

**Subject : CZ Speed Control System of Single Crystal Silicon on PID
Neural Network**

Specialty : Power Electronics & Power Drives

Name : Ding Hongwei

(Signature) Ding Hongwei

Instructor : Guo Xiucan

(Signature) Guo Xiucan

Abstract

It is well-known that the symmetry of IC is better than separate element's. Besides the difference of techniques in the process of producing IC, another reason is the performance of its raw materials which is single crystal silicon. Its performance is relative to equipment's performance, the transportation of quantity of heat and crystal imperfection. The single crystal silicon is produced by CZ in the crystal develop stove. And the CZ speed will directly affect the transportation of quantity of heat and crystal defect in the CZ process, thereby affect the performance of material.

When the single crystal silicon is produced by CZ, its weight is increased continuous. The high performance of diameter requires the speed control system faster dynamic respond rapid, smaller overshoot and stronger anti-disturbance. The performance of speed control system is related to the structure of the system, the performance of components and the control method and strategy. The control strategies are not the same in the speed control system of present single crystal silicon. For example PID as well as the improvement PID control, the fuzzy control and so on, these control methods also have their own characteristics. These control strategies solve the problem of the quality of single crystal silicon to some extent. But the quality of single crystal silicon which is obtained by these control methods is not perfect.

Firstly, this article analysis the quality of heat transportation in the crystal process as well as the crystal defect, finds the main factors which affect the quality of single crystal silicon in its CZ process. Secondly, apply one kind of new control method named PID neural network and build a model of the CZ system of single crystal silicon in order to improvement the quality of it. Thirdly, this system is feasible by simulating based on MATLAB/SIMULINK. Lastly, design the hardware and some software the of the SPIDNN control system.

Key words: Single crystal silicon CZ speed control of single crystal silicon
PID neural network

Thesis : Applied Research

西安科技大学

学位论文独创性说明

本人郑重声明：所呈交的学位论文是我个人在导师指导下进行的研究工作及取得研究成果。尽我所知，除了文中加以标注和致谢的地方外，论文中不包含其他人或集体已经公开发表或撰写过的研究成果，也不包含为获得西安科技大学或其他教育机构的学位或证书所使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中做了明确的说明并表示了谢意。

学位论文作者签名：丁红卫 日期：2007.6.20

学位论文知识产权声明书

本人完全了解学校有关保护知识产权的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属于西安科技大学。学校有权保留并向国家有关部门或机构送交论文的复印件和电子版。本人允许论文被查阅和借阅。学校可以将本学位论文的全部或部分内容编入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和汇编本学位论文。同时本人保证，毕业后结合学位论文研究课题再撰写的文章一律注明作者单位为西安科技大学。

保密论文待解密后适用本声明。

学位论文作者签名：丁红卫

指导教师签名：郭秀才

2007年6月20日

1 绪论

1.1 单晶体的发展

制造半导体器件的材料，主要是单晶体，包括体单晶和薄膜单晶。单晶体的质量问题对于半导体材料研制，是一个极为重要的问题。

晶体生长作为一个学科领域，历史很久。近百年来，晶体生长在理论与实践两方面都有了很大的发展。主要包括：20 世纪 20 年代柯塞尔（Kossel）等人提出的完整晶体生长微观理论模型；40 年代费兰克（Frank）发展了的缺陷晶体生长理论；50 年代后伯顿（Burton）等人在晶体生长及其界面的平衡结构理论以及杰克逊（Jackson）平衡界面理论等方面的研究。这些的研究使晶体生长从生长技术进入了微观理论定性半定量的研究阶段。近年来计算机技术的广泛应用，使晶体生长理论研究向微观定量计算大步前进^[1]。

自 20 世纪 50 年代以来，人工晶体对发展高新技术的作用越来越大，人工晶体的品种也越来越多，对晶体生长技术要求越来越高，所涉及的领域也越来越广。

单晶硅是电子工业的“食粮”，目前仍在半导体材料中占首位，年产量达数千吨，在已实现产业化的晶体材料中名列前茅，其中约 80% 单晶硅是提拉出来的，主要用于大规模集成电路（LSI）和超大规模集成电路（VLSI）的芯片。目前，以硅为基础材料的众多器件已成为电子器件的主体。

在过去半个世纪里，无论是提拉单晶硅的完整性、均匀性和纯度，还是提拉单晶硅的直径和长度都得到了不断的提高和壮大。目前国外在实验室拉制的最大硅单晶直径已经达到 300mm。

1961 年，在中国科学院半导体物理研究所林兰院士的亲自指挥下，北京机械学院工厂（西安理工大学工厂的前身）的技术人员与半导体物理研究所的技术人员共同研制出了我国第一台人工晶体生长设备—TDK-36 型单晶炉，并且成功制出了我国第一根无位错的硅单晶，单晶质量接近当时的国际先进水平，且 TDK-36 型单晶炉荣获国家级新产品奖。TDK-36 型单晶炉投料量只有 1kg，拉制单晶直径 $\phi 35\text{mm}$ 。1973 年开发了 TDK-40 型单晶炉，投料量 3kg，拉制单晶直径 $\phi 50\text{mm}$ 。20 世纪 80 年代后期，我国半导体工业迅速发展，国内半导体材料制造厂家大量引进美国 KAYCX-CG3000 型软轴提拉单晶炉。为满足我国半导体工业不断发展的需要，1988 年西安理工大学工厂承担了国家“七五”科技攻关项目，研制成功了 TDR-62 系列软轴单晶炉，投料量增值 30kg，拉制单晶直径 $\phi 125\text{mm}$ 。该炉型采用软轴提拉，大大降低了设备高度^[2]。

近年来，随着半导体产业的发展，国内晶体生长炉也正在向大直径迈进。

1.2 单晶硅提拉概况

单晶硅的提拉是在单晶生长炉中进行的，提拉的方法又叫直拉法（Czochralski, CZ 法），单晶生长炉结构如图 1.1 所示。炉体由下面的加热室和上面的主炉室组成。整个系统包括提拉速度控制系统、引线旋转系统、坩锅旋转与上升系统、加热系统、炉内气氛系统、观察窗以及熔融硅七部分。

提拉速度控制系统主要由钨丝、籽晶保持器、IRCON 光学高温计等组成，主要实现单晶硅的提拉与速度控制功能。

引线旋转系统由旋转电机等组成，用于使单晶硅在上升过程中作旋转运动，以加速接触面的散热。

坩锅旋转与上升系统主要包括石墨坩锅、石英坩锅、坩锅支撑轴等，其作用是使坩锅上升，以保持熔融硅液面始终在同一水平面上；其二是使熔融硅与单晶硅保持良好的散热。

加热系统由电极、热屏蔽层、加热器等组成，其作用是给炉内供热。

炉内气氛系统由抽气泵、氩气入口、及氩气组成。

观察窗用于操作者随时观察炉内情况。

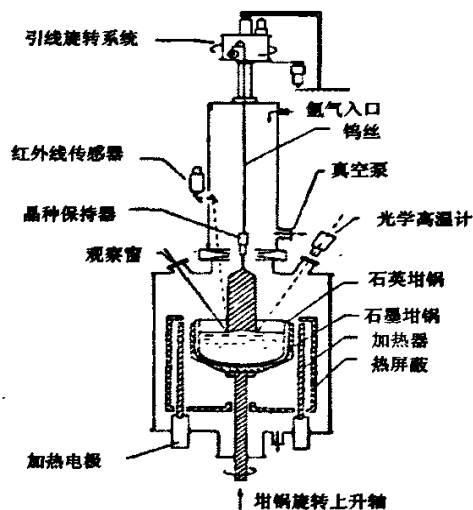


图 1.1 晶体生长炉示意图

单晶硅的提拉是在单晶炉内氩气环境下进行的，提拉过程是把原料多晶硅放入石英坩锅中，在单晶炉中加热融化，再将一根直径只有 10mm 的棒状晶种（称籽晶）浸入熔液中。在合适的温度下，熔液中的硅原子会顺着晶种硅原子的排列结构在固—液交界面上形成规则的结晶，形成单晶体。把晶种微微的旋转向上提升，熔液中的硅原子会在前面形成的单晶体上继续结晶，并延续其规则的原子排列结构。若整个结晶环境稳定，

就可以周而复始的形成结晶，最后形成一根圆柱形的原子排列整齐的单晶硅晶体，即单晶硅晶锭。当结晶加快时，晶体直径会变粗，提高拉速可以使直径变细，升高温度能抑制结晶速度。反之，若结晶变慢，直径变细，则通过降低拉速和降低温度去实现增径控制。拉晶开始，需要引出一定长度的直径为 3~5mm 的细颈，以消除籽晶中的位错，这个过程叫做缩颈。然后放大晶体直径至工艺要求，进入等径阶段。在长晶接近于尾声时提高加热器的输出功率及提拉速度，逐渐收小晶棒直径，最后生成圆锥形底部，完成提拉^[1]。

1.3 本课题研究的内容

在单晶硅提拉过程中，单晶硅的重量是连续增长的，且由于单晶硅对直径精度的要求非常高，所以要求控制系统的动态响应速度非常快，系统的超调量非常小，抗干扰性能很强。当用普通控制方式时因为其系数之间的配合不能确保是最佳匹配，所以经常出现系统快速性、超调量和抗干扰性不能同时实现最佳的现象。这种由系统控制方式引起系统性能的问题反映到单晶硅的提拉过程，就会影响到单晶硅的质量。

针对以上问题，本课题旨在将一种先进控制策略—PID 神经网络（Proportion-Integral-Derivative Neural Network, PIDNN）应用于单晶硅提拉系统中，为解决单晶硅提拉中存在的品质不稳定问题寻求一种新的解决途径。

由于本课题涉及到材料学、电力电子技术、控制理论和控制技术等多方面知识的综合，考虑到时间有限，本论文主要进行了以下几项工作：

- (1) 研究晶体生长、晶体缺陷等材料理论，建立晶体生长数学模型；
- (2) 研究 PID 神经网络理论，并将其予以应用到提拉系统中；
- (3) 通过仿真证明将单输出的 PIDNN（Single-output PIDNN, SPIDNN）应用于单晶硅提拉速度控制具有可行性；
- (4) 设计并实现提拉系统的硬件电路和部分软件程序。

1.4 本章小结

本章主要论述了以下三方面的内容

- (1) 阐述单晶硅的发展背景；
- (2) 介绍单晶硅提拉概况：单晶硅提拉系统是一个多系统组成的综合系统，提拉速度控制系统主要完成拉晶过程；
- (3) 确定课题的研究内容。

就可以周而复始的形成结晶，最后形成一根圆柱形的原子排列整齐的单晶硅晶体，即单晶硅晶锭。当结晶加快时，晶体直径会变粗，提高拉速可以使直径变细，升高温度能抑制结晶速度。反之，若结晶变慢，直径变细，则通过降低拉速和降低温度去实现增径控制。拉晶开始，需要引出一定长度的直径为 3~5mm 的细颈，以消除籽晶中的位错，这个过程叫做缩颈。然后放大晶体直径至工艺要求，进入等径阶段。在长晶接近于尾声时提高加热器的输出功率及提拉速度，逐渐收小晶棒直径，最后生成圆锥形底部，完成提拉^[1]。

1.3 本课题研究的内容

在单晶硅提拉过程中，单晶硅的重量是连续增长的，且由于单晶硅对直径精度的要求非常高，所以要求控制系统的动态响应速度非常快，系统的超调量非常小，抗干扰性能很强。当用普通控制方式时因为其系数之间的配合不能确保是最佳匹配，所以经常出现系统快速性、超调量和抗干扰性不能同时实现最佳的现象。这种由系统控制方式引起系统性能的问题反映到单晶硅的提拉过程，就会影响到单晶硅的质量。

针对以上问题，本课题旨在将一种先进控制策略—PID 神经网络（Proportion-Integral-Derivative Neural Network, PIDNN）应用于单晶硅提拉系统中，为解决单晶硅提拉中存在的品质不稳定问题寻求一种新的解决途径。

由于本课题涉及到材料学、电力电子技术、控制理论和控制技术等多方面知识的综合，考虑到时间有限，本论文主要进行了以下几项工作：

- (1) 研究晶体生长、晶体缺陷等材料理论，建立晶体生长数学模型；
- (2) 研究 PID 神经网络理论，并将其予以应用到提拉系统中；
- (3) 通过仿真证明将单输出的 PIDNN（Single-output PIDNN, SPIDNN）应用于单晶硅提拉速度控制具有可行性；
- (4) 设计并实现提拉系统的硬件电路和部分软件程序。

1.4 本章小结

本章主要论述了以下三方面的内容

- (1) 阐述单晶硅的发展背景；
- (2) 介绍单晶硅提拉概况：单晶硅提拉系统是一个多系统组成的综合系统，提拉速度控制系统主要完成拉晶过程；
- (3) 确定课题的研究内容。

2 提拉系统原理与结构

2.1 晶体生长

2.1.1 晶体生长理论

热力学认为：晶体生长是一个动态过程，是从非平衡态向平衡态过渡的过程。当体系达到两相热力学平衡时，并不能生成新相，只有在旧相处于过饱和（过冷）状态时，才会出现新相。图 2.1 是一个单组分体系 P-T 图，图中分气（V）、液（L）、固（S）三个相区，在任何两相相区之间均有相应的亚稳区（阴影区）。当热力学条件处于亚稳区时才有新相生成，并不断使界面向旧相推移，以完成成核与晶体长大的过程。

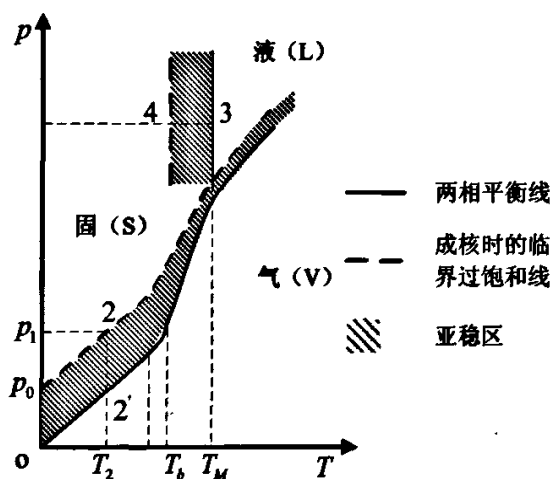


图 2.1 单组分体系 P-T 相图与晶体生长的临界过饱和关系

晶体生长方式总的可以分为三大类：

- ①固相生长：是通过固-固相转变完成的晶体生长过程；
- ②液相生长：包括溶液中生长和熔体中生长两种；
- ③气相生长：由气相向固体转变的气-固相转变过程^[1]。

(1) 熔体中晶体生长的热量运输

从熔体中生长单晶体，是目前半导体材料制备单晶的主要方法。这种方法要求生长材料在熔点附近性能稳定，不发生分解、升华和相变。

熔体生长是典型的液-固相转变过程。在此相变过程中，生长的原子或分子要完成从无序随机排列转变为有序排列，从无对称转变为有对称的阵列结构，同时伴随相变还要释放相变潜热。这种相变不是整体效应，而是由固-液界面不断推进而逐渐完成的一

2 提拉系统原理与结构

2.1 晶体生长

2.1.1 晶体生长理论

热力学认为：晶体生长是一个动态过程，是从非平衡态向平衡态过渡的过程。当体系达到两相热力学平衡时，并不能生成新相，只有在旧相处于过饱和（过冷）状态时，才会出现新相。图 2.1 是一个单组分体系 P-T 图，图中分气（V）、液（L）、固（S）三个相区，在任何两相相区之间均有相应的亚稳区（阴影区）。当热力学条件处于亚稳区时才有新相生成，并不断使界面向旧相推移，以完成成核与晶体长大的过程。

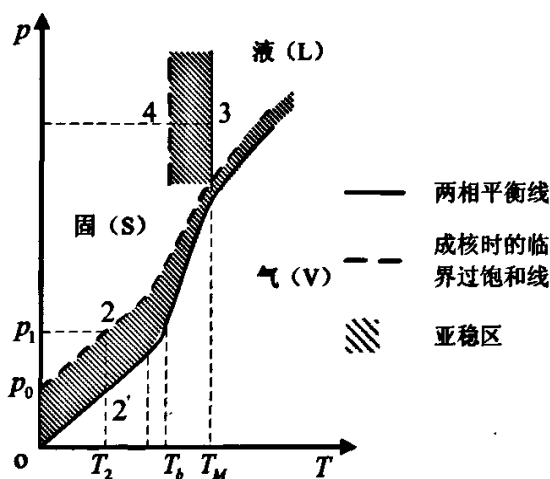


图 2.1 单组分体系 P-T 相图与晶体生长的临界过饱和关系

晶体生长方式总的可以分为三大类：

- ①固相生长：是通过固-固相转变完成的晶体生长过程；
- ②液相生长：包括溶液中生长和熔体中生长两种；
- ③气相生长：由气相向固体转变的气-固相转变过程^[1]。

（1）熔体中晶体生长的热量运输

从熔体中生长单晶体，是目前半导体材料制备单晶的主要方法。这种方法要求生长材料在熔点附近性能稳定，不发生分解、升华和相变。

熔体生长是典型的液-固相转变过程。在此相变过程中，生长的原子或分子要完成从无序随机排列转变为有序排列，从无对称转变为有对称的阵列结构，同时伴随相变还要释放相变潜热。这种相变不是整体效应，而是由固-液界面不断推进而逐渐完成的一

个类似正常凝固的过程。

晶体生长过程中,为保证晶体生长顺序进行,必须考虑解决热量和质量运输的问题。通常在气相和液相中生长晶体时,质量输运起重要作用。在熔体中生长主要是热量的输运问题,它直接影响晶体生长参数、生长界面形态和晶体的完整性等。

图 2.2 为在直拉单晶炉生长晶体的热传递示意图。热传递包括以下几种方式:

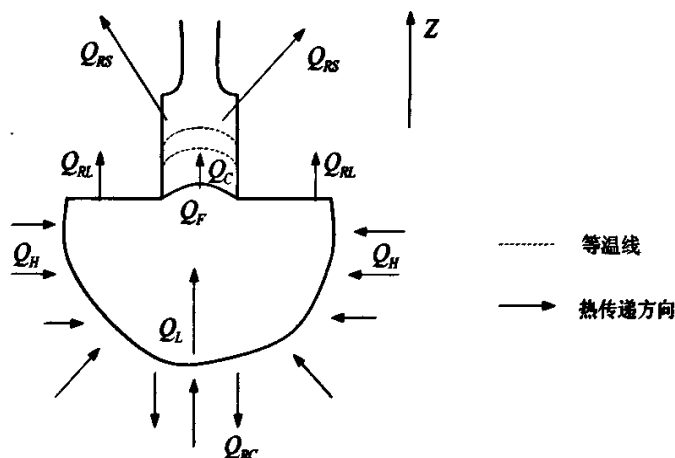


图 2.2 直拉法生长晶体的热传递

Q_H : 加热器传给坩锅的热;

Q_{RC} : 坩锅向外辐射的热 (特别是在底部);

Q_L : 熔体向固-液界面传导的热;

Q_{RL} : 熔体通过液面向外散热;

Q_F : 相变时释放出相变潜热;

Q_{RS} : 晶体的表面散热;

Q_C : 从固-液界面传走的热,它是从晶体表面散发的热 Q_{RS} 和从籽晶传导散失热 Q_C 之和。

在上述八种热传递途径中,比较重要的是直接通过固-液界面的几种热流,即 Q_L 、 Q_F 、 Q_C 三种。在晶体稳定生长时,它的必要条件是在生长的固-液界面上保持热平衡,即在单位时间内

$$Q_L + Q_F = Q_C \quad (2.1)$$

设生长晶体的截面积为 A , 生长速度为 v , 相变潜热为 H (J/g), 晶体密度为 d , 那么结晶放出的相变潜热 Q_F 为

$$Q_F = vAdH \quad (2.2)$$

从熔体传到界面的热量 Q_L 为

$$Q_L = K_L \left(\frac{dT}{dZ} \right)_L A \quad (2.3)$$

从固-液界面向晶体传走的热量 Q_C 为

$$Q_C = K_S \left(\frac{dT}{dZ} \right)_S A \quad (2.4)$$

将式 (2.2)、(2.3)、(2.4) 代入式 (2.1), 得

$$K_L \left(\frac{dT}{dZ} \right)_L A + vAdH = K_S \left(\frac{dT}{dZ} \right)_S A \quad (2.5)$$

式中, K_L 、 K_S 分别为熔体和晶体的热传导系数, $\left(\frac{dT}{dZ} \right)_L$ 和 $\left(\frac{dT}{dZ} \right)_S$ 分别为融体和晶体在固-液界面处的温度梯度^[2]。

(2) 晶体生长速度的估算

式 (2.5) 可变换成生长速度 v 的函数关系式

$$v = \frac{K_S \left(\frac{dT}{dZ} \right)_S - K_L \left(\frac{dT}{dZ} \right)_L}{dH} \quad (2.6)$$

当生长体系固定后, 式 (2.6) 中的 K_S 、 K_L 、 d 、 H 可均为定值, 结果上式成为 $v \sim \left(\frac{dT}{dZ} \right)_L$ 、

$\left(\frac{dT}{dZ} \right)_S$ 的关系式。若想使生长速度尽可能大, 则要求 $\left(\frac{dT}{dZ} \right)_S$ 越大越好, $\left(\frac{dT}{dZ} \right)_L$ 越小越好。

但是 $\left(\frac{dT}{dZ} \right)_S$ 过大可能使晶体残留较大的热应力, 同时也不利于晶体在降温过程中晶格整理,

减少了位错滑移攀移的机会, 从而影响晶体的完整性; 另外, $\left(\frac{dT}{dZ} \right)_L$ 也不可能太小,

因为那样会使固-液界面不平坦, 晶体缺陷密度变大, 甚至会造成枝蔓状生长, 严重时

可转为多晶。可见二者都有一定限度。尽管如此, 还是可以用式 (2.6) 估算一个体系的最大生长速度, 这是个极限值, 即在理论上的 $\left(\frac{dT}{dZ} \right)_L = 0$ 时, 可得

$$v_{\max} = \frac{K_S}{dH} \left(\frac{dT}{dZ} \right)_S \quad (2.7)$$

如测定出 $\left(\frac{dT}{dZ} \right)_S$, 就可以进一步算出 v_{\max} 值。按照式 (2.7) 鲁恩扬 (Runyan) 于 1965

年对一个单晶硅生长系统进行了估算, $v_{\max} = 2.96 \text{ cm/min}$ 。实际测得的 $v_{\max} = 2.53 \text{ cm/min}$,

可见理论值与实验值大体上是相符的^[2]。

(3) 生长速度与晶体外形的关系

在单晶生长时，一般籽晶的截面积都较小，如果生长了一段时间后，晶体有了一定长度，那么从籽晶传导走的热量 Q_c 可以忽略不计，这时晶体的热耗散仅是晶体表面的辐射耗散，根据斯蒂芬-波尔兹曼定律，通过单位高度表面向外辐射的热量 $Q_c = Q_{rs}$ ，

$$Q_c = 4\pi\epsilon\sigma T^4 D \quad (2.8)$$

式中， ϵ 为热发散率， σ 为斯蒂芬-波尔兹曼常数， T 为体系（晶体表面）温度， D 为晶体直径，则式 (2.5) 可改写为

$$K_L \left(\frac{dT}{dZ} \right)_L \pi \frac{D^2}{4} + v H d \pi \frac{D^2}{4} = 4\pi\epsilon\sigma T^4 D \quad (2.9)$$

$$v = \frac{16\epsilon\sigma T^4}{Hd} \frac{1}{D} - \frac{K_L \left(\frac{dT}{dZ} \right)_L}{Hd} \quad (2.10)$$

设 $\frac{16\epsilon\sigma T^4}{Hd} = B_1$ ， $\frac{K_L \left(\frac{dT}{dZ} \right)_L}{Hd} = B_2$ 则式 (2.10) 变为

$$v = \frac{B_1}{D} - B_2 \quad (2.11)$$

如潜热项 B_1 比熔体的热传导项 B_2 大，即 $B_1 \gg B_2$ 条件成立，则

$$v \propto \frac{1}{D} \quad (2.12)$$

从式 (2.12) 可知晶体生长速度与生长晶体的直径成反比^[2]。

在实际生产中，常用改变拉速来控制晶体直径。

2.1.2 单晶硅的晶体缺陷

晶体中偏离理想空间点阵的区域或结构，通常称为晶体的缺陷。晶体中的缺陷通常分为点缺陷、线缺陷、面缺陷和体缺陷。

(1) 点缺陷

点缺陷是尺寸最小的一种晶体缺陷，其大小与原子尺寸相当。点缺陷的基本类型是点阵空位、填隙原子和杂质原子。

(2) 线缺陷

线缺陷主要是指位错，通常以位错密度来衡量晶体的完整性。为了获得优质晶体，人们总是想办法把位错减至最少，即生长无位错的单晶体。位错在某些情况下也可能是有益的，这时就要想办法在晶体生长过程中引入一定数量的可控位错。

可见理论值与实验值大体上是相符的^[2]。

(3) 生长速度与晶体外形的关系

在单晶生长时，一般籽晶的截面积都较小，如果生长了一段时间后，晶体有了一定长度，那么从籽晶传导走的热量 Q_c 可以忽略不计，这时晶体的热耗散仅是晶体表面的辐射耗散，根据斯蒂芬-波尔兹曼定律，通过单位高度表面向外辐射的热量 $Q_c = Q_{rs}$ ，

$$Q_c = 4\pi\epsilon\sigma T^4 D \quad (2.8)$$

式中， ϵ 为热发散率， σ 为斯蒂芬-波尔兹曼常数， T 为体系（晶体表面）温度， D 为晶体直径，则式 (2.5) 可改写为

$$K_L \left(\frac{dT}{dZ} \right)_L \pi \frac{D^2}{4} + v H d \pi \frac{D^2}{4} = 4\pi\epsilon\sigma T^4 D \quad (2.9)$$

$$v = \frac{16\epsilon\sigma T^4}{Hd} \frac{1}{D} - \frac{K_L \left(\frac{dT}{dZ} \right)_L}{Hd} \quad (2.10)$$

设 $\frac{16\epsilon\sigma T^4}{Hd} = B_1$ ， $\frac{K_L \left(\frac{dT}{dZ} \right)_L}{Hd} = B_2$ 则式 (2.10) 变为

$$v = \frac{B_1}{D} - B_2 \quad (2.11)$$

如潜热项 B_1 比熔体的热传导项 B_2 大，即 $B_1 \gg B_2$ 条件成立，则

$$v \propto \frac{1}{D} \quad (2.12)$$

从式 (2.12) 可知晶体生长速度与生长晶体的直径成反比^[2]。

在实际生产中，常用改变拉速来控制晶体直径。

2.1.2 单晶硅的晶体缺陷

晶体中偏离理想空间点阵的区域或结构，通常称为晶体的缺陷。晶体中的缺陷通常分为点缺陷、线缺陷、面缺陷和体缺陷。

(1) 点缺陷

点缺陷是尺寸最小的一种晶体缺陷，其大小与原子尺寸相当。点缺陷的基本类型是点阵空位、填隙原子和杂质原子。

(2) 线缺陷

线缺陷主要是指位错，通常以位错密度来衡量晶体的完整性。为了获得优质晶体，人们总是想办法把位错减至最少，即生长无位错的单晶体。位错在某些情况下也可能是有益的，这时就要想办法在晶体生长过程中引入一定数量的可控位错。

晶体生长过程中，位错形成的主要因素有如下几个方面：

①籽晶中原有位错向晶体延伸，如“遗传位错”。如果籽晶中位错的方向指向生长界面，那它就将延伸到新生长的晶体中。

②由热应力引起的切应变也是引起位错的一个重要原因。

③晶体中的杂质或某种组分出现不均与偏析时，局部的点阵常数与基质的点阵常数不同，为了缓和有这种差异的积累所形成的应力，晶体中可能诱发出新的位错。

④生长界面不稳定会导致枝蔓生长、组分过冷和溶质包裹体，产生应变位错。

⑤晶体中的空位，在位错应力场作用下会向位错线靠拢并沉积于位错线上，这可能导致位错的运动，攀移和扩展位错增加。这样产生的位错，其密度随空位的增加而增加，同时也将随晶体直径和降温速率的增加而增加。

上述几种位错形成的机制，有时可能对晶体中位错的形成同时起作用，一个晶体的所有位错，很难说是由某种单一的原因形成的。选择适当的籽晶和生长方向，或采用“缩颈”技术，凸界面生长等方法，可以从有错位的籽晶中生长出无位错的晶体来。

(3) 面缺陷

面缺陷为一个或几个原子间距，但在广度上具备宏观尺寸。

(4) 体缺陷

体缺陷即三维缺陷，包括包裹体和晶体的开裂。

①包裹体

包裹体是晶体中某些与基质中不同的物相所占据的区域。它是晶体生长中常见的缺陷之一，对晶体质量影响也比较大。

②晶体的开裂

在晶体生长过程中或在晶体的退火、加工过程中，开裂现象是经常遇到的。晶体开裂是由于晶体内部应力产生的应变超出了范性形变范围，而且局部超出了屈服点而造成的。晶体往往沿交接面开裂。

生长速率太快以及过大的温度梯度是造成晶体开裂的重要原因。所以提拉过程中必须将拉速控制在一定范围之内。

晶体中缺陷的种类、数量是鉴别晶体质量的重要标志。晶体的缺陷通常能够吸收、反射或散射晶体内部产生的或是由外部输入的磁、光、声和电能量，从而影响晶体的性能。缺陷的性质、分布状态以及缺陷之间的相互作用，造成了晶体宏观性能的差别^[2]。

