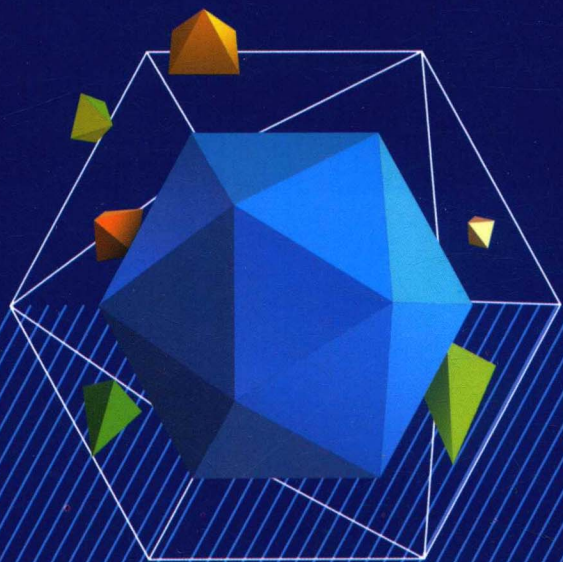


操作系统BIOS设计

CAOZUO XITONG BIOS SHEJI

杨立平 安旭阳 杨 延 编著



北京邮电大学出版社
www.buptpress.com

操作系统 BIOS 设计

杨立平 安旭阳 杨 延 编著



北京邮电大学出版社
www.buptpress.com

图书在版编目(CIP)数据

操作系统 BIOS 设计 / 杨立平, 安旭阳, 杨延编著. -- 北京: 北京邮电大学出版社, 2018. 1
ISBN 978-7-5635-5318-1

I. ①操… II. ①杨… ②安… ③杨… III. ①操作系统—程序设计 IV. ①TP316

中国版本图书馆 CIP 数据核字 (2017) 第 272399 号

书 名: 操作系统 BIOS 设计

著作责任者: 杨立平 安旭阳 杨 延 编著

责任编辑: 刘 颖

出版发行: 北京邮电大学出版社

社 址: 北京市海淀区西土城路 10 号(100876)

发 行 部: 电话: 010-62282185 传真: 010-62283578

E-mail: publish@bupt.edu.cn

经 销: 各地新华书店

印 刷: 北京九州迅驰传媒文化有限公司

开 本: 787 mm×1 092 mm 1/16

印 张: 16.75

字 数: 440 千字

版 次: 2018 年 2 月第 1 版 2018 年 2 月第 1 次印刷

ISBN 978-7-5635-5318-1

定价: 48.00 元

· 如有印装质量问题, 请与北京邮电大学出版社发行部联系 ·

操作系统是为离散设备打造数据通道,创建该系统工程的大数据平台。没有自己的操作系统,就没有自己的大数据平台。

(1) 操作系统的组成

操作系统是现代控制系统的核心,是以 BIOS 为内核的设备管理系统。操作系统由三部分组成:基本 BIOS、扩展 BIOS 和网架结构。

① 基本 BIOS。如显示器、磁盘、打印机、鼠标等。

② 扩展 BIOS。扩展 BIOS 面向对象,为谁服务就搭建谁的平台,选用谁的设备,如卫星、雷达、电机、银行的 POS 机等。

③ 网架结构。网架结构是 BIOS 的一种特例。

什么是 BIOS? BIOS 就是以设备表为核心,以域名为最小组成单位,与其他各种表项整合构成的逻辑电路设计系统。

设备表的操作过程受操作系统语言——原语控制,如通信机制原语。BIOS 属于底层设计,其存放的是与该设备的数据通道相关的各种参数,如 DMA、块、源地址与目标地址等。BIOS 支持块传和高速传输。

(2) 体系结构的设计理念

操作系统的设计是平台的设计,计算机体系结构是平台设计中的一种。平台必须分为三级架构:主板架构;I/O 板(即适配器,也叫 I/O 缓冲区)架构;设备架构。

保证数据从设备级内存到 I/O 级内存到主板级内存的数据通道的整体性、准确性。操作系统解决设备上的数据的来源问题,与软件无关。操作系统是体系结构的重要设计理念和基础。

