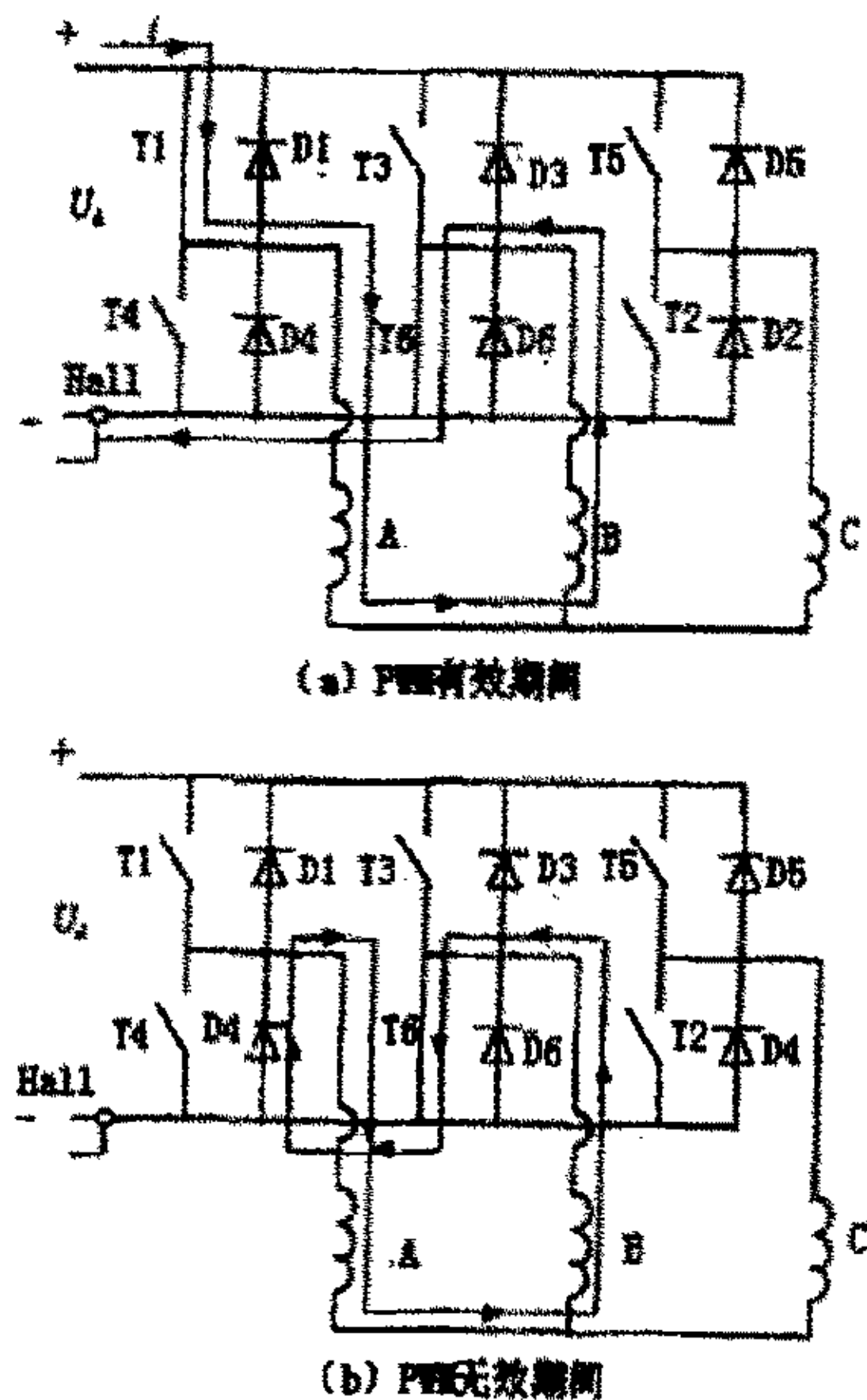


图 4-3 PWM 半桥调制逆变器电流回路示意图

从图中可以看出,若每次电流信号采样均在 PWM 有效期间,则就可有效地检测到电机的相电流。为此,我们做了一些特殊的设计^[10],如图 4-4 所示。上半部为 DSP 通用定时器 T1 的波形, T1 设定为连续加减计数方式,每三次又下溢中断为一个电流环周期,启动 ADC 采样一次电流信号。并根据不同的导通状态,对 DSP 的 ACTR 写入相应的控制字,以保证 T1 产生下溢中断时对应 IGBT 上的 PWM 信号为低电平,即 PWM 有效,这样每次进行电流信号采样的时刻均在 PWM 有效期间,从而实现了单电流传感器相电流检测。

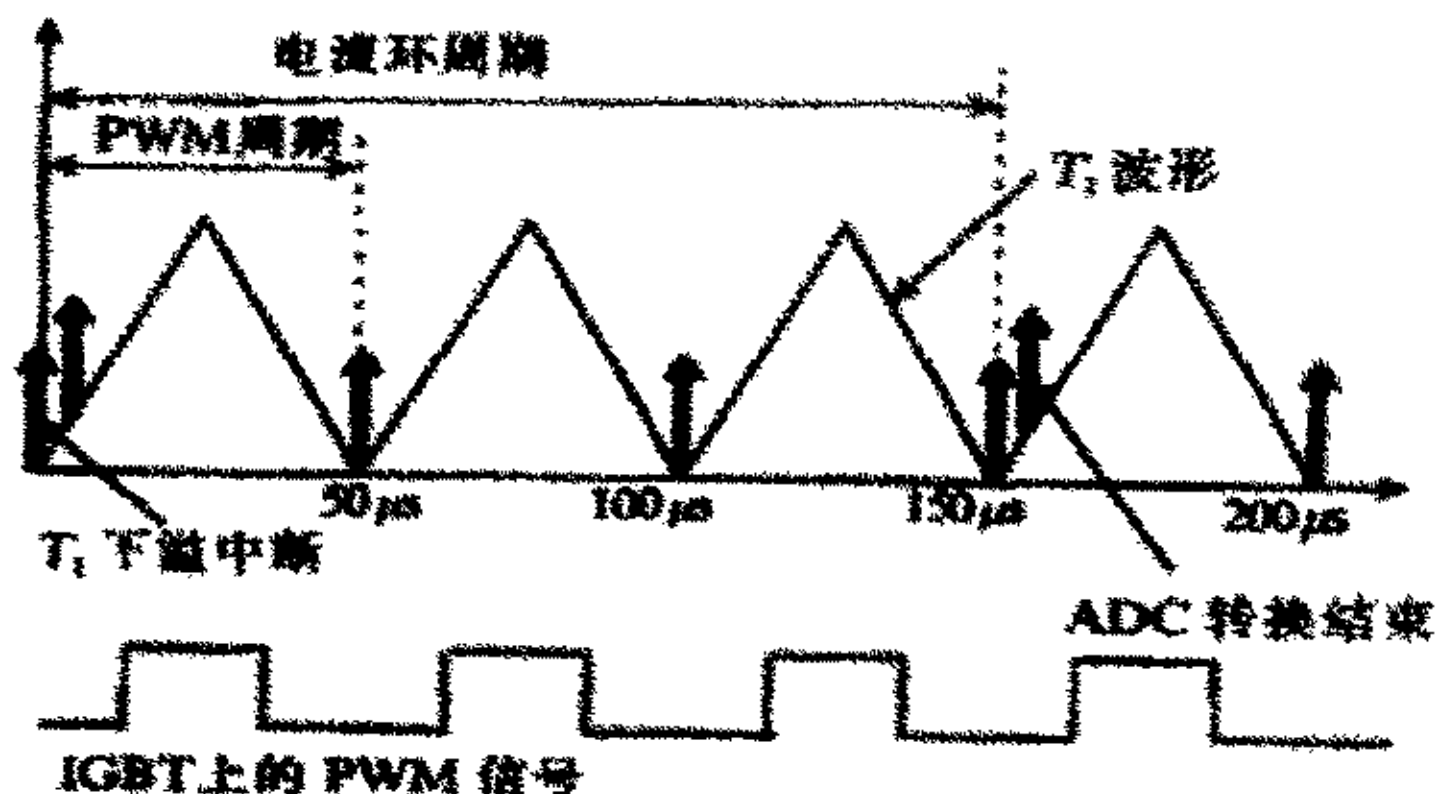


图 4-4 电流信号采样及 PWM 信号的时序

4.4 位置检测

有本系统的被控直流无刷伺服电机是内带霍尔效应传感器的,所以位置检测就显得非常容易。该传感器由功放电路板供电,传感器的输出端经差动放大隔离后直接连到 TMS320LF2407 的捕获单元 (CAP/QEP)。电机在正常运转的时候,通过霍尔传感器可得到 3 个脉宽为 180 电角度的互相重叠的信号,这样就得到了 6 个强制换相点, DSP 芯片通过检测到传感器输出信号的上升沿和下降沿后,即可得到转子位置。在中断处理程序中改变 ACTR 的值,产生对应的 PWM 输出信号,给对应的绕组通电,起到电子换相器的作用。

本系统所采用的被控直流无刷伺服电机自带的三个霍尔元件在空间上互成 120 度角,在电机旋转时可以得到三路信号,再经外部信号调理电路直接输入到 LF2407 的 CAP 进行捕捉,其硬件结构原理图如图 4-5 所示。

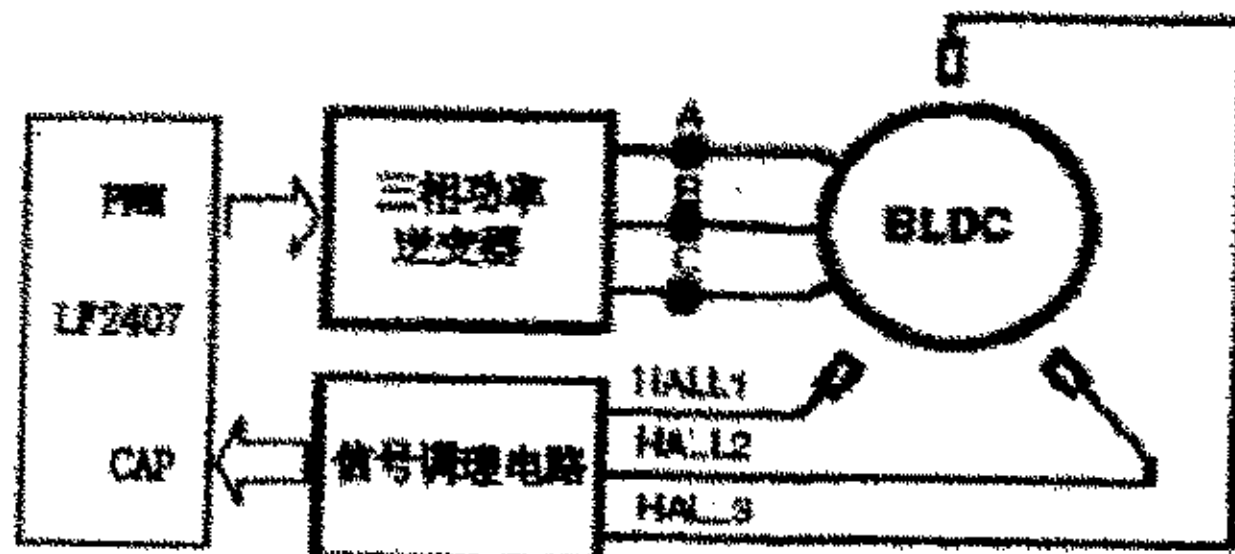


图 4-5 位置传感器接口结构原理图

4.5 转速检测

本系统的转速检测采用正交编码的增量式光电编码盘。由于 LF2407 的捕获单元 CAP1 和 CAP2 可配置为正交编码脉冲(QEP)模式,在这种模式下,两个 16 位通用定时器 T2、T3 构成一个 32 位的双向计数器。QEP 电路直接处理光电编码器输出的两路相位差 90° 的脉冲,只要将这两路脉冲接到 LF2407 的 CAP1/QEP1 和 CAP2/QEP2 上就可, QEP 模式对两路脉冲前后沿进行计数,无需外部倍频电路。LF2407 可根据脉冲的先后次序判别电机的方向,大大简化了系统的硬件,增加了可靠性。

增量式编码盘实际上是一个光电脉冲发生器和一个可逆计数器。在光电脉冲发生器圆盘上刻有节距相等的窄缝,另外还有 a、b 两组检测窄缝群(图 4—6a),两组检测窄



缝与圆盘上窄缝的对应位置错开 $1/4$ 节距,其目的是使 a、b 两个光电变换器的输出信号在相位上相差 90° 。圆盘与校验轴相连接,而两组检测窄缝是静止不动的。当被测轴转动时,两个光电变换器就输出相位相差 90° 的两个近似正弦波,如图 4—6b 所示。再经过简单的电路处理,可得到相应的脉冲信号。当圆盘正转(如图中箭头所示)时,信号 b 超前信号 a 90° 度,逻辑电路 f 端输出脉冲信号;圆盘反转时,信号 a 超前信号 b 90° 度,逻辑电路 g 端输出反转脉冲信号。若将这些脉冲信号送给可逆计数器进行累计,就可测出轴的旋转角度。

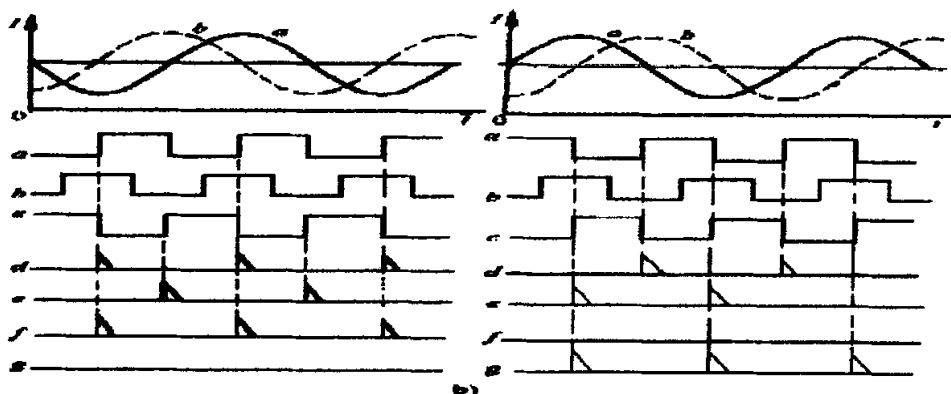


图 4—6 增量式光电编码盘的工作原理

a) 结构原理 b) 输出波形

此外,由于电机自带霍尔位置传感器,所以可以通过位置传感器的输出作为速度调

节器的给定。霍尔位置传感器使直流无刷伺服电机每个旋转周期内会产生六个交变信号。由于速度信号可以写成 $\Delta \theta / \Delta T$ ，又由于霍尔传感器相对于电机位置是固定的，也就是两交变信号之间的相位差是不变的，即 $\Delta \theta$ 不变(每两个信号交变的相位差为 60°)，因此速度计算可简化成简单的除法。只要测出两交变信号间的时间间隔就可以得到速度信号。虽然做除法的时候会产生浮点问题，但也可以用扩展的数学算法克服 LF2407 DSP 控制器除法的浮点问题，减小误差。

4.6 控制策略

正如前面的总的结构框图所示出的那样，本系统主要有三闭环构成^[12]，即位置闭环（外环）、速度闭环（中环）、电流闭环（内环）。位置反馈信号来自于电机自带的霍尔位置传感器，速度反馈信号来自于光电编码，电流反馈信号来自于单相霍尔电流检测。位置信号的输出作为速度信号的给定，速度信号的输出作为电流信号的给定。形成及联式的可靠高效的直流无刷伺服电机的控制方式。各控制环的结构图如图 4-7 所示。

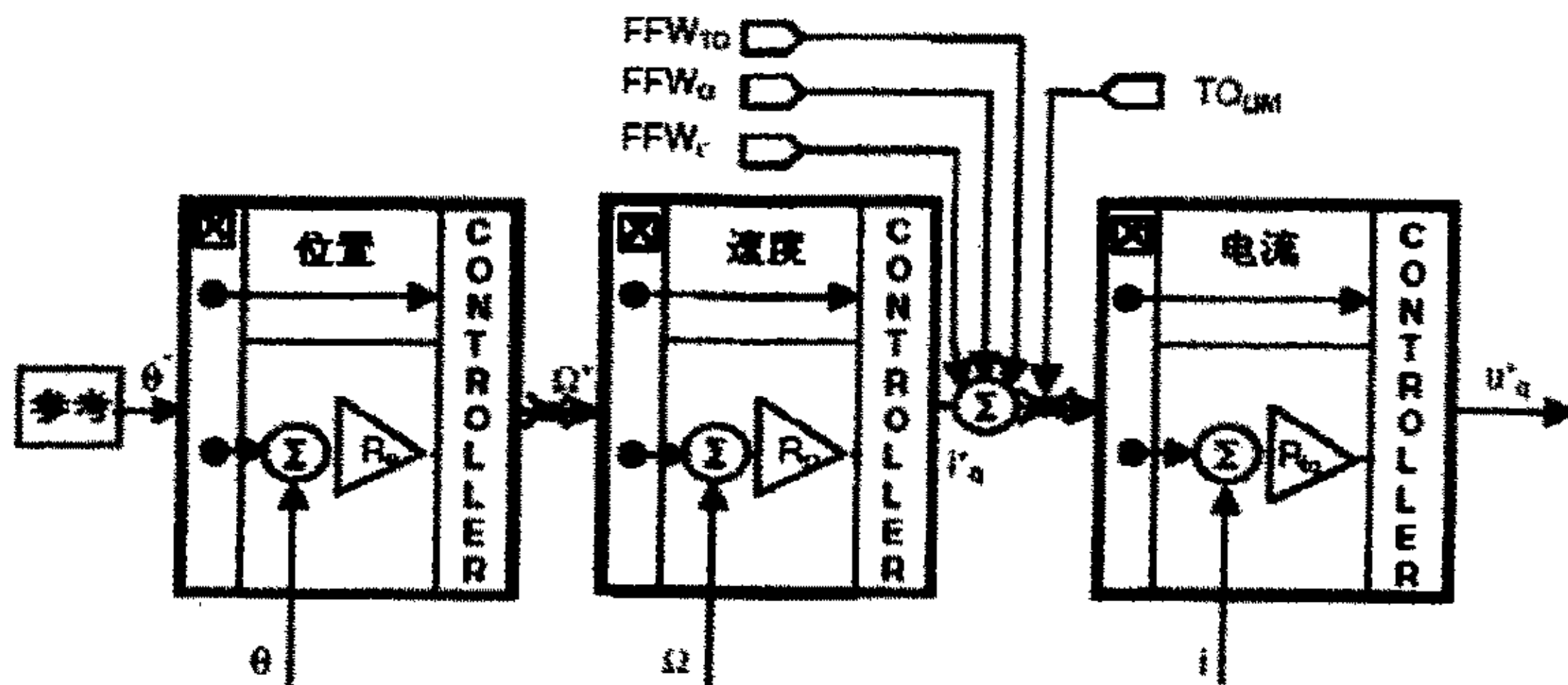


图 4-7 直流无刷伺服电机控制环结构图

4.6.1 电流闭环

电流环的调节采用传统工程设计方法设计成典型 PI 控制模型，结合 LF2407 高速数字运算性能，整合成高速有效可靠的数字 PI 控制。

由于电枢电流直接影响到直流无刷伺服电机的输出转矩，对控制性能有着直接的影响，所以电流内环采用高速控制，即 LF2407 以全速度采样电流值经数字 PID 算法调节，直接输出给三相逆变装置，这样可以较大的提高电机运行的稳定性和可靠性。

$I_{\text{error}} = I_{\text{ref}} - I_{\text{measured}}$ 经过一 PI 调节器产生一定宽度的 PWM 波。电流误差 I_{error} 的大小正负决定了 PWM 波的宽度。当 I_{error} 等于零时, PWM 的宽度不变, 当 I_{error} 过大即参考电流大于实际电流很多, 使得 PWM 宽度大于控制周期时, 就令 PWM 的宽度为控制周期, 此时输出最宽的 PWM 波以最快地增大转速, 当 I_{error} 过小(为负值)即参考电流小于实际电流很多, 使 PWM 宽度小于零, 就令 PWM 的宽度为零, 此时以最快的速度降低转速。由 DSP 输出的六相 PWM 波, 经由功率驱动电路控制功率逆变器晶体管的开关模式, 进而控制直流无刷伺服电机的转矩和转速。

电流闭环的转移函数如下:

$$R(z) = K_p + \frac{K_i}{1-z^{-1}} T_s C$$

这里, K_p 是比例系数, K_i 是积分系数 $T_s C$ 是电流环采样周期。

系数 K_i , K_p 通常用来调节闭环系统, 如图 4-8 所示, 通常它们是建立在期望的动态闭环模型上的参数。

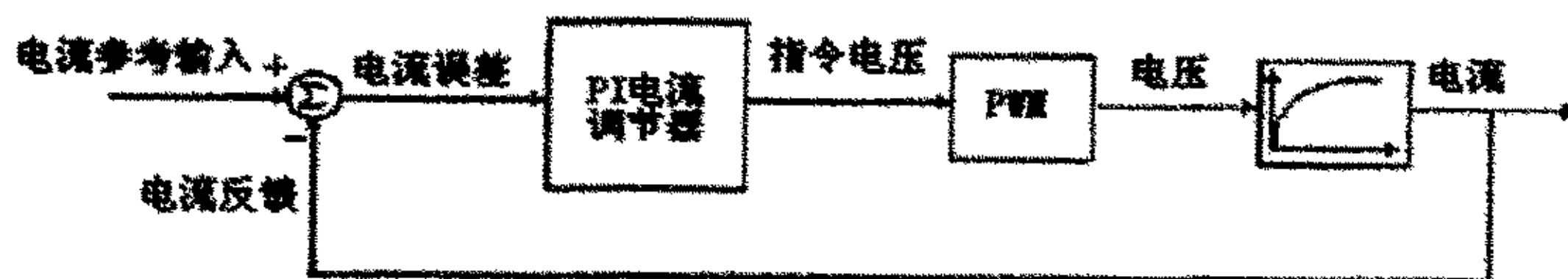


图 4-8 电流环的结构图

电流控制器的实现原理图如图 4-9 所示:

在这里引入了饱和系数 (SATI), 这是为了抑制积分调节器在调节过程中可能出现的积分饱和现象^[8]。

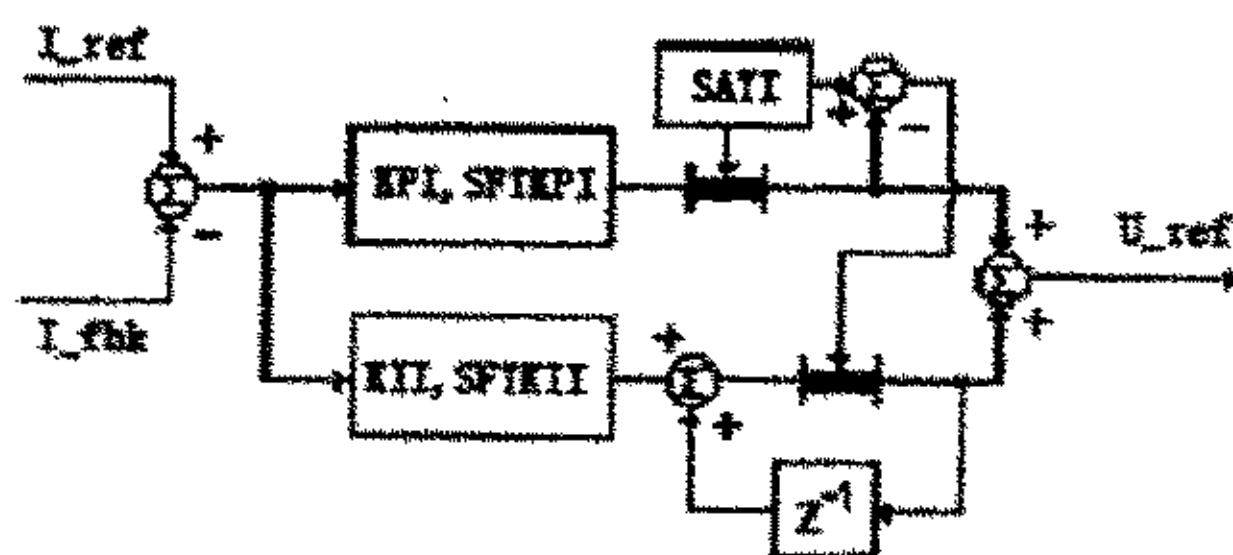


图 4-9 电流控制器的实现原理图

在实际电机控制系统中, 控制变量 u 的输出值要受到元器件或执行机构性能的约束 (如电源的限制、放大器饱和等) 而限制在有限范围 $[u_{\text{max}}, u_{\text{min}}]$ 内, 即 $u_{\text{min}} \leq u \leq u_{\text{max}}$

