

中国科学技术大学

专业硕士学位论文



一种无位置传感器电机控制算法 设计及优化

作者姓名：谈伟
专业类别：电子信息
专业领域：计算机技术
校内导师：赵云波
实践导师：吴俊
完成时间：二〇二四年五月十日

University of Science and Technology of China

A dissertation for master's degree



Design and optimization of a position sensorless motor control algorithm

Author: Tan Wei

Speciality: Electronic Information

Major: Computer Technology

Supervisors: Prof. Zhao Yunbo

Instructor: Prof. Wu Jun

Finished time: May 10th, 2024

中国科学技术大学学位论文原创性声明

本人声明所呈交的学位论文,是本人在导师指导下进行研究工作所取得的成果。除已特别加以标注和致谢的地方外,论文中不包含任何他人已经发表或撰写过的研究成果。与我一同工作的同志对本研究所做的贡献均已在论文中作了明确的说明。

作者签名: 谈伟

签字日期: 2024.5.10

中国科学技术大学学位论文授权使用声明

作为申请学位的条件之一,学位论文著作权拥有者授权中国科学技术大学拥有学位论文的部分使用权,即:学校有权按有关规定向国家有关部门或机构送交论文的复印件和电子版,允许论文被查阅和借阅,可以将学位论文编入《中国学位论文全文数据库》等有关数据库进行检索,可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。本人提交的电子文档的内容和纸质论文的内容相一致。

控阅的学位论文在解除后也遵守此规定。

公开 控阅 (____年)

作者签名: 谈伟

导师签名: 李波

签字日期: 2024.5.10

签字日期: 2024.5.10

摘 要

永磁同步电机具有结构简单、功率因数高、体积小、调速范围广的优点，已经成为电动汽车首选的驱动电机。为了能够高精度控制永磁同步电机，通常采用机械式传感器，但是这类传感器增加了控制系统的成本，在复杂环境下使用也受到限制，因此，无传感器的控制技术逐渐成为研究热点。本文主要针对全速域无感电机控制展开研究并进行优化。

首先针对电机控制中大量的三角函数运算，本文设计了一种多区间二次函数拟合三角函数的方法，设计三角函数硬件指令，实现在 5 个系统时钟周期内完成相应的指令计算并输出结果。

其次，说明了电机矢量控制的流程，封装算法中的相关函数接口，针对无感控制无法获得转子位置和速度信息的问题，本文采用非线性磁链观测器的方案，并针对传统方案中使用反正切计算导致的电机运行高频抖振问题，提出了基于锁相环的位置解算方法，使电机在中高速下运行更加平稳。

再次，对于非线性磁链观测器在低速下效果较差的问题，采用脉振高频注入法，并对传统方案中的解调函数、滤波器、PI 控制器进行优化，将转子位置误差信号中的高频信号从二次谐波提升为四次谐波，使解算出的转子位置误差更小。

最后，将设计的非线性磁链观测器和脉振高频注入相结合，采用混合控制策略，对权重函数进行优化，使电机能在全速域范围内运行。

关键词：硬件指令；无位置传感器；磁链观测器；脉振高频注入；混合控制

ABSTRACT

With the advantages of simple structure, high power factor, small size, and wide speed range, permanent magnet synchronous motor has become the preferred drive motor for electric vehicles. In order to be able to control the permanent magnet synchronous motor with high accuracy, mechanical sensors are usually used, but these sensors increase the cost of the control system, and their use in complex environments is also limited, so sensorless control technology has gradually become a research hotspot. This dissertation mainly focuses on the full-speed domain sensorless motor control to carry out research and optimization.

Firstly, for the large number of trigonometric operations in motor control, this dissertation designs a method of fitting trigonometric functions with multi-interval quadratic functions, and designs trigonometric hardware instructions to realize that the corresponding instruction calculations are completed and the results are outputted within 5 system clock cycles.

Secondly, the process of motor vector control is explained, and the related function interface in the encapsulation algorithm is described. For the problem that rotor position and speed information cannot be obtained by inductorless control, this dissertation adopts the scheme of nonlinear magnetic chain observer, and for the problem of high-frequency jitter of motor operation caused by the use of the inverse tangent computation in the traditional scheme, the phase-locked-loop based position solving method is proposed, which makes the motor run more smoothly at medium and high speeds.

Again, for the problem that the nonlinear magnetic chain observer is less effective at low speeds, the pulsed vibration high-frequency injection method is used, and the demodulation function, filter, and PI controller in the traditional scheme are optimized to enhance the high-frequency signal in the rotor position error signal from the second harmonic to the fourth harmonic, so as to make the solved rotor position error smaller.

Finally, the designed nonlinear magnetic chain observer and pulsation high frequency injection are combined with a hybrid control strategy to optimize the weight function so that the motor can operate in the full speed domain range.

Key Words: Hardware instructions; sensorless; flux observer; pulse vibration high-frequency injection; compound control

目 录

第 1 章 绪论.....	1
1.1 研究背景及意义.....	1
1.2 国内外研究现状.....	2
1.2.1 DSP 三角函数指令研究现状	2
1.2.2 无位置传感器控制研究现状.....	2
1.3 本文的主要工作及创新.....	4
第 2 章 三角函数硬件指令设计与优化	7
2.1 电机控制中的三角函数.....	7
2.1.1 三角函数的使用场景.....	7
2.1.2 当前三角函数计算的不足.....	8
2.2 指令设计路线.....	9
2.2.1 常用算法与优缺点.....	9
2.2.2 多区间二次函数拟合.....	10
2.3 软件仿真设计.....	11
2.3.1 软件仿真流程.....	11
2.3.2 定浮点转换模块.....	12
2.3.3 两级乘累加模块.....	15
2.3.4 正余弦指令仿真与优化.....	16
2.3.5 反正切指令仿真与优化.....	17
2.4 软件仿真结果及比较.....	17
2.5 本章小结.....	22
第 3 章 矢量控制及中高速域无感控制方法	23
3.1 电机控制数学模型.....	23
3.1.1 三相静止坐标系下的数学模型.....	24
3.1.2 两相静止坐标系下的数学模型.....	25
3.1.3 两相旋转坐标系下的数学模型.....	25

3.2 磁场定向控制策略与库函数封装.....	26
3.2.1 坐标变换.....	27
3.2.2 空间矢量脉宽调制算法.....	28
3.2.3 矢量控制函数库封装.....	30
3.2.4 磁场定向控制流程.....	34
3.3 中高速无感控制方案.....	35
3.3.1 磁链观测器原理.....	36
3.3.2 基于 PLL 的磁链观测器设计	39
3.3.3 中高速无感控制流程.....	39
3.4 本章小结.....	41
第 4 章 低速域高频注入控制方案	42
4.1 非线性饱和凸极性原理.....	42
4.2 脉振高频电压信号注入.....	43
4.2.1 脉振高频电压信号注入原理.....	43
4.2.2 转子位置提取.....	46
4.2.3 转子极性判断.....	50
4.2.4 脉振高频注入控制流程.....	51
4.3 本章小结.....	51
第 5 章 电机全速域混合控制方案及系统搭建	52
5.1 全速域无感控制方案.....	52
5.1.1 全速域混合控制策略.....	52
5.1.2 改进后的混合控制策略.....	54
5.2 硬件系统搭建.....	55
5.2.1 主控芯片最小系统.....	55
5.2.2 电源电路.....	56
5.2.3 驱动电路.....	57
5.2.4 采样电路.....	57
5.2.5 保护电路.....	60
5.3 软件程序设计.....	60
5.3.2 主程序设计.....	60

5.3.3 中断子程序设计.....	61
5.4 实验与分析.....	62
5.4.1 电机运行验证.....	62
5.4.2 电机控制性能验证.....	70
5.5 本章小结.....	73
第 6 章 总结与展望	74
6.1 工作总结.....	74
6.2 未来展望.....	74
参 考 文 献.....	76
致 谢.....	81
附录 1 浮点转定点仿真代码	82
附录 2 定点转浮点仿真代码	83
附录 3 乘累加仿真代码	84
附录 4 sin 指令区间划分与二次函数系数.....	86
附录 5 atan 指令区间划分与二次函数系数.....	87
附录 6 sin 指令移位区间与移位值.....	88
附录 7 atan 指令移位区间与移位值.....	89

第 1 章 绪论

1.1 研究背景及意义

近年来,能源与环境问题逐渐成为社会发展中的重大影响因素,在此背景下,新能源汽车产业大力发展,电机在其中发挥着电能转换的作用,电机控制方案也随之成为研究热点。常见的电机包括直流无刷电机(Brushless DC Motor, BLDC)、永磁同步电机(Permanent Magnet Synchronos Motor, PMSM)、步进电机(Stepping Motor, SM)等,其中直流无刷电机与永磁同步电机都可以使用矢量控制来对转子的转速进行控制。永磁同步电机通过永磁体产生磁场,其反电动势为正弦信号,控制效果也更加优秀,具有功率密度高、控制效率高等优点,而我国丰富的稀土金属资源也促进了永磁同步电机的发展^[1]。

在电机控制中,常见的控制方法有直接转矩控制^[2-3](Direct torque control, DTC)和磁场定向控制(Field Oriented Control, FOC),其中磁场定向控制具有动态响应快、转矩大的优点,应用场景更为广泛。使用 FOC 对电机进行控制,需要测定转子的位置和速度信息,一般场景下选择利用霍尔传感器^[4]、光电编码器^[5]、旋转变压器^[6]等位置传感器获取,实现对电机的闭环控制。但是在电机上加装位置传感器会使其体积变大,控制结构变复杂,不利于维护,造成系统整体稳定性降低、抗干扰能力变差。为了解决位置传感器带来的不利影响,利用反电动势或磁链观测转子位置信息的无感电机控制技术开始出现与发展。

无位置传感器控制放弃传感器在电机中的使用,它通过对电机内部的电压信号和电流信号进行分析,根据电机状态空间方程重构转子磁链、速度。该方法大大提高了系统的稳定性,降低了成本,实现对电机在高温、高湿度场景下的正常控制。

在半导体行业中,数字信号处理器(Digital Signal Processor, DSP)逐渐成为信息技术的重要组成部分,它们在通信、雷达、语音处理等多个领域发挥着重要的作用。其中,在电机控制应用领域,数字信号处理器(DSP)的使用场景更加多样,不仅用于实施高效的电机控制算法,还对电机的各种状态信号进行实时的监控与分析,通过对矢量控制方法的使用,DSP 能够更加精确的控制调整电机转子速度、扭矩、位置,从而进一步提升整个控制系统的性能和效率。DSP 在电机控制中面临的较大挑战是中间存在的大量浮点运算,其计算精度问题是浮点中的一个核心挑战^[7]。对于矢量控制,需要进行大量的三角函数计算,这些计算如果采用软件算法或查表等方式,在精度与效率上都有一定的不足,设计一个体积小、低功耗

且性能均衡的三角函数浮点运算模块对于 DSP 的优化处理至关重要。

基于以上两点, 提供一套高效稳定的无位置传感器电机控制方案, 并在 DSP 上针对三角函数计算性能的不足, 提出专门的解决方案, 能够使用户在工业控制领域更加快速完成对电机的控制。

1.2 国内外研究现状

1.2.1 DSP 三角函数指令研究现状

在计算机科学领域中, 三角函数的计算在工程、图形设计等多个方向发挥着关键作用。为了提高这些运算的效率, 相关研究人员致力于在硬件层面寻找更高效的实现方法, 旨在缩短计算时间的同时得到较高精度的结果。

传统的处理器大多为通用型处理器, 对于三角函数计算通常调用软件代码, 使得计算速度缓慢, 而采用多核处理器, 虽然能在一定程度上提高计算性能, 但需要额外考虑 Cache 一致性和数据安全性等问题, 使得该方法对计算速度并无明显提升。

能否高效计算三角函数值并降低运行功耗对于应用开发至关重要, 为实现此目标, Volder 提出了一种经典的 CORDIC^[8](Coordinate Rotation Digital Computer) 算法, 该方法通过加法和位移的迭代操作代替复杂乘除法运算, 以旋转的方式逼近三角函数值, 适用于正余弦、正切、反正切等函数的计算。CORDIC 算法以其硬件结构简单的优点, 适用于资源受限的嵌入式系统。在此基础上, Debaprasad^[9]提出了一种基于 CORDIC 的统一架构, 进一步降低硬件资源消耗。Gao^[10]等人设计了一种轻量的 RISC-V 微处理器, 优化 CORDIC 算法, 实现低延迟、高精度的三角函数硬件加速单元, 平均减少了 33% 的精度误差和 46% 的迭代周期。Vinh^[11]等人提出了基 4-CORDIC 算法, 通过仅三次迭代完成计算, 实现快速返回结果且误差小于 0.005。Yousif 在 FPGA 中通过修正多项式曲线拟合正余弦函数, 从 LUT(Look Up Table)和切片占用角度改进, 分别实现了 95% 和 96% 的面积优化, 但是增加了硬件的执行时间。Flynn 提出了一种对专门硬件指令(正弦、余弦)的优化加速方法, 实现了对三角函数的矢量计算, 该方法利用单指令流多数据流(Single Instruction Multiple Data, SIMD)^[12]的矢量化来达到数据级别的并行计算, 同时利用 Estrin^[13]提出的多项式近似计算, 在计算效率上, 比最先进的快速方法还高出 1.35 倍, 同时对于计算结果精度并无明显的降低。

1.2.2 无位置传感器控制研究现状

随着现代控制理论和电子电力技术的不断发展,主控芯片的数据处理能力和计算速度不断提高,国内外学者针对电机的无感控制方案也取得新的研究成果。

当前,根据电机运行转速范围的不同,研究方向大致可以分为适用于低转速下的控制算法和适用于中高转速下的控制算法。电机工作原理以及转子转速大小的不同,采用的无感控制策略也各不相同。

(1) 中高转速无感方案研究

当电机运行在中高转速时,一般是利用采集到的定子电压和电流信息,对转子速度进行计算。这种控制方法使得系统在稳定性、准确性、动态响应方面,具有良好的性能。常见的针对电机转速中高转速无感控制方案有:滑模观测器(Sliding Mode Observe, SMO)、模型参考自适应控制系统(Model Reference Adaptive Control System, MRACS)、状态观测器(State Observer, SO)、扩展卡尔曼滤波器(Extended Kalman Filter, EKF)。

滑模观测器本质上是对电机的电流误差进行重构,将电流误差设置为滑模面,使估计值不断跟随真实值,通过结构开关的变化,不断进行高频率的切换,使系统能够在一定时间内稳定到一个平面,该方法的鲁棒性好、受外界因素干扰小。忽略某些影响因素构建的数学模型,在滑模观测器的控制下也具有良好的动态响应能力^[14]。但是该方法最大的问题就是电机控制过程中可能存在抖振现象,对此祝新阳^[15]等人提出,将滑模控制中的滑模面函数用 sigmoid 函数替代,同时为进一步减少估算环节中的浮点计算,使用全阶滑模观测器代替原本控制中的低通滤波器。不过 sigmoid 函数对于减少系统抖振问题的能力有限,朱喜华^[16]等人则采用双曲正切函数代替原本的开关函数,进一步减少系统抖振,并且该方法实现简单,同时能够降低系统的相位延迟。

模型参考自适应控制系统基于自适应理论,核心在于其能够根据系统的实际运行情况进行动态调整,该方法由三个关键部分组成:预设的控制模型,可调参数的模型和依据系统结构差异调整自适应率的机制。这种自适应控制系统本质上是一种参数辨识的方法,在稳定性理论基础上,为保证估计精度的准确性,需不断调整参考模型的选择。有研究^[17]提出根据转子估计位置与实际位置之间的偏差来确定转子的位置信息和速度信息,基于 $d-q$ 坐标系的数学模型,估计了待求解信息,并在不同转速和负载下进行测试验证,结果表明该方法相较于传统的滑模观测器,能够更加精确地估计转子转速,且大幅提升系统的动态响应速度。

状态观测器又叫状态重构器,其核心是对动态系统进行重构。它通过分析外部变量,包括输入变量和输出变量的实测值,推导出状态变量的估计值,从而实现对系统内部状态的有效控制和预测。状态观测器通过对电机数学模型进行估计和重构,将电压和电流信号作为输入量,位置估计量为输出量,通过引入校正环

节,不断调整估计值使其接近真实值,同时利用校正机制对估算结果的误差进行补偿,有效减少系统的延迟,提高控制精度^[18]。

扩展卡尔曼滤波是对卡尔曼滤波在非线性系统中应用的扩展,本质上建立在最小方差估计的基础上,是一种具有明显随机性和非线性特征的递推滤波方法。卡尔曼滤波的一个优势是其实时性好,可以在系统运行的同时,一边对数据进行收集,一边进行相应的数学计算。Gaona^[19]等人将改进后的卡尔曼滤波和自适应算法相结合,在一定程度上提升了系统的动态响应能力,同时明显降低了电机运行过程中的噪声。Rui^[20]等人则利用卡尔曼滤波来对电机转速进行辨识,该方法不仅减少了估算过程中的计算量,还避免了因为滤波器的引入而造成的相位延迟问题,从而进一步提高了控制系统的响应速度和运行效率。

(2) 低速域无感方案研究

当电机启动或者运行在低转速区域时,其内部有效信号的信噪比通常较低,这使得用户想准确提取转子的位置和速度信息较为困难,传统的中高速无感控制方案在这种情况下可能无法成功提取转子信息。高频信号注入法的出现,有效的克服了这一问题,确保在低转速甚至零速条件下,也能准确获取转子信息。

该方法的核心思想是将高频电压或者电流信号注入到电机的基波模型中,这些信号会随着矢量控制过程被输出到 U、V、W 三相绕组上。而由此产生的高频电压或电流信号包含转子的位置和速度信息,将这部分信息经过滤波处理后,即可得到转子的动态信息。目前常见的高频信号注入有两种:旋转高频注入^[21-22]和脉振高频注入^[23-24]。

旋转高频信号注入,最早由 M.J.Corley 和 Robert D.Lorenz^[25]提出,适用于多种低速应用场景,其原理是在估计的 $d-q$ 坐标系中施加旋转高频信号,使 q 轴产生一个带有位置和转速信息的高频正弦响应电流,之后经过适当的处理,估算出转子信息。

脉振高频信号注入由 Jung IK Ha 首次提出,并在表贴式永磁同步电机的无位置传感器控制中得到应用^[26]。该方法通过在估计的 d 轴上施加高频余弦电压信号,利用电机自身的非线性饱和凸极效应,产生相应的高频响应电流,对此信号进行处理,能得到转子的位置速度信息。

1.3 本文的主要工作及创新

本论文所有工作在 DSP 上完成,DSP 在计算机中扮演着至关重要的角色。它是一种处理数字信号的技术,涉及从模拟信号中获取数据,然后对数据进行数字化、处理和分析的过程。在计算机中,DSP 通常用于处理音频、视频、图像等

数字信号。它被广泛应用于各种领域，包括通信、音频处理、图像处理、医学影像、雷达、无人机、汽车电子系统等。

本论文实现并优化了对于永磁同步电机在无位置传感器下，进行矢量控制的方法，并最终在实验室自研的 DSP 芯片上部署该系统。本文的主要工作包括如下四个部分：设计硬件三角函数指令实现方案，降低矢量控制中相应计算时间；封装矢量控制中相关算法库，采用非线性磁链观测器估计转子位置，并针对反正切计算带来的电机抖振问题，利用锁相环使电机运行平稳；在低转速下采用高频注入法，优化控制中的解调函数等，提高系统稳定性；改进混合控制策略，实现全速域下对电机的矢量控制，并在实验室的 DSP 上搭建无感矢量控制系统。本文的主要创新性如下：

(1) 设计硬件三角函数指令实现方案。

针对电机进行矢量控制的过程中，需要频繁进行正余弦以及反正切的三角函数计算，分析传统查表以及多项式拟合在计算时间以及结果精度上的不足，本文提出多区间二次函数拟合的方法。该方法利用了多项式拟合的优点，将定义域内的三角函数分为若干个区间，在每一个区间内用一个单独的二次函数进行拟合，提高计算结果的精度，而二次函数的系数则利用查表法进行获取，可以在一个时钟周期内获得三个系数。同时由于拟合的是二次函数，仅需两次乘累加即可完成计算，保证了三角函数的硬件指令可以在 5 个时钟周期内得到计算结果。在此基础上，考虑硬件采用的定点计算位宽会影响指令执行周期，设计软件代码，模拟硬件实现过程，通过对定点位宽的调整保证在设计的执行周期内，达到较高的计算精度。并且针对部分区间计算精度较差的问题，通过对输入值做位移操作，降低误差。

(2) 封装矢量控制算法库，优化非线性磁链观测器。

矢量控制中包含大量的坐标变换运算以及对脉冲宽度调制(Pulse Width Modulation, PWM)占空比的调整，这部分代码的运行时间占据算法运行时间的一半以上，为提高 DSP 对电机实时控制能力以及保证用户可以直接调用相关接口，减少对控制代码的编写，本文设计相关结构体并将算法库进行封装成对应接口。针对无位置传感器的控制需求，采用非线性磁链观测器的控制方案，通过对电机磁链、相电流、相电压采样分析，计算得到矢量控制所需要的转子位置信息。同时分析了传统磁链观测器中，采用的反正切计算中的除法，会将电机磁链中的噪声引入到转子角度值之中，造成电机运行时会产生高频抖振。为解决这一问题，设计加入锁相环控制方案，把估计的角度差作为输入，使误差不断接近零，从而锁定实际角度。

(3) 低转速下设计并优化高频注入法

本论文的设计目标是针对自研 DSP 芯片，为用户提供无感的全速域矢量控制方案，在低转速下，磁链观测器对转子角度估算能力较差，采用脉振高频注入法，通过向直轴注入高频信号，再对交轴的响应电流进行解调、滤波，分离出信号中的转子位置信息。传统高频注入法位置提取使用的调制函数，仅能将高频信号变为基频信号的二次谐波，对于低通滤波器的设置要求较高，通过优化调制函数，将其变为四次谐波；同时采用 IIR 滤波器(Infinite Impulse Response Filter)以及在锁相环中增加微分环节，提高高频注入法提取的转子位置精度，使在更低的转速下也能对永磁同步电机进行矢量控制。

(4) 改进混合控制策略，实现全速域电机矢量控制，搭建无感控制系统。

为实现对电机的全速域控制，将非线性磁链观测器与脉振高频注入相结合，分析现有的滞环切换与线性加权切换的优缺点，采用改进的混合控制策略，保证在低转速与中高转速分别使用对应控制策略的同时，在两种控制方法的过渡区间内，能综合两种无感方案的计算结果对电机进行控制，且在区间端点的切换更加平稳。最终在实验室自研的 DSP 芯片上，搭建完整的全速域无位置传感器矢量控制系统，包括外设驱动以及软件算法等部分，使用户在使用时，仅需调整文件中的相关电机参数等，即可实现对电机电流以及转速双闭环控制。

本文的章节安排如下：

第一章：研究背景及国内外现状。通过阐述 PMSM 无感矢量控制策略的背景，介绍该领域内的研究动态，进而引出电机无感控制下的难点，说明文章的主要研究内容与创新点。

第二章：三角函数硬件指令设计与优化。介绍 PMSM 矢量控制中常用的三角函数与性能要求，针对该问题提出一种多区间二次函数拟合的方案，并设计仿真程序与 TI(Texas Instruments)公司的精度与性能进行对比。

第三章：矢量控制及中高速域无感控制方法。分析介绍矢量控制中的基本流程，对相关算法模块进行封装，针对无传感器控制的问题，提出采用非线性磁链观测器，并针对该方法存在的高频抖动问题，采用锁相环进行优化控制。

第四章：低速域高频注入控制方案。针对磁链观测器在低速域下控制效果不佳的问题，采用脉振高频注入法，并优化其调制函数、滤波器与 PI 控制器，进一步提升观测效果。

第五章：电机全速域混合控制方案及系统搭建。针对低速与中高速不同的无感控制方案，采用混合控制策略将两种方法结合，实现全速域内对电机的矢量控制；同时搭建对应的控制系统，选择主控芯片，并设计主程序，中断子程序等。

第六章：总结与展望。对本文研究内容的成果进行总结，指出研究过程中的不足，对后续工作进行展望。

第 2 章 三角函数硬件指令设计与优化

在电机控制领域，DSP 具有广泛的应用，不仅可以对电机进行高效控制^[27]，还能进行信号的及时处理和对系统状态的实时监控。其中 DSP 在电机控制领域中较大的挑战在于，该场景下存在大量的浮点计算，特别是在磁场定向控制方法中，需要计算正余弦值和反正切值，常规的计算方法，在速度和精度上需要做一定的取舍，而设计一种三角函数硬件指令则可以很好的解决上述两种问题。本文基于多项式拟合的思想，将三角函数定义域划分为若干区间，每个区间内使用一个二次函数拟合，该方法能够保证三角函数计算结果的精度，同时因为拟合函数简单，使硬件能够在 5 个时钟周期内完成结果的计算。本章主要介绍三角函数硬件指令设计的区间划分的结果，以及软件模拟硬件计算的具体流程。

2.1 电机控制中的三角函数

近些年来，随着半导体行业的迅速发展，与电机控制相关的电子电力技术和控制芯片的性能得到了大幅提升，电机控制技术更加完善、成熟。在电机控制领域，DSP 能够提供高性能、实时、精确控制等功能，有助于实现电机的高效运行、实时控制和故障检测，而 DSP 的三角函数也被广泛应用于电机的六步控制、磁场定向控制以及电源系统当中^[28-29]。

2.1.1 三角函数的使用场景

目前在电机控制领域，最常见的控制方法为磁场定向控制，该方法将电机的磁链与其转矩解耦，使励磁与转矩分离，从而更好的控制电机的力矩与转速^[30-31]。但是 FOC 对于计算速度也有较高的要求，因为所有的控制过程都是在 DSP 的 PWM 中断或者模数转换器(Analog to Digital Converter, ADC)中断中完成，计算时间较长会影响电机控制效果。

正弦和余弦硬件指令主要用于生成空间矢量脉宽调制信号(Space Vector Pulse Width Modulation, SVPWM)，使控制的三相电机的相电流为正弦信号，保证输出平稳的转矩，实现对电机转矩、速度、位置的精准控制。在 FOC 中，正余弦主要作用环节为坐标变换模块，通过将转子在静止坐标系与旋转坐标系下相互转换，达到对电机转子状态信息的精确分析，经过计算得到的结果用于生成 PWM 信号^[32]，控制逆变器开关元件的通断，保证输入到电机的相电压波形为马

鞍波。除此之外在定位转子位置以及低转速下对电机进行控制时，需要生成特定频率的正余弦信号，该信号也需要对应的硬件指令辅助生成。

同样的，反正切硬件指令在电机控制领域也有广泛的应用，对于电机控制来说，最关注转子的位置与速度信息，对于有位置传感器的电机，一般通过霍尔传感器或者旋转变压器等有感器件获得转子的位置；而在无位置传感器的电机当中，一般通过对电机的磁链或者反电动势进行观测，推测出转子当前的位置，在观测完成后，通过对观测量进行反正切计算得到估算的转子位置。所以反正切计算结果的精度会直接对电机控制效果造成影响，需要保证反正切的硬件指令有较高的计算精度。

考虑到三角函数在电机控制中的重要性，一款专用于此领域的 DSP 芯片是否支持相关的硬件指令显得至关重要。本文展示了基于作者所在实验室开发的一款高性能 DSP 芯片的三角函数硬件指令方案的设计，旨在为硬件设计人员提供必要的分析，并通过对计算结果进行优化，进一步提升自研指令集的 DSP 在计算方面的性能。

2.1.2 当前三角函数计算的不足

在 DSP 的设计中，加减乘除等基本运算通常是依靠内核的算数逻辑单元 (Arithmetic Logic Unit, ALU) 的硬件指令来快速执行，与这些基础运算相比，三角函数的计算频率相对较低。因此，DSP 中的相关三角函数计算通常采用数学库或者其他数值方法，如泰勒级数展开等，来近似求解。尽管通过软件算法来实现 DSP 中的三角函数可以在一定程度上解决某些问题，但这种方法也会带来一定的局限性。首先，由于相关计算是在软件层面进行的，它需要占用一定的存储空间来存储软件代码和相关的查表数据；其次，软件的执行速度通常远远低于硬件指令的处理速度，尤其是当需要进行大量三角函数计算时，软件处理时间会显著增加，从而导致执行效率降低。

随着 DSP 在工业应用中的广泛使用，满足特定应用对三角函数计算速度的要求，开发专门的硬件指令用于加速这一计算过程变得尤为重要。这其中一种较为直接的解决方案是通过调用底层硬件指令，利用 ALU 内的指令组合，以等效的形式执行这些计算。这种方法能够在不修改现有的硬件架构的前提下，做到支持三角函数的硬件指令计算，显著降低执行时间，提高运行效率。然而这一方案本质上仍旧是在硬件层面对原本的软件代码进行实现，并未引入专为三角函数计算设计的硬件指令，因此在执行时间与运行速度方面虽有所改进但仍存在较大的优化空间。

随着芯片制造技术的日益精进和晶体管密度的显著提升，将三角函数计算的

专用硬件指令集成到 DSP 处理器中已经成为现实。2015 年, TI 公司引入了浮点三角函数运算单元 TMU(Trigonometric Math Unit), 此单元能够使三角函数的计算通过单个指令实现。这种集成方法涉及将特定算法的实现配置于相应的硬件资源之中(包括逻辑门、电路和触发器等), 从而在硬件层面完成对三角函数的计算。与软件代码或硬件指令组合相比, 该方法大幅提升了运算速度, 能够满足电机控制等场景的计算需求。尽管引入专门硬件指令大大缓解了三角函数计算速度的瓶颈, 计算精度仍是一个巨大挑战。

目前三角函数硬件指令面临的主要问题在于计算精度不足, 难以满足不断提高的应用需求。不同硬件指令的实现方案对结果精度有着显著影响, 而定点与浮点计算的选择也影响这一精度。特别是在电机控制等需要实时信号处理的应用场景中, 确保三角函数指令能够快速且准确地完成计算至关重要。本文以多项式拟合为基础, 提出了一种硬件指令加速设计方案, 将三角函数的定义域划分为 512 个区间, 每个区间采用一个简单的二次函数进行拟合, 以得到高精度的计算结果。由于拟合函数结构简单, 该方案易于硬件实现, 同时具有高效和高精度的优点。该指令能够在 5 个时钟周期内完成计算, 在实验室 400M 主频的 DSP 上, 相应三角函数的计算能够在 12.5ns 内完成。

2.2 指令设计路线

三角函数, 如正弦、余弦、反正切, 是数学中一类重要的函数, 为了在硬件层面高效计算函数的值, 必须采用专门的算法和技术。同时由于三角函数计算中频繁涉及到浮点数运算, 确保输入与输出均为浮点数格式也十分关键。

2.2.1 常用算法与优缺点

硬件实现三角函数的算法各有其优缺点, 当前技术路线下常见算法包括查表法、CORDIC 算法、多项式近似法。

查表法: 用该方法实现硬件三角函数的计算, 具有简洁与高效的特点。它通过预设的查找表, 将常用的三角函数值存储于计算机内存中, 使得在接收到特定角度输入时, 能够迅速匹配并输出相应结果。这种方法的显著优势在于其稳定的响应时间和一致的结果精度, 因为硬件仅需依据输入值在查找表中查询即可找出结果, 大大简化了计算过程。但是这种方法也存在一些不足, 它的应用范围受限于查找表内的角度值, 若输入角度超过预设范围, 系统只能通过四舍五入的方式进行近似计算, 会导致结果的精度降低, 也使得查找过程变得繁琐, 从而影响计算速度。查表法在特定领域内可以显著提升计算速度, 有较高的使用性, 但在某

些情况下也存在一定的精度损失。

CORDIC 算法：即坐标旋转数字计算方法，是 J.D.Volder 于 1959 年提出的，用于三角函数、双曲线、指数、对数计算的方法，该方法通过基本的加法和移位运算代替乘法运算得到结果。该方法的优点在于加法和移位运算对于硬件而言既简单又高效，能够做到在有限的迭代次数内，达到较高的计算精度，且使用的角度输入范围也更加广泛；缺点是在高精度应用场合下，难以在少量迭代次数下取得较高的精度，需要在执行周期与计算精度之间做平衡。

多项式近似法：本质是通过选择不同的多项式函数来模拟复杂的三角函数曲线，该方法可以在保证计算精度的同时，实现高效的计算速度。通过调整多项式的次数，使得在不同的应用场景中找到计算精度与计算资源之间的平衡点。该方法的优点在与其灵活性和适应性，能够根据具体的计算需求来优化多项式的选择，从而确保在一定的计算精度范围内，达到理想的计算效率；缺点是想要准确地拟合三角函数的曲线，可能需要较高次数的多项式，导致计算的复杂度和所需的硬件资源增加，在实际使用中，需要在多项式的计算精度和硬件资源消耗之间做出权衡，找到最适合其应用场景的解决方案。

不同的实现算法有不同的使用场景，本文所设计的三角函数硬件指令方案需要在 5 个时钟周期内完成结果的计算，同时考虑到计算的精度、功耗、面积等因素，需要对不同的指标进行权衡。

2.2.2 多区间二次函数拟合

在 DPS 处理器中，硬件能够使用的资源是有限的，设计三角函数硬件指令方案需要在精度方面做出一定的取舍。本文所采用的多区间二次函数拟合方案是一种在不同区间内将三角函数（正弦、余弦、反正切），使用不同的二次函数来拟合逼近的方案。该方法利用了多项式拟合的优点，保证了在不同的输入下，硬件都可以在相同的时间内完成运算，同时因为在不同的区间下采用不同的二次函数拟合，可以保证在指令定义域范围内，仅使用两次乘累加即可保证有较好的拟合结果。采用多区间二次函数拟合主要分为两个步骤：