操作系统是设备级管理,指令系统是用户级管理,操作系统与指令系统的接口是 INT 指令,即 BIOS 调用。

国外为我们提供的所谓操作系统往往是把 BIOS 调用外包的,存在着很多问题。

(3) 芯片设计理念

脱离了操作系统的 CPU 设计没有任何物理意义。CPU 设计必须满足体系结构的需求,体系结构由操作系统来定义,因此,它们的先后关系是:操作系统设计—体系结构设计—CPU 设计。

芯片设计是 PCB 印刷电路板的进步和延伸。

(4) 通信机制和进程调度

制定行业的通信协议及通信标准,用操作系统语言来描述,即 CPU 管理的通信机制与进

程调度。这是非常重要、非常关键的问题。

我国的通信标准必须由自己建立,否则我们运行的数据将遵循其他国家的通信标准,那么我国的信息安全和国防安全将无法保障。

制定我国自己的通信标准是实现现代化的重中之重!

最后作者以本书理论为指引,以航空母舰战斗群为背景,探索研究了如何改造已有成熟商业操作系统,来创建军用大数据平台,并完成了一套包含 200 台设备、1 024 个域名的操作系统模型机的理论设计工作,为各类军用设备打造数据通道,夯实理论基础。

在本书中,作者阐述了操作系统设计的来龙去脉,还操作系统以本来面目,解读操作系统到底是怎么回事,如何设计操作系统,为操作系统设计指明方向,引领道路。系统工程庞大而复杂的数据链条如何形成,如何处理,如何实现一个统一、完整、有效、快速的作战主体,这都是操作系统要解决的问题。

阅读本书的读者最好有一些自动控制专业的基础,从自动控制的角度来阅读、理解本书内容会容易些。另外,本书阐述的所有内容(从操作系统设计一直到底层的逻辑电路的实现)都是为工程实践服务的。因此,难免与有些教材的内容发生冲突,希望各位读者仁者见仁,智者见智。

作 者

第 1 章 系统概述	1
1.1 国内的研究状况和教育现状	2
1.2 体系结构的四个子系统	5
1.2.1 计算机体系结构下设的四个子系统	6
1.2.2 各学科之间的结构关系	9
第 2 章 操作系统与自动控制原理	14
2.1 环和域	15
2.2 PID 调节	21
第 3 章 操作系统与离散数学	25
第 4 章 操作系统与数据结构	38
4.1 表、树、图	42
4.2 表的逻辑电路设计与实现	59
4.3 文件管理与计数器结构	66
第 5 章 操作系统	73
5.1 操作系统的 CPU 管理	81
5.1.1 通信机制	82
5.1.2 进程调度	92
5.1.3 通信机制和进程调度的具体实施方法和策略	101
5.2 操作系统的内存管理	111
5.2.1 大数据平台	112
5.2.2 内存分配	117
5.2.3 磁盘管理	124
5.3 操作系统的设备管理与 I/O 卡管理	136
5.3.1 设备管理	136
5.3.2 I/O 卡管理	143

5.3.3 IOP 管理	153
5.4 网络操作系统	170
5.4.1 技术路线	174
5.4.2 网络通信机制	177
5.4.3 网络进程调度	188
5.4.4 网络操作系统的拓扑结构	198
5.5 网架结构的逻辑电路设计与实现	206
第 6 章 习行性分析	246
致谢	262

操作系统的核心是打造离散设备的数据通道,创建该系统工程的大数据平台,是现代控制理论的基础之一。

我国现代化工业的发展,科技的创新离不开计算机操作系统,计算机操作系统是一个对各类不同属性离散设备的控制系统。计算机本身就是一个将各类不同属性的离散的设备打造成一个大数据平台的控制系统。计算机操作系统是将各学科研究成果集成控制并整合到同一个数据平台上,是多学科的综合学术体。其中主要包括:自动控制原理、离散数学(群、环、域,图论)、拓扑学、数据结构、计算机体系结构等众多的学科。

现在时髦的语言是大数据,大数据需要大数据平台来存放。底层的数据从何处来,到何处去,谁给的,这些都是操作系统需要解决的问题。其中,很重要的是大数据的安全问题。

没有自己的操作系统,就没有自己的大数据平台,没有自己的现代控制系统。如何解决这些问题。

系统工程对设备的需求往往是按行业来划分的,创造行业的操作系统是要为该行业所需系统工程的离散的不同属性的设备制定标准化的、统一的通信协议及通信约定,以满足操作系统对设备跟踪、管理的需要。这个进程也是创建该设备进入大数据平台的通道过程。