如果微机通过位置式 PI 算法得到的控制变量 u 在上述范围内; 那么 PID 控制可以达到预期的效果。一旦超出上述范围, 那么实际执行的控制量就不再是计算值, 由此将引起不期望的效应, 这类效应通常称为饱和效应。这类现象在电机启动、停车过程中, 以及负载或给定值发生突变时, 特别容易发生。

如果内在给定值从零突变,引起偏差的阶跃,控制量 u 可能超出范围,那么实际上控制量只能取上限值 u_{\max} 而不是计算值。此时系统输出值虽然仍在不断上升,但由于控制量受到限制,其增长要比没有限制时慢,其偏差 e 将比正常情况下持续更长的时间保持在正值,而使积分项有较大的累积值。当输出量超出给定值后,开始出现负偏差,但出于此时积分项的累积值已很大,还要经过相当一段时间后,控制量 u 才脱离饱和区,这样会使系统出现明显的超调。这种饱和作用是由积分项引起的,故称为积分饱和。

所以引入饱和系数 (SATI) 是为了使控制量在超出 u_{\max} 时削弱积分的作用,使控制系统不进入积分饱和区,从而有效地抑制地积分饱和现象地发生。

为了防止调节过程中产生过高冲击电流,对调节电流设置上限,电流调节规律设计如下:

$$I_{\text{error}} = I_{\text{ref}} - I_{\text{measured}}$$

$$\text{Duty_cyclenew} = \text{adjust}(I_{\text{error}})$$

If Duty_cyclenew \geq Suplimit Then

$$\text{Duty_cyclenew} = \text{Suplimit}$$

If Duty_cyclenew ≤ 0 Then

$$\text{Duty_cyclenew} = 0$$

同样, I_{error} 为电流偏差; I_{ref} 为电流设定值; I_{measured} 为电流测量值; Suplimit 为 PWM 波形脉冲宽度上限; Duty_cyclenew 为 PWM 波形脉冲宽度更新值。

4.6.2 速度闭环

其次是速度环,速度中环同样采用离散数字 PI 控制,其速度参考信号为位置外环的输出,其速度调整的输出作为电流信号的参考给定。

$$\text{速度中环的传递函数定义为: } R(z) = K_p + \frac{K_i}{1-z^{-1}} T_s C$$

同样, K_p 是比例系数, K_i 是积分系数 $T_s C$ 是速度环采样周期。

系数 K_i , K_p 通常用来调节闭环系统,如图 4-10 所示,它们同样是建立在期望的动态闭环模型上的参数。

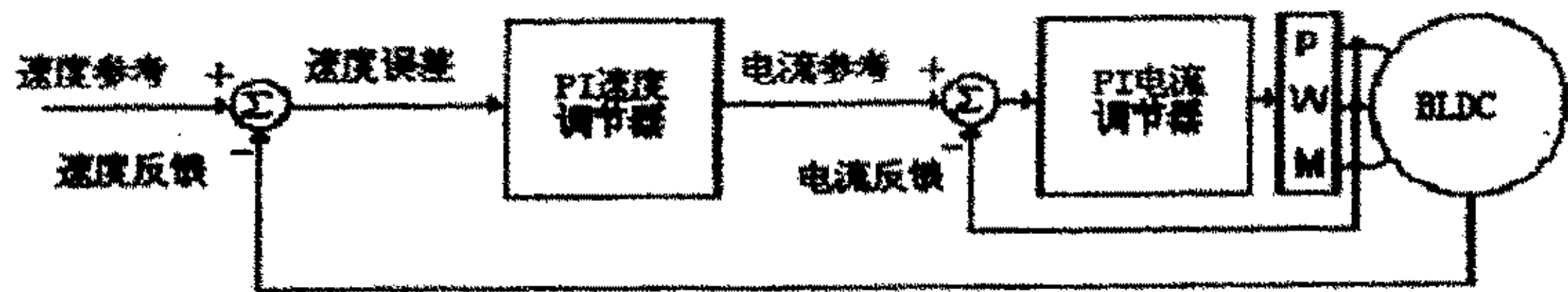


图 4-10 速度环的结构图

速度控制器的实现原理图如图 4-11 所示：

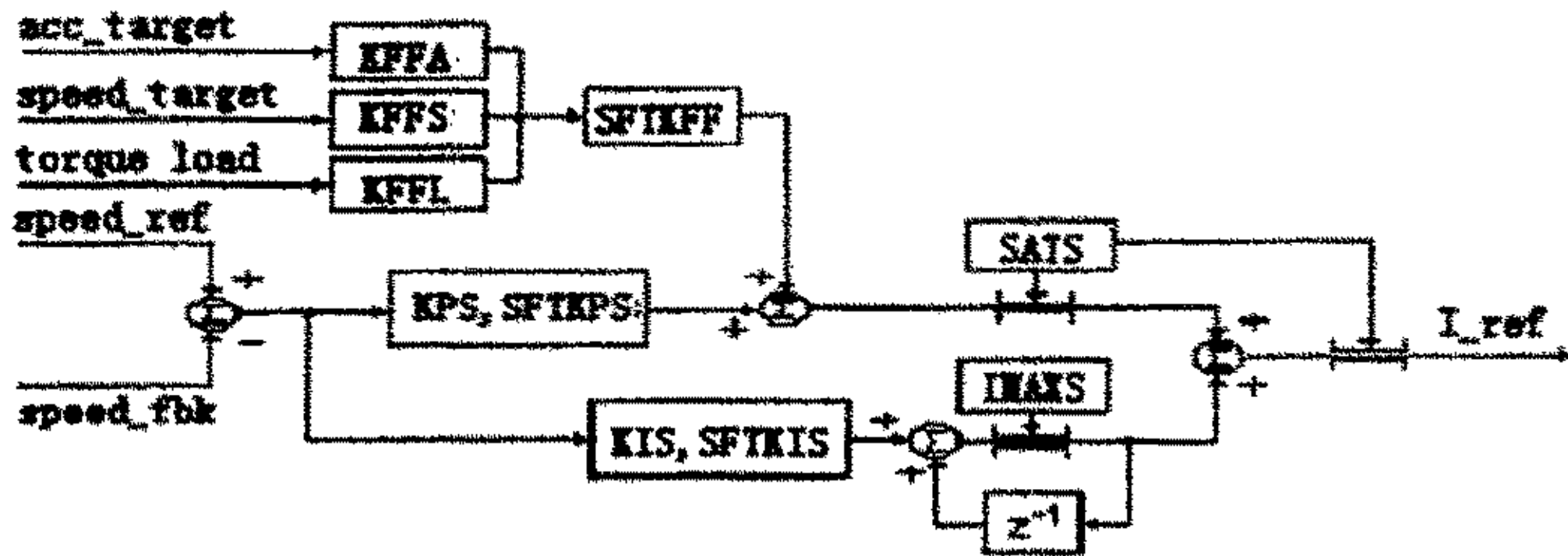


图 4-11 速度控制器的实现原理图

在这里同样引入了饱和系数（SATI），这是为了抑制积分调节器在调节过程中可能出现的积分饱和现象。除了接受标准的控制结构，速度控制器还可以接受前馈信号参数，如加速度相关部分（惯性作用模型 acc_target ），速度相关部分（粘滞摩擦所用模型 $speed_target$ ），负载转矩部分（负载转矩作用模型 $torque_target$ ）。如果这些变量的相关参数已知，控制系统则能得到较大的改进。

4.6.3 位置闭环

为了提高系统快速响应性，位置闭环采用变结构控制，如图 4-12 所示，控制器是一个 PID 调节器与常值切换的简单滑模变结构控制。令位移误差量 e_L 为给定值 g 与反馈值 f 之差，若 $|e_L| > \delta$ （为可调值）时，系统受常值 K_0 控制，这时 PWM 占空比为 1，电机加全电压输入，高速运行，以使 e_L 迅速减小。当 $e_L < \delta$ 时，电机采用 PID 调节，随着误差的减小，PWM 占空比成比例减小，电机端电压和转速也成比例减小，以利于电机无超调地逼近 $e_L = 0$ ，这里取误差量绝对值是因为系统另设有正反转判别电路。当 $e_L > 0$ 时，电机正转； $e_L < 0$ 为反转。一旦正反转确定以后，电机调节仅为斩波调速，速度调节指令只跟误差大小有关，与其正负无关。由于系统根据 e_L 所处不同范围采用不同控制结构，所以只要选择合适的参数 K_0 和 δ 值就可实现滑模变结构控制，从而产生不同的速度调节指令。

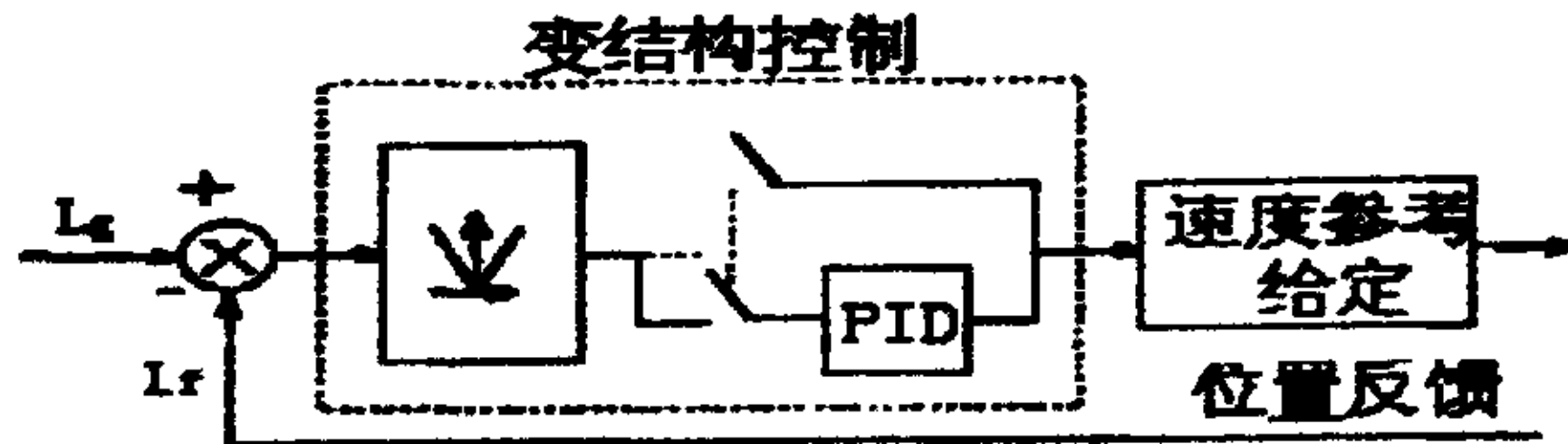


图 4-12 位置控制系统框图

位置 PID 调节器的传递函数定义为：

$$R(z) = K_p + \frac{K_i}{1-z^{-1}} T_s - S + \frac{K_d}{T_s - S} (1-z^{-1})(1+z^{-1}K_{df})$$

其中， K_p 是数字比例系数， K_i 是数字积分系数， K_d 是数字微分系数， K_{df} 是微分的过滤系数， $T_s - S$ 是位置闭环的采样周期，采样周期等同于速度闭环的采样周期。

由位置变结构外环，速度中环和电流内环的位置控制器结构图如图 4-13 所示：

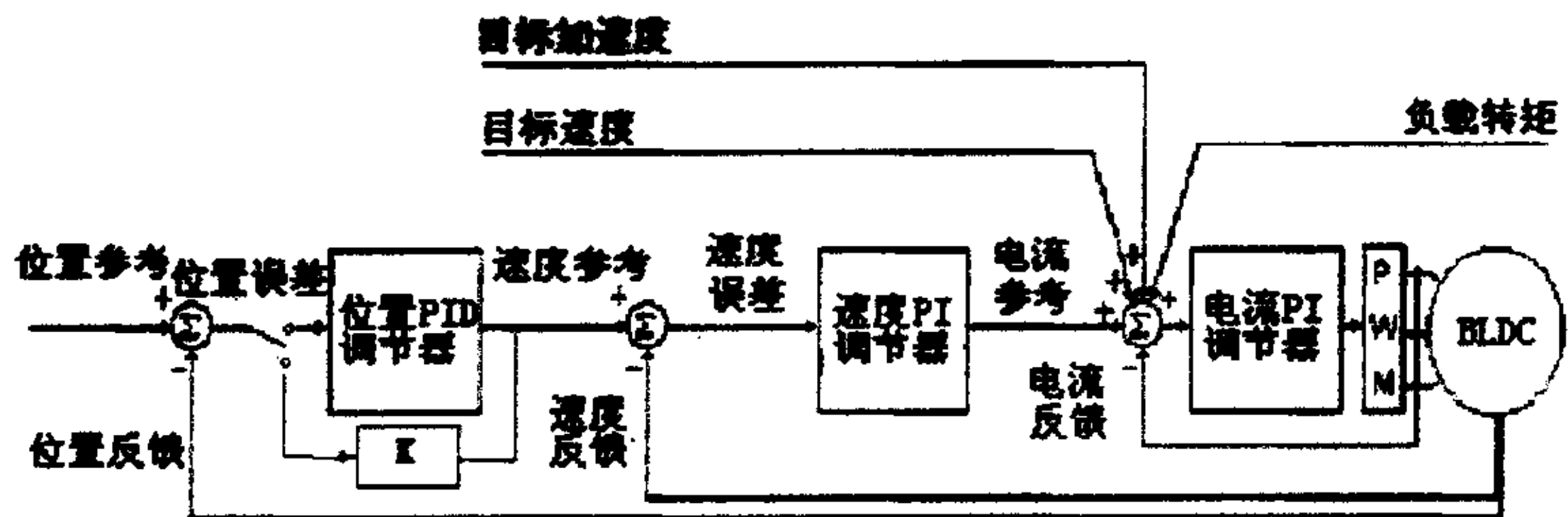


图 4-13 带速度电流内环的位置控制器结构图

位置 PID 调节器的原理框图如图 4-14 所示：

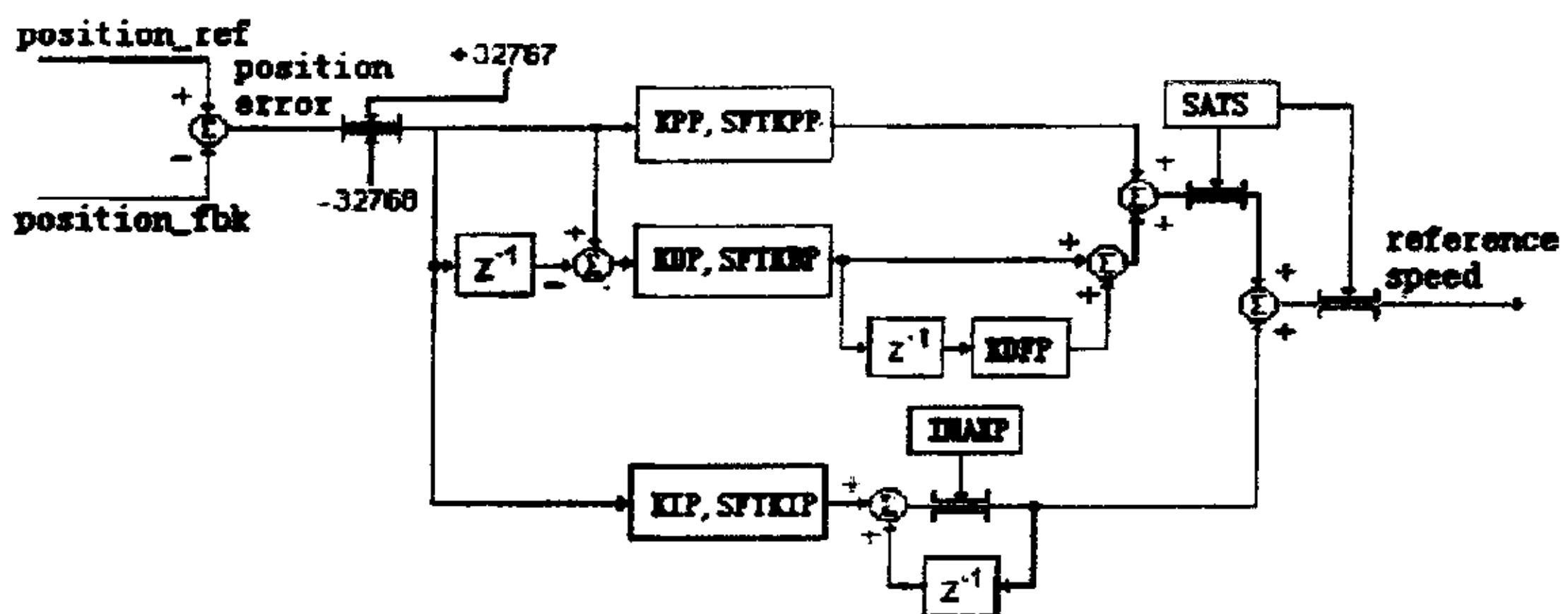


图4-14 位置控制器的原理框图^[14]

其中，position_ref 为位置参考量，speed_fbk 为速度反馈量，speed_ref 为速度参考输出量。

比例部分（KPP，SFTKPP），积分部分（KIP，SFKIP，IMAXP），微分部分（KDP，

SFTKDP, KDFP), 饱和系数 (SATP)。

此外, 本系统还在位置外环中添加了位置环的挂起功能, 即当电机旋转到特定位置并开始静止不动时, 其当前位置于目标位置的误差小于一定规定值并保持以给定时间时, 控制闭环系统被挂起。当位置误差重新超过规定误差范围时, 控制闭环又被重新唤醒。这样在电机停止不动的时候可以减小系统的功耗。

4.7 直流无刷伺服电机正反转、启动、调速实现

直流无刷伺服电机本体由多相定子绕组和一定极对数的永磁转子所组成, 结构如同常规的永磁同步电机, 这些绕组分别与电子开关线路的相应开关元件连接。电枢绕组的电流随着转子位置的变化而按一定规律换流, 功率开关元件的导通与截止是与转角同步变化的, 通过改变加在绕组上的直流电压就可实现调速, 因此直流无刷伺服电机具有直流电机的特性。

由于直流无刷伺服电机定子绕组串联了晶体管开关, 转子采用了永磁钢, 直流无刷伺服电机不能采用通常直流电机改变磁极极性的方式, 而只能通过定子绕组磁势和转子磁场相对关系来改变转子的旋转方向, 只要能得到空间上相差 180° 的驱动信号去控制相应的绕组导通, 就能实现正反转, 这在 DSP 中通过一定延迟的查表程序很易实现。

直流无刷伺服电机和一般直流电机一样, 靠改变电二端的直流电压大小来改变电机的速度, 考虑到直流无刷伺服电机的特殊情况, 可以有两种改变电压的方法, 一是每相导通时间不变, 改变加在线圈上的电压幅度, 电压幅度可用串在电源的调整管来实现, 但线路损耗大; 二是电压幅度不变, 只改变加在线圈上的电压波形占空比。如改变每相导通时间、周期或关断时间, 实质是改变加在线圈上的平均电压的大小, 使用时通过 DSP 根据控制要求, 经过计算, 输出一个宽度可调的脉冲信号, 实现调速。通过实验发现, 改变关断时间(定宽调频法)电机运行较平稳, 控制效果较好。

4.8 直流无刷伺服电机的升降速的控制

直流无刷伺服电机作为执行元件的通过速度控制可以实现快速启停能力, 若负荷不超过直流无刷伺服电机所提供的动态转矩值, 就能迅速使直流无刷伺服电机启动或停止。如果直流无刷伺服电机是以逐渐加速到最大值, 然后逐渐减速到零的方式工作, 其速度可增加 2~8 倍, 而不失控。在实际应用中, 在大多时候需要直流无刷伺服电机

以较高的速度工作,这就需要我们根据直流无刷伺服电机及负载的机械惯性以最优化的曲线升速和降速,即直流无刷伺服电机启动时,以低于响应占空比 P_r 的速度运行,然后慢慢加速到一定速度后,就以此速恒速运行,当快到终点时慢慢减速至 P_r 以下,直至停机定位。这样,直流无刷伺服电机就可以以最快的速度走完全程,且可以达到较准确的定位。

4.8.1 升降速曲线设计

在机械系统的负载惯性一定的条件下,为了保证直流无刷伺服电机高效运转,对其加速度必须加以控制。目前常用的是定常控制,即按照一个确定的运动规律进行控制,例如使直流无刷伺服电机的加速度为一个常数,这时的升降速曲线为一条直线,如图 4-15 所示。

电机在启动的时候以恒定的加速度提升速度,在电机速度达到指定恒定值时以恒速运行,在要停止时以固定的减加速度降速,直至电机停转。这种速度控制方法在对电机进行定位时能达到比较好的效果。

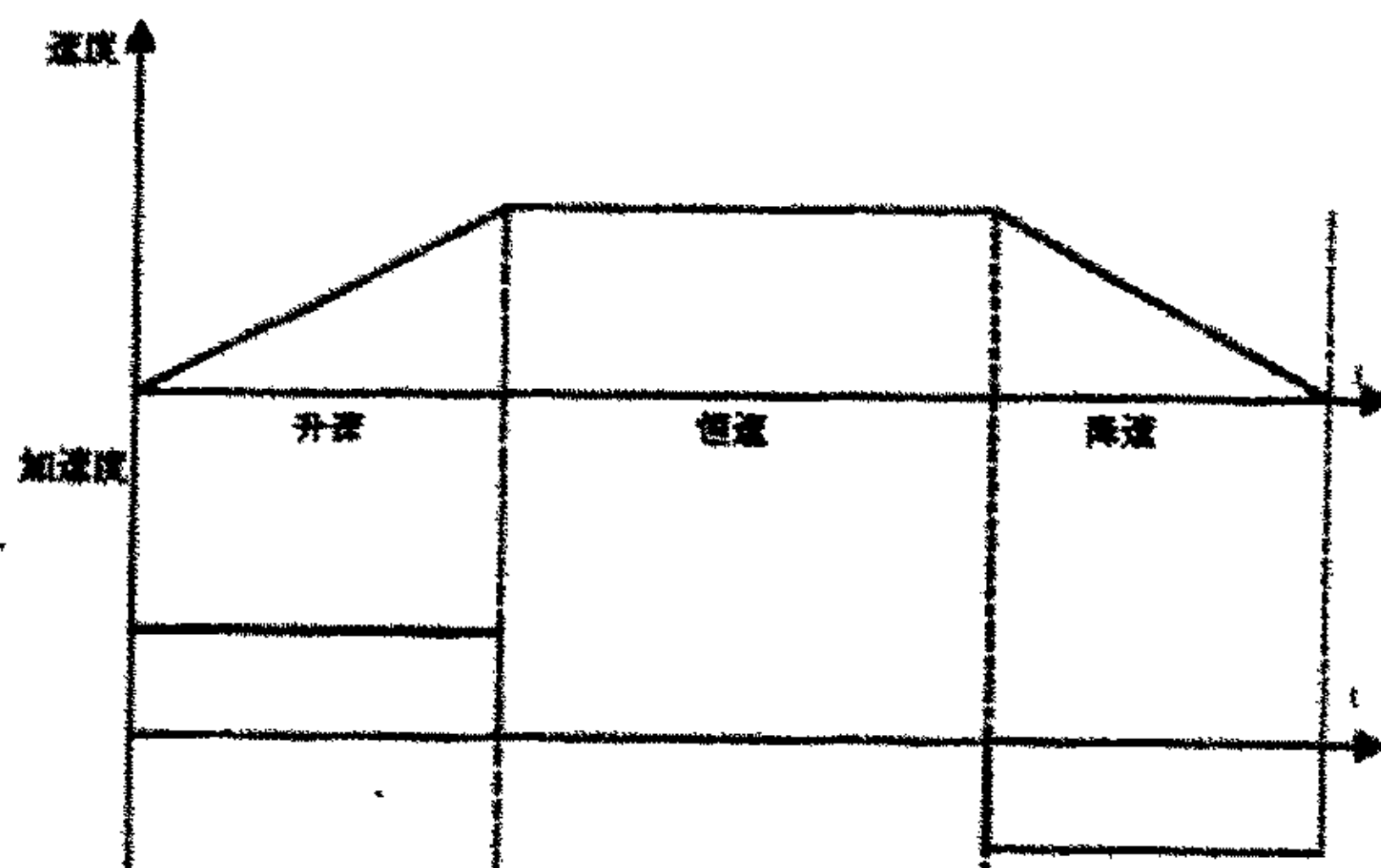


图 4-15 加速度为常数的升降速曲线

4.8.2 升降速控制的实现

为实现直流无刷伺服电机的升降占空比,在系统设计中,一般采用硬件设计或软件设计。硬件实现使用积分电路和微分电路实现,其元器件参数一经选定,曲线形状就固定不变,但因为每个电机及其负载的机械特性不同,所以它不能最好地与所有的电机适配。而且随着系统使用时间越来越长,元器件参数会发生变化,造成电机升降频中的阻尼。为克服硬件设计这些不能灵活变动和元器件变值的缺点,我们采用程序来实现升降占空比的控制。软件控制比起硬件来说,具有运行可靠,改动方便的特点,而且可以降低成本。针对不同电机的机械特性,我们专门设计了一个数据区,这个数据区中的数据