从本节的论述中我们可以看出，晶体的生长速度既影响到晶体的直径，又影响到晶体的缺陷，所以提拉过程对速度的控制至关重要。

2.1.3 提拉法单晶生长工艺

从以上两节的分析可以看出合理的提拉工艺对抑制晶体的缺陷至关重要。提拉法单

晶生长工艺流程如图 2.3 所示。在工艺流程中，最关键的是“单晶生长”或称拉晶过程，可分为：润晶、缩颈、放肩、等径生长、拉光等步骤。

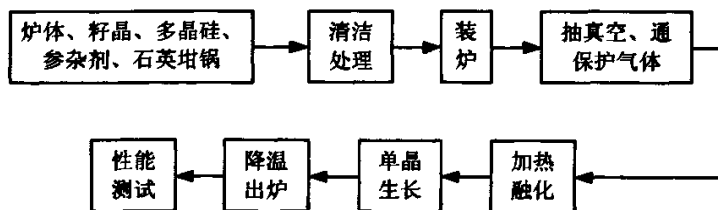


图 2.3 提拉法生长工艺流程

在熔体温度稳定在稍高于熔点时，将籽晶放在上面烘烤几分钟后与熔体熔接，这一步叫润晶或下种；为了消除位错要将籽晶拉细一段叫缩颈；之后要把晶体放粗到要求的直径叫放肩；有了正常的直径后就保持在此直径生长，称之为等径生长；最后将熔体全部拉光，如图 2.4 所示。

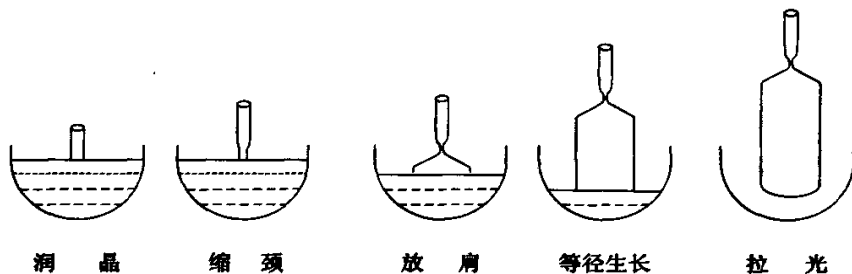


图 2.4 提拉单晶生长过程示意图

在晶体生长过程中，为了保持单晶体等径生长，控制的参数主要是拉速和加热功率。提高拉速、加热功率则晶体直径变小；反之降低拉速和加热功率则使晶体直径变大。

2.2 提拉系统结构

在单晶硅提拉过程中，单晶硅的重量是连续增长的，且由于单晶硅对直径精度的要求非常高，所以要求控制系统的动态响应速度非常快，系统的超调量非常小，抗干扰性能很强。

反馈控制可以实现被控量的无静差；转速、电流双闭环控制能使电动机获得良好的动态性能；小惯量直流伺服电动机最大限度的减小了转动惯量，所以能获得最好的快速性。因此可以确定提拉系统为带有转速、电流双闭环的直流伺服系统，如图 2.5 所示。

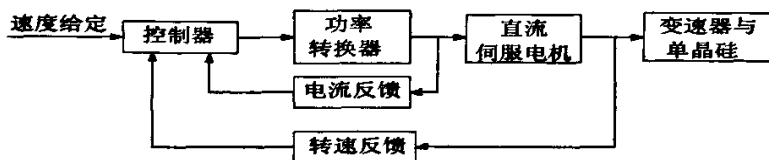


图 2.5 提拉系统组成结构框图

晶生长工艺流程如图 2.3 所示。在工艺流程中，最关键的是“单晶生长”或称拉晶过程，可分为：润晶、缩颈、放肩、等径生长、拉光等步骤。

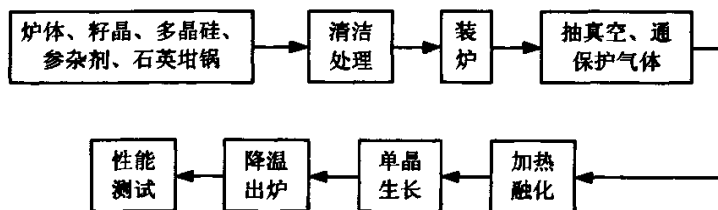


图 2.3 提拉法生长工艺流程

在熔体温度稳定在稍高于熔点时，将籽晶放在上面烘烤几分钟后与熔体熔接，这一步叫润晶或下种；为了消除位错要将籽晶拉细一段叫缩颈；之后要把晶体放粗到要求的直径叫放肩；有了正常的直径后就保持在此直径生长，称之为等径生长；最后将熔体全部拉光，如图 2.4 所示。

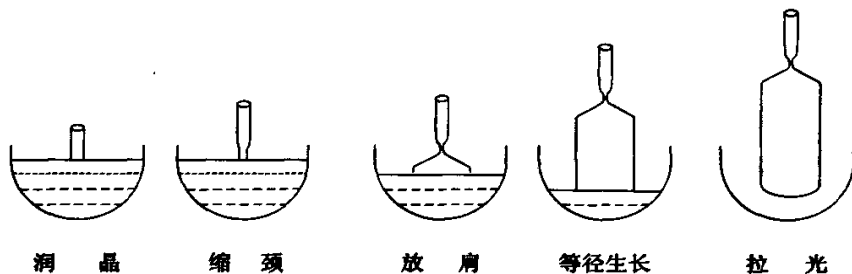


图 2.4 提拉单晶生长过程示意图

在晶体生长过程中，为了保持单晶体等径生长，控制的参数主要是拉速和加热功率。提高拉速、加热功率则晶体直径变小；反之降低拉速和加热功率则使晶体直径变大。

2.2 提拉系统结构

在单晶硅提拉过程中，单晶硅的重量是连续增长的，且由于单晶硅对直径精度的要求非常高，所以要求控制系统的动态响应速度非常快，系统的超调量非常小，抗干扰性能很强。

反馈控制可以实现被控量的无静差；转速、电流双闭环控制能使电动机获得良好的动态性能；小惯量直流伺服电动机最大限度的减小了转动惯量，所以能获得最好的快速性。因此可以确定提拉系统为带有转速、电流双闭环的直流伺服系统，如图 2.5 所示。

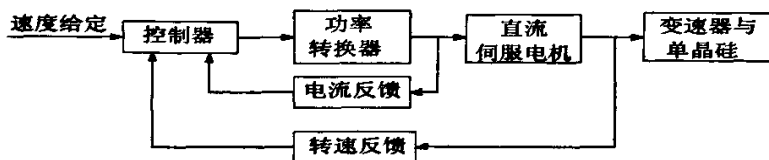


图 2.5 提拉系统组成结构框图

转速调节器和电流调节器的作用可归纳如下:

(1) 转速调节器的作用

①转速调节器是系统的主导调节器,它使转速很快地跟随给定值变化,采用 PI 调节器时可实现转速无静差;

②对负载变化起抗扰作用;

(2) 电流调节器的作用

①作为内环的调节器,在转速外环的调节过程中,它的作用是使电流紧紧跟随其给定值(即外环调节器的输出)变化;

②对电网的波动起及时抗扰的作用;

③在转速动态过程中,保证获得电动机允许的最大电流,从而加快动态过程;

④当电动机过载甚至堵转时,限制电枢电流的最大值,起快速的自动保护作用。一旦故障消失,系统立即自动恢复正常。这个作用对系统的可靠运行来说是十分重要的^[3]。

2.3 PID 控制

(1) PID 控制的作用

自 20 世纪 30 年代末期 PID 控制出现以来,无论是在控制理论,还是在控制仪表、设备方面,都有很大的发展。它是应用最早、最广泛的一种自动控制器,具有原理简单、易于实现、参数整定方便、结构改变灵活、适应性强、鲁棒性强等特点,对于大多数控制对象都能获得满意的控制效果。即使在控制理论取得了丰富成果、出现了许多新型控制规律的今天, PID 控制仍是应用最广泛的控制方法之一。

PID 控制包含有比例(P)、积分(I)和微分(D)三种控制规律。由这三种控制规律可以组合成不同的控制器。在电力传动控制系统中,常采用的串联校正控制装置有比例-微分(PD)控制器、比例-积分(PI)控制器及比例-积分-微分(PID)控制器。它们都是线性控制器,其作用是按偏差的比例加微分,或比例加积分,或比例加积分加微分形成控制量。由 PD 控制器构成的超前校正,可以提高稳定裕度并获得足够的快速性,但稳态精度可能受到影响;由 PI 控制器构成的滞后校正,可以保证稳态精度,却是对快速性的限制来换取系统的稳定的;用 PID 控制器实现的滞后-超前校正则兼有二者的优点,可以全面提高系统的控制性能。PID 控制器的控制规律为

$$u(t) = K_p [e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt}] \quad (2.13)$$

式中, K_p 为比例系数、 T_i 为积分时间常数、 T_d 为微分时间常数。

PID 控制器对偏差的阶跃响应如图 2.6 所示。它在偏差 $e(t)$ 阶跃变化的瞬间($t=t_0$) 有一冲击式瞬间响应,这是微分作用引起的。由图 2.6 可知,对于 PID 控制器,在阶跃信号作用下,首先是比例、微分部分起作用,使其控制作用加强,然后再进行积分直到

转速调节器和电流调节器的作用可归纳如下:

(1) 转速调节器的作用

①转速调节器是系统的主导调节器,它使转速很快地跟随给定值变化,采用 PI 调节器时可实现转速无静差;

②对负载变化起抗扰作用;

(2) 电流调节器的作用

①作为内环的调节器,在转速外环的调节过程中,它的作用是使电流紧紧跟随其给定值(即外环调节器的输出)变化;

②对电网的波动起及时抗扰的作用;

③在转速动态过程中,保证获得电动机允许的最大电流,从而加快动态过程;

④当电动机过载甚至堵转时,限制电枢电流的最大值,起快速的自动保护作用。一旦故障消失,系统立即自动恢复正常。这个作用对系统的可靠运行来说是十分重要的^[3]。

2.3 PID 控制

(1) PID 控制的作用

自 20 世纪 30 年代末期 PID 控制出现以来,无论是在控制理论,还是在控制仪表、设备方面,都有很大的发展。它是应用最早、最广泛的一种自动控制器,具有原理简单、易于实现、参数整定方便、结构改变灵活、适应性强、鲁棒性强等特点,对于大多数控制对象都能获得满意的控制效果。即使在控制理论取得了丰富成果、出现了许多新型控制规律的今天, PID 控制仍是应用最广泛的控制方法之一。

PID 控制包含有比例(P)、积分(I)和微分(D)三种控制规律。由这三种控制规律可以组合成不同的控制器。在电力传动控制系统中,常采用的串联校正控制装置有比例-微分(PD)控制器、比例-积分(PI)控制器及比例-积分-微分(PID)控制器。它们都是线性控制器,其作用是按偏差的比例加微分,或比例加积分,或比例加积分加微分形成控制量。由 PD 控制器构成的超前校正,可以提高稳定裕度并获得足够的快速性,但稳态精度可能受到影响;由 PI 控制器构成的滞后校正,可以保证稳态精度,却是对快速性的限制来换取系统的稳定的;用 PID 控制器实现的滞后-超前校正则兼有二者的优点,可以全面提高系统的控制性能。PID 控制器的控制规律为

$$u(t) = K_p [e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt}] \quad (2.13)$$

式中, K_p 为比例系数、 T_i 为积分时间常数、 T_d 为微分时间常数。

PID 控制器对偏差的阶跃响应如图 2.6 所示。它在偏差 $e(t)$ 阶跃变化的瞬间($t=t_0$) 有一冲击式瞬间响应,这是微分作用引起的。由图 2.6 可知,对于 PID 控制器,在阶跃信号作用下,首先是比例、微分部分起作用,使其控制作用加强,然后再进行积分直到

最后消除静差为止。因此,采用PID控制器,无论从静差、还是从动态的角度来说,控制品质都得到了改善。

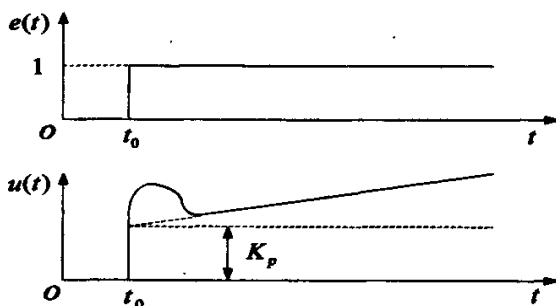


图 2.6 PID 控制器阶跃响应

(2) 数字 PID 控制算法

在计算机进入到控制领域后,PID 控制规律的应用不但没有受到影响,而且有了新的发展,它是当今工业过程计算机控制系统中应用最广泛的一种。由于计算机控制是一种采样控制,它只能根据采样得到的偏差值来计算控制量,而采样是一拍一拍的进行的,所以在计算机控制系统中,不能直接采用式(2.13)的PID模拟运算形式,必须进行离散化处理,求得数字式的PID控制算法。

① 位置式 PID 控制算法

令 T 为采样周期,我们以一系列采样时刻点 kT 代替连续时间 t ,以和式代替积分,以增量代替微分,对式(2.13)作如下近似变换

$$\int_0^t e(t)dt \approx T \sum_{j=0}^k e(jT) \quad (2.14)$$

$$\frac{de(t)}{dt} \approx \frac{e(kT) - e[(k-1)T]}{T} \quad (2.15)$$

$$t = kT \quad k = 0, 1, 2, \dots$$

$$\begin{aligned} u(k) &= K_p \left\{ e(k) + \frac{T}{T_i} \sum_{j=0}^k e(j) + \frac{T_d}{T} [e(k) - e(k-1)] \right\} \\ &= K_p e(k) + K_i \sum_{j=0}^k e(j) + K_d [e(k) - e(k-1)] \end{aligned} \quad (2.16)$$

在式(2.16)中,为了书写方便,将 $e(kT)$ 简化为 $e(k)$,即省去了 T 。

式中, k 为采样序号, $k = 0, 1, 2, \dots$, $e(k)$ 为第 k 次采样周期测得的偏差值; $e(k-1)$ 为第 $k-1$ 次采样测得的偏差值, $u(k)$ 为 PID 控制器第 k 次输出值; K_p 为比例系数;

$K_i = \frac{K_p T}{T_i}$ 为积分系数; $K_d = \frac{K_p T_d}{T}$ 为微分系数。

由于控制器的输出 $u(k)$ 直接去控制执行机构(如阀门), $u(k)$ 的值和执行机构的位置

(如阀门开度)是一一对应的,所以,我们通常称式(2.16)为位置式PID控制算法。

由于位置式PID控制计算机输出的 $u(k)$ 与执行机构的实际位置一一对应,一旦计算机出现故障, $u(k)$ 的大幅度变化会引起执行机构位置的大幅度变化,这种情况是生产中所不允许出现的,在某些场合,还可能造成重大的生产事故。此外,位置式PID的输出不仅与本次偏差有关,而且与历次测量偏差值有关,计算时要对 $e(k)$ 累加,计算机运算工作量大。为了克服上述位置式PID的不足,各种改进的PID控制算法应用而出^[4]。

②增量式PID控制算法

增量式PID控制算法是改进PID算法中较为常用的一种。由式(2.16)可得 $k-1$ 时刻的控制量 $u(k-1)$ 为

$$u(k-1) = K_p e(k-1) + K_i \sum_{j=1}^{k-1} e(j) + K_d [e(k-1) - e(k-2)] \quad (2.17)$$

则

$$\begin{aligned} \Delta u(k) &= u(k) - u(k-1) \\ &= K_p [e(k) - e(k-1)] + K_i e(k) + K_d [e(k) - 2e(k-1) + e(k-2)] \end{aligned} \quad (2.18)$$

由于式(2.18)中 $\Delta u(k)$ 为第 k 次相对于第 $k-1$ 次的控制量的增量,所以称该PID算法为增量式算法^[4]。

可以看出增量式PID控制器的输出仅仅与最近三次的偏差值有关。这样,在确定了 K_p 、 K_i 、 K_d 之后,根据最近三次的偏差值即可求出控制增量。

有时为了方便,也可将式(2.16)改写为

$$\Delta u(k) = A e(k) - B e(k-1) + C e(k-2) \quad (2.19)$$

式中

$$A = K_p + K_i + K_d = K_p \left(1 + \frac{T}{T_i} + \frac{T_d}{T}\right)$$

$$B = K_p + 2K_d = K_p \left(1 + 2\frac{T_d}{T}\right)$$

$$C = K_d = \frac{K_p T_d}{T}$$

$\Delta u(k)$ 对应的是本次执行机构位置(如阀门开度)的增量,对应阀门实际位置的控制量是通过累积历次控制增量形成的,常用的累积元件有步进电动机等。

当执行机构前没有安排硬件累积元件时,我们也可采用增量式算法,只是此时控制增量的累积改用软件完成。算法如下

$$u(k) = u(k-1) + \Delta u(k) \quad (2.20)$$

(3) PID控制算法子程序

根据式 2.19 和 2.20 可得系统电流调节器和速度调节器 PID 算法程序流程图如图 2.7 所示。

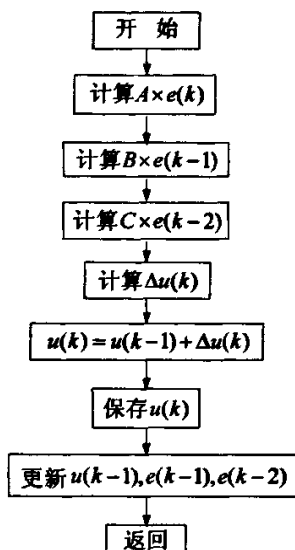


图 2.7 PID 控制算法子程序框图

2.4 本章小结

本章主要论述了以下三方面的问题

- (1) 分析了晶体材料的热量运输、晶体缺陷与提拉速度的关系;
- (2) 确定提拉速度系统的结构;
- (3) 阐述 PID 控制的原理。

根据式 2.19 和 2.20 可得系统电流调节器和速度调节器 PID 算法程序流程图如图 2.7 所示。

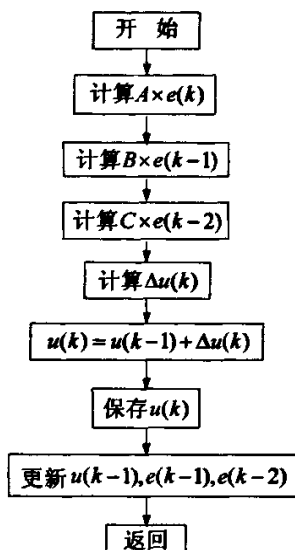


图 2.7 PID 控制算法子程序框图

2.4 本章小结

本章主要论述了以下三方面的问题

- (1) 分析了晶体材料的热量运输、晶体缺陷与提拉速度的关系;
- (2) 确定提拉速度系统的结构;
- (3) 阐述 PID 控制的原理。

3 PID 神经网络

3.1 传统控制理论

自 20 世纪 40 年代以来,传统控制理论得到了快速的发展,形成了完整的理论体系和控制系统的大规模产业化。20 世纪 70 年代以来的近 30 年中,为了解决航天、军事、工业、社会系统等复杂的控制任务,控制理论以科学史上前所未有的速度经历了现代控制理论和大系统理论两个重要的发展阶段,但是,它对精确数学模型的依赖性,使其应用受到很大的限制。无论是现代控制理论还是大系统理论,其分析、综合、设计都是建立在严格和精确的数学模型基础之上的。而在科学技术和生产力水平高速发展的今天,人们对大规模、复杂和不确定性系统实行自动控制的要求不断提高。因此传统的基于精确数学模型的控制理论的局限性日益明显。

传统控制理论在具体的控制系统中的局限性表现在以下几个方面:

(1) 不适应不确定性系统的控制。传统控制是基于模型的控制,即认为控制、对象及干扰的模型是已知的或是经过辨识可以得到的。但是由于被控系统的结构和参数往往难以测量或辨识,很多被控制系统的结构和参数还具有不确定性,甚至常常会发生突变。对于这些未知、不确定或知之甚少的被控系统,难以建立数学模型,因而采用传统控制理论无法实现有效控制。

(2) 不适应非线性系统的控制。一般被控系统都具有非线性,当非线性特性的影响较小时,传统控制理论通常将其近似线性化后设计控制器。当被控系统具有高度非线性特性时,在传统控制理论中虽然也有一些方法可资利用,但只是针对一些具体问题的,有较多的附加条件,大多数过于复杂而难以实际运用。

(3) 不适应时变系统的控制。实际被控系统的结构和参数随时间而发生变化,绝对对不变的系统是不存在的。当这种变化较小时,经过一系列的近似后,才能利用传统控制理论进行系统综合。如果时变因素较大,传统控制理论则无法应用。

(4) 不适应多变量系统的控制。多变量系统的控制问题一直是控制理论界和控制工程界研究的重点和难点问题,多变量系统除了与单变量系统一样存在不确定性、非线性和时变问题以外,还存在着各要素间相互耦合、互相制约等特殊的问题。如果多变量系统为线性时不变而且结构和参数已知,还可以应用传统控制理论设计解耦器和控制器,对其进行控制。如果以上条件不成立,传统控制理论则无法应用。而在实际中,这些条件一般很难满足。

传统的控制理论虽然也有办法对付控制对象的不确定性和复杂性,如自适应控制和鲁棒(Robust)控制也可以克服系统中所包含的不确定性,达到优化控制的目的。但是

3 PID 神经网络

3.1 传统控制理论

自 20 世纪 40 年代以来,传统控制理论得到了快速的发展,形成了完整的理论体系和控制系统的大规模产业化。20 世纪 70 年代以来的近 30 年中,为了解决航天、军事、工业、社会系统等复杂的控制任务,控制理论以科学史上前所未有的速度经历了现代控制理论和大系统理论两个重要的发展阶段,但是,它对精确数学模型的依赖性,使其应用受到很大的限制。无论是现代控制理论还是大系统理论,其分析、综合、设计都是建立在严格和精确的数学模型基础之上的。而在科学技术和生产力水平高速发展的今天,人们对大规模、复杂和不确定性系统实行自动控制的要求不断提高。因此传统的基于精确数学模型的控制理论的局限性日益明显。

传统控制理论在具体的控制系统中的局限性表现在以下几个方面:

(1) 不适应不确定性系统的控制。传统控制是基于模型的控制,即认为控制、对象及干扰的模型是已知的或是经过辨识可以得到的。但是由于被控系统的结构和参数往往难以测量或辨识,很多被控制系统的结构和参数还具有不确定性,甚至常常会发生突变。对于这些未知、不确定或知之甚少的被控系统,难以建立数学模型,因而采用传统控制理论无法实现有效控制。

(2) 不适应非线性系统的控制。一般被控系统都具有非线性,当非线性特性的影响较小时,传统控制理论通常将其近似线性化后设计控制器。当被控系统具有高度非线性特性时,在传统控制理论中虽然也有一些方法可资利用,但只是针对一些具体问题的,有较多的附加条件,大多数过于复杂而难以实际运用。

(3) 不适应时变系统的控制。实际被控系统的结构和参数随时间而发生变化,绝对对不变的系统是不存在的。当这种变化较小时,经过一系列的近似后,才能利用传统控制理论进行系统综合。如果时变因素较大,传统控制理论则无法应用。

(4) 不适应多变量系统的控制。多变量系统的控制问题一直是控制理论界和控制工程界研究的重点和难点问题,多变量系统除了与单变量系统一样存在不确定性、非线性和时变问题以外,还存在着各要素间相互耦合、互相制约等特殊的问题。如果多变量系统为线性时不变而且结构和参数已知,还可以应用传统控制理论设计解耦器和控制器,对其进行控制。如果以上条件不成立,传统控制理论则无法应用。而在实际中,这些条件一般很难满足。

传统的控制理论虽然也有办法对付控制对象的不确定性和复杂性,如自适应控制和鲁棒(Robust)控制也可以克服系统中所包含的不确定性,达到优化控制的目的。但是

自适应控制是以自动调节控制器的参数,使控制器与被控对象和环境达到良好的“匹配”,以削弱不确定性的影响为目标的。从本质上说,自适应和自校正控制都是通过对系统某些重要参数的估计,以补偿的方法来克服干扰和不确定性的。它较适合于参数在一定范围内变化的情况。鲁棒控制则是在外部干扰和内部参数变化作用下,以提高系统不灵敏度为宗旨来抵御不确定性的。根据这一思想和原则所导出的算法,其鲁棒区域是很有限的。

3.2 人工神经网络

3.2.1 人工神经网络概述

智能控制是人工智能、自动控制理论、计算机技术、运筹学等领域的许多知识交叉结合而成的,包括模糊控制、专家系统控制、神经网络控制等方法。尽管智能控制尚需进一步完善和发展,但在复杂系统的控制中已经显示出其在很多方面较之传统控制更好的性能。智能控制的概念和原理主要是针对控制对象、环境、控制目标或任务的复杂性提出来的。而计算机科学、人工智能、信息科学、思维科学、认识科学和人工神经网络的连接机制等方面的新进展和智能机器人的工程实践,从不同角度为智能控制的诞生奠定了必要的理论基础和技术基础。与传统的控制理论相比,智能控制对环境和任务的复杂性有更大的适配程度。它不仅是对建立的模型,而且对环境和任务能抽取多级的描述精度,进而发展了自学习、自适应和自组织等概念,所以能在更广泛的领域中获得应用。

神经网络是智能控制的一个重要分支,神经网络是以大脑生理研究成果为基础,模拟大脑的某些机理与机制,由人工建立的以有向图为拓扑结构的网络,通过对连续或断续的输入做状态响应而进行信息处理;神经网络是本质性的并行结构,并且可以用硬件实现,它在处理对实时性要求很高的自动控制问题显示出很大的优越性;神经网络是本质性的非线性系统,多层神经网络具有逼近任意函数的能力,它能同时处理大量不同类型的输入信息,能很好地解决输入信息之间的冗余问题,能恰当的协调相互矛盾的输入信息,可以处理那些难以用模型或规则描述的信息。神经网络在复杂系统的控制方面有明显的优势,神经网络控制和辨识的研究已经成为智能控制研究的主流。

基于神经网络的智能控制系统也称作基于连接机制的智能控制系统。随着人工神经网络研究的进展,神经网络越来越多的应用于控制领域的各个方面。但不能不看到一般神经网络弱点制约了其在控制系统中的广泛应用,包括以下问题:

(1) 一般神经网络收敛速度慢,训练和学习时间长,这是大多数控制系统所不能接受的。

自适应控制是以自动调节控制器的参数,使控制器与被控对象和环境达到良好的“匹配”,以削弱不确定性的影响为目标的。从本质上说,自适应和自校正控制都是通过对系统某些重要参数的估计,以补偿的方法来克服干扰和不确定性的。它较适合于参数在一定范围内变化的情况。鲁棒控制则是在外部干扰和内部参数变化作用下,以提高系统不灵敏度为宗旨来抵御不确定性的。根据这一思想和原则所导出的算法,其鲁棒区域是很有限的。

3.2 人工神经网络

3.2.1 人工神经网络概述

智能控制是人工智能、自动控制理论、计算机技术、运筹学等领域的许多知识交叉结合而成的,包括模糊控制、专家系统控制、神经网络控制等方法。尽管智能控制尚需进一步完善和发展,但在复杂系统的控制中已经显示出其在很多方面较之传统控制更好的性能。智能控制的概念和原理主要是针对控制对象、环境、控制目标或任务的复杂性提出来的。而计算机科学、人工智能、信息科学、思维科学、认识科学和人工神经网络的连接机制等方面的新进展和智能机器人的工程实践,从不同角度为智能控制的诞生奠定了必要的理论基础和技术基础。与传统的控制理论相比,智能控制对环境和任务的复杂性有更大的适配程度。它不仅是对建立的模型,而且对环境和任务能抽取多级的描述精度,进而发展了自学习、自适应和自组织等概念,所以能在更广泛的领域中获得应用。

神经网络是智能控制的一个重要分支,神经网络是以大脑生理研究成果为基础,模拟大脑的某些机理与机制,由人工建立的以有向图为拓扑结构的网络,通过对连续或断续的输入做状态响应而进行信息处理;神经网络是本质性的并行结构,并且可以用硬件实现,它在处理对实时性要求很高的自动控制问题显示出很大的优越性;神经网络是本质性的非线性系统,多层神经网络具有逼近任意函数的能力,它能同时处理大量不同类型的输入信息,能很好地解决输入信息之间的冗余问题,能恰当的协调相互矛盾的输入信息,可以处理那些难以用模型或规则描述的信息。神经网络在复杂系统的控制方面有明显的优势,神经网络控制和辨识的研究已经成为智能控制研究的主流。

基于神经网络的智能控制系统也称作基于连接机制的智能控制系统。随着人工神经网络研究的进展,神经网络越来越多的应用于控制领域的各个方面。但不能不看到一般神经网络弱点制约了其在控制系统中的广泛应用,包括以下问题:

(1) 一般神经网络收敛速度慢,训练和学习时间长,这是大多数控制系统所不能接受的。

自适应控制是以自动调节控制器的参数,使控制器与被控对象和环境达到良好的“匹配”,以削弱不确定性的影响为目标的。从本质上说,自适应和自校正控制都是通过对系统某些重要参数的估计,以补偿的方法来克服干扰和不确定性的。它较适合于参数在一定范围内变化的情况。鲁棒控制则是在外部干扰和内部参数变化作用下,以提高系统不灵敏度为宗旨来抵御不确定性的。根据这一思想和原则所导出的算法,其鲁棒区域是很有限的。