制定行业的通信协议及通信标准,用操作系统语言来描述,即 CPU 管理的通信机制与进程调度,这是非常重要、关键的问题之一。

操作系统是一个控制系统,目的是控制设备,核心是管住设备上的数。站在系统工程的角度上,体系结构是一个最大的系统工程,下设四个子系统:操作系统、指令系统、编译系统和人机界面系统。操作系统是一个小的系统工程,包含多个子系统:自动控制原理、离散数学、拓扑学、数据结构、计算机体系结构、网络架构等。

操作系统设计就是围绕自控原理中的环和域,开环、闭环(即 PID)调节,离散数学中的群、环、域,拓扑学中的图论(有向图和无向图),网架结构中的规则网络和随机网络及有源网络和无源网络,数据结构中的表、树、图等几方面来阐述如何为设备打造数据通道,创建数据平台。

操作系统、数据结构不是软件,离散数学也不是纯数学。那么,什么是操作系统?什么是离散数学?什么是数据结构?

操作系统是为设备打造数据通道,创建数据平台,并制定设备统一的通信标准。根据系统工程的需求,将不同属性的设备集中控制在一个数据平台上。

自动控制原理的核心是 PID 调节,PID 调节包括开环控制调节和闭环控制调节,目的是

打造一个大数据平台的控制平台,属于现代控制理论的核心。

离散数学的核心是群、环、域,环和域是操作系统所管辖的设备级,群是系统级,若干个设备级构成系统级,即群。群也是属于打造大数据平台的核心概念之一。

群就是系统环,系统环包含多个设备环,设备环对应一个域名,域名指的是设备的控制参数,对应着设备传感器。自控原理中的环和域就是离散数学中的环和域,离散数学是工程数学,是现代控制理论的基础数学之一。

数据结构的核心是表、树、图,表、树、图属于大规模集成电路设计的基础理论之一,主要作用是为该设备打造进入大数据平台的数据通道,为设备分配各级通道的内存空间。

拓扑结构主要解决的就是路径问题,包括网架结构。根据图的节点、链路、映射函数为设备打造数据通道,创建数据平台。节点、链路、映射函数构成一条路径,路径就是数据通道。

网络架构的重点就是规则网络和随机网络,有源网络和无源网络,无论是哪种网络架构,都要符合电工学原理。

操作系统设计是体系结构的设计理念,是大规模集成电路的设计理念,是 PCB 印刷线路板的设计理念,是芯片的设计理念,目的是为设备制定统一的通信标准,并根据系统工程的需求,将不同属性的设备集中控制在同一个大数据平台上。

下面主要介绍三方面内容:①我国计算机发展的基本情况以及国内高等教育的现状;②操作系统、指令系统、编译系统、人机界面系统各自功能、内容的介绍;③操作系统、自动控制原理、离散数学、数据结构这四个学科之间的因果关系。

1.1 国内的研究状况和教育现状

(1) 理论体系:我国计算机界大部分认为操作系统、数据结构、编译系统都是软件内容,均是从软件编程的角度来研究。

(2) 实际应用:我国计算机发展主要以软件开发为主,而控制系统的底层芯片大量需要进口,脱离操作系统的 CPU 设计没有任何物理意义。

这些理论将计算机的发展道路引向软件编程,导致大部分人认为计算机就是软件的开发和应用,而计算机的真正价值在控制系统。因为控制系统是自动化的核心,是国防现代化的关键,对国家安全起到至关重要的作用。

计算机在控制系统中的应用就是数字化的应用,而这种应用的本质是硬件及其管理,本书研究的是硬件电路,从硬件的角度分析 CPU、操作系统、编译系统、指令系统和总线结构,并已经具备了相对成熟的基于硬件的科研理论,按照该理论深入研究就能够解决我国存在的上述问题,进而实现从硬件上设计分析 CPU、操作系统等,最终改变计算机发展过于依赖国外的现状,使计算机回归到控制系统的本源上来。

系统工程按行业来划分,设备往往属于系统工程的需要,是面向对象的,不同的行业有不同的需求。例如,电力系统的操作系统和石油化工系统的操作系统在设备上以及数据平台上各有各的设备,各有各的系统工程的需求。