即确定了指数曲线的形状。我们改变电机时，只要根据电机的特性改变数据区的一些数据，即可使曲线很好的与电机适配。具体设计如下：首先用细密的阶梯逼近曲线，使直流无刷伺服电机按照每个阶梯的占空比间隔和速度一个阶梯一个阶梯地运行，其占空比也就随着阶梯的升高而升高，实现升速；减速时则反之。直流无刷伺服电机在每个阶梯上以一定的速度改变一定的占空比变化量，速度由所输出的不同占空比控制，将每一变化的占空比常数按照从低到高的顺序存入数据表，用下面的程序采集数据表中数据，控制直流无刷伺服电机运行，实现升降速的控制。

首先将各占空比的数值按实际使用的 DSP 类型转换成相应占空比常数依序存入表中，在开始运行后，将表首数据即突跳占空比数据放入定时器中，待第二次时钟中断到来时送下一占空比常数，同时把表中下一个占空比数据即下一步的数据常数送入定时器中，这样重复直至升至最速，然后依实际需要始终把最高占空比常数送入定时器中以保持最高速运行，至需要减速时则反查表送数据，或者另有一张减速的表格（减速时加速度可取得大一些，占空比变化数目可少些），这样便实现了平稳升降速的控制。升降速使用同一张表时程序大致流程图如图 4-16 所示。

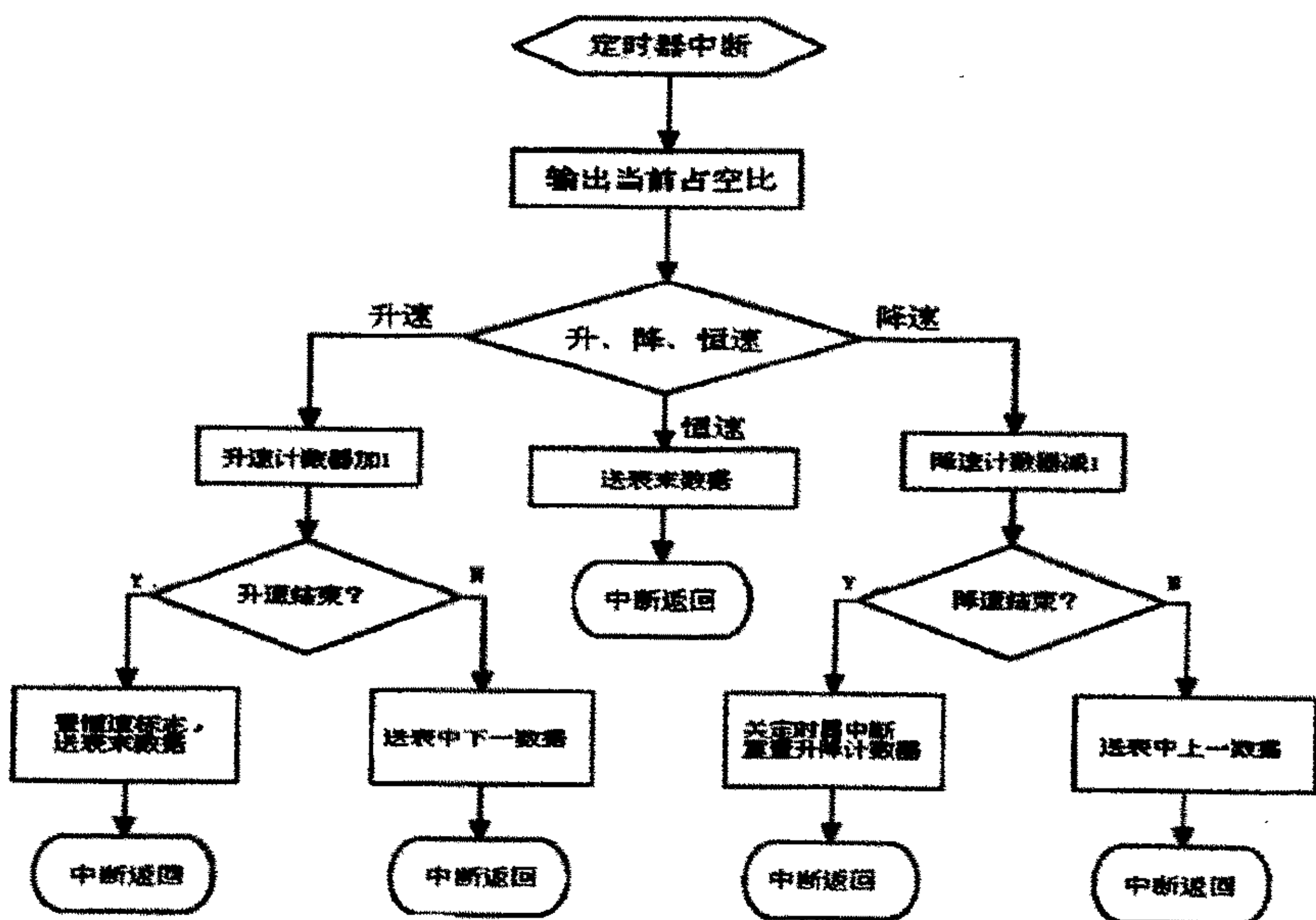


图 4-16 升降速控制程序流程图

4.9 键盘、显示电路及上位机监控的实现

由于 LF2407 本身提供了比较多的 GPIO(general purpose IO), 所以可以直接利用 IO 口实现简单可靠的键盘电路, 通过阵列式键盘配置可以实现多个不同键码的取值。显示方面 LF2407 通过扩展的 IO 空间采用直接控制方式直接控制点阵式液晶模块, 显示文字与图象信息, 达到良好的人机界面。点阵式液晶模块采用 DGM-12232 (122*32) 集成 EPSON 的 SED1520 控制器的液晶显示模块, 它控制简单, 价格经济, 并且和 DSP 之间提供了比较好的接口。

本系统还应用了 DSP 的 SCI 接口实现了与上位机和现场总线的通讯, 主控板提供了两个 RS232C 的标准串行接口, 一个是连接上位机和和执行机构 CPU, 另一个是连接现场总线 (Profibus, HART) 与执行机构主 CPU。

由于 LF2407 自带 CAN 总线的接口, 所以本系统采用 TI 公司的 SN65HVD230D 作为 CAN 总线收发器, 外扩了 CAN 总线接口。

为了将来的扩展需要, 系统还预设了 USB 及 RJ45 接口和实时时钟管理系统 (采用 Xicor 公司的 X1228) 从而为各种外设提供了强大的支持, 具有极高的通用性。

此外, 本系统采用 TI 公司的 SN74LVC164245ADW 16 位总线缓冲和 SN74AHC16373 输出锁存器提供一定的 IO 口的输出, 通过对 LF2407 的 IO 寻址操作, 能方便的实现 IO 口的扩展。

4.10 保护环节

一些保护功能可以在直流无刷伺服电机的运行过程中被激活。一旦激活后, 系统将会不断监测, 当产生对应的超过预设的参量时, 系统将会通过 PDPINT 引脚的输入, 封锁 PWM 波形的输出。

本系统主要采取了以下几个保护环节:

1. 最大电流保护

系统通过电流传感器的输入, 监视当前电机电流的大小, 当电机运行的作大电流超过预定值运行一段规定的时间时, 电流保护系统被激活, 并禁止所有 PWM 波形的输出, 同时采取相应的保护措施。只得注意的是, 电流保护是基于 DSP 的快速闭环控制, 所以电流保护具有较高的可靠性。

2. I_{2t} 过温保护

除了通过模拟测量方法测量外部温度，达到过温保护外，LF2407可以采用基于电机和功率运放的热模型的温度估算模型来检测过温。

热模型是指：先假设一热电阻 R_{th} 和一定时间 τ_{th} ，因为热过程速度比较缓慢，所以这里可以采用积分模式进行。

积分过程的实现分为两个层次：在每一个慢闭环的采样周期（ T_{s_S} ）内更新电机电流值的平方和，并在给定的时间 h_{th} 中重复此过程（相应的求和个数位 N_{I2} （ $h_{th}=N_{I2} \cdot T_{s_S}$ ））在每一个 h_{th} 的时间内，最后一个采样 $N_{I2}t$ 的求和用来更新温度 θ 的估计值。

$$S_i = \beta \sum_{j=1}^{N_{I2}t} I_j^2 \quad \theta_i = \theta_{i-1}(1-\alpha) + S_i$$

这里

$$\beta = \frac{R_{th} g R g T_{s_S} - S}{\tau_{th}}, \alpha = \frac{h_{th} g T_{s_S} - S}{\tau_{th}}$$

参数（ α 和 β 取决于电机参数（电阻 R ，热电阻 R_{th} ，时间限制 τ_{th} ）和采样参数（ T_{s_S} 和 h_{th} ），这些参数必须由用户定义。

3. 接地保护

接地保护是对于三相直流无刷伺服电机的三相电流和大于规定值，这意味着可能存在单相对地短路。接地保护也是在 LF2407 的高速实时闭环内核控制系统中完成，具有高可靠性。一旦发生此类情况，LF2407 同样会产生中断，并采取中断保护措施。

4. 过电压保护

过电压保护是检测DC侧的电压是否超过一预定值，同样在LF2407高可靠性的高速实时内核中完成。当过电压发生时产生中断，并采取中断措施。

5. 欠电压保护

该保护是检测DC侧的电压是否低于一预定值，同过电压保护。

6. 过温度保护

过温度保护是检测稳定传感器的电机运行时的温度是否超过预定的温度，温度传感器采用 DALLAS 公司产的 DS18B20 单总线温度传感器，和连接 LF2407 只需一根线即可，接口简单操作方便。当所测得的温度超过预设值时，通过软件中断采取一定的保护措施。

7. 控制错误保护

LF2407 可以在位置外环中不断监视参考值和实际值之间的误差。如果误差值大于

给定值达到给定时间,则产生控制错误保护,它同样在 LF2407 的高速实时内环中完成,发生后产生中断。

4.11 数字逻辑部分

数字逻辑部分主要解决的是数字电路中的译码,电平转换,数字信号控制等,其主要作用是辅助 DSP 进行一些外围的数字逻辑信号的硬件处理。

考虑到单独的数字逻辑处理芯片所构建的数字系统的复杂性,本系统采用 ATERA 公司产的 MAX7000 系列的一款复杂可编程逻辑器件(CPLD) MAX7128S^[15]来实现数字逻辑的处理。

4.11.1 MAX7128S 简介

MAX 7000 系列是工业界中速度最快的高集成度可编程逻辑器件系列。MAX 7000 系列(包括 MAX 7000E、MAX 7000S 和 MAX 7000A 器件)的集成度为 600—5000 可用门,有 32—256 个宏单元和 36—155 个用户 I/O 引脚。这些基于 EEPROM 的器件能够提供组合传输延迟快至 5.0 ns, 16 位计数器的频率为 178MHz。此外,它们的输入寄存器的建立时间非常短,能提供多个系统时钟且有可编程的速度/功率控制。MAX7000E 器件具有最高集成度,是 MAX 7000 系列的增强型。MAX7000S 器件也具有 MAX 7000E 器件的增强特性,是通过工业标准 4 引脚 JTAG 接口实现在线可编程的。

MAX7128S 是用 CMOS EEPROM 单元实现逻辑函数的。用户可编程的 MAX 7128 结构可容纳各种各样,独立的组合逻辑和时序逻辑函数。在设计开发相调试阶段,MAX7128 器件可以快速而有效地重新编程,并保证可编程擦除 100 次。

MAX7128 包含 128 个宏单元。每 16 个宏单元组成一个逻辑阵列块(LAB),同时,每个宏单元有一个可编程的“与”阵和固定的“或”阵,以及一个具有独立可编程时钟、时钟使能、清除和置位功能的可配置触发器。为了能构成复杂的逻辑函数,每个宏单元可使用共享扩展乘积项和高速并联扩展乘积项,向每个宏单元提供多达 32 个乘积项。

MAX7128 系列提供可编程的速度/功率优化。在设计中,影响速度的关键部分工作在高速/全功率状态,而其余部分工作在低速/低功率状态。速度/功率优化的特性允许用户可以配置一个或多个宏单元工作在 50%或更低功率下,且仅增加了一个微不足道的延迟。MAX7128 器件也提供减缓输出缓冲器的电压摆率选择项,以降低不要求速度信

号的开关噪声。所有 MAX7128 器件的输出驱动器能设置在 3.3V 或 5.0V 电压下工作,并且允许 MAX 7128 器件在混合电压的系统中使用。

MAX7128 系列由 Altera MAX + PLUS II 开发系统支持,该系统是个单一的集成软件包,它提供电路图、文本和波形等设计输入方式;执行编译和逻辑综合、仿真相定时分析,以及器件编程等工作。MAX + PLUS II 具有 EDIF、VHDL、Verilog HDL,以及其它的网表接口,便于为工作在工业标准 PC 机和工作站上的各种 EDA 工具提供附加的设计输入和仿真支持。

4.11.2 基于 MAX + PLUS II 的数字逻辑电路的设计

本系统的数字逻辑电路设计主要是 DSP 地址线的译码,3.3V 和 5V 的电平转换以及一些 IO 口的扩展。

通过对 LF2407 数据空间,程序空间,以及 IO 空间的地址译码可以充分利用 DSP64K \times 3 的寻址空间。译码工作主要由所设计的译码器完成,通过对真值表的配置,来选定指定的输出。译码器逻辑电路图如图 4-17 所示。

图中加入了逻辑跳线 JP12, JP13 的输入是为了满足 DSP 对不同存储芯片寻址空间的分配,在这里主要是为了增加系统的应用范围提高系统的灵活性。

3.3V 和 5V 的电平转换主要是定义在 CPLD 的输入输出 IO 管脚的电平,由于 MAX7128S 自身能实现各种不同逻辑电平的逻辑转换,所以在其内部能用混合逻辑信号 3.3V 和 5V 进行操作并按指定的电平输出。

IO 口的扩展是指通用输入输出口的扩张,由于 LF2407 有 64K 的 IO 寻址空间,通过 MAX7218S 内部的 IO 空间译码,并同过 MAX7128S 内部构建的锁存器件。IO 扩展的逻辑电路图如图 4-18 所示:

图中,锁存器件采用 373,并由译码器控制锁存信号,这样在 DSP 中只需一条简单的 IO 操作指令即可以实现 16 位数字信号的输出。

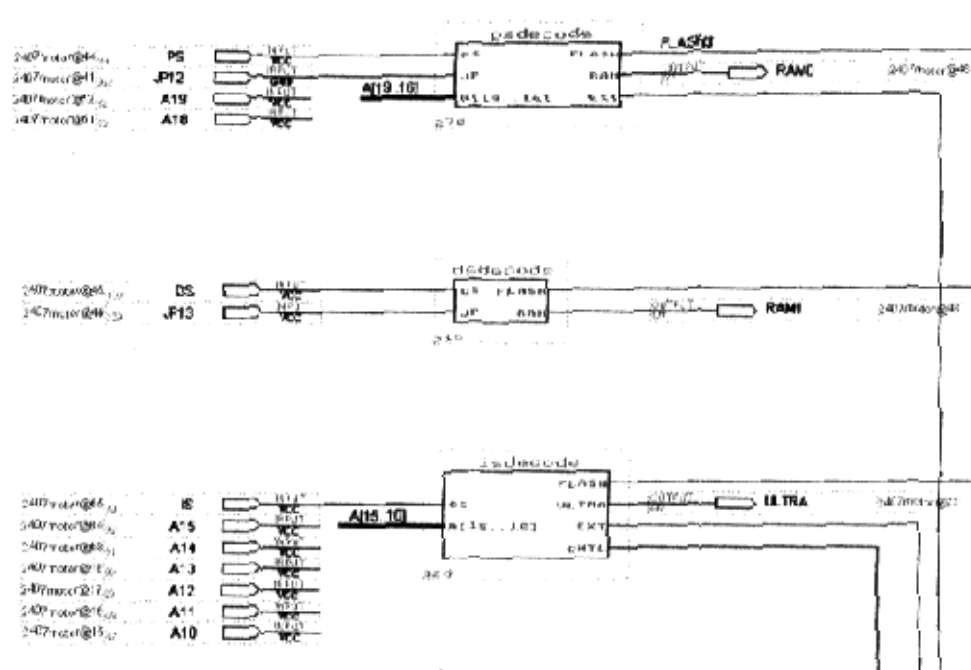


图 4-17 译码器逻辑电路图

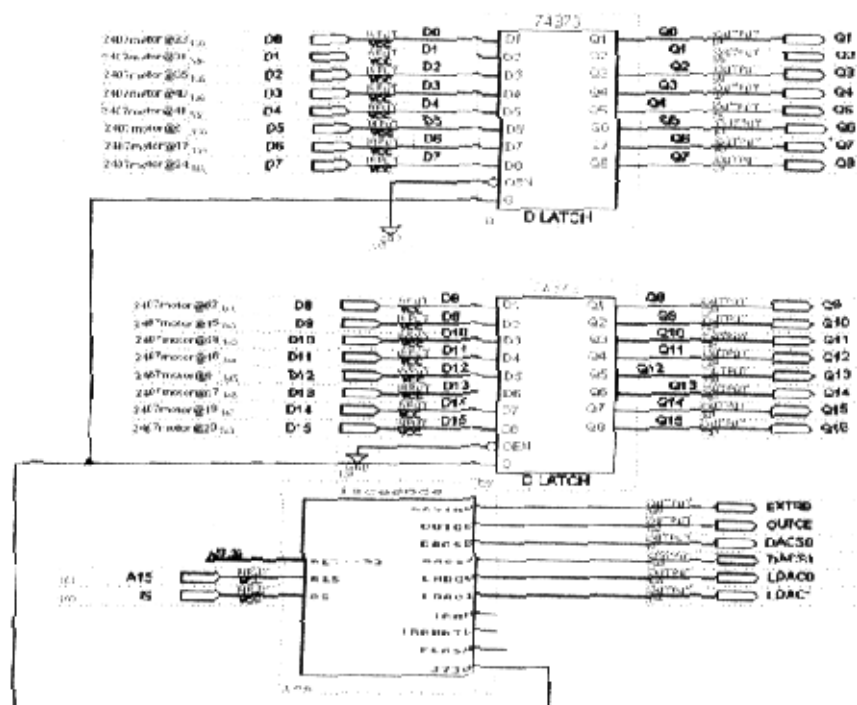


图 4-18 IO 扩展的逻辑电路图

此外, MAX7128S 在本系统中还提供了各种跳线的控制, 使系统具有很大的灵活性, 适应于各种不同的工作环境。其中很多跳线控制是为了方便调试所设置, 是系统无需改变 PCB 板的结构, 只需重新对 MAX7128S 编程, 即可实现各种不同工作环境的调试方案, 这样大大方便的系统维护。

第五章 DSP 的直流无刷伺服电机的控制系统的硬件实现

硬件系统由控制和驱动两部分完成, 并且由两块 PCB 板构成, 即主控制板与驱动板。主控板主要完成对整个系统信号输入, 信号处理与信号输出。驱动板主要完成功率放大, 三相逆变等工作。

5.1 主控制系统

主控制系统主要包括 TMS320LF2407 DSP 数字信号处理器最小系统, 模拟信号的输入接口, 外围接口器件。

5.1.1 TMS320LF2407 的最小系统

TMS320LF2407 DSP 数字信号处理器最小系统^[11]主要包括了 DSP 主芯片 (TI 公司生产的 TMS320LF2407), 内存芯片包括 ROM, RAM (ISSI 公司产的 64KSRAM 芯片 IS61LV6416) 电源管理芯片 (TI 公司的 TPS767D301), CPLD 数字逻辑门处理和译码等 (ALTERA 公司的 EMS7128), 7.5M 晶振, JTAG 口以及一些阻容器件。

TMS320LF2407 最小系统是构成主控制系统的基本单元, 当最小系统建立起以后, 就可以进行一些最基本的硬件仿真。

TMS320LF2407 最小系统的原理图见附录 1。

5.1.2 模拟信号接口

模拟接口主要有 DA 输出^[14] (TI 公司产的 TLV5620), 温度传感器 (DALLAS 的 18B20) 信号调理电路以及模拟运放等。

模拟信号接口的硬件原理图见附录 2

5.1.3 外围设备接口

外围设备接口主要包括通信接口, SCI (MAXIM 公司的 MAX232), CAN (TI 公

司的 SN65HVD230D) 8 位总线缓冲 (TI 公司的 SN74ALV4246), 8 位三态寄存器 (TI 公司的 SN74AHC373), 实时时钟芯片 (XICOR 公司的 X1228), 点阵液晶显示 (DGM-12232) 和 3 位半的数字型液晶显示及其驱动芯片 CD4543 等

外围设备接口的原理图见附录 3

5.2 功率放大驱动

本系统中三相逆变器的驱动采用富士公司生产的 R 系列 IGBT 智能功率驱动模块 7MBP25RA120^[9], 由于该智能驱动模块整合了驱动、逆变、保护、制动等七项功能, 所以可省去很多外围器件, 而且操作起来非常简单, 大大方便了驱动部分的硬件设计。

5.2.1 逆变部分

逆变部分采用采用 FUJI 公司 R 系列 IPM 模块, 带制动电阻的 7MBP25RA120。该智能驱动模块与过去的 IGBT 模块和驱动电路的组合电路相比, 有很多优点。它是一款含内置制动电路的驱动, 逆变保护等为一体的智能逆变驱动模块。其驱动电路与 IGBT 的连线短, 驱动回路阻抗低, 无须反向偏置电源, 上下桥臂分别由 4 路电源供给上臂侧 3 路独立, 下臂侧 1 路公用, 大大提高了驱动的可靠性。其保护电路内部集成了过流保护 (OC), 短路保护 (SC), 控制电源欠压保护 (UV), 过热保护 (OH), 及报警输出 (ALM) 等均为内置, 这样对过电流, 过电压, 欠电压, 过温度以及短路等都提供了有效的保护。其内部结构框图如图 5-1 所示:

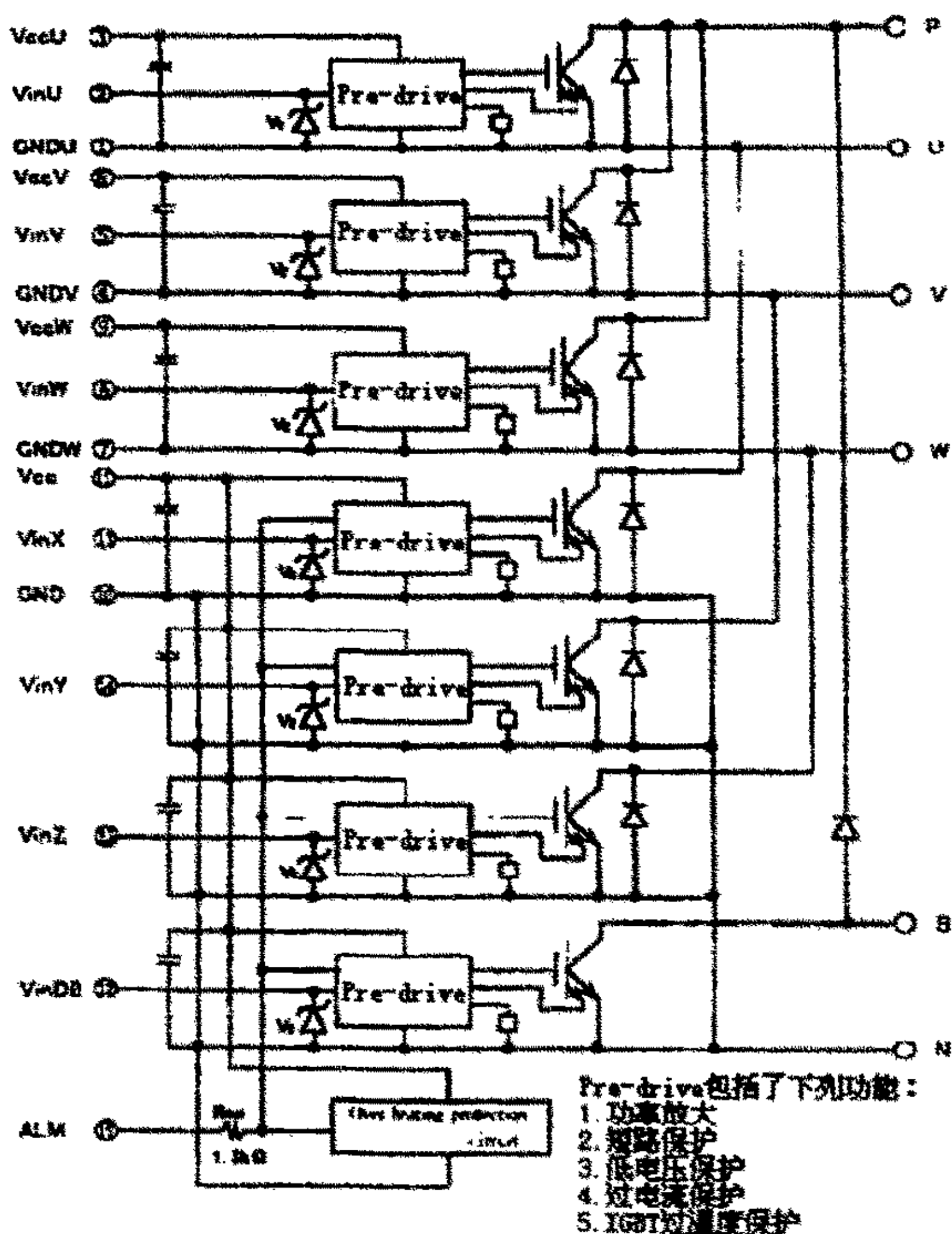


图 5-1 7MBP25RA120 内部结构框图^[9]

5.2.2 整流部分

整流部分采用 FUJI 公司 6R130G-160 (30A, 800V, 三相全桥) 整流模块, 它是一款不可控三相全桥整流模块, 最大能达到电压 800V, 电流 30A 的输出, 结构简单, 价格经济。其内部结构图如图 5-2 所示:

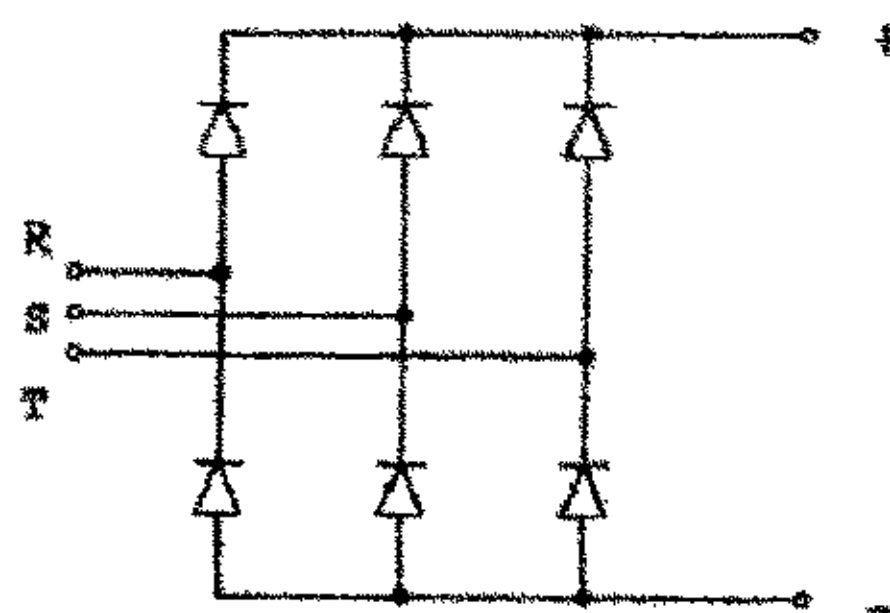


图 5-2 6R130G-160 内部结构图

5.2.3 隔离部分

驱动部分的所有输入输出都采用光耦隔离，其中驱动输入部分采用高速光耦 TOSHIBA 公司产的 TLP559（内部结构如图 5—3 所示）其他隔离采用 TOSHIBA 的廉价光耦 TLP521，用以实现控制回路驱动回路的隔离，提高了可靠性（内部结构如图 5—4 所示）。

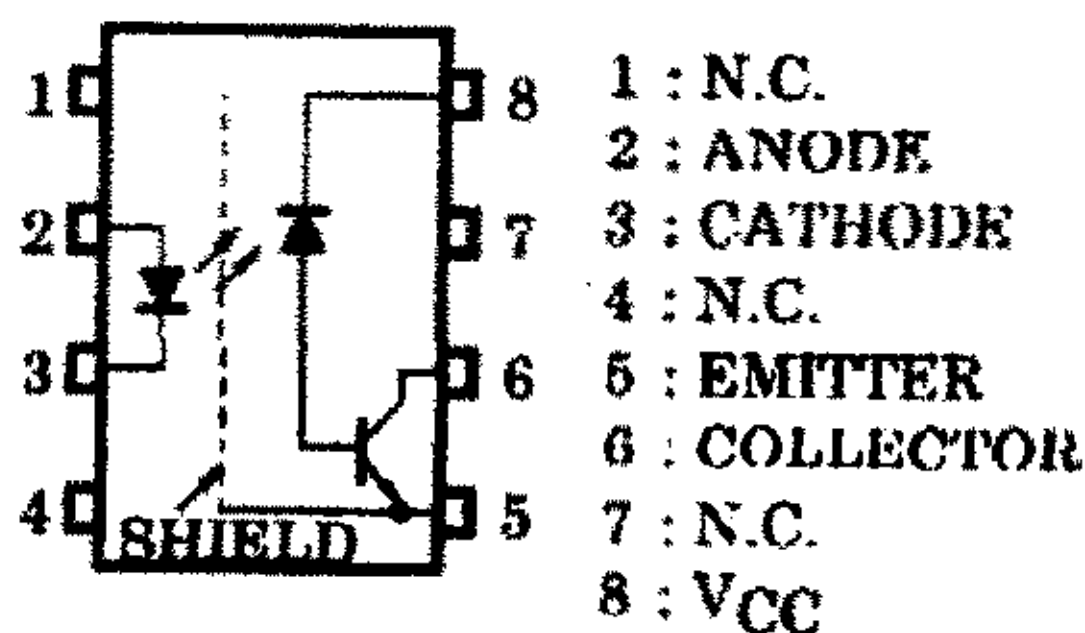


图 5—3 TLP559 的内部结构图

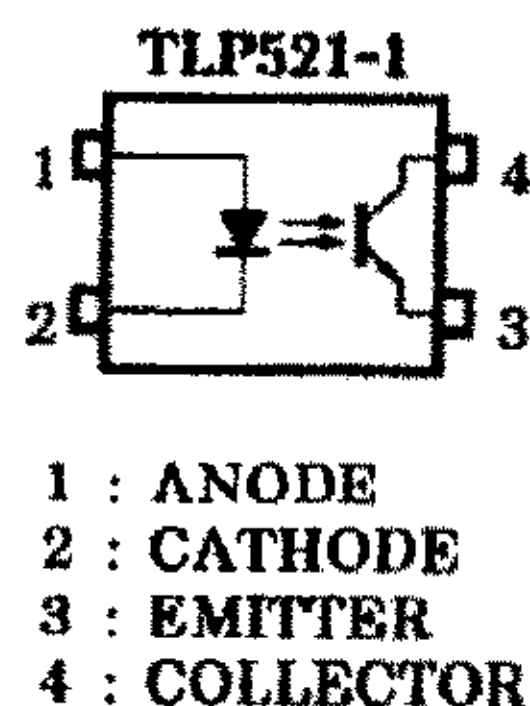


图 5—4 TLP521 的内部结构图

5.2.4 驱动电源部分

控制电源采用 4 组独立供电电源。

220VAC-15VDC。每组不共地，采用 SUPLET 公司 LAP5 模块。220AC-15VDC。

220VAC-5VDC：采用 SUPLET 公司 LAP15 模块。带 PFC 功率因数校正。

驱动输入：220V 单相 AC（线电压），380V 三相 AC（线电压），450V 直流输入

5.2.5 驱动板电路原理图设计

驱动板主要包括电源部分，智能驱动模块 IPM，隔离部分，传感器接口及其信号调理等（包括霍尔位置传感信号，增量式光电编码器，和电流采样电阻）。驱动板电路原理图见附录 4

第六章 基于 DSP 的直流无刷伺服电机控制系统的软件设计

系统的软件开发是在 TI 公司提供的 C2xxx (Code Composer ver4.10.36) IDE 开发环境中完成。该 IDE 集成开发环境内部集成了才提供汇编编译环境的同时还提供了 C 编译环境, 而且还提供程序连接下载, 直接支持 JTAG 口的在线调试功能。

系统的软件设计主要是控制算法的实现, 包括电流内环, 速度中环和位置外环。各个不同的环所采用的算法都有各自的特点, 其中速度内环采用经典的工业 PI 算法, 速度中环采用模糊单神经元自适应智能双模控制, 位置外环采用基于 PID 变结构控制, 各个环采用及联方式, 即上一环的输出作为下一环的输入。通过各个调节器的数字调节计算出满足要求的 PWM 波送入驱动部分。需要强调的是, 各个环的算法都由软件完成, 这正是充分发挥了 DSP 快速离散信号的数字处理的强大功能。

系统软件设计的源程序见附录 5。

6.1 软件开发环境 CC (Code Composer)

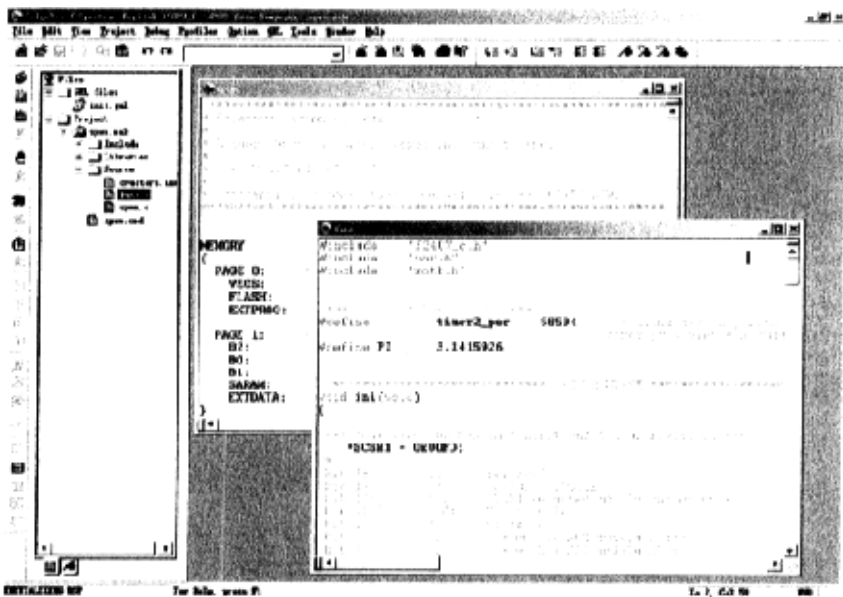
CC 是一种针对标准 TMS320 调试器接口的交互式方法。它包括了 HLL 调试器的主要特征, 除此以外, 它还有以下特性:

- 一个集成 TI 编译器的完全集成的环境 (CC 目标管理系统、内建编辑器和所有的调试和分析能力集成在一个 Windows 环境中)。
- 对 C 和 DSP 汇编文件的目标管理: 目标编译器保持对所有文件及相关内容的跟踪。它只对最近一次编译中改变过的文件重新编译, 以节省编译的时间。
- 高集成的编辑器调整 C 和 DSP 汇编代码: CC 的内建编辑器支持 C 和汇编文件的动态语法加亮显示, 使用户能很容易地阅读代码和当场发现语法错误。
- 编辑和调试时的后台编辑: 用户在使用编译器地汇编器时没有必要推出系统, 因为 CC 会自动将这些工具装载到它的环境中。在其出错窗口会加亮显示, 只要双击错误就可以直接达到错误处。
- 在含有浮点并行调试管理器 (PDM) 的原有 MS 窗口下支持多处理器。
- 在任何算法点观察信号的图形窗口指针。
- 文件探针在算法处通过文件提取或加入信号或数据。

- 支持图形分析能力。
- 可在后台执行用户的 DOS 程序。
- 提供技术状态观察窗口。
- 提供代码分解窗口。
- 支持对目标 DSP 的帮助。

CC 的可视窗口使用户能够容易理解复杂的结构。只要将光标放在相关的变量处并按 Enter 键, 诸如数组、结构、指针变量就可以递归地增加或减小。另外, 添加到可视窗口的变量, 也可以通过双击该变量来编辑。C 表达式和 GEL 函数也可以添加到可视窗口。将 GEL 函数添加到可视窗口, 就可以在每个断点处执行。由 GEL 函数可以执行更复杂的任务, 将结果输出到可视窗口。

探针允许用户观察信号或算法上加入或提取数据。它可以连接到结构点或存储空间。到达算法的指定点时, 已连接的信号探针就会从 DSP 中提取出数据并显示。如果将文件同指定点相连, 数据就会在指定的存储空间与文件之间传输。操作一结



6.2 控制算法的实现

控制算法主要是分别对三个环的控制,即电流内环的 PI 调节,速度中环的 PI 控制和位置外环的变结构 PID 调节。

6.2.1 电流环的控制

电流内环采用经典的 PI 调节器,该环的参考电流输入是速度中环的输出电流参考和通过 AD 采用回来的实际直流无刷伺服电机的电流大小,输出是调节后的电流值所对应的一定占空比的 PWM 波形。由于 PI 算法的积分环节存在积分饱和的可能,一旦积分饱和发生的话,将可能引起较大的超调量,从而影响到控制的性能。所以,在本控制系统中,电流闭环采用遇限消弱积分的 PI 算法,这种方法的基本思路是:一旦控制变量进入饱和区,停止进行增大积分项的运算。具体编程时,在计算 u_{ci} 时,先判断上一时刻的控制量 u_{ci-1} 是否已超出限制范围,如果已超出,即已进入饱和区,那么再根据偏差 e 的符号,判断系统输出是否在超调区域、由此决定是否将相应偏差值计入积分项。其算法流程如图 6-2 所示。

由于速度内环属于高速控制环,所以为了提高闭环的速度,在计算控制量时,采用增量式数字 PI 算法,这样可以大大节省 DSP 的运算时间,达到高速电流环 PI 控制的目的。

6.2.2 速度环的控制

速度环同样采用数字 PI 控制。速度环的参数输入是位置环的控制输出,另一个输入是增量式光电编码器速度信号输入,输出的是调整好的速度信号供电流闭环输入。速度控制的程序流程图大致和电流还相同,这里不再列举。

由于速度控制控制算法中需经过一定的迭代,所以速度输出给电流参考信号的速度可能不等同于电流控制的速度。但因为速度闭环对电机转矩没有直接影响,所以本系统中速度闭环采用较慢速的采样周期,这样可以减小 DSP 的运算量,通过采用合理的算法程序,达到高效的控制目的。

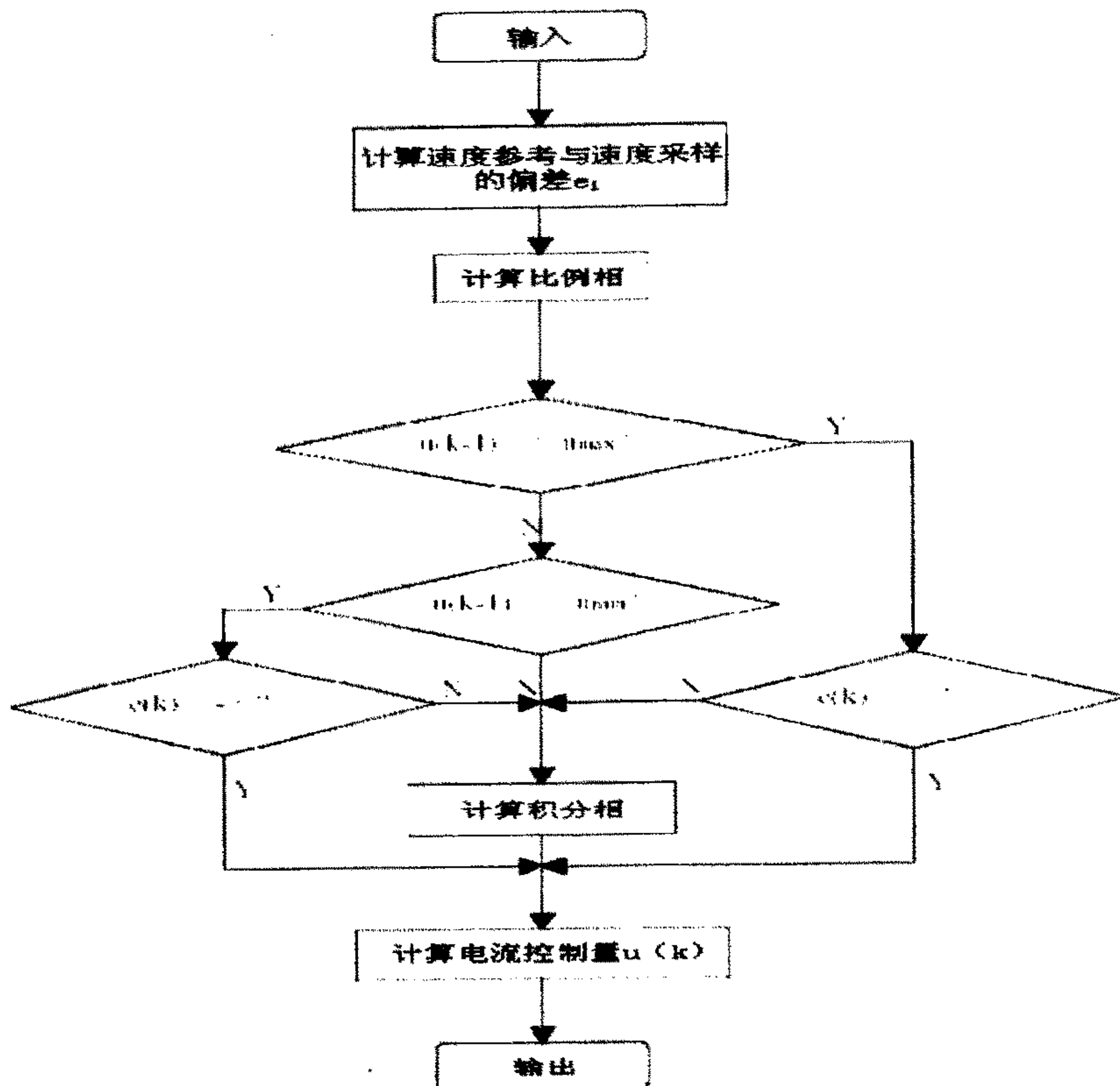


图 6—2 遇限消弱积分的 PI 算法流程图

6.2.3 位置环控制

位置环采用 PID 调节器与常值切换的简单滑模变结构控制。令位移误差量 e_L 为给定值 g 与反馈值 f 之差，若 $|e_L| > \delta$ (为可调值) 时，系统受常值 K_0 控制，这时 PWM 占空比为 1，电机加全压输入，高速运行，以使 e_L 迅速减小。当 $e_L < \delta$ 时，电机采用 PID 调节，随着误差的减小，PWM 占空比成比例减小，电机端电压和转速也成比例减小，以利于电机无超调地逼近 $e_L = 0$ 。位置环的输入是用户指定的位置参考和霍尔位置传感器位置信号的输入，输出是位置控制信号作为速度信号的参考出入。位置环控制的程序流程图如图 6—3 所示：

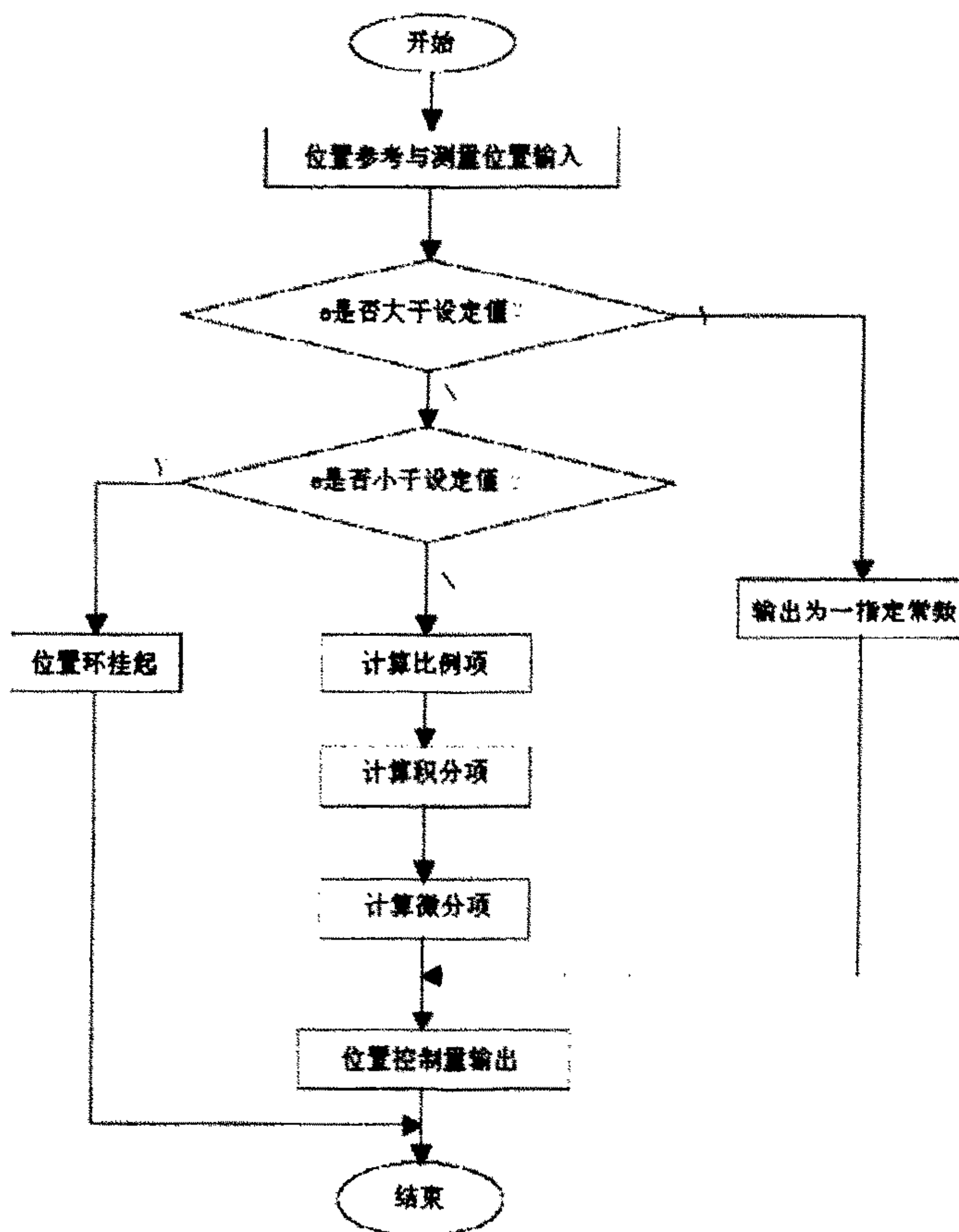


图 6-3 位置环控制的程序流程图

注意到，当位置环的误差小于某一特定值时（很小），此时可以认为电机已经静止，所以位置环相应地被挂起。

第七章 系统软件仿真模拟

为提高本系统的实际运行效果,必须进行复杂的参数调试。控制参数可以通过电机控制的数学模型来确定,但考虑到系统实际运行时的种种不确定因素,如被控电机本身的性能参数,环境温度,信号干扰,还有传感器的测量精度等等。所以采用直接的数学模型参数确定的方法和理想运行的参数之间存在一定的出入。一种解决的方法是反复试验,通过 DSP 侧软件的反复编程来改变控制参数设置,同时与实际输出的控制效果相比较,从中找到最好的输入输出关系。