区间划分：首先需要将硬件指令定义域划分为若干个区间，每个子区间用单独的二次函数进行拟合。区间划分的依据需要根据指令以及硬件实现的实际情况考虑，在采用二次函数拟合的过程中，函数中的三个系数均需根据输入的浮点值查表得到，当区间数划分过大时，会导致存放系数所需的空间过大，其次区间划分不规则，会使得硬件查表所需确定的浮点值的尾数位更多，导致指令执行周期增加。因此在区间划分时需综合考虑以上两种影响因素。

拟合的二次函数：在确定好的每个子区间内，用二次函数拟合三角函数，每

个函数的表达式为：

$$f(x) = C_2x^2 + C_1x + C_0 \quad (2-1)$$

其中 C_2 、 C_1 、 C_0 分别为二次函数的二次项、一次项和常数项系数。

为了减少硬件计算过程中乘法器的使用，将式(2-1)改成如下的两级乘累加的形式：

$$f(x) = (C_2x + C_1)x + C_0 \quad (2-2)$$

本文提出的拟合方法，最为关键的是在较高精度的条件下，减少区间划分的数量同时保证硬件浮点值查表的便捷。出于以上因素考虑，硬件直接设计指令对三角函数进行计算存在以下问题：硬件设计周期较长，底层设计较为复杂，当需要更改区间数量或区间划分范围时，需要改动大量的代码才能完成目标，工作量大；其次为保证计算结果精度符合设计要求，需要对定义域内的大量数据进行测试比对，硬件仿真运行时间较长，不足以快速得到结果；最后，硬件设计对于计算采用定点计算而非浮点计算的方式，定点计算的位宽会直接影响到结果精度与运行时间，增加定点位宽会提高精度但可能导致 5 个周期内无法完成计算，需要确定最佳的定点位宽保证精度与效率的平衡。因此在硬件设计之初，需要软件模拟硬件实现流程，确定指令设计所需的相关数据。

2.3 软件仿真设计

软件仿真的目的是协助硬件设计人员确定在多区间二次函数拟合方案下，完成三角函数指令（正弦、余弦、反正切）所需要的数据，具体包括：区间划分数、各区间端点值、各区间查表的三个系数、两级乘累加模块的定点计算位宽。同时为了软件模拟过程与硬件实现过程一致，需要保证中间定点计算流程与硬件相同。

2.3.1 软件仿真流程

首先对于多区间二次函数拟合做初步分析，取 TI 公司的 TMS320F280049C 这款芯片中 TMU 内部的 `sinpuf32`、`cospuf32`、`atanpuf32` 指令的一百万测试数据与结果，并将这些输入值在 Matlab 中运行，把得到的所有结果保存起来，用于后续分析。考虑到硬件定点计算结果精度小于等于浮点计算精度，先通过软件模拟浮点查表与计算过程，确定区间划分情况。

浮点模拟流程较为简单，基本为重复测试，本文在此仅指出模拟流程与结果，不对中间过程进行阐述。浮点计算的区间划分与二次函数系数，利用 maple 数学

工具的 **minimax rational approximation** 方法生成，之后将得到的一百万测试数据在该拟合结果下，依据式(2-2)进行计算，将拟合方法与 Matlab 结果的相对误差和 TI 与 Matlab 结果的相对误差进行对比分析，确保浮点计算下，拟合方法的精度大于等于 TI，如果精度小于 TI 则进一步增大区间划分数量。经过多轮模拟、测试、分析，最终确定的拟合方案为，在 $[0, 1]$ 的区间内，将输入划分为 512 个小区间，每个小区间内分别用不同的二次函数拟合计算三角函数值，本文所拟合的三角函数形式分别为：

$$f(x) = \sin\left(\frac{\pi}{2}x\right) \quad x \in [0, 1] \quad (2-3)$$

$$f(x) = \cos\left(\frac{\pi}{2}x\right) \quad x \in [0, 1] \quad (2-4)$$

$$f(x) = \frac{\text{atan}(x)}{2\pi} \quad x \in [0, 1] \quad (2-5)$$

在完成对区间划分和查表系数的确定之后，开始利用软件模拟硬件计算过程，仿真流程如图 2-1 所示，包括输入的浮点转定点模块、输入查表模块、两级乘累加模块、输出定点转浮点模块。

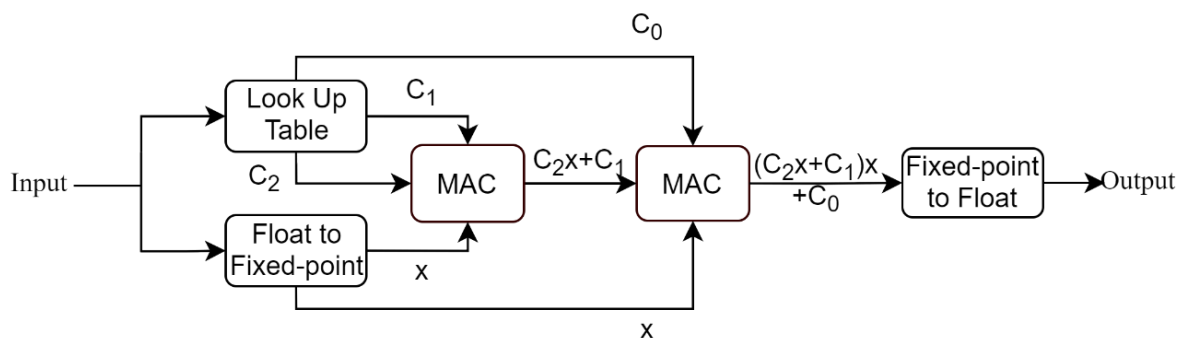


图 2-1 仿真流程图

软件仿真过程中需要确定的定点位宽为乘法位宽和加法位宽，其中乘法位宽与定浮点转换的位宽一致。其次为减少硬件设计过程中的加法器，非必要的舍入环节均采取舍去操作，减少因进位导致的全加器使用。

2.3.2 定浮点转换模块

定点数与浮点数代表了数字在计算机系统中的两种基本小数表示方法，定点数通过固定小数点的位置来表示数字，适用于需要精确小数位数表示的场景。而浮点数依赖于科学计数法来表示数字，允许使用有限的存储空间来表达更广泛的数值范围和更高的数值精度。在软件仿真中，使用 IEEE754 单精度浮点数格式^[33]，该格式占用 32 位（4 字节）内存空间。根据 IEEE754 标准，这 32 位被划分

为三个部分，1 位符号位，用于表示数值的正负；8 位指数位，确定数值的大小范围；23 位尾数位，定义数值的精度；除此之外还包含一个隐藏位。定浮点转换模块的作用在于将浮点数转换为定点数进行处理，经过对应的计算之后，再将结果从定点数转换回浮点数。

浮点运算主要包括四个步骤：对阶、计算、规格化以及舍入。在这一过程中，对阶是指将两个浮点数调整至相同的指数级别，以便能够对它们的尾数位和隐藏位执行定点数计算。在浮点模拟过程中，已经将三角函数指令的定义域限制在 $[0, 1]$ 内，同时 atan 拟合的二次函数系数均在 $(-1, 1)$ 内，所以在对阶过程可以将指数位向 127 对齐即可，而 \sin 、 \cos 拟合的二次函数次数在 $(-2, 2)$ 内，在对阶过程需要将指数位向 129 对齐，以保证计算结果正确。软件仿真需要确定的定点乘法位宽与浮点转定点的位宽一致，在仿真过程中，根据计算需求的变化调整定点数的位宽，获取在不同定点精度条件下的结果，使硬件设计人员能够根据最佳仿真结果来设计硬件指令，优化硬件性能。

浮点转定点的大致步骤如下：

- 1) 将输入浮点数分解为符号位、指数位、尾数位三部分；
- 2) 将输入浮点数的指数位与定点计算所代表的指数相减，确认尾数位与隐藏位组成的定点数所需移动的位数；
- 3) 依据符号位确定定点计算是否需要转换为补码计算；
- 4) 将符号位与定点数合并，组成最终输入的定点结果。

在设计浮点转定点模块时，因为转换成定点的输入会直接与二次函数的二次项系数相乘，故转换的位宽与乘法位宽一致，且需根据不同需求更改位宽大小。其次因为定点数表示的范围小于浮点数，在转换过程中需要考虑定点是否能够表示全部的浮点位数结果，如果不能则采去四舍五入，硬件部分加入一个全加器。浮点转定点流程图如图 2-2 所示，在硬件流水线的第一个执行周期内，采用移位运算将浮点数转换成定点数，同时进行浮点数查表，获得二次函数的三个系数。

软件仿真的浮点转定点还包括系数的定点结果获取，因为利用 maple 数学工具获得的二次函数系数全是浮点数，为减少硬件资源的消耗，需要保证硬件浮点数查表能直接获得到对应系数的定点值。拟合的系数存在负值，故需要将定点结果转换成对应的补码以便后续的乘累加计算，同时二次项系数会直接与输入定点相乘，其位宽与乘法位宽一致，一次项与常数项系数则是进行加法操作，位宽与加法位宽一致即可。

定点转浮点的大致步骤如下：

- 1) 将定点数进行左移，直至最高位为 1，小数点位于最高位后，此时定点数除去最高位 1 后的高 23 位即为浮点的尾数位（如果最高位后无 23 位则

- 补 0，多于 23 位结果则采取四舍五入)；
- 2) 计算步骤 1) 中左移的位数，与浮点指数位标准值 127 相减，得到浮点结果的指数位；
 - 3) 根据定点结果的符号位确定浮点结果符号位；
 - 4) 将上述步骤得到的符号位、指数位、尾数位合并，组成浮点结果。

与浮点转定点模块相同，转换过程中需要考虑舍入操作，当最高位后的定点结果位数大于 23 位时，需对第 23 位进行舍入，硬件部分加入一个全加器，当位数小于 23 位时，则直接补 0，直至位数等于 23 位。定点转浮点流程图如图 2-3 所示，在硬件流水线的第五个执行周期内，利用优先编码和移位运算将定点结果转换成浮点结果。关于符号位，考虑到正余弦函数的周期性以及反正切函数的需求，输出结果均为正值，即结果符号位为 0。

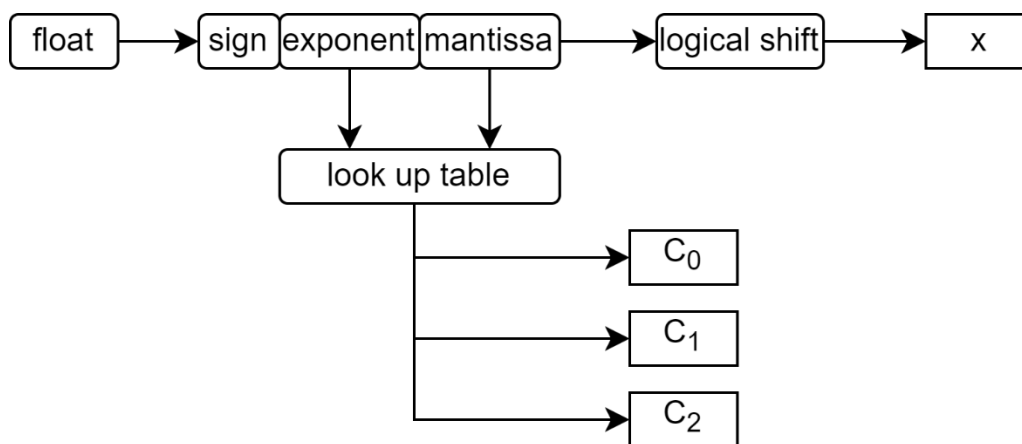


图 2-2 浮点转定点流程图

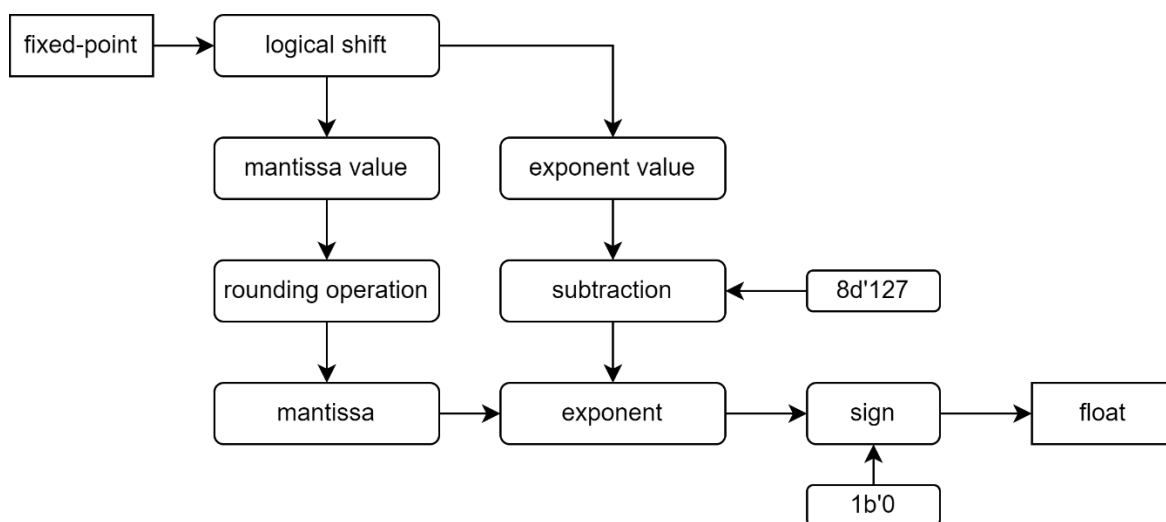


图 2-3 定点转浮点流程图

2.3.3 两级乘累加模块

乘累加^[34](Multiply-Accumulate, MAC)是一种在 DSP 中常见的数值计算方法,它将加法与乘法相结合,用于数学运算、数字信号处理等方法。乘累加一般的操作形式为: $y = ax + b$ 。硬件实现乘法一般通过移位和加法操作实现,例如当 32 位乘上 32 位时,将第一个 32 位数作为乘数,逐位读取另一个 32 位数的 bit 位,若读取的当前位为 1,则在结果中加上乘数并左移一位,否则直接左移一位,直到读取完所有的 bit 位,即可通过 32 个加法器实现 32 位的乘法。而对于乘累加操作则是在乘法器的基础上,增加一级加法,硬件通过相关的优化算法,将加法并行执行,使得乘法与乘累加执行速度与资源消耗上相同。

硬件为加速乘累加计算性能,采用华莱士树^[35](Wallace Tree)算法与 Booth4 编码^[36],减少乘法器中执行的乘法次数,降低计算复杂度。考虑到硬件对于位域操作简便,软件模拟华莱士树和 Booth 编码流程复杂,故在仿真过程中仅采用 Booth2 编码,保证软件结果与硬件优化后的乘累加计算结果一致即可。

软件仿真两级乘累加模块流程图如图 2-4 所示,定点计算的精度与执行效率与乘累加模块的加法位宽和乘法位宽有关。在仿真过程中,需要不断调整这两个位宽以保证结算结果的精度,并使硬件设计人员在三个时钟周期内,利用华莱士树算法和 Booth4 编码完成对定点结果的计算。

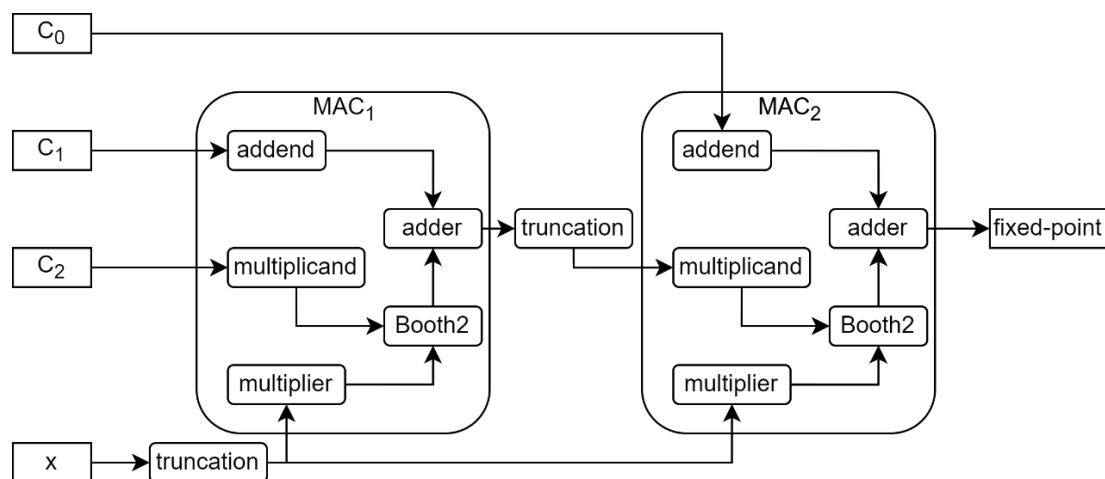


图 2-4 两级乘累加流程图

软件利用 Booth2 获得乘累加结果的步骤如下:

- 1) 将乘法计算的乘数与被乘数位宽从加法位宽截断成乘法位宽,降低乘累加运算时间,其中对于截断的部分直接做舍去操作,减少因进位导致的全加器的引入;
- 2) 获得乘数 x 与 $-x$ 的补码,被乘数 y 的补码;

- 3) 进行 Booth2 编码计算, 依据不同的编码结果, 分别将结果加上 x 、 $-x$ 、 0 并右移一位;
- 4) 将 Booth2 编码计算后的定点结果位宽截断成加法位宽, 对于截断的部分做舍去操作, 减少因进位导致的全加器的引入;
- 5) 将乘法结果与定点加数相加, 两者位宽皆为加法位宽, 结果位宽不会增加, 不需要做舍入操作。

2.3.4 正余弦指令仿真与优化

根据图 2-1 所示的流程图, 编写 c 代码进行软件仿真, 因为正弦函数与余弦函数可以相互转换, 故本文以正弦函数实现为例, 余弦函数可以按照相同的方法进行处理。软件代码主要包括三个部分, 浮点转定点 F32toIQ、定点转浮点 IQtoF32 和乘累加模块 IQmul、IQadd。软件代码见附录 1, 附录 2, 附录 3 所示。

仿真代码中的 BitWidth 与 MulWidth 分别为定点加法位宽和乘法位宽, 这两个变量通过 define 进行宏定义, 使得在仿真测试环境下可以不断调整位宽大小, 模拟硬件计算正弦指令流程。仿真测试步骤如下:

- 1) 收集 TI 与 Matlab 上定义域内一百万测试数据, 作为精度参考;
- 2) 设置加法位宽和乘法位宽, 运行仿真代码得到计算结果, 并与步骤 1 中的结果进行比对;
- 3) 分析三种结果的 IEEE754 下浮点数 bit 位结果与十进制下小数结果, 比较 bit 位误差与十进制相对误差;
- 4) 重复步骤 2、3, 直至 bit 位误差与相对误差在可接受范围之内。

经过多轮仿真测试之后, 最终确定正弦函数指令的定点加法计算位宽为 34 位, 乘法计算位宽为 30 位。因此, 对于查表取得的三个系数, 二次项系数 C_2 存储位宽为 30 位, 一次项系数 C_1 和常数项系数 C_0 存储位宽为 34 位, 硬件指令在拿到角度进行计算时, 输入的浮点数在转换成定点时的位宽也为 30 位, 减少额外的截断操作。

以上测试均针对定义域范围内的常规数值, 对于特殊区间内的数据需要单独进行处理。比如在第一个区间内, 因为正弦函数值大部分为 0, 并且函数曲线与一次函数接近, 二次函数拟合效果较差, 同时参考市面上常见 DSP 的三角函数运算结果, 本文对第一个区间做出如下调整: 当输入角度在 $[0, 0x3220\ 0000)$ 内, 因输入角度过小, 结果近似于 0, 故直接将输出的浮点值设置为 0; 当输入角度在 $[0x3220\ 0000, 0x3700\ 0000)$ 内, 此时三角函数曲线与一次函数拟合效果更好, 因为当输入角度较小时 $\sin(x) \approx x$, 故将输入直接乘上 $\pi/2$ 。

经过上述处理后的正弦函数计算结果, 在分成 512 个区间内, 大部分区间结

果优于 TI 的 TMU 但在小区间内效果不太理想。对这部分区间进行分析可知，该区间浮点表示的数值过小，IEEE754 的浮点利用指数位可以保证始终有 24 位的有效精度，但在进行定点计算时指数位通过向高位补 0 体现，在高位有较多的 0 时，原本的 24 位尾数精度会丢失，导致二次函数拟合的点向左偏移，计算结果精度降低。对于这一问题，本文对原本的方案进行如下调整：将输入的定点值向左移 n 位，以减少高位的有效 0，增减有效尾数位，为保证乘累加计算结果正确，将一次项系数 C_1 向右移 n 位，二次项系数 C_2 向右移 $2n$ 位，以抵消输入左移导致的计算错误。移位的 n 值同样需要在软件层面进行仿真，针对每个区间单独进行测试，得到最优的 n 值。为达到此目的，对于定点转浮点的仿真代码需要进行调整，增加 MoveBit 变量，来测试不同移位下结果精度。最终软件仿真完成得到的区间划分、二次函数系数如附录 4 所示，移位区间与移位值如附录 6 所示。

2.3.5 反正切指令仿真与优化

因为指令实现采用的方案都为多区间二次函数拟合，所以仿真测试流程相同，软件代码也与正弦函数软件代码类似，在此不进行展示。经过多轮仿真测试之后，最终确定反正切函数指令的定点加法计算位宽为 34 位，乘法计算位宽为 28 位。对于查表取得的三个系数，二次项系数 C_2 存储位宽为 28 位，一次项系数 C_1 和常数项系数 C_0 存储位宽为 34 位，指令拿到输入进行计算时，输入的浮点数在转换成定点时的位宽也为 28 位，减少额外的截断操作。

同样的在第一个区间内，当输入在 $[0, 0x3160\ 0000)$ 内，因输入角度过小，结果近似于 0，故直接将输出的浮点值设置为 0；当输入在 $[0x3160\ 0000, 0x3200\ 0000)$ 内，输出 $0x3080\ 0000$ 。

对于小区间内拟合结果差的问题，采用与正弦函数同样的处理方法，将输入的定点值向左移 n 位，一次项系数 C_1 向右移 n 位，二次项系数 C_2 向右移 $2n$ 位，以抵消输入左移导致的计算错误。采用与正弦函数同样的仿真优化措施，最终得到区间划分、二次函数系数如附录 5 所示，移位区间与移位值如附录 7 所示。

2.4 软件仿真结果及比较

为了证明软件仿真模拟多区间二次函数拟合三角函数的正确性，本文与 TI 公司用于电机控制领域的 DSP 处理器 TMS320F280049C 中的三角函数硬件指令进行对比。比对主要从效率和精度两个方面进行比较。

首先在效率方面，TI 的硬件指令执行需要 4 个时钟周期，处理器主频为 100MHz，总消耗时间为 40ns；本文基于实验室自研指令集的 DSP，硬件指令执

行需要 5 个时钟周期，处理器主频为 400MHz，总消耗时间为 12.5ns。

精度方面，前期仿真测试基于软件代码进行，利用一百万数据进行分析比较，本章节展示的精度数据来自硬件 VCS 仿真测试平台，因前仿效率较慢，故在所有区间内共取 50000 个数据进行测试。在正弦函数所有的测试结果中，有 68.9% 的结果与 TI 一致，26.65% 的结果优于 TI，仅有 4.45% 的结果低于 TI。在 Matlab 中将拟合方案的相对误差与 TI 的相对误差进行展示，考虑到当输入较小时，拟合方案与 TI 的相对误差都较大，故分成两部分展示，图 2-5 展示正弦函数小区间内相对误差分析图，图 2-6 展示正弦函数剩余区间内相对误差分析图。

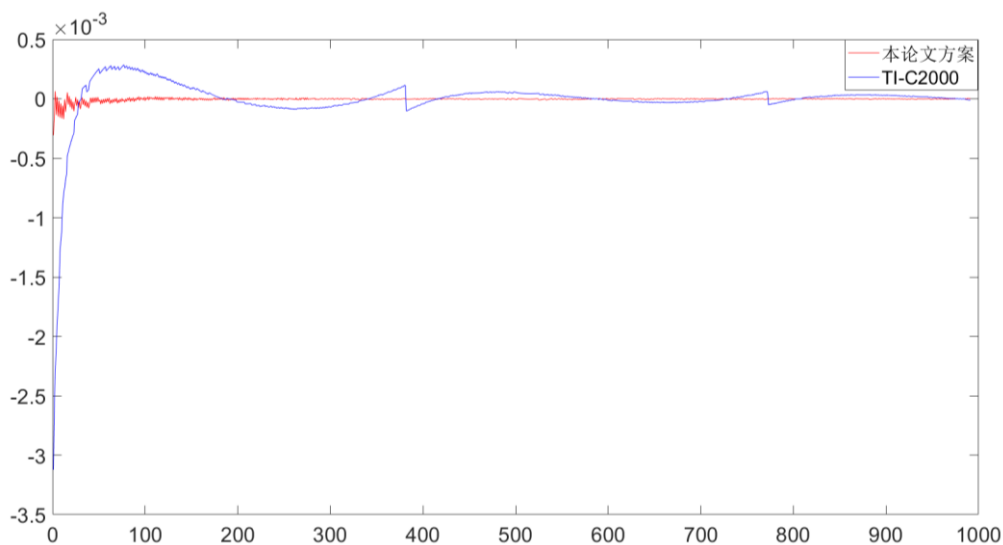


图 2-5 正弦函数小区间内相对误差分析图

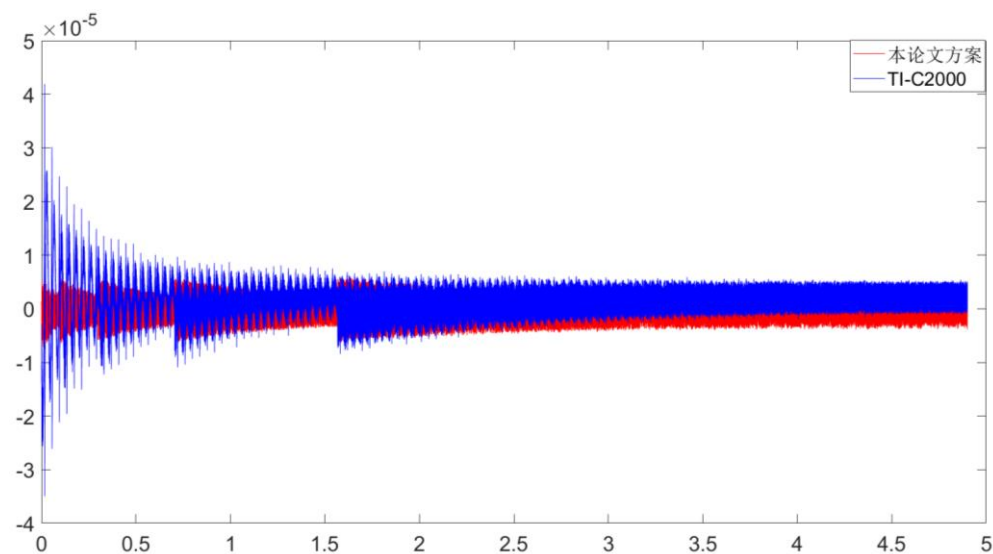


图 2-6 正弦函数剩余区间内相对误差分析图

图中红色曲线为本文与标准结果的相对误差，蓝色曲线为 TI 与标准结果的相对误差。通过曲线可以看出，大部分情况下，蓝色曲线覆盖红色曲线，即 TI 相对误差大于本文设计的方案。由于两者指令实现原理不同，TI 相对误差偏向于 x 轴上方，本文的方案分布在 x 轴两侧，当考虑相对误差的绝对值时，蓝色曲线则会基本覆盖在红色曲线之上。

除了相对误差之外，因为计算机中，对于小数采用的是 IEEE754 格式的浮点数，其表示的数据精度有限，会出现测试中优于或低于 TI 的结果在 16 进制下无法体现，故也对 16 进制的 bit 位误差进行分析，与相对误差分析相同，图 2-7 展示正弦函数小区间内 bit 位误差分析图，图 2-8 展示正弦函数剩余区间内 bit 位误差分析图。

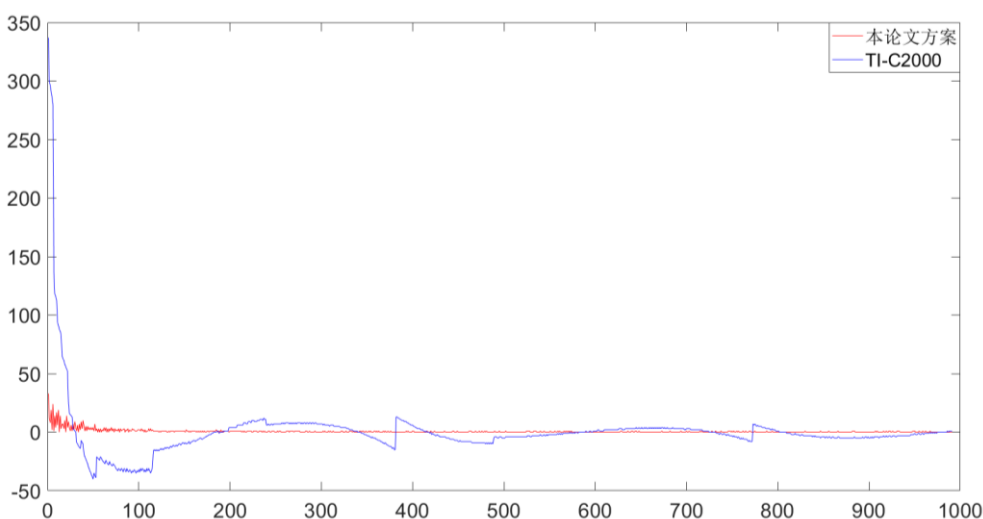


图 2-7 正弦函数小区间内 bit 位误差分析图

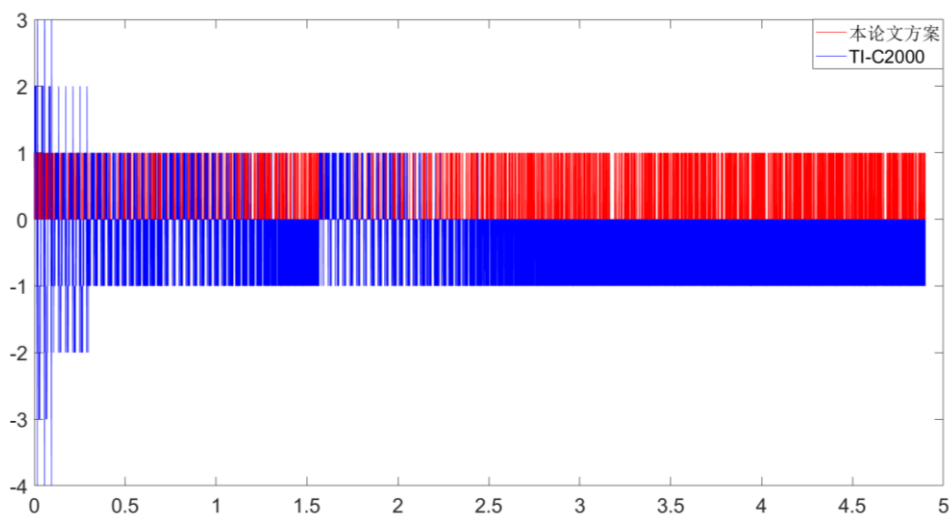


图 2-8 正弦函数剩余区间内 bit 位误差分析图

图中红色曲线为本文与标准结果的 bit 位误差,蓝色曲线为 TI 与标准结果的 bit 位误差。通过图中可知,当输入角度较大时, TI 的 bit 位误差在某些情况下会大于 1, 本论文提供方案的 bit 位误差基本为 1, 且存在误差的数据量小于 TI; 当输入角度较小时, 本论文的 bit 位误差远远小于 TI。综上所述, 对于正弦函数, 本文提出的多区间二次函数拟合的方案, 在性能和精度上都优于 TI 的 DSP 处理器。

反正切函数对比结果与正弦函数类似, 图 2-9 展示反正切函数小区间内相对误差分析图, 图 2-10 展示反正切函数剩余区间内相对误差分析图, 图 2-11 展示反正切函数小区间内 bit 位误差分析图, 图 2-12 展示反正切函数剩余区间内 bit 位误差分析图。