3.2 人工神经网络

3.2.1 人工神经网络概述

智能控制是人工智能、自动控制理论、计算机技术、运筹学等领域的许多知识交叉结合而成的,包括模糊控制、专家系统控制、神经网络控制等方法。尽管智能控制尚需进一步完善和发展,但在复杂系统的控制中已经显示出其在很多方面较之传统控制更好的性能。智能控制的概念和原理主要是针对控制对象、环境、控制目标或任务的复杂性提出来的。而计算机科学、人工智能、信息科学、思维科学、认识科学和人工神经网络的连接机制等方面的新进展和智能机器人的工程实践,从不同角度为智能控制的诞生奠定了必要的理论基础和技术基础。与传统的控制理论相比,智能控制对环境和任务的复杂性有更大的适配程度。它不仅是对建立的模型,而且对环境和任务能抽取多级的描述精度,进而发展了自学习、自适应和自组织等概念,所以能在更广泛的领域中获得应用。

神经网络是智能控制的一个重要分支,神经网络是以大脑生理研究成果为基础,模拟大脑的某些机理与机制,由人工建立的以有向图为拓扑结构的网络,通过对连续或断续的输入做状态响应而进行信息处理;神经网络是本质性的并行结构,并且可以用硬件实现,它在处理对实时性要求很高的自动控制问题显示出很大的优越性;神经网络是本质性的非线性系统,多层神经网络具有逼近任意函数的能力,它能同时处理大量不同类型的输入信息,能很好地解决输入信息之间的冗余问题,能恰当的协调相互矛盾的输入信息,可以处理那些难以用模型或规则描述的信息。神经网络在复杂系统的控制方面有明显的优势,神经网络控制和辨识的研究已经成为智能控制研究的主流。

基于神经网络的智能控制系统也称作基于连接机制的智能控制系统。随着人工神经网络研究的进展,神经网络越来越多的应用于控制领域的各个方面。但不能不看到一般神经网络弱点制约了其在控制系统中的广泛应用,包括以下问题:

(1) 一般神经网络收敛速度慢,训练和学习时间长,这是大多数控制系统所不能接受的。

(2) 在构成控制器时, 一般神经网络的结构选取, 特别是隐含层单元个数的选取尚无定则, 还需要通过反复试验才能确定, 这给实际应用带来困难。

(3) 一般神经网络连接权重初值多被取为随机数, 存在陷入局部极小值的可能, 使控制性能难以达到逾期的效果; 特别是由于连接权重值的随机性, 很难保证控制系统初始运行的稳定性, 而如果控制系统初始运行不稳定, 便失去了应用的基础。

(4) 传统神经网络的结构、参数和机能, 难以与控制系统所要求的响应快、超调小、无静差等动态和静态性能指标相联系。

(5) 传统神经网络在构成控制器时, 为了满足系统性能的要求, 大量增加神经元的个数, 网络的计算量很大, 使在当前的技术水平下很难保证控制的实时性。

(6) 具有任意函数逼近能力的多层前向神经网络是应用最多的一种神经网络, 但传统的多层前向神经网络的神经元仅具有静态输入—输出特性, 在它构成控制系统时必须附加其他动态部件^[9]。

3.2.2 人工神经网络与 PID 的结合

近年来, 随着神经网络的研究和应用, 人们开始采用神经网络和 PID 控制相结合的方式改变传统 PID 控制的性能。目前已经提出的神经网络和 PID 控制相结合的方法可以归结为三种类型:

①采用神经网络确定 PID 参数;

②单神经元结构 PID 控制器;

③PIDNN。

前两种结合方式的共同之处是它们都是利用神经网络来整定 PID 的参数, 且都不适用于多输入—多输出的多变量控制系统。第三种结合方式, PIDNN 也是神经网络与 PID 控制规律的结合, 不过与前两种结合方式不同的是其分别定义了具有比例、积分、微分功能的神经元, 从而将 PID 控制规律融合进神经网络之中。

3.3 PID 神经网络

3.3.1 PID 神经网络的特点

PIDNN 是神经网络与 PID 控制规律的结合, 其基础是其分别定义了具有比例、积分、微分功能的神经元, 从而将 PID 控制规律融合进神经网络之中。它是一种多层前向神经网络, 它的形式与一般多层前向神经网络类似, 但它与一般多层前向神经网络有所不同, 主要不同之处在隐含层。一般多层前向神经网络中的全部神经元的输入—输出特性都是相同的和静态的, 而 PIDNN 的隐含层是由比例元、积分元和微分元组成的。这些神经元的输入—输出函数分别为比例、积分和微分函数, 既有静态的, 又有动

(2) 在构成控制器时, 一般神经网络的结构选取, 特别是隐含层单元个数的选取尚无定则, 还需要通过反复试验才能确定, 这给实际应用带来困难。

(3) 一般神经网络连接权重初值多被取为随机数, 存在陷入局部极小值的可能, 使控制性能难以达到逾期的效果; 特别是由于连接权重值的随机性, 很难保证控制系统初始运行的稳定性, 而如果控制系统初始运行不稳定, 便失去了应用的基础。

(4) 传统神经网络的结构、参数和机能, 难以与控制系统所要求的响应快、超调小、无静差等动态和静态性能指标相联系。

(5) 传统神经网络在构成控制器时, 为了满足系统性能的要求, 大量增加神经元的个数, 网络的计算量很大, 使在当前的技术水平下很难保证控制的实时性。

(6) 具有任意函数逼近能力的多层前向神经网络是应用最多的一种神经网络, 但传统的多层前向神经网络的神经元仅具有静态输入—输出特性, 在它构成控制系统时必须附加其他动态部件^[9]。

3.2.2 人工神经网络与 PID 的结合

近年来, 随着神经网络的研究和应用, 人们开始采用神经网络和 PID 控制相结合的方式来改变传统 PID 控制的性能。目前已经提出的神经网络和 PID 控制相结合的方法可以归结为三种类型:

①采用神经网络确定 PID 参数;

②单神经元结构 PID 控制器;

③PIDNN。

前两种结合方式的共同之处是它们都是利用神经网络来整定 PID 的参数, 且都不适用于多输入—多输出的多变量控制系统。第三种结合方式, PIDNN 也是神经网络与 PID 控制规律的结合, 不过与前两种结合方式不同的是其分别定义了具有比例、积分、微分功能的神经元, 从而将 PID 控制规律融合进神经网络之中。

3.3 PID 神经网络

3.3.1 PID 神经网络的特点

PIDNN 是神经网络与 PID 控制规律的结合, 其基础是其分别定义了具有比例、积分、微分功能的神经元, 从而将 PID 控制规律融合进神经网络之中。它是一种多层前向神经网络, 它的形式与一般多层前向神经网络类似, 但它与一般多层前向神经网络有所不同, 主要不同之处在隐含层。一般多层前向神经网络中的全部神经元的输入—输出特性都是相同的和静态的, 而 PIDNN 的隐含层是由比例元、积分元和微分元组成的。这些神经元的输入—输出函数分别为比例、积分和微分函数, 既有静态的, 又有动

(2) 在构成控制器时,一般神经网络的结构选取,特别是隐含层单元个数的选取尚无定则,还需要通过反复试验才能确定,这给实际应用带来困难。

(3) 一般神经网络连接权重初值多被取为随机数,存在陷入局部极小值的可能,使控制性能难以达到逾期的效果;特别是由于连接权重值的随机性,很难保证控制系统初始运行的稳定性,而如果控制系统初始运行不稳定,便失去了应用的基础。

(4) 传统神经网络的结构、参数和机能,难以与控制系统所要求的响应快、超调小、无静差等动态和静态性能指标相联系。

(5) 传统神经网络在构成控制器时,为了满足系统性能的要求,大量增加神经元的个数,网络的计算量很大,使在当前的技术水平下很难保证控制的实时性。

(6) 具有任意函数逼近能力的多层前向神经网络是应用最多的一种神经网络,但传统的多层前向神经网络的神经元仅具有静态输入—输出特性,在它构成控制系统时必须附加其他动态部件^[9]。

3.2.2 人工神经网络与 PID 的结合

近年来,随着神经网络的研究和应用,人们开始采用神经网络和 PID 控制相结合的方式来改变传统 PID 控制的性能。目前已经提出的神经网络和 PID 控制相结合的方法可以归结为三种类型:

①采用神经网络确定 PID 参数;

②单神经元结构 PID 控制器;

③PIDNN。

前两种结合方式的共同之处是它们都是利用神经网络来整定 PID 的参数,且都不适用于多输入—多输出的多变量控制系统。第三种结合方式, PIDNN 也是神经网络与 PID 控制规律的结合,不过与前两种结合方式不同的是其分别定义了具有比例、积分、微分功能的神经元,从而将 PID 控制规律融合进神经网络之中。

3.3 PID 神经网络

3.3.1 PID 神经网络的特点

PIDNN 是神经网络与 PID 控制规律的结合,其基础是其分别定义了具有比例、积分、微分功能的神经元,从而将 PID 控制规律融合进神经网络之中。它是一种多层前向神经网络,它的形式与一般多层前向神经网络类似,但它与一般多层前向神经网络有所不同,主要不同之处在隐含层。一般多层前向神经网络中的全部神经元的输入—输出特性都是相同的和静态的,而 PIDNN 的隐含层是由比例元、积分元和微分元组成的。这些神经元的输入—输出函数分别为比例、积分和微分函数,既有静态的,又有动

态的。PIDNN 的各层神经元个数、连接方式、连接权重初值是按 PID 控制规律的基本原则确定的。PIDNN 采用误差反相传播算法去修改连接权重初值，通过在线训练和学习，使系统目标函数达到最优值。其主要特点如下：

(1) PIDNN 属于多层前向神经网络的范畴，因此它具备多层前向神经网络的任意函数逼近能力和其他优秀品质。

(2) PIDNN 参照 PID 控制规律的要求构成，它的结构比较简单和规范。它的结构由网络的输出量个数决定，而网络输出量个数和被控制的变量个数相等。

(3) PIDNN 的连接权重初值按 PID 控制规律的基本原则确定，在此初值的基础上进而进行网络训练、学习和调整，使网络的收敛速度很快且不易陷入局部极小点。更重要的是，可以利用现有的 PID 控制的大量经验数据确定网络权重的初值，从而使控制系统保持初始稳定，使系统的全局稳定成为可能。

(4) PIDNN 中包含了具有微分特性的处理单元，因此它是一种本质动态网络，容易适应控制系统的要求，在构成系统时不必在附加其他部件，它的输入信号无需经过网络外部的微分和积分处理。

(5) 比例元、积分元和微分元的存在，使 PIDNN 控制系统的响应快、超调小、无静差。

(6) PIDNN 采用无“教师”的自学习方式，根据控制效果进行在线学习和调整，使系统具备较好的性能。

(7) PIDNN 可以适应多种类型的单变量系统和多变量系统，无需测量或辨识控制对象的内部结构和参数^[5]。

3.3.2 PID 神经网络的结构形式

PIDNN 的结构形式随着网络输出量的个数的增减而变化，可分为单输出的 PIDNN(Single-output PIDNN, SPIDNN)和多输出的 PIDNN(Multi-output, PIDNN)。SPIDNN 用于单变量的控制，MPIDNN 用于多变量的控制（鉴于本文使用的缘故，对 MPIDNN 不作介绍）。SPIDNN 的基本形式为 $2 \times 3 \times 1$ 结构，如图 3.1 所示。

SPIDNN 输入层的神经元采用比例神经元，这两个神经元接受外部输入信息：其中一个接受单变量控制系统的目标值或给定值；另一个接受被控变量值。输入层神经元的输出经过连接权重，进入隐含层进行综合和处理，隐含层的三个神经元分别为比例元、积分元和微分元，分别对输入信号进行比例、积分和微分。隐含层的输出再通过连接权重，进入输出层，输出层完成整个网络控制规律的综合和输出。

SPIDNN 的结构简单，实现比较容易。如果采用硬件实现，PID 神经元均可用运算放大器和电阻、电容实现。当采用计算机软件实现时，PID 神经元的计算工作量均不大。下面是 SPIDNN 的比例元、积分元和微分元的定义以及前向算法和误差反相传播的学习

态的。PIDNN 的各层神经元个数、连接方式、连接权重初值是按 PID 控制规律的基本原则确定的。PIDNN 采用误差反相传播算法去修改连接权重初值，通过在线训练和学习，使系统目标函数达到最优值。其主要特点如下：

(1) PIDNN 属于多层前向神经网络的范畴，因此它具备多层前向神经网络的任意函数逼近能力和其他优秀品质。

(2) PIDNN 参照 PID 控制规律的要求构成，它的结构比较简单和规范。它的结构由网络的输出量个数决定，而网络输出量个数和被控制的变量个数相等。

(3) PIDNN 的连接权重初值按 PID 控制规律的基本原则确定，在此初值的基础上进而进行网络训练、学习和调整，使网络的收敛速度很快且不易陷入局部极小点。更重要的是，可以利用现有的 PID 控制的大量经验数据确定网络权重的初值，从而使控制系统保持初始稳定，使系统的全局稳定成为可能。

(4) PIDNN 中包含了具有微分特性的处理单元，因此它是一种本质动态网络，容易适应控制系统的要求，在构成系统时不必在附加其他部件，它的输入信号无需经过网络外部的微分和积分处理。

(5) 比例元、积分元和微分元的存在，使 PIDNN 控制系统的响应快、超调小、无静差。

(6) PIDNN 采用无“教师”的自学习方式，根据控制效果进行在线学习和调整，使系统具备较好的性能。

(7) PIDNN 可以适应多种类型的单变量系统和多变量系统，无需测量或辨识控制对象的内部结构和参数^[5]。

3.3.2 PID 神经网络的结构形式

PIDNN 的结构形式随着网络输出量的个数的增减而变化，可分为单输出的 PIDNN(Single-output PIDNN, SPIDNN)和多输出的 PIDNN(Multi-output, PIDNN)。SPIDNN 用于单变量的控制，MPIDNN 用于多变量的控制（鉴于本文使用的缘故，对 MPIDNN 不作介绍）。SPIDNN 的基本形式为 $2 \times 3 \times 1$ 结构，如图 3.1 所示。

SPIDNN 输入层的神经元采用比例神经元，这两个神经元接受外部输入信息：其中一个接受单变量控制系统的目标值或给定值；另一个接受被控变量值。输入层神经元的输出经过连接权重，进入隐含层进行综合和处理，隐含层的三个神经元分别为比例元、积分元和微分元，分别对输入信号进行比例、积分和微分。隐含层的输出再通过连接权重，进入输出层，输出层完成整个网络控制规律的综合和输出。

SPIDNN 的结构简单，实现比较容易。如果采用硬件实现，PID 神经元均可用运算放大器和电阻、电容实现。当采用计算机软件实现时，PID 神经元的计算工作量均不大。下面是 SPIDNN 的比例元、积分元和微分元的定义以及前向算法和误差反相传播的学习

算法^[5]。

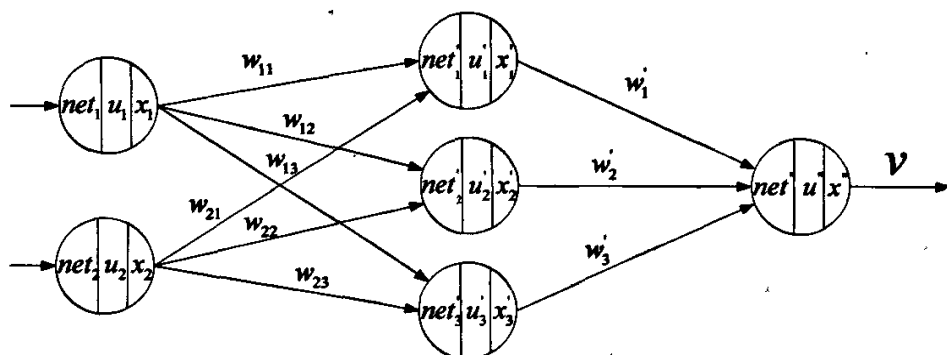


图 3.1 SPIDNN 的结构形式

3.3.3 PID 神经元

(1) 比例元

设比例元为神经网络中的第 j 个神经元，有 $n-1$ 个输入支路与其相连，则在任意 k 时刻，单元的总输入

$$net_j(k) = \sum_{i=1}^{n-1} w_{ji} x_i(k) \quad (3.1)$$

式中 $x_i(k)$ ($i=1,2,\dots,n-1, i \neq j$) 为 $n-1$ 个与其输入端相连的神经元的输出值， w_{ji} 为相应的连接权重值。

比例元的状态函数 $g_p(\bullet)$ 为比例函数，但其比例系数等于 1。比例元在 k 时刻的状态式为

$$u_j(k) = g_p[net_j(k)] = net_j(k) \quad (3.2)$$

比例元的输出函数可以选择非线性函数的一种，如果选择比例阈值函数，则输出为

$$x_j(k) = f[u_j(k)] = \begin{cases} 1, & u_j(k) > 1 \\ u_j(k), & -1 \leq u_j(k) \leq 1 \\ -1, & u_j(k) \leq -1 \end{cases} \quad (3.3)$$

比例元的输入—输出特性由其状态函数和输出函数共同决定，其输入—输出特性曲线如图 3.2 所示。

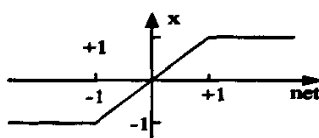


图 3.2 比例元输入—输出特性

算法^[5]。

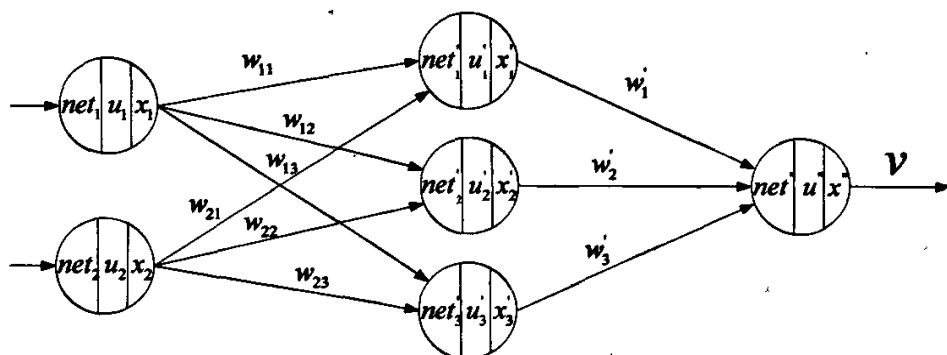


图 3.1 SPIDNN 的结构形式

3.3.3 PID 神经元

(1) 比例元

设比例元为神经网络中的第 j 个神经元，有 $n-1$ 个输入支路与其相连，则在任意 k 时刻，单元的总输入

$$net_j(k) = \sum_{i=1}^{n-1} w_{ji} x_i(k) \quad (3.1)$$

式中 $x_i(k)$ ($i=1,2,\dots,n-1, i \neq j$) 为 $n-1$ 个与其输入端相连的神经元的输出值， w_{ji} 为相应的连接权重值。

比例元的状态函数 $g_p(\bullet)$ 为比例函数，但其比例系数等于 1。比例元在 k 时刻的状态式为

$$u_j(k) = g_p[net_j(k)] = net_j(k) \quad (3.2)$$

比例元的输出函数可以选择非线性函数的一种，如果选择比例阈值函数，则输出为

$$x_j(k) = f[u_j(k)] = \begin{cases} 1, & u_j(k) > 1 \\ u_j(k), & -1 \leq u_j(k) \leq 1 \\ -1, & u_j(k) \leq -1 \end{cases} \quad (3.3)$$

比例元的输入—输出特性由其状态函数和输出函数共同决定，其输入—输出特性曲线如图 3.2 所示。

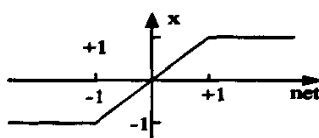


图 3.2 比例元输入—输出特性

(2) 积分元

设积分元为神经网络中的第 j 个神经元，有 $n-1$ 个输入支路与其相连，则在任意时刻，单元的总输入为

$$net_j(k) = \sum_{i=1}^{n-1} w_{ij} \cdot x_i(k) \quad (3.4)$$

式中： $x_i (i=1, 2, \dots, n-1; i \neq j)$ 为 $n-1$ 个与其输入端相连的神经元的输出值； w_{ij} 为相应的连接权重。

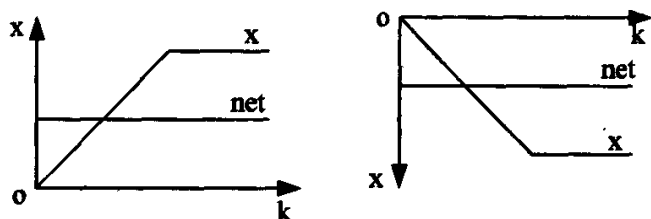
积分元的状态转换函数 $g_i(\bullet)$ 为积分函数，但其积分系数恒为 1。积分元在 k 时刻的状态式为

$$u_j(k) = g_i[net_j(k), u_j(k-1)] = u_j(k-1) + net_j(k) \quad (3.5)$$

积分元的输出函数与比例元的输出函数相同，积分元的输出为：

$$x_j(k) = f[u_j(k)] = \begin{cases} 1, & u_j(k) > 1 \\ u_j(k), & -1 \leq u_j(k) \leq 1 \\ -1, & u_j(k) \leq -1 \end{cases} \quad (3.6)$$

同样积分元的输入—输出特性由其状态函数和输出函数决定，如图 3.3 所示。



(a) net 为正阶跃信号

(b) net 为负阶跃信号

图 3.3 积分元输入—输出特性

(3) 微分元

设微分元为神经网络中的第 j 个神经元，有 $n-1$ 个输入支路与其相连，则在任意 k 时刻，单元的总输入为：

$$net_j(k) = \sum_{i=1}^{n-1} w_{ij} \cdot x_i(k) \quad (3.7)$$

式中： $x_i (i=1, 2, \dots, n-1; i \neq j)$ 为 $n-1$ 个与其输入端相连的神经元的输出值； w_{ij} 为相应的连接权重。

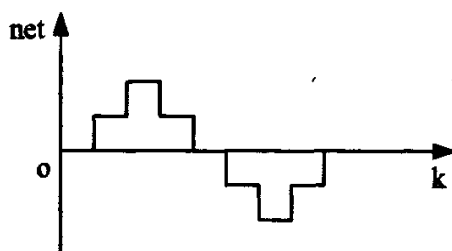
微分元的状态转换函数 $g_D(\bullet)$ 为有上下限幅的微分函数，但其系数恒为 1。微分元在 k 时刻的状态式为：

$$u_j(k) = g_D[net_j(k), net_j(k-1)] = net_j(k) - net_j(k-1) \quad (3.8)$$

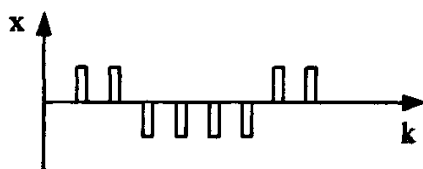
微分元的输出函数也与比例元的输出函数相同，微分元的输出为：

$$x_j(k) = f[u_j(k)] = \begin{cases} 1, & u_j(k) > 1 \\ u_j(k), & -1 \leq u_j(k) \leq 1 \\ -1, & u_j(k) \leq -1 \end{cases} \quad (3.9)$$

微分元的输入—输出关系如图 3.4 所示，如果微分元的输入为图 3.4(a) 所示曲线，则微分元的输出即为图 3.4(b) 所示的形式^[5]。



(a) 微分元的输入曲线



(b) 微分元的输出曲线

图 3.4 微分元输入—输出特性

3.3.4 SPIDNN 的算法

(1) 前向算法

SPIDNN 的前向算法根据网络的两个输入值，按网络当前权重值和各层状态函数和输出函数形成网络的输出。

① 输入层

SPIDNN 的输入层有两个神经元，在构成控制系统时分别输入被调量的给定值和实际值。在任意采样时刻 k ，其输入为：

$$\begin{aligned} net_1(k) &= r(k) \\ net_2(k) &= y(k) \end{aligned} \quad (3.10)$$

输入层神经元的状态为：

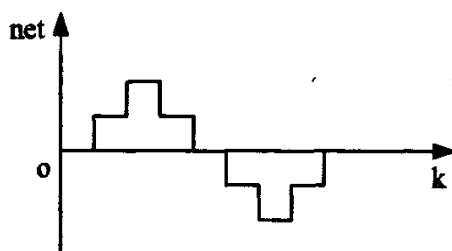
$$u_i(k) = net_i(k) \quad (3.11)$$

输入层神经元的输出为：

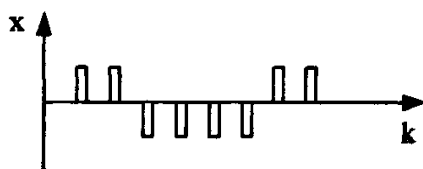
微分元的输出函数也与比例元的输出函数相同，微分元的输出为：

$$x_j(k) = f[u_j(k)] = \begin{cases} 1, & u_j(k) > 1 \\ u_j(k), & -1 \leq u_j(k) \leq 1 \\ -1, & u_j(k) \leq -1 \end{cases} \quad (3.9)$$

微分元的输入—输出关系如图 3.4 所示，如果微分元的输入为图 3.4(a) 所示曲线，则微分元的输出即为图 3.4(b) 所示的形式^[5]。



(a) 微分元的输入曲线



(b) 微分元的输出曲线

图 3.4 微分元输入—输出特性

3.3.4 SPIDNN 的算法

(1) 前向算法

SPIDNN 的前向算法根据网络的两个输入值，按网络当前权重值和各层状态函数和输出函数形成网络的输出。

① 输入层

SPIDNN 的输入层有两个神经元，在构成控制系统时分别输入被调量的给定值和实际值。在任意采样时刻 k ，其输入为：

$$\begin{aligned} net_1(k) &= r(k) \\ net_2(k) &= y(k) \end{aligned} \quad (3.10)$$

输入层神经元的状态为：

$$u_i(k) = net_i(k) \quad (3.11)$$

输入层神经元的输出为：

$$x_i(k) = \begin{cases} 1, & u_i(k) > 1 \\ u_i(k), & -1 \leq u_i(k) \leq 1 \\ -1, & u_i(k) \leq -1 \end{cases} \quad (3.12)$$

以上各式中 $i=1, 2$ 。

② 隐含层

隐含层是神经网络中最重要的层次，SPIDNN 的隐含层有三个神经元，分别为比例元、积分元、微分元，它们各自的输入总值为：

$$net'_j(k) = \sum_{i=1}^2 w_{ji} x_i(k) \quad (3.13)$$

式中： $j=1, 2, 3$ ； w_{ji} 为输入层至隐含层的连接权重值；上标“'”为隐含层变量标记。

比例元的状态为：

$$u'_1(k) = net'_1(k) \quad (3.14)$$

积分元的状态为：

$$u'_2(k) = u'_2(k-1) + net'_2(k) \quad (3.15)$$

微分元的状态为：

$$u'_3(k) = net'_3(k) - net'_3(k-1) \quad (3.16)$$

隐含层各神经元的输出为：

$$x'_j(k) = \begin{cases} 1, & u'_j(k) > 1 \\ u'_j(k), & -1 \leq u'_j(k) \leq 1 \\ -1, & u'_j(k) < -1 \end{cases} \quad (3.17)$$

式中 $j=1, 2, 3$ 。

③ 输出层

SPIDNN 的输出层结构比较简单，只包含一个神经元，完成网络的综合输出功能，其总输入为：

$$net''(k) = \sum_{j=1}^3 w_j x'_j(k) \quad (3.18)$$

式中： $x'_j(k)$ 为隐含层各神经元的输出值； w_j 为隐含层至输出层的连接权重值。

输出层神经元的状态函数与比例元的状态函数相同，状态为：

$$u''(k) = net''(k) \quad (3.19)$$

输出层神经元的输出函数与网络中其他神经元的输出函数相同，输出 $x''(k)$ 为：

$$x^*(k) = \begin{cases} 1, & u^*(k) > 1 \\ u^*(k), & -1 \leq u^*(k) \leq 1 \\ -1, & u^*(k) \leq -1 \end{cases} \quad (3.20)$$

而 SPIDNN 的输出 $v(k)$ 就等于神经网络输出层的输出, 即

$$v(k) = x^*(k) \quad (3.21)$$

(2) 反传算法

反传算法即误差反向传播学习算法, 其完成网络权重值的修改, 完成学习和记忆的功能。SPIDNN 的反传算法与传统多层前向神经元的反传算法类似。

对于图 3.1 所示的 SPIDNN, 设训练输入的集合为:

$$R = \{R_1, R_2, \dots, R_p, \dots\} \quad (3.22)$$

其中某个训练输入矢量为:

$$R_p = (r_{p1}, r_{p2})^T \quad (3.23)$$

式中

$$r_{pi} = \{r_{pi}(1), r_{pi}(2), \dots, r_{pi}(l)\} \quad (3.24)$$

为输入量时间序列, l 为采样点数, $i=1, 2$ 。

SPIDNN 的实际输出矢量的集合为:

$$V = (v_1, v_2, \dots, v_p, \dots)^T \quad (3.25)$$

式中

$$v_p = \{v_p(1), v_p(2), \dots, v_p(l)\} \quad (3.26)$$

为输出量的时间序列, l 为采样点数。

SPIDNN 的理想输出矢量的集合为:

$$V' = \{v'_1, v'_2, \dots, v'_p, \dots\} \quad (3.27)$$

式中

$$v'_p = \{v'_p(1), v'_p(2), \dots, v'_p(l)\} \quad (3.28)$$

为理想输出量的时间序列, l 为采样点数。

SPIDNN 训练和学习的目的是使网络实际输出和理想输出的对应时间序列的偏差平方均值

$$E = \frac{1}{l} \sum_{k=1}^l [v'(k) - v(k)]^2 \quad (3.29)$$

为最小。

按梯度法调节 SPIDNN 权重值, 经 n_0 步训练和学习后, SPIDNN 各权重的迭代方程为:

$$W(n_0+1) = W(n_0) - \eta \frac{\partial E}{\partial W} \quad (3.30)$$

隐含层至输出层的权重 w_j 和输入层至隐含层的权重值 w_j 的具体计算方法分别由以下各式确定。

1. 隐含层至输出层的权重迭代公式为:

$$w_j'(n_0+1) = w_j'(n_0) - \eta_j \frac{\partial E}{\partial w_j'} \quad (3.31)$$