这种直接测试方法固然非常直观简单,但反复的调试试验显然要增加系统的软件负担,而且通过这种简单的输入输出比较来确定控制参数的方法对不同运行环境的适应能力比较差。为减小反复调试的工作量,我们可以采用软件仿真模拟的方法来确定控制参数的取值。Technosoft 公司专为电机控制领域生产的一系列电机控制工具集(包括硬件和软件)为电机控制的设计提供了很大的方便。在这里我们选用 Technosoft 公司生产的电机运行仿真软件 DMAS-ac Toolbox 来实现系统的软件仿真。这样不但节省了开发设计周期,更大大减小了开发设计者的调试工作量。

7.1 DMAS-ac Toolbox

DMCS-ac 工具箱^[16]作为一款高效的图象化数字电机控制的软件,它是基于 Windows 运行环境,通过简单的鼠标点击就可以实现希望的电机控制结构。这个工具包同时还包括了一系列的调节算法来优化计算控制器的相关系数来达到期望的电机运行效果。为了实现这个目的而开发的这种新的设计方法,允许用户自由调节电流,速度位置控制器,而软件会自动计算出相关的系数送入 DSP 硬件侧予以实现。

DMCS-ac 工具箱应用是一种理想的软件包,其目的是根据观察和比较不同数字控制电机的运行状态,来定制出与之相适应的电机控制算法。

DMAS-ac Toolbox 是建立在两个高级图形化界面的开发工具,它们是 DMCTUN 和 DSPMOT32 程序。它由一系列实时运行的电机控制应用组成,如交流电机,无刷电机或感应电机,采用了不同传感的速度控制及位置控制。

DMCTUN 程序是一种专为电机控制应用而设计的 CAD 软件,其中的被控电机,放大器,各类传感器以及控制结构都是用户可自由选择的。此外电流、位置和速度控制器可

以被自动调节,从而计算出相关的控制系数来满足 DSP 侧优化控制系统的实现。DMACTUN 的应用使得我们可以选定特定的电机运行结构,为它设置参数以及为其设计控制器(电流,速度和位置)。最后,所选定的电机控制模型可以通过参数化的 DSPMOT32 程序下载到试验 DSP 系统使之自动运行。

DSPMOT32 程序代表一种图形化的工作平台,供分析和评价 DSP 侧电机控制的运行状态。通过 DSPMOT32 程序基本上可以实现如变量的取值,图形化设置的调节,特殊数据的实时采集以及参考源的建立等。DSPMOT32 的应用可以是我们所设计的控制结构下载到 DSP 硬件侧,以便我们对实时电机控制系统进行现场测试。

图 7-1 示出了 DMCS-ac 工具箱的功能框图

从图中可以看出 DMCS-ac 主要由硬件的定义,控制器的调节,应用参数重载, DSP 侧电机控制的实现和图形化分析和评估组成。

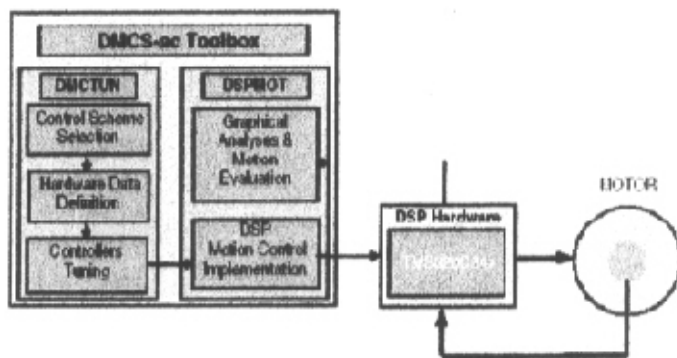


图 7-1 DMCS-ac 工具箱的功能框图

7.2 DMAS-ac Toolbox 的配置

DMAS-ac 的仿真环境由被控电机、功率放大器、传感器、位置/速度/电流控制模型等模块组成。各模块部分同过用户参数的自行配置来实现整体的仿真运行。

7.2.1 被控电机的配置

在 DMCTUN 程序中,在这里我们选定电机控制的应用类型为 BLDC_HP,即为带霍尔位置传感的直流无刷伺服电机。在电机配置的对话框中,所有的电机相关参数都可知直接在编辑框中进行修改,如图 7-2 所示

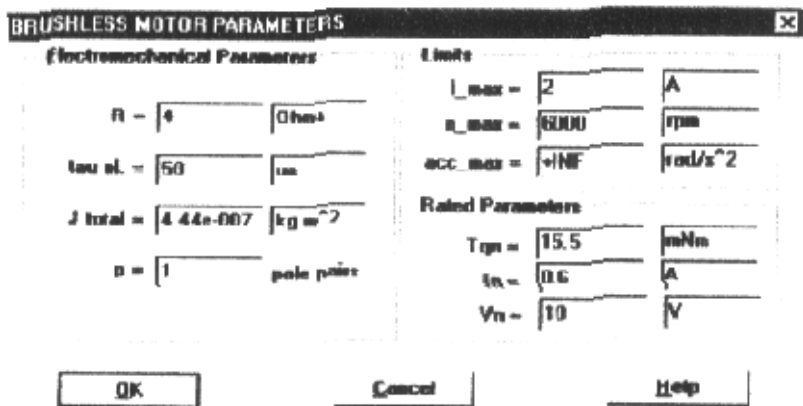


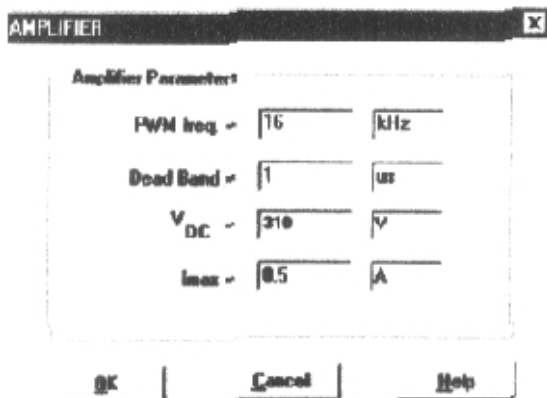
图 7-2 无刷电机配置对话框

图中：R 为电机的绕组电阻； τ_{el} 为电气时间常数； J_{total} 为电机的转动惯量；P 为极对数； T_{qn} 为额定转矩； I_n 为额定电流； V_n 为额定电压；图中还包括了电流、转速，最大加速度的最大值。

7.2.2 功率放大器的配置

功率放大器的配置对话框如图 7-3 所示

放大器的设置比较简单，主要是 PWM 频率，死区时间，承受直流电压和承受电流最大值。



7.2.3 传感器的配置

传感器的图标和配置对话框如图 7-4 所示

其中电流增益因数 A_{crt} 是由电流反馈的输出电压和所测取的电流实际值的比值。即

$$A_{CRT} = \frac{I[Volt]}{I[Ampere]}$$

除了电流增益因数外还有电流比例因数 K_{CT} ，它是电流表达的数字值与电流表达的安培值的比值。它和具体的 DSP 硬件相关。

$$k_{CRT} = \frac{I[\text{number}]}{I[\text{Ampere}]}$$

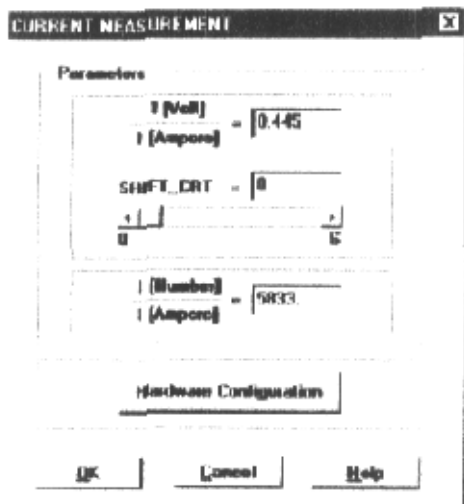
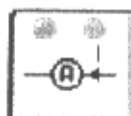


图 7-4 传感器的图标和配置对话框

7.2.4 速度/位置测量对话框的配置

速度位置控制的图标和配置对话框如图 7-5 所示

在本系统中，采用正交编码信号作为电机反馈的速度信号，图中 Nr.lines 是电机转一周所产生的脉冲数，在 10-18384 的范围内。

Speed Sampling Time 为采样的时间间隔。

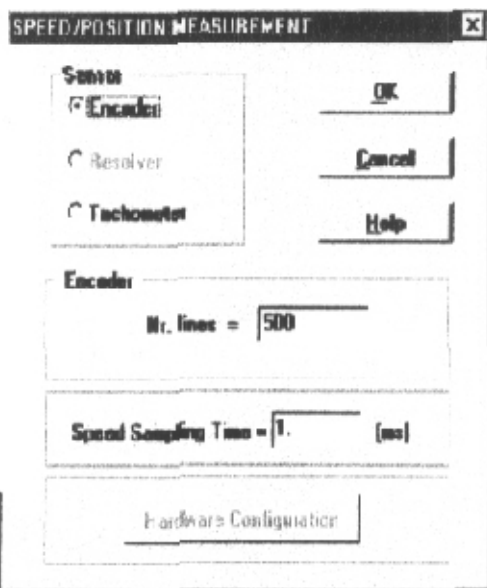


图 7-5 速度位置控制的图标和配置对话框

7.2.5 霍尔传感器的配置

霍尔传感器的图标和配置对话框如图 7-6 所示

霍尔配置本身提供了霍尔器件分布的 12 种情况对应着 12 中不同的霍尔器件的分布, 当然如果被控电机的霍尔分布未知, 选择 none 系统可以

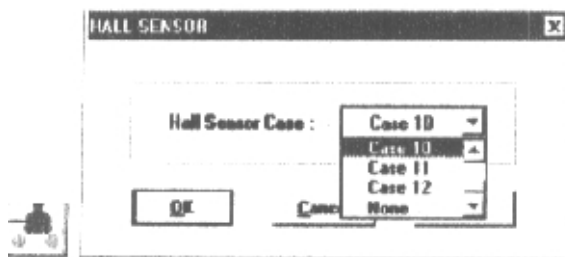


图 7-6 霍尔传感器的图标和配置对话框

自动从 DSP 侧读取信息并选择相应的空间霍尔器件的分布情况。

7.2.6 控制器模型

电流控制器的图标和配置对话框如图 7-7 所示

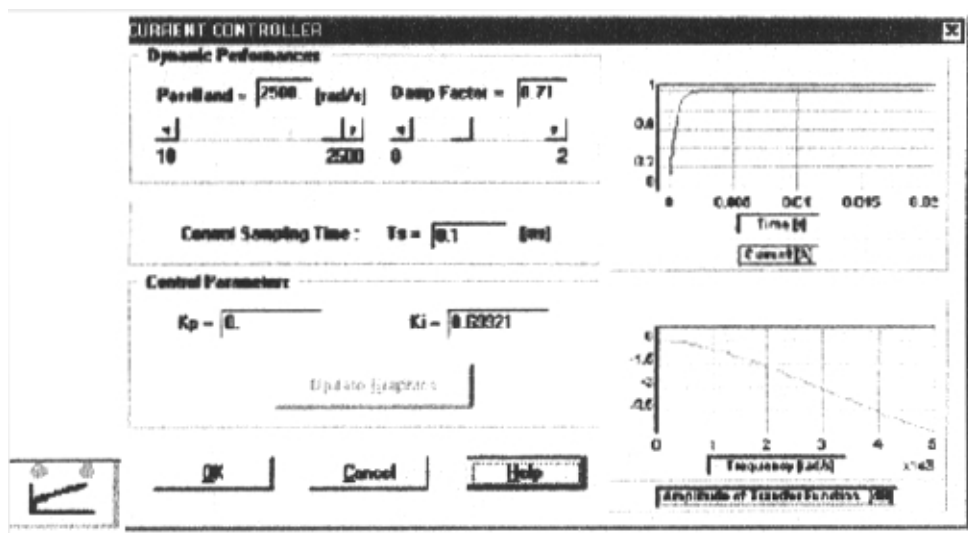


图 7-7 电流控制器的图标和配置对话框

速度控制器的图标和配置对话框如图7—8所示

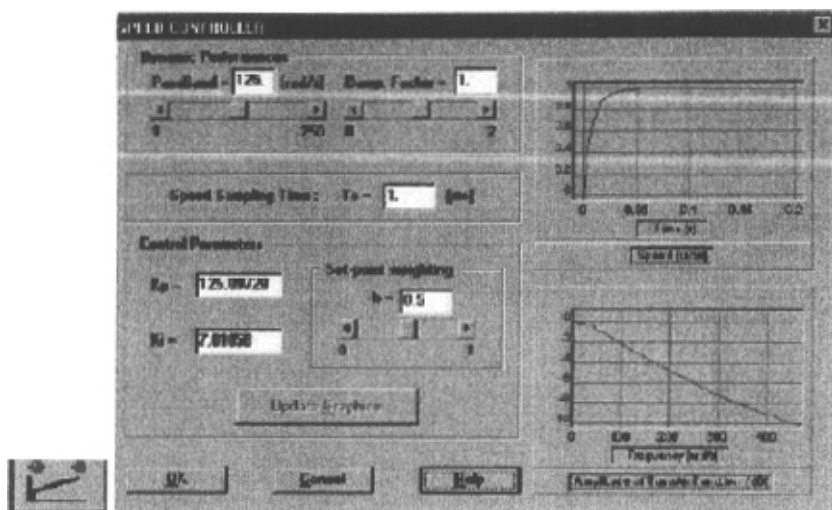


图7—8速度控制器的图标和配置对话框

位置控制器的图标和配置对话框如图 7—9 所示

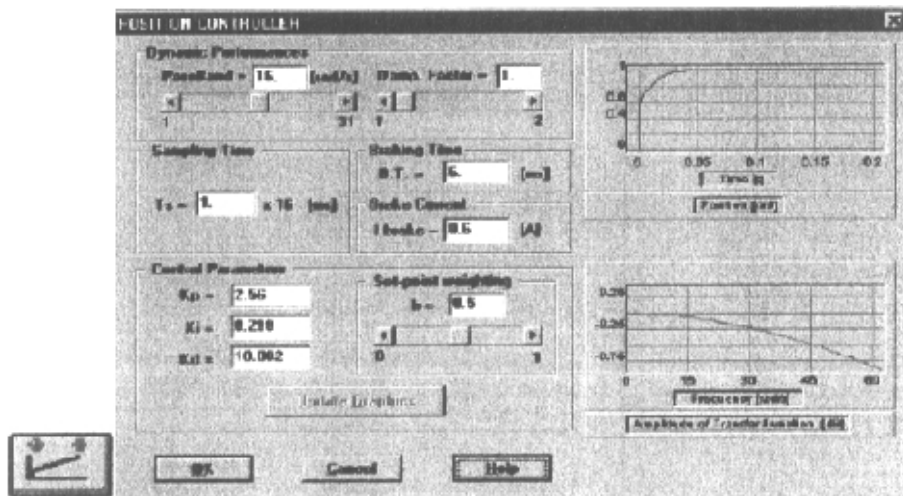


图 7—9 位置控制器的图标和配置对话框

图中，在动态特性部分可以设置相应带宽和相应的阻尼系数。控制部分可以设置指定的控制采样时间和控制参数（主要是比例系数和积分系数）

右边的两个图分别对应了所设置参数的时域和频域的响应情况，从中可以比较直观地看出电流控制器的控制效果。

位置控制器比速度，电流控制器多了微分调节即 K_d 微分系数的设置，改善了系统的动态性能。

控制器的设计基本上是建立在以下几点假设上的:

- 控制器的操作带宽要远小于采样频率。
- 外环的通带要远小于内环的通带。
- 传递函数如: 电流环和速度环 (2 阶)

$$H_{02}(s) = \frac{P(s)}{s^2 + 2\xi\omega_0 s + \omega_0^2}$$

位置环 (3 阶)

$$H_{03}(s) = \frac{P(s)}{(s + \omega_0)(s^2 + 2\xi\omega_0 s + \omega_0^2)}$$

其中 ω_0 为自然振荡频率, ξ 为阻尼系数 (可以根据不同的控制要求选取)。

7.3 直流无刷伺服电机控制的仿真

DMCS-ac Toolbox 仿真软件自带了很多被控电机的模型, 包括 (交流异步电机, 直流电机, 直流无刷伺服电机, 交流同步电机等)。在本系统中, 我们选用 BLDC_EP 模型作为被控电机的模型。

BLDC_EP 是带霍尔位置传感的无刷电机模型。BLDC_EP 的控制主要是实现带位置控制的直流霍尔反馈的无刷电机的驱动控制。

BLDC_EP 的控制原理框图如图 7-10 所示。从图中可以看出, 控制原理图主要是两相电流检测、速度编码检测和霍尔位置检测。被测的相电流根据霍尔位置传感器反馈回来的信息被用来计算电机等效的直流电流, 每个霍尔位置传感器给出了对应 60 电角度的位置信息。

位置, 速度和电流控制器都采用了数字 PI 调节。电流控制器只需一个, 这样就相当于一台直流电机的控制。电压变换模块 (VC) 通过软件实现了各相作用于逆变器的相电压的计算。在控制直流无刷伺服电机时, 在给定位置的情况下只需对逆变器的 4 个功率管作出相应的激励即可。控制系统在霍尔位置传感器每转过 60 电角度会生成特定的指令结构。

位置参考信号是通过 DSPMOT32 程序的参考值发生器产生的。

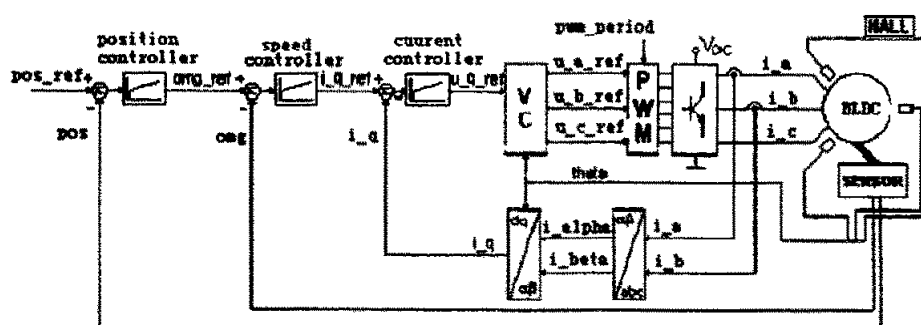


图7—10 BLDC_EP的控制原理框图^[16]

7.4 DSPMOT32 程序

DSPMOT32程序是分析和评估电机运行状态的工作平台，它能给指定的电机应用定义特定的工作环境，这种工作环境可以由各种不同的应用参数，可执行的DSP应用程序，变量地址，初始值，图形设定以及特定的数据传递信息和参考值。它通过给定的要求于DSP侧进行通信，从而达到测试与评估的目的。

值得一提的是DSPMOT32的图形化测试界面为用户提供了非常直观的测试方法，使用户能在计算机中直接读取被控对象的特征数据的取值与波形，大大方便了控制系统的开发与调试。图7—11示出了实际运行电机反馈回来的电流波形。

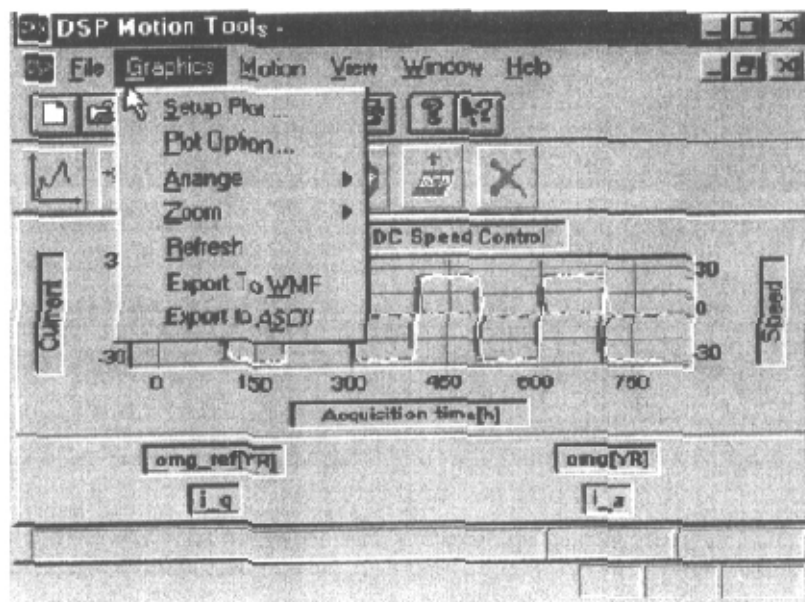


图7—11 实际运行电机反馈回来的电流波形

7.5 控制参数的确定^[16]

根据一定的数学模型, 可以对控制参数的初始值作一定的估计。

7.5.1 电流闭环控制参数的确定

对于电机感性负载, 电路的电压平衡方程式可以看成:

$$u = Ri + L \frac{di}{dt} + E$$

对感性负载的电路的电流控制一般采用PI控制(即比例和积分控制), 控制器的原理图如图7-12所示。

其中 $R(z)$ 即为数字控制中的传递函数, 即:

$$R(z) = K_p + \frac{K_i}{1 - z^{-1}}$$

其中: K_p 为比例因数, K_i 为积分因数。

当回路的时间常数 $\tau_{el} > 4 T_s$ (T_s 为AD采样

时间)时, K_{i_crit} 和 K_{p_crit} 可以用以下关系式确定:

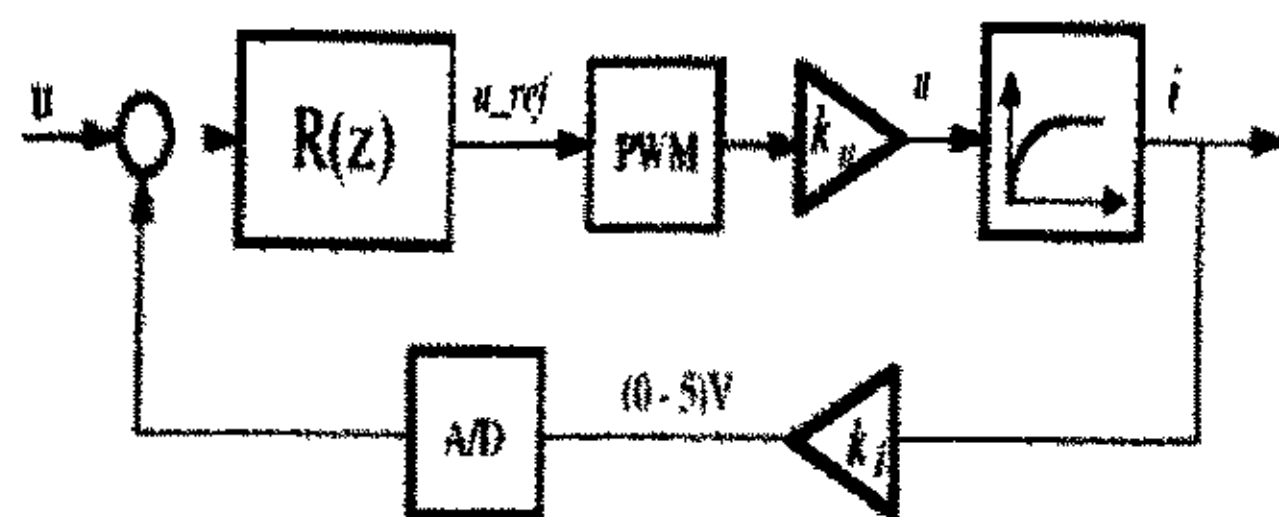


图7-12 电流PI控制器原理图

$$K_{i_crit} = R \omega_0^2 \tau_{el} T_s \frac{k_u}{k_i}$$

$$K_{p_crit} = R(2\xi\omega_0\tau_{el} - 1) \frac{k_u}{k_i}$$

其中 ω_0 和 ξ 分别为自然角频率和阻尼系数, 通过实际的控制要求分别可以确定。在这里为保证系统具有稳定的特性, K_{p_crit} 必须大于等于零。

当 $\tau_{el} > T_s$ 但 $\tau_{el} < 4T_s$, 则电路可能因为高灵敏性而可能变得不稳定, 硬件解决的方法就是加一个附加电感, 这样可以提高电路的时间常数。但附加电感不但提高了系统的成本, 而且在直流无刷伺服电机中还可能产生自感应与过电压的情况。比较好的方法就是在PI调节的控制器中外加一个比例环节, 产生的效果相当于在电路中串联了电感, 从而提高了电路的时间常数。考虑到系统稳定性的边界条件, 可以用以下等式来确定相应的控制参数:

$$K_{i_crit} = (R + K_p) \omega_0 T_s \frac{k_u}{k_i} = R \omega_0 T_s \frac{k_u}{k_i}$$

$$K_{p_crit} = R \left(\frac{\tau_{el}}{T_s 2\sqrt{2}} - 1 \right) \frac{k_u}{k_i}$$

7.5.2 速度闭环控制参数的确定

速度控制器是实现速度调节的控制环节。根据经典的运动理论，速度控制中的机械运动方程式由下式确定：

$$J \frac{d\Omega}{dt} = T - T_r = K\phi I - T_r$$

系统的传递函数可以表示成：

$$H(s) = \frac{\mu}{s}$$

这里 μ 表示加速度因数，其数值为：

$$\mu = \frac{K\phi}{J}$$

通常，在控制系统中，电流闭环作为内环具有比较高的带宽，当电气响应的速度远高于机械响应速度是，可以认为电枢电流是比较好地跟踪指令电流。这就要求速度闭环地带宽必须比电气回路低一个数量级。

通常，选择 ω_0 （速度环） $< 0.1 * \omega_0$ （电流环）

速度必须工作在跟踪模式下，而且要有强抗干扰能力地稳定运行状态。比较好地选择是在及联控制系统中采用积分环节，原理图如图 7-13 所示。

为抑制震荡，将阻尼系数 ξ 设置在 0.7 到 1 之间。

假设一理想的电流控制器，速度控制的传递函数相当于机械系统地传递函数：

$$H(s) = \frac{\mu}{s}$$

采用 PI 控制器：

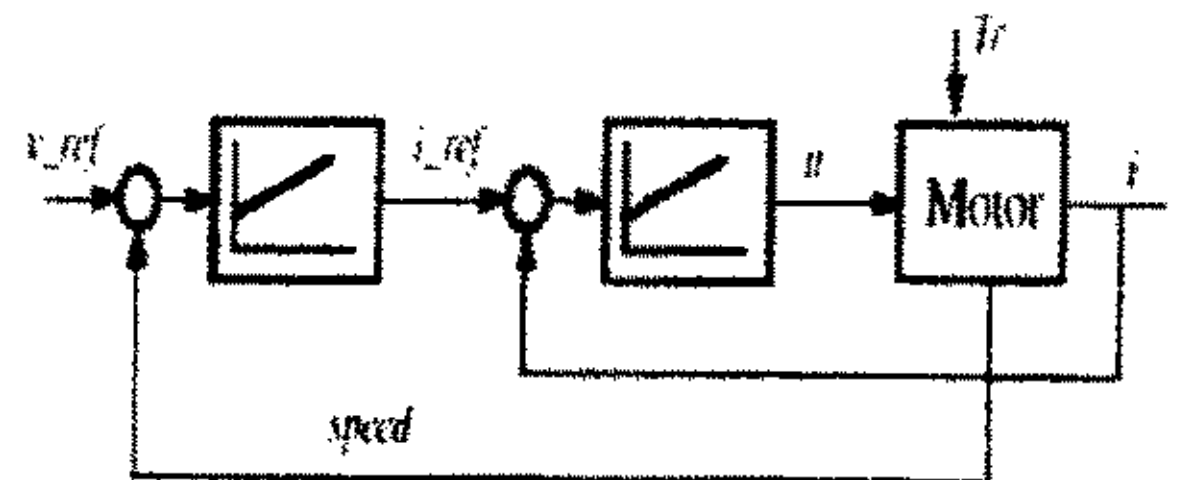


图 7-13 带积分环节地速度控制器

$$R(s) = \frac{sK_{p\Omega} + K_{i\Omega}}{s}$$

则可以得到内环传递函数:

$$H_0(s) = \frac{s\mu K_{p\Omega} + \mu K_{i\Omega}}{s^2 + s\mu K_{p\Omega} + \mu K_{i\Omega}}$$

式中地 $K_{p\Omega}$ 和 $K_{i\Omega}$ 是由以下二型控制函数的极点所确定的

$$H_{02}(s) = \frac{P(s)}{s^2 + 2\xi\omega_0 s + \omega_0^2}$$

$$\begin{cases} K_{p\Omega} = \frac{2\xi\omega_0}{\mu} \\ K_{i\Omega} = \frac{\omega_0^2}{\mu} \end{cases}$$

这里, 自然角频率 ω_0 必须比电流控制器的自然角频率 ω_{0i} 小一个数量级。

将其离散化, 则可以得到速度控制的传递函数为

$$R(z) = K_{p\Omega} + \frac{K_{i\Omega} T_s}{1 - z^{-1}}$$

7.5.3 位置闭环控制参数的确定

位置控制器的输出作为速度控制器的指令输入, 其机械系统运动方程如下:

$$J \frac{d\Omega}{dt} = T = K\phi I \quad \text{或} \quad \frac{d^2\theta}{dt^2} = \mu I$$

这里, μ 表示加速度因数, 其数值为:

$$\mu = \frac{K\phi}{J}$$

控制系统可以由以下传递函数表示:

$$H(s) = \frac{\mu}{s^2}$$

采用 PID 控制:

$$R(s) = \frac{(s^2 K_{d_{pos}} + s K_{p_{pos}} + K_{i_{pos}}) \mu}{s^3}$$

则可以获得其内环传递函数:

$$H_0(s) = \frac{(s^2 K_{d_{pos}} + s K_{p_{pos}} + K_{i_{pos}}) \mu}{s^3 + s^2 K_{d_{pos}} \mu + s K_{p_{pos}} \mu + K_{i_{pos}} \mu}$$

$K_{p_{pos}}$, $K_{i_{pos}}$ 和 $K_{d_{pos}}$ 的确定是有三型控制函数的极点决定。

$$H_{03}(s) = \frac{P(s)}{(s + \omega_0)(s^2 + 2\xi\omega_0 s + \omega_0^2)}$$

$$\begin{cases} K_{i_{pos}} = \frac{\omega_0^3}{\mu} \\ K_{p_{pos}} = (2\xi + 1) \frac{\omega_0^2}{\mu} \\ K_{d_{pos}} = (2\xi + 1) \frac{\omega_0}{\mu} \end{cases}$$

将其离散化, 可以等到位置环的数字控制器:

$$R(z) = K_{p_{pos}} + \frac{K_{i_{pos}} T_s}{1 - z^{-1}} + K_{d_{pos}} \frac{1 - z^{-1}}{T_s}$$

第八章 运行结果测试

本系统经过软件仿真后,通过对 LF2407 的 DSP 的编程和调试,将软件代码下载到自制的 DSP 硬件平台,调试运行,分别作了速度实验,位置实验和加速度实验。实验结果证明,采用 LF2407 控制的无刷直流电机运行平稳,定位准确,启动快,转矩波动小,速度控制灵活。

8.1 速度实验

为了测试本系统控被控电机的速度特性,我们设计了一理想梯形速度曲线作为指令速度曲线,通过硬件仿真,测得电机实际运行速度曲线如图 8-1 所示。

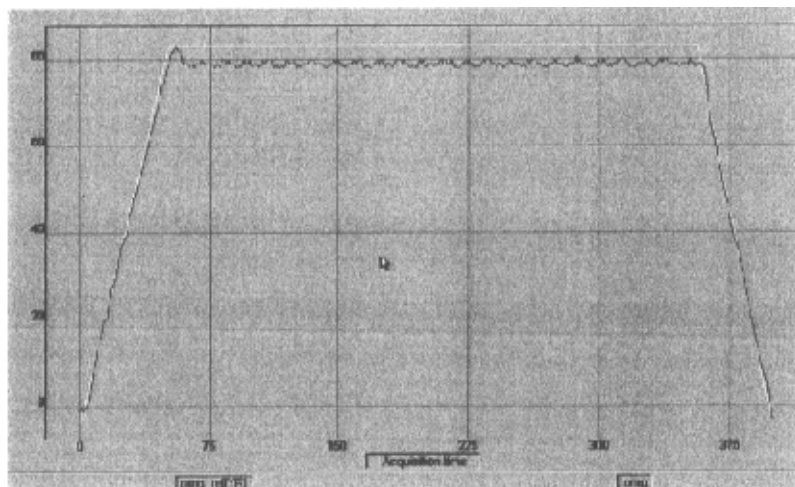


图 8-1 理想速度曲线和实际运行速度曲线

图中,白色曲线为理想参考速度曲线,蓝色曲线为实际运行速度曲线。

从图中可以看出,实际运行的速度曲线基本上能比较好的逼近速度曲线。其中,在升速和降速的阶段,实际速度曲线能非常好地跟踪理想速度曲线;在恒速阶段实际速度曲线变换比较平稳,基本上没有很大的波动,但实际速度曲线在恒速阶段和理想速度曲线有一段小的误差,这是由于所用的 PID 调节参数未设置到最理想。从图中可以看出系统的稳态误差为:

$$e = (82 - 80) / 80 = 2.5\%$$

可见,由于采用了软件仿真设置 PID 控制参数的方法,使得系统的稳态误差能保持

在比较小的范围内,而系统在响应速度上基本上没有延迟,保持了良好的快速响应的特性,而且超调量也能控制在相当小的范围内。

为了减小系统的稳态误差,在保持系统相应时间在一定范围内,可以适当地增大积分因数 K_i 来减小系统的稳态误差,经过细微地调节,测得实际速度曲线和理论速度曲线如图 8-2 所示。

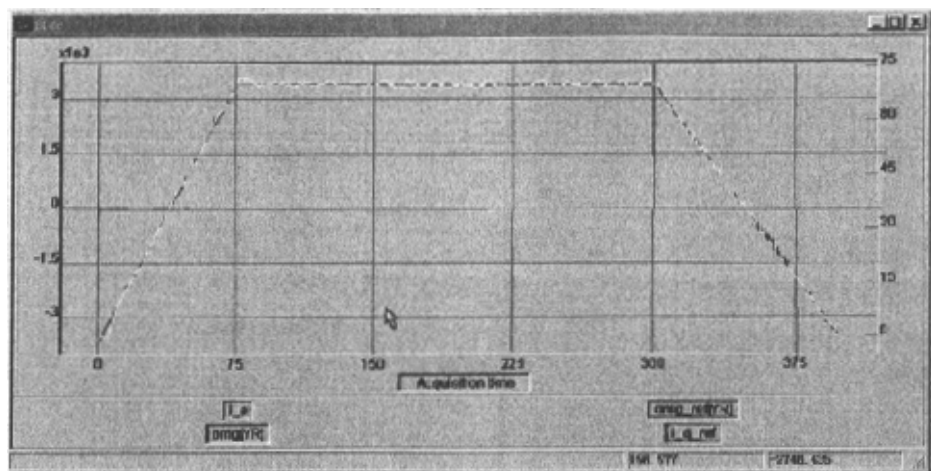


图 8-2 改进后的理论速度曲线和实际速度曲线

其中,红色曲线为理论速度曲线,白色曲线为实际速度独曲线。

从图中可以看出,改进后的实际速度曲线能非常好地逼近理论速度曲线,而且,系统无论在稳态误差,响应速度,最大超调量上都能很好地满足控制的要求。

8.2 位移实验

为测定本系统被控电机的定位特性,我们给定电机指定的位移量,控制电机通过 DSP 内部指定的控制算法,经历升速、恒速和降速三个阶段,实现无刷直流电机的定位。定位实验曲线如图 8-3 所示。

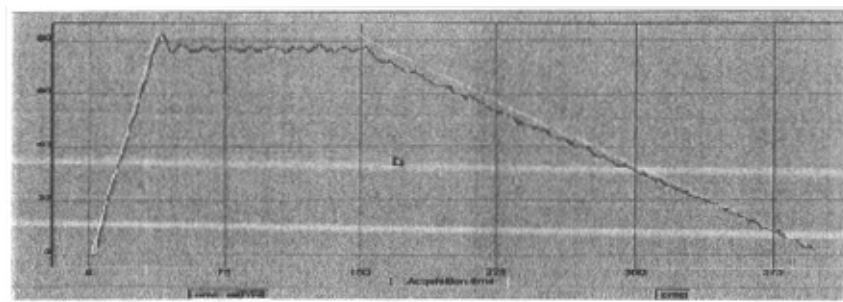


图 8-3 定位实验曲线

图中纵轴表示电机的转速相关量,横轴表示时间。白色曲线表示理想的基于位移控制的速度曲线,蓝色曲线为实际速度曲线。

显而易见,位移控制的关键在于控制好升速、恒速和降速三者之间的取值与时间关系。为了达到高效的定位特性(在最短的时间内达到规定的位移长度),必须对这三阶段的速度进行合理的分配。考虑到电机本身最大加速度,最高转速以及最大减速度的限制,软件定位控制的基本原则是一最大加速度达到最高转速的时间定位升速时间;从最高速度以最大减速度使速度降到零的时间定为降速时间;剩余的位移以最高速度运行的时间定为恒速时间,形成系统定位的参考速度曲线,如图中白线所示。

从图中可以看出,实际位移定位速度曲线基本上能很好地逼近理想定位速度曲线。从定位角度来看,实际定位速度曲线所包围的面积即位移基本上等于理想定位曲线所包围的面积,只是稍稍略小于理想位移。其原因在于稳态误差的存在,通过对 PID 控制参数的细调,可以使稳态误差保持在比较小的范围内。

8.3 电流实验

由于系统的转矩波动和电枢电流的跟踪能力密切相关,所以电枢电流的稳定性对转矩波动有着积极的作用。本实验通过对电机急进,急停和快速反转对电机电流进行测量,并根据指令电流的值检测实际电流的跟踪能力,实验结果如图 8—4 所示:

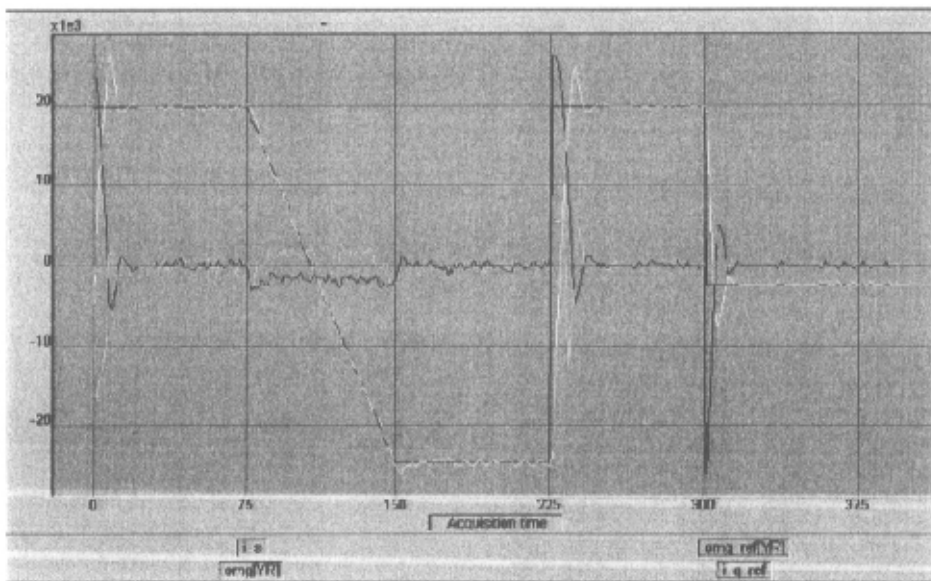


图 8—4 电流跟踪实验结果图

图中,红色曲线为参考速度曲线,白色曲线为实际速度曲线,蓝色曲线为参考电流曲

线，黄色曲线为实际电流曲线。

从图中可以看出，系统在恒速阶段电流有着较好的跟踪能力，无论在响应速度，稳态误差等方面都表现得较好。在速度突变的时候实际电流也能在短时间内恢复对指令电流的跟踪，并表现出一定的超调，其原因在于速度突变而引起的给定电流参考值的突变，但还是能恢复电流跟踪，响应速度较快。

总体来说，本系统无论在速度、定位还是电流都能时间较好的跟踪能力，而且都表现出稳态误差小、响应速度快，超调小的特点，在控制精度上能比较好地达到控制目的。

8.4 实际应用效果

本课题所研究的直流无刷伺服电机的控制系统，已成功地应用到巨型广告打印机和纺织行业的喷绘机的驱动控制系统中。该系统采用 500 线的光电编码盘，电机的定位精度达 $3/(500 \times 4) = 0.15\%$ ，速度误差达 3%，转矩波动小，运行平稳。客户对该系统给予很高的评价，市场反映很好，部分产品销往国外。

该控制系统还应用在为客户开发的雕刻机的驱动部分，成功取代了原来的变频器+单片机的控制模式，降低了生产成本、提高了产品的性价比。

此外，在机床，电控阀门，纺织行业的绣花机的控制系统中，利用该控制系统为客户进行了产品的改装和升级，提高了机器的性能和质量，取得了比较好的效果。

居于 DSP 的直流无刷伺服电机控制技术虽然初露端倪，但已显示出其强大的生命力和广阔的市场前景，本人将一如既往，更深入地开发通用的直流无刷伺服电机的控制器，以 DSP 为核心，将各种复杂的算法移植到 DSP 中。目前，带位置传感器的居于 PID 算法的开发已经完成。无位置传感器的直流无刷伺服电机的控制正在研究中。

第九章 结论

通过对本课题的研究,可以得到如下结论:

1. 本课题提出了一种基于 DSP 的无刷直流电动机的控制系统。系统主要硬件由 TI 公司的 TMS320LF2407 作为本控制系统的中央控制单元,富士的智能功率驱动模块(IPM)作为系统的功率驱动及逆变,被控电机选取带三相霍尔传感器的直流无刷伺服电机作为控制对象来完成此次系统的研究与设计。
2. 在硬件设计方面,系统主要分为控制部分和驱动部分两大块,控制部分为以 DSP 最小系统为核心的一系列周边外围器件,以弱电控制为主;驱动部分主要是功率驱动与逆变,主要是以强电为主。所有的强电与弱电之间皆采用光电耦合来隔离,起到保护的作用。
3. 在软件设计方面,主要是增量式数字 PID 算法。基本思想是移用 DSP 芯片的高速数字运算处理能力,所有的控制算法都在 DSP 内部实现,其输入与输出对用户来说是透明的。其中,电流内环和速度中环采用经典的工业 PI 算法,通过加入抑制积分饱和,使之达到高速实时的控制效果。位置外环采用变结构控制,即是一个 PID 调节器与常值控制相结合的控制,常值选取使得系统的启动性能得到了很好的改善。
4. 本控制系统的控制方法采用三闭环控制,即电流内环,速度中环和位置外环,各控制闭环之间采用了及联的方式连接(上一环的输出作为下一环的输入),通过电流 AD 采样,速度光电正交编码输入,位置霍尔器件的捕捉,来达到系统控制的目的。

实验调试结果证明,本系统在很大控制范围内都能对直流无刷伺服电机进行平稳的控制,而且控制精度也达到了一定高度。在比较大的范围内能多速度进行调节,并能实现较为准确地定位,而且系统对负载转矩的变化有很快的响应,并能在很短的时间内完成转矩变化的平稳过渡,转矩波动能抑制在很小的范围内。

结束语

感谢我的恩师管伯良教授在我研究生阶段,对我的教诲和关怀,在课题的研究阶段和论文的写作过程中,给予我的指导和帮助。

感谢李恩光教授对我学习上的指导。

感谢上海交通大学电院的刘为老师对我课题研究阶段的指导。

感谢上海交通大学的王家敏研究生在我研究生课题阶段的相互交流和帮助。

由于时间的限制,本人只是做了初步的研究,设计了控制电路并且调试成功,达到满意的结果。当然也存在一定问题,总体器件和接口的布局、电路的抗电磁干扰性(EMC)等方面还需要进一步的研究和改善。学海无涯,DSP 在电机控制方面的应用还是一个新领域,各种先进和复杂控制算法的实现还有待进一步研究。

参考文献

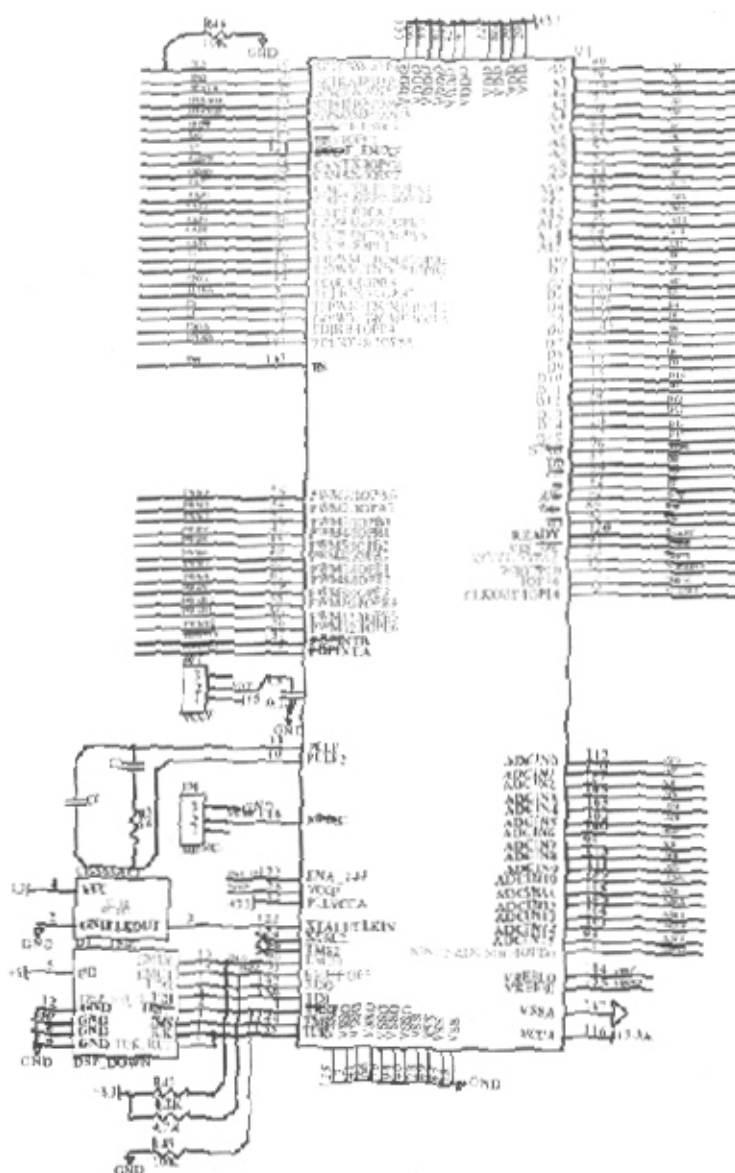
1. 张卫宁,《TMS320C2000 系列 DSPs 原理及应用》, 国防工业出版社, 2002
2. 刘和平等,《TMS320LF240X DSP 结构、原理及应用》, 北京航空航天大学出版社, 20
3. 王强等,《无刷直流电动机系统的数学模型及其局部化线性传递函数》 中国期刊网
1997
4. 张琛,《直流无刷电动机原理及应用》 机械工业出版社 , 1996
5. 魏佳等,《基于 C-dump 变换器的无刷直流电机调速系统的研究》, 南京航空航天大学
学报, 2000
6. Acaunley P P, Ceng, Fiee 《Current measurement in three-phase brushless I
drives》, IEE Proceedings -B, 1993, 140(1): 71~79
7. 《TMS320LF2407 Datasheet》, Texas Instrument
8. 米建国等,《对常规数字 PID 调节器算法的改进》, 中国期刊网, 1998
9. 富士公司,《IGBT-IPM R Series datasheet》, 1997
10. 谭徽等,《单个电流传感器实现的无刷直流电动机相电流检测》, 电机与控制学报
1998
11. 谭弗娃, 金如麟,《大功率电子学和电机控制》 上海交通大学出版社, 1999
12. 陈伯时,《电力拖动自动控制系统》, 机械工业出版社, 1999
13. 李仁定,《电机的微机控制》 机械工业出版社 , 1999
14. 孙力等,《无刷直流力矩电动机系统传函的精确描述》, 中国期刊网, 1997
15. 宋万杰,《CPLD 应用技术》, 西安电子科技大学出版社, 1999
16. Techonsoft,《DMCS—ac Toolbox DSP Motion Solution》, 2001
17. 席军强等,《基于 TPU 模块的无刷直流电机控制器》, 北京理工大学学报, 2001