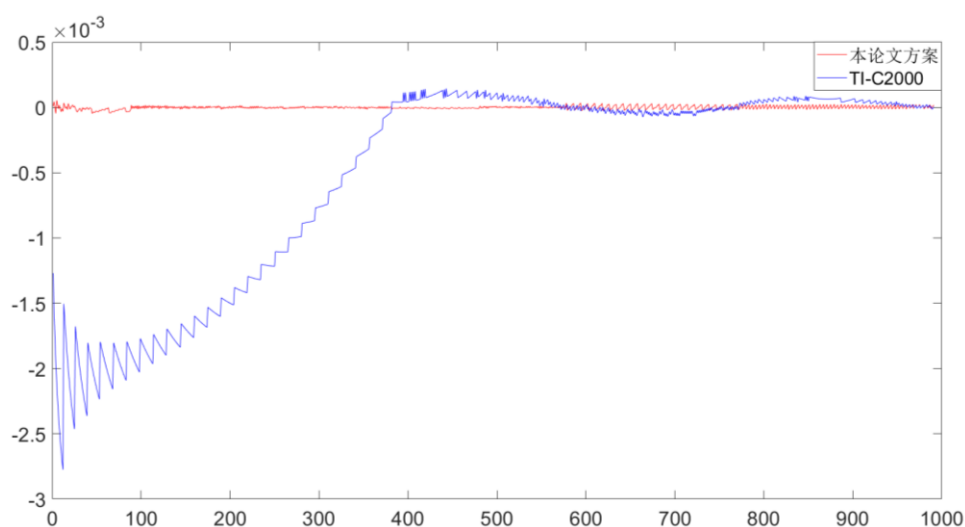


图 2-9 反正切函数小区间内相对误差分析图

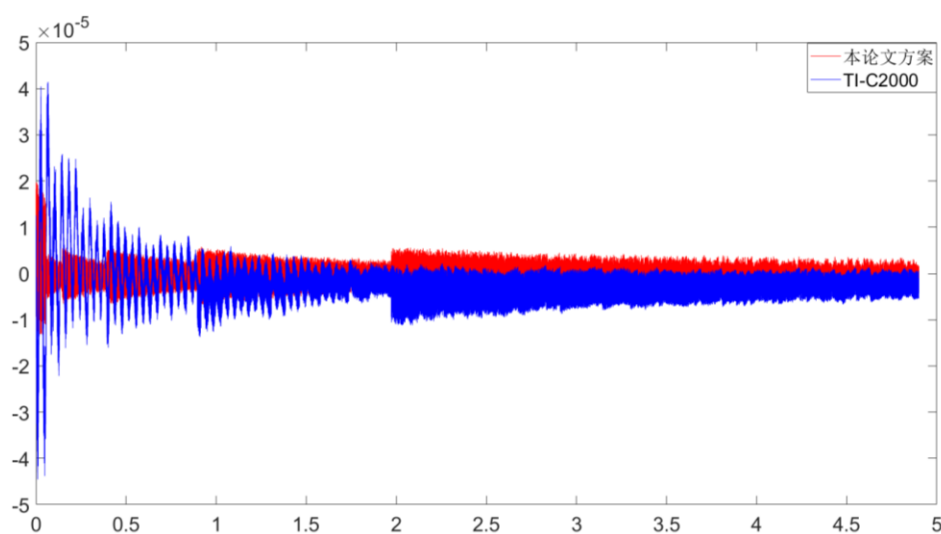


图 2-10 反正切函数剩余区间内相对误差分析图

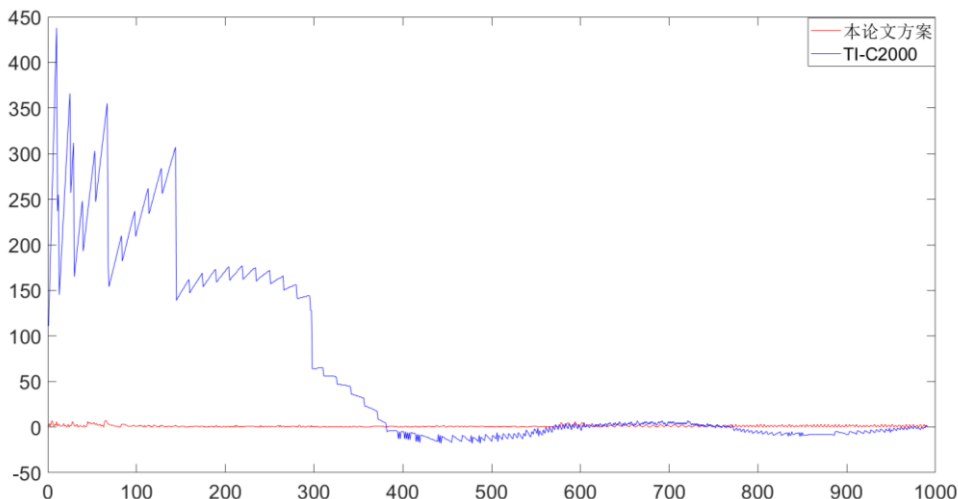


图 2-11 反正切函数小区间内 bit 位误差分析图

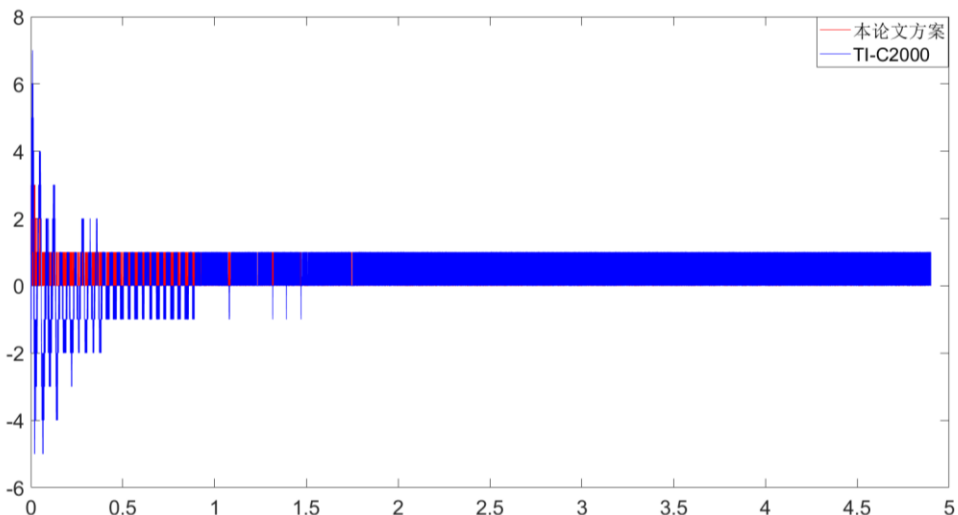


图 2-12 反正切函数剩余区间内 bit 位误差分析图

在反正切函数所有的测试结果中，有 63.84%的结果与 TI 一致，33.18%的结果优于 TI，仅有 2.98%的结果低于 TI。从图中可以看出，大部分情况下，蓝色曲线覆盖红色曲线，即 TI 相对误差大于本文提供的方案。TI 相对误差偏向于 x 轴下方，本文的方案分布在 x 轴两侧，当考虑相对误差的绝对值时，蓝色曲线则会基本覆盖在红色曲线之上。当输入较大时，TI 的 bit 位误差在某些情况下会大于 1，本论文提供方案的 bit 位误差基本为 1，且存在误差的数据量小于 TI；当输入角度较小时，本论文的 bit 位误差远远小于 TI。综上所述，对与反正切函数，本文提出的多区间二次函数拟合的方案，在性能和精度上都优于 TI 的 DSP 处理器。

2.5 本章小结

本章介绍了电机控制领域使用的三角函数硬件指令的加速方案。先是说明三角函数硬件实现的各种算法与优缺点，在此基础上提出多区间二次函数拟合的方案，并采用软件仿真的方式进行验证；之后介绍软件仿真设计的流程与各模块的功能与思路；随后通过仿真代码确定具体实现步骤，并针对设计过程中的不足进行优化；最终展示了仿真测试的结果并与其他方案进行对比，验证本文提出的方案的有效性。

第3章 矢量控制及中高速域无感控制方法

永磁同步电机以其结构紧凑、效率高、体积小和广泛的调速范围等特点^[37-38],成为新能源汽车领域首选的驱动电机。随着全球新能源汽车行业的蓬勃发展,永磁同步电机的应用前景日益广阔。在多样化的应用场景中,永磁同步电机采用不同的控制策略以满足特定需求,常见的包括矢量控制、直接转矩控制和模型预测控制等方法。本文所研究的矢量控制策略,本质上是一种利用变频器调节输出电压的幅值和频率,从而驱动三相电机的变频控制技术。它通过坐标变换技术,将电机中的相关物理量从三相坐标系转换到旋转坐标系,通过控制旋转坐标系中的电流矢量的方向和大小来驱动电机旋转。本章将详细介绍矢量控制的理论基础,并在此基础上将相关算法部分进行封装,提供电机控制的标准方案,同时针对无传感器控制要求,分析优化非线性磁链观测器方案。

3.1 电机控制数学模型

常见的永磁同步电机有两种,其根据电机中永磁体的安装位置不同分为:表贴式永磁同步电机(Surface Mounted PMSM, SPMSM)^[39,40,41]和内置式永磁同步电机(Interior Mounted PMSM, IPMSM)^[42-43]。两者的关键区别在于永磁体相对于转子的具体位置:表贴式电机的永磁体安装于转子表面,而内置式电机的永磁体嵌入转子内部。表贴式电机由于电感的交轴和直轴值基本一致,凸极性效应较弱,无磁阻转矩,因此具备结构简单、成本低廉等优势,适合于对控制精度有较高要求的应用场合;相反,内置式电机在交轴和直轴的电感差异显著,凸极性效应明显,非常适用于弱磁控制策略,虽然结构更为复杂、提供的力矩更大,但它更适用于需求低速精确控制的场景。

永磁同步电机是一个非线性、复杂的时变系统,为了分析的便利性,需要对数学模型进行一定程度上的简化,具体包括如下几点:

- 1) 线圈绕组分布规律,两两相差 120° ;
- 2) 磁路工作在线性段,忽略磁滞和涡流损耗的影响;
- 3) 转子上无阻尼绕组。

在分析电机控制的数学模型时,主要采用三种坐标系进行描述:三相坐标系 $A-B-C$; 静止坐标系 $\alpha-\beta$; 旋转坐标系 $d-q$ 。这三种坐标系可以实现相互转换,从而适应不同的分析和控制需求。 $A-B-C$ 坐标系直接与定子绕组的物理位置对应; $\alpha-\beta$ 坐标系中, α 轴与 A 轴对齐, β 轴与 α 轴垂直; $d-q$ 坐标系的 d 轴与电机的永

磁体轴线同向，并随着永磁体转子的旋转而旋转，保持与 α 轴的角度差为 θ ，而 q 轴则与 d 轴垂直。这种坐标系的设置便于将电机的动态控制问题转换为更易于分析和计算的形式，从而实现对电机性能的精确控制。三个坐标系相对位置关系如图 3-1 所示。

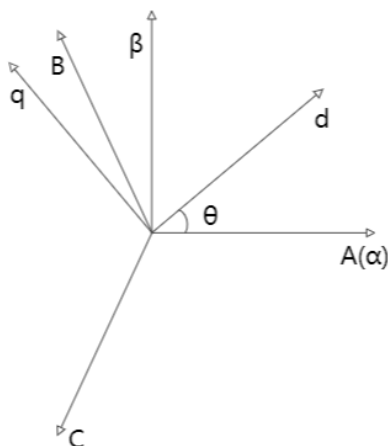


图 3-1 三种坐标系位置关系图

3.1.1 三相静止坐标系下的数学模型

在三相静止坐标系 A-B-C 下，定子的电压方程如式 3-1 所示

$$\begin{pmatrix} u_A \\ u_B \\ u_C \end{pmatrix} = \begin{pmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{pmatrix} \begin{pmatrix} i_A \\ i_B \\ i_C \end{pmatrix} + \begin{pmatrix} \psi_A \\ \psi_B \\ \psi_C \end{pmatrix} \quad (3-1)$$

其中 u_A 、 u_B 、 u_C 为三相电机各相间电压； i_A 、 i_B 、 i_C 为三相电机各相间电流； R_s 为定子电阻； ψ_A 、 ψ_B 、 ψ_C 为三相电机各相间绕组的磁链，在三相坐标系下，磁链方程如式 3-2 所示

$$\begin{pmatrix} \psi_A \\ \psi_B \\ \psi_C \end{pmatrix} = \begin{pmatrix} L_{AA} & L_{AB} & L_{AC} \\ L_{BA} & L_{BB} & L_{BC} \\ L_{CA} & L_{CB} & L_{CC} \end{pmatrix} \begin{pmatrix} i_A \\ i_B \\ i_C \end{pmatrix} + \begin{pmatrix} \psi_f^A \\ \psi_f^B \\ \psi_f^C \end{pmatrix} \quad (3-2)$$

ψ_f^A 、 ψ_f^B 、 ψ_f^C 为三相绕组上的磁链，其大小与定子磁链 ψ_f 有关； L_{AA} 、 L_{BB} 、 L_{CC} 为定子绕组的自感，理想情况下等于 L_s ； L_{AB} 、 L_{AC} 、 L_{BA} 、 L_{BC} 、 L_{CA} 、 L_{CB} 为三相绕组之间的互感，若假定不同绕组之间的互感值相等，而三相线圈之间采用星形连接，根据基尔霍夫电流定律有 $i_A + i_B + i_C = 0$ ，则绕组间的互感可视为 0。

可将磁链方程改写成式 3-3。

$$\begin{pmatrix} \psi_A \\ \psi_B \\ \psi_C \end{pmatrix} = \begin{pmatrix} L_s & 0 & 0 \\ 0 & L_s & 0 \\ 0 & 0 & L_s \end{pmatrix} \begin{pmatrix} i_A \\ i_B \\ i_C \end{pmatrix} + \begin{pmatrix} \psi_f \cos \theta \\ \psi_f \cos(\theta - \frac{2\pi}{3}) \\ \psi_f \cos(\theta + \frac{2\pi}{3}) \end{pmatrix} \quad (3-3)$$

3.1.2 两相静止坐标系下的数学模型

根据图 3-1 可知, 在电机控制的数学建模过程中, 三相坐标系 A-B-C 到静止坐标系 α - β 的转换是通过 Clark 变换实现的, 而从静止坐标系 α - β 回到三相坐标系 A-B-C 则采用反 Clark 变换。通过这种坐标变换, 可以将三相坐标系下的电机数学模型与相关物理量转化为静止坐标系下, 简化电机控制算法的设计与实现。将式 3-1 经过变换得到:

$$\begin{pmatrix} u_\alpha \\ u_\beta \end{pmatrix} = \begin{pmatrix} R_s & 0 \\ 0 & R_s \end{pmatrix} \begin{pmatrix} i_\alpha \\ i_\beta \end{pmatrix} + \frac{d}{dt} \begin{pmatrix} \psi_\alpha \\ \psi_\beta \end{pmatrix} \quad (3-4)$$

其中 u_α 、 u_β 、 i_α 、 i_β 、 ψ_α 、 ψ_β 分别为电压、电流、定子磁链在 α 轴和 β 轴的分量。

同样的, 对磁链方程进行坐标变换, 得到 α - β 坐标系下的定子磁链方程:

$$\begin{pmatrix} \psi_\alpha \\ \psi_\beta \end{pmatrix} = \begin{pmatrix} L_\alpha & 0 \\ 0 & L_\beta \end{pmatrix} \begin{pmatrix} i_\alpha \\ i_\beta \end{pmatrix} + \begin{pmatrix} \psi_f \cos \theta \\ \psi_f \sin \theta \end{pmatrix} \quad (3-5)$$

将式 3-5 代入式 3-4, 定子电压方程可表示为:

$$\begin{pmatrix} u_\alpha \\ u_\beta \end{pmatrix} = \begin{pmatrix} R_s & 0 \\ 0 & R_s \end{pmatrix} \begin{pmatrix} i_\alpha \\ i_\beta \end{pmatrix} + \begin{pmatrix} L_\alpha & 0 \\ 0 & L_\beta \end{pmatrix} \begin{pmatrix} \frac{d}{dt} i_\alpha \\ \frac{d}{dt} i_\beta \end{pmatrix} + \begin{pmatrix} -\omega_e \psi_f \sin \theta \\ \omega_e \psi_f \cos \theta \end{pmatrix} \quad (3-6)$$

3.1.3 两相旋转坐标系下的数学模型

由于电机的结构较为复杂, 为了保证控制算法的精度, 以及实时跟踪转子位置速度信息, 将两相静止坐标系下的数学模型变换到旋转坐标系下实现解耦。与 Clark 变换类似, 将 α - β 坐标变换到 d - q 坐标的过程称为 Park 变换, d - q 坐标变换到 α - β 坐标的过程称为反 Park 变换。将式(3-4)经过变换得到两相旋转坐标系下的电压方程:

$$\begin{pmatrix} u_d \\ u_q \end{pmatrix} = \begin{pmatrix} R_s & 0 \\ 0 & R_s \end{pmatrix} \begin{pmatrix} i_d \\ i_q \end{pmatrix} + \frac{d}{dt} \begin{pmatrix} \psi_d \\ \psi_q \end{pmatrix} - \omega_e \begin{pmatrix} \psi_d \\ \psi_q \end{pmatrix} \quad (3-7)$$

此时定子磁链方程表示为:

$$\begin{pmatrix} \psi_d \\ \psi_q \end{pmatrix} = \begin{pmatrix} L_d & 0 \\ 0 & L_q \end{pmatrix} \begin{pmatrix} i_d \\ i_q \end{pmatrix} + \begin{pmatrix} \psi_f \\ 0 \end{pmatrix} \quad (3-8)$$

其中 u_d 、 u_q 、 i_d 、 i_q 、 ψ_d 、 ψ_q 、 L_d 、 L_q 分别为电压、电流、定子磁链、定子电感在 d 轴和 q 轴的分量。

将式(3-8)带入式(3-7), d - q 坐标系下定子电压方程可表示为:

$$\begin{pmatrix} u_d \\ u_q \end{pmatrix} = \begin{pmatrix} R_s + L_d \frac{d}{dt} & -\omega_e L_q \\ \omega_e L_d & R_s + L_q \frac{d}{dt} \end{pmatrix} \begin{pmatrix} i_d \\ i_q \end{pmatrix} + \omega_e \begin{pmatrix} 0 \\ \psi_f \end{pmatrix} \quad (3-9)$$

d - q 坐标系与电机转子的位置保持一致, 此时可对电机负载能力进行分析, 电机转矩方程为:

$$T_e = \frac{3}{2} n_p (\psi_d i_q - \psi_q i_d) = \frac{3}{2} n_p [\psi_f i_q + (L_d - L_q) i_d i_q] \quad (3-10)$$

对于表贴式永磁同步电机或者凸极性不明显的直流无刷电机, d 轴与 q 轴电感相差较小, 且矢量控制的目标中 $i_d = 0$, 对式 3-10 化简可得到电机转矩方程:

$$T_e = \frac{3}{2} n_p \psi_f i_q \quad (3-11)$$

其中 n_p 、 ψ_f 分别为电机极对数和永磁体磁链。

基于此, 可以看出电机产生的转矩与其 q 轴电流的大小密切相关。因此, 在需要调整电机的负载能力时, 可以通过 PID 控制策略来调节 q 轴电流的大小, 从而达到预期的性能目标。

3.2 磁场定向控制策略与库函数封装

为了能够精准控制电机转子位置、速度、负载能力, 提出了矢量控制的理论, 其本质是将电机中的电流矢量分解为转矩分量和磁场分量, 两个矢量互相垂直, 对其采取不同的控制策略。

对于一个完善的矢量控制系统, 由外环转速环和内环电流环双闭环控制构成, 其中转速环用以控制电机转速, 电流环根据式(3-11)可知, 用以控制电机转矩, 本文的矢量控制采用 $i_d = 0$ 的控制策略, 通过 PID 控制, 使 d 轴的励磁电流分量为0。矢量控制首先采集电机线圈三相电流, 将三相电流经过坐标变换转换成 d - q 轴电流, 通过位置传感器检测转子的位置和速度信息, 将速度值作为速度环反馈, 通过 PI 得到 q 轴电流的参考值, 与检测得到的 d - q 轴反馈电流做 PI 控制, 再通过坐标变换产生 PWM 信号驱动逆变电路, 控制电机。

3.2.1 坐标变换

(1) Clark 变换

将三相坐标系中的物理量转换成静止坐标系下的物理量，这一变换称为 Clark 变换，坐标变化示意图如图 3-2 所示。由图可知，两相坐标系 α - β 中的 α 轴与三相坐标系 A-B-C 的 A 轴重合。同时变换过程中遵循赋值等效原则，即合成的矢量大小和方向相等，在变换过程中需要乘上 $2/3$ ，则 Clark 变换公式为：

$$\begin{bmatrix} I_\alpha \\ I_\beta \end{bmatrix} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} \quad (3-12)$$

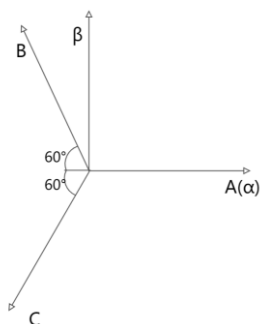


图 3-2 A-B-C 坐标系与 α - β 坐标系关系图

(2) Park 变换

将静止坐标系中的物理量转换成旋转坐标系下的物理量，这一变换称为 Park 变换，坐标变化示意图如图 3-3 所示，图中的角度 θ 为转子当前电角度。

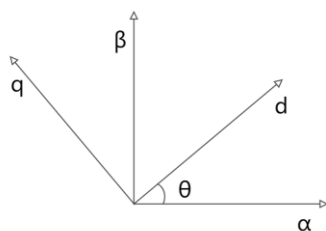


图 3-3 α - β 坐标系与 d - q 坐标系关系图

d - q 轴始终与转子位置一致，同样变换过程中遵循赋值等效原则，即合成的矢量大小和方向相等，Park 变换公式为：

$$\begin{bmatrix} I_d \\ I_q \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} I_\alpha \\ I_\beta \end{bmatrix} \quad (3-13)$$

(3) 反 Park(iPark)变换

将旋转坐标系下的物理量转换成静止坐标系的物理量，这一变换称为 iPark 变换，公式为：

$$\begin{bmatrix} I_\alpha \\ I_\beta \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} I_d \\ I_q \end{bmatrix} \quad (3-14)$$

与 Park 和 iPark 变换类似，Clark 变化也存在与其对应的反变换，对于简单的控制场合下，可以采用反 Clark 变换，将静止坐标系转换成三相坐标系，通过变换后的结果计算 PWM 波形的占空比，驱动电机；而对于实时性强、控制精度要求较高的场景，则采用空间矢量脉宽调制，保证驱动电机的三相电流为正弦信号，减小电机转动时的抖动。

3.2.2 空间矢量脉宽调制算法

空间矢量脉宽调制技术(Space Vector Pulse Width Modulation, SVPWM)：通过精确计算三路 PWM 波的占空比来控制逆变器的开关状态，其开关通断的变化会产生特定的调制波形。经过滤波处理后，这一算法能够生成接近理想正弦波的电流信号，用以驱动电机。这种技术在电力电子驱动领域中被广泛应用，能够提高系统的功率效率和控制精度，同时减少谐波干扰。三相逆变器驱动电路如图 3-4 所示。

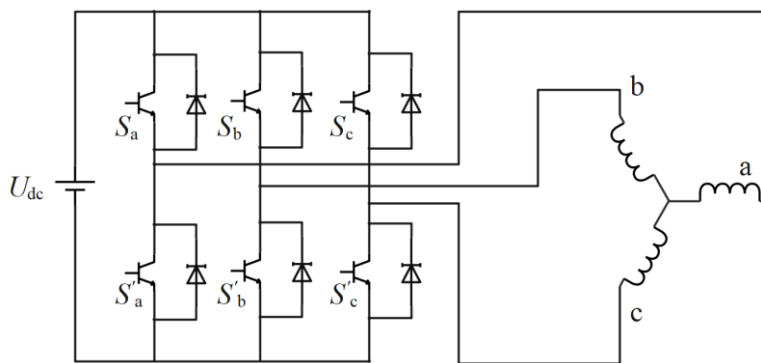


图 3-4 三相逆变器驱动电路图

逆变电路的功能是将直流电转换成交流电，实现不同电流方向的切换，通常利用半桥 MOS 电路实现。MOS 管作为一种快速电子开关，通过在栅极施加高低电平，实现对源极/漏极的导通关闭。图中 S_a 、 S'_a 、 S_b 、 S'_b 、 S_c 、 S'_c 为 6 个桥臂的开关，上桥开通下桥关断定义为状态“1”，上桥关断下桥导通定义为状态“0”。因此三组半桥电路组成 8 种状态，分别为： $U_0(000)$ 、 $U_1(001)$ 、 $U_2(010)$ 、 $U_3(011)$ 、 $U_4(100)$ 、 $U_5(101)$ 、 $U_6(110)$ 、 $U_7(111)$ 。其中 U_0 、 U_7 因无法构成回路，

称为零矢量，剩下 6 个矢量相位间隔 60° ，六个非零矢量将磁场分成六个扇区，各扇区内电压矢量分布如图 3-5 所示。空间矢量脉宽调制(SVPWM)便是利用这六个非零空间电压矢量作为基础向量，通过选取每个扇区内相邻的两个电压矢量及两个零矢量，合成任意方向的电压矢量。

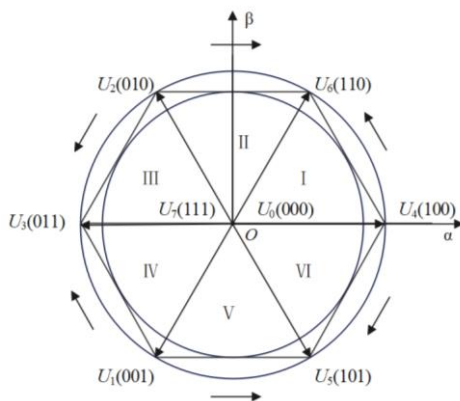


图 3-5 电压空间矢量图

任意电压矢量合成公式为：

$$\int_0^T U_{ref} dt = \int_0^{T_x} U_x dt + \int_{T_x}^{T_x+T_y} U_y dt + \int_{T_x+T_y}^T U_0 dt \quad (3-15)$$

离散后等效为：

$$U_{ref} \cdot T = U_x \cdot T_x + U_y \cdot T_y + U_0 \cdot T_0 \quad (3-16)$$

其中 U_x 、 U_y 、 U_0 为用于合成 U_{ref} 的两个基准矢量和零矢量； T 为 PWM 周期； T_x 、 T_y 、 T_0 为基准矢量和零矢量作用时间。

第一扇区内矢量合成过程如图 3-6 所示。

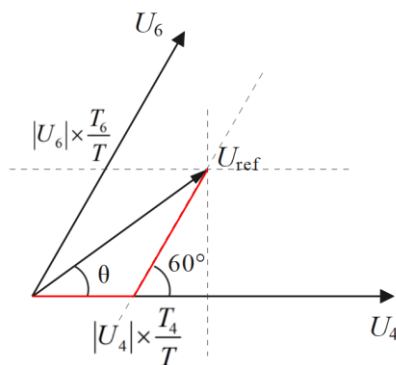


图 3-6 第一扇区电压矢量合成示意图

为实现以上目标，本文采用经典的七段式 SVPWM 技术，即每个扇区以 U_0 矢量作为开始和结束，中间使用 4 段非零基准矢量，两个基准矢量中间插入 U_7 矢量，各矢量作用时间相同。保证每个 PWM 周期内，MOS 管只开关一次，减少

其开关损耗。

以电压矢量 $U_1(001)$ 为例,对空间矢量脉宽调制进行分析,此时C相上桥臂导通,A、B相下桥臂导通,逆变电路输出的电压关系为:

$$\begin{cases} U_{ab} = 0, U_{bc} = -U_{dc}, U_{ca} = U_{dc} \\ U_{cn} - U_{an} = U_{dc}, U_{cn} - U_{bn} = U_{dc} \\ U_{an} + U_{bn} + U_{cn} = 0 \end{cases} \quad (3-17)$$

对上述方程组求解可以得到: $U_{an} = -U_{dc}/3, U_{bn} = -U_{dc}/3, U_{cn} = 2U_{dc}/3$,按照相同方法可以得到八种电压矢量的线电压与相电压关系如表3-1所示。

表3-1 八种空间矢量电压关系

S_a	S_b	S_c	电压 矢量	线电压			相电压		
				U_{ab}	U_{bc}	U_{ca}	U_{an}	U_{bn}	U_{cn}
0	0	0	U_0	0	0	0	0	0	0
0	0	1	U_1	0	$-U_{dc}$	U_{dc}	$-1/3U_{dc}$	$-1/3U_{dc}$	$2/3U_{dc}$
0	1	0	U_2	$-U_{dc}$	U_{dc}	0	$-1/3U_{dc}$	$2/3U_{dc}$	$-1/3U_{dc}$
0	1	1	U_3	$-U_{dc}$	0	U_{dc}	$-2/3U_{dc}$	$1/3U_{dc}$	$1/3U_{dc}$
1	0	0	U_4	U_{dc}	0	$-U_{dc}$	$2/3U_{dc}$	$-1/3U_{dc}$	$-1/3U_{dc}$
1	0	1	U_5	U_{dc}	$-U_{dc}$	0	$1/3U_{dc}$	$-2/3U_{dc}$	$1/3U_{dc}$
1	1	0	U_6	0	U_{dc}	$-U_{dc}$	$1/3U_{dc}$	$1/3U_{dc}$	$-2/3U_{dc}$
1	1	1	U_7	0	0	0	0	0	0

依据上表,当转子运行到不同的扇区内时,可以将其拆解成两个相邻的基准矢量与两个零矢量,根据其在扇区内位置的不同,各基准矢量作用时间也随之变化。不同作用时间会导致对应桥臂的MOS管开关时间发生变化,这一变化通过ePWM外设产生的PWM波的占空比改变,因此只需通过空间矢量脉宽调制算法计算得到三路PWM波在不同时刻所需要的占空比,即可保证永磁同步电机内部的相电压为马鞍波,相电流为正弦波,实现对电机的矢量控制。

3.2.3 矢量控制函数库封装

基于以上分析可知,对永磁同步电机进行矢量控制的核心部分主要包括坐标变换和SVPWM,本论文方案实现基于实验室的DSP芯片,该芯片设计目标是为直流无刷电机、永磁同步电机等,提供一套矢量控制方案。针对这一需求,将矢量控制中的相关算法模块进行封装,以使用户进行调用。

函数库封装主要包括:坐标变换模块、SVPWM模块,不同模块分别设置结构体用于在不同接口中进行调用。函数封装情况如下所示:

(1) 坐标变换模块

FOC控制中所使用的坐标变换包括:Clark变换、Park变换、iPark变换,针

对这三种变换，设置如下所示的结构体。

```

1.  /** three-phase variables in stationary coordinate **/
2.  typedef struct {
3.      float Ia;
4.      float Ib;
5.      float Ic;
6.  } stc_abc_t;
7.
8.  /** two-phase variables in stationary coordinate **/
9.  typedef struct {
10.     float Ialpha;
11.     float Ibeta;
12. } stc_ab_t;
13.
14. /** two phase variables in rotation coordinate **/
15. typedef struct {
16.     float Id;
17.     float Iq;
18.     float sin;
19.     float cos;
20. } stc_dq_t;

```

根据 3.2.1 章节中对坐标变换原理的描述可知，Clark 变换是将三相坐标系下的物理量转换到静止坐标系下，故该变换的输入输出与电机转子角度无关，结构体内仅需包含对应坐标系下的物理量即可；而 Park 变换和 iPark 变换的输入或输出中包含旋转坐标系下的物理量，该变换会随着转子位置改变而变化，故此坐标系下的结构体中包含角度的正弦和余弦值。

因坐标变换模块原理简单，根据 3.2.1 章节中描述的原理即可实现该功能，故在此列出封装后库函数代码，如下所示。

```

1. void Transf_Clarke(stc_abc_t stcab, stc_ab_t *pstcab) {
2.     pstcab->Ialpha = stcab.Ia;
3.     pstcab->Ibeta  = INV_SQRT3 * stcab.Ia + TwoINV_SQRT3 * stcab.Ib;
4. }
5.
6. void Transf_Park(stc_ab_t stcab, stc_dq_t *pstcdq) {
7.     pstcdq->Id = stcab.Ialpha * pstcdq->cos + stcab.Ibeta * pstcdq->sin;
8.     pstcdq->Iq = -stcab.Ialpha * pstcdq->sin + stcab.Ibeta * pstcdq->cos;
9. }
10.
11. void Transf_InvPark(stc_dq_t stcdq, stc_ab_t *pstcab) {
12.     pstcab->Ialpha = stcdq.Id * stcdq.cos - stcdq.Iq * stcdq.sin;
13.     pstcab->Ibeta  = stcdq.Id * stcdq.sin + stcdq.Iq * stcdq.cos;
14. }

```

章节 3.2.1 中介绍的 Clark 变换依赖于三相电流，考虑在实际工程中，根据基尔霍夫电流定律，流过三相线圈的电流和为零，故 C 相电流可由 A 相和 B 相表示；同时根据该定律，可以减少驱动板上对采样电路的需求。基于以上考虑，代码中的 Clark 变换仅使用 i_a 、 i_b ；INV_SQRT3 和 TwoINV_SQRT3 分别表示变换中的常量 $1/\sqrt{3}$ 和 $2/\sqrt{3}$ ；Park 变换和 iPark 变换处理简单，代码仅需根据式(3-13)和(3-14)编写即可。

(2) SVPWM 模块

该模块用于生成三路相同频率不同占空比的 PWM 波，根据 3.2.2 章节可知，完成该模块需要分为三个步骤：1. 确定当前转子所在扇区位置；2. 确定该扇区下需要合成目标空间矢量的基准矢量作用时间；3. 根据基准矢量作用时间确定三路 PWM 占空比。