“面向对象”的设计是为特定设备服务的,是计算机结构的设计,此处的对象指的是计算机的控制对象,比如显示器、鼠标、键盘、打印机、电机等。对它们的控制是各不相同的,需要根据设备的特点设计不同的计算机硬件结构和与之配套的操作系统,以发挥这些系统的最佳性能,

具有最高的使用效率。这绝不是靠软件开发可以实现的,它是实实在在的硬件电路设计。

20 世纪 80 年代以前,经典控制论主导整个控制理论领域。经典控制理论的研究对象是单输入、单输出的自动控制系统,特别是线性定常系统,限于对标量的控制。随着控制对象扩展为多输入多输出、非线性、时变、离散的系统,则必须站在现代控制论的基础上来分析、研究。

现代控制论的核心是计算机系统。现代控制论以离散数学、数据结构、线性代数为基础,以计算机操作系统、逻辑电路为实现基础。离散数学为现代控制论提供数学模型,将离散的多个相关标量综合考虑进行处理,即对向量进行控制,从而能够从全局对被控对象进行自动调节。这与经典控制论不同,经典控制论只能做到标量控制。

我国高等教育存在以下几个问题:

(1) 离散数学当作纯数学来讲,偏离了离散数学作为工程数学对工程设计的指导意义。

(2) 把数据结构理解为软件,而事实上软件只能对数据结构进行描述,而不是其本质,数据结构为硬件设计提供指导。

(3) 认为计算机操作系统是软件,设计计算机操作系统看成了软件开发。

(4) 把编译系统完全看作软件,忽略了编译系统的硬件支持。

(5) 计算机专业的所有学科我们都学了,别人说起某些内容我们也有印象。但是,我们不知道学了这些课程用来做什么,应用于工程上的哪些方面。我们一直在把计算机专业当作文科来记忆、背诵,而忽略了它实际的工程应用。

计算机的发展是一个庞大的系统工程,它是一门综合性的学科,必须在现代控制理论的指导下完成对体系结构的设计理念,在这个架构中应该包括系统工程所需的各种设备,如在航母上,这个设备的架构就应该是满足航母整个作战指挥平台的需求,即扩展 BIOS,这个架构必须自己设定,因为没有人给你。

这个架构主要是指各类设备进入操作系统底层数据通道的芯片设计,只有该设备的数据进入该系统的数据平台之后,怎么处理这些数据才是软件编程的事情。所以说,操作系统与软件无关,操作系统传输数据。也就是说,为该设备怎么分配内存地址是操作系统要解决的事情。

操作系统是自动控制原理进一步的理论延伸,自动控制原理体现的是环和域的理念,操作系统将离散的环和域集成控制在同一个大数据平台上,体现的是群的理念。

市面上流行的 UNIX 源代码,它的局限性很大,它提供的设备非常狭窄,无法满足我国现代控制操作系统设计的需要。

除了基本的管理设备(如鼠标、键盘、显示器、磁盘等)之外,我们设计了一款由 200 台设备、1 024 个域名组成的操作系统模型,将在后面分成几个章节全部展开。

操作系统设计是以设备表为核心的一系列表操作的展开过程,复杂而清晰,路径简单而明了,域名是操作系统设计的最小单元,一个域名对应着一个设备号的一个入口参数。例如,INT xxH,入口参数是 nnH。

每台设备都对应着相应的设备号,都有固定的中断类型码,在该台设备的若干个子功能调用中,每个最小的子功能调用就是它的域名。入口参数 nnH 即代表它的域名。例如,在设备表中,雷达的设备号为 10,中断类型码为 200,域名(入口参数)为 002,即表示雷达的接收功能。

域名是操作系统中的最小单位,每个域名对应着一个设备号,一个中断类型码,一个设备表的入口地址,一个设备的控制参数,一个 DCT 表,一个 PCB 表,一个文件表,一个 TSS 表,

一个内存分配表(PAT表),一个通道控制表,形成了该设备的一个数据通道,在各级的内存地址里一定放着该域名对应的数,其中包括大数据平台的存储空间。即不仅将该雷达设备的数据域名的通道划分好,并且把该雷达接收数据的大数据平台的存储空间也已经规划好。

自控原理的环和域就是离散数学中的环和域,这就证明了离散数学是工程数学。操作系统是自动控制原理的理论延伸,两者是一脉相承,不可分割的整体。操作系统的核心是形成数据链,解决数的存储及传输问题。