发表的论文:

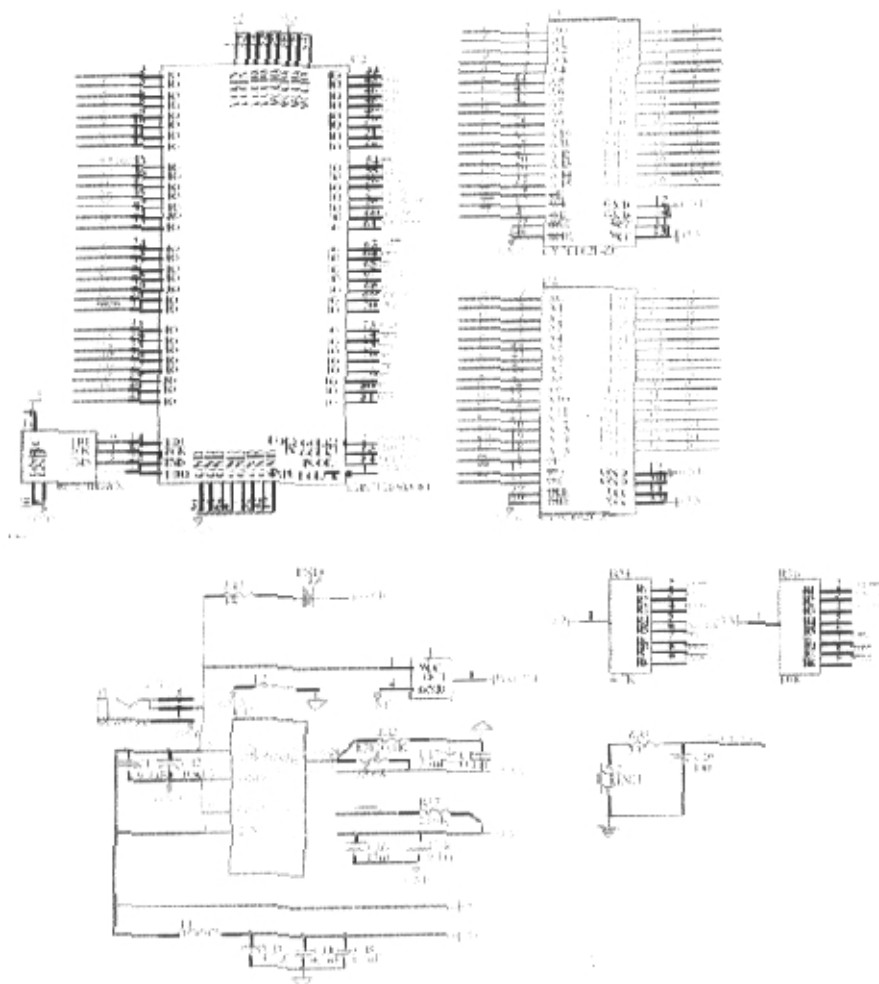
- 1, 彭 军 杜复旦 管伯良 “DSP 控制电机中减少电磁干扰的几项技术 ”
《现代电子技术》 2003.1
- 2, 彭 军 杜复旦 管伯良 “直流无刷伺服电机的 DSP 控制”
《机电一体化》 2003.3

附录 1

TMS320LF2407 最小系统的原理图

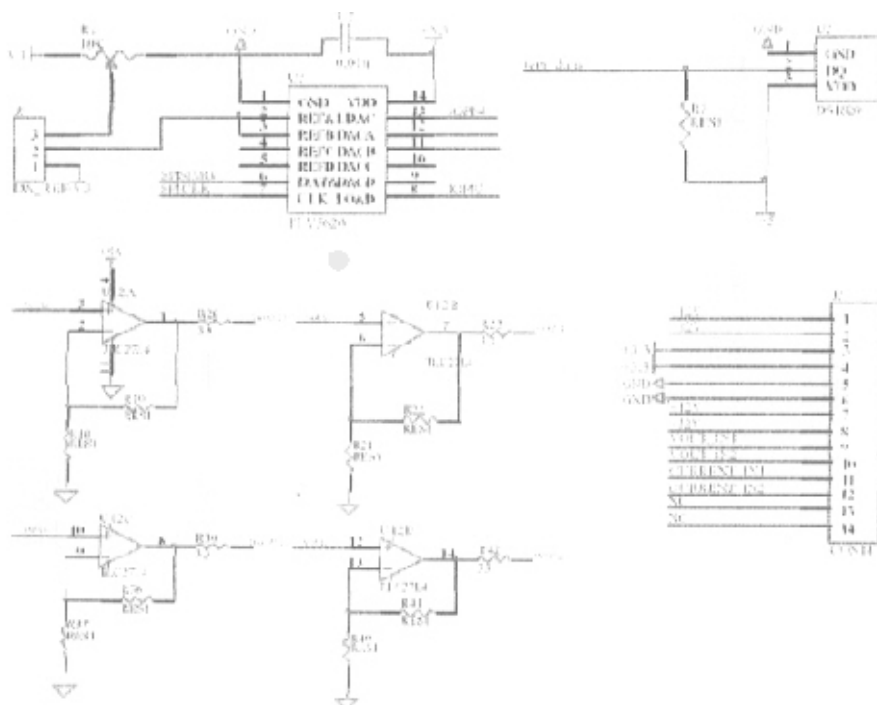


TMS320LF2407 最小系统 (CPLD, POWER, RAM)



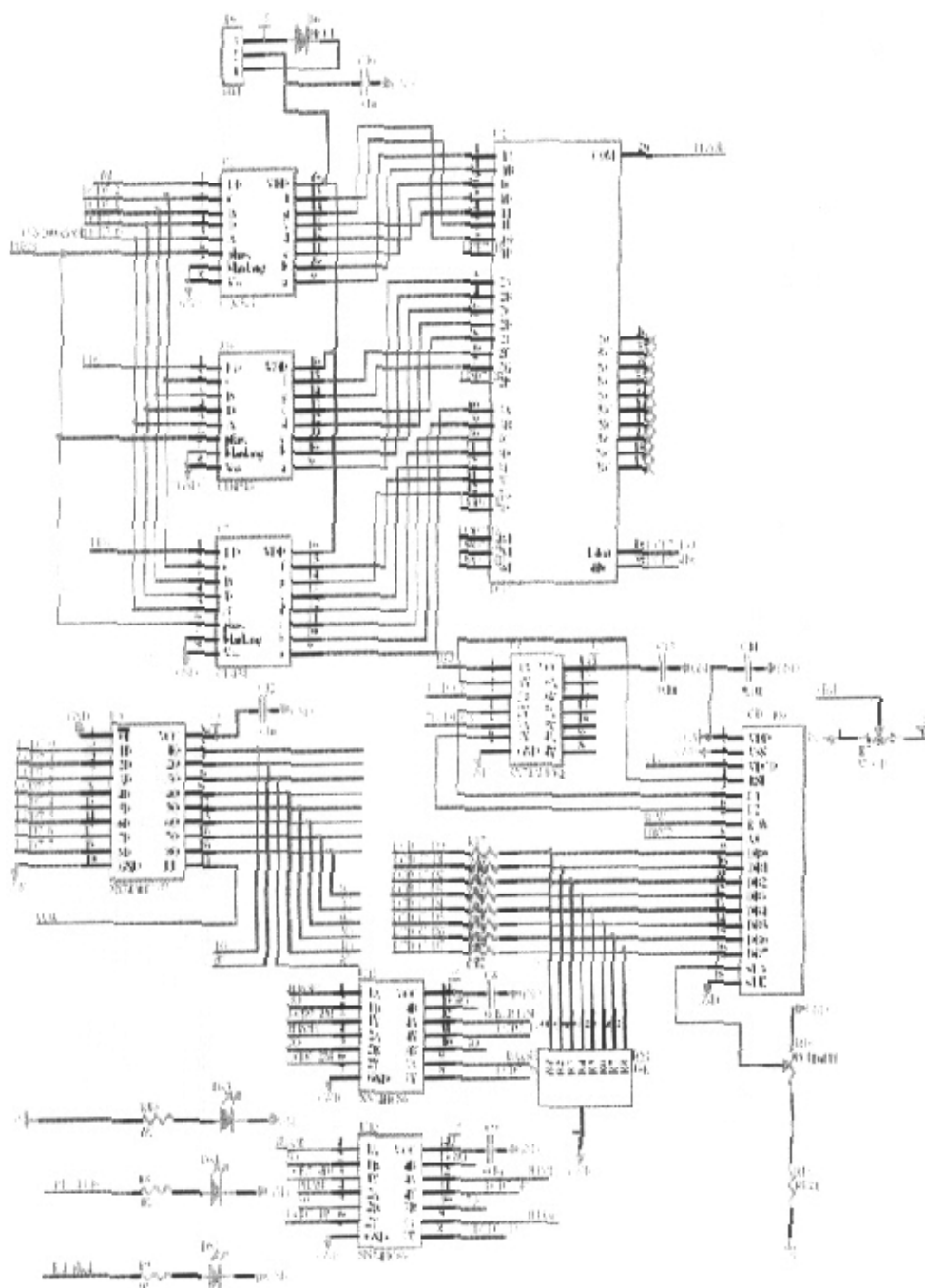
附录 2

模拟信号接口的硬件原理图

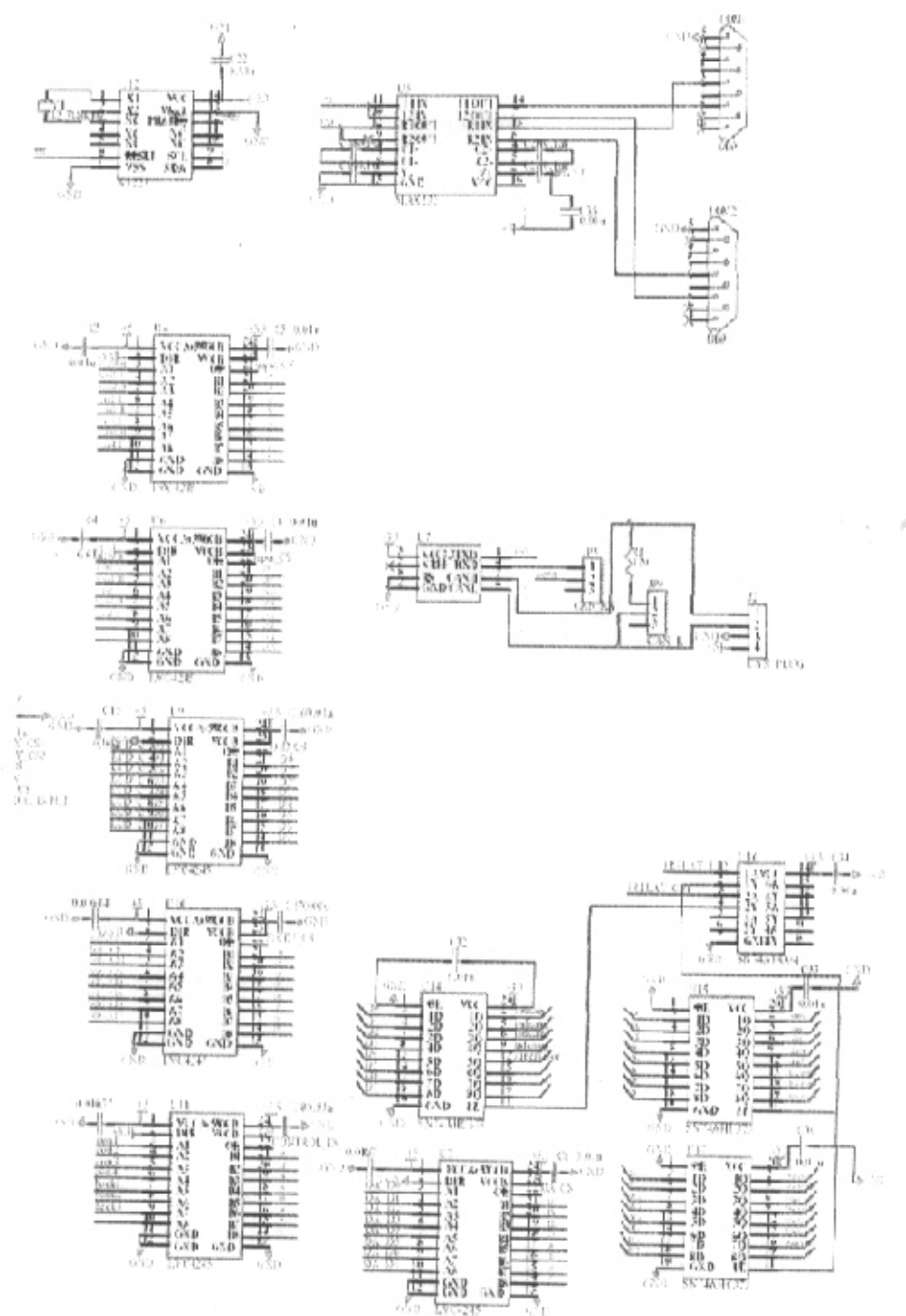


附录 3

LCD 接口原理图

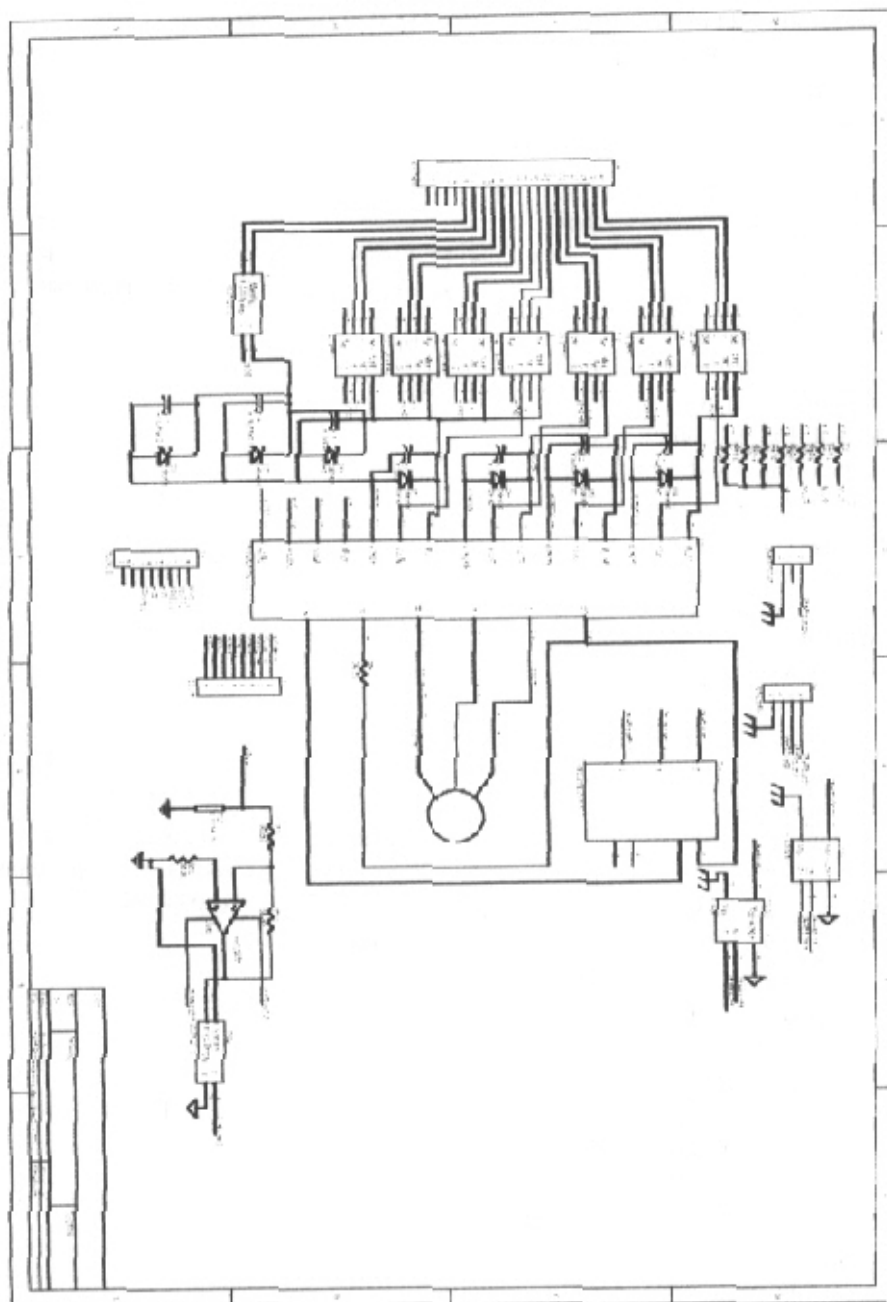


通讯, 缓冲器, 锁存器, 实时时钟芯片接口原理图



附录 4

驱动板电路原理图



附录 5

DSP 侧源程序:

```

/*****
/* NAME:      DEMO1 FOR DSP2707
/* WRITTEN BY:   Peng Jun
/* REVISED TIME: OCT ,21st, 2002
/*              Betal.0
/* THE MOTOR RUN IN SEQUENCE:FORWARD, STOP, REWARD, STOP,
/* THEN BACK TO FORWARD AGAIN
/* IF tpvalue = 30 , The XF LED will flash every 0.96s
*****/
#include "extsymb.h"
unsigned int tpvalue  = 60 ;
unsigned int tpcounter = 0;
unsigned int xfsignal = 0;
unsigned int fordrewd = 0;
unsigned int CMPRREF = 300;
unsigned int temp;
void nega();
void posi();
void brakestop();
void main()
{
    asm("      CLRC   XF");
    asm("      CLRC   CNF");      /*SET B0 AS DATA MEMORY FROM 200-300H*/
    asm("      CLRC   OVM");      /*Reset overflow mode*/
    asm("      SETC   SXM");      /*Reset sign extension mode*/
    asm("      SETC   INTM");      /*Set global interrupt mask */
    /*Disable watchdog (Vccp=5v)& watchdog counter reset P6-12*/
    RTICR  = 0x07;    /*disable the RTI INT*/
    WD_CNTL = 0x2f; /*enable WD 1s*/
    WD_KEY  = 0x55;
    WD_KEY  = 0xaa;
    /*initialize WDT registers*/
    WD_CNTL = 0x6f ;    /*clear WDFLAG,Disable WDT if VCCP =5v, set WDT*/
    /*initialize WSGR MEMORY wait state = 0*/
    WSGR = 0x0f;
    /*Set up CLKOUT to be SYSCLK P6-6*/
    SYSCR = 0x40c0;
    asm("      LDP     #0");
    asm("      SPLK    #1ch, 04h");
    asm("      LACC    _SYSSR");
    asm("      AND     #069FFH");

```

```

    asm("        SACL    _SYSSR");
/*Set up PLL clockin=20MHZ, cpuclock=20MHZ, sysclk=10MHZ*/
    CKCR1 = 0x60      ;
    CKCR0 = 0x41      ; /*bit0=1  f(sysclk)=f(cpuclock)/2*/
/*I/O setting pll-11*/
    OCRA = 0x0ff; /*IOPB0 - 7enable, IOPA disable*/
    OCRB = 0x0fff0; /*IOPB0=IOPB1=0.XF, BIO*/
    PBDATDIR = 0x0ff; /*all are input, and is high*/
    CMPR1 = CMPRREF;
    CMPR2 = CMPRREF;
/*Input Clock Prescaler = 1/2, CPU clock frequency = 20M*/
    T1PER = 5000; /*PEROID = 128/20*5 MS*/
    T1CNT = 0;
    GPTCON= 0x0;
    T1CON = 0x9700; /*continuous up counting mode, prescaler = 1/128*/
    T1CON = 0x9740;

/*IMR, IFR SET
;ENABLE DSP INT1, 2, 4, 5
;EVIMRA, B, C, EVIFRA, B, C SET
;ENABLE T1PINT , SCI RECEIVER INT*/

    IMR = 0x02; /*enable int2*/
    IMR = 0x02;
    IFR = 0x0ffff;
    EVIFRA= 0x0ffff;
    EVIMRA= 0x0080; /*enable T1 peroid int*/
    NMICR = 0x0004;
    COMCON= 0x0207;
    COMCON= 0x8207;

    asm("    CLRC    INTM");
    while(1)
    {
        if (xfsignal == 0)  asm("    clrc  XF");
        else                asm("    setc  XF");
    }
}

void c_int1()
{
}
/*EVERY 0.032MS ENTRY ONCE*/
void c_int2()
{

```



```

int evivra;
evivra = EVIVRA;
switch(evivra)
{
    case 0x27: /*TIPINT*/
        tpcounter++;
        temp= PBDATDIR;
        temp= temp & 0x60;
        if (temp != 0 ) fordrewd = 1;
        /*if ((temp = PBDATDIR & 0x0c0) == 0) fordrewd = 1;*/
        else
        {
            if ((temp=PBDATDIR&0x01) == 0x00)        fordrewd = 1;
            else
            {
                if ((temp=PBDATDIR&0x02) == 0)  fordrewd = 0;
                else if ((temp=PBDATDIR&0x04) == 0) fordrewd = 2;
            }
        }
        switch(fordrewd)
        {
            case 0x0:    ACTR  = 0x3D; /*1032*//*posi()*/;
                        break;
            case 0x1:    ACTR = 0x33;
                        break;
            case 0x2:    ACTR = 0xd3;
                        break;
            case 0x3:    ACTR = 0x33;
                        break;
        }
        CMPR1 = CMPRREF;
        CMPR2 = CMPRREF;

        if (tpcounter == tpvalue) /*every 1ms process the cmpr*/
        {
            tpcounter = 0;          /*clear the counter*/
            WD_KEY  = 0x55 ;        /*kick watchdog*/
            WD_KEY  = 0xaa ;
            xfsignal = !xfsignal;

            /*
            fordrewd++;
            if (fordrewd >= 4)  fordrewd = 0;
            */
        }
}