因为该模块代码较多，在此通过伪代码形式分成三部分展示并加以说明。

1. 扇区确认：

```

1. SVPWM_CalcSector (Va, Vb)
2.   Vbc = Vb * SQRT3;
3.   Vab = Va * 1.5 - Vbc * 0.5;
4.   Vca = -Vab - Vbc;
5.   if Vab >= 0
6.     if Vbc >= 0
7.       Sector = 1;
8.     else
9.       if Vca >= 0
10.        Sector = 5;
11.      else
12.        Sector = 6;
13.    else
14.      if Vbc <= 0
15.        Sector = 4;
16.      else
17.        if Vca <= 0
18.          Sector = 2;
19.        else
20.          Sector = 3;

```

转子所在扇区由 iPark 变换的输出量 u_α 、 u_β 判断，假设电压矢量落在第 I 扇区，则电压 u_α 、 u_β 关系为：

$$0 < \arctan(u_\beta / u_\alpha) < \pi / 3$$

据此可知，电压矢量在哪一扇区由 u_β 、 $\sqrt{3}/2u_\alpha - 1/2u_\beta$ 、 $-\sqrt{3}/2u_\alpha - 1/2u_\beta$ 共同决定，故令：

$$\begin{aligned}
 u_{bc} &= \sqrt{3}u_{\beta} \\
 u_{ab} &= \frac{3}{2}u_{\alpha} - \frac{1}{2}u_{bc} \\
 u_{ca} &= -u_{ab} - u_{bc}
 \end{aligned} \tag{3-18}$$

根据中间变量 u_{bc} 、 u_{ab} 、 u_{ca} 的正负性即可判断所在扇区。

2. 基准矢量作用时间:

```

1. SVPWM_CalcDuration(V1, V2, Vbus, PeakCnt)
2.     T1 = V1 / Vbus * PeakCnt;
3.     T2 = V2 / Vbus * PeakCnt;
4.     if T1 + T2 > PeakCnt
5.         T1 = T1 / (T1 + T2) * PeakCnt;
6.         T2 = T2 / (T1 + T2) * PeakCnt;
7.     T0 = PeakCnt - T1 - T2;

```

因为各扇区内的电压矢量由两个零矢量和两个相邻的基准矢量合成得到，故作用时间 $T1$ 、 $T2$ 分别为基准矢量持续时间， $T0$ 为零矢量持续时间。代码中的 $PeakCnt$ 表示矢量总持续时间，该时间由 PWM 波的计数值决定， $V1$ 、 $V2$ 由经过扇区确认后的 u_{bc} 、 u_{ab} 、 u_{ca} 得到，因为两相之间电压均由 $Vbus$ （母线电压）得到，故需根据其在目前电压中的占比决定基准矢量持续时间，当两基准矢量时间超过 PWM 总时间时，需要根据比例调整时间大小，避免控制出错。

3. 计算 PWM 占空比:

```

1. SVPWM_7SegPwm(Sector, T0, T1, T2)
2.     if Sector = 1
3.         Uon = T0 >> 1;
4.         Von = Uon + T1;
5.         Won = Von + T2;
6.     else if Sector = 2
7.         Von = T0 >> 1;
8.         Uon = Von + T1;
9.         Won = Uon + T2;
10.    else if Sector = 3
11.        Von = T0 >> 1;
12.        Won = Von + T1;
13.        Uon = Won + T2;
14.    else if Sector = 4
15.        Won = T0 >> 1;
16.        Von = Won + T1;
17.        Uon = Von + T2;
18.    else if Sector = 5
19.        Won = T0 >> 1;
20.        Uon = Won + T1;
21.        Von = Uon + T2;

```

```

22.   else if Sector = 6
23.       Uon = T0 >> 1;
24.       Won = Uon + T1;
25.       Von = Won + T2;
    
```

以第一扇区为例，该扇区内的矢量有两个基准矢量 U_4 、 U_6 和两个零矢量 U_0 、 U_7 组成，为减少开关损耗，七段式 SVPWM 作用顺序为 $U_0 - U_4 - U_6 - U_7 - U_6 - U_4 - U_0$ ，由于发波要求对称，两个零矢量被平均分成两段，如图 3-7 所示。

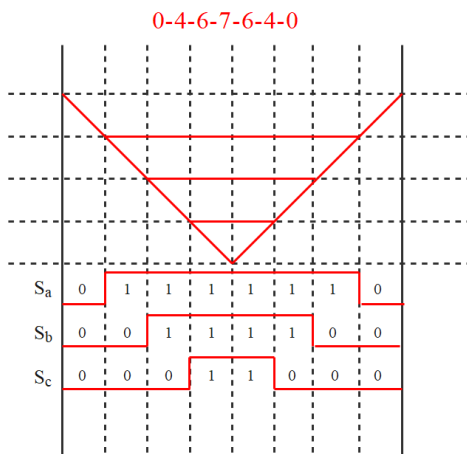


图 3-7 第一扇区电压矢量切换时间

同时需要注意，当转子处于不同扇区时， T_0 时间所作用的线圈也不相同，故伪代码中根据扇区不同，U、V、W 三相 PWM 占空比的值也不相同。

3.2.4 磁场定向控制流程

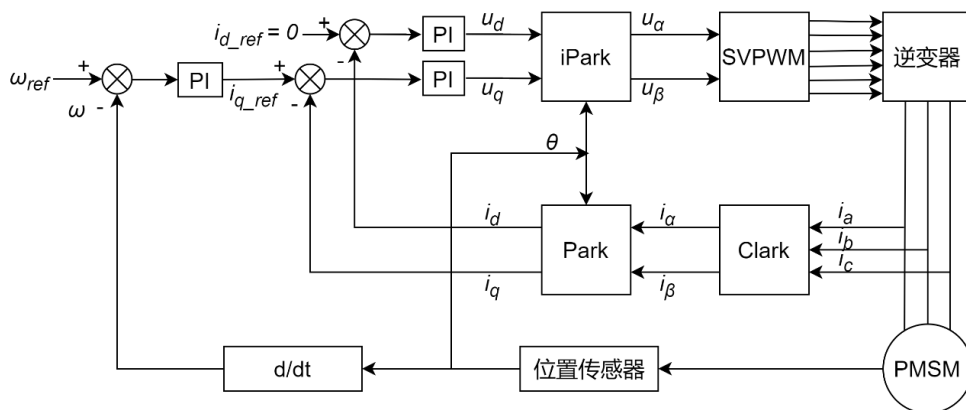


图 3-8 磁场定向控制框图

磁场定向控制(FOC)控制框图如图 3-8 所示，包括双闭环控制，坐标变换模块、PI 控制、SVPWM 模块和角度速度检测。硬件电流采集模块采集三相电流，经过 Clark、Park 变换，得到旋转坐标系下的电流 i_d 、 i_q ；再根据 d 轴和 q 轴控

制要求进行 PI 计算和 iPark 变换，将结果输入到 SVPWM 模块，产生 3 路 PWM 波驱动电机转动，实现矢量控制。FOC 具体控制流程为：

- 1) 利用 ADC 采集电机三相坐标系下电流得到 i_a 、 i_b 、 i_c ；
- 2) 将电机三相坐标系下电流 i_a 、 i_b 、 i_c 经过 Clark 变换得到静止坐标系下电流 i_α 、 i_β ；
- 3) 将静止坐标系下电流 i_α 、 i_β 经过 Park 变换得到旋转坐标系下电流 i_d 、 i_q ；
- 4) 计算 i_d 、 i_q 和其设定值 i_{d_ref} 、 i_{q_ref} 的差值；
- 5) 将上述误差输入两个 PI 控制器，得到 u_d 、 u_q ；
- 6) 将 u_d 、 u_q 经过 iPark 变换得到 u_α 、 u_β ；
- 7) 将 u_α 、 u_β 输入到 SVPWM 模块进行调制，控制三相逆变器的 MOS 管开关，驱动电机；
- 8) 重复上述步骤。

为配合 FOC 中的软件计算，需要根据芯片的相关外设编写相应的驱动程序，电机控制中需要的外设配置及功能如下所示：

- 1) ePWM 外设配置：
 - a) 配置 Up-Down 计数模式，用于生成对称互补的 PWM；
 - b) 配置死区时间；
 - c) 配置 PWM 计数值中的某一设定值产生触发事件，触发 ADC 采样。
- 2) ADC 配置
 - a) 配置电流采样和电源电压采样，触发源选择 PWM 计数器的设定值；
 - b) 配置 ADC 中断，PWM 设定值触发 ADC 采样，采样完成进入中断。
- 3) 中断定时器配置
 - a) 配置 1ms 周期定时器，并配置中断定时器，用于运行电机状态机；
 - b) 配置 10ms 定时器，用于用户功能定义（包括速度值的设置等）。

通过以上配置实现对目标电机的矢量控制，使电机在速度环、电流环双闭环控制下，按照设定的速度和力矩转动。

3.3 中高速无感控制方案

为了实现对永磁同步电机的高精度控制，常见的做法是使用机械传感器，如霍尔、编码器等。这些传感器虽然能提高控制精度，但同时也会增加控制系统的成本，并且在复杂的环境中使用，由于易受外界因素的影响，其可靠性可能会受到限制。因此，采用无位置传感器的控制技术成为了一种可行的替代方案。该技术通过测量电机的电压电流等物理量，能够估计出转子的当前位置和速度信息，

从而实现对电机的高性能控制。

在无感矢量控制中，根据速度的不同，通常需要采用不同的控制策略，这些策略大致可以分为低速域和中高速域的控制算法。本节所研究的内容主要是在中高速域，通过使用非线性磁链观测器来观测电机的磁链信息，从而准确地获得转子的位置和速度信息。

3.3.1 磁链观测器原理

磁链观测器是依托电机的磁链信息进行工作，由于磁链的幅值通常保持稳定，即便在较低的转速下，其幅值也不会明显减小。这一特性使得在电机的中高速运行阶段，仍能设定较低的转速阈值进行控制。状态观测器^[44]则基于系统的外部变量（包括输入和输出变量）的实际测量值，推导出系统状态变量的估计值。状态观测器方程通常是由电机的运行方程转换而来，结合了观测量及其导数，从而实现对电机状态的实时估计和监控。

非线性磁链观测器利用电机在静止坐标系 α - β 下的电压方程推导得到，电机的电压方程为：

$$\begin{bmatrix} u_\alpha \\ u_\beta \end{bmatrix} = \begin{bmatrix} R_s + pL_d & \omega_e(L_d - L_q) \\ -\omega_e(L_d - L_q) & R_s + pL_q \end{bmatrix} \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} + \begin{bmatrix} E_\alpha \\ E_\beta \end{bmatrix} \quad (3-19)$$

其中 u_α 、 u_β 、 i_α 、 i_β 、 E_α 、 E_β 为 α - β 轴电压、电流、反电动势； L_d 、 L_q 为 d - q 轴定子电感； R_s 为定子电阻； ω_e 为电角速度； p 为微分算子。

电机在两相静止坐标系下的反电动势与转子的角度、磁链等信息有关，其表达式为：

$$\begin{bmatrix} E_\alpha \\ E_\beta \end{bmatrix} = \begin{bmatrix} (L_d - L_q)(\omega_e i_d - p i_q) + \omega_e \psi_f \\ \omega_e \psi_f \end{bmatrix} \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix} \quad (3-20)$$

ψ_f 为永磁体磁链； θ 为电角度。

对于表贴式永磁同步电机或者凸极性不明显的直流无刷电机来说，其 d 轴与 q 轴电感近似，可以认为 $L_d = L_q = L_s$ ，再将式(3-20)带入(3-19)重写 α - β 轴电压方程可以得到：

$$\begin{bmatrix} u_\alpha \\ u_\beta \end{bmatrix} = \begin{bmatrix} R_s + pL_s & 0 \\ 0 & R_s + pL_s \end{bmatrix} \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} + \omega_e \psi_f \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix} \quad (3-21)$$

之后对电机在两相旋转坐标系下的磁链进行分析，磁链方程为：

$$\begin{bmatrix} \psi_d \\ \psi_q \end{bmatrix} = \begin{bmatrix} L_s i_d + \psi_f \\ L_s i_q \end{bmatrix} \quad (3-22)$$

依据 iPark 变换公式可以将 d - q 轴磁链方程转化为 α - β 轴磁链方程：

$$\begin{bmatrix} \psi_\alpha \\ \psi_\beta \end{bmatrix} = L_s \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} \psi_f \cos \theta \\ \psi_f \sin \theta \end{bmatrix} \quad (3-23)$$

结合电流的 iPark 变换公式(3-14)，带入式(3-23)得到 α - β 轴磁链方程为：

$$\begin{bmatrix} \psi_\alpha \\ \psi_\beta \end{bmatrix} = \begin{bmatrix} L_s i_\alpha + \psi_f \cos \theta \\ L_s i_\beta + \psi_f \sin \theta \end{bmatrix} \quad (3-24)$$

这里将旋转坐标系下的磁链方程转换到静止坐标系下是考虑到，在矢量控制流程中，进行 Park 变换需要利用电机转子的位置信息，如果在旋转坐标系下构建状态观测器会使设计更加复杂。

在确定静止坐标系下的磁链方程后，定义状态观测器的状态变量和输出变量，因为磁链观测器是对电机磁链进行观测，所以将式(3-24)定义为状态变量：

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \psi_\alpha \\ \psi_\beta \end{bmatrix} = \begin{bmatrix} L_s i_\alpha + \psi_f \cos \theta \\ L_s i_\beta + \psi_f \sin \theta \end{bmatrix} \quad (3-25)$$

关注转子位置信息，对式(3-25)移项得到：

$$\psi_f \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} = \begin{bmatrix} x_1 - L_s i_\alpha \\ x_2 - L_s i_\beta \end{bmatrix} \quad (3-26)$$

依据状态观测器定义，由观测量和其微分构成，所以对状态变量 x 进行求导，可以得到：

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} L_s p i_\alpha - \omega_e \psi_f \sin \theta \\ L_s p i_\beta + \omega_e \psi_f \cos \theta \end{bmatrix} \quad (3-27)$$

将电机电压方程(3-21)带入上式，整理后可以看出，状态变量的微分由 α - β 轴电压、电流和定子电阻计算得到，其关系为：

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} u_\alpha - R_s i_\alpha \\ u_\beta - R_s i_\beta \end{bmatrix} \quad (3-28)$$

之后定义状态观测器的输出变量 y ，建立起输入变量、输入变量微分、输出变量三者之间的关系。

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} u_\alpha - R_s i_\alpha \\ u_\beta - R_s i_\beta \end{bmatrix} = \dot{x} \quad (3-29)$$

考虑构建非线性磁链观测器方程，在对反电动势进行积分运算，获得磁链的过程中，为避免直流偏置或积分漂移带来的影响，通常使用高通滤波器或者自适应补偿等方法进行抑制。构建非线性处理模块的思路就是把估算的磁链幅值与实际磁链的幅值的差，作为估算的磁链分量的补偿。

为构建非线性磁链观测器，取估计的状态变量为矢量函数：

$$\eta(\hat{x}) = \begin{bmatrix} \eta(\hat{x}_1) \\ \eta(\hat{x}_2) \end{bmatrix} = \begin{bmatrix} \hat{x}_1 - L_s i_\alpha \\ \hat{x}_2 - L_s i_\beta \end{bmatrix} \quad (3-30)$$

其中 $\|\eta(\hat{x})\|^2 = \eta(\hat{x}_1)^2 + \eta(\hat{x}_2)^2 = \psi_f^2$ 。

考虑构建观测器模型为：

$$\dot{\hat{x}} = y + \frac{\gamma}{2} \eta(\hat{x}) \left[\psi_f^2 - \|\eta(\hat{x})\|^2 \right] \quad (3-31)$$

γ 为观测器增益，由此可以看出，利用非线性磁链观测器获取电机转子位置和速度信息，在调试过程中仅需要调整增益 γ 即可。

据此，根据电机磁链、输入变量观测值、 α - β 轴电流以及电机定子电阻的关系，可以得到转子的角度信息：

$$\begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} = \frac{1}{\psi_f} \begin{bmatrix} \hat{x}_1 - L_s i_\alpha \\ \hat{x}_2 - L_s i_\beta \end{bmatrix} \quad (3-32)$$

通过反正切计算，得到转子角度的计算公式为：

$$\theta = \hat{\theta} = \arctan \left(\frac{\hat{x}_2 - L_s i_\beta}{\hat{x}_1 - L_s i_\alpha} \right) \quad (3-33)$$

将式(3-30)带入(3-31)，进过整理可以状态观测器迭代公式为：

$$\begin{bmatrix} \dot{\hat{x}}_1 \\ \dot{\hat{x}}_2 \end{bmatrix} = \begin{bmatrix} y_1 + \frac{\gamma}{2} (\hat{x}_1 - L_s i_\alpha) \left[\psi_f^2 - (\hat{x}_1 - L_s i_\alpha)^2 - (\hat{x}_2 - L_s i_\beta)^2 \right] \\ y_2 + \frac{\gamma}{2} (\hat{x}_2 - L_s i_\beta) \left[\psi_f^2 - (\hat{x}_1 - L_s i_\alpha)^2 - (\hat{x}_2 - L_s i_\beta)^2 \right] \end{bmatrix} \quad (3-34)$$

考虑 DSP 进行软件计算的过程中，无法对状态变量进行求导处理，故对上式进行离散化计算，得到：

$$\begin{bmatrix} \frac{\hat{x}_1(k+1) - \hat{x}_1(k)}{T} \\ \frac{\hat{x}_2(k+1) - \hat{x}_2(k)}{T} \end{bmatrix} = \begin{bmatrix} y_1(k) + \frac{\gamma}{2} (\hat{x}_1(k) - L_s i_\alpha) \left[\psi_f^2 - (\hat{x}_1(k) - L_s i_\alpha)^2 - (\hat{x}_2(k) - L_s i_\beta)^2 \right] \\ y_2(k) + \frac{\gamma}{2} (\hat{x}_2(k) - L_s i_\beta) \left[\psi_f^2 - (\hat{x}_1(k) - L_s i_\alpha)^2 - (\hat{x}_2(k) - L_s i_\beta)^2 \right] \end{bmatrix} \quad (3-35)$$

式中 T 为 PWM 周期

最后将式(3-29)带入整理合并，得到非线性磁链观测器，状态变量迭代计算公式：

$$\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} = \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + T \begin{bmatrix} u_\alpha - R_s i_\alpha + \frac{\gamma}{2} (\hat{x}_1 - L_s i_\alpha) \left[\psi_f^2 - (\hat{x}_1 - L_s i_\alpha)^2 - (\hat{x}_2 - L_s i_\beta)^2 \right] \\ u_\beta - R_s i_\beta + \frac{\gamma}{2} (\hat{x}_2 - L_s i_\beta) \left[\psi_f^2 - (\hat{x}_1 - L_s i_\alpha)^2 - (\hat{x}_2 - L_s i_\beta)^2 \right] \end{bmatrix} \quad (3-36)$$

利用非线性磁链观测器对电机转子磁链矢量进行观测，获取转子位置信息，这种中高速无感控制方法与电机其他参数关联性较小，观测精度更高。

3.3.2 基于 PLL 的磁链观测器设计

通过式(3-33)可知, 当完成对非线性磁链观测器的状态变量观测之后, 计算矢量函数的反正切值即可得到转子的位置信息, 对位置做微分处理, 可以得到速度信息, 进而实现电机矢量控制中的速度闭环控制。在对电机控制精度要求不高的情况, 可以利用上一章设计的反正切硬件指令进行加速计算, 完成双闭环控制。

然而使用反正切函数计算转子位置时, 观测器估计的磁链中的噪声和低频振荡会直接影响计算结果, 尤其是在执行除法运算时, 这些低频振荡的误差可能会被放大, 从而导致较大的角度估计误差。为了避免这一问题, 本文采用锁相环策略, 用以精准估计位置和速度信息。通过将非线性磁链观测器集成到锁相环中, 提高无感估计性能和准确性。

锁相环结构框图如图 3-9 所示。

采用锁相环计算转子位置信息, 首先引入和差公式:

$$\sin(\theta - \hat{\theta}) = \sin\theta \cos\hat{\theta} - \cos\theta \sin\hat{\theta} \quad (3-37)$$

当估算的角度和实际角度一致时, 上式等于 0, 此时有:

$$\sin(\theta - \hat{\theta}) = \sin\theta \cos\hat{\theta} - \cos\theta \sin\hat{\theta} = \frac{1}{\psi_f} \eta(\hat{x}_2) \cos\hat{\theta} - \frac{1}{\psi_f} \eta(\hat{x}_1) \sin\hat{\theta} \quad (3-38)$$

把角度误差作为 PI 控制器的输入, 当角度估计值接近真实角度时, 误差不断趋近于 0, 最终锁定实际电角度。同时为了减少计算量, 不在输入端除以 ψ_f , 而是通过调节 PI 控制器的参数实现得到。通过 PI 控制器得到转子角速度, 进而积分得到角度, 此时角度因为积分操作一直累加, 需要将其限制在 $(-\pi, \pi)$ 范围内。

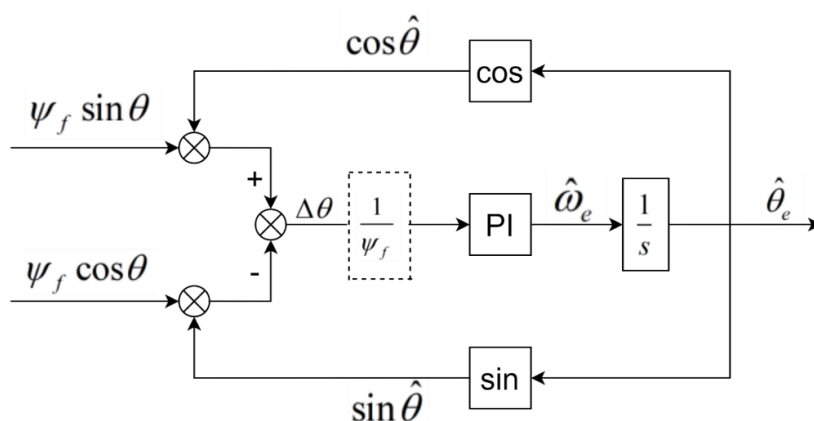


图 3-9 锁相环结构框图

3.3.3 中高速无感控制流程

中高速下无位置传感器矢量控制，采用非线性磁链观测器方案，其结构框图如图 3-10 所示。将观测器计算后的结果经过锁相环处理，得到电机转子位置和速度信息。

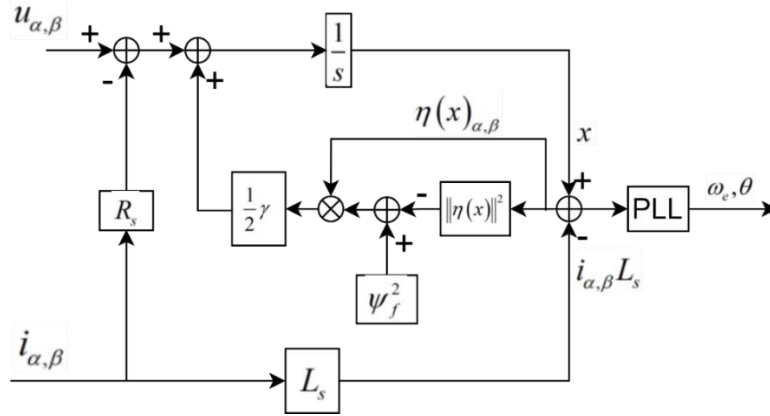


图 3-10 非线性磁链观测器结构框图

中高速域矢量控制流程如图 3-11 所示。矢量控制部分与图 3-8 所介绍内容相同，不同的是对于转子速度和角度的获取不在通过机械传感器获得，而是利用图 3-10 所展示的非线性磁链观测器得到。

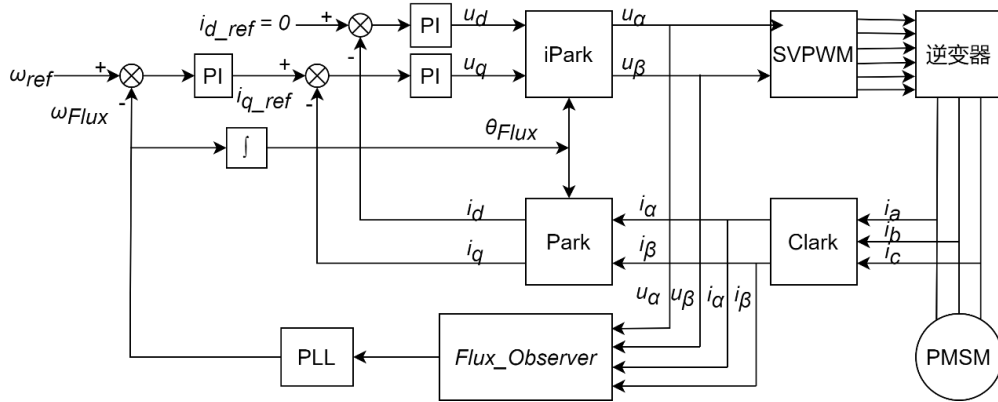


图 3-11 基于磁链观测器的无传感器矢量控制框图

通过图中可以看到，磁链观测器计算的输入为 α - β 轴电压、电流 u_α 、 u_β 、 i_α 、 i_β 。其中 i_α 、 i_β 根据 ADC 采样得到的三相电流 i_a 、 i_b 、 i_c 经过 Clark 变换得到。 u_α 、 u_β 有两种获取方式，图中所演示的是，将前一次矢量控制中 iPark 变换的输出，作为这一次矢量控制中磁链观测器的输入，另一种方法是在硬件设计采样电路，分别利用 ADC 得到三相电压 u_a 、 u_b 、 u_c ，同样经过 Clark 变换得到 u_α 、 u_β 。这两种方法各有其优缺点，前者不需要增加采样电路，但因为是计算得到的电压，与真实电压存在偏差，导致磁链观测器得到位置精度不如后者；第二种则

增加了硬件电路成本。关于 u_α 、 u_β 的获取需要根据实际使用情况做考虑,两种方法得到的结果对于中高速域无位置传感器电机矢量控制影响较小。该优化方案性能在论文第 5 章中通过搭建全速域矢量控制系统进行测试。

3.4 本章小结

本章首先介绍了电机的分类及其运行原理,随后介绍了不同坐标系下的电机数学模型与 Clark 变换、Park 变化、iPark 变换,在此基础上阐述了 SVPWM 与经典矢量控制系统的结构与处理流程,并设计了 FOC 计算的相关结构体与基本的库函数。最后针对中高速域下的无位置传感器控制,介绍了非线性磁链观测器这一无感位置方案,对于计算处理中的高频抖振问题,提出了锁相环的处理方法,提高了研究速度下,电机控制的稳定性。

第4章 低速域高频注入控制方案

在第三章讨论的电机无感控制方案中，所采用的磁链观测器，一般适用于电机的中高速运行场景。在低转速状态下，由于电流峰值较小，观测器所计算的结果往往不够稳定，难以准确捕获转子的位置和速度信息。因此当电机运行在低速区域时，控制策略常采用高频信号注入法^[45-47]。这一方法通过向电机的 d - q 轴注入高频信号，然后利用 ADC 采样，从电机响应的电流中提取包含转子位置信息的信号，并通过特定的调制和解调过程，最终估算转子的位置和速度。

高频信号注入法主要分为两种：（1）旋转高频电压信号注入^[48-50]；（2）脉振高频电压信号注入^[51-53]。本章所采用的方案为脉振高频注入，该方法利用电机的非线性饱和凸极性，可以适用于直流无刷电机、表贴式永磁同步电机等。同时针对传统高频注入法的不足，优化控制中的滤波器、PI 控制器以及调制函数，使包含转子信息的高频信号更易被分离出来，从而保证该方法估计的角度与速度值更加准确。

4.1 非线性饱和凸极性原理

PMSM 的凸极效应是指电抗对电机性能影响的现象，即旋转坐标系 d - q 轴上的电枢反应，通常情况下，电机的凸极率由交轴电感 L_q 和直轴电感 L_d 的比值表示。根据 PMSM 内永磁体安装位置的不同，其凸极特性分为结构性的凸极与饱和性的凸极两种。结构性的凸极指 PMSM 内永磁体安装转子和定子的空隙中导致的 d - q 轴电感不相等，所体现出来的凸极性；饱和性的凸极指通过外在的某些方式让 d - q 轴电感不相等，所体现出来的凸极性。

对于表贴式 PMSM，其直轴电感和交轴电感相等，自身没有凸极性，不适用于高频注入方法，但可以向 d 轴注入高频信号，使其电感发生变化，表现出饱和凸极性。电感 L 与磁链 ψ 和电流 i 的关系为：

$$L_d = \frac{d\psi}{di_d} \quad (4-1)$$

根据式(4-1)可以得到如图 4-1 所示的直轴磁路特性曲线。

对于表贴式永磁同步电机而言，磁场结构由定子电流和永磁体各自产生的磁场共同构成。当使用 $i_d = 0$ 的控制方案时， d 轴的工作状态位于图中 A 点（线性区与饱和区交接点），假设此时永磁体的磁链与 d 轴保持一致，且为定值，可将

其视为由励磁电流 i_f 产生的磁链。由于 q 轴在结构上不包含永磁体，磁场特性与励磁电流无关。当沿 d 轴方向注入与其正方向一致的高频信号时，由于高频信号的磁场与永磁体磁场方向相同，导致 d 轴磁路饱和，电感降低，工作状态移动至图中 C 点；相反，若注入反方向信号，工作状态则移动到图中 B 点。

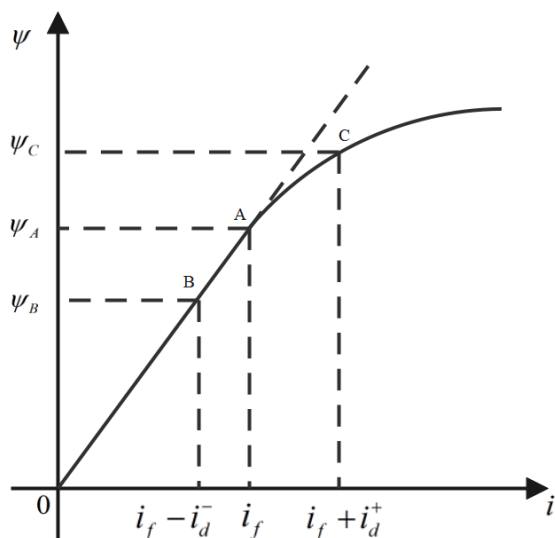


图 4-1 直轴磁路 $\psi - i$ 特性曲线

综上所述，向表贴式永磁同步电机的 d 轴注入不同方向的高频信号，均会使其电感大小发生变化，这一现象称为非线性饱和凸极性。通过调节注入的高频信号，使 d 轴与 q 轴电感不相等，最终在 q 轴的响应电流中提取出转子的位置和速度，实现对电机在低转速下的矢量控制。

4.2 脉振高频电压信号注入

脉振高频电压信号注入是将高频信号添加到基波信号之中，对采集得到的响应信号进行调制与解调处理，估算出转子的位置和速度信息。脉振高频注入的步骤主要包括信号注入、信号提取以及信号解调，信号注入通常是选择高频正弦信号；信号提取采用带通滤波器、低通滤波器等数字滤波器算法；信号解调一般是根据电机机械方程来构建相应的观测器对转子的位置进行估算。

4.2.1 脉振高频电压信号注入原理

当 PMSM 工作在低转速下时，利用电机的凸极性获取转子位置和速度信息，重写 d-q 轴电机电压方程：

$$\begin{bmatrix} u_d \\ u_q \end{bmatrix} = \begin{bmatrix} R_s & 0 \\ 0 & R_s \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} p\psi_d \\ p\psi_q \end{bmatrix} + \omega_e \begin{bmatrix} -\psi_q \\ \psi_d \end{bmatrix} \quad (4-2)$$

其中 u_d 、 u_q 、 i_d 、 i_q 、 L_d 、 L_q 、 ψ_d 、 ψ_q 分别为 d - q 轴下电压、电流、电感、磁链； R_s 为定子电阻； ω_e 为转子转速； p 为微分算子。

式(4-2)中的磁链方程为：

$$\begin{bmatrix} \psi_d \\ \psi_q \end{bmatrix} = \begin{bmatrix} L_d i_d + \psi_f \\ L_q i_q \end{bmatrix} \quad (4-3)$$

其中 ψ_f 为永磁体磁链。

将式(4-3)带入(4-2)中整理得到：

$$\begin{bmatrix} u_d \\ u_q \end{bmatrix} = \begin{bmatrix} R_s & 0 \\ 0 & R_s \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} L_d & 0 \\ 0 & L_q \end{bmatrix} \begin{bmatrix} p i_d \\ p i_q \end{bmatrix} + \omega_e \begin{bmatrix} -L_q i_q \\ L_d i_d + \psi_f \end{bmatrix} \quad (4-4)$$

高频注入法估计的 \hat{d} - \hat{q} 轴与电机实际 d - q 轴的位置关系如图 4-2 所示。

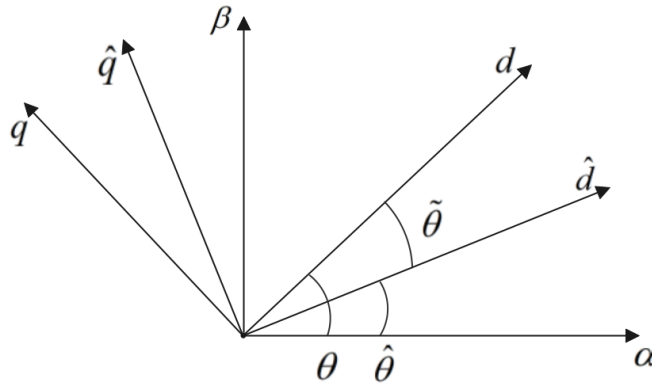


图 4-2 估计 \hat{d} - \hat{q} 轴与实际 d - q 轴位置关系图

其中 $\hat{\theta}$ 、 θ 分别为估计 \hat{d} - \hat{q} 轴和实际 d - q 轴与 α - β 轴的角度； $\tilde{\theta}$ 为角度误差。