式中

$$\frac{\partial E}{\partial w_j'} = \frac{\partial E}{\partial v} \cdot \frac{\partial v}{\partial x^*} \cdot \frac{\partial x^*}{\partial u^*} \cdot \frac{\partial u^*}{\partial net^*} \cdot \frac{\partial net^*}{\partial w_j'} \quad (3.32)$$

将式 (3.29) 代入, 可求出

$$\frac{\partial}{\partial v} \left(\frac{1}{l} \sum_{k=1}^l [v'(k) - v(k)]^2 \right) = -\frac{2}{l} \sum_{k=1}^l [v'(k) - v(k)] \quad (3.33)$$

将式 (3.21) 代入, 可求得

$$\frac{\partial v}{\partial x^*} = \frac{\partial x^*}{\partial x} = 1 \quad (3.34)$$

将式 (3.20) 代入, 可求出

$$\frac{\partial x^*}{\partial u^*} = \frac{\partial u^*}{\partial u} = 1 \quad (3.35)$$

将式 (3.19) 代入, 可求出

$$\frac{\partial u^*}{\partial net^*} = \frac{\partial net^*}{\partial net} = 1 \quad (3.36)$$

由式 (3.17) 和式 (3.18), 可求出

$$\frac{\partial net^*}{\partial w_j'} = \frac{\partial}{\partial w_j'} \left(\sum_{j=1}^3 w_j' x_j'(k) \right) = x_j'(k) \quad (3.37)$$

将式 (3.33) ~ 式 (3.37) 代入式 (3.32) 可得

$$\frac{\partial E}{\partial w_j'} = -\frac{2}{l} \sum_{k=1}^l [v'(k) - v(k)] x_j'(k) \quad (3.38)$$

令

$$\delta'(k) = 2[v'(k) - v(k)] \quad (3.39)$$

式 (3.38) 可简化为:

$$\frac{\partial E}{\partial w_j} = -\frac{1}{l} \sum_{k=1}^l \delta'(k) x_j'(k) \quad (3.40)$$

将式 (3.40) 代入式 (3.31) 得

$$w_j'(n_0+1) = w_j'(n_0) + \frac{\eta_j}{l} \sum_{k=1}^l \sigma'(k) x_j'(k) \quad (3.41)$$

2. 输入层至隐含层

输入层至隐含层的权重值迭代公式为:

$$w_{ij}(n_0+1) = w_{ij}(n_0) - \eta_{ij} \frac{\partial E}{\partial w_{ij}} \quad (3.42)$$

由误差反向传播关系, 其中

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial v} \cdot \frac{\partial v}{\partial x_j^*} \cdot \frac{\partial x_j^*}{\partial u_j^*} \cdot \frac{\partial u_j^*}{\partial net_j^*} \cdot \frac{\partial net_j^*}{\partial x_j} \cdot \frac{\partial x_j}{\partial u_j'} \cdot \frac{\partial u_j'}{\partial net_j'} \cdot \frac{\partial net_j'}{\partial w_{ij}} \quad (3.43)$$

由式 (3.35) ~ 式 (3.37) 结果, 可以得出

$$\frac{\partial E}{\partial v} \cdot \frac{\partial v}{\partial x_j^*} \cdot \frac{\partial x_j^*}{\partial u_j^*} \cdot \frac{\partial u_j^*}{\partial net_j^*} = -\frac{1}{l} \sum_{k=1}^l \delta(k) \quad (3.44)$$

由式 (3.18), 可求出

$$\frac{\partial net_j^*}{\partial x_j} = \frac{\partial}{\partial x_j} \sum_{j=1}^3 w_j x_j' = w_j' \quad (3.45)$$

由式 (3.17), 可求出

$$\frac{\partial x_j}{\partial u_j'} = \frac{\partial u_j'}{\partial u_j'} = 1 \quad (3.46)$$

式 (3.43) 中的

$$\frac{\partial u_j'}{\partial net_j'} \approx \frac{\Delta u_j'}{\Delta net_j'} = \frac{u_j'(k) - u_j'(k-1)}{net_j'(k) - net_j'(k-1)} \quad (3.47)$$

很显然, 如果 $net_j'(k)$ 和 $net_j'(k-1)$ 相等或相差很小时, 上式的结果可能会趋于无穷大,

计算机无法进行计算, 因此需要采用近似方法。由式 (3.43) 可知, 式 (3.47) 为式 (3.44)

中的一个乘积因子, $\frac{\partial u_j'}{\partial net_j'}$ 绝对值的大小影响权重值收敛变化速度, $\frac{\partial u_j'}{\partial net_j'}$ 的正负则影响

权重收敛的方向。对于神经网络权重的收敛, 收敛方向是最重要的, 收敛速度还可以

通过其他因子调节。所以, 可以用 net_j' 和 u_j' 的相对变化量的符号函数近似代替式 (3.47),

即采用

$$\operatorname{sgn} \frac{u_j'(k) - u_j'(k-1)}{net_j'(k) - net_j'(k-1)} \quad (3.48)$$

来代替 $\frac{\partial u_j'}{\partial net_j'}$ ，使计算简化可行。结果表明，这种代替对学习效果没有明显影响。

由式 (3.13)，可求出

$$\frac{\partial net_j'}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} (x_i(k) \cdot w_{ij}) = x_i(k) \quad (3.49)$$

将式 (3.44) ~ 式 (3.49) 代入式 (3.43)，可得

$$\frac{\partial E}{\partial w_{ij}} = -\frac{1}{l} \sum_{k=1}^l \delta_j'(k) \cdot w_{ij}' \cdot \operatorname{sgn} \frac{u_j'(k) - u_j'(k-1)}{net_j'(k) - net_j'(k-1)} \cdot x_i(k) \quad (3.50)$$

令

$$\delta(k) = \delta_j'(k) \cdot w_{ij}' \cdot \operatorname{sgn} \frac{u_j'(k) - u_j'(k-1)}{net_j'(k) - net_j'(k-1)} \quad (3.51)$$

则式 (3.50) 可简化为：

$$\frac{\partial E}{\partial w_{ij}} = -\frac{1}{l} \sum_{k=1}^l \delta_j(k) x_i(k) \quad (3.52)$$

由以上各式可得到 SPIDNN 输入层至隐含层的权重值的公式修改为：

$$w_{ij}(n_0+1) = w_{ij}(n_0) + \frac{\eta_{ij}}{l} \sum_{k=1}^l \delta_j(k) x_i(k) \quad (3.53)$$

由 SPIDNN 的算法可知，SPIDNN 的算法没有依赖被控对象参数，也不需要对被控对象的数学模型进行辨识，而是依据控制效果在线自主学习，调整网络连接权重值，实现控制作用的^[5]。

3.4 PID 神经网络单变量控制系统

控制一个单变量系统，需要一个 SPIDNN，构成控制系统。SPIDNN 和被控对象是一种串联关系，SPIDNN 网络的输入为被控量设计值和实际值，网络的输出作为控制量送入被控对象的输入端，如图 3.5 所示。

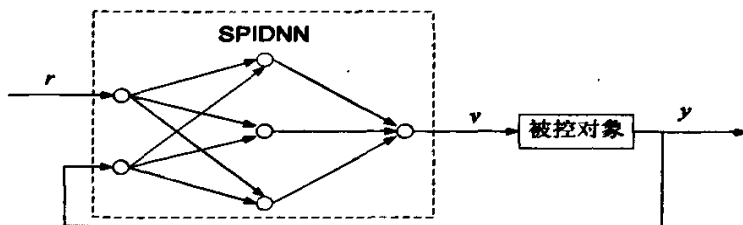


图 3.5 PIDNN 单变量控制系统结构

$$\operatorname{sgn} \frac{u_j'(k) - u_j'(k-1)}{net_j'(k) - net_j'(k-1)} \quad (3.48)$$

来代替 $\frac{\partial u_j'}{\partial net_j'}$ ，使计算简化可行。结果表明，这种代替对学习效果没有明显影响。

由式 (3.13)，可求出

$$\frac{\partial net_j'}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} (x_i(k) \cdot w_{ij}) = x_i(k) \quad (3.49)$$

将式 (3.44) ~ 式 (3.49) 代入式 (3.43)，可得

$$\frac{\partial E}{\partial w_{ij}} = -\frac{1}{l} \sum_{k=1}^l \delta_j'(k) \cdot w_{ij}' \cdot \operatorname{sgn} \frac{u_j'(k) - u_j'(k-1)}{net_j'(k) - net_j'(k-1)} \cdot x_i(k) \quad (3.50)$$

令

$$\delta(k) = \delta_j'(k) \cdot w_{ij}' \cdot \operatorname{sgn} \frac{u_j'(k) - u_j'(k-1)}{net_j'(k) - net_j'(k-1)} \quad (3.51)$$

则式 (3.50) 可简化为：

$$\frac{\partial E}{\partial w_{ij}} = -\frac{1}{l} \sum_{k=1}^l \delta_j(k) x_i(k) \quad (3.52)$$

由以上各式可得到 SPIDNN 输入层至隐含层的权重值的公式修改为：

$$w_{ij}(n_0+1) = w_{ij}(n_0) + \frac{\eta_{ij}}{l} \sum_{k=1}^l \delta_j(k) x_i(k) \quad (3.53)$$

由 SPIDNN 的算法可知，SPIDNN 的算法没有依赖被控对象参数，也不需要对被控对象的数学模型进行辨识，而是依据控制效果在线自主学习，调整网络连接权重值，实现控制作用的^[5]。

3.4 PID 神经网络单变量控制系统

控制一个单变量系统，需要一个 SPIDNN，构成控制系统。SPIDNN 和被控对象是一种串联关系，SPIDNN 网络的输入为被控量设计值和实际值，网络的输出作为控制量送入被控对象的输入端，如图 3.5 所示。

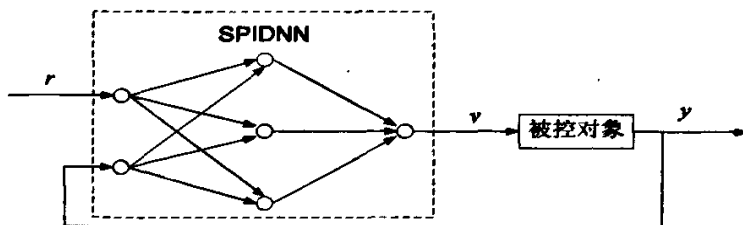


图 3.5 PIDNN 单变量控制系统结构

如果控制系统在控制的整个过程中保持稳定，称之为全过程稳定。稳定性是控制系统最重要的性能之一，神经网络控制系统也必须满足稳定性要求，否则将失去使用价值。全过程稳定的前提条件是神经网络的初始稳定，在此基础上收敛，才有可能实现全过程稳定。神经网络的结构和连接权重初值决定系统是否初始稳定，神经网络控制系统的学习步长则决定了收敛性。

SPIDNN 控制收敛时的学习步长范围是用李雅普诺夫 (Lyapunov) 稳定性原理获得的，从而可以证明系统收敛时是稳定的 (证明从略)。

3.5 本章小结

本章主要论述了以下三方面的内容：

(1) 阐述了 PIDNN 的特点：以经验 PID 参数为连接权重初值，通过训练修改可使权重达到最佳匹配；又可以避免训练过程中出现局部极小值现象，从而是使控制系统的性能最优；

(2) 讨论了 PIDNN 的结构形式与算法；

(3) 建立了 SPIDNN 控制系统模型。

如果控制系统在控制的整个过程中保持稳定，称之为全过程稳定。稳定性是控制系统最重要的性能之一，神经网络控制系统也必须满足稳定性要求，否则将失去使用价值。全过程稳定的前提条件是神经网络的初始稳定，在此基础上收敛，才有可能实现全过程稳定。神经网络的结构和连接权重初值决定系统是否初始稳定，神经网络控制系统的学习步长则决定了收敛性。

SPIDNN 控制收敛时的学习步长范围是用李雅普诺夫 (Lyapunov) 稳定性原理获得的，从而可以证明系统收敛时是稳定的 (证明从略)。

3.5 本章小结

本章主要论述了以下三方面的内容：

(1) 阐述了 PIDNN 的特点：以经验 PID 参数为连接权重初值，通过训练修改可使权重达到最佳匹配；又可以避免训练过程中出现局部极小值现象，从而是使控制系统的性能最优；

(2) 讨论了 PIDNN 的结构形式与算法；

(3) 建立了 SPIDNN 控制系统模型。

4 基于 PID 神经网络的提拉系统设计

4.1 设计的总体任务

针对某厂 TDR-50 型单晶生长炉设计一套以 PID 神经网络为控制核心的单晶硅提拉速度控制试验系统, 为下一步全面改造系统作准备。目标系统相关参数如下:

最高熔炼温度: 1420°C ;

最大投料量: 8kg;

晶棒规格: $\phi 70\text{mm} \times 870\text{mm}$;

提拉速度: $1 \sim 10\text{mm/min}$;

提拉角速度: $0.3 \sim 3\text{n/min}$;

电机调速范围: $150 \sim 1500\text{n/min}$;

提拉行程: 1250mm。

4.2 提拉工作过程

在工艺流程中, 最关键的是拉晶过程, 即: 润晶、缩颈、放肩、等径生长、拉光等步骤。当熔体温度稳定在稍高于拉晶温度时, 将籽晶润晶, 然后进行缩颈, 放肩到正常的直径, 等径生长, 最后将熔体全部拉光, 如图 2.4 所示。拉制过程中各段的长度如图 4.1 所示。其中 h_0 为籽晶长度、 h_1 拉制启动段形成的长度、 h_2 缩颈段长度、 h_3 放肩段长度、 h_4 等径段长度、 h_5 拉光段长度。

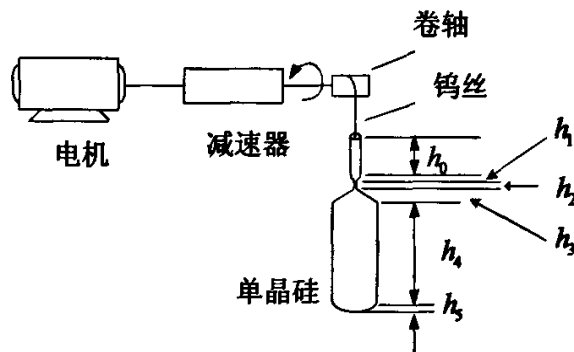


图 4.1 单晶硅提拉过程示意图

(1) 单晶硅的速度—直径关系

依据第 2 章的推导, 可以得出单晶硅的提拉速度—直径关系如图 4.2 所示。图中横轴表示直径, 单位 mm; 纵轴表示速度, 单位 mm/min, 依据图 4.2 并结合拉晶过程可以得出如图 4.3 所示的提拉过程速度曲线。

4 基于 PID 神经网络的提拉系统设计

4.1 设计的总体任务

针对某厂 TDR-50 型单晶生长炉设计一套以 PID 神经网络为控制核心的单晶硅提拉速度控制试验系统, 为下一步全面改造系统作准备。目标系统相关参数如下:

最高熔炼温度: 1420°C ;

最大投料量: 8kg ;

晶棒规格: $\phi 70\text{mm} \times 870\text{mm}$;

提拉速度: $1 \sim 10\text{mm/min}$;

提拉角速度: $0.3 \sim 3\text{n/min}$;

电机调速范围: $150 \sim 1500\text{n/min}$;

提拉行程: 1250mm 。

4.2 提拉工作过程

在工艺流程中, 最关键的是拉晶过程, 即: 润晶、缩颈、放肩、等径生长、拉光等步骤。当熔体温度稳定在稍高于拉晶温度时, 将籽晶润晶, 然后进行缩颈, 放肩到正常的直径, 等径生长, 最后将熔体全部拉光, 如图 2.4 所示。拉制过程中各段的长度如图 4.1 所示。其中 h_0 为籽晶长度、 h_1 拉制启动段形成的长度、 h_2 缩颈段长度、 h_3 放肩段长度、 h_4 等径段长度、 h_5 拉光段长度。

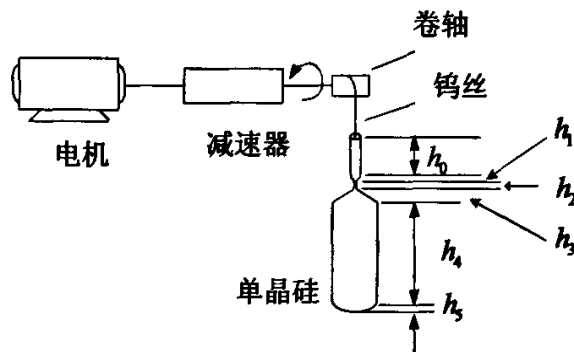


图 4.1 单晶硅提拉过程示意图

(1) 单晶硅的速度—直径关系

依据第 2 章的推导, 可以得出单晶硅的提拉速度—直径关系如图 4.2 所示。图中横轴表示直径, 单位 mm ; 纵轴表示速度, 单位 mm/min , 依据图 4.2 并结合拉晶过程可以得出如图 4.3 所示的提拉过程速度曲线。

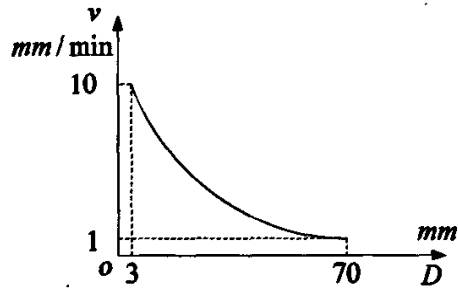


图 4.2 提拉速度—直径关系曲线

图 4.3 中, $t_0 \sim t_1$ 时间电机启动由零上升到 v_1 , 对应图 4.1 中的 h_1 ; $t_1 \sim t_2$ 时间进行缩颈, 如图 4.1 中的 h_2 ; $t_2 \sim t_3$ 时间进行放肩, 如图 4.1 中的 h_3 ; $t_3 \sim t_4$ 时间为等径生长段, 如图 4.1 中的 h_4 ; $t_4 \sim t_5$ 时间进行拉光, 对应图 4.1 中的 h_5 , 至此晶体拉制结束, 电机停车。

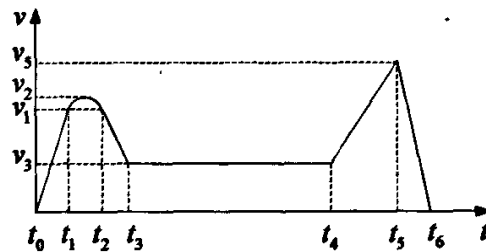


图 4.3 提拉过程速度曲线

4.3 提拉系统设计

按第 2.2 节确定的系统结构建立系统模型如图 4.4 所示。

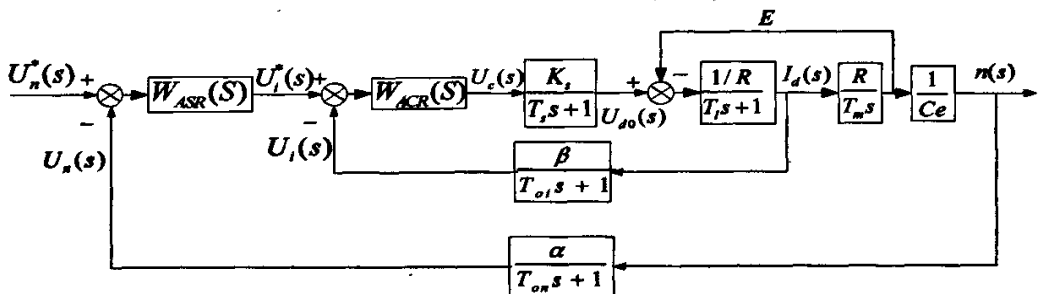


图 4.4 提拉速度控制系统的动态结构

(1) 电机模型

直流电机在额定励磁下的等效电路如图 4.5 所示:

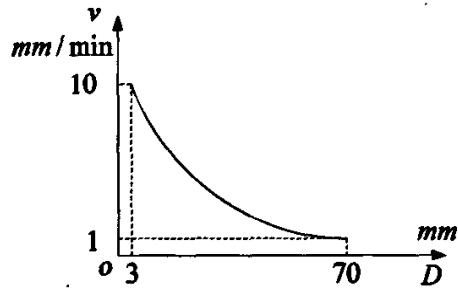


图 4.2 提拉速度—直径关系曲线

图 4.3 中, $t_0 \sim t_1$ 时间电机启动由零上升到 v_1 , 对应图 4.1 中的 h_1 ; $t_1 \sim t_2$ 时间进行缩颈, 如图 4.1 中的 h_2 ; $t_2 \sim t_3$ 时间进行放肩, 如图 4.1 中的 h_3 ; $t_3 \sim t_4$ 时间为等径生长段, 如图 4.1 中的 h_4 ; $t_4 \sim t_5$ 时间进行拉光, 对应图 4.1 中的 h_5 , 至此晶体拉制结束, 电机停车。

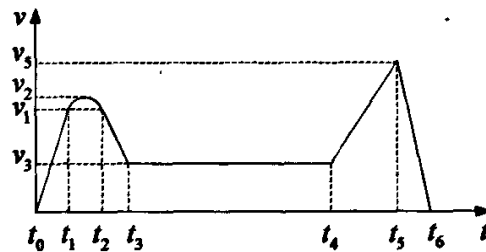


图 4.3 提拉过程速度曲线

4.3 提拉系统设计

按第 2.2 节确定的系统结构建立系统模型如图 4.4 所示。

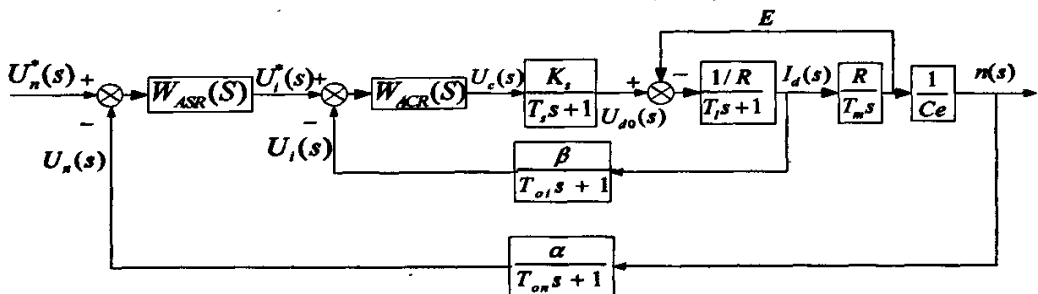


图 4.4 提拉速度控制系统的动态结构

(1) 电机模型

直流电机在额定励磁下的等效电路如图 4.5 所示:

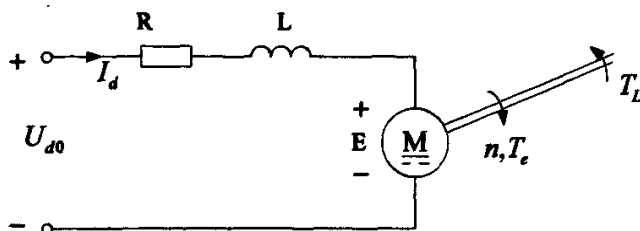


图 4.5 额定励磁下直流电动机等效电路

其中电枢回路总电阻 R 和电感 L 包含电力电子变换器、电枢电阻和电感以及在主电路中接入的其他电阻和电感，规定的正方向如图所示。

直流伺服电机 (90sz55) 与减速器参数如下：

额定电压： $U_N = 24V$ ；

额定电流： $I_N = 5A$ ；

额定功率： $P_N = 80W$ ；

额定转速： $n_N = 1500 r/min$ ；

转矩系数： $C_m = 0.1 N \cdot m / A$ ；

感应电动势系数： $C_e = 0.01 N \cdot m / A$ ；

电枢回路总电阻： $R = 1.6 \Omega$ ；

电枢回路总电感： $L = 2.6mH$ ；

电动机及负载折算到电机轴上的转矩为： $DG^2 = 0.01 N \cdot m^2$

机电时间常数为： $T_m = \frac{DG^2 R}{375 C_m C_e} = 0.048s$ ；

电磁时间常数为： $T_l = \frac{L}{R} = 0.016s$ ；

减速器选用行星减速机，减速比为 500:1。

在主电路电流连续的情况下，电机动态电压方程为：

$$U_{d0} = RI_d + L \frac{dI_d}{dt} + E \quad (4.1)$$

忽略粘性摩擦及弹性转矩，电动机轴上的动力学方程为：

$$T_e - T_L = \frac{GD^2}{375} \frac{dn}{dt} \quad (4.2)$$

额定励磁下的感应电动势和电磁转矩分别为：

$$E = C_e n \quad (4.3)$$

$$T_e = C_m I_d \quad (4.4)$$

式中 T_L ：包括电动机空载转矩在内的负载转矩 ($N \cdot m$)；

GD^2 : 电力拖动系统折算到电动机轴上的飞轮惯量 ($N \cdot m^2$);

C_m : 额定励磁下电动机的转矩系数 ($N \cdot m / A$)

$$C_m = \frac{30}{\pi} C_e \quad (4.5)$$

令 T_l 为电枢回路电磁时间常数 (s)

$$T_l = \frac{L}{R}; \quad (4.6)$$

T_m 为电力拖动系统机电时间常数 (s)

$$T_m = \frac{GD^2 R}{375 C_e C_m} \quad (4.7)$$

代入式 (4.1) 和式 (4.2), 并考虑式 (4.3) 和 (4.4), 整理得:

$$U_{d0} - E = R \left(I_d + T_l \frac{dI_d}{dt} \right) \quad (4.8)$$

$$I_d - I_a = \frac{T_m}{R} \frac{dE}{dt} \quad (4.9)$$

式中 I_a : 负载电流 (A), $I_a = \frac{T_L}{C_m}$ 。

在零初始条件下, 取等式两侧的拉氏变换, 得电压与电流间的传递函数为:

$$\frac{I_d(s)}{U_{d0}(s) - E(s)} = \frac{1}{T_l s + 1} \quad (4.10)$$

电流与电动势间的传递函数为:

$$\frac{E(s)}{I_d(s) - I_a(s)} = \frac{R}{T_m s} \quad (4.11)$$

式 (4.10) 和 (4.11) 的动态结构框图分别画在图 4.6 中的 (a) 和 (b) 中。将两图合在一起, 并考虑到 $n = \frac{E}{C_e}$, 即得额定励磁下直流电动机的动态结构框图如图 4.6(c) 所示。

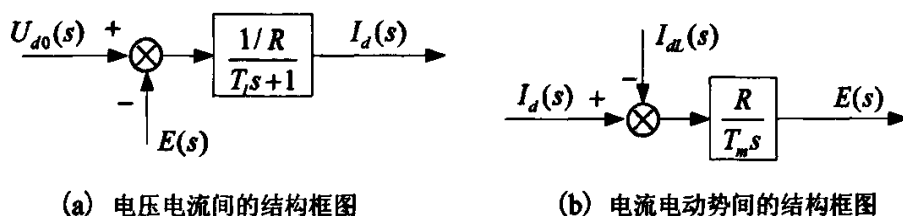
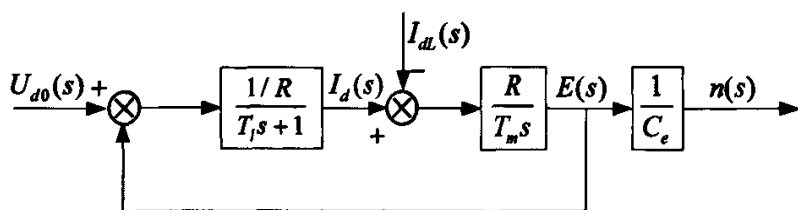