```

```

}

void c_int3()
{
}
void c_int4()
{
}
void c_int5()
{
}
void c_int6()
{
}
void posi()
{
    ACTR = 0x3D; /*1032*/
    /*ACTR = 0x00e; 1031*/
    COMCON= 0x0207;
    COMCON= 0x8207;
}

void nega()
{
    ACTR = 0xd3;
    /* ACTR = 0x00e0;*/
    COMCON= 0x0207;
    COMCON= 0x8207;
}

void brakestop()
{
    ACTR = 0x033;
    COMCON= 0x0207;
    COMCON= 0x8207;
    /* stpsignal =1;*/
}

;*****
; File Name : sensor.asm
; Target System : F240
; Description : Sensored speed control
; Position sensor: Three Hall effects sensors
; handled by Capture Unit
; Current measurement: Line shunt resistor
; handled by ADC module
; PWM generator: full Compare Unit at 20kHz
; ISR: Capture and End Of Conversion

```

```
; Date      : Feb 2002 (Ref. BPRA055/064)
;*****
    .include "D2407app.h"
;-----
;Speed regulator coefficients setting
;-----
Kp .set    015
Ki .set    004
;-----
; Variable definitions
;-----
    .bss    temp,1
    .bss    GPRO,1
    .bss    pbl_svo_r,1
    .bss    mSEC,1
    .bss    SEC,1
;    .bss    RPT_NUM,1
    .bss    CAPT,1      ; Capture indication
    .bss    COMP,1      ; Duty cycle
    .bss    COMP_BUF,1 ; PWM Test $$$$ $$$$
    .bss    SPEED_VAL,1
    .bss    SPEED_REF,1 ; Speed reference
    .bss    SPEED_COUNT,1 ; Speed loop frequency
    .bss    stack,6     ; Context save stack
;PID_REG .set    300H
;cap_k .set    PID_REG+0 ; EQ T2CNT
;cap_k1 .set    PID_REG+1
;Te .set    PID_REG+2
;spd_k .set    PID_REG+3
;spd_k1 .set    PID_REG+4
;-----
; Reset & interrupt vectors
;-----
    .sect "vectors"
RSVECT:
    B    _c_int0      ; Reset Vector
INT1:      ; SYSIVR (701Eh)
    B    PHANTOM      ; INT1_ISR
INT2:      ; EVIVRA (7432h)
    B    PHANTOM      ; INT2_ISR
INT3:      ; EVIVRB (7433h)
    B    PHANTOM      ; INT3_ISR
INT4:
    B    CAPINT
INT5:      ; SYSIVR (701Eh)
```

```
B PHANTOM ; INT5_ISR
INT6:
B ADCINT
ANLY_INT: ; 41 000Eh MON_EINTR
B PHANTOM ; User-defined software interrupt
INT8: ; 0010h
B PHANTOM ; User-defined software interrupt
INT9: ; 0012h
B PHANTOM ; User-defined software interrupt
INT10: ; 0014h
B PHANTOM ; User-defined software interrupt
INT11: ; 0016h
B PHANTOM ; User-defined software interrupt
INT12: ; 0018h
B PHANTOM ; User-defined software interrupt
INT13: ; 001Ah
B PHANTOM ; User-defined software interrupt
INT14: ; 001Ch
B PHANTOM ; User-defined software interrupt
INT15: ; 001Eh
B PHANTOM ; User-defined software interrupt
INT16: ; 0020h
B PHANTOM ; User-defined software interrupt
TRAP_INT: ; 0022h
B TRAP_INT ; TRAP instruction vector
NMI_INT: ; 3 0024h
B NMI_INT ; Nonmaskable interrupt
INT19: ; 0026h
B PHANTOM ; User-defined software interrupt
INT20: ; 0028h
B PHANTOM ; User-defined software interrupt
INT21: ; 002ah
B PHANTOM ; User-defined software interrupt
INT22: ; 002ch
B PHANTOM ; User-defined software interrupt
INT23: ; 002Eh
B PHANTOM ; User-defined software interrupt
INT24: ; 0030h
B PHANTOM ; User-defined software interrupt
INT25: ; 0032h
B PHANTOM ; User-defined software interrupt
INT26: ; 0034h
B PHANTOM ; User-defined software interrupt
INT27: ; 0036h
B PHANTOM ; User-defined software interrupt
```



```

INT28:      ;    0038h
    B  PHANTOM    ; User-defined software interrupt
INT29:      ;    003Ah
    B  PHANTOM    ; User-defined software interrupt
INT30:      ;    003Ch
    B  PHANTOM    ; User-defined software interrupt
INT31:      ;    003Eh
    B  PHANTOM    ; User-defined software interrupt

    .text
    .global  _c_int0
_c_int0:
    SETC    CNF
    CLRC    OVM    ; Reset overflow mode
    SETC    SXM    ; Reset sign extension mode
    CLRC    XF
    SETC    INTM    ; Set global interrupt mask

    MAR    *,AR2    ; Init speed variables stack
    LAR    AR2,#0300h ; 300H
    SPLK    #0,++
    SPLK    #0,++
    SPLK    #0FFFFH,++
    SPLK    #029H,++
    SPLK    #029H,++
    SPLK    #0,*
    LAR    AR2,#0300h
    LAR    ARI,#stack ; Init context save stack pointer

    LDP    #0E0H
    SPLK    #028Fh,SYSCR ;scsr1
    SPLK    #02ch,SYSSR ;scsr2

; Disable watchdog (Vccp=5v)& watchdog counter reset p6-12
    LDP    #00E0h
    SPLK    #6Fh,WDCR
    SPLK    #05555h,WDKEY
    SPLK    #0AAAAh,WDKEY

; I/O setting pli-11
;2407
    LDP    #00E1H
    SPLK    #0FFFFH,OCRA ;USE IOPB4-7,DISABLE IOPA,PB0-3,USE CAP1-3,PWM1-6
    SPLK    #0FFFFH,OCRB ;DISABLE IOPC

```

```

;TEMP TEST SET IOPC2 AS IO
    SPLK    #0FFFBH, OCRB
;END TEMP TEST SET IOPC2 AS IO

    SPLK    #0ffffH, OCRC
    SPLK    #00H, PADATDIR
    SPLK    #00H, PBDATDIR
    ;temp define IOPC2 as output test
    SPLK    #0400H, PCDATDIR ;IOPC. 2 AS OUTPUT HIGH
    ;end test
;END 2407

; Clear EV control registers

    LDP #0E8h
    SPLK    #0000h, TICON ; no timer enable
    SPLK    #0000h, T1PR
    SPLK    #0000h, T1CNT
    SPLK    #0000h, T1CMP
    SPLK    #0000h, T2CON
    SPLK    #0000h, T2PR
    SPLK    #0000h, T2CNT
    SPLK    #0000h, T2CMP
    SPLK    #0000h, COMCON
    SPLK    #0000h, ACTR
    SPLK    #0000h, CMPR1
    SPLK    #0000h, CMPR2
    SPLK    #0000h, CMPR3
    SPLK    #0000h, CAPCON ; no capture
    SPLK    #00ffh, CAPFIFO ; Clear CAPFIFO
    LACC    FIF01
    LACC    FIF02

; GP Timer 2 Setting start with GPT1 presc 128
    LDP    #T2PR>>7

    SPLK    #0ffffh, T2PR ; Limit Max
    SPLK    #0000h, T2CNT
    SPLK    #17C0h, T2CON ; 17C0
; Stop, Cus, x/128, UT1, Eto, Inr, 00, Dtc, Uop

; PWM Unit setting
    SPLK    #0500, T1PR ; 3.2mS = 500/156250
    SPLK    #0000h, T1CNT

```

```

    SPLK    #0FFFh, ACTR    ; All Forced high
    SPLK    #0508h, DBTCN ; All disa, X/4
    SPLK    #0000h, CMPR1
    SPLK    #0000h, CMPR2
    SPLK    #0287h, COMCON ; 0287 Clear COMCON
; Dfc, WT10, Dsv, WT10, FCOE, SCOD, GPt1, WTy0, WTy0, SPM
    SPLK    #8287h, COMCON
    SPLK    #2900h, T1CON
; SPLK #2800h, T1CON ; 2800
; Stop, Cud, x/1, UoT, Dto, Inr, 00, Dtc, Uop
    SPLK    #2940h, T1CON
; SPLK #2840h, T1CON ; 2840
; Stop, Cud, x/1, UoT, Eto, Inr, 00, Dtc, Uop
    SPLK    #0100h, GPTCON ; 0100
; Cd, Cd, Cd, NA, NA, NA, T1Sp, TP0d, T3PF1, T2PF1, T1PF1
    SPLK    #0140h, GPTCON
; Capture Unit Setting
    SPLK    #0b0fch, CAPCON ; B0FC
; CAPRES, Ecap1&2, Ecap3, Dcap4, SGPt2, SGPt2, Na, Dbe1&2&3, Nd
    SPLK    #00ffh, CAPFIFO ; 00FF
; Clear bit 15-8

; Core Mask Setting
    LDP #0
    LACC    #028h
    SACL    IMR ; INT6 & INT4 is masked.
    LACC    IFR
    SACL    IFR

    LDP #0E8h
    LACC    EVIFRA ; EVIFRx Reset
    SACL    EVIFRA
    LACC    EVIFRB
    SACL    EVIFRB
    LACC    EVIFRC
    SACL    EVIFRC
    SPLK    #0, EVIMRA ; EVIVRax INT Disable
    SPLK    #0, EVIMRB ; EVIVRBx INT Disable
    SPLK    #7, EVIMRC ; Cap1-3 INT Enable

; 2407
    LDP    #0E1H
    SPLK    #0100000000000000b, ADCCTRL1 ; Reset ADC module
    NOP
    SPLK    #3AB0H, ADCCTRL1

```

```

    SPLK    #0100000000000000B, ADCCTRL2
    SPLK    #0772H, ADCCTRL2
    SPLK    #00H,    MAXCONV
    SPLK    #1H,    CHSELSEQ1
;END2407

    CLRC    INTM        ; All INT Enable
    LDP #0        ; Variables init
    SPLK    #0, pbl_svo_r
    SPLK    #0, mSEC
    SPLK    #0, SEC
    SPLK    #0500, SPEED_REF    ; 40H
    SPLK    #0500, COMP
    SPLK    #0500, COMP_BUF    ; $$$$ $$$$
    SPLK    #0625, SPEED_COUNT
    SPLK    #0000H, CAPT
    SPLK    #0, temp

*****
*   STARTING PROCEDURE   *
*****
; Disable CAPT Input p2-75
    LDP #0E8h
    SPLK    #0000h, CAPCON ; Capture reset

; Set I/O unit as I
    LDP #00E1h
;2407
    SPLK    #0FC7h, OCRA; CAP1, 2, 3 AS IOPA
    SPLK    #0000h, PADATDIR    ; IOPAx Input
    LACC    PADATDIR        ; READ IOPAX
;END 2407
;   SPLK    #0FF00h, OCRB    ; IOPCx I/O Setting
;   SPLK    #0000h, PCDATDIR    ; IOPCx Input

; WHICH SEQUENCE?
;   LACC    PCDATDIR    ; Read IOPCx
;2407
    AND     #0038H
    LDP #0
    SACL    CAPT                ; Save temp !!!!
    LACC    CAPT, 13
    SACH    CAPT                ; Save into !!!!
    LDP     #0E1H
    SPLK    #0FFFH, OCRA
;END 2407

```



```

        ;AND    #0070H      ; Mask Cap1&2&3
;   LDP #0
;   SACL  CAPT              ; Save temp !!!!
;   LACC  CAPT, 12
;   SACH  CAPT              ; Save into !!!!

; Disable I/O Unit
;   LDP #0E1h
;   SPLK  #0FF70h, OCRB    ; Set Cap1&2&3

; Capture Unit Setting
    LDP #0E8H
    SPLK  #0b0fch, CAPCON
;CAPRES, Ecap1&2, Ecap3, Dcap4, SGpt2, SGpt2, Na, Dbe1&2&3, Nd
    SPLK  #00ffh, CAPFIFO  ; Clear bit 15-8

MAINLOOP:
    LDP #00E0h
    SPLK  #05555h, WDKEY
    SPLK  #0AAAAh, WDKEY
    B     MAINLOOP

*****
*           END OF STARTING PROCEDURE       *
*****
FAULT_CLEAR:      ; Init the pre-driver
    LDP #0E1h
    LACC  PBDATDIR  ; Read IOPBx
    AND #010h      ; Mask IOPB4
    BZ   FAULT_CLEAR  ; If ACC = 0
    SPLK  #0210h, PBDATDIR  ; 2820
    ; not shut down clearing fault

MAIN:              ; Polling loop
    LDP #0E0h
    KICK_DOG
F_PB4_CLR:        ; Init the pre-driver
    LDP #0E1H
    LACC  PBDATDIR  ; Read IOPBx
    AND #010h      ; Mask IOPB4
    BZ   MAIN      ; If ACC = 0

    LDP #0E1H
    SPLK  #0210h, PBDATDIR  ; 2820

```

```

; not shut down clearing fault
    LDP #0h
    LACL    pbl_svo_r
    BCND    MAIN_10, EQ
    LDP #0E1H
    SPLK    #0212h, PBDATDIR    ; 2800    IOPB1_out data_hi

    SETC    XF
    CALL    SEC_DELAY
    CALL    SEC_DELAY
    CLRC    XF
    LDP     #PBDATDIR >> 7
    SPLK    #0210h, PBDATDIR    ; 2800    IOPB1_out data_low
    LDP     #0
    SPLK    #0, pbl_svo_r
MAIN_10:
    LDP #0E0H
    NOP
    NOP
    B    MAIN

*****
ADCINT:
; save status registers
    MAR *,    ARI
    MAR      *+    ; skip one position
    SST      #1, *+    ; save ST1
    SST      #0, *+    ; save ST0
    SACH     *+    ; save acc high
    SACL     *      ; save acc low

    LDP #0
    LACC     temp
    ADD      #1
    SACL     temp

    LDP     #0E1H
;    SPLK    #0100000000000000B, ADCCTRL2
;    NOP
    SPLK    #0772H, ADCCTRL2

;2407
;temp define IOPC2 as output test
    LDP     #0E1H
    BIT     PCDATDIR, 13 ;IOPC2
    
```

```

        BCND    PCSET0,      TC
PCSET1:
        LACC    #4H
        OR      PCDATDIR
        B        PCSETEND
PCSET0:
        LACC    #0FFFBH
        AND     PCDATDIR
PCSETEND:
        SACL    PCDATDIR    ;PB4 AS OUTPUT low
        ;end test

        LDP     #0EOH
        LACC    PIVR
;end 2407

        CLRC    SXM
        LDP     #RESULT0>>7
        LACC    RESULT0, 10

        B       ADCRET

        LDP #0
        SACH    SPEED_VAL
        LACL    SPEED_VAL
        SUB #716
        BLEZ    RD_ADC_10
        LACC    #495
        B       RD_ADC_20
RD_ADC_10:
        LACC    SPEED_VAL
        SUB #216
        BGEZ    RD_ADC_20
        LACC    #0
RD_ADC_20:
        SACL    SPEED_VAL
        LACC    #500
        SUB SPEED_VAL
        SACL    SPEED_VAL

        LACL    SPEED_COUNT    ; Load PID loop Spd_value
        SUB #625                ; 1250
        BNZ NO_SPEED_REG    ; If Var_Spd_Done then
        CALL    SPEED_REG
NO_SPEED_REG:

```

```

    LACC  SPEED_COUNT    ; SPEED_COUNT ++
    ADD #1
    SACL  SPEED_COUNT

; output PWM
    CALL  SEQUENCE

ADCRET:
; restore status registers
    MAR   *, AR1        ; make stack pointer active
    LACL  *-             ; Restore Acc low
    ADDH  *-             ; Restore Acc high
    LST   #0, *-         ; load ST0
    LST   #1, *-         ; load ST1
    CLRC  INTM
    RET

*****
SEQUENCE:
    LDP #0

    LACL  SEC
    SUB #1
    SACL  SEC

    LACC  COMP_BUF      ; $$$$ $$$$
    SACL  COMP          ; $$$$ $$$$

    LACC  CAPT
    SUB #1
    SFL   ; ACC * 2
    ADD #CAPT_DETER
    BACC

CAPT_DETER:
    B    FALLING3
    B    FALLING1
    B    RISING2
    B    FALLING2
    B    RISING1

RISING3:
                                ; S5
    LACC  COMP
    LDP #0E8h
    SPLK  #0DF3h, ACTR
; SVP, , CMP6Fh, CMP5A1, CMP4Fh, CMP3Fh, CMP2F1, CMP1Fh

```



```
SACL  CMPR3
SACL  CMPR1
SPLK  #0000, CMPR2
B  END
```

```
FALLING3:                                ; S2
    LACC  COMP
    LDP #0E8h
    SPLK  #03FDh, ACTR
; SVP, , CMP6F1, CMP5Fh, CMP4Fh, CMP3Fh, CMP2Fh, CMP1A1
    SACL  CMPR1
    SACL  CMPR3
    SPLK  #0000, CMPR2
B  END
```

```
RISING2:                                ; S3
    LACC  COMP
    LDP #0E8h
    SPLK  #03DFh, ACTR
; SVP, , CMP6F1, CMP5Fh, CMP4Fh, CMP3A1, CMP2Fh, CMP1Fh
    SACL  CMPR2
    SACL  CMPR3
    SPLK  #0000, CMPR1
B  END
```

```
FALLING2:                                ; S6
    LACC  COMP
    LDP #0E8h
    SPLK  #0D3Fh, ACTR
; SVP, , CMP6Fh, CMP5A1, CMP4F1, CMP3Fh, CMP2Fh, CMP1Fh
    SACL  CMPR3
    SACL  CMPR2
    SPLK  #00000, CMPR1
B  END
```

```
RISING1:                                ; S1
    LACC  COMP
    LDP #0E8h
    SPLK  #0F3Dh, ACTR
; SVP, , CMP6Fh, CMP5Fh, CMP4F1, CMP3Fh, CMP2Fh, CMP1A1
    SACL  CMPR1
    SACL  CMPR2
    SPLK  #0000, CMPR3
B  END
```

```
FALLING1:                                ; S4
```

```

    LACC    COMP
    LDP #0E8h
    SPLK    #0FD3h, ACTR
; SVP, , CMP6Fh, CMP5Fh, CMP4Fh, CMP3A1, CMP2F1, CMP1Fh
    SACL    CMPR2
    SACL    CMPR1
    SPLK    #00000, CMPR3
END:
    RET
*****
SPEED_REG:
; Speed and speed error calculation
    B TEMPADD
    LDP     #0
    SPLK    #0, SPEED_COUNT

    LDP #0
    CLRC    SXM

    LDP #PBDATDIR >> 7
    LACC    PBDATDIR ; detect IOPB1 (SLO)
    AND #0002h      ; Mask IOPB1
    BCND    SPD_PROC_S0, EQ      ; If ACC = 0
    LDP #0
    SPLK    #500, SPEED_REF
    B SPD_PROC_END
SPD_PROC_S0:
    LDP #PBDATDIR >> 7
    LACC    PBDATDIR ; detect IOPB4 (OC)
    AND #0010h      ; Mask IOPB4
    BCND    SPD_PROC_00, NEQ     ; If ACC <> 0
    LDP #0
    SPLK    #500, SPEED_REF
    B SPD_PROC_END
SPD_PROC_00:
    LDP #XINT2CR >> 7
    LACL    XINT2CR      ; detect top position
    AND #0040H
    BCND    SPD_PROC_10, NEQ     ; If ACC <> 0 (XINT2 = 1)
    LACL    XINT1CR      ; detect home position
    AND #0040H
    BCND    SPD_PROC_20, NEQ     ; If ACC <> 0 (XINT1 = 1)
    LDP #0
    SPLK    #500, SPEED_REF
    SPLK    #1, pb1_svo_r

```

```

    B    SPD_PROC_END
SPD_PROC_10:
    LACL  XINT1CR      ; detect home position
    AND #0040H
    BCND  SPD_PROC_20, NEQ  ; ACC <> 0 (XINT1 = 1)
SPD_PROC_11:
    LDP #0
    SPLK  #400, SPEED_VAL  ; $$$$ $$$$
SPD_PROC_20:
    LDP #0
    LACL  SPEED_VAL
SPD_PROC_30:
    SUB SPEED_REF

    BLZ SPD_DEC_PROC  ; <
    BGZ SPD_ACC_PROC  ; >
    B    SPD_PROC_END  ; =
SPD_DEC_PROC:
    LDP  #0
    LACL SPEED_REF
    SUB #3
    SACL SPEED_REF
    B    SPD_PROC_END
SPD_ACC_PROC:
    LACL SPEED_REF
    ADD #1
    SACL SPEED_REF
SPD_PROC_END:
    LDP  #0
    LACL SPEED_REF
    SACL COMP_BUF

; TEMP ADD
TEMPADD:
    LDP  #0
    LACC #300
    SACL SPEED_REF
    SACL COMP_BUF
; ENDED TEMP ADD
; Restore
    RET
*****
CAPINT:
    MAR *, AR1
    MAR  *+    ; skip one position

```

```

    SST    #1,*+    ; save ST1
    SST    #0,*+    ; save ST0
    SACH   *+       ; save acc high
    SACL   *        ; save acc low

; Capture interrupt handling
;2407
    LDP #0E0h
    LACC   PIRQR1    ;
    AND    #1070H    ;enable ADCINT and CAPINT1,2,3
    AND    #00H      ;clear the pending flag
    SACL   PIRQR1
    LACC   PIVR
    AND    #00H
    SACL   PIVR
;end 2407

; Set I/O unit as I
    LDP #00E1h
;2407
    SPLK   #0FC7h,OCRA ; CAP1,2,3 AS IOPA
    SPLK   #0000h,PADATDIR ; IOPAx Input
    LACC   PADATDIR    ;READ IOPAX
;END 2407
;   SPLK   #0FF00h,OCRB ; IOPCx I/O Setting
;   SPLK   #0000h,PCDATDIR ; IOPCx Input

; WHICH SEQUENCE?
;   LACC   PCDATDIR    ; Read IOPCx

;2407
    AND    #0038H
    LDP #0
    SACL   CAPT        ; Save temp !!!!
    LACC   CAPT,13
    SACH   CAPT        ; Save into !!!!

; Commutation
    CALL   SEQUENCE    ; Modify Commu_States
    LDP    #0E1H
    SPLK   #0FFFH,OCRA
;END 2407

; Capture Unit Setting
    LDP #0E8H

```

```

    SPLK    #0b0fch, CAPCON    ; 0B0FC
;CAPRES, Ecap1&2, Ecap3, Dcap4, SGPt2, SGPt2, Na, Dbel1&2&3, Nd
    SPLK    #00ffh, CAPFIFO    ; Clear bit 15-8

```

```

    MAR     *, AR1             ; make stack pointer active
    LACL    *-                 ; Restore Acc low
    ADDH    *-                 ; Restore Acc high
    LST     #0, *-             ; load ST0
    LST     #1, *-             ; load ST1
    CLRC    INTM
    RET

```

```

*****

```

```

;=====

```

```

; I S R - PHANTOM

```

```

; Description: Dummy ISR, used to trap spurious interrupts.

```

```

;=====

```

```

PHANTOM:

```

```

    B    PHANTOM

```

```

*****

```

```

;=====

```

```

; Routine Name:    mS_DELAY

```

```

; Description:    Produces a multiple of 1mS delays using the RPT

```

```

; instruction. The Delay produced is based on the

```

```

; value loaded in mSEC (i.e. Delay = mSEC x 1mS).

```

```

; Indirect addressing is used to count the number

```

```

; of times the delay loop is repeated.

```

```

; Calling Convention:

```

```

; Variables    on Entry    on Exit

```

```

;=====

```

```

; DP    XX    0x0000

```

```

; ARP    XX    AR4

```

```

; ACC    XX    XX

```

```

; mSEC    value in 1 mS    un-touched

```

```

;=====

```

```

SEC_DELAY:

```

```

    LDP #0h    ; DP-->0000h-007Fh

```

```

    SPLK    #25000, SEC ; 25000 * 0.00004 = 1 SEC

```

```

SEC_LOOP:

```

```

    KICK_DOG

```

```

    LDP #0

```

```

    LACL    SEC

```

```

    BCND    SEC_LOOP, NEQ

```

```

    RET    ; Return from DELAY SR

```

```

;=====

```