对式(4-4)变形可以得到：

$$\begin{cases} \frac{d}{dt} i_d = \frac{1}{L_d} (-R_s i_d + u_d + \omega_e L_q i_q) \\ \frac{d}{dt} i_q = \frac{1}{L_q} (-R_s i_q + u_q - \omega_e L_d i_d - \omega_e \psi_f) \end{cases} \quad (4-5)$$

此时考虑到，当注入高频电压信号时，信号频率远高于基波频率，式(4-5)中电阻产生的阻抗远小于电感产生的阻抗，对于定子电阻 R_s 可以忽略不计；同时采用高频注入法解决的是零速和低速条件下的转子位置估计，转速 ω_e 同样可以忽略不计。所以在注入高频信号之后， d - q 轴下的电压方程可以简写为：

$$\begin{bmatrix} \frac{d}{dt} i_d \\ \frac{d}{dt} i_q \end{bmatrix} = \begin{bmatrix} \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \end{bmatrix} \begin{bmatrix} u_d \\ u_q \end{bmatrix} \quad (4-6)$$

化简式(4-5)是为了得到旋转坐标系下的高频电压信号方程,它描述了高频电压信号与高频电流信号之间的物理关系。因为程序运行过程中并不清楚实际 d - q 轴位置,只有算法收敛,即估计 \hat{d} - \hat{q} 轴与实际 d - q 轴重合时,才能得到转子真实位置。所以在进行高频注入的过程中,只能在估计的 \hat{d} - \hat{q} 轴上进行注入,但实际产生作用是在 d - q 轴上,即对于注入的高频的电压信号 u_d 、 u_q 进行坐标变换。

$$\begin{bmatrix} \frac{d}{dt} \hat{i}_d \\ \frac{d}{dt} \hat{i}_q \end{bmatrix} = \begin{bmatrix} \cos \tilde{\theta} & -\sin \tilde{\theta} \\ \sin \tilde{\theta} & \cos \tilde{\theta} \end{bmatrix} \begin{bmatrix} \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \end{bmatrix} \begin{bmatrix} \cos \tilde{\theta} & \sin \tilde{\theta} \\ -\sin \tilde{\theta} & \cos \tilde{\theta} \end{bmatrix} \begin{bmatrix} \hat{u}_d \\ \hat{u}_q \end{bmatrix} \quad (4-7)$$

其中 \hat{i}_d 、 \hat{i}_q 为估计的两相旋转坐标系 \hat{d} - \hat{q} 轴的高频电流分量, $\tilde{\theta}$ 为算法估计的角度误差。

对式(4-7)整理计算得到:

$$\begin{bmatrix} \frac{d}{dt} \hat{i}_d \\ \frac{d}{dt} \hat{i}_q \end{bmatrix} = \begin{bmatrix} \frac{\cos \tilde{\theta} \cos \tilde{\theta}}{L_d} + \frac{\sin \tilde{\theta} \sin \tilde{\theta}}{L_q} & \frac{\sin \tilde{\theta} \cos \tilde{\theta}}{L_d} - \frac{\sin \tilde{\theta} \cos \tilde{\theta}}{L_q} \\ \frac{\sin \tilde{\theta} \cos \tilde{\theta}}{L_d} - \frac{\sin \tilde{\theta} \cos \tilde{\theta}}{L_q} & \frac{\sin \tilde{\theta} \sin \tilde{\theta}}{L_d} + \frac{\cos \tilde{\theta} \cos \tilde{\theta}}{L_q} \end{bmatrix} \begin{bmatrix} \hat{u}_d \\ \hat{u}_q \end{bmatrix} \quad (4-8)$$

将式(4-8)用平均电感和半差电感化简,可以得到:

$$\begin{bmatrix} \frac{d}{dt} \hat{i}_d \\ \frac{d}{dt} \hat{i}_q \end{bmatrix} = \frac{1}{\bar{L}^2 - \Delta L^2} \begin{bmatrix} \bar{L} + \Delta L \cos 2\tilde{\theta} & \Delta L \sin 2\tilde{\theta} \\ \Delta L \sin 2\tilde{\theta} & \bar{L} - \Delta L \cos 2\tilde{\theta} \end{bmatrix} \begin{bmatrix} \hat{u}_d \\ \hat{u}_q \end{bmatrix} \quad (4-9)$$

其中, \bar{L} 是平均电感,即 $\bar{L} = \frac{L_d + L_q}{2}$; ΔL 是半差电感,即 $\Delta L = \frac{L_d - L_q}{2}$ 。

根据非线性饱和凸极性分析可知,脉振注入法是向估计的旋转坐标系的 \hat{d} 轴注入信号,即:

$$\begin{bmatrix} \hat{u}_d \\ \hat{u}_q \end{bmatrix} = \begin{bmatrix} u_h \cos \omega_h t \\ 0 \end{bmatrix} \quad (4-10)$$

其中, u_h 、 ω_h 分别为注入的电压信号的幅值与频率。

将(4-10)带入式(4-9)化简并进行积分运算可以得到:

$$\begin{cases} \hat{i}_d = \frac{u_h \sin \omega_h t}{\omega_h (\bar{L}^2 - \Delta L^2)} (\bar{L} + \Delta L \cos 2\tilde{\theta}) \\ \hat{i}_q = \frac{u_h \sin \omega_h t}{\omega_h (\bar{L}^2 - \Delta L^2)} \Delta L \sin 2\tilde{\theta} \end{cases} \quad (4-11)$$

通过式(4-11)可知，当 \hat{d} 轴与 \hat{q} 轴电感不相等时，半差电感 $\Delta L \neq 0$ ，在估计的旋转坐标系下 \hat{i}_d 、 \hat{i}_q 都包含估计的角度误差 $\tilde{\theta}$ 。当估计误差接近零时， \hat{q} 轴的响应电流也趋近于零，故将该电流作为位置观测器的输入。通过调整观测器，使响应电流趋于零，此时估计的转子位置便与实际转子磁极位置重合，从而得到转子的位置信息。

4.2.2 转子位置提取

为了得到转子位置信息，向 d 轴注入高频电压信号，对 q 轴响应电流中包含的位置误差信息进行处理。具体处理流程为，将 q 轴电流与相同频率的正弦信号相乘，然后利用锁相环(PLL)对转子的误差信号进行调整，从而得到角度误差函数 $f(\tilde{\theta})$ 。图 4-3 展示了转子位置观测器的处理流程图。

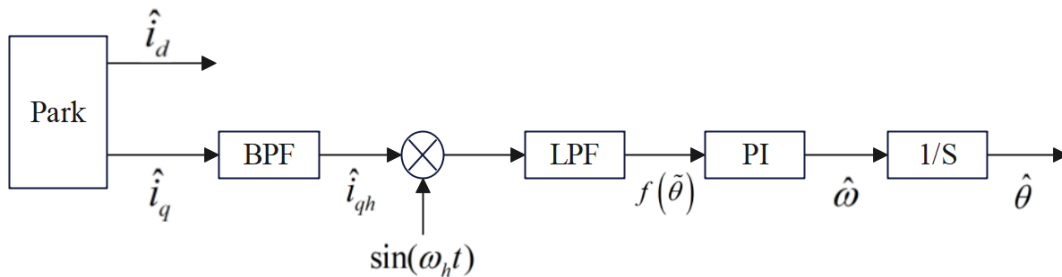


图 4-3 转子位置观测器流程图

从图可知，为获得转子位置信息，高频信号注入法中主要包括信号提取，信号解调和 PI 控制器三个部分，其中信号提取包括带通滤波和低通滤波两种数字滤波器，下面依据观测器处理流程进行分析。

(1) 带通滤波(BPF)，在电机运行过程中，ADC 采集到的电流信号不仅包含用于提取转子位置的高频信号，还包括电机正常运行产生的基频信号以及逆变器高频开关导致的谐波成分。由于转子位置的误差信号主要存在于高频信号中，因此需要引入带通滤波器，过滤掉对位置估计无关紧要的信号成分，仅保留含有误差信息的高频信号。经过滤波后的信号为：

$$BPF(\hat{i}_q) = \hat{i}_{qh} = \frac{u_h \sin \omega_h t}{\omega_h (\bar{L}^2 - \Delta L^2)} \Delta L \sin 2\tilde{\theta} \quad (4-12)$$

(2) 信号解调，将经过 BPF 过滤出来的高频响应电流与高频注入同频率的

正弦信号相乘，把位置误差 $\tilde{\theta}$ 与高频信号 $\sin(\omega_h t)$ 解调，对解调过程进行处理可以得到：

$$\begin{aligned}
 \hat{i}_{qh} \cdot \sin \omega_h t &= \frac{u_h \Delta L \sin 2\tilde{\theta} \sin \omega_h t}{\omega_h (\bar{L}^2 - \Delta L^2)} \cdot \sin \omega_h t \\
 &= \frac{u_h \Delta L \sin 2\tilde{\theta} \sin^2 \omega_h t}{\omega_h (\bar{L}^2 - \Delta L^2)} \\
 &= \frac{u_h \Delta L \sin 2\tilde{\theta}}{2\omega_h (\bar{L}^2 - \Delta L^2)} (1 - \cos 2\omega_h t) \\
 &= \frac{u_h \Delta L \sin 2\tilde{\theta}}{2\omega_h (\bar{L}^2 - \Delta L^2)} - \frac{u_h \Delta L \sin 2\tilde{\theta} \cos 2\omega_h t}{2\omega_h (\bar{L}^2 - \Delta L^2)}
 \end{aligned} \tag{4-13}$$

对式(4-13)化简得到：

$$\begin{cases} \hat{i}_{qh} \cdot \sin(\omega_h t) = k \sin 2\tilde{\theta} - k \sin 2\tilde{\theta} \cos 2\omega_h t \\ k = \frac{u_h \Delta L}{2\omega_h (\bar{L}^2 - \Delta L^2)} \end{cases} \tag{4-14}$$

(3) 低通滤波(LPF)，观察式(4-14)可以发现，高频电流与同频率的正弦信号相乘解调之后，信号被分解为两个部分：一部分是包含位置误差信息的低频信号，另一部分则是高频信号的二次谐波分量。由于低频信号中含有关键的转子磁极位置信息，因此，通过低通滤波器过滤掉高频的二次谐波信号，便能提取出转子信息，可以得到：

$$f(\tilde{\theta}) = LPF(\hat{i}_{qh} \cdot \sin(\omega_h t)) = k \sin 2\tilde{\theta} \tag{4-15}$$

当转子位置误差之间趋近于 0 时，此时可以将误差角函数线性化，得到：

$$f(\tilde{\theta}) = k \sin 2\tilde{\theta} \approx 2k\tilde{\theta} \tag{4-16}$$

从式(4-16)可知，通过调节 $f(\tilde{\theta})$ 趋近于 0，转子位置误差也趋近于 0，此时估计 \hat{d} - \hat{q} 轴与实际 d - q 轴重合，即估计的转子磁极位置与实际位置重合，锁定转子角度位置。

(4) PI 控制器，将 PI 控制器与低通滤波器串联，构成锁相环(PLL)，这一环节能够对转子的位置和速度信息进行更加精确的估计。这种配置能够利用 PI 控制器的稳定性、响应能力，以及低通滤波器的信号过滤功能，共同作用于转子位置的精确估计，提高整个控制系统的性能。

在传统的转子位置提取方法中，本文针对信号解调、滤波器设计和 PI 控制器部分提出以下优化改进方案。

(1) 调整调制函数，通过图 4-3 可以看出，在位置提取过程中，使用低通滤波器，不仅会产生相位滞后，经过调制处理的信号，对于注入的高频信号并不会完全被滤波器过滤掉，导致包含转子位置信息的误差信号中仍然还有部分高频

信号，这些高频信号被视为基频信号，进入 PI 观测器进行调节，不仅使转子位置估计产生误差，同时使速度估计产生高频脉动。常规的方法是提高低通滤波器阶数或者降低滤波器截止频率来减少基频信号中的高频分量，但这种处理方式会使得基频信号的相位滞后更加严重，同时使信号产生较大的幅值衰减，导致转子位置估计的误差增大。

基于以上因素考虑，本文设计了新的调制信号 $\cos\omega_h t \cdot \sin 2\omega_h t$ ，使用该信号对 \hat{q} 轴的高频响应电流进行调制可以得到：

$$\begin{aligned}
 \hat{i}_{qh} \cdot \cos \omega_h t \cdot \sin 2\omega_h t &= \frac{u_h \Delta L \sin 2\tilde{\theta} \sin \omega_h t}{\omega_h (\bar{L}^2 - \Delta L^2)} \cdot \cos \omega_h t \cdot \sin 2\omega_h t \\
 &= \frac{u_h \Delta L \sin 2\tilde{\theta} \sin^2 2\omega_h t}{2\omega_h (\bar{L}^2 - \Delta L^2)} \\
 &= \frac{u_h \Delta L \sin 2\tilde{\theta}}{4\omega_h (\bar{L}^2 - \Delta L^2)} (1 - \cos 4\omega_h t) \quad (4-17) \\
 &= \frac{u_h \Delta L \sin 2\tilde{\theta}}{4\omega_h (\bar{L}^2 - \Delta L^2)} - \frac{u_h \Delta L \sin 2\tilde{\theta} \cos 4\omega_h t}{4\omega_h (\bar{L}^2 - \Delta L^2)} \\
 &= \frac{1}{2} k \sin 2\tilde{\theta} - \frac{1}{2} k \sin 2\tilde{\theta} \cos 4\omega_h t
 \end{aligned}$$

经过新的调制信号处理后，转子位置误差信号被分解为低频信号和注入高频信号的四次谐波分量。与传统的调制方法相比，高频信号从二次谐波提升到四次谐波，对于低通滤波器来说，信号分离更加容易，并使得误差信号中的高频分量更少，能够有效减少转子位置估计的误差。

(2) 滤波器，在数字滤波器的设计过程中，采用 IIR 滤波器。IIR 滤波器是一种基于差分方程的滤波器，通过将输入信号与过去输出信号的加权计算来实现滤波效果，其传递函数和差分方程为：

$$H(z) = \frac{\sum_{r=0}^M b_r z^{-r}}{1 + \sum_{k=1}^N a_k z^{-k}} \quad (4-18)$$

$$y(n) = \sum_{k=1}^N a_k y(n-k) + \sum_{k=1}^M a_k x(n-k) \quad (4-19)$$

IIR 滤波器的传递函数具有无限冲激相应特性，输出信号可以持续响应与输入信号的变化，使得其能够实现复杂的频率选择和滤波功能。同时因为 IIR 滤波器利用了反馈结构，在处理信号时，可以利用过去的输出结果，从而减少计算所需的数据量，提高运算效率。在滤波特性上，IIR 滤波器通常可以实现更抖的滤

波特特性，具有更快的过渡带；在稳定性上，与其传递函数的极点位置有关，当极点位于单位圆内，系统稳定。

本文采用直接 II 型的二阶 IIR 滤波器，其基本结构如图 4-4 所示。

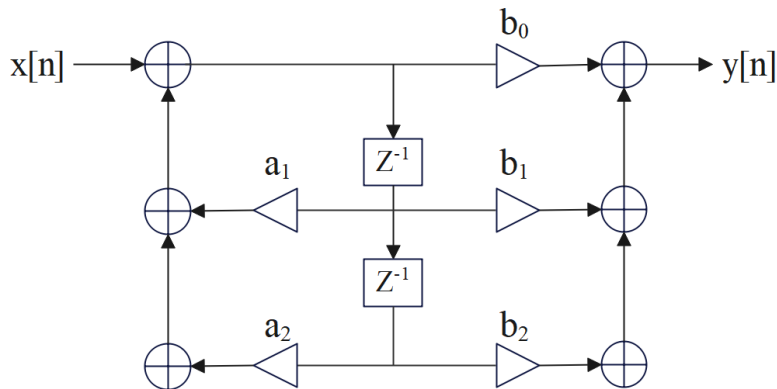


图 4-4 直接 II 型 IIR 滤波器结构图

在 Matlab 中设计对应的低通滤波器和带通滤波器，其幅值响应和相位响应曲线如图 4-5 和 4-6 所示。

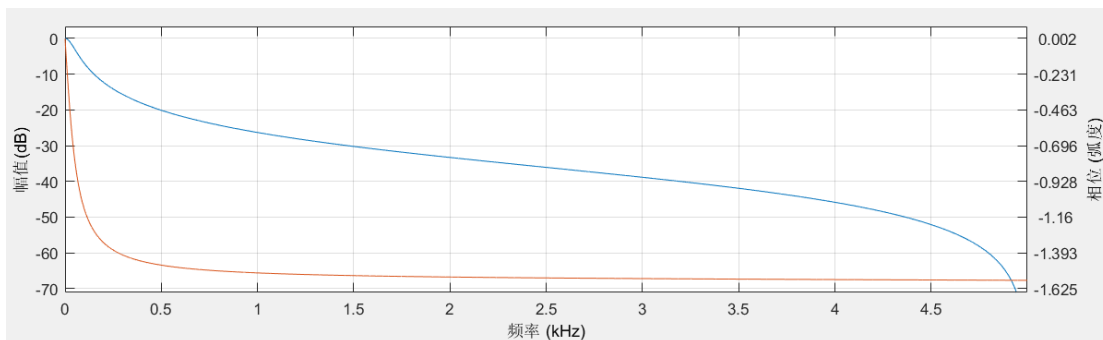


图 4-5 LPF 幅值相位响应曲线

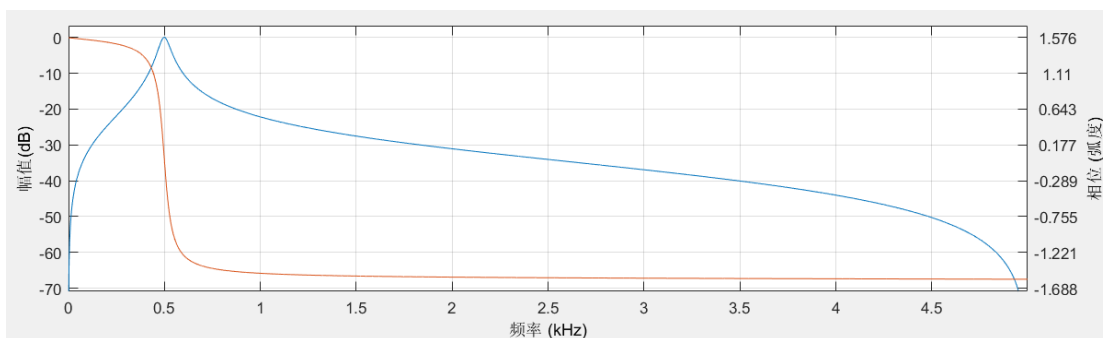


图 4-6 BPF 幅值相位响应曲线

(3) PI 控制器，在传统的 PI 控制策略中，设计者需要在系统稳定性与响应速度之间做出权衡，一味追求响应速度可能会降低系统的鲁棒性，甚至会导致电

机在特定条件下出现反转现象。为了增强系统的抗干扰能力并提升鲁棒性，同时避免由于观测误差增大而引起的估计失效，在控制策略中引入微分项。在 PI 控制器中引入微分环节，可以显著降低转子速度在初始阶段可能出现的反转问题，实现对输入的快速精准跟踪。不仅提高了系统的抗干扰能力，还能在面对阶跃信号或斜坡负载等情况时，有效改善系统的动态响应特性，从而确保电机控制系统的高效稳定运行。

根据以上方法得到新的转子位置观测器结构图如图 4-7 所示。

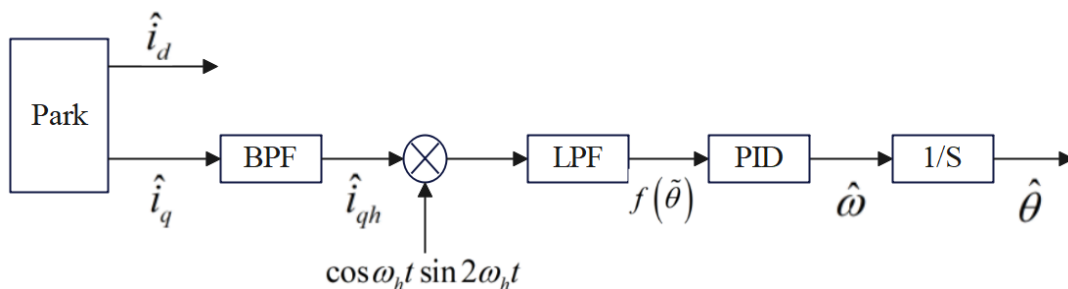


图 4-7 优化后转子位置观测器结构图

4.2.3 转子极性判断

本章所采用的脉振高频信号注入法，可以确认凸极轴线的位置，但无法判断此时的直轴方向为正还是负^[54]，故还需要使用另外的方法判断转子 d 轴正方向，即转子的极性判断。

转子的极性判断可以在高频注入法收敛的角度 $\hat{\theta}$ 上加上正负偏置电压，根据响应电流大小来识别 NS 极，具体步骤如下：

- 1) 在估计的旋转坐标系的 \hat{d} 轴注入高频信号，得到转子位置初值 $\hat{\theta}$ ；
- 2) 向 \hat{d} 轴施加正向偏置电压，高频注入叠加该正偏置电压，将响应的 α - β 高频电流幅值累加，得到 i_{d1} ；
- 3) 向 \hat{d} 轴施加负向偏置电压，高频注入叠加该负偏置电压，将响应的 α - β 高频电流幅值累加，得到 i_{d2} ；
- 4) 比较 i_{d1} 与 i_{d2} 的大小，若 $i_{d1} > i_{d2}$ ，则说明是 N 极， $\hat{\theta}$ 即为正确角度，反之则说明是 S 极， $\hat{\theta}$ 应该加上 180° 。

该转子极性判断的原理是，若 $\hat{\theta}$ 对应的是 N 极，施加正向偏置电压后，该偏置电压差生的磁场和永磁体磁场方向相同，d 轴进入深度饱和区， L_d 减小，使高频响应电流的幅值增加；相反的，当施加负向偏置电压后，磁场减弱， L_d 增大，高频响应电流的幅值减小。若 $\hat{\theta}$ 对应的是 S 极，现象与上面相反，正向偏置电压导致响应电流幅值减小，负向偏置电压导致高频响应电流的幅值增加。通过以上方法得到高频注入法得到的真实转子位置信息，实现电机闭环控制。

4.2.4 脉振高频注入控制流程

对于低转速下电机无感矢量控制，采用脉振高频注入法获得转子位置和速度信息，高频注入的控制流程如图 4-8 所示。

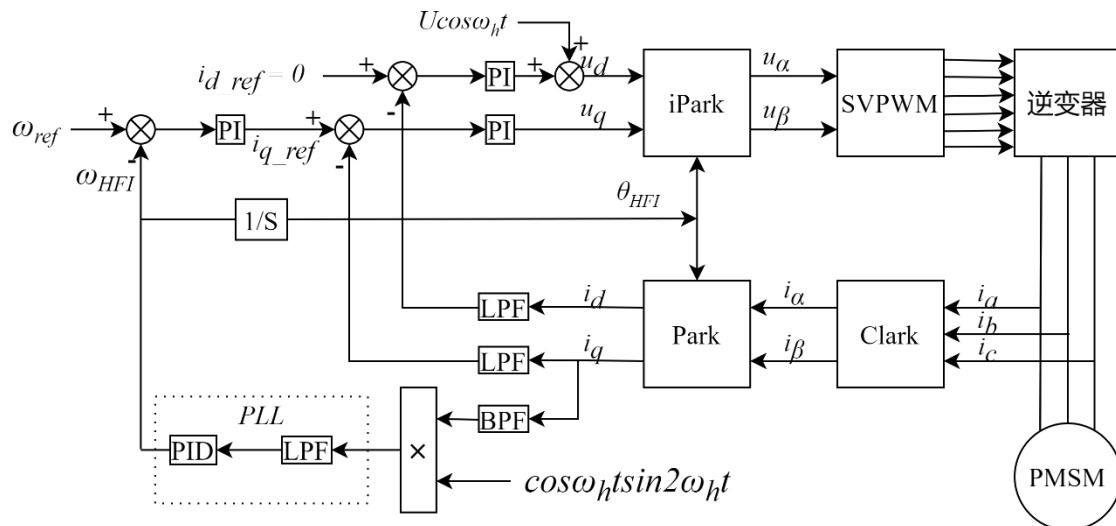


图 4-8 基于脉振高频注入法的矢量控制框图

通过图中可以看出，采用脉振高频注入的方法可以实现电机的双闭环控制，具体控制过程与矢量控制类似，但是在低转速下，通过向电流环 PI 控制后的 d 轴注入高频电压信号实现闭环控制。当完成信号注入后，通过 ADC 采集得到 A、B、C 三相电流，经过 Clark 变换和 Park 变换，得到 q 轴的高频响应电流，对该相应电流经过带通滤波后，与设置的调制信号相乘，将转子位置误差信号解调出来，之后通过低通滤波器和 PID 控制器组成的 PLL，锁定转子的速度信息，后续通过积分处理便可得到位置信息，实现在零速和低速域下对电机的双闭环控制。该优化方案性能在论文第 5 章中通过搭建全速域矢量控制系统进行测试。

4.3 本章小结

本章首先介绍了电机非线性饱和凸极性原理，根据这一原理采用脉振高频电压信号注入获得低速域内转子位置和速度信息，并对该方法的不足之处，修改解调函数，将二次谐波分量提升为四次谐波；同时优化高频注入中的滤波器与 PI 控制器设计，并分析极性判断的方法，使电机在低转速下，也能获得转子的位置和速度信息；最后说明优化后的基于高频注入的矢量控制方案及软件处理流程。

第5章 电机全速域混合控制方案及系统搭建

前面,本文分析了永磁同步电机矢量控制的基本原则,并针对低速域和中高速域,分别展示了利用非线性磁链观测器和脉振高频注入获得转子位置和速度信息的方法。为实现对电机全速域的控制,采用混合控制策略,设计一种加权方案,对两种无感方案的转速过渡区域进行优化,保证转子在该区域能平滑转动。之后根据本文所研究内容,针对实验室的 DSP 芯片,搭建全速域矢量控制系统平台,分析驱动板中的电源、采样等电路,阐述本文设计的电机控制方案的相关性能,并为 DSP 提供完整的驱动和算法代码,以使用户仅需调整电机响应参数即可完成电流速度双闭环控制。

5.1 全速域无感控制方案

根据第三章和第四章的分析可知在低速域和中高速域采用两种不同的控制算法,低速域下采取脉振高频信号注入法,中高速域下采用非线性磁链观测器,这两种方法的任何一种都无法保证在全速域下有良好的实现效果。为了实现对电机全速域内精准的矢量控制,需要将脉振高频注入和磁链观测器在不同的转速范围内进行切换。

5.1.1 全速域混合控制策略

常见的控制切换方案有:直接切换、滞环切换和线性加权切换,每种切换方法各有其优缺点,下面对三种切换方法进行说明。

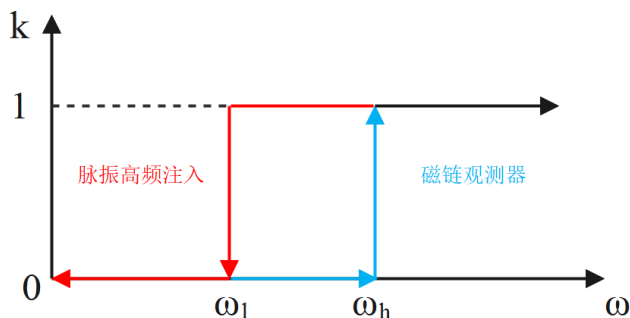
(1) 直接切换

直接切换法,顾名思义是一种在电机的全速域内根据设定的转速切换点进行控制方法切换的策略。具体来说,如果目标转速低于这个切换点,系统判定电机处于低速域,采用高频注入;若转速高于切换点,判定处于中高速域,采用非线性磁链观测器。该方法的优点在于实现简单,仅需确定一个合适的切换点,即可在全速域内控制电机。然而,这种方法由于考虑的实际情况较少,可能会导致控制系统在稳定性方面表现不佳。

(2) 滞环切换

滞环切换基于将转速切换过程细分为加速切换和减速切换两个阶段,旨在减轻电机控制中频繁切换引发的问题。为了应对直接切换可能带来的转速波动问题,

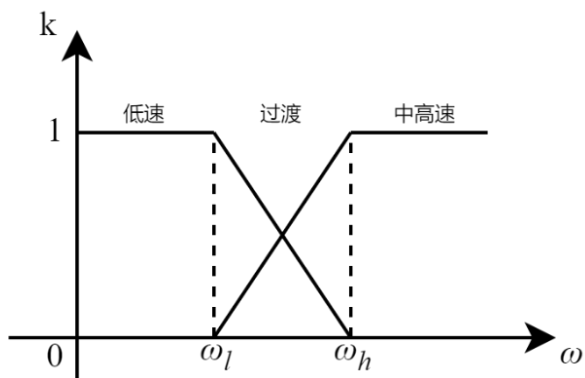
此方法在全速域内设置两个切换点，并据此将转速划分为不同的区间，从而形成一个滞环区间，其原理图如 5-1 所示。



滞环切换虽然有效避免了频繁切换引起的问题，但也有其缺点。若滞环区间设置过小，无法充分利用滞环效应，使实际效果与直接切换无异；若区间过大，原本的低速运行方案在中高速时无法有效进行控制，进而导致电机失控。

(3) 线性加权切换

线性加权切换原理如图 5-2 所示，该方法将速度划分为三个区域，来实现对电机控制策略的平滑过渡，具体步骤为：当电机目标转速低于 ω_l ，采用脉振高频注入；大于 ω_h ，采用磁链观测器；位于 $[\omega_l, \omega_h]$ 区间内时，根据转速值，动态调整低速域和中高速域控制策略结果的权重分配，使两种策略的计算结果共同决定转子的转速大小。



线性加权切换法的权重系数 k 值与转速的关系为：

$$K = \begin{cases} 1 & , \omega \leq \omega_l \\ \frac{\omega_h - \omega}{\omega_h - \omega_l} & , \omega_l < \omega \leq \omega_h \\ 0 & , \omega > \omega_h \end{cases} \quad (5-1)$$

转子位置和速度最终值为：

$$\begin{bmatrix} \hat{\theta} \\ \hat{\omega} \end{bmatrix} = K \begin{bmatrix} \hat{\theta}_{HFI} \\ \hat{\omega}_{HFI} \end{bmatrix} + (1-K) \begin{bmatrix} \hat{\theta}_{Flux} \\ \hat{\omega}_{Flux} \end{bmatrix} \quad (5-2)$$

其中 ω_{HFI} 、 ω_{Flux} 、 θ_{HFI} 、 θ_{Flux} 分别为高频注入和磁链观测器观测到的转子角度和转速值。

5.1.2 改进后的混合控制策略

为了解决传统切换方法的不足，本文提出了一种新型的切换算法，该方法基于传统的线性加权切换法，将线性的切换方法转换成非线性切换。切换函数在保持线性切换的优点的同时，对于过渡区的权值做出平滑处理，加权系数和转速的关系曲线如图 5-3 所示。该曲线的最大优点是，在保证低速域内使用脉振高频注入，中高速域内使用磁链观测器的同时，对于过渡区域，采用更加平滑的切换方式，避免电机运行在过渡区时，因为切换导致的抖振现象。

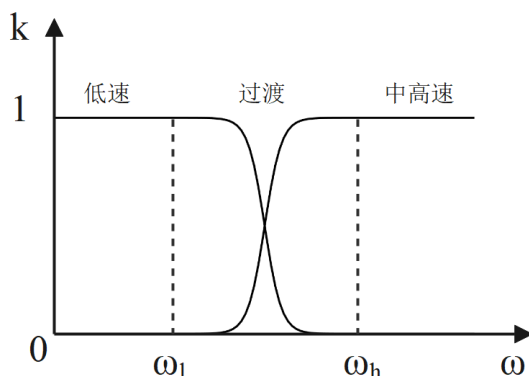


图 5-3 非线性加权切换法原理图

非线性加权切换法的权重系数 k 值与转速的关系为：

$$K = \begin{cases} 1 & , \omega \leq \omega_l \\ 1 - \frac{1}{1 + e^{-a(\omega - \omega_m)}} & , \omega_l < \omega \leq \omega_h \\ 0 & , \omega > \omega_h \end{cases} \quad (5-3)$$