图 4.6 额定励磁下直流电动机的动态结构框图



(c) 直流电动机的动态结构框图

图 4.6 额定励磁下直流电动机的动态结构框图

根据电机参数计算可得直流电机数学模型如图 4.7 所示。

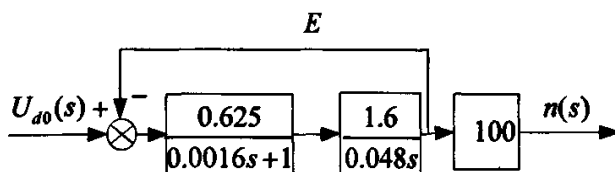


图 4.7 电机模型

(2) 功率转换器为

$$\frac{U_{d0}(s)}{U_c(s)} = \frac{K_s}{T_s s + 1} \quad (4.12)$$

式中, K_s 为功率转换器放大系数 $K_s = 24/5 = 4.8$; $T_s = 0.025ms$ 为滞后时间常数, 很小可以忽略不计。

(3) 电流调节器设计

取 $\beta = 1$, $T_\alpha = 0.0014s$, 则电流环滤波传函

$$\frac{\beta}{T_\alpha s + 1} = \frac{1}{0.0014s + 1} \quad (4.13)$$

按双闭环系统的工程设计方法, 将电流环开环传函校正为典型 I 型系统, 则

$$W_{ACR}(s) = \frac{0.21s + 134}{s} \quad (4.14)$$

电流环的结构如图 4.8 所示

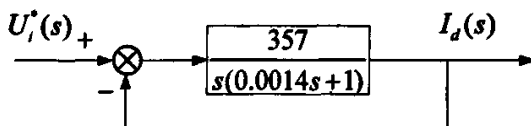


图 4.8 电流环结构图

(4) 转速调节器设计

电流环近似处理后的等效传函为:

$$\frac{I_d(s)}{U_i^*(s)} = \frac{1}{0.0028s+1} \quad (4.15)$$

取 $\alpha = 0.003$, $T_{om} = 0.0025ms$, 则转速环滤波传函为:

$$\frac{\alpha}{T_{om}s+1} = \frac{0.0033}{0.0025s+1} \quad (4.16)$$

则系统可简化为图 4.9 所示等效模型。

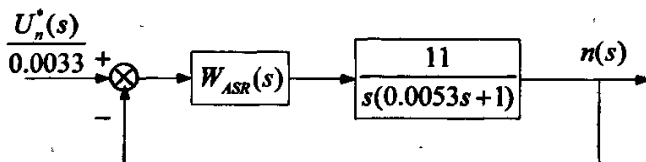


图 4.9 系统等效模型

将转速环校正为典型 II 型系统, 按“对称最佳”三阶系统, 可得转速调节器

$$W_{ASR}(s) = \frac{7.6s+358}{s} \quad (4.17)$$

(5) 根据以上计算得提拉系统数学模型如图 4.10 所示。

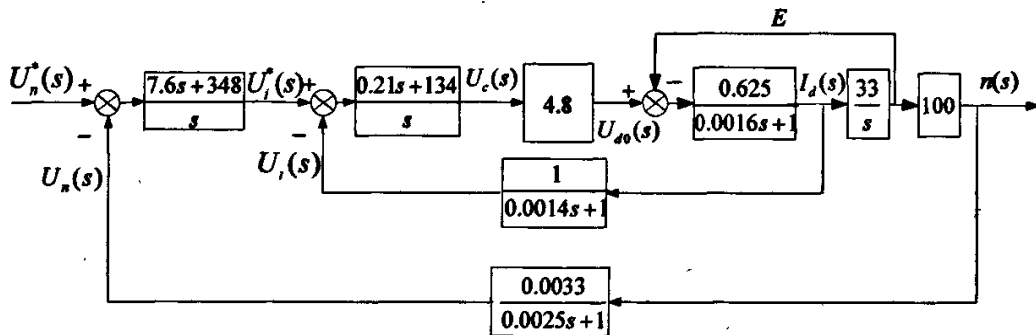


图 4.10 提拉系统数学模型

4.4 仿真

4.4.1 PID 仿真

按照 4.3 节的结构在 MATLAB/Simulink 下进行 PID 控制的仿真建模, 模型如图 4.11 所示。图中 Contant 表示单位阶跃输入, Transfer Fcn 表示速度给定滤波, Transfer Fcn7 表示速度反馈与速度滤波, PI 表示速度 PI 调节器, Transfer Fcn1 为电流给定滤波, Transfer Fcn6 为电流反馈与滤波, PI1 为电流调节器, Transfer Fcn2 为功率转换器, Transfer Fcn3 为电机电磁环节, Transfer Fcn4 为机电环节, Ramp 为连续增长的斜坡负载, Gain 表示 $1/C_e = 100$, Scope 为速度输出示波器, Clock 为仿真时钟, Scope 为时间输出示波器。

$$\frac{I_d(s)}{U_i^*(s)} = \frac{1}{0.0028s+1} \quad (4.15)$$

取 $\alpha = 0.003$, $T_{om} = 0.0025ms$, 则转速环滤波传函为:

$$\frac{\alpha}{T_{om}s+1} = \frac{0.0033}{0.0025s+1} \quad (4.16)$$

则系统可简化为图 4.9 所示等效模型。

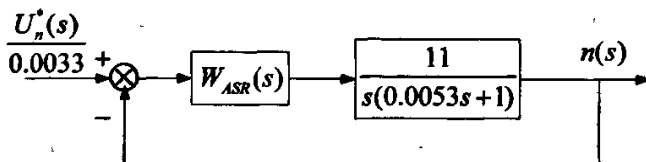


图 4.9 系统等效模型

将转速环校正为典型 II 型系统, 按“对称最佳”三阶系统, 可得转速调节器

$$W_{ASR}(s) = \frac{7.6s+358}{s} \quad (4.17)$$

(5) 根据以上计算得提拉系统数学模型如图 4.10 所示。

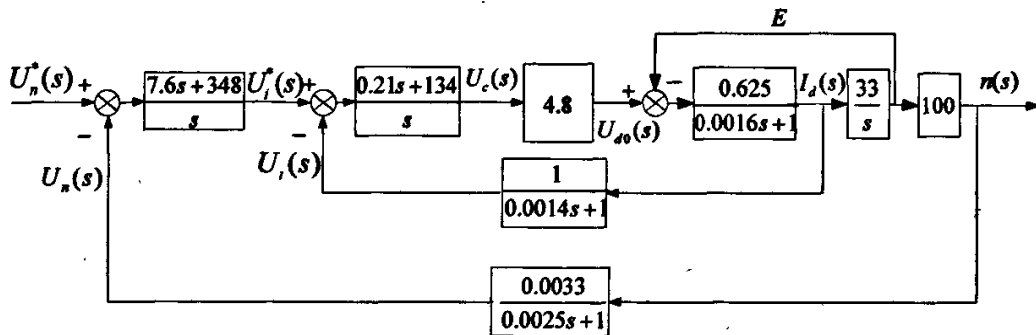


图 4.10 提拉系统数学模型

4.4 仿真

4.4.1 PID 仿真

按照 4.3 节的结构在 MATLAB/Simulink 下进行 PID 控制的仿真建模, 模型如图 4.11 所示。图中 Contant 表示单位阶跃输入, Transfer Fcn 表示速度给定滤波, Transfer Fcn7 表示速度反馈与速度滤波, PI 表示速度 PI 调节器, Transfer Fcn1 为电流给定滤波, Transfer Fcn6 为电流反馈与滤波, PI1 为电流调节器, Transfer Fcn2 为功率转换器, Transfer Fcn3 为电机电磁环节, Transfer Fcn4 为机电环节, Ramp 为连续增长的斜坡负载, Gain 表示 $1/C_e = 100$, Scope 为速度输出示波器, Clock 为仿真时钟, Scope 为时间输出示波器。

$$\frac{I_d(s)}{U_i^*(s)} = \frac{1}{0.0028s+1} \quad (4.15)$$

取 $\alpha = 0.003$, $T_{om} = 0.0025ms$, 则转速环滤波传函为:

$$\frac{\alpha}{T_{om}s+1} = \frac{0.0033}{0.0025s+1} \quad (4.16)$$

则系统可简化为图 4.9 所示等效模型。

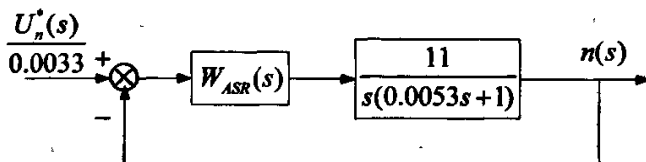


图 4.9 系统等效模型

将转速环校正为典型 II 型系统, 按“对称最佳”三阶系统, 可得转速调节器

$$W_{ASR}(s) = \frac{7.6s+358}{s} \quad (4.17)$$

(5) 根据以上计算得提拉系统数学模型如图 4.10 所示。

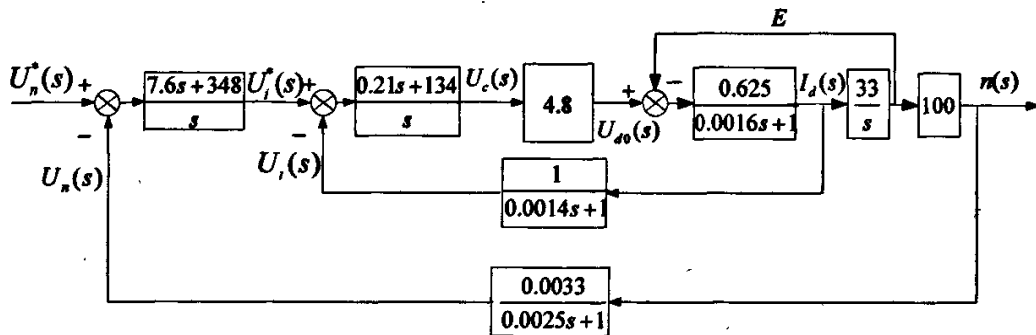


图 4.10 提拉系统数学模型

4.4 仿真

4.4.1 PID 仿真

按照 4.3 节的结构在 MATLAB/Simulink 下进行 PID 控制的仿真建模, 模型如图 4.11 所示。图中 Contant 表示单位阶跃输入, Transfer Fcn 表示速度给定滤波, Transfer Fcn7 表示速度反馈与速度滤波, PI 表示速度 PI 调节器, Transfer Fcn1 为电流给定滤波, Transfer Fcn6 为电流反馈与滤波, PI1 为电流调节器, Transfer Fcn2 为功率转换器, Transfer Fcn3 为电机电磁环节, Transfer Fcn4 为机电环节, Ramp 为连续增长的斜坡负载, Gain 表示 $1/C_e = 100$, Scope 为速度输出示波器, Clock 为仿真时钟, Scope 为时间输出示波器。

单位阶跃输入时, 系统速度响应及示波器 Scope 的输出如图 4.12 所示。从图中可以看出系统速度调节器为 PID 调节器时, 系统速度响应有超调, 且调节时间比较长。

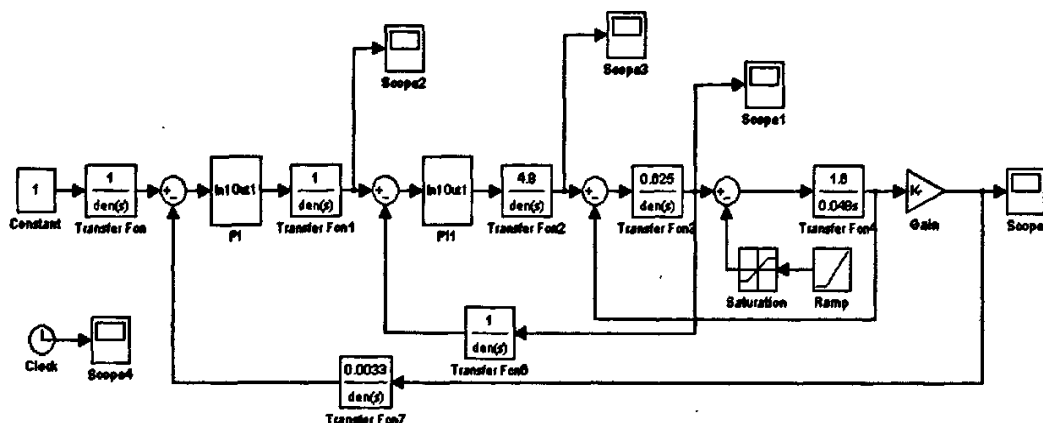


图 4.11 系统 PID 仿真图

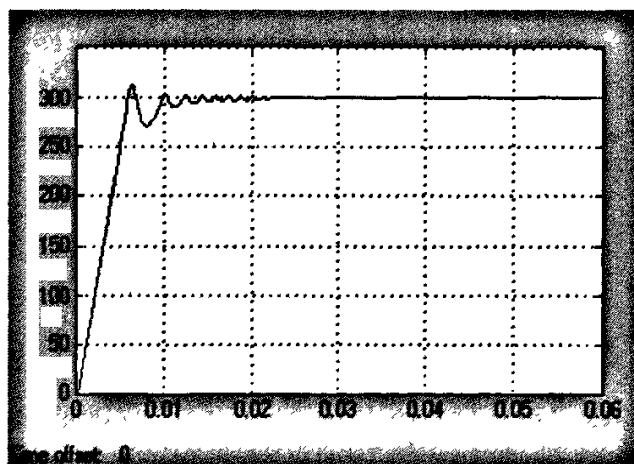


图 4.12 速度响应图

4.4.2 SPIDNN 控制网络仿真

在 4.3 节所建立的系统数学模型的基础上, 利用 SPIDNN 进行系统建模, 如图 4.13 所示。

取 SPIDNN 网络输入层至隐含层的比例元和微分元的连接权重初值为:

$$w_{11}=1, w_{21}=-1, w_{13}(0)=1, w_{23}(0)=-1$$

输入层至隐含层积分元的连接权重初值为:

$$w_{21}(0)=1, w_{22}(0)=-1$$

网络隐含层至输出层的连接权重初值按 PI 速度调节器值设定为:

$$w_1(0)=7.6, w_2=348, w_3=0$$

单位阶跃输入时, 系统速度响应及示波器 Scope 的输出如图 4.12 所示。从图中可以看出系统速度调节器为 PID 调节器时, 系统速度响应有超调, 且调节时间比较长。

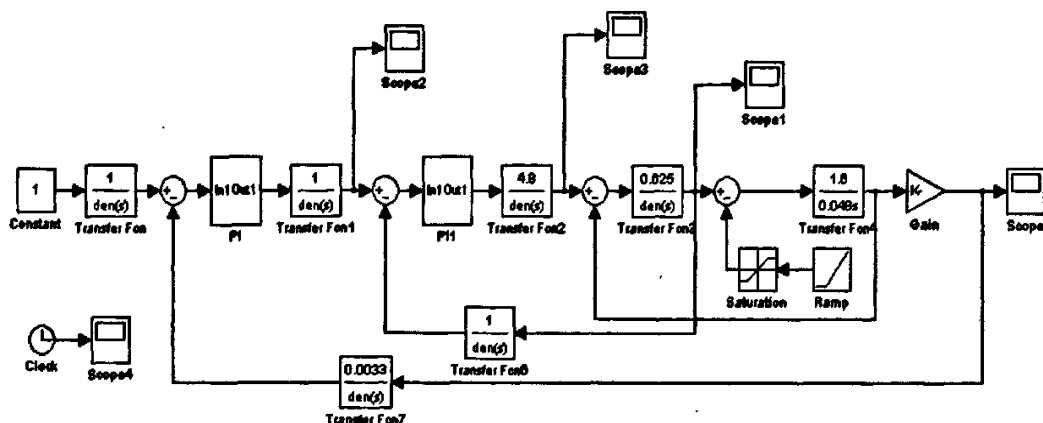


图 4.11 系统 PID 仿真图

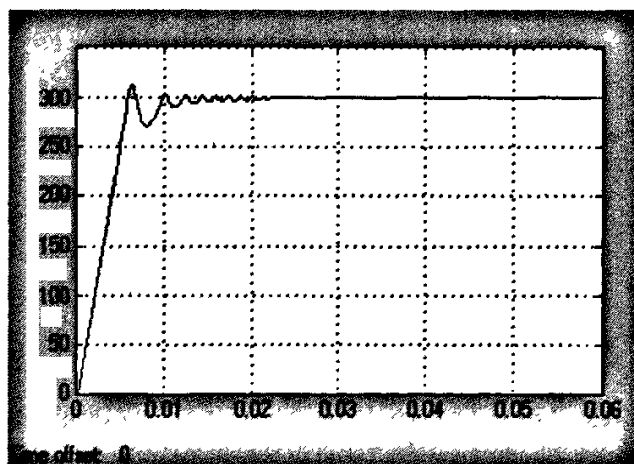


图 4.12 速度响应图

4.4.2 SPIDNN 控制网络仿真

在 4.3 节所建立的系统数学模型的基础上, 利用 SPIDNN 进行系统建模, 如图 4.13 所示。

取 SPIDNN 网络输入层至隐含层的比例元和微分元的连接权重初值为:

$$w_{11}=1, w_{21}=-1, w_{13}(0)=1, w_{23}(0)=-1$$

输入层至隐含层积分元的连接权重初值为:

$$w_{21}(0)=1, w_{22}(0)=-1$$

网络隐含层至输出层的连接权重初值按 PI 速度调节器值设定为:

$$w_1(0)=7.6, w_2=348, w_3=0$$

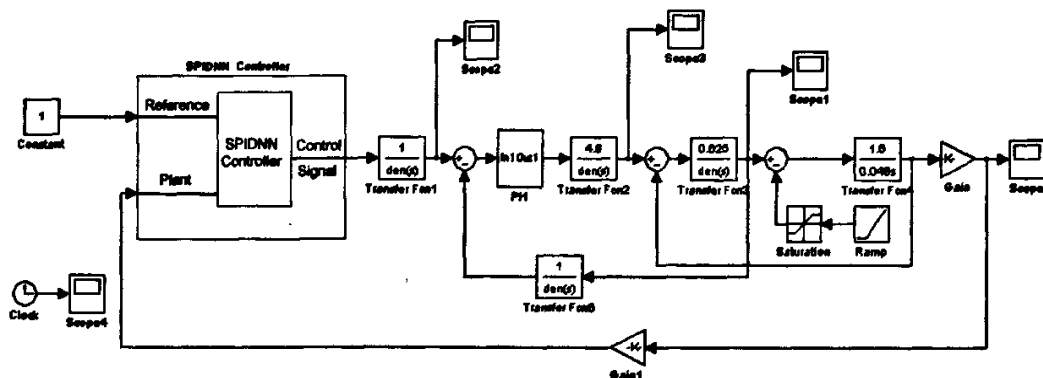


图 4.13 SPIDNN 仿真图

按输入为 $r(k)=1(k)$ ，学习步长 $\eta=0.2$ ，每步采样点数取 $m=200$ 进行训练，网络学习 50 步后的速度输出曲线如图 4.14 所示。从图中可以看出系统速度调节器为 SPIDNN 调节器时系统速度响应无超调，且上升时间比较短。

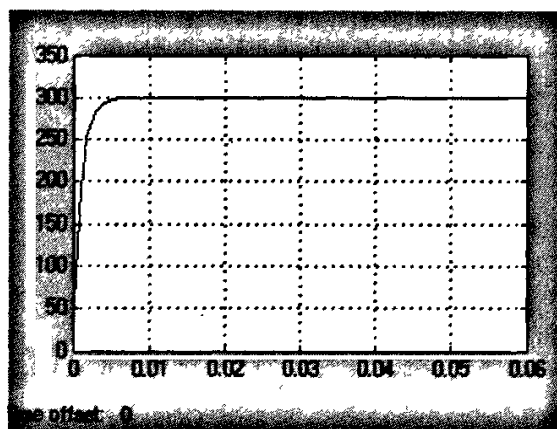


图 4.14 SPIDNN 网络速度响应图

4.5 系统仿真分析

(1) PID 仿真输出描述

PID 控制仿真输出如图 4.12 所示，从图中可以看出采用 PID 控制器时，在单位阶跃输入下，系统响应的超调量约为 3.3%，上升时间约为 0.006s，峰值时间约为 0.007s，调节时间约为 0.023s。

(2) SPIDNN 仿真输出描述

SPIDNN 仿真输出如图 4.14 所示，由图可以看出采用 SPIDNN 做控制器时，在单位输入下，系统输出没有超调，网络的调节时间约为 0.005s。

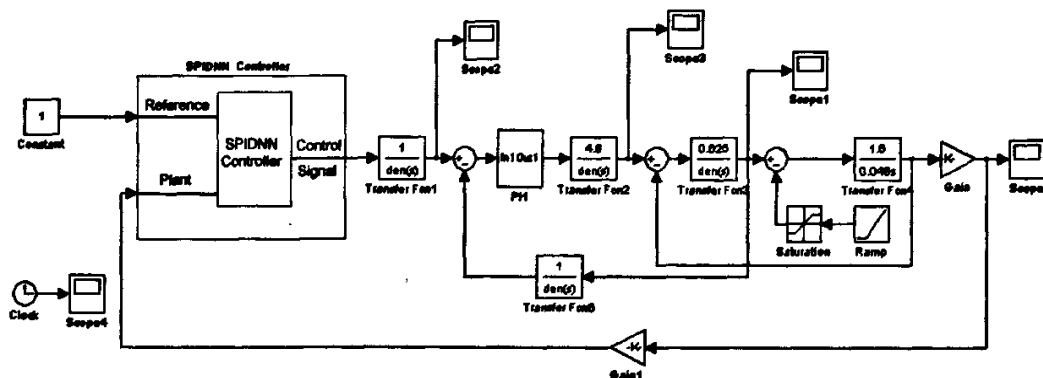


图 4.13 SPIDNN 仿真图

按输入为 $r(k)=1(k)$ ，学习步长 $\eta=0.2$ ，每步采样点数取 $m=200$ 进行训练，网络学习 50 步后的速度输出曲线如图 4.14 所示。从图中可以看出系统速度调节器为 SPIDNN 调节器时系统速度响应无超调，且上升时间比较短。

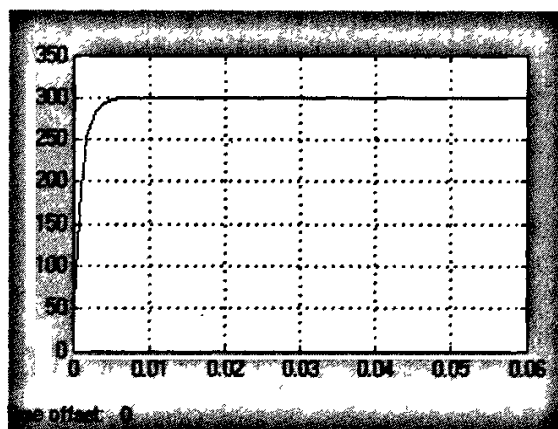


图 4.14 SPIDNN 网络速度响应图

4.5 系统仿真分析

(1) PID 仿真输出描述

PID 控制仿真输出如图 4.12 所示，从图中可以看出采用 PID 控制器时，在单位阶跃输入下，系统响应的超调量约为 3.3%，上升时间约为 0.006s，峰值时间约为 0.007s，调节时间约为 0.023s。

(2) SPIDNN 仿真输出描述

SPIDNN 仿真输出如图 4.14 所示，由图可以看出采用 SPIDNN 做控制器时，在单位输入下，系统输出没有超调，网络的调节时间约为 0.005s。

(3) 仿真结果分析

从(1)和(2)的描述看出,采用 PID 控制时系统有超调且调节时间长;采用 SPIDNN 控制器时系统无超调且系统调节时间较 PID 控制调节时间短。说明:

①本次试验建立 PID 调节器的方法与调节器的参数设定是正确的;

②采用 PIDNN 调节器时,通过在 PID 调节器基础上训练得到的参数匹配得当,用其构成的控制系统速度响应快,无超调,在斜坡干扰下抗干扰能力强;

③证明 SPIDNN 控制器具有优良的控制品质;

④将 PID 神经网络应用于单晶硅提拉速度控制系统,解决单晶硅质量问题,提高其成品率的构思是正确的。

4.6 本章小结

本章主要论述以下三方面的问题:

(1) 分析了单晶硅提拉的工作过程;

(2) 建立了提拉系统的数学模型;

(3) 建立了基于 SPIDNN 的单晶硅提拉速度控制系统 MATLAB/SIMULINK 模型;

(4) 进行了仿真与分析:从仿真结果可以看出 SPIDNN 控制系统具有优良的性能;分析认为这种性能是由于 SPIDNN 可以在经验 PID 参数基础上找到参数最佳匹配带来的,因此将其应用于单晶硅提拉速度控制系统具有可行性。

(3) 仿真结果分析

从(1)和(2)的描述看出,采用 PID 控制时系统有超调且调节时间长;采用 SPIDNN 控制器时系统无超调且系统调节时间较 PID 控制调节时间短。说明:

①本次试验建立 PID 调节器的方法与调节器的参数设定是正确的;

②采用 PIDNN 调节器时,通过在 PID 调节器基础上训练得到的参数匹配得当,用其构成的控制系统速度响应快,无超调,在斜坡干扰下抗干扰能力强;

③证明 SPIDNN 控制器具有优良的控制品质;

④将 PID 神经网络应用于单晶硅提拉速度控制系统,解决单晶硅质量问题,提高其成品率的构思是正确的。

4.6 本章小结

本章主要论述以下三方面的问题:

(1) 分析了单晶硅提拉的工作过程;

(2) 建立了提拉系统的数学模型;

(3) 建立了基于 SPIDNN 的单晶硅提拉速度控制系统 MATLAB/SIMULINK 模型;

(4) 进行了仿真与分析:从仿真结果可以看出 SPIDNN 控制系统具有优良的性能;分析认为这种性能是由于 SPIDNN 可以在经验 PID 参数基础上找到参数最佳匹配带来的,因此将其应用于单晶硅提拉速度控制系统具有可行性。

5 系统的软硬件设计

5.1 系统总体设计思想

系统以直流伺服电机(90sz55)为控制对象,以单晶硅提拉速度为被控量,采用速度、电流双闭环结构,以 PWM 驱动全控型功率转换器对电机电枢供电。速度调节器分别采用 PI、PID 神经网络控制方式。

系统硬件结构如图 5.1 所示。整个系统硬件由主电路、检测电路、保护电路和控制电路四部分组成。

在硬件电路的基础上,系统的控制要求和功能主要由软件实现,根据本系统的设计原则及控制功能要求,可将软件分成主程序、初始化子程序、中断服务程序、控制算法程序等几个模块^[6]。

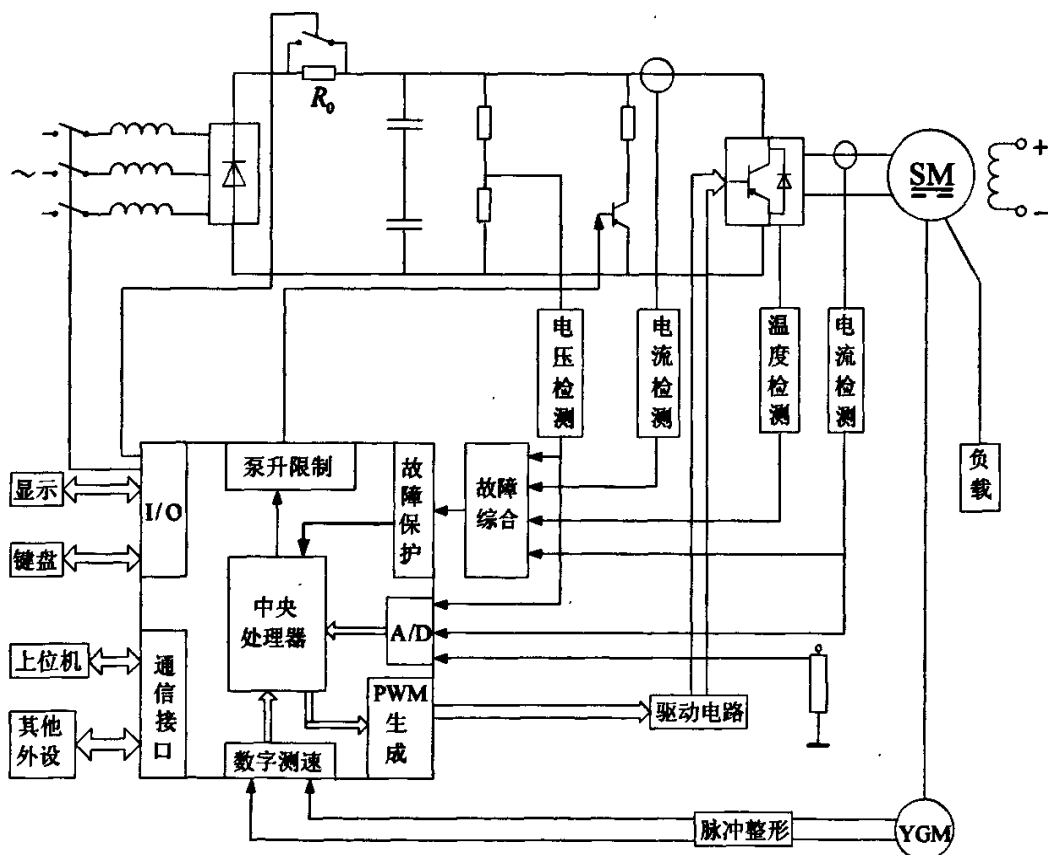


图 5.1 系统硬件结构图

5.2 主电路

系统主电路由电容滤波的三相不控整流电源、泵升限制电路、H 桥式功率转换电路及驱动电路等几部分组成。

5.2.1 整流电路

电容滤波的三相不控整流电路中,采用的是三相桥式结构,整流二极管 VD_1 、 VD_3 、 VD_5 接成共阴极, VD_4 、 VD_6 、 VD_2 接成共阳极,其组成如图 5.2 所示。

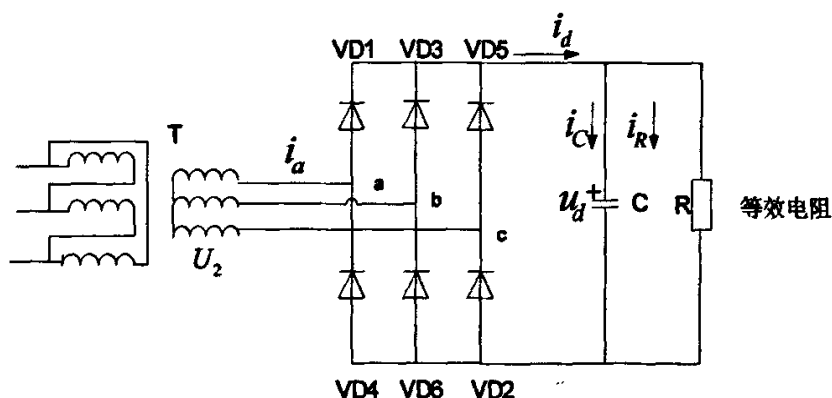


图 5.2 电容滤波三相桥式不控整流电路

(1) 基本原理

该电路中,当某一对二极管导通时,输出直流电压等于交流侧线电压中最大的一个,该线电压即向电容供电,也向负载供电。当没有二极管导通时,由电容向负载放电, u_d 按指数规律下降。