根据式(5-3)可以得到，随着电机转速的升高，脉振高频注入观测到的转子位置和速度信息的权重回逐渐降低，与此同时，设置向 d 轴注入的高频电压信号的幅值为：

$$\hat{u}_d = k \cdot u_h \cos \omega_h t \quad (5-4)$$

通过降低高频信号的幅值，来减少信号对磁链观测器算法的影响，从而提高电机全速域范围内无位置传感器算法估计转子位置和速度的精度。

得到全速域内无位置传感器矢量控制流程图如图 5-4 所示。

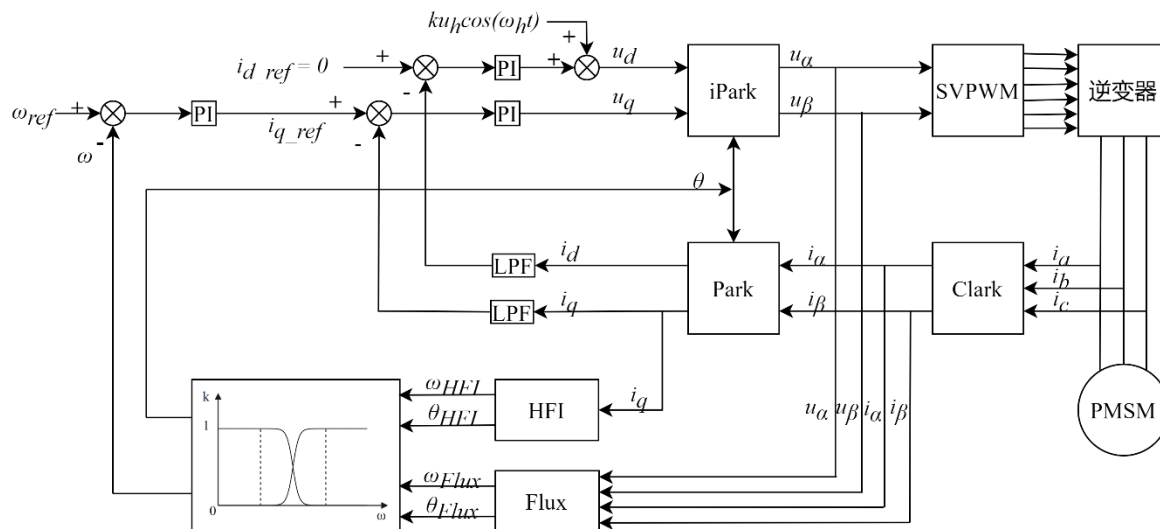


图 5-4 全速域无传感器矢量控制流程图

5.2 硬件系统搭建

本实验平台选用 QX320F280049C 作为主控芯片，电源电路、驱动模块、三相采样电路模块等组成矢量控制系统。控制系统框图如图 5-5 所示。

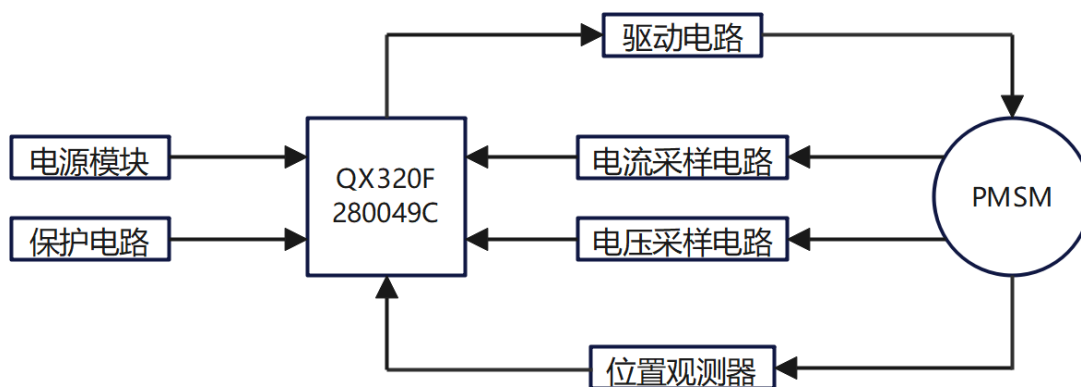


图 5-5 永磁同步电机无传感器控制框图

5.2.1 主控芯片最小系统

本章实验部分采用的主控芯片为 QX320F280049C，该芯片为 32 位双核 CPU，

支持 IEEE754 单精度浮点运算单元(FPU)，同时采用本文第二章设计的三角函数运算加速方案，支持常见的三角函数硬件加速。片上存储器包括 1MB 的 FLASH 和 512KB 的 SRAM，其中 SRAM 分为 256K 的指令 RAM 和 256K 的数据 RAM。该 DSP 芯片支持片上晶振振荡器和外部时钟输入，晶振频率可选 10~16MHz，系统时钟采用 PLL，通过对晶振倍频和分频得到，最大支持 400MHz 时钟。该芯片内核电压为 1.2V，数字 I/O 电压为 3.3V，可生成 1.2V 的内部 VERG 电压。具有 DMA 通道、增强型 PWM 通道、3 个 3M 采样率的 12 位 ADC 等模拟、数字和通信外设，芯片引脚图如图 5-6 所示。

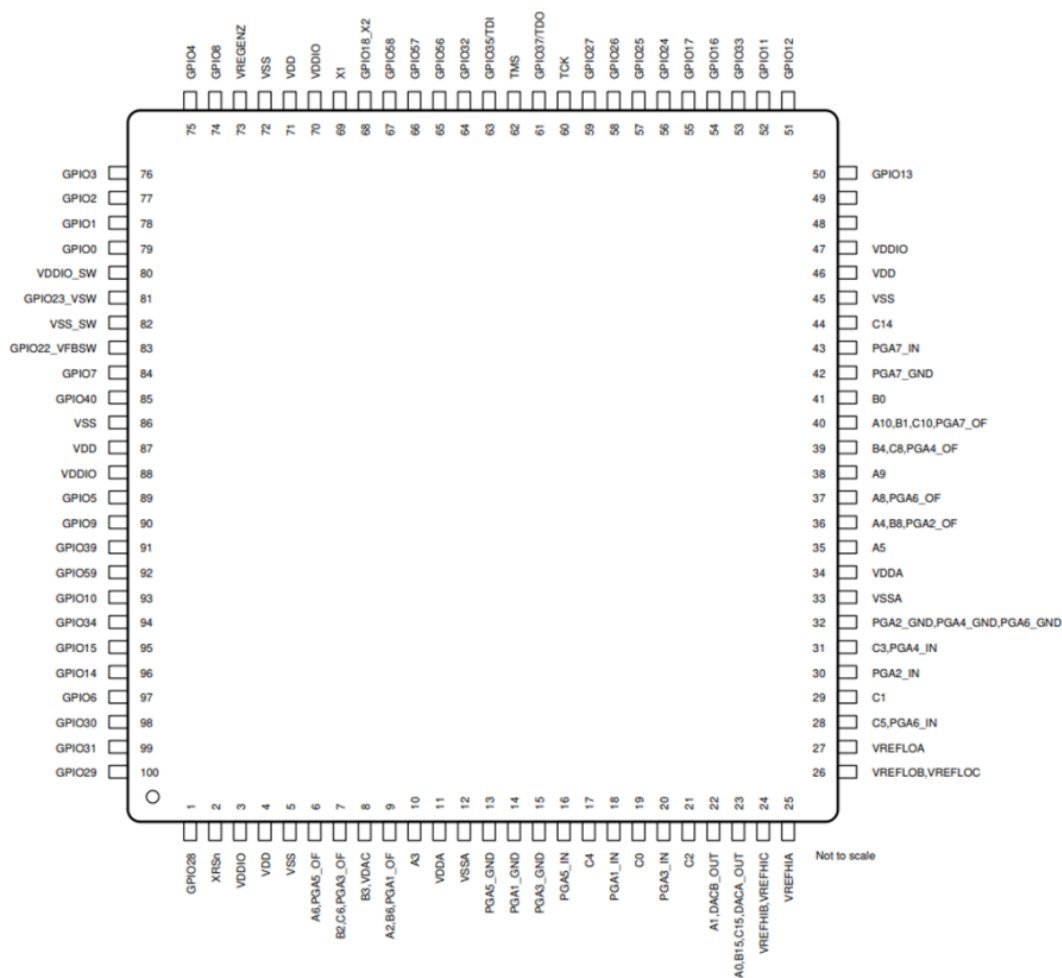


图 5-6 主控芯片最小系统引脚图

5.2.2 电源电路

控制系统的电源电路如图 5-7 所示，输入为 24V，为了给驱动电路提供 12V 的输入电压，本系统通过一个电流模式同步降压稳压器 RT7272B 将 24V 降压为 12V。系统中三相电流采样电路以及主控芯片工作所需输入电压为 3.3V，经过同样的稳压器得到，通过改变输出端的电感电阻大小，实现对 3.3V 电压的输出。

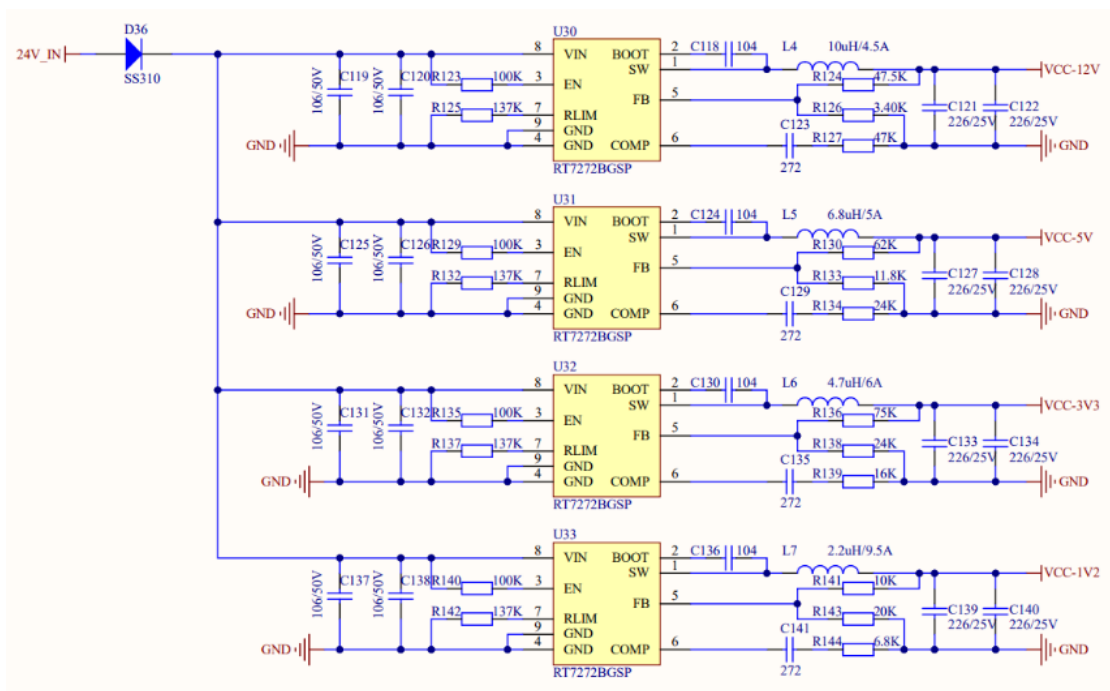


图 5-7 系统电源电路

5.2.3 驱动电路

DSP 控制系统输出的 3.3V 的 PWM 信号无法直接驱动 MOSFET，需要设置驱动电路将 DSP 发出的 PWM 信号进行放大。本文采用的驱动电路芯片为 IR2110STRPBF，该驱动芯片可以同时驱动逆变器同一相上的两个 MOSFET，三相逆变桥驱动电路如图 5-8 所示。

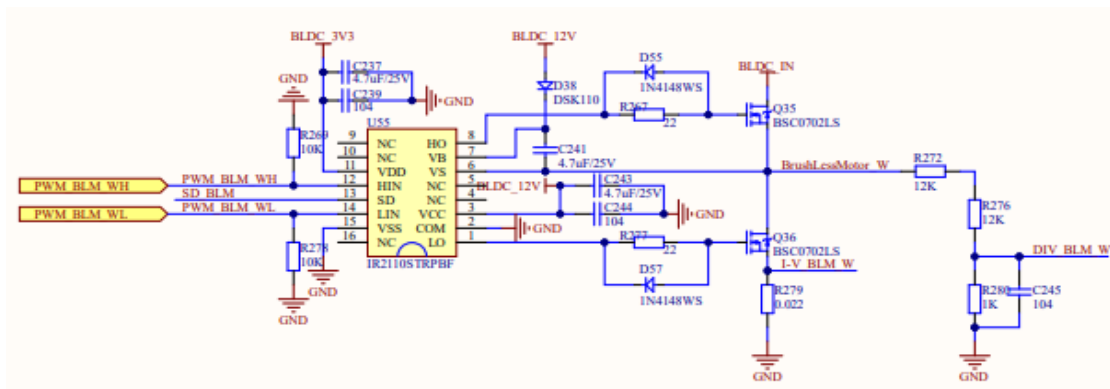


图 5-8 三相逆变驱动电路图

5.2.4 采样电路

根据第三章对电机矢量控制原理的分析可知，电流采样结果的精度影响着矢

量控制效果，同样也影响着无传感器方案对转子位置和速度信息估计的精度，所以电流采样精度越高，电机控制效果越好。本文针对 FOC 电流环控制部分，选择三电阻采样，获取 i_a 、 i_b 、 i_c ，其采样电路如图 5-9 所示。采样电路两端连接采样电阻 R229、R252、R279，C230、C235、C240 为滤波电容，下面以 A 相采样电路为例，分析电流采样的放大倍数。A 相电流经过电阻 R229 转换成采样电压，再经过由电阻 R247、R248 和 R251 构成的运放电路，得到放大倍率 K 为：

$$K = \frac{R251}{R247 + R248} = \frac{12K}{1K + 1K} = 6 \quad (5-5)$$

同时采样电路中，当 A 相电流为 0 时，存在 1.25V 的电压偏置，需将偏置电压减去才能得到真实的采样结果，故 A 相电流与采样得到的电压值之间的关系为：

$$i_a = \frac{V_a - 1.25}{0.022 \times 6} \quad (5-6)$$

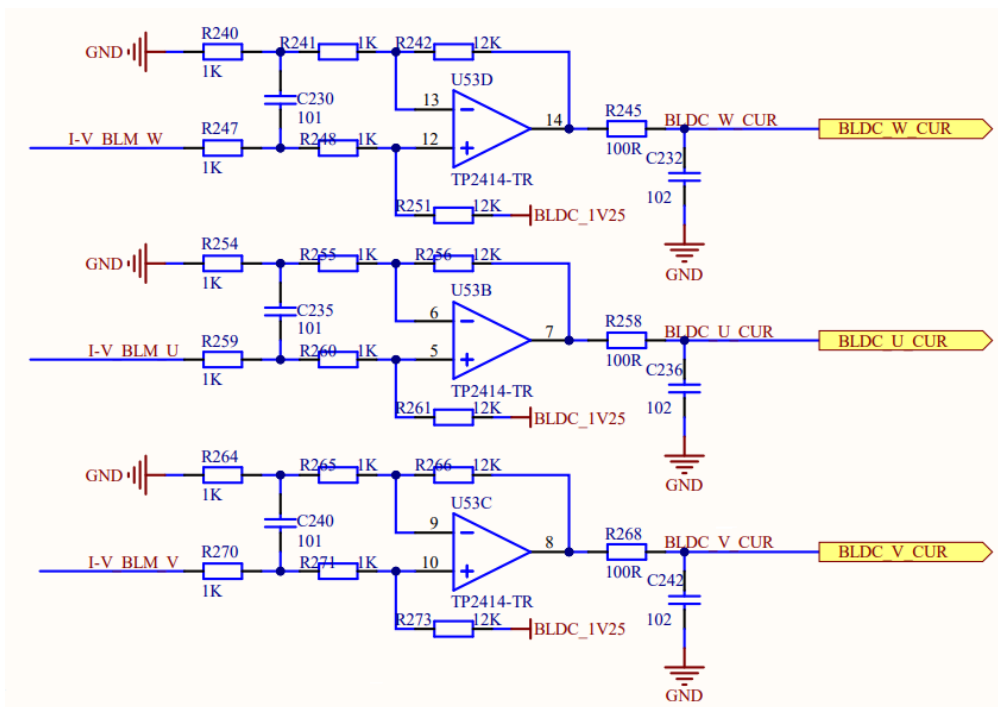


图 5-9 三相电流采样电路图

除电流采样外，考虑到本文针对无传感器下转子位置信息估算采用非线性磁链观测器，该方案需要获得 α - β 轴电压，故增加三相电压的采样电路，其原理图如图 5-10 所示。

下面同样以 A 相采样电路为例，分析电压采样的放大倍数。A 相电压经过 R220、R222、R230 分压，采样电路接在采样电阻 R230 端，同时因为电压采样电路不存在偏置问题，故 A 相电压与采样得到的电压值之间的关系为：

$$u_a = \frac{V_a \times R230}{R220 + R222 + R230} = \frac{V_a}{25} \quad (5-7)$$

最后针对电机在不同的负载下，电源电压可能出现波动，需增加针对该电压的采样电路，其原理图如图 5-11 所示。

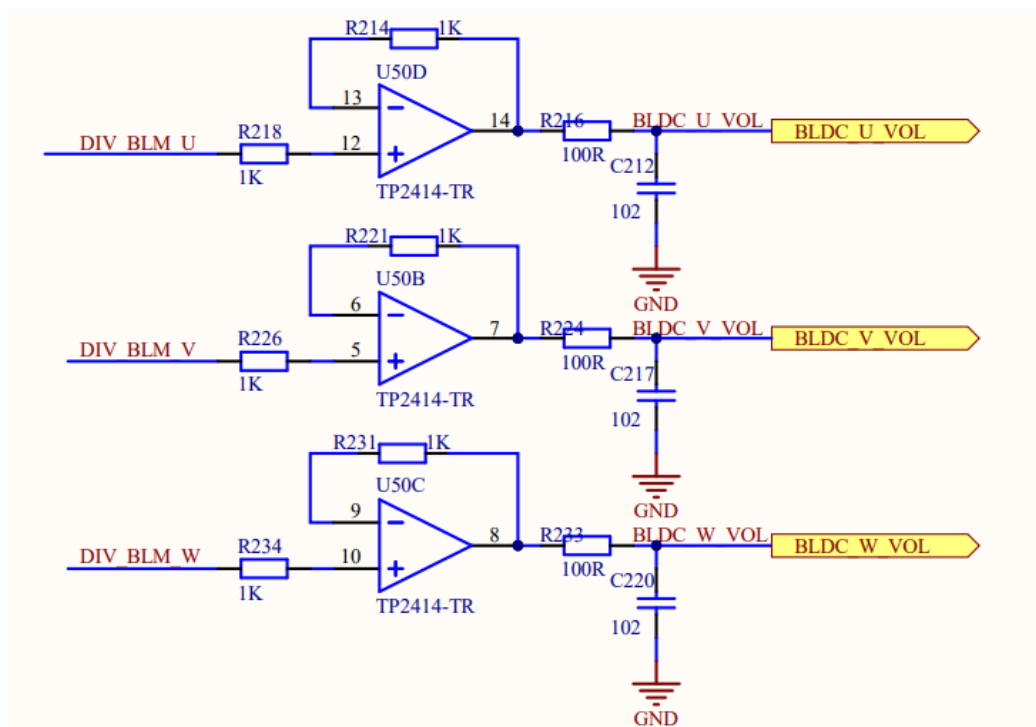


图 5-10 三相电压采样电路图

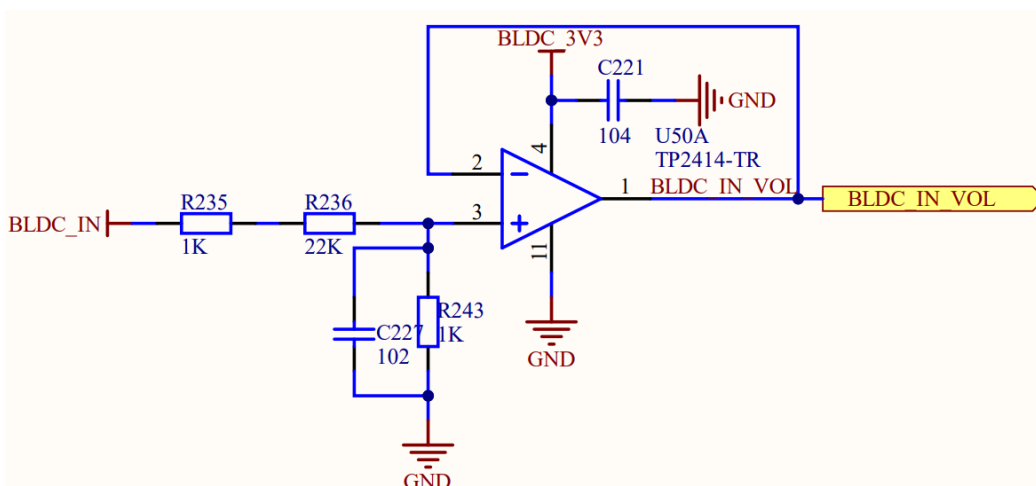


图 5-11 电源电压采样电路图

根据采样电路可知，电源电压与采样得到的电压值之间的关系为：

$$u_{dc} = \frac{V_{dc} \times R243}{R235 + R236 + R243} = \frac{V_{dc}}{24} \quad (5-8)$$

5.2.5 保护电路

本系统的过流保护电路如图 5-12 所示。该电路采用 LM339APWR 比较器比较三相电压与给定电压之间的大小关系。当某一相采样电压大于给定的电压时，SN74LVC1G00DBVR 的 4 引脚输出信号被拉高，使灯 D58 常亮，同时该电平信号与驱动芯片 IR2110S 的 SD 引脚相连接，接收到故障信号后，会让驱动芯片停止 PWM 信号的输出，逆变器工作停止，从而保证不被损毁。

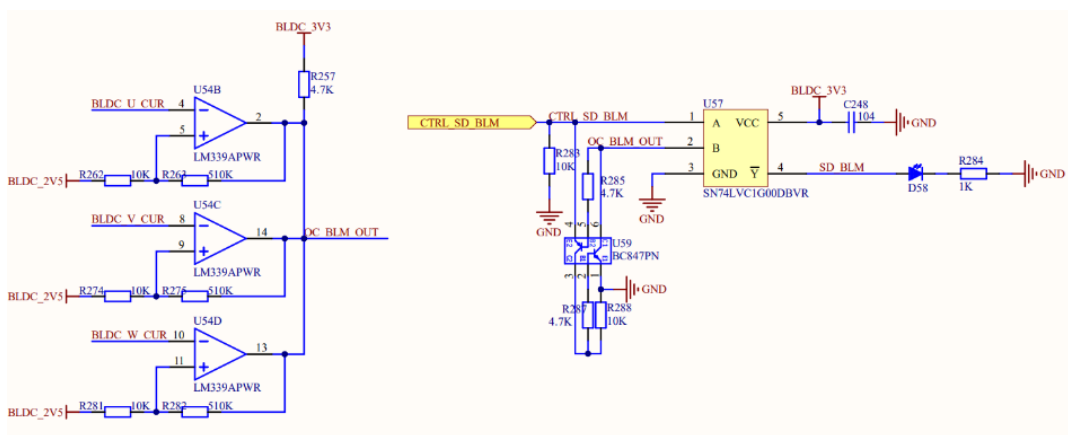


图 5-12 过流保护电路原理图

5.3 软件程序设计

根据本文设计的无传感器永磁同步电机矢量控制系统硬件实验平台进行软件系统设计，软件程序设计主要包括两大部分：接口处理和数据处理。其中接口处理主要作用是与硬件接口相交互，包括 ADC 采样、PWM 信号输出等部分。数据处理部分用于矢量控制及转子位置信息等数值的计算。

5.3.2 主程序设计

主程序软件设计主要包含对系统的相关参数，功能接口的初始化。具体包括以下部分：

- 1) 主控芯片系统时钟：QX320F280049C 采用 PLL 对晶振进行倍频与分频得到芯片主频，最大支持 400M，需根据具体需求进行配置，保证系统时钟能正常拉起；
- 2) 电机参数：极对数、相电阻、 d - q 轴电感、磁链等；
- 3) 外设模块参数：ePWM、ADC 参数配置；
- 4) PI 控制器参数：电流环和速度环的 K_P 、 K_I ；

5) 相关模块参数：驱动板采样和放大参数、磁链观测器参数、高频注入参数。

主程序设计流程图如图 5-13 所示。

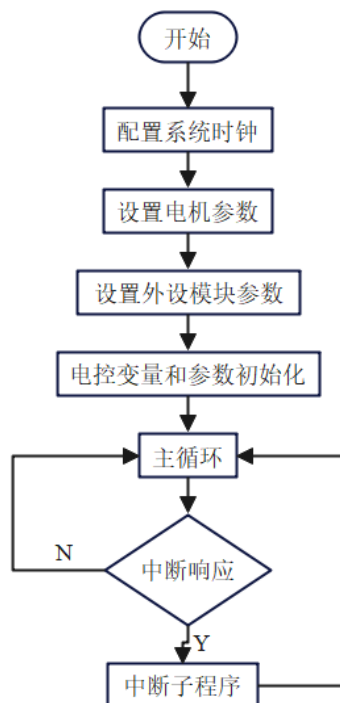


图 5-13 主程序设计流程图

5.3.3 中断子程序设计

因为矢量控制通过获得三相电流进行 Clark 变换开始，所以在 ADC 中断中进行 FOC 的相关计算。中断服务程序主要作用服务于主函数，电流采样、AD 转换、坐标变换、转速环 PI 控制、电流环 PI 控制、SVPWM 模块计算、磁链观测器、高频注入等。

中断子程序流程图如图 5-14 所示，当 ADC 中断产生，进入对应的中断服务程序之后，需要首先获取采样得到电压电流信号，并根据放大倍数、偏置等信息还原成电机三相电流电压值，将电流值经过坐标变换后，通过磁链观测器模块，估算转子的位置和速度信息，将得到的速度值带入速度环 PI 控制，从而实现速度闭环；之后进行电流闭环，经过逆变换后执行 SVPWM 得到该时刻下三路 PWM 信号的占空比，输出到硬件驱动电路模块，驱动电机转动。虽然硬件驱动电路设计了过流保护电路，会在电路过流时关闭 PWM 信号的输出，软件算法部分同样会对采样得到的电流进行判断，如果此时认为处于过流状态，会将 PWM 信号的占空比设置为 0，停止信号输出，从而对电机以及驱动电路进行保护。

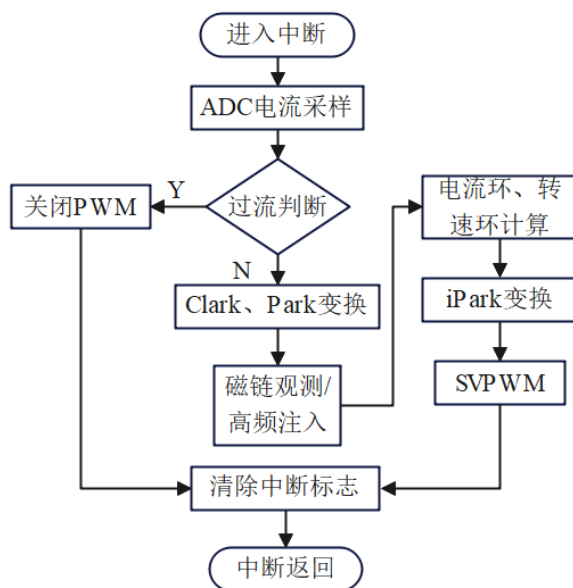


图 5-14 中断子程序设计流程图

5.4 实验与分析

5.4.1 电机运行验证

系统平台如图 5-15 所示，图中为电机主控板、电机驱动板、直流稳压电源、jtag 工具、永磁同步电机，整个实验过程通过 SCI 串口检测电机转动过程中相关的参数。电机控制板选择 QX-C2000 系列的开发板，编译 debug 工具为 QX-IDE v1.5.2。

实验电机具体参数如下：

表 5-1 电机参数

参数	数值	单位	参数	数值	单位
额定功率 P	70	W	扭矩	0.22	N·m
供电电压	24	V	相电阻	0.3	Ω
d 轴电感	1.5	mH	q 轴电感	1.5	mH
反电势系数	2.0	Vrms/Krpm	磁极对数	4	
额定转速 ω	3000	RPM	额定电频率	100	Hz

本实验通过代码，使三相逆变器的 6 个桥臂输出特定频率和占空比的 PWM 信号，使用示波器测量驱动电路中上下桥臂之间的信号，验证驱动电路与逆变电路功能正常。

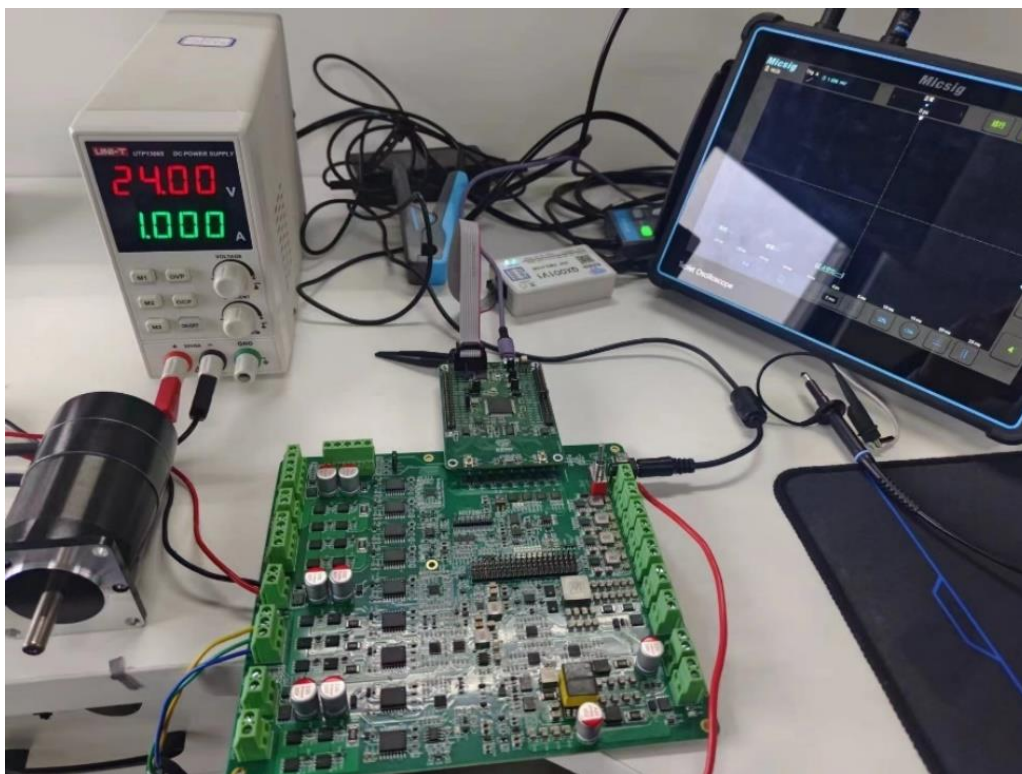


图 5-15 实验平台

两路互补的 PWM 信号波形图如图 5-16 所示，其中蓝色为 A 相，紫色为 B 相，幅值为 3.3V，频率为 16kHz。

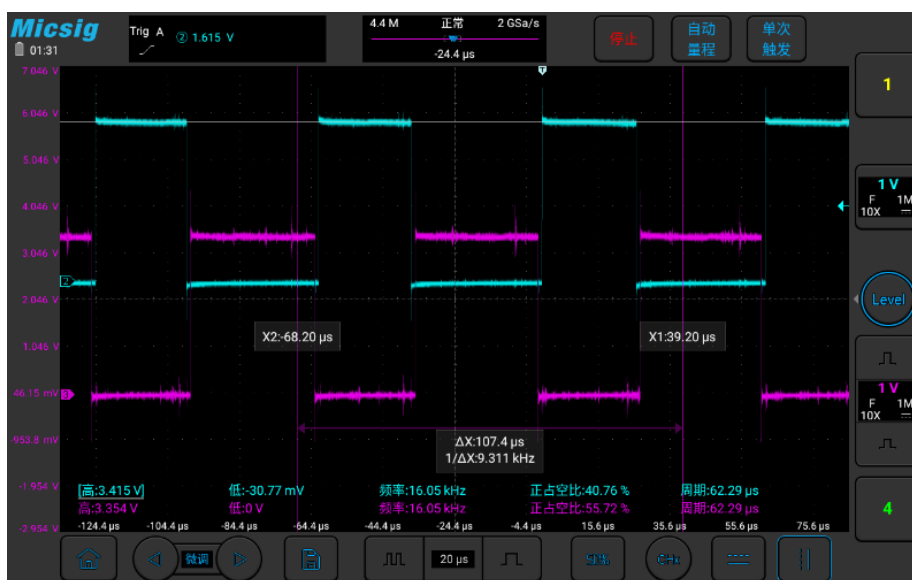


图 5-16 PWM 输出波形

MOSFET 驱动电压为 12V，IR2110S 驱动电路形成的两路互补 PWM 信号的波形图如图 5-17 所示，其中黄色为 A 相，蓝色为 B 相。

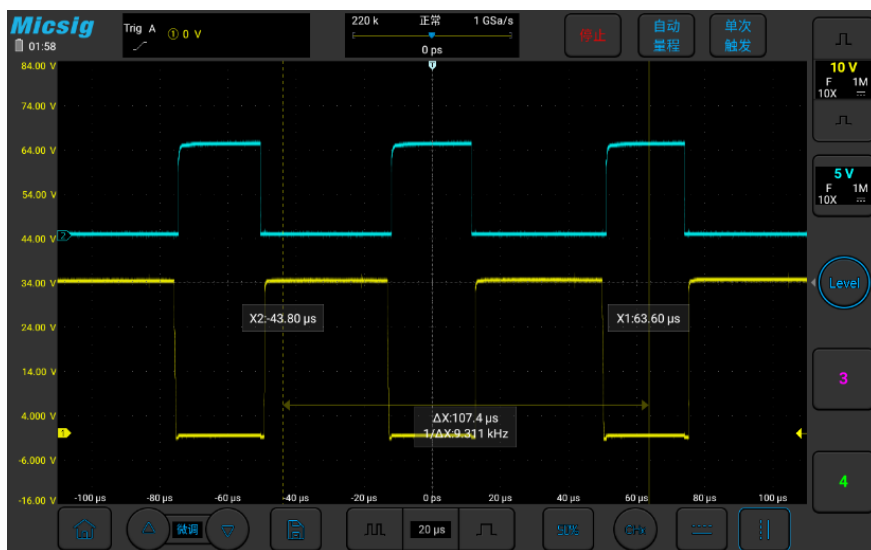


图 5-17 MOSFET 驱动信号

三相逆变器中 U 相输出电压波形图如图 5-18 所示,从图中可以看到该 PWM 信号的高电平为 24V,证明逆变电路工作正常。

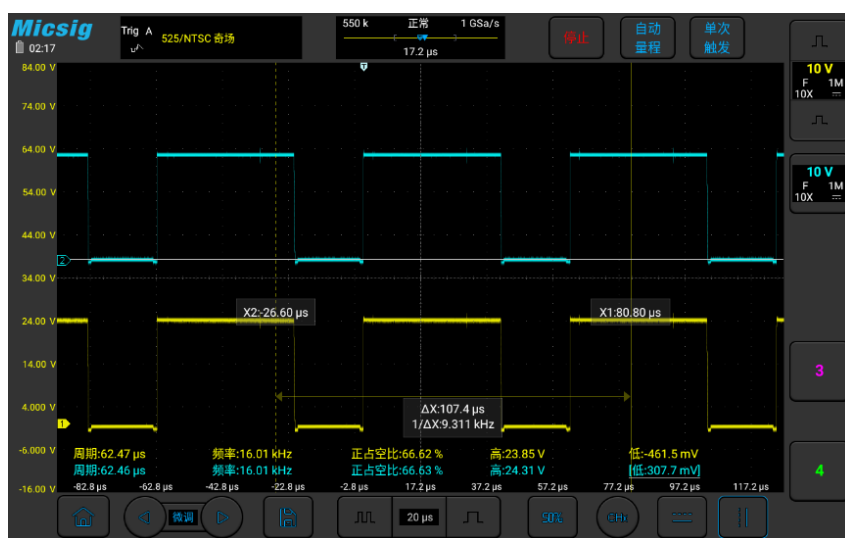


图 5-18 逆变器输出信号

电机控制试验调试步骤分为电流开环、电流闭环、速度闭环、高频注入以及全速域测试 5 个部分,下面按照该顺序分别进行测试,并利用示波器、串口抓去对应数据进行分析。

1) 电流开环

通过在 ADC 中断程序中,每次递增固定角度用于 iPark 计算,因为是开环控制,所以电流环 PI 控制输出的 d 轴和 q 轴电流设定为恒定值,通过示波器抓取 U 相电流波形,同时利用串口打印 ADC 采集得到的三相电流值,观察三相电

流值的关系。

通过实验得到，U 相电流波形如图 5-19 和 5-20 所示，可以看到电机在电流开环运行下电流波形大致为正弦信号，因为此时没有采用闭环控制，正弦效果较差。

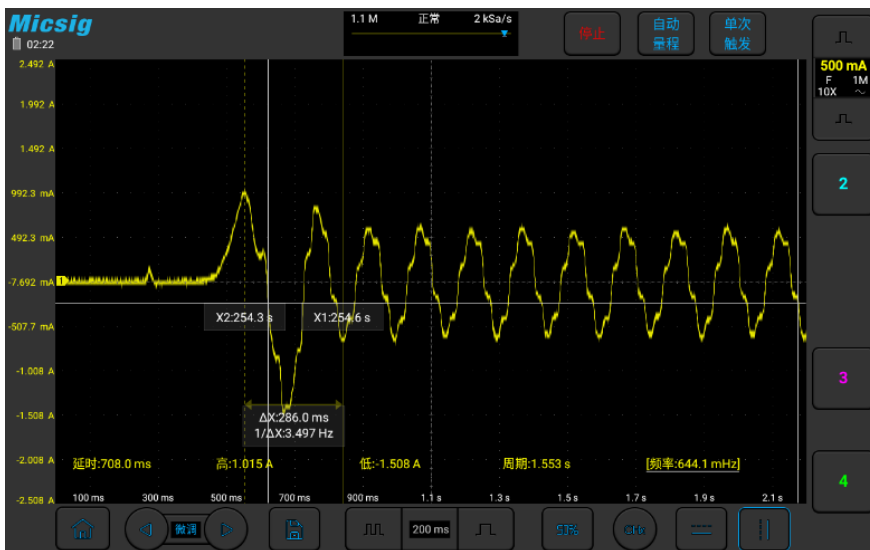


图 5-19 电流开环启动时 U 相电流信号

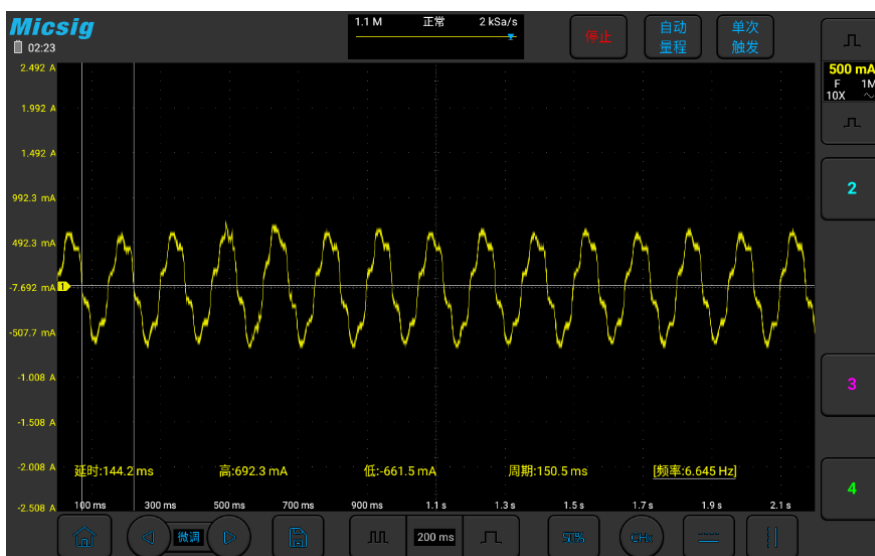


图 5-20 电流开环运行时 U 相电流信号

ADC 采集的三相电流以及电源电压信号如图 5-21 和 5-22 所示，三相电流为 120° 相位差的正弦信号，且电源电压也在 24V 左右，考虑到接线部分的压降，可以认为该采样与计算结果正确。

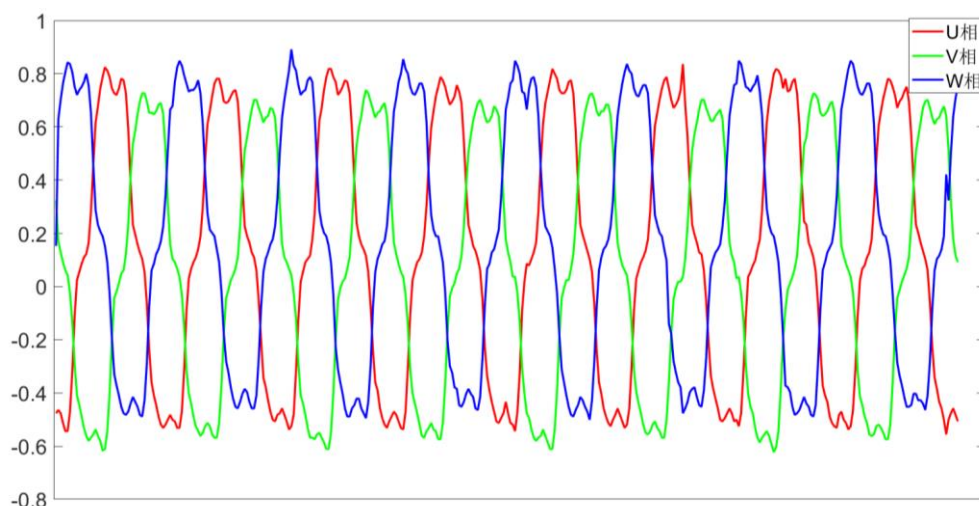


图 5-21 电流开环三相电流采样结果波形图

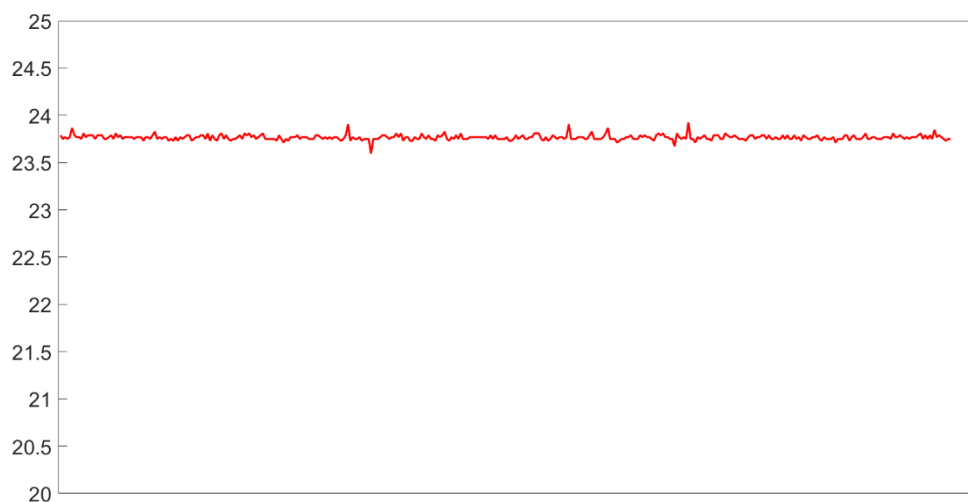


图 5-22 电源电压采样结果波形图

2) 电流闭环

当 ADC 采样结果正确之后，对电机进行电流闭环控制，将程序中 d-q 轴的恒定电压注释，采用 PI 控制器对采样得到的结果进行控制，同时观察三相电流波形与设定的 q 轴参考电流是否一致，程序中设定 q 轴电流值为 1.8A。

通过实验得到，U 相电流波形如图 5-23 所示，可以看到电机在电流闭环运行下电流波形为正弦信号，且幅值与设定值一致。

ADC 采集的三相电流信号如图 5-24 所示，三相电流为 120° 相位差的正弦信号，采样计算得到的电流值也在 $\pm 1.8V$ 之间，认为电机电流闭环控制正常。

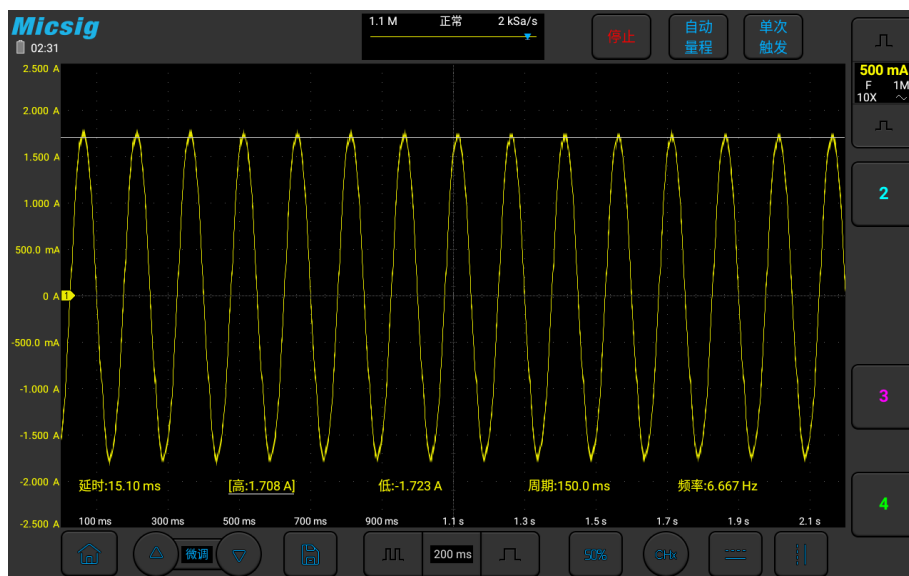


图 5-23 电流闭环运行时 U 相电流信号

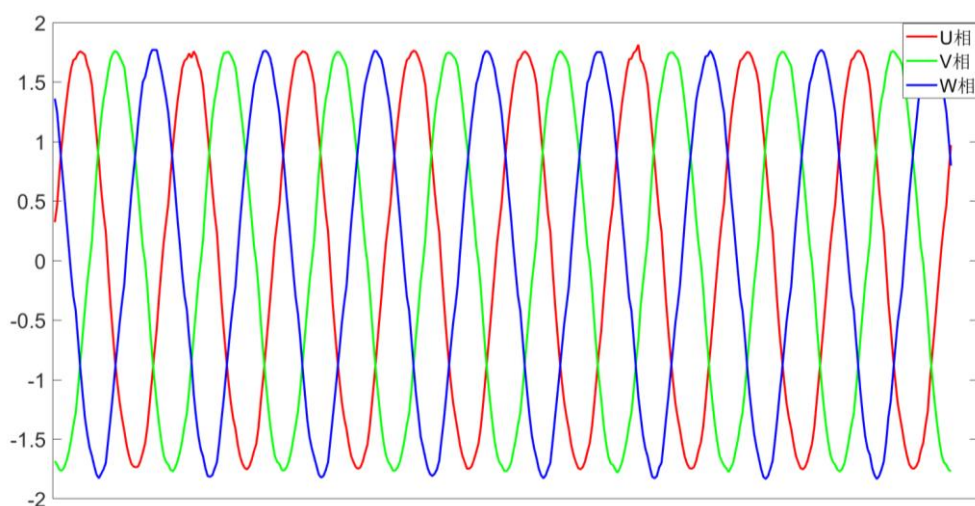


图 5-24 电流闭环三相电流采样结果波形图

3) 速度闭环

在对电机进行速度闭环控制之前，需要在电流闭环正常运行的基础上，观察磁链观测器计算的角度和给定的开环角度是否一致。通过记录观测器计算的角度和开环给定角度，并画出其数据波形，结果如图 5-25 所示。

从图中可以看出，观测器计算的角度与给定角度的波形都为三角波，且二者频率相同，因为观测器计算得到的是转子相对于三相线圈的真实角度位置，其与电流闭环所使用的角度必然存在一定的相位差，所以而且之间的波形不是重合。同时因为没有进行速度闭环控制，电机转速存在波动的现象，故观测器计算的角度波形存在抖动问题。

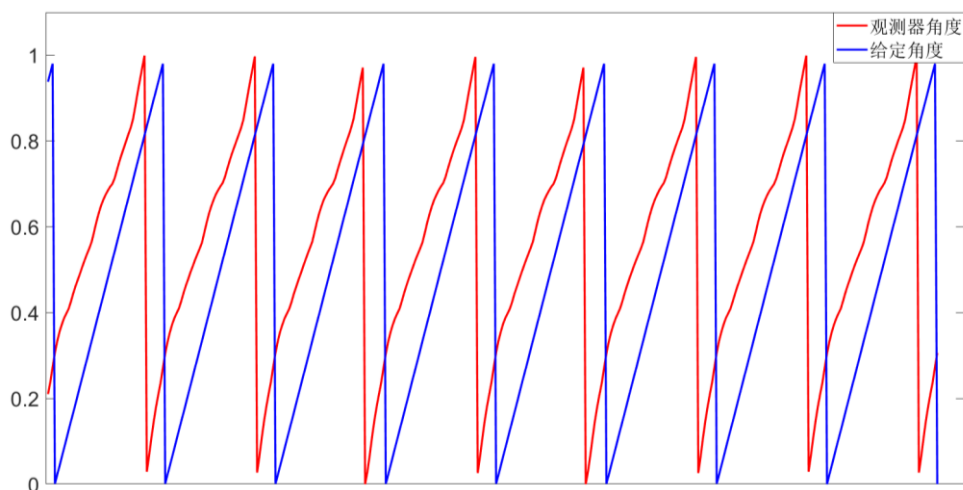


图 5-25 电流闭环下观测器计算角度与给定角度波形

在观测器计算值没有明显问题的情况下，开始速度闭环调试。将切换闭环状态的程序打开，打印此时的角度值并画出波形如图 5-26 所示。可以看出此时观测器计算角度值波形也为三角波，且波形平滑。

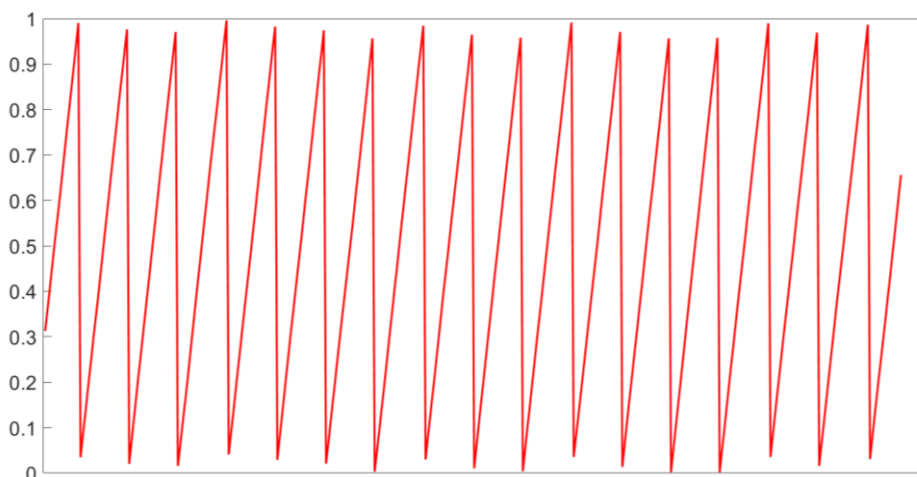


图 5-26 速度闭环下观测器计算角度波形

4) 高频注入

针对电机在低转速下运行，采用脉振高频注入的方法，首先在程序中打开高频注入的代码，观察在 200RPM 转速下，电机能否正常转动，测量电机 U 相电流波形，如图 5-27 所示。

从图中可以看出，在脉振高频注入法下，U 相电流在高频震荡的基础上形成了正弦信号，通过该方法解决了在只采用非线性磁链观测器对永磁同步电机进行矢量控制时，无法获取低转速运行的转子位置和速度信息。

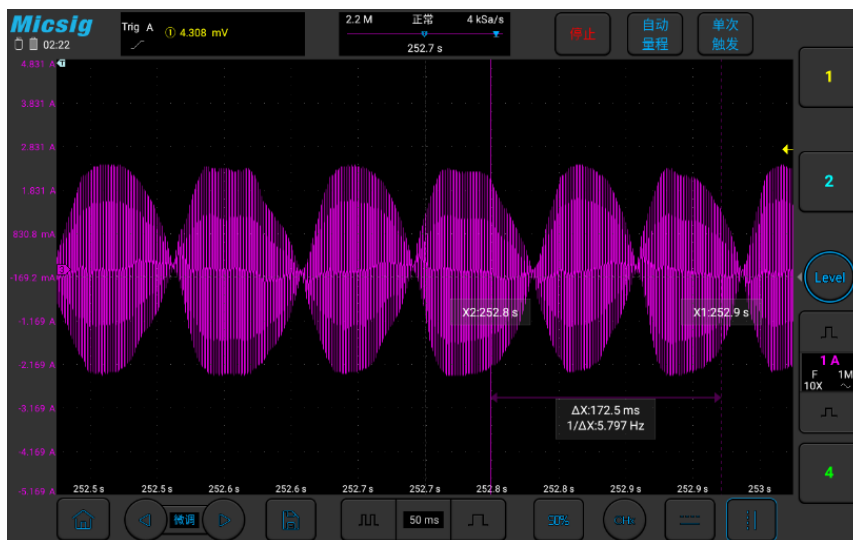


图 5-27 U 相电流输出波形

5) 全速域混合控制与正反转切换

针对电机转动在零低速域和中高速域间的切换，采用混合控制策略，保证在过渡域电机转动流畅，用示波器抓取 U 相电流输出波形如图 5-28 所示。

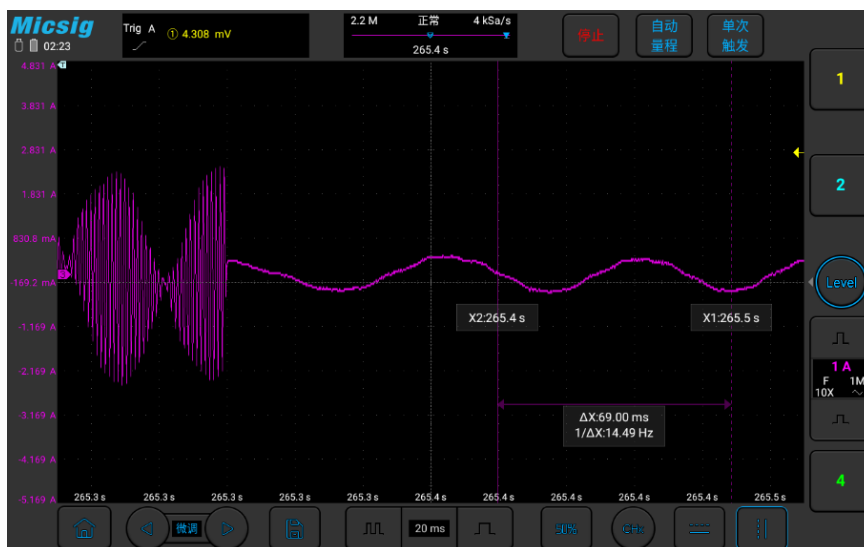


图 5-28 过渡域 U 相电流输出波形

从图中可以看出当电机从高频注入适用的转速切换到磁链观测器适用的转速范围后，电流由原本的高频正弦信号转为正常的低频正弦信号，保证了电机在不同转速下切换的流畅性。

当对电机进行正反转切换时，电机从磁链观测器控制下的中高速域切换到高频注入控制的低速域，然后对电机进行翻转控制，当反转完成之后，再重新将速度提升到中高速域，U 相电流输出波形如图 5-29 所示。

从图中可以看出，当转子转动方向发生改变，电流始终为高频正弦信号，表明这一过程中，高频注入法一直参与对电机的控制，当转速大于过渡域时，仅为正弦信号，由观测器进行控制。同时因为 PI 控制的缘故，当反转之后，转子速度仍小于设定速度，需通过进一步加速达到设定转速，故图中左侧正弦信号频率高于右侧正弦信号频率。

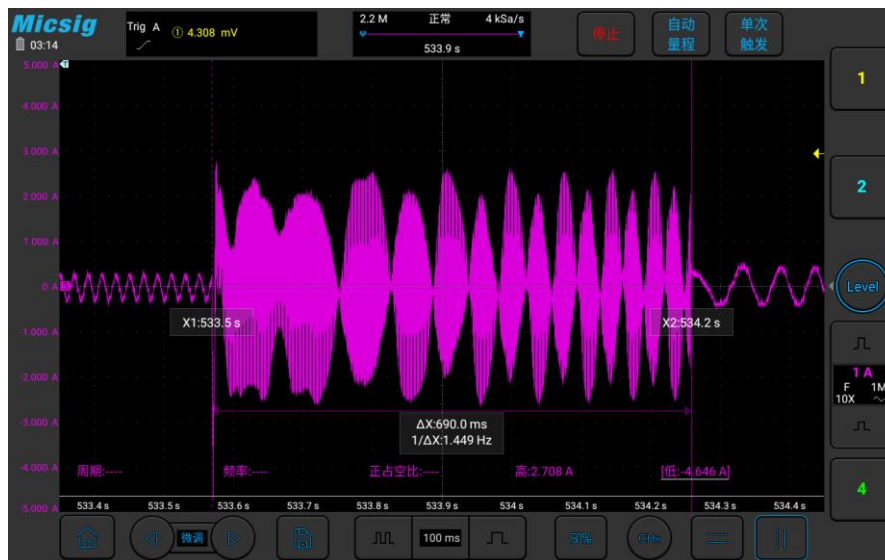


图 5-29 正反转切换下 U 相电流输出波形

5.4.2 电机控制性能验证

本文从三个方面对无位置传感器电机矢量控制进行优化，下面分别针对这三个方面的优化效果进行验证分析。

(1) 三角函数硬件指令优化

根据第三、四章的内容可知，矢量控制中存在大量的三角函数运算，如果采用软件代码进行计算，会增加中断计算时间，采用硬件指令可减少中断处理负担。使用软件查表的中断处理时间如图 5-30 所示，使用硬件指令的中断处理时间如图 5-31 所示。

从图中可以看出，当采用软件查表实现正余弦函数计算，在 ADC 中断中运行整个电流闭环与非线性磁链观测器的程序需要 15.36us，此时芯片运行的主频为 280MHz，即需要约 4300 个周期完成电流闭环控制；而当采用本文设计的多区间二次函数拟合的硬件方案计算正余弦函数，中断运行时间为 13.78us，需要约 3860 个周期。本文设计的方案相比于软件查表，减少了约 440 个时钟周期，对于 PWM 控制频率较高或使用多种外设中断的场景，采用硬件指令能够减少执行周期，提高运行速度。

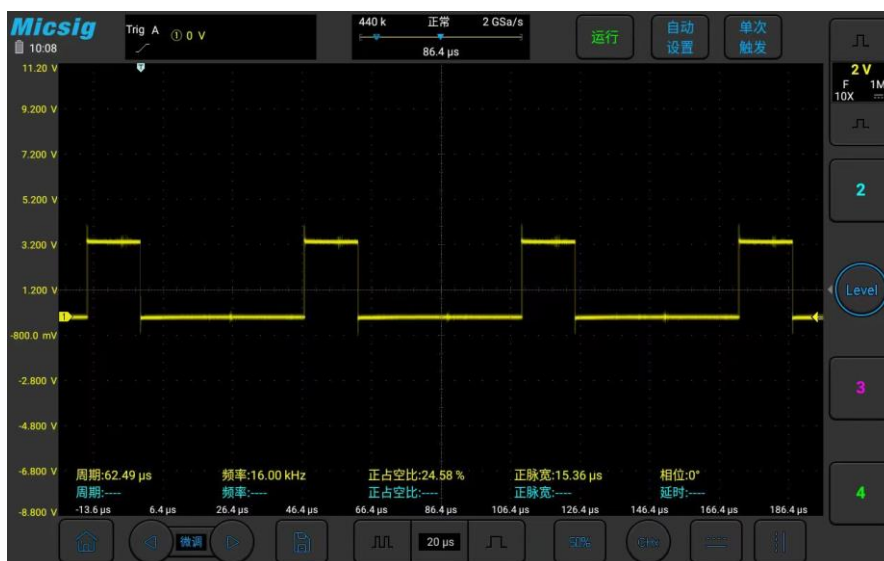


图 5-30 使用软件查表的电流闭环中断处理时间

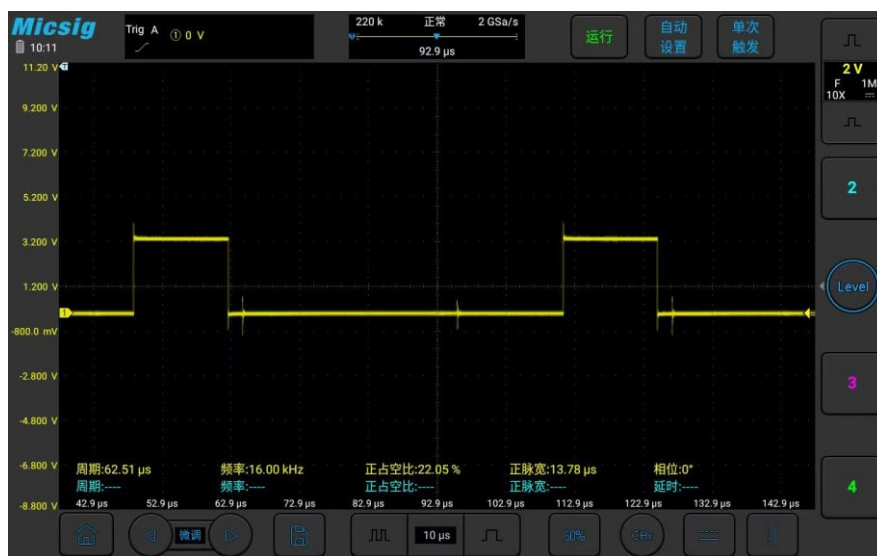


图 5-31 使用硬件指令的电流闭环中断处理时间

(2) 中高速非线性磁链观测器优化

在中高速下，采用锁相环代替原本磁链观测器中反正切函数的计算，实验采用的电机额定转速为 3000RPM，测试在额定转速下，传统磁链观测器估计转速与改进后的观测器转速估计情况分别如图 5-32 和图 5-33 所示。

通过图片可以看出，当电机运行在额定转速下时，采用反正切函数估算转子位置和速度信息，其估计结果与设定转速之间的误差约为 $\pm 2\text{r/min}$ ；采用锁相环代替反正切函数计算转子转速，结果与设定转速的误差约为 $\pm 1\text{r/min}$ 。对于改进后的非线性磁链观测器转速估计误差更小，对电机转速控制更加精确。

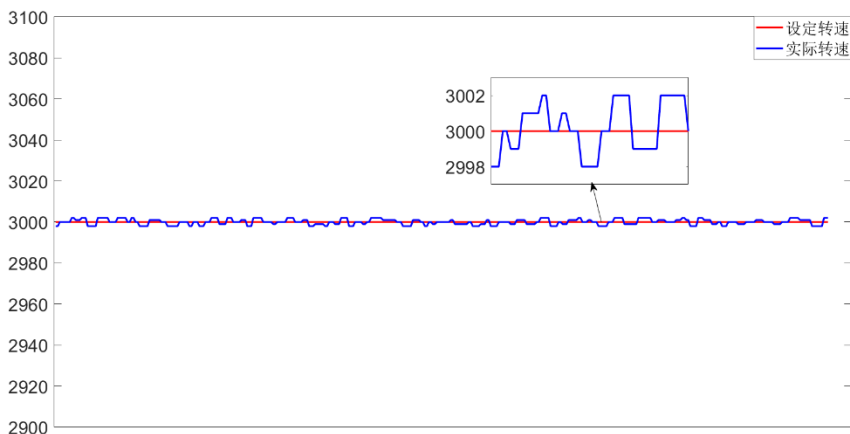


图 5-32 传统磁链观测器设定转速与实际转速波形图

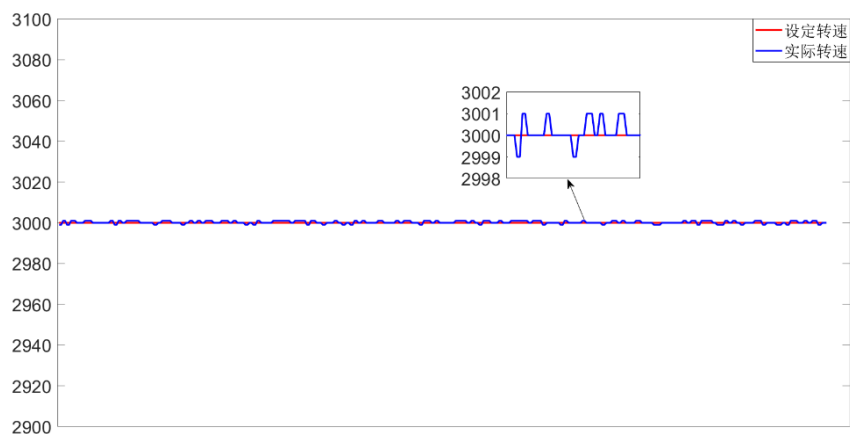


图 5-33 改进磁链观测器设定转速与实际转速波形图

(3) 低速高频注入优化

在低速下，优化高频注入的调制函数、滤波器、PI 控制器，测试电机运行在 200RPM 下，传统高频注入与改进后估计转速分别如图 5-34 和图 5-35 所示。

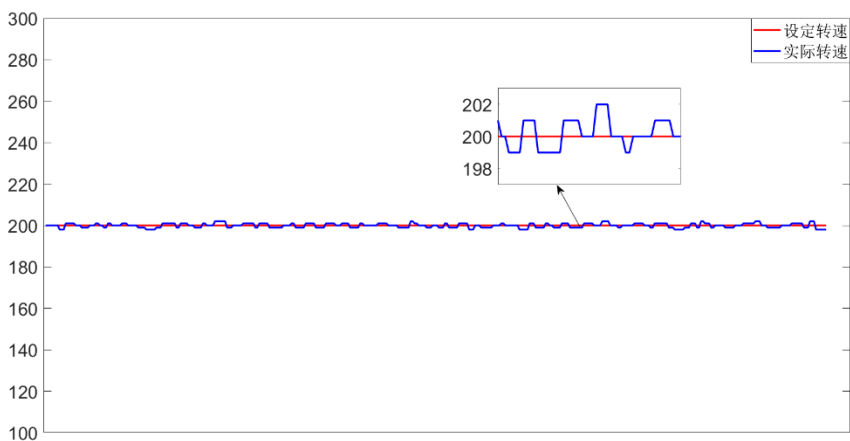


图 5-34 传统高频注入设定转速与实际转速波形图

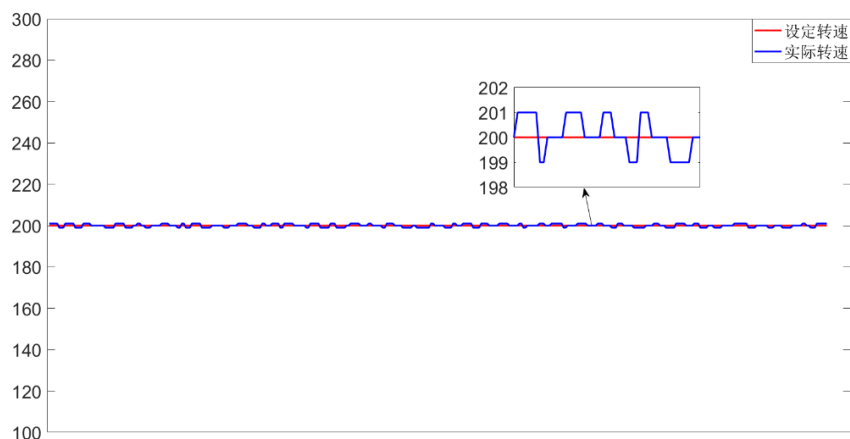


图 5-35 改进高频注入设定转速与实际转速波形图

通过图片可以看出，当电机运行在低转速时，采用优化后的高频注入法，转速误差约为 $\pm 1\text{r/min}$ ，有较好的控制效果。

针对高频注入低转速性能进行测试，发现在 60RPM，即 2Hz 的转速下，无感观测器依然能对转子速度进行有效跟踪，其设定转速与实际转速如图 5-36 所示。

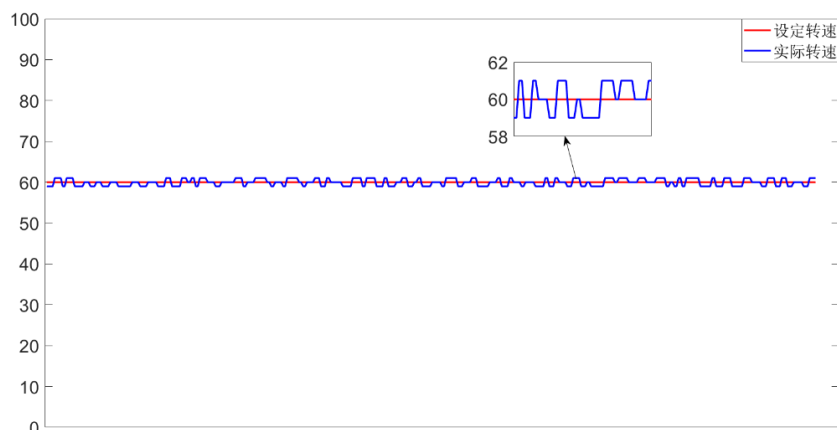


图 5-36 60RPM 下设定转速与实际转速波形图

5.5 本章小结

本章首先介绍了全速域矢量控制的方法，在现有的控制策略基础上提出了改进的混合控制策略，之后搭建以 QX320F280049C 为主控芯片无感控制系统，设计硬件电路与软件程序，并在该系统上对电流、转速双闭环控制进行测试，分析测试了本文提出的硬件指令、磁链观测器、高频注入法的优化方案对系统性能的提升，结果显示该方案能够有效代替机械式传感器进行控制，比传统方法控制稳定性更好。