设二极管在距线电压过零点 δ 角处开始导通,并以二极管 VD_6 和 VD_1 开始同时导通的时刻为时间零点,则线电压为:

$$u_{ab} = \sqrt{6}U_2 \sin(\omega t + \delta) \quad (5.1)$$

而相电压为:

$$u_a = \sqrt{2}U_2 \sin\left(\omega t + \delta - \frac{\pi}{6}\right) \quad (5.2)$$

在 $\omega t = 0$ 时,二极管 VD_6 和 VD_1 开始导通,直流侧电压等于 u_{ab} ; 下一次同时导通的一对管子是 VD_1 和 VD_2 , 直流侧电压等于 u_{ac} 。这两段导通过程之间有两种交替情况,一种是在 VD_1 和 VD_2 同时导通之前 VD_6 和 VD_1 是关断的,交流侧向直流侧的充电电流 i_d 断续的,如图 5.3 所示;另一种是 VD_1 一直导通,交替时由 VD_6 导通换相至 VD_2 导通, i_d 是连续的。介于二者之间的临界情况是, VD_6 和 VD_1 同时导通的阶段与 VD_1 和 VD_2 同时导通的

5.2 主电路

系统主电路由电容滤波的三相不控整流电源、泵升限制电路、H 桥式功率转换电路及驱动电路等几部分组成。

5.2.1 整流电路

电容滤波的三相不控整流电路中,采用的是三相桥式结构,整流二极管 VD_1 、 VD_3 、 VD_5 接成共阴极, VD_4 、 VD_6 、 VD_2 接成共阳极,其组成如图 5.2 所示。

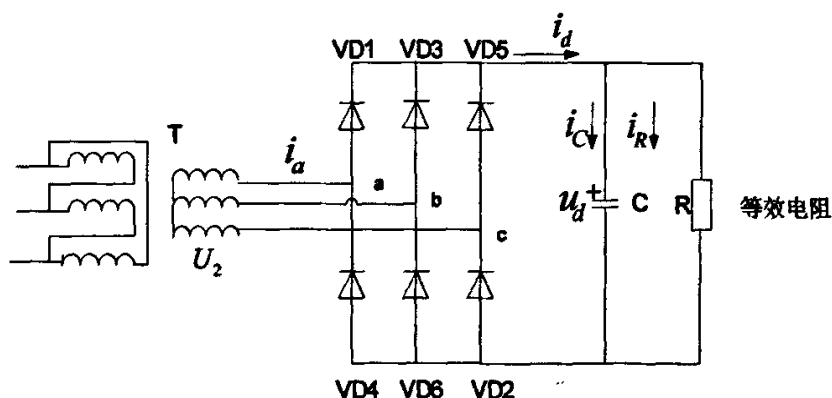


图 5.2 电容滤波三相桥式不控整流电路

(1) 基本原理

该电路中,当某一对二极管导通时,输出直流电压等于交流侧线电压中最大的一个,该线电压即向电容供电,也向负载供电。当没有二极管导通时,由电容向负载放电, u_d 按指数规律下降。

设二极管在距线电压过零点 δ 角处开始导通,并以二极管 VD_6 和 VD_1 开始同时导通的时刻为时间零点,则线电压为:

$$u_{ab} = \sqrt{6}U_2 \sin(\omega t + \delta) \quad (5.1)$$

而相电压为:

$$u_a = \sqrt{2}U_2 \sin\left(\omega t + \delta - \frac{\pi}{6}\right) \quad (5.2)$$

在 $\omega t = 0$ 时,二极管 VD_6 和 VD_1 开始导通,直流侧电压等于 u_{ab} ; 下一次同时导通的一对管子是 VD_1 和 VD_2 , 直流侧电压等于 u_{ac} 。这两段导通过程之间有两种交替情况,一种是在 VD_1 和 VD_2 同时导通之前 VD_6 和 VD_1 是关断的,交流侧向直流侧的充电电流 i_d 断续的,如图 5.3 所示;另一种是 VD_1 一直导通,交替时由 VD_6 导通换相至 VD_2 导通, i_d 是连续的。介于二者之间的临界情况是, VD_6 和 VD_1 同时导通的阶段与 VD_1 和 VD_2 同时导通的

阶段在 $\omega t + \delta = \frac{2\pi}{3}$ 处恰好衔接了起来。按照“下降电压相等”则有

$$\left| \frac{d[\sqrt{6}U_2 \sin(\omega t + \delta)]}{d(\omega t)} \right|_{\omega t + \delta = \frac{2\pi}{3}} = \left| \frac{d \left\{ \sqrt{6}U_2 \sin \frac{2\pi}{3} e^{\frac{1}{\omega RC} [\omega t - (\frac{2\pi}{3} - \delta)]} \right\}}{d(\omega t)} \right|_{\omega t + \delta = \frac{2\pi}{3}} \quad (5.3)$$

可得 $\omega RC = \sqrt{3}$ ，这就是临界条件。 $\omega RC > \sqrt{3}$ 和 $\omega RC \leq \sqrt{3}$ 分别是电流 i_d 断续和连续的条件。图 5.4 给出了 $\omega RC \leq \sqrt{3}$ 时的电流波形。对一个确定的装置来讲，通常只有 R 是可变的，它的大小反映了负载的轻重。因此可以说，在轻载时直流侧获得的充电电流是断续的，重载时是连续的，分界点就是 $R = \sqrt{3}/(\omega C)$ [9]。

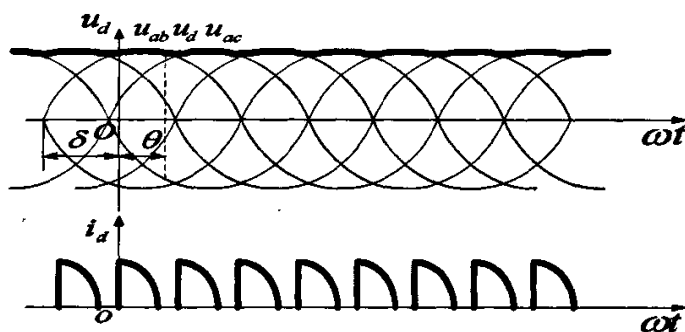


图 5.3 电容滤波三相桥式整流电路波形图

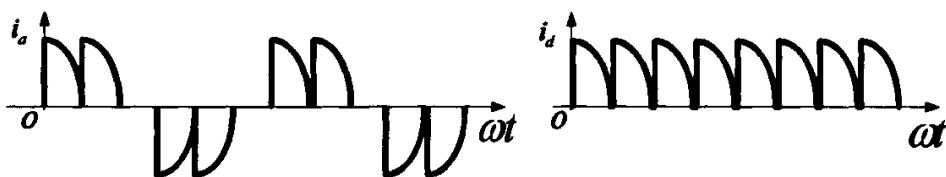


图 5.4 电容滤波的三相不控整流电路 $\omega RC \leq \sqrt{3}$

(2) 主要数量关系

① 输出电压平均值

空载时，输出电压平均值最大为 $U_d = \sqrt{6}U_2 = 2.45U_2$ 。随着负载的加重，输出电压

平均值减小至 $\omega RC = \sqrt{3}$ ，进入 i_d 连续情况后，输出电压波形为线电压包络线，其平均

值为 $U_d = 2.34U_2$ 。可见 U_d 在 $2.34U_2 \sim 2.45U_2$ 之间变化。

② 电流平均值

输出电流平均值 I_R 为：

$$I_R = U_d / R \quad (5.4)$$

因电容电流 i_c 平均值为零，从而有

$$I_d = I_R \quad (5.5)$$

在一个电源周期中， i_d 有 6 个波头，流过每一个二极管的是其中的两个波头，因此二极管的电流平均值为 I_d 的 $1/3$ ，即

$$I_{VD} = I_d / 3 = I_R / 3 \quad (5.6)$$

③ 二极管承受的电压

二极管承受的最大反向电压为线电压的峰值，为 $\sqrt{6}U_2$ [9]。

(3) 二极管选型

在整流二极管选型时，通常都要考虑电网电压的波动，且取其波动系数为 1.1，安全系数为 2，所以整流二极管的耐压值为：

$$U_{RRM} \geq \sqrt{6}U_2 \times 2.2 \quad (5.7)$$

5.2.2 桥式可逆 PWM 变换器

(1) PWM 变换器的工作原理

可逆 PWM 变换器主电路有多种形式，最常用的是桥式（亦称 H 形）电路，如图 5.5 所示。这时电动机 SM 两端电压 U_{AB} 的极性随开关器件驱动电压极性的变化而改变，其控制方式有单极式、双极式、受限单极式等多种，这里我们选用最常用双极式控制的可逆 PWM 变换器。

双极式可逆 PWM 变换器的 4 个驱动电压波形如图 5.6 所示，它们的关系是： $U_{g1} = U_{g4} = -U_{g2} = -U_{g3}$ 。在一个开关周期内，当 $0 \leq t < t_{on}$ 时， $U_{AB} = U_s$ ，电枢电流 i_d 沿回路 1 流通；当 $t_{on} \leq t < T$ 时驱动电压反向， i_d 沿回路 2 经二极管续流， $U_{AB} = -U_s$ 。因此， U_{AB} 在一个周期内具有正负相间脉冲波形，这是双极式名称的由来。

图 5.6 绘出双极式控制时的输出电压和电流波形。 i_{d1} 相当于一般负载的情况，脉动电流的方向始终为正； i_{d2} 相当于轻载情况，电流可在正负方向之间脉动，但平均值仍为正，等于负载电流。电动机的正反转则体现在驱动电压正、负脉冲的宽窄上。当正脉冲较宽时， $t_{on} > \frac{T}{2}$ ，则 U_{AB} 的平均值为正，电动机正转，反之则反转；如果正负脉冲相等，

值为 $U_d = 2.34U_2$ 。可见 U_d 在 $2.34U_2 \sim 2.45U_2$ 之间变化。

② 电流平均值

输出电流平均值 I_R 为：

$$I_R = U_d / R \quad (5.4)$$

因电容电流 i_c 平均值为零，从而有

$$I_d = I_R \quad (5.5)$$

在一个电源周期中， i_d 有 6 个波头，流过每一个二极管的是其中的两个波头，因此二极管的电流平均值为 I_d 的 $1/3$ ，即

$$I_{VD} = I_d / 3 = I_R / 3 \quad (5.6)$$

③ 二极管承受的电压

二极管承受的最大反向电压为线电压的峰值，为 $\sqrt{6}U_2$ [9]。

(3) 二极管选型

在整流二极管选型时，通常都要考虑电网电压的波动，且取其波动系数为 1.1，安全系数为 2，所以整流二极管的耐压值为：

$$U_{RRM} \geq \sqrt{6}U_2 \times 2.2 \quad (5.7)$$

5.2.2 桥式可逆 PWM 变换器

(1) PWM 变换器的工作原理

可逆 PWM 变换器主电路有多种形式，最常用的是桥式（亦称 H 形）电路，如图 5.5 所示。这时电动机 SM 两端电压 U_{AB} 的极性随开关器件驱动电压极性的变化而改变，其控制方式有单极式、双极式、受限单极式等多种，这里我们选用最常用双极式控制的可逆 PWM 变换器。

双极式可逆 PWM 变换器的 4 个驱动电压波形如图 5.6 所示，它们的关系是： $U_{g1} = U_{g4} = -U_{g2} = -U_{g3}$ 。在一个开关周期内，当 $0 \leq t < t_{on}$ 时， $U_{AB} = U_s$ ，电枢电流 i_d 沿回路 1 流通；当 $t_{on} \leq t < T$ 时驱动电压反向， i_d 沿回路 2 经二极管续流， $U_{AB} = -U_s$ 。因此， U_{AB} 在一个周期内具有正负相间脉冲波形，这是双极式名称的由来。

图 5.6 绘出双极式控制时的输出电压和电流波形。 i_{d1} 相当于一般负载的情况，脉动电流的方向始终为正； i_{d2} 相当于轻载情况，电流可在正负方向之间脉动，但平均值仍为正，等于负载电流。电动机的正反转则体现在驱动电压正、负脉冲的宽窄上。当正脉冲较宽时， $t_{on} > \frac{T}{2}$ ，则 U_{AB} 的平均值为正，电动机正转，反之则反转；如果正负脉冲相等，

$t_{on} = \frac{T}{2}$, 平均输出电压为零, 则电动机停止。图 5.6 所示的波形是电动机正转时的情况。

双极性控制可逆 PWM 变换器的输出平均电压为 $U_d = \frac{t_{on}}{T} U_s - \frac{T-t_{on}}{T} U_s = \left(\frac{2t_{on}}{T} - 1 \right) U_s$, 若占空比 $\rho = \frac{t_{on}}{T}$ 和电压系数 $\gamma = \frac{U_d}{U_s}$, 则在双极性控制的可逆变换器中 $\gamma = 2\rho - 1$ 。

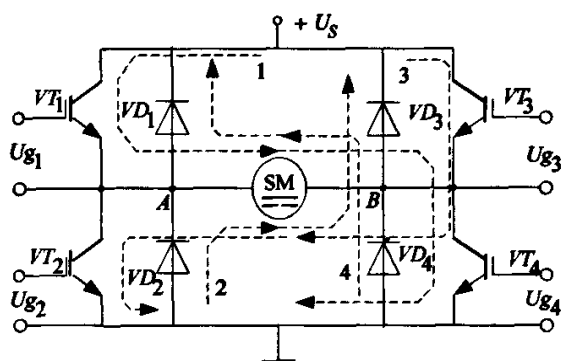


图 5.5 桥式可逆 PWM 变换器

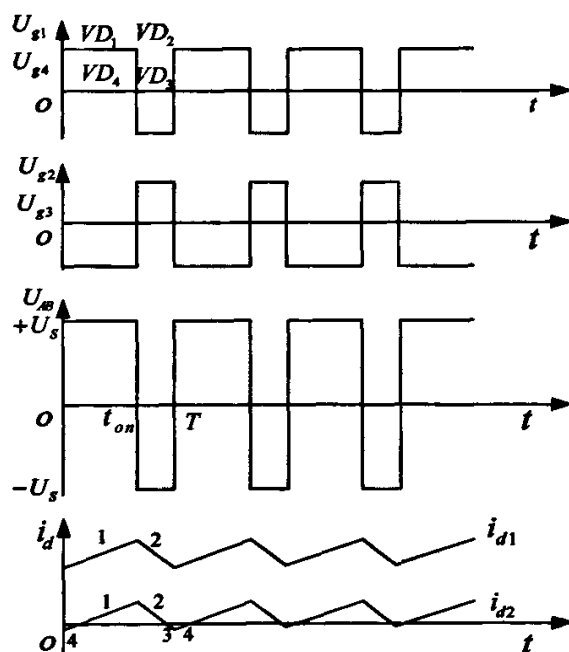


图 5.6 双极性 PWM 变换器波形

调速时, ρ 的可调范围为 $0 \sim 1$, 相应地, $\gamma = -1 \sim +1$ 。当 $\rho > \frac{1}{2}$ 时, γ 为正, 电动机正转; 当 $\rho < \frac{1}{2}$ 时, γ 为负, 电动机反转; 当 $\rho = \frac{1}{2}$ 时, $\gamma = 0$, 电动机停止。但电

动机停止时电枢电压并不等于零,而是正负脉宽相等的交变脉冲电压,因而电流也是交变的。这个交变电流的平均值为零,不产生平均转矩,徒然增大电动机的损耗,这是双极式控制的缺点。但其好处是,在电动机停止时仍有高频微振电流,从而消除了正、反向时的静摩擦死区,起着所谓动力润滑的作用。

双极式控制方式的不足之处是:在工作过程中,4个开关器件可能都处于开关状态,开关损耗大,而且在切换时可能发生上下桥臂直通的事故,为了防止直通,在上、下桥臂的驱动脉冲之间应设置逻辑延时^[11]。

(2) 微处理机与双极性 PWM 功率转换器的匹配

① 处理周期 T_p

为了用微处理机系统产生 PWM 波形,首先要确定处理周期 T_p ,即经过多少时间就要对系统实况采样、处理,以求得新的脉宽调制信号,它由系统处理要求和速度决定。也就是说,在系统现有的处理速度下,在处理周期内完成系统规定的处理要求,并求得新的控制信息量。 T_p 小,控制精度高。但 T_p 取得过小,不但对系统提出较高的速度要求,还有可能引起系统不稳定。它的最小值还受系统控制要求的限制。设系统要求电机最高转速为 $n_{M\cdot\max}$,最低转速为 $n_{M\cdot\min}$,分辨率为 δ ,则最大控制信号比 K 为:

$$K = \frac{n_{M\cdot\max}}{n_{M\cdot\min}\delta} = \frac{D}{\delta} \quad (5.8)$$

式中 D 为系统调速比。

在双极模式 PWM 控制中的脉宽可调范围为 $T_p/2$,如果最小调制时间(当量)为 a_i ,则其可控制信号比也就确定为:

$$Q = T_p/2a_i \quad (5.9)$$

为保证系统控制要求,必须使 $Q \geq K$,则可得

$$T_{p\cdot\min} \geq 2a_i D / \delta \quad (5.10)$$

a_i 受微处理机系统硬件物理特性的限制,一般 $a_i \geq 1\mu s$ 。因此对一般工业调速系统, $D=10\sim 100$ 左右,这里取 $D=10$, $\delta < 0.001$,则可得

$$T_{p\cdot\min} = \frac{2 \times 1 \times 10}{0.001} = 20ms \quad (5.11)$$

恰当地选择 T_p 是微处理机控制系统总体设计中的重要一环。

② 执行周期 T_k

执行周期是指直接加到电机电枢上的控制脉冲周期 T_k (开关周期),它的最大值受电机工作平稳性和开关特性的限制。 T_k 越小,电流脉动越小。但 T_k 过小又会增加系统输出处理时间。实际取值时,为便于数字处理总是满足:

$$T_p = 2^i \times T_k \quad (5.12)$$

式中, i 为大于零的正整数。如果 $T_p = 25ms$, i 取 5, 倍频系数 $2^i = 32$, 则 $T_k = 0.78ms$, 作用到电枢两端的脉冲电压频率约为 $1.5kHz$ 。

③ 余量补偿问题

由于 T_k 与 T_p 不同, 根据处理周期 T_p 求得的脉宽调制量必须除以倍频系数 2^i 以后再分配给每个执行周期来完成。但是这个脉宽调制的当量数 (1us 的倍数) 不一定被 32 整除。为提高控制精度必须进行余数补偿, 即将不满 32 的余数一次补偿到每个执行周期中去。每次补偿一个当量, 补完后, 后面的执行周期就无需补偿了^[14]。

(3) IGBT 选型

IGBT 选型时, 通常按电动机额定电压的 2~3 倍确定耐压 U_{CEP} , 按电动机额定电流的 1.7~2 倍选择其 I_C 值。

5.2.3 驱动电路

本系统以 MICREL 公司的 MIC5014/MIC5015 芯片作为桥式可逆 PWM 功率转换器的栅极驱动电路芯片, 由其构成的驱动电路如图 5.7 所示。

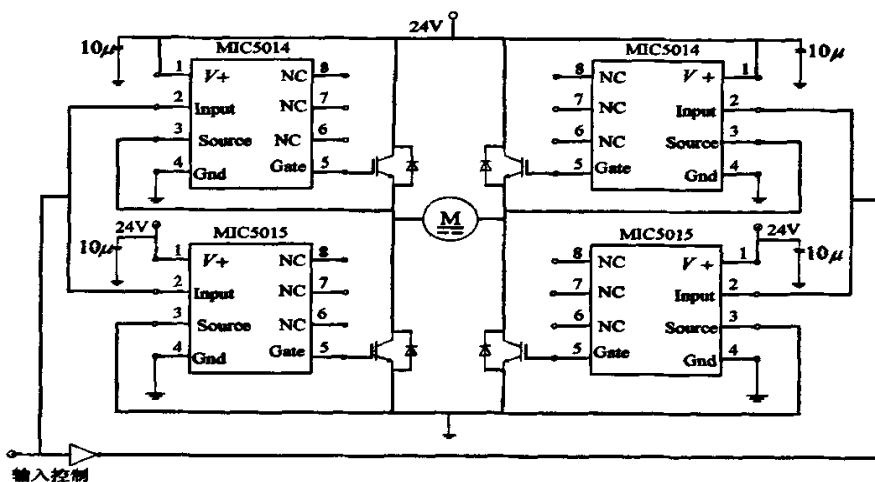


图 5.7 驱动电路

MIC5014/MIC5015 是 MICREL 半导体公司生产的高侧/低侧功率器件栅极驱动电路。其工作电压在 2.75~30V 之间, 高侧 MIC5014 可以驱动负载电压为 30 伏的功率器件, 低侧 MIC5015 的最大可开关电压决定于功率器件本身的耐压。

其带有一个适应 TTL 电平的控制输入端, MIC5014 的控制输入端为高电平有效, MIC5015 输入端为低电平有效。其内含充电泵, 能产生 15V 以内的栅极驱动电压。

由于器件的关闭时间比开通时间快, 所以不存在上下桥臂直通现象。

式中, i 为大于零的正整数。如果 $T_p = 25ms$, i 取 5, 倍频系数 $2^i = 32$, 则 $T_k = 0.78ms$, 作用到电枢两端的脉冲电压频率约为 $1.5kHz$ 。

③ 余量补偿问题

由于 T_k 与 T_p 不同, 根据处理周期 T_p 求得的脉宽调制量必须除以倍频系数 2^i 以后再分配给每个执行周期来完成。但是这个脉宽调制的当量数 (1us 的倍数) 不一定被 32 整除。为提高控制精度必须进行余数补偿, 即将不满 32 的余数一次补偿到每个执行周期中去。每次补偿一个当量, 补完后, 后面的执行周期就无需补偿了^[14]。

(3) IGBT 选型

IGBT 选型时, 通常按电动机额定电压的 2~3 倍确定耐压 U_{CEP} , 按电动机额定电流的 1.7~2 倍选择其 I_C 值。

5.2.3 驱动电路

本系统以 MICREL 公司的 MIC5014/MIC5015 芯片作为桥式可逆 PWM 功率转换器的栅极驱动电路芯片, 由其构成的驱动电路如图 5.7 所示。

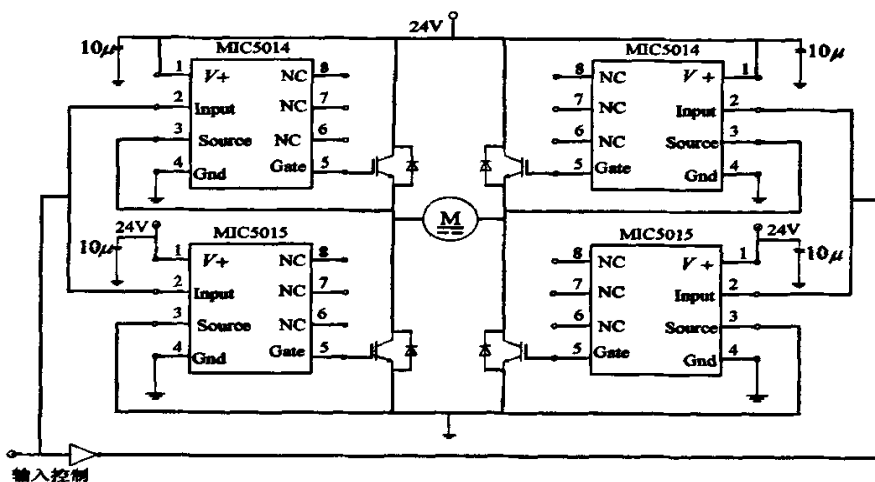


图 5.7 驱动电路

MIC5014/MIC5015 是 MICREL 半导体公司生产的高侧/低侧功率器件栅极驱动电路。其工作电压在 2.75~30V 之间, 高侧 MIC5014 可以驱动负载电压为 30 伏的功率器件, 低侧 MIC5015 的最大可开关电压决定于功率器件本身的耐压。

其带有一个适应 TTL 电平的控制输入端, MIC5014 的控制输入端为高电平有效, MIC5015 输入端为低电平有效。其内含充电泵, 能产生 15V 以内的栅极驱动电压。

由于器件的关闭时间比开通时间快, 所以不存在上下桥臂直通现象。

其中 1 脚 V+ 为电源口, 2 脚 Input 为输入口, 3 脚 Source 接功率 IGBT 的发射极, 4 脚 Ground 接地, 5 脚接 IGBT 的栅极, 6、7、8 脚 NC 为空。

5.2.4 泵升限制

PWM 变换器的直流电源由交流电网变压后经不可控的二极管整流产生, 并采用大电容滤波, 以获得恒定的直流电压 U_i 。由于电容容量较大, 突加电源时相当于短路, 势必产生比较大的充电电流, 容易损坏整流二极管。为了限制充电电流, 在整流器与滤波电容之间加入限流电阻 R_0 (或电抗), 合上电源以后, 延时用开关将电阻短路, 以免在运行中造成附加损耗。

滤波电容器往往在 PWM 装置的体积和重量中占有不小的份额, 因此电容器容量选择是 PWM 装置设计中的重要问题。PWM 变换器中的滤波电容, 其作用除滤波以外还有当电机制动时吸收系统动能的作用。由于直流电源靠二极管整流供电, 不可能回馈电能, 电机制动时只好对滤波电容充电, 这将使电容两端电压升高, 称作“泵升电压”。假设电压由 U_i 升高到 U_m , 则电容储能由 $\frac{1}{2}CU_i^2$ 升高到 $\frac{1}{2}CU_m^2$, 储能的增量基本上等于运动系统在制动时释放的全部动能 A_d , 于是

$$\frac{1}{2}CU_m^2 - \frac{1}{2}CU_i^2 = A_d \quad (5.13)$$

按制动储能要求选择的电容量应为

$$C = \frac{2A_d}{U_m^2 - U_i^2} \quad (5.14)$$

电力电子器件的耐压限制着最高泵升电压 U_m , 因此电容量就不可能很小, 一般几千瓦的调速系统所需的电容量达数千微法。在大容量或负载有较大惯量的系统中, 不可能只靠电容器来限制泵升电压, 这时, 可以采用图中镇流电阻 R_0 来消耗掉部分动能。 R_0 的分流电路靠开关器件 VT₀ 在泵升电压达到允许数值时接通。

对于更大容量的系统, 为了提高效率, 可以在二极管整流器输出端并接逆变器, 把多余的能量逆变后回馈电网^[9]。

本系统选取的电容为 2000μF。

5.3 检测回路

5.3.1 电流检测与软件

(1) 电流传感器

PWM 电路要求对电流做快速检测, 电流检测器件有良好的频响特性和极小的延迟时

其中 1 脚 V+ 为电源口, 2 脚 Input 为输入口, 3 脚 Source 接功率 IGBT 的发射极, 4 脚 Ground 接地, 5 脚接 IGBT 的栅极, 6、7、8 脚 NC 为空。

5.2.4 泵升限制

PWM 变换器的直流电源由交流电网变压后经不可控的二极管整流产生, 并采用大电容滤波, 以获得恒定的直流电压 U_i 。由于电容容量较大, 突加电源时相当于短路, 势必产生比较大的充电电流, 容易损坏整流二极管。为了限制充电电流, 在整流器与滤波电容之间加入限流电阻 R_0 (或电抗), 合上电源以后, 延时用开关将电阻短路, 以免在运行中造成附加损耗。

滤波电容器往往在 PWM 装置的体积和重量中占有不小的份额, 因此电容器容量选择是 PWM 装置设计中的重要问题。PWM 变换器中的滤波电容, 其作用除滤波以外还有当电机制动时吸收系统动能的作用。由于直流电源靠二极管整流供电, 不可能回馈电能, 电机制动时只好对滤波电容充电, 这将使电容两端电压升高, 称作“泵升电压”。假设电压由 U_i 升高到 U_m , 则电容储能由 $\frac{1}{2}CU_i^2$ 升高到 $\frac{1}{2}CU_m^2$, 储能的增量基本上等于运动系统在制动时释放的全部动能 A_d , 于是

$$\frac{1}{2}CU_m^2 - \frac{1}{2}CU_i^2 = A_d \quad (5.13)$$

按制动储能要求选择的电容量应为

$$C = \frac{2A_d}{U_m^2 - U_i^2} \quad (5.14)$$

电力电子器件的耐压限制着最高泵升电压 U_m , 因此电容量就不可能很小, 一般几千瓦的调速系统所需的电容量达数千微法。在大容量或负载有较大惯量的系统中, 不可能只靠电容器来限制泵升电压, 这时, 可以采用图中镇流电阻 R_0 来消耗掉部分动能。 R_0 的分流电路靠开关器件 VT₀ 在泵升电压达到允许数值时接通。