第6章 总结与展望

6.1 工作总结

本文以永磁同步电机为研究对象,对无位置传感器下矢量控制方法的难点进行分析,提出一种全速域的控制方案,并搭建了电机控制系统,对方案进行验证。论文完成的主要工作如下:

1.分析矢量控制中三角函数计算存在的瓶颈与不足,提出用多区间二次函数拟合的方案实现相应的硬件加速指令,该方案结合查表法与多项式拟合的优点,将指令的定义域划分成512个区间,每个区间用两级乘累加完成计算。为确认硬件定点计算位宽,设计软件仿真程序。仿真得到硬件所需数据后与TI公司的TMU进行对比,证明该方案具有更好的精度更短的执行时间。

2.阐述了电机矢量控制的基本原理,封装了算法中的相关函数库,针对机械式传感器控制的缺点提出了非线性磁链观测器的控制方案。同时分析了传统磁链观测器中,采用的反正切计算中的除法,会将电机磁链中的噪声引入到转子角度值之中,造成电机运行时产生高频抖振。对此问题,利用锁相环对转子位置和速度信息进行解算,保证电机运行的稳定。

3.针对非线性磁链观测器在低速控制下的不足,采用脉振高频注入法,考虑传统高频注入法位置提取使用的调制函数,仅能将高频信号变为基频信号的二次谐波,通过调整调制函数,将其变为四次谐波,并优化滤波器和PI控制器,使从高频电流信号中解算出的转子位置信息更加稳定。

4.利用改进后的混合控制策略将磁链观测器与高频注入结合,实现在全速域内对电机进行矢量控制,设计了以QX320F280049C为主控芯片的永磁同步电机控制系统,验证了本文设计的无位置矢量控制方案的可行性。并设计外设驱动与软件程序,使用户在使用时,仅需调整文件中的电机等相关参数,即可实现对电机电流以及转速双闭环控制。

6.2 未来展望

永磁同步电机作为一种高精度控制电机,由于作者水平和时间等因素限制,本文在方案设计上仍存在一些不足之处,有待后续研究和解决:

1.本文所使用的控制算法需要在代码中添加电机相关参数,同时PI控制器的系数也是根据经验进行调节,当选择一个陌生的电机时,需要经过调试才能保

证正常运行，需要增加对相关参数的自适应识别代码，以保证能对任意电机进行全速域的无感矢量控制。

2.本文设计的控制方案获取的三相电流用的是三电阻采样，对于某些低成本的单电阻场景，需改进算法使其具有通用性。

3.某些场景下用户选择氮化镓作为功率器件，其开关频率能达到 50KHz，对矢量控制算法执行时间要求更高，未来工作可以针对软件算法进行优化，进一步提高计算速度。

参考文献

- [1] QU L, QIAO W, QU L. An enhanced linear active disturbance rejection rotor position sensorless control for permanent magnet synchronous motors[J]. IEEE Transactions on Power Electronics, 2019, 35(6): 6175-6184.
- [2] MOGHADDAM R R, MAGNUSSEN F, SADARANGANI C, et al. Theoretical and experimental reevaluation of synchronous reluctance machine[J]. IEEE Transactions on Industrial Electronics, 2009, 57(1): 6-13.
- [3] BETZ R E, LAGERQUIST R, JOVANOVIĆ M, et al. Control of synchronous reluctance machines [J]. IEEE Transactions on Industry Applications, 1993, 29(6): 1110-1122.
- [4] ZHANG P, ZHANG Q F. A controller of PMSM for electrical bicycle with hall effect sensors[C]//2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA). Hefei: IEEE, 2016:619-623.
- [5] Incze I I, Szabó C, Imecs M. Incremental encoder in electrical drives: modeling and simulation[M]//Rudas I J, Fodor J, Kacprzyk J. Computational Intelligence in Engineering. Berlin Heidelberg: Springer Berlin Heidelberg, 2010: 287-300.
- [6] ZHANG J, LONG Z, WANG C, et al. Compensation modeling and optimization on contactless rotary transformer in rotary ultrasonic machining[J]. Journal of Manufacturing Science and Engineering, 2020, 142(10): 101001.
- [7] JORGENSEN A, MASTERS A, GUHA R. Assurance of Accuracy in Floating-Point Calculations - A Software Model Study[C]//2019 International Conference on Computational Science and Computational Intelligence (CSCI). Las Vegas, NV, USA: IEEE, 2019: 471-475.
- [8] VOLDER J E. The CORDIC trigonometric computing technique[J]. IEEE Transactions on Electronic Computers, 1959, 8(3): 330-334.
- [9] DE D, GHOSH A, KUMAR K G, et al. Multiplier-less hardware realization of trigonometric functions for high speed applications[C]//2018 IEEE Applied Signal Processing Conference (ASPCON). Kolkata, India: IEEE, 2018: 149-152.
- [10] GAO Z Y, ZHAO L P, CHEN H A. A trigonometric function instruction set extension method based on risc-v[C]//2022 IEEE/ACIS 22nd International Conference on Computer and Information Science (ICIS). Zhuhai: IEEE, 2022: 119-126.
- [11] VINH T Q, THANH T B, VIET D H. FPGA Implementation of Trigonometric Function Using Loop-Optimized Radix-4 CORDIC[C]//2022 9th NAFOSTED Conference on Information and Computer Science (NICS). Ho Chi Minh City, Vietnam: IEEE, 2022: 217-222.

- [12] FLYNN M J. Very high-speed computing systems[J]. Proceedings of the IEEE, 1966, 54(12): 1901-1909.
- [13] ESTRIN G. Organization of computer systems: the fixed plus variable structure computer[C]//Papers Presented at the May 3-5, 1960, Western Joint IRE-AIEE-ACM Computer Conference. 1960: 33-40.
- [14] WANG H, WU S, WANG Q. Global sliding mode control for nonlinear vehicle antilock braking system[J]. IEEE Access, 2021, 9: 40349-40359.
- [15] 祝新阳, 曾国辉, 黄勃, 等.改进滑模观测器的永磁同步电机矢量控制[J].信息与控制, 2020, 49(6): 708-713, 721.
- [16] 朱喜华, 李颖晖, 张敬.基于一种新型滑模观测器的永磁同步电机无传感器控制[J].电力系统保护与控制, 2010, 38(13): 6-10.
- [17] ABO-KHALIL A G, ELTAMALY A M, ALSAUD M S, et al. Sensorless control for PMSM using model reference adaptive system[J]. International Transactions on Electrical Energy Systems, 2021, 31(2): e12733.
- [18] 谷善茂, 何凤有, 谭国俊, 等.永磁同步电动机无传感器控制技术现状与发展[J].电工技术学报, 2009, 24(11): 14-20.
- [19] ERAZO D E G, WALLSCHEID O, BÖCKER J. Improved fusion of permanent magnet temperature estimation techniques for synchronous motors using a Kalman filter[J]. IEEE Transactions on Industrial Electronics, 2019, 67(3): 1708-1717.
- [20] YANG R, WANG M, LI L, et al. Robust predictive current control of PMLSM with extended state modeling based Kalman filter: For time-varying disturbance rejection[J]. IEEE Transactions on Power Electronics, 2019, 35(2): 2208-2221.
- [21] LORENZ R D. Practical issues and research opportunities when implementing zero speed sensorless control[C]//ICEMS'2001. Proceedings of the Fifth International Conference on Electrical Machines and Systems. Shenyang: IEEE, 2001, 1: 1-10.
- [22] 王子辉.永磁同步电机全速度范围无位置传感器控制策略研究[D].杭州: 浙江大学, 2012.
- [23] WANG G, YANG R, XU D. DSP-based control of sensorless IPMSM drives for wide-speed-range operation[J]. IEEE Transactions on Industrial Electronics, 2012, 60(2): 720-727.
- [24] JANG J H, SUL S K, HA J I, et al. Sensorless drive of surface-mounted permanent-magnet motor by high-frequency signal injection based on magnetic saliency[J]. IEEE Transactions on Industry Applications, 2003, 39(4): 1031-1039.
- [25] JEONG Y S, LORENZ R D, JAHNS T M, et al. Initial rotor position estimation of an interior permanent-magnet synchronous machine using carrier-frequency injection methods[J]. IEEE Transactions on Industry Applications, 2005, 41(1): 38-45.

-
- [26] KANG J, KIM T, SHIN H, et al. Direct energy recovery system for membrane capacitive deionization[J]. *Desalination*, 2016, 398: 144-150.
- [27] GUO J, ZHAO Y, WANG H. DSP practical solutions for motor control using DSP-Controller[C]//2010 International Conference on Networking and Digital Society. Wenzhou: IEEE, 2010, 2: 629-632.
- [28] ASAI H, OMURA M, SHIMONO T, et al. Bilateral control of a half-circle-shaped tubular linear motor with disturbance model based on trigonometric function of two variables[C]//7th International Conference on Information and Automation for Sustainability. Colombo, Sri Lanka: IEEE, 2014: 1-6.
- [29] SHI M, WANG Q, LI G, et al. A new adaptive analytical model for the spherical reluctance motor based on hybrid trigonometric function-power function[J]. *IEEE Transactions on Industrial Electronics*, 2022, 70(6): 6099-6109.
- [30] EL OUANJLI N, DEROUICH A, EL GHZIZAL A, et al. A comparative study between FOC and DTC control of the Doubly Fed Induction Motor (DFIM)[C]//2017 International Conference on Electrical and Information Technologies (ICEIT). Rabat, Morocco: IEEE, 2017: 658-692.
- [31] MEHAZZEM F, NEMMOUR A L, REAMA A. Real time implementation of backstepping-multiscalar control to induction motor fed by voltage source inverter[J]. *International Journal of Hydrogen Energy*, 2017, 42(28): 17965-17975.
- [32] OLESCHUK V, BLAABJERG F, BOSE B K. Analysis and comparison of algebraic and trigonometric methods of synchronous PWM for inverter drives[C]//2002 IEEE 33rd Annual IEEE Power Electronics Specialists Conference. Proceedings (Cat. No.02CH37289). Cairns, Australia: IEEE, 2002, 3: 1439-1444.
- [33] ANONYMITY. IEEE Standard for Floating-Point Arithmetic[C]//IEEE Std 754-2019 (Revision of IEEE 754-2008). IEEE, 2019: 1-84.
- [34] LAHARI P L, BHARATHI M, SHIRUR Y J M. High Speed Floating Point Multiply Accumulate Unit using Offset Binary Coding[C]//2020 7th International Conference on Smart Structures and Systems (ICSSS). Chennai, India: IEEE, 2020: 1-5.
- [35] WALLACE C S. A suggestion for a fast multiplier[J]. *IEEE Transactions on Electronic Computers*, 1964, EC-13(1): 14-17.
- [36] BOOTH A D. A signed binary multiplication technique[J]. *The Quarterly Journal of Mechanics and Applied Mathematics*, 1951, 4(2): 236-240.
- [37] DONG Z, LIU C, SONG Z, et al. Suppression of dual-harmonic components for five-phase series-winding PMSM[J]. *IEEE Transactions on Transportation Electrification*, 2021, 8(1):

- 121-134.
- [38] WU G, HUANG S, WU Q, et al. Predictive torque and stator flux control for N^* 3-phase PMSM drives with parameter robustness improvement[J]. IEEE Transactions on Power Electronics, 2020, 36(2): 1970-1983.
- [39] QUAN L, WANG Z, LIU X. Sensorless control of SPMSM using complex number model based position estimation and EKF[C]//The 26th Chinese Control and Decision Conference (2014 CCDC). Changsha: IEEE, 2014: 2663-2668.
- [40] LIAO G, XU W, TANG Y, et al. Speed Model Predictive Control for SPMSM Based on EKF Observer[C]//2023 IEEE International Conference on Predictive Control of Electrical Drives and Power Electronics (PRECEDE). Wuhan: IEEE, 2023: 1-6.
- [41] ZHANG Y, QI R. High-Efficiency Drive for IPMSM Based on Model Predictive Control[C]//2021 24th International Conference on Electrical Machines and Systems (ICEMS). Gyeongju, Korea, 2021: 588-592.
- [42] LIU T H, CHEN Y, DAI B C. MTPA control for an IPMSM drive system using high frequency injection method[C]//2016 IEEE International Conference on Industrial Technology (ICIT). Taipei, Taiwan: IEEE, 2016: 181-186.
- [43] ZHANG Y, QI R. High-Efficiency Drive for IPMSM Based on Model Predictive Control[C]//2021 24th International Conference on Electrical Machines and Systems (ICEMS). Gyeongju, Korea: IEEE, 2021: 588-592.
- [44] ZHENG Z, ZONG X. Control method based on nonlinear extended state observer to tracking the trajectory for mechanical arm[C]//2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). Chongqing: IEEE, 2021: 673-677.
- [45] TANG Q, SHEN A, LUO P, et al. IPMSMs sensorless MTPA control based on virtual q-axis inductance by using virtual high-frequency signal injection[J]. IEEE Transactions on Industrial Electronics, 2019, 67(1): 136-146.
- [46] BIAN P, WANG B, XU Q, et al. Research on the Control of Permanent Magnet Synchronous Motor Based on High Frequency Injection and Optimized Sliding Mode Controller[C]//Proceedings of 2019 Chinese Intelligent Systems Conference: Volume II 15th. Singapore: Springer, 2020: 83-90.
- [47] LU Q, GAO S, ZHANG T, et al. Rotor Position Estimation Strategy Based on Rotating High-Frequency Voltage Injection Using Synchronous Frequency Filter[J]. Applied Sciences, 2022, 12(23): 11945.
- [48] SHUANG B, ZHU Z, WU X. Improved cross-coupling effect compensation method for sensorless control of IPMSM with high frequency voltage injection[J]. IEEE Transactions on

- Energy Conversion, 2021, 37(1): 347-358.
- [49] GUO L, YANG Z, LIN F. A novel strategy for sensorless control of IPMSM with error compensation based on rotating high frequency carrier signal injection[J]. Energies, 2020, 13(8): 1919.
- [50] TANG Q, SHEN A, LUO X, et al. IPMSM sensorless control by injecting bidirectional rotating HF carrier signals[J]. IEEE Transactions on Power Electronics, 2018, 33(12): 10698-10707.
- [51] MESSALI A, HAMIDA M A, GHANES M, et al. Estimation procedure based on less filtering and robust tracking for a self-sensing control of IPMSM[J]. IEEE Transactions on Industrial Electronics, 2020, 68(4): 2865-2875.
- [52] MAI Z, XIAO F, FU K, et al. HF pulsating carrier voltage injection method based on improved position error signal extraction strategy for PMSM position sensorless control[J]. IEEE Transactions on Power Electronics, 2021, 36(8): 9348-9360.
- [53] WOLDEGIORGIS A T, GE X, LI S, et al. An improved sensorless control of IPMSM based on pulsating high-frequency signal injection with less filtering for rail transit applications[J]. IEEE Transactions on Vehicular Technology, 2021, 70(6): 5605-5617.
- [54] SUN X, HU C, LEI G, et al. Speed sensorless control of SPMSM drives for EVs with a binary search algorithm-based phase-locked loop[J]. IEEE Transactions on Vehicular Technology, 2020, 69(5): 4968-4978.

致 谢

三年的硕士学习生活很快就要结束，这三年不论是在学习上还是个人能力等方面都有着较大的进步，也取得了一定的成果。这些成绩也离不开导师、同学们的帮助。在这篇论文的致谢部分，我想衷心感谢那些在我学术生涯中给予我帮助与鼓励的人们。

首先我想感谢我的实践导师吴俊教授，吴老师以其博学和谆谆教诲成为我学术道路上的明灯。吴老师扎根数字信号处理多年，对我论文中三角函数指令方案设计提供了耐心细致的指导，同时还在生活上、职业道路方面给予了宝贵的建议。吴老师的悉心教导让我受益匪浅，十分感谢老师对我的教导。

感谢我的校内导师赵云波教授，在最后论文提交阶段，赵老师细细翻阅我的论文，对其中内容章节安排不合理的部分指出修改意见，也指导了我论文写作的相关技巧，保证了最后论文能成功通过盲审。

感谢董嘉禹、沈文祥师兄，在我刚进入实验室，对数字信号处理与嵌入式开发了解不多的时候，是他们分享了自己的经验，告诉我如何去发现问题解决问题。

最后，衷心感谢所有支持我、帮助我、鼓励我的人们，因为有你们，我的硕士研究生学业才得以圆满结束。

附录 1 浮点转定点仿真代码

```
1. void F32toIQ(float x, int *IQ_x) {
2.     memset(IQ_x, 0, ArrayWidth * sizeof(int));
3.     unsigned int *p = &x;
4.     unsigned int hex = *p;
5.     unsigned int e = (hex & 0x7F800000) >> 23;
6.     int exp = 127 - (int)e;
7.     unsigned long long man = (hex & 0x7FFFFFFF) + 0x800000;
8.     int number = exp + 26;
9.     int len = 0;
10.    unsigned long long num;
11.    if (number <= MulWidth) {
12.        SetArray(man, &IQ_x[exp], number, 39);
13.        num = LLSetNum(IQ_x, number, 0);
14.    } else {
15.        len = MulWidth - exp;
16.        if (len < 0)
17.            len = -1;
18.        SetArray(man, &IQ_x[exp], len + 1, 39);
19.        num = LLSetNum(IQ_x, MulWidth, 0);
20.        unsigned int count = SetNum(IQ_x, 1, MulWidth);
21.        num += count;
22.    }
23.    memset(IQ_x, 0, ArrayWidth * sizeof(int));
24.    if (number <= MulWidth)
25.        SetArray(num, IQ_x, MulWidth, 64 - number + 1);
26.    else
27.        SetArray(num, IQ_x, MulWidth, 64 - MulWidth + 1);
28. }
```

附录 2 定点转浮点仿真代码

```
1. unsigned int IQtoF32(int *IQ_x) {
2.     unsigned long long hex = LLSetNum(IQ_x, BitWidth, 0);
3.     if (IQ_x[0])
4.         hex = ~(hex - 1);
5.     hex = (hex << (64 - BitWidth)) >> (64 - BitWidth);
6.     long long num = (long long)hex;
7.     int count = 0;
8.     while (num) {
9.         count++;
10.        num >>= 1;
11.    }
12.    num = (long long)hex;
13.    unsigned int exp = (127 - (BitWidth - count) + 2) << 23;
14.    unsigned int flag = (IQ_x[0]) << 31;
15.    unsigned long long number = 0x1;
16.    unsigned long long llman = (unsigned long long)num - (number << (count - 1));
17.    int a = count - 24;
18.    number = 0x1;
19.    unsigned int check = 0;
20.    if (a > 0) {
21.        number <<= (a - 1);
22.        number &= llman;
23.        if (number)
24.            check = 1;
25.        llman >>= a;
26.    } else if (a < 0) {
27.        llman <<= (-a);
28.    }
29.    unsigned int man = (unsigned int)llman + check;
30.    return exp + flag + man;
31. }
```


附录 3 乘累加仿真代码

```
1. void IQadd(int *IQ_x, int *IQ_y, int *IQ_ret) {
2.     unsigned long long hex_x = LLSetNum(IQ_x, BitWidth, 0);
3.     unsigned long long hex_y = LLSetNum(IQ_y, BitWidth, 0);
4.     unsigned long long hex_ret = hex_x + hex_y;
5.     SetArray(hex_ret, IQ_ret, BitWidth, 64 - BitWidth + 1);
6. }
7.
8. //采用 booth2 计算乘法
9. void IQmul(int *IQ_x, int *IQ_y, int *IQ_ret) {
10.    int IQ_temp_x[ArrayWidth], IQ_temp_y[ArrayWidth], IQ_temp_negx[ArrayWidth],
11.        IQ_ACCMQ[ArrayWidth];
12.    memset(IQ_temp_negx, 0, ArrayWidth * sizeof(int));
13.    memset(IQ_ACCMQ, 0, ArrayWidth * sizeof(int));
14.    //取 x 的 MulWidth 位到 IQ_temp_x 中, 并四舍五入
15.    IQcut1(IQ_x, IQ_temp_x, BitWidth, MulWidth);
16.    //取 y 的 MulWidth 位到 IQ_temp_y 中, 并四舍五入
17.    IQcut(IQ_y, IQ_temp_y, BitWidth, MulWidth);
18.    //获得[x]补
19.    ShiftRight(IQ_temp_x, MulWidth, 1);
20.    unsigned long long numx = LLSetNum(IQ_temp_x, MulWidth + 1, 0);
21.    //获得[-x]补
22.    numx = ~numx + 1;
23.    SetArray(numx, IQ_temp_negx, MulWidth + 1, 64 - MulWidth);
24.    unsigned long long negnumx = LLSetNum(IQ_temp_negx, MulWidth + 1, 0);
25.    numx = ~(numx - 1);
26.    //获得[y]补
27.    IQ_temp_y[MulWidth] = 0;
28.    // booth2
29.    int n = MulWidth - 1;
30.    memcpy(&IQ_ACCMQ[n + 2], IQ_temp_y, (n + 2) * sizeof(int));
31.    for (int i = 1; i <= n + 1; i++) {
32.        unsigned int num = SetNum(IQ_ACCMQ, 2, 2 * MulWidth);
33.        unsigned long long number = 0;
34.        //部分积右移一位
35.        if ((num == 0 || num == 3) && i != n + 1) {
36.            ShiftRight(IQ_ACCMQ, 2 * MulWidth + 1, 1);
37.        }
38.        //部分积加[x]补, 右移一位
39.        else if (num == 1) {
40.            number = LLSetNum(IQ_ACCMQ, MulWidth + 1, 0);
41.            number += numx;
```

```
42.     SetArray(number, IQ_ACCMQ, MulWidth + 1, 64 - MulWidth);
43.     if (i != n + 1)
44.         ShiftRight(IQ_ACCMQ, 2 * MulWidth + 1, 1);
45.     }
46.     //部分积加[-x]补, 右移一位
47.     else if (num == 2) {
48.         number = LLSetNum(IQ_ACCMQ, MulWidth + 1, 0);
49.         number += negnumx;
50.         SetArray(number, IQ_ACCMQ, MulWidth + 1, 64 - MulWidth);
51.         if (i != n + 1)
52.             ShiftRight(IQ_ACCMQ, 2 * MulWidth + 1, 1);
53.     }
54. }
55. unsigned long long ret = LLSetNum(IQ_ACCMQ, BitWidth + 1 + 3, 0);
56. SetArray(ret, IQ_ret, BitWidth, 64 - BitWidth);
57. }
```

附录 4 sin 指令区间划分与二次函数系数

注：由于表格含有 512 区间的系数，这里仅展示前 30 个区间的划分结果与响应系数。

左区间	右区间	C0	C1	C2
0x00000000	0x3affffff	0x3FFFFFFF	0x0C90FE643	0x3FFC1FCD
0x3b000000	0x3b7fffff	0x3FFFFFFE0	0x0C9106249	0x3FF45F69
0x3b800000	0x3bbfffff	0x3FFFFFF63	0x0C9115A55	0x3FEC9F0C
0x3bc00000	0x3bfffff	0x3FFFFFFE4C	0x0C912CE64	0x3FE4DEBB
0x3c000000	0x3c1fffff	0x3FFFFFFC5B	0x0C914BE74	0x3FDD1E7A
0x3c200000	0x3c3fffff	0x3FFFFFF953	0x0C9172A81	0x3FD55E50
0x3c400000	0x3c5fffff	0x3FFFFFF4F7	0x0C91A1287	0x3FCD9E3F
0x3c600000	0x3c7fffff	0x3FFFFFFE07	0x0C91D7680	0x3FC5DE4E
0x3c800000	0x3c8fffff	0x3FFFFFFE747	0x0C9215667	0x3FBE1E80
0x3c900000	0x3c9fffff	0x3FFFDD77	0x0C925B233	0x3FB65EDB
0x3ca00000	0x3cafffff	0x3FFFDD15C	0x0C92A89DE	0x3FAE9F64
0x3cb00000	0x3cbfffff	0x3FFFDC2B5	0x0C92FDD5D	0x3FA6E01F
0x3cc00000	0x3ccfffff	0x3FFFDB147	0x0C935ACA8	0x3F9F2110
0x3cd00000	0x3cdfffff	0x3FFF9CD2	0x0C93BF7B2	0x3F97623E
0x3ce00000	0x3cefffff	0x3FFF851A	0x0C942BE72	0x3F8FA3AC
0x3cf00000	0x3cfffff	0x3FFF69E0	0x0C94A00D9	0x3F87E55F
0x3d000000	0x3d07ffff	0x3FFF4AE8	0x0C951BEDC	0x3F80275D
0x3d080000	0x3d0fffff	0x3FFF27F2	0x0C959F86A	0x3F7869A9
0x3d100000	0x3d17ffff	0x3FFF00C3	0x0C962AD76	0x3F70AC49
0x3d180000	0x3d1fffff	0x3FFFED51C	0x0C96BDDEF	0x3F68EF41
0x3d200000	0x3d27ffff	0x3FFFEA4C0	0x0C97589C3	0x3F613297
0x3d280000	0x3d2fffff	0x3FFF6F72	0x0C97FB0E2	0x3F59764E
0x3d300000	0x3d37ffff	0x3FFF34F5	0x0C98A5338	0x3F51BA6C
0x3d380000	0x3d3fffff	0x3FFFDF50B	0x0C99570B1	0x3F49FEF6
0x3d400000	0x3d47ffff	0x3FFFD7AF77	0x0C9A1093A	0x3F4243F0
0x3d480000	0x3d4fffff	0x3FFF63FC	0x0C9AD1CBC	0x3F3A895F
0x3d500000	0x3d57ffff	0x3FFF125E	0x0C9B9AB21	0x3F32CF48
0x3d580000	0x3d5fffff	0x3FFFCBA60	0x0C9C6B452	0x3F2B15AF
0x3d600000	0x3d67ffff	0x3FFFC5BC4	0x0C9D43837	0x3F235C9A
0x3d680000	0x3d6fffff	0x3FFF6F64E	0x0C9E236B6	0x3F1BA40D

附录 5 atan 指令区间划分与二次函数系数

注：由于表格含有 512 区间的系数，这里仅展示前 30 个区间的划分结果与响应系数。

左区间	右区间	C0	C1	C2
0x00000000	0x3affffff	0x000000000	0x145F31624	0xFFEBA0D
0x3b000000	0x3b7fffff	0x3FFFFFFD6	0x145F3B91C	0xFFC2E2C
0x3b800000	0x3bbfffff	0x3FFFFFF32	0x145F4FF07	0xFF9A255
0x3bc00000	0x3bfffff	0x3FFFFFFDC3	0x145F6E7D9	0xFF71692
0x3c000000	0x3c1fffff	0x3FFFFFFB36	0x145F97381	0xFF48AEA
0x3c200000	0x3c3fffff	0x3FFFFFF73B	0x145FCA1E8	0xFF1FF64
0x3c400000	0x3c5fffff	0x3FFFFFF181	0x1460072F1	0xFE7408
0x3c600000	0x3c7fffff	0x3FFFFE9B6	0x14604E67A	0xFECE8DD
0x3c800000	0x3c8fffff	0x3FFFFDF89	0x14609FC5B	0FEA5DEC
0x3c900000	0x3c9fffff	0x3FFFFD2AA	0x1460FB467	0xFE7D33B
0x3ca00000	0x3cafffff	0x3FFFFC2C9	0x146160E69	0xFE548D4
0x3cb00000	0x3cbfffff	0x3FFFFAF93	0x1461D0A29	0xFE2BEBE
0x3cc00000	0x3ccfffff	0x3FFFF98BB	0x14624A768	0xFE034FB
0x3cd00000	0x3cdfffff	0x3FFFF7DEF	0x1462CE5E2	0FDDAB9A
0x3ce00000	0x3cefffff	0x3FFFF5EE0	0x14635C54C	0xFDB22A0
0x3cf00000	0x3cfffff	0x3FFFF3B3F	0x1463F4556	0xFD89A14
0x3d000000	0x3d07ffff	0x3FFFF12BD	0x1464965AC	0xFD611FE
0x3d080000	0x3d0fffff	0x3FFFFE50B	0x1465425F3	0xFD38A65
0x3d100000	0x3d17ffff	0x3FFFFB1DB	0x1465F85C9	0xFD10352
0x3d180000	0x3d1fffff	0x3FFFFE78DF	0x1466B84C9	0FCE7CCA
0x3d200000	0x3d27ffff	0x3FFFFE39CA	0x146782287	0FCBF6D7
0x3d280000	0x3d2fffff	0x3FFFFDF44E	0x146855E93	0FC9717F
0x3d300000	0x3d37ffff	0x3FFFFDA81E	0x146933875	0FC6ECC9
0x3d380000	0x3d3fffff	0x3FFFFD54F0	0x146A1AFB1	0FC468BE
0x3d400000	0x3d47ffff	0x3FFFCFA77	0x146B0C3C6	0FC1E565
0x3d480000	0x3d4fffff	0x3FFFC9868	0x146C0742D	0FBF62C4
0x3d500000	0x3d57ffff	0x3FFFC2E78	0x146D0C059	0FBCE0E4
0x3d580000	0x3d5fffff	0x3FFFBBC5D	0x146E1A7B9	0FBA5FCC
0x3d600000	0x3d67ffff	0x3FFFB41CF	0x146F329B5	0FB7DF83
0x3d680000	0x3d6fffff	0x3FFFA8E83	0x1470545B1	0FB56010

附录 6 sin 指令移位区间与移位值

sin 区间	MoveBit	C1	C2
0	3	0x01921FCC8	0x3FFFF07F
1	3	0x019220C49	0x3FFFD17E
2	3	0x019222B4B	0x3FFFB27C
3	4	0x00C912CE6	0x3FFFE4DF
4	3	0x0192297CE	0x3FFF747A
5	3	0x01922E550	0x3FFF5579
6	3	0x019234251	0x3FFF3679
7	3	0x01923AED0	0x3FFF1779
8	2	0x03248559A	0x3FFBE1E8
9	2	0x032496C8D	0x3FFB65EE
10	2	0x0324AA277	0x3FFAE9F6
11	2	0x0324BF757	0x3FFA6E02
12	2	0x0324D6B2A	0x3FF9F211
13	2	0x0324EFDED	0x3FF97624
14	2	0x03250AF9C	0x3FF8FA3B

附录 7 atan 指令移位区间与移位值

atan 区间	MoveBit	C1	C2
0	5	0x00A2F98B1	0xFFFFFAF
1	3	0x028BE7724	0xFFFF0B9
2	3	0x028BE9FE1	0xFFFE689
3	3	0x028BEDCFB	0xFFFD5A
4	3	0x028BF2E70	0xFFFD22C
5	2	0x0517F287A	0xFFF1FF6