对于更大容量的系统, 为了提高效率, 可以在二极管整流器输出端并接逆变器, 把多余的能量逆变后回馈电网^[9]。

本系统选取的电容为 2000μF。

5.3 检测回路

5.3.1 电流检测与软件

(1) 电流传感器

PWM 电路要求对电流做快速检测, 电流检测器件有良好的频响特性和极小的延迟时

其中 1 脚 V+ 为电源口, 2 脚 Input 为输入口, 3 脚 Source 接功率 IGBT 的发射极, 4 脚 Ground 接地, 5 脚接 IGBT 的栅极, 6、7、8 脚 NC 为空。

5.2.4 泵升限制

PWM 变换器的直流电源由交流电网变压后经不可控的二极管整流产生, 并采用大电容滤波, 以获得恒定的直流电压 U_i 。由于电容容量较大, 突加电源时相当于短路, 势必产生比较大的充电电流, 容易损坏整流二极管。为了限制充电电流, 在整流器与滤波电容之间加入限流电阻 R_0 (或电抗), 合上电源以后, 延时用开关将电阻短路, 以免在运行中造成附加损耗。

滤波电容器往往在 PWM 装置的体积和重量中占有不小的份额, 因此电容器容量选择是 PWM 装置设计中的重要问题。PWM 变换器中的滤波电容, 其作用除滤波以外还有当电机制动时吸收系统动能的作用。由于直流电源靠二极管整流供电, 不可能回馈电能, 电机制动时只好对滤波电容充电, 这将使电容两端电压升高, 称作“泵升电压”。假设电压由 U_i 升高到 U_m , 则电容储能由 $\frac{1}{2}CU_i^2$ 升高到 $\frac{1}{2}CU_m^2$, 储能的增量基本上等于运动系统在制动时释放的全部动能 A_d , 于是

$$\frac{1}{2}CU_m^2 - \frac{1}{2}CU_i^2 = A_d \quad (5.13)$$

按制动储能要求选择的电容量应为

$$C = \frac{2A_d}{U_m^2 - U_i^2} \quad (5.14)$$

电力电子器件的耐压限制着最高泵升电压 U_m , 因此电容量就不可能很小, 一般几千瓦的调速系统所需的电容量达数千微法。在大容量或负载有较大惯量的系统中, 不可能只靠电容器来限制泵升电压, 这时, 可以采用图中镇流电阻 R_0 来消耗掉部分动能。 R_0 的分流电路靠开关器件 VT₀ 在泵升电压达到允许数值时接通。

对于更大容量的系统, 为了提高效率, 可以在二极管整流器输出端并接逆变器, 把多余的能量逆变后回馈电网^[9]。

本系统选取的电容为 2000μF。

5.3 检测回路

5.3.1 电流检测与软件

(1) 电流传感器

PWM 电路要求对电流做快速检测, 电流检测器件有良好的频响特性和极小的延迟时

间,所以系统选择 LEM 公司的电流传感器,其响应时间在 $1\mu\text{s}$ 之内,且使用“磁补偿原理”使霍尔器件始终处于检测零磁通的状态,其原理图如图 5.8 所示。

其测量关系为:

$$|I_p \cdot N_p| = |I_s \cdot N_s| \quad (5.15)$$

本文所选的型号是 LA50-P, 参数如下:

匝数比: 1: 1000;

测量范围: $0 \sim 70\text{A}$;

电源电压: $\pm 15\text{V}$;

根据系统检测量与给定关系可得:

$$|I_s| \cdot R_M = \frac{|I_p \cdot N_p|}{N_s} \cdot R_M = 5\text{V} \quad (5.16)$$

所以

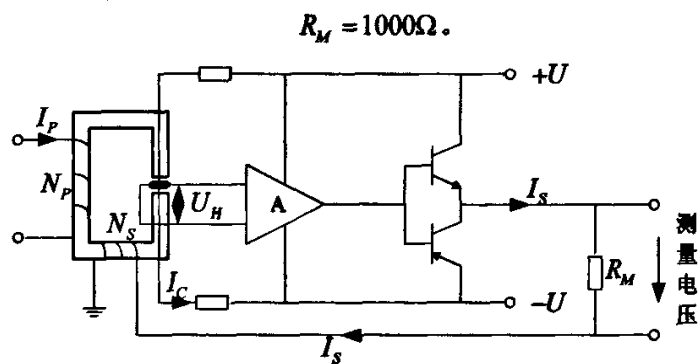


图 5.8 LEM 电流传感器原理图

(2) 电流调节中断服务子程序

电流调节中断服务子程序流程图如图 5.9 所示,其主要完成电流 PI 调节和 PWM 生成功能,然后启动 A/D 转换,为下一步调节准备。

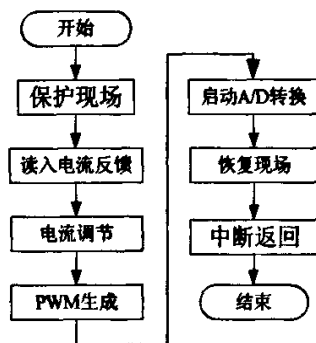


图 5.9 电流调节中断服务子程序

5.3.2 数字测速与软件

测速元件是速度闭环控制系统中的关键元件。通常为了扩大调速范围，改善低速平稳性，要求测速元件低速输出稳定，纹波小，线性度好。对于模拟测速元件，通常采用直流测速发电机。对于数字式测速元件，为便于控制和提高测速性能，通常采用光电式脉冲编码器（亦称增量式编码器）作为速度反馈检测元件。

(1) 光电式脉冲编码器原理

光电式脉冲编码器是测转速或转角的检测元件，其原理如图 5.10 所示。编码器与电动机轴相连，当电动机转动时，带动码盘旋转，便发出转速或转角信号。编码器在码盘上均匀的刻制一定数量的光栅，当电动机旋转时，码盘随之一起转动。通过光栅的作用，持续不断的开放或封闭光通路，因此，在接收装置的输出端便得到频率与转速成正比的方波脉冲序列，从而可以计算转速。

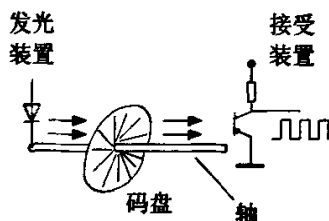


图 5.10 光电式脉冲编码器示意图

上述脉冲序列可以正确的反映转速的高低，但不能鉴别转向。为了获得转速的方向，可增加一对发光与接受装置，使两对接受与发光装置错开光栅节距的 $1/4$ ，则两组脉冲序列 A 和 B 的相位相差 90° ，如图 5.11 所示。正转时 A 相超前 B 相；反转时 B 相超前 A 相。采用简单的辨相电路就可以分辨出转向。

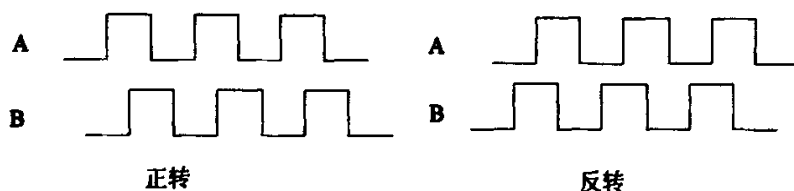


图 5.11 区分旋转方向的 A、B 两组脉冲序列

若编码盘的光栅数为 N ，则转速分辨率为 $1/N$ ，常用的编码器光栅数有 1024、2048、4096 等。再增加光栅数将大大增加旋转器的制作难度和成本。采用倍频电路可以提高转速分辨率，而不增加编码器的光栅数，一般多采用四倍频电路，大于四倍频则较难实现。

(2) 常用测速方法

通常编码器的数字测速方法有三种：M 法、T 法和 M/T 法。

① M 法测速

在一定的时间 T_c 内测取编码器的输出脉冲个数 M_1 ，用以计算这段时间内的平均转速，称作 M 法测速，如图 5.12 所示。把 M_1 除以 T_c 就可以得到旋转编码器输出脉冲的频率 $f_1 = M_1 / T_c$ ，所以又称频率法。电动机每转一圈共产生 Z 个脉冲 ($Z = \text{倍频系数} \times \text{编码器光栅数}$)，把 f_1 除以 Z 就得到电机转速。在习惯上，时间 T_c 以秒为单位，而转速以每分钟的转数 r/min 为单位，则电动机的转速为

$$n = \frac{60M_1}{ZT_c} \quad (5.17)$$

在上式中， Z 跟 T_c 均为常数值，因此转速 n 正比于脉冲个数 M_1 。高速时 M_1 大，量化误差较小，随着转速的降低误差增大，转速过低时 M_1 将小于 1，测速装置便不能正常工作。

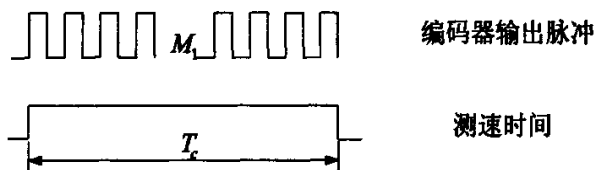


图 5.12 M 法测速

②T 法测速

在编码器两个相邻输出脉冲间隔内，用一个计数器对已知频率为 f_0 的高频时钟脉冲进行计数，并由此来计算转速，称作 T 法测速，如图 5.13 所示。在这里，测速时间缘于编码器输出脉冲的周期，所以又称周期法。在测速中，准确的测速时间 T_i 是用所得的高频时钟脉冲个数计算的得到的，即 $T_i = M_2 / f_0$ ，则电动机转速为：

$$n = \frac{60f_0}{ZM_2} \quad (5.18)$$

高速时 M_2 小，量化误差大，随着转速的降低误差减小，所以 T 法测速使用于低速段，与 M 测速恰好相反。

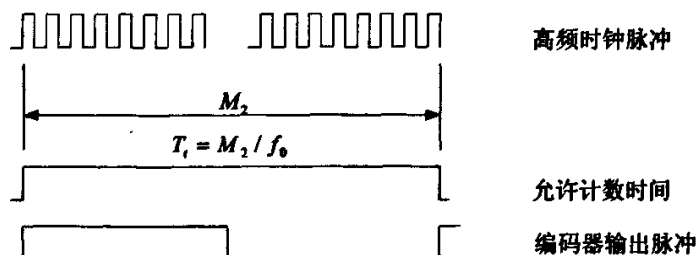


图 5.13 T 法测速

③M/T 法测速

把 M 法和 T 法结合起来，既检测 T_c 时间内旋转编码器输出的脉冲个数 M_1 ，又检测同一时间内高频时钟脉冲的个数 M_2 ，用来计算转速，称作 M/T 法测速，如图 5.14 所示。

设高频时钟脉冲的频率为 f_0 ，则准确的测速时间 $T_i = M_2 / f_0$ ，而电动机的转速为：

$$n = \frac{60M_1f_0}{ZM_2} \quad (5.19)$$

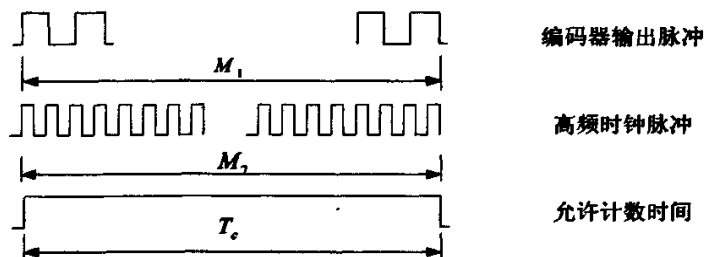


图 5.14 M/T 测速

采用 M/T 法测速时，应保证高频时钟脉冲计数器与编码器同时开启与同时关闭，以减小误差，只有等到编码器输出脉冲前沿到达时，两个计数器才同时允许开始或停止计数。

本系统采用 M/T 测速法。

(3) 整形辨相电路

为了实现可逆控制，用编码器的波形实现可逆速度检测时，必须将 A、B 两相方波波形进行整形和辨相处理，产生出反映电动机转向的脉冲信号。整形辨相电路如图 5.15 所示，它是采用逻辑门组合电路的思路进行设计的，即将编码器输出的两路信号，分别输入到一个单稳态触发器和一个非门，然后再将信号经过一次与逻辑，得到反映电动机正转的信号 ($CP+$) 和反转的信号 ($CP-$)。

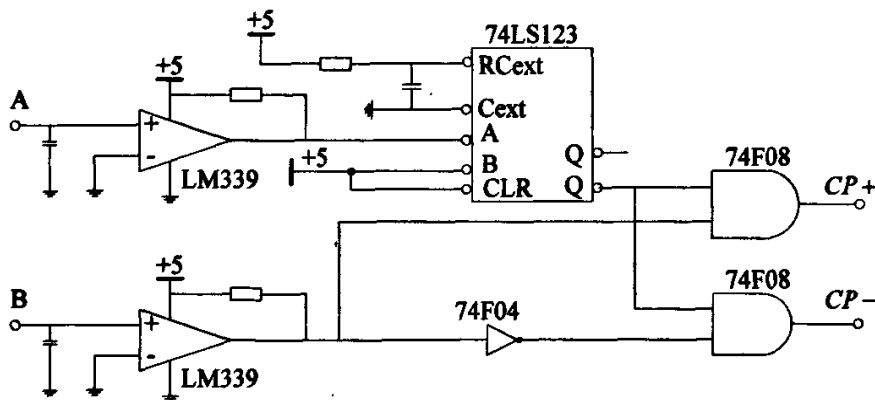


图 5.15 整形辨相电路

用逻辑表达式表示即为：

$$\begin{aligned} CP+ &= \bar{A} \cdot B \\ CP- &= \bar{A} \cdot \bar{B} \end{aligned} \quad (5.20)$$

其中 \bar{A} 为 A 信号的单稳态触发脉冲。整形辨相电路的波形如图 5.16 所示。由图可知，

当 A 超前 B 到 $T/4$ 周期时, $CP+$ 输出信号, 而 $CP-$ 没有信号输出, 如图 5.16(a) 所示; 相反, 就只有 $CP-$ 有信号输出, 从而实现了电动机正反转的识别, 如图 5.16(b) 所示。

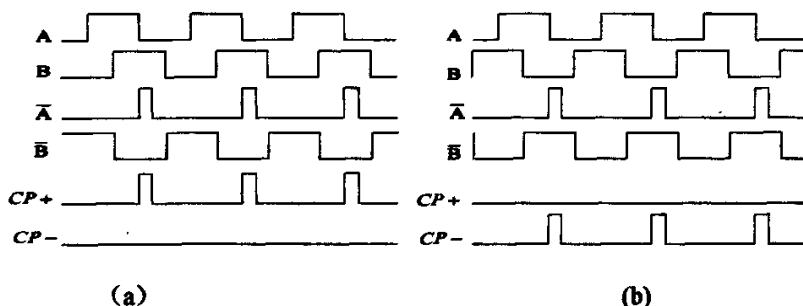


图 5.16 整形辨相电路波形

(4) 测速及调速软件

①测速子程序

测速子程序由捕捉中断服务子程序和测速时间中断服务子程序构成, 其流程如图 5.17 和 5.18 所示。转速调节中断服务子程序 (见图 5.19) 运行到“允许测速”时, 开放捕捉中断, 但只有当编码器脉冲前沿到达时, 进入捕捉中断服务子程序, 编码器脉冲计数器 M_1 和高频脉冲计数器 M_2 才真正开始计数, 同时打开测速时间计数器 T_c , 禁止捕捉中断, 使之不再干扰计数器计数。待测速时间计数器到达计数值, 发出停止测速信号, 再次开放捕捉中断, 到编码器脉冲前沿到达时停止计数。在此程序中, 测速软件仅完成 M_1 和 M_2 计数, 转速计算是在转速调节中断服务子程序中完成的。

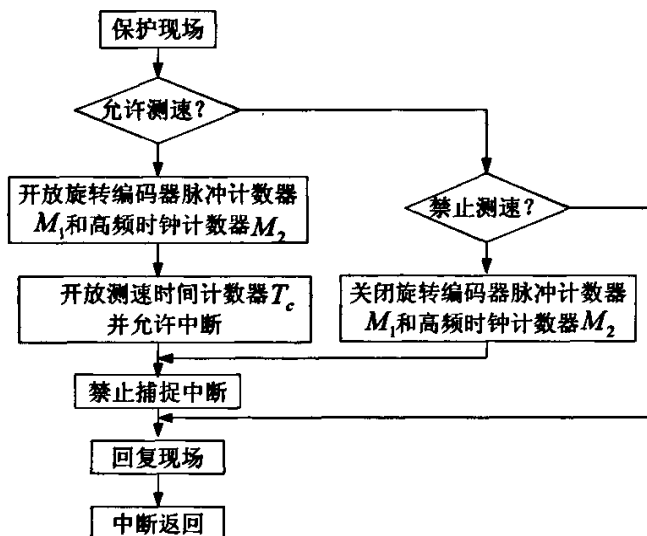


图 5.17 捕捉中断服务子程序

②转速调节中断服务子程序

转速调节中断服务子程序主要实现实际转速计算, 转速 PI 调节, 并启动转速检测的

功能，如图 5.19 所示。

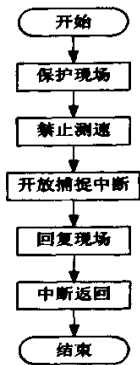


图 5.18 测速时间中断服务子程序

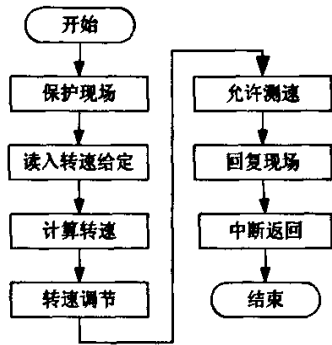


图 5.19 转速调节中断服务子程序

进入转速调节中断服务子程序后，首先完成现场保护，然后读取转速给定和转速反馈，计算转速并完成转速 PI 调节，最后启动转速检测，为下一步调节作准备。

5.3.3 故障检测

计算机控制系统的一大优点是，除了能实现控制以外，还能实现故障的自诊断。任何一个自动控制系统难免会出现各种故障，产生故障的原因可能来自外部，也可能来自系统内部。因此，必须采用得当的保护措施，减少或避免故障发生，如果故障已经发生，还应避免故障进一步扩大，使损失降到最低程度。

数字控制系统除了控制手段灵活、可靠性高等优点以外，在故障检测、保护、自诊断方面也有着模拟系统无法比拟的优势。

(1) 供电电源故障检测

三相供电电源常见的故障为过电压、欠电压和缺相，电源故障检测电路如图 5.20 所示。

由三相供电电源经电压互感器、三相不可控整流桥和滤波环节，得到与供电电压幅值成正比的直流电压，经 A/D 转换输入至控制器。数字控制系统定时采样该直流电压，并与上下限值进行比较，即可判断是否出现过电压或欠电压故障。

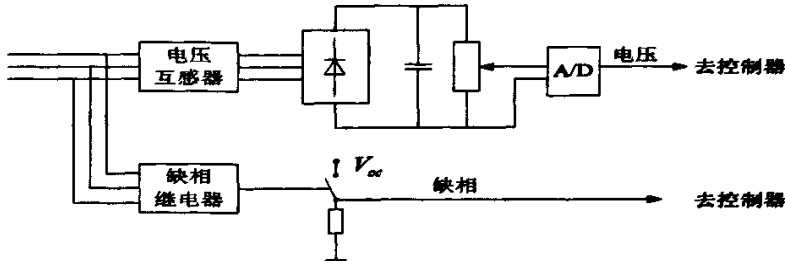


图 5.20 供电电源故障检测电路

缺相检测采用缺相继电器，正常时，继电器常开触点打开，缺相输出信号为低电平；

功能，如图 5.19 所示。

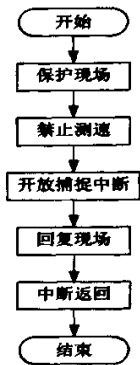


图 5.18 测速时间中断服务子程序

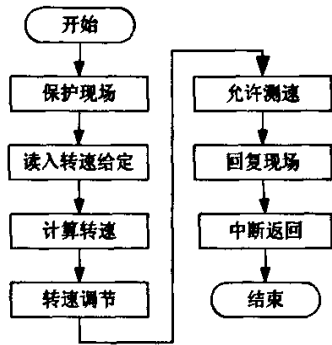


图 5.19 转速调节中断服务子程序

进入转速调节中断服务子程序后，首先完成现场保护，然后读取转速给定和转速反馈，计算转速并完成转速 PI 调节，最后启动转速检测，为下一步调节作准备。

5.3.3 故障检测

计算机控制系统的一大优点是，除了能实现控制以外，还能实现故障的自诊断。任何一个自动控制系统难免会出现各种故障，产生故障的原因可能来自外部，也可能来自系统内部。因此，必须采用得当的保护措施，减少或避免故障发生，如果故障已经发生，还应避免故障进一步扩大，使损失降到最低程度。

数字控制系统除了控制手段灵活、可靠性高等优点以外，在故障检测、保护、自诊断方面也有着模拟系统无法比拟的优势。

(1) 供电电源故障检测

三相供电电源常见的故障为过电压、欠电压和缺相，电源故障检测电路如图 5.20 所示。

由三相供电电源经电压互感器、三相不可控整流桥和滤波环节，得到与供电电压幅值成正比的直流电压，经 A/D 转换输入至控制器。数字控制系统定时采样该直流电压，并与上下限值进行比较，即可判断是否出现过电压或欠电压故障。

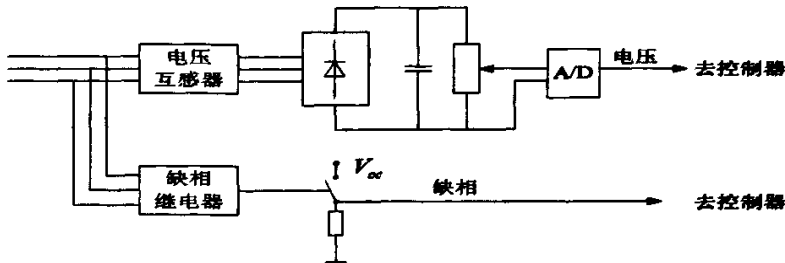


图 5.20 供电电源故障检测电路

缺相检测采用缺相继电器，正常时，继电器常开触点打开，缺相输出信号为低电平；

缺相时，继电器常开触点闭合，缺相输出信号呈高电平。定时检测缺相检测输出电路电平，即可判断是否出现缺相故障。

(2) 过电流、过载故障检测

过电流、过载是电力拖动控制系统较易出现的故障，检测功率变换器的输入、输出电流，可判断是否出现故障及故障类型，如图 5.21 所示。

当 I_{in} 或 I_{out} 大于或等于设定的故障电流值 I_{GZ} 时，认为出现过电流故障；当 I_{in} 或 I_{out} 大于过载电流值 I_{GL} ，且持续时间 $t > t_{on}$ 时，认为出现过载故障。如果仅需检测过电流或过载故障，只要检测 I_{in} 或 I_{out} 即可。同时检测 I_{in} 和 I_{out} ，还可以判别故障点。若 $I_{in} \geq I_{GZ}$ ，而 $I_{out} \approx 0$ ，则表明故障出现在功率变换装置内部；若 $I_{in} \geq I_{GZ}$ ， $I_{out} \geq I_{GZ}$ ，则表明故障出现在负载回路。

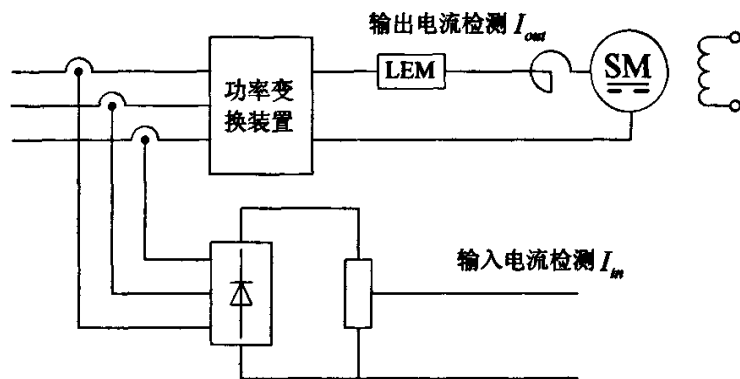


图 5.21 电流故障检测

5.4 控制部分

5.4.1 C8051F404 控制器

C8051F 系列单片机是 51 家族的一种单片机，其处理器为 CIP51 处理器，其在指令集上与 MCS-51 完全兼容；与传统的 MCS-51 系列单片机不同的是，C8051F 系列单片机是完全集成的混合信号片上系统型 MCU，其集成度更高，运行速度更快。C8051F040 外围结构如图 5.22 所示。

与 MCS-51 相比，C8051F040 有以下特点：

(1) 低电压供电（2.7~3.6V），输入端口兼容 5V 电平，输出有开漏和推挽两种模式。在开漏模式下，逻辑 1 时，输出为高阻态，可通过上拉电阻接至 5V，使逻辑 1 状态下，输出为 5V 电平，可与 5V 供电芯片相容；

(2) 集成 JTAG 调试器，可在线调试和下载；

(3) 多复位源，有看门狗复位、电源电平检测复位、时钟失步复位及比较器复位等，

缺相时，继电器常开触点闭合，缺相输出信号呈高电平。定时检测缺相检测输出电路电平，即可判断是否出现缺相故障。

(2) 过电流、过载故障检测

过电流、过载是电力拖动控制系统较易出现的故障，检测功率变换器的输入、输出电流，可判断是否出现故障及故障类型，如图 5.21 所示。

当 I_{in} 或 I_{out} 大于或等于设定的故障电流值 I_{GZ} 时，认为出现过电流故障；当 I_{in} 或 I_{out} 大于过载电流值 I_{GL} ，且持续时间 $t > t_{on}$ 时，认为出现过载故障。如果仅需检测过电流或过载故障，只要检测 I_{in} 或 I_{out} 即可。同时检测 I_{in} 和 I_{out} ，还可以判别故障点。若 $I_{in} \geq I_{GZ}$ ，而 $I_{out} \approx 0$ ，则表明故障出现在功率变换装置内部；若 $I_{in} \geq I_{GZ}$ ， $I_{out} \geq I_{GZ}$ ，则表明故障出现在负载回路。

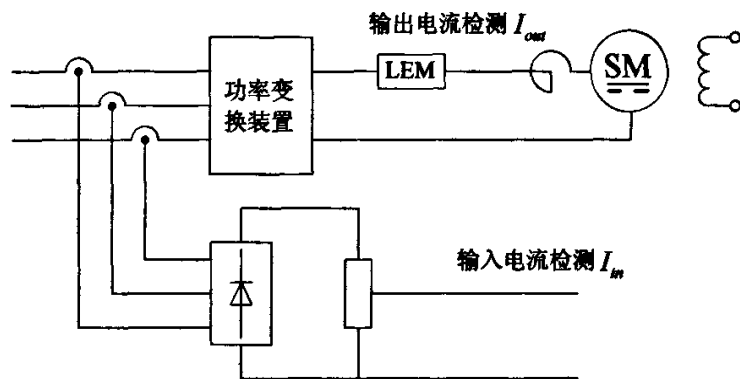


图 5.21 电流故障检测

5.4 控制部分

5.4.1 C8051F404 控制器

C8051F 系列单片机是 51 家族的一种单片机，其处理器为 CIP51 处理器，其在指令集上与 MCS-51 完全兼容；与传统的 MCS-51 系列单片机不同的是，C8051F 系列单片机是完全集成的混合信号片上系统型 MCU，其集成度更高，运行速度更快。C8051F040 外围结构如图 5.22 所示。

与 MCS-51 相比，C8051F040 有以下特点：

(1) 低电压供电（2.7~3.6V），输入端口兼容 5V 电平，输出有开漏和推挽两种模式。在开漏模式下，逻辑 1 时，输出为高阻态，可通过上拉电阻接至 5V，使逻辑 1 状态下，输出为 5V 电平，可与 5V 供电芯片相容；

(2) 集成 JTAG 调试器，可在线调试和下载；

(3) 多复位源，有看门狗复位、电源电平检测复位、时钟失步复位及比较器复位等，

缺相时，继电器常开触点闭合，缺相输出信号呈高电平。定时检测缺相检测输出电路电平，即可判断是否出现缺相故障。

(2) 过电流、过载故障检测

过电流、过载是电力拖动控制系统较易出现的故障，检测功率变换器的输入、输出电流，可判断是否出现故障及故障类型，如图 5.21 所示。

当 I_{in} 或 I_{out} 大于或等于设定的故障电流值 I_{GZ} 时，认为出现过电流故障；当 I_{in} 或 I_{out} 大于过载电流值 I_{GL} ，且持续时间 $t > t_{on}$ 时，认为出现过载故障。如果仅需检测过电流或过载故障，只要检测 I_{in} 或 I_{out} 即可。同时检测 I_{in} 和 I_{out} ，还可以判别故障点。若 $I_{in} \geq I_{GZ}$ ，而 $I_{out} \approx 0$ ，则表明故障出现在功率变换装置内部；若 $I_{in} \geq I_{GZ}$ ， $I_{out} \geq I_{GZ}$ ，则表明故障出现在负载回路。

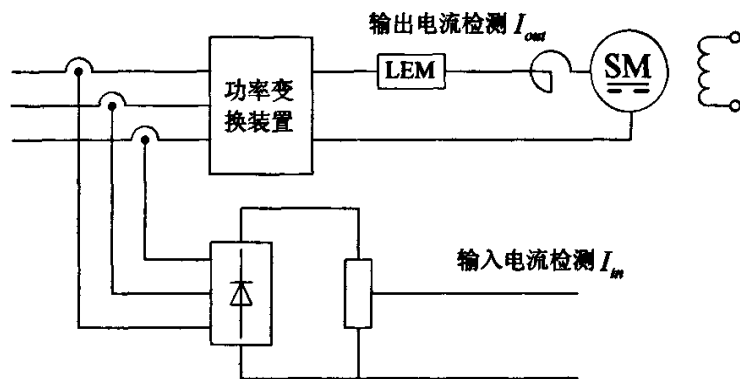


图 5.21 电流故障检测

5.4 控制部分

5.4.1 C8051F404 控制器

C8051F系列单片机是51家族的一种单片机，其处理器为CIP51处理器，其在指令集上与MCS-51完全兼容；与传统的MCS-51系列单片机不同的是，C8051F系列单片机是完全集成的混合信号片上系统型MCU，其集成度更高，运行速度更快。C8051F040外围结构如图5.22所示。

与MCS-51相比，C8051F040有以下特点：

(1) 低电压供电（2.7~3.6V），输入端口兼容5V电平，输出有开漏和推挽两种模式。在开漏模式下，逻辑1时，输出为高阻态，可通过上拉电阻接至5V，使逻辑1状态下，输出为5V电平，可与5V供电芯片相容；

(2) 集成JTAG调试器，可在线调试和下载；

(3) 多复位源，有看门狗复位、电源电平检测复位、时钟失步复位及比较器复位等，

(4) 处理器最高运行速度可达25MHz, 可用内部时钟, 也可用外部时钟。外部时钟有多种接法, 既可接入高精度晶振高速稳定运行, 也可接入RC振荡。低频运行时可降低能耗, 且可动态改变系统时钟;

(5) 2个12位DAC模块, DAC模块为电压型, 与DAC0832(电流型)不同, 输出无需再接运放将电流转换为电压;

(6) 1个12位ADC模块和一个8位ADC模块, 12位ADC模块带有一个HVDA通道, 可接入最大差压为 ± 60 伏的测量模拟信号, 并有可编程增益运放;

(7) 一个芯片温度传感器, 可检测芯片温度;

(8) 3个史密斯比较模块;

(9) 控制器局域网(CAN2.0B)控制器, 具有32个消息对象, 每个消息对象有其自己的标识;

(10) 5个16位通用定时器;

(11) 6个PAC模块, 可实现方波输出, PWM输出和事件捕捉等;

(12) 带有1个SPI串行外围设备接口控制器;

(13) 1个SMBUS总线控制器, 与 I^2C 总线完全相容;

(14) 1个带硬件地址控制的串行总线UART0和1个普通串行总线UART1;

(15) 两个外部时间中断, 中断源可达20个;

(16) 64KB Flash ROM, 256字节片内内存和4KB外部内存;

(17) 引入交叉开关配置, 可灵活将外围设备配置到P0~P3口, 如图5.23所示^[10]。

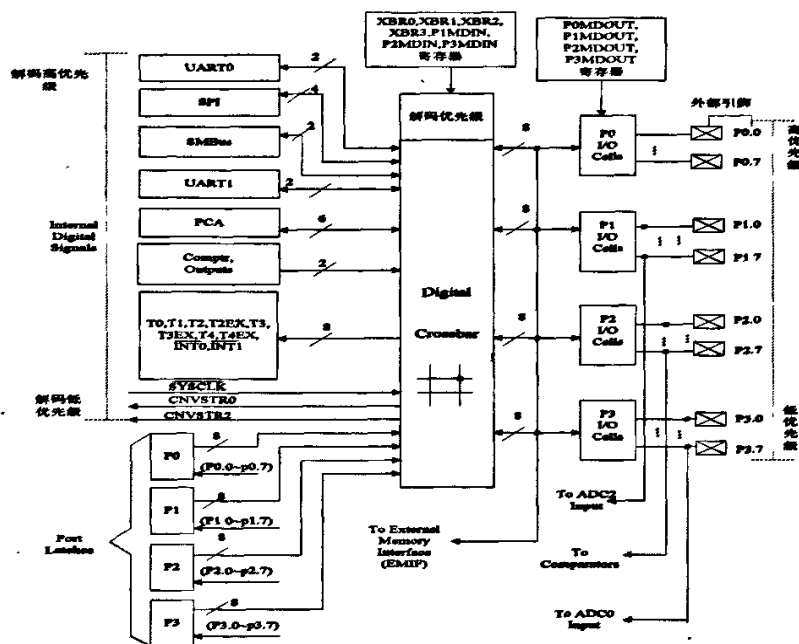


图5.23 交叉开关配置图

5.4.2 控制板功能模块设计

(1) 供电电源

控制板电源是经变压整流得到 9 伏直流电源，其电压转换过程如图 5.24 所示，7805 是输出 3.3V 和 5V 电压的调节器，AS1117 是输出 3.3V 电压的直流电压调解器，其中 5V 电压为 C8051F040 芯片与其他 5V 器件提供接口电源。

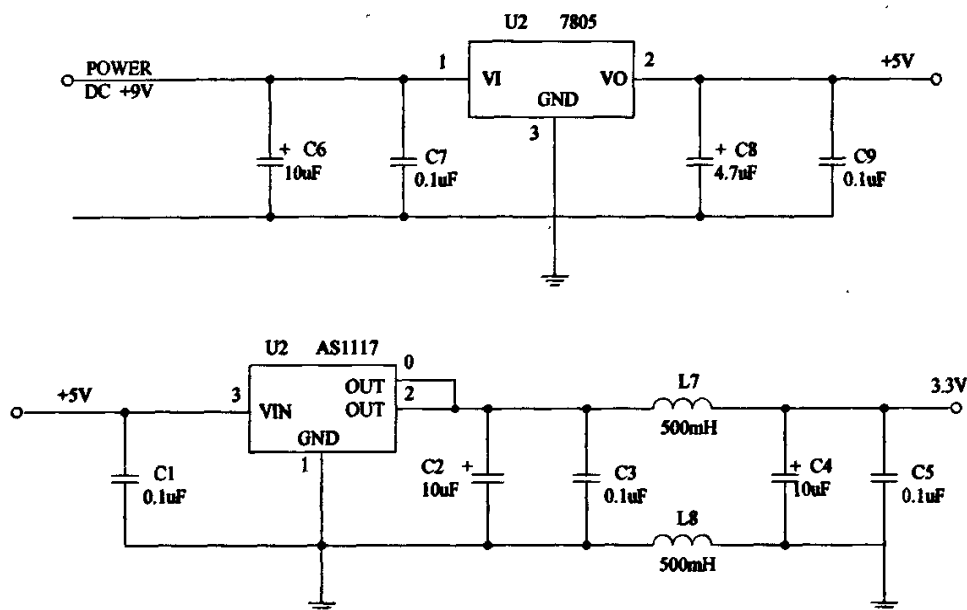


图 5.24 控制板电源电路

(2) 晶振驱动电路

控制板晶振驱动电路由石英晶体、电容 (20pF)、电阻组成，在布线时为了减少干扰，应尽量使晶振和电容靠近 XTAL1、XTAL2 端，如图 5.25 所示。

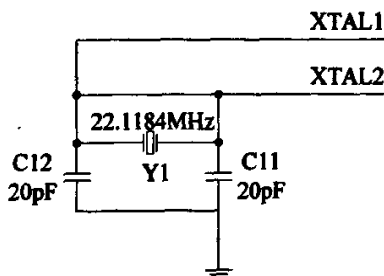


图 5.25 外部晶振驱动电路

(3) 复位电路

复位电路用于在系统发生故障时，对系统作强制外部保护，当按键 S_1 按下时，RST

引脚为低电平，系统被强制复位，图 5.26 所示。

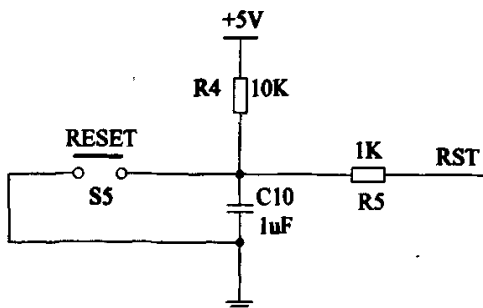


图 5.26 复位电路

(4) 串行通信接口电路

串行口通信接口电路用于单片机 UART0 与微机串行口的通信，以实现上位机与单片机的数据交换，主要组成部分是 MAX3232 接口芯片，如图 5.27 所示。

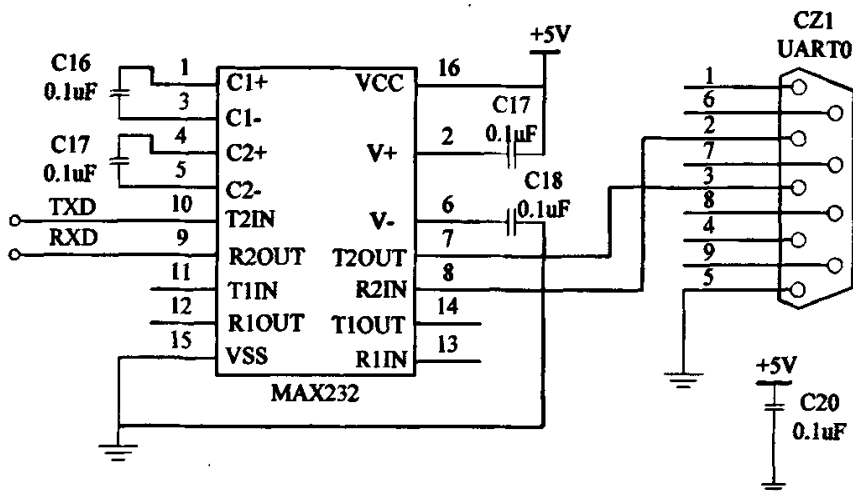


图 5.27 串行口通信接口电路

5.4.3 主程序与初始化子程序

系统的控制规律是靠软件实现的，硬件也是由软件来实施管理的。系统的软件主要有主程序、初始化子程序和中断服务子程序等。

(1) 主程序

主程序完成实时性要求不高的功能，完成系统初始化后，实现键盘处理、刷新显示、与上位机和其他外设通信等功能，如图 5.28 所示。

(2) 初始化子程序

初始化子程序实现硬件器件工作方式的设定、系统运行参数和变量的初始化等功能，初始化子程序框图如图 5.29 所示。

引脚为低电平，系统被强制复位，图 5.26 所示。

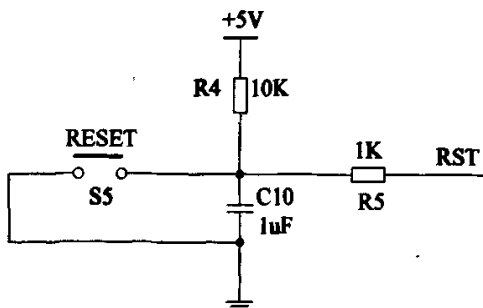


图 5.26 复位电路

(4) 串行通信接口电路

串行口通信接口电路用于单片机 UART0 与微机串行口的通信，以实现上位机与单片机的数据交换，主要组成部分是 MAX3232 接口芯片，如图 5.27 所示。

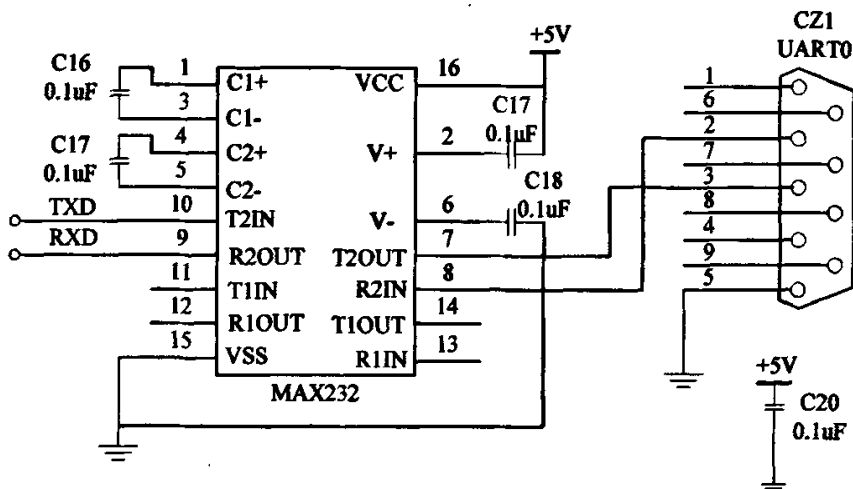


图 5.27 串行口通信接口电路

5.4.3 主程序与初始化子程序

系统的控制规律是靠软件实现的，硬件也是由软件来实施管理的。系统的软件主要有主程序、初始化子程序和中断服务子程序等。

(1) 主程序

主程序完成实时性要求不高的功能，完成系统初始化后，实现键盘处理、刷新显示、与上位机和其他外设通信等功能，如图 5.28 所示。

(2) 初始化子程序

初始化子程序实现硬件器件工作方式的设定、系统运行参数和变量的初始化等功能，初始化子程序框图如图 5.29 所示。

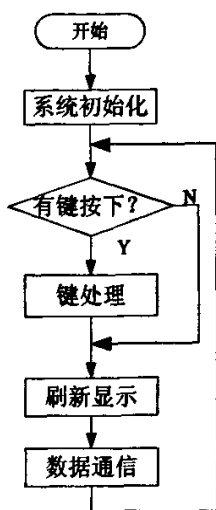
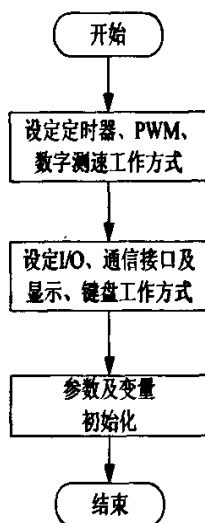


图 5.28 主程序框图



5.29 初始化子程序框图

5.5 本章小结

本章的主要内容是：

- (1) 设计试验系统的硬件整体结构；
- (2) 设计硬件系统的功能模块：控制板电源电路、外部晶振驱动电路、复位电路、串行口通信电路、供电电源故障检测电路、电流故障检测电路、测速脉冲整形辨相电路、电流检测电路；
- (3) 设计系统软件：主程序、初始化子程序、测速时间中断服务子程序、转速调节中断服务子程序、捕捉中断服务子程序、电流调节中断服务子程序。

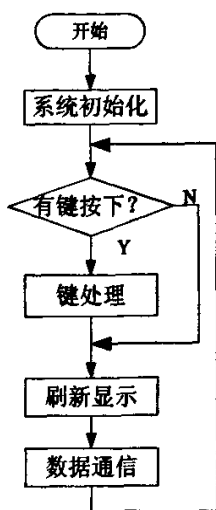
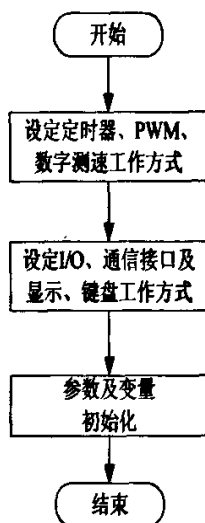


图 5.28 主程序框图



5.29 初始化子程序框图

5.5 本章小结

本章的主要内容是：

(1) 设计试验系统的硬件整体结构；

(2) 设计硬件系统的功能模块：控制板电源电路、外部晶振驱动电路、复位电路、串行口通信电路、供电电源故障检测电路、电流故障检测电路、测速脉冲整形辨相电路、电流检测电路；

(3) 设计系统软件：主程序、初始化子程序、测速时间中断服务子程序、转速调节中断服务子程序、捕捉中断服务子程序、电流调节中断服务子程序。

6 结论

(1) 本文总结

单晶硅的提拉是一个涉及多学科综合的问题,影响单晶硅质量的因素也是多方面的,本课题的研究是在假设其他各种因素均保持正常的条件下进行的,本文主要进行了下面几项研究:

1) 分析了晶体结晶过程的热量运输以及晶体缺陷的形成机理,找出了单晶硅提拉过程中影响其质量问题的主要因素;

2) 将 PID 神经网络应用于单晶硅提拉速度控制系统并建立了 PID 神经网络提拉系统数学模型;

3) 建立了基于 MATLAB/SIMULINK 的 PID 神经网络控制系统模型,通过仿真证明了 PID 神经网络应用于单晶硅提拉速度控制系统的可行性;

4) 设计了 PID 神经网络控制系统的硬件电路和部分软件。

通过以上的研究,取得如下结论:

本次试验建立 PID 调节器的方法与调节器的参数设定是正确的;采用 PID 神经网络调节器时,通过在 PID 调节器基础上训练得到的参数匹配得当,用其构成的控制系统响应速度快,无超调,抗干扰能力强;PID 神经网络控制器具有优良的控制品质,将其应用于提拉单晶硅质量问题的方案具有可行性;将 PID 神经网络应用于单晶硅提拉速度控制系统,解决单晶硅质量问题,提高其成品率的构思是正确的。

(2) 展望

单晶硅提拉是一个包括提拉速度控制系统、引线旋转系统、坩锅旋转上升系统、加热系统的综合系统,各系统之间相互联系紧密,并且互有影响。实际生产中影响单晶硅质量的因素是多方面的,对速度的控制是对多种影响因素的综合。本次试验虽然取得了一定成果,但因为条件所限没能将多输出 PID 神经网络应用于单晶硅提拉。所以 PID 神经网络应用于单晶硅生产还有大量工作可做。如能够真正将 PID 神经网络应用于单晶硅提拉系统,不仅会使提拉单晶硅从质量上上一个台阶,而且从直径范围上提拉单晶硅会再上一个新的台阶。

致 谢

本文是在我的导师郭秀才副教授的悉心指导和语重心长的教育下完成的。在西安科技大学攻读硕士学位三年的学习中，我有幸师从于郭老师，使我在离开大学 10 多年后又学到了新的知识、新的思维、受益匪浅。郭老师不仅启发我怎么去学知识，而且也潜移默化地告诉了我如何去工作。郭老师严谨的治学态度、广博的学识、敏锐的洞察力以及忘我的工作精神使我终身难忘。郭老师身上，有太多的品质让我去学习，不仅现在是，将来也还是。在此，向郭老师致以最诚挚的敬意和谢意！

同时，非常感谢电气与控制工程学院其他老师的帮助和指导，也向他们表示诚挚的敬意和谢意！

同时非常感谢广州大学舒怀林教授的支持与帮助！

也向宁夏九零五厂的相关技术人员表示深深的谢意！

上学期间，还受到不少同学的支持和帮助，也向他们表示深深的谢意！

谢 谢

参考文献

- [1] 杨树人,王宗昌等.半导体材料[M].科学出版社.2004:35~36,54~56
- [2] 陈力俊.微电子材料与工程[M].上海:复旦大学出版社,2005
- [3] 杨仲平,郑均忠,程斌.自动控制系统[M].北京:煤炭工业出版社,1992
- [4] 孙 亮,杨 鹏.自动控制原理[M].北京:北京工业大学出版社,2001
- [5] 舒怀林.PID 神经元及其控制系统[M].北京:国防工业出版社.2006:6~8,52~58
- [6] 吴 坚,赵英凯,黄玉清.计算机控制系统[M].武汉: 武汉理工大学出版社,2001
- [7] Chen Junhui,Huang Tien-chih.Applying neural network to on-line updated PID controllers for nonlinear process control.Journal of Process Control[J],2004,14(2):211
- [8] 固绳谷.电机与拖动基础[M].北京:机械工业出版社,1997
- [9] 王兆安,黄 俊.电力电子技术(第四版)[M].北京:机械工业出版社,2001
- [10] 童常飞.C8051F 系列单片机开发与 C 语言编程[M]. 北京航空航天大学出版社.北京:2005
- [11] 陈国呈.PWM 变频调速及软开关电力变换技术[M].北京:机械工业出版社,2001.3
- [12] 周航慈.单片机程序设计基础[M].北京:航空航天大学出版社,2002
- [13] 周航慈.单片机应用程序设计技术[M].北京:航空航天大学出版社,2002
- [14] 秦继荣,沈安俊.现代直流伺服控制技术及其系统设计[M].北京:机械工业出版社,1999
- [15] 徐晓刚,高兆法,王秀娟.VC++6.0 入门与提高[M].北京:清华大学出版社,1998
- [16] Marshall Brain,Lance Lorette.MFC 开发人员指南[M].译星翻译组 译.北京:机械工业出版社,1999
- [17] 阳宪慧.数据通信与控制网络[M].北京:清华大学出版社,2002
- [18] 常健生.检测与转换技术[M].吉林:吉林工业大学出版社,2001
- [19] 邵裕森,戴先中.过程控制系统[M].南京:东南大学出版社,2001
- [20] 郑大钟.线性系统理论[M].北京:清华大学出版社,2001
- [21] 钱 平.伺服系统[M].北京:机械工业出版社.2005
- [22] Stephen,A.Campbell The Science and Engineering Microelectronic Fabrication[M].曾莹,严利人,王纪民,张 伟 译.北京:电子工业出版社,2003
- [23] Morarim,Zaqirouse,New Jersey.Robust Process Control. Prenticed-Hall[J]. ine Englewood.Cliffs 1989
- [24] T.Kan machi I.Takahashi.Sensor-less Speed Control of an induction Motor with No Influnce of Secondary Resistance Variation. Proce[J]. IEEE IAS Annu.Meet.1993.408

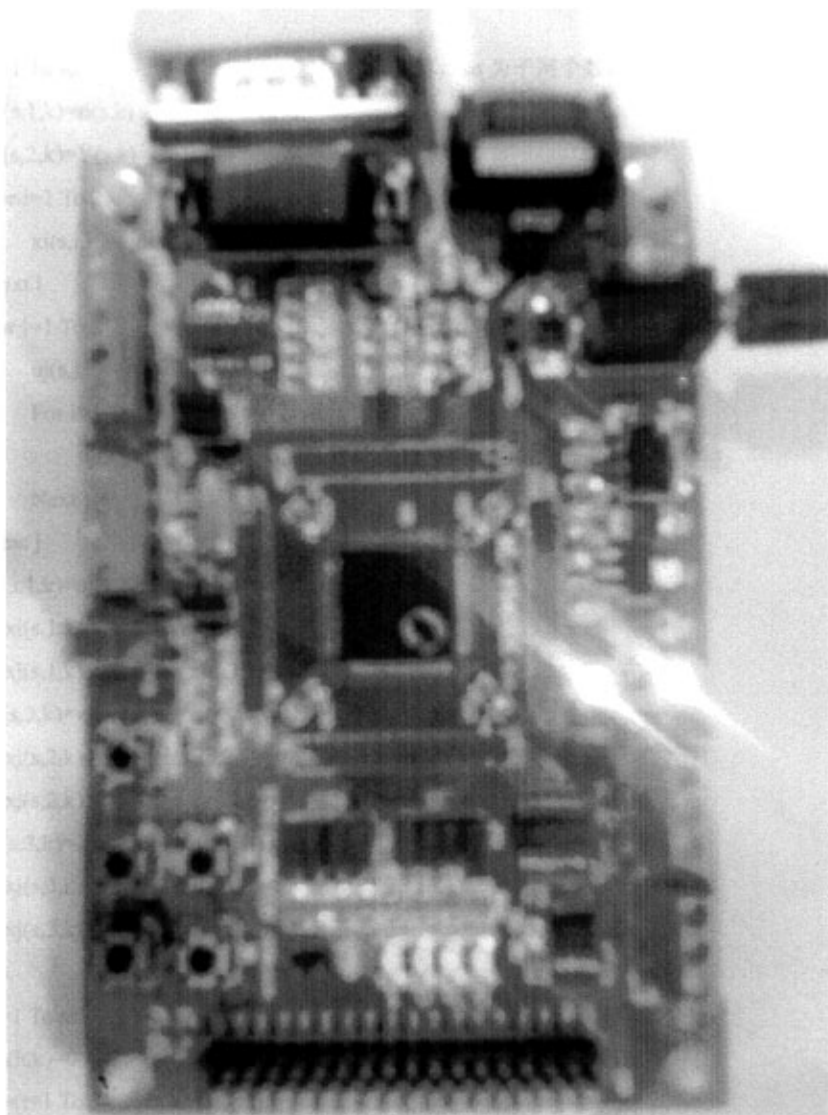
- [25] Walter Savitch. Problem solving with c++: the object of programming[M]. Addison-wesley Publishing Company, 1995
- [26] Peter Coad, Edward Yourdon. Object-oriented analysis[J]. Prentice-Hall, Englewood cliffs New Jersey, 1990
- [27] Ian Granham. object Oriented Methods[M], Second Edition Addison-Wesley Publishing company 1994
- [28] Athans M and Falb P L. Optimal control New[J]. York McGraw-Hill 1966
- [29] Callier F M and Desoer C A. Linear system theory[J]. New York: Springer-verlag 1991
- [30] Driels M. Linear control systems engineering[J]. New York: Mc Graw-Hill 1996
- [31] 李仁定. 电机的微机控制[M]. 北京: 机械工业出版社, 1999
- [32] 于海生等. 微型计算机控制技术[M]. 北京: 清华大学出版社, 2003

附 录

附录 I 攻读学位期间发表的论文

- [1].丁红卫, 郭秀才, 薛建勇. 关于风力发电的研究. 西安科技大学学报. 2007. 增刊

附录 II PID 神经元网络控制电路



附录III 部分主要程序代码

1.前向算法子程序:

```

Sub PIDNNcontrol ()                                \前向算法子程序\
For k=1 To 200
    For s=1 To so                                    \输入层计算, so 为子网个数, 下同\
        ui(s,1,k)=ri(s,k)
        ui(s,2,k)=Y(s,k)
        For i=1 To 2
            xi(s,i,k)=ui(s,i,k)
        Next i
        For j=1 To 3                                \隐含层计算\
            uj(s,j,k)=0
            For i=1 To 2
                uj(s,j,k)=uj(s,j,k)+ xi(s,i,k)*wij(s,i,j,n)
            Next i
        Next j
        xj(s,1,k)=uj(s,1,k)
        If xj(s,1,k) > 1 Then xj(s,1,k)=1
        If xj(s,1,k) < -1 Then xj(s,1,k)=-1
        xj(s,2,k)=xj(s,2,k-1)+uj(s,2,k)
        If xj(s,2,k) > 1 Then xj(s,2,k)=1
        If xj(s,2,k) < -1 Then xj(s,2,k)=-1
        xj(s,3,k)=uj(s,3,k) - uj(s,3,k-1)
        If xj(s,3,k) > 2 Then xj(s,3,k)=1
        If xj(s,3,k) < -2 Then xj(s,3,k)=-1
    Next s
    For h=1 To ss                                    \输出层计算, ss 为输出层神经元个数\
        uo(h,k)=0
        For j=1 To 3
            For s=1 To so
                uo(h,k)=uo(h,k)+xj(s,j,k)*wjk(s,j,h,n)
            Next s
        Next j
    Next j

```

```

If uo(h,k) > 1 Then uo(h,k)=1
If uo(h,k) < -1 Then uo(h,k)= -1
xo(h,k)=uo(h,k)
vo(h,k)=xo(h,k)
Next h
Call modelCout           \调用对象模型计算子程序\
For s=1 To so             \计算偏差和偏差平方值\
    e(s,k)=ri(s,k) - Y(s,k)
    ee(s,k)=e(s,k)^2
Next s
Next k
End Sub

2.反传算法子程序

Sub back ()               \反传算法子程序\
    For k=1 To 200        \计算目标函数\
        For s=1 To so
            Je(n)=Je(n)+ee(s,k)/200
        Next s
    Next k
    If Je(n) < miniE Then Go To finish \如果目标函数小于预定值, 停止\
    If Je(n) <=Je(n-1) Then Go To Change
    \如果本次目标函数小于或等于上次目标函数值, 权重值改变为新值\
    For s=1 To so         \否则恢复为上次权重值\
        For j=1 To 3      \隐含层至输出层权重值恢复\
            For h=1 To ss
                wjk(s,j,h,n)=wjk(s,j,h,n-1)
            Next h
        Next j
        For j=1 To 3      \输入层至隐含层权重值恢复\
            For i=1 To 2
                wij(s,i,j,n)= wij(s,i,j,n-1)
            Next i
        Next j
    Next s

```

```

Next s
Go To finish                                \跳转结束\
Change 1:                                  \修改权重\
For s=1 To ss                              \计算隐含层至输出层权重值修改量\
    For j=1 To 3
For h=1 To ss
    x1=0
    For k=1 To 200
        x1=x1+e(s,k)*Sgn((y(s,k+1)-y(s,k))*(xo(h,k)-xo(h,k-1)))*xj(s,j,k)
    Next k
    dwj(s,j,h)=- x1/200
    Next h
Next j
For j=1 To 3                                \计算输入层至输出层权重值修改量\
    For i=1 To 2
        x2=0
        For h=1 To ss
            For k=1 To 200
                x2=x2+ e(s,k)*Sgn((y(s,k+1)-y(s,k))*(xo(h,k)-xo(h,k-1)))*
                    wjk(s,j,h,n)*Sgn(xj(s,j,k)-xj(s,j,k-1))*(uj(s,j,k)-uj(s,j,k-1))*xi(s,i,k)
            Next k
        Next h
        dwi(s,i,j)=- x2/200
    Next i
Next j
For j=1 To 3                                \修改隐含层至输出层权重\
    For h=1 To ss
        wjk(s,j,h,n+1)=wjk(s,j,h,n)- dwj(s,j,h)*stepLength1
    Next h
Next j
For j=1 To 3                                \修改输入层至隐含层权重\
    For i=1 To 2
        wij(s,i,j,n+1)=wij(s,i,j,n)- dwi(s,i,j)*stepLength2
    Next i

```

$w_{ij}(s,2,2,n+1) = -w_{ij}(s,1,2,n+1)$

Next j

Next s

Finish:

End Sub