设备数据以文件的形成存在,文件是数据结构重要的表现形式,数据结构的核是表、树、图,表中存放地址用来给文件分配内存地址和磁盘地址。文件包括顺序文件、索引文件和散列文件,顺序文件是线性结构,索引文件是树型结构,而散列文件是图型结构。

设备就是文件,数以文件的形式进行存储,所以说设备属性决定文件属性。例如,光盘只支持顺序文件。操作系统是控制设备的,以文件结构的形式对设备进行管理,设备的总线统称为外部总线,数据通道指的是外部总线、I/O总线和系统总线三级总线通道之间的关系,该通道分为三级管理:系统级、I/O缓冲级和设备级。各有各的CPU,各有各的内存空间,各有各的专属接口。

操作系统是体系结构的设计理念,体系结构是在离散数学和数据结构共同指导下的逻辑电路的设计,是由硬件实现的,其中,离散数学提供数学模型,数据结构体现各部件之间的逻辑关系,操作系统是把各设备提供的数据集成到同一个数据平台上。

操作系统是计算机体系结构的设计理念,体系结构是操作系统的物理支撑和实现形式。操作系统设计是逻辑电路的设计理念,与软件无关,与指令无关。

操作系统是规则的制定者,规则指的是设备的统一通信标准。软件只是操作系统设定的规则之内的舞者,它不能超越于规则而存在。体系结构用三级的设计理念来体现操作系统的逻辑设计思想,此三级设计包括CPU设计、I/O卡设计和设备CPU设计。主要设计三级CPU中各自的功能和作用以及它们之间的通信,最终目的就是控制外部设备。

本书的几个主要论点:

(1) 操作系统是通过自控原理的开环、闭环,离散数学的群、环、域,数据结构的表、树、图将离散的不同属性的设备集成控制到同一个大数据平台上,并为设备打造数据通道,制定设备统一的通信标准。

(2) 操作系统不是软件,它是一个数据控制系统,是自动控制原理的理论延伸,是大规模集成电路的设计理念。操作系统只管数据,保障数据链的形成。

(3) 没有通用的操作系统,也没有通用的CPU,操作系统设计是面向对象的设计,面向对象即面向设备,面向对象的属性不同,操作系统也会不同。任何游离于操作系统之外的,单一某种CPU的设计没有任何物理意义。

(4) 离散数学是计算机专业的基础课程,是工程数学,它是自动控制原理的理论基础,为现代控制论提供理论指导和依据。

(5) 数据结构指的是数的存储结构,核心解决的问题是通过表给设备分配内存空间和磁盘空间,表中存放着大量的地址,由表给计数器(SI、DI)赋值。归根结底,数据结构是对各个计数器的设计,数据结构的本质是大规模集成电路的设计理念,是印刷线路板的设计理念,是芯片的设计理念,与指令无关,与软件无关,与高级语言(C语言、Java语言)无关。

本书的几个目标,解决的几个问题:

(1) 建立我国各行业自己的操作系统的通信协议及通信标准,将各离散的设备纳入该行

业的数字化管理当中来,这是数字化建设的重大措施和步骤。

(2) 建立我国自己的操作系统标准。实际上操作系统设计包括了核、高、基、设计,包括了大规模集成电路设计,包括了网架结构设计,包括了数控设计,它们是四位一体的关系,属于一个不可分割的统一的研究课题。

(3) 设计我国自己的以操作系统理论为指导的,系统工程需求为目标的“系列中国芯”。单一的、游离于操作系统之外的芯片设计没有任何物理意义。

(4) 我国计算机的面向对象仍然局限于软件开发,扩大面向对象在国防、工业上的应用,以保障我国自己的国防信息安全。

当底层的逻辑电路是我们自己设计的时候,就不会存在后门问题、加固问题等。此时,信息就是安全的。信息安全问题指的就是数据安全问题,信息即数据。当我们对数来源于哪里,数据链是如何形成的,数是如何传输的等一系列问题都搞明白的时候,信息安全问题自然也就解决了。

在当今如此复杂的国际环境下,国防事业显得尤为重要,在国防上,我国必须要有自己的操作系统标准和工业总线标准,这些标准规则的设计的最终体现是逻辑电路的设计。例如,航母战斗群,它本身就是一个系统工程,是一个群,包括驱逐舰、护卫舰、潜艇、天上的卫星、舰上的雷达及各武器装备之间的因果关系,庞大而复杂的数据链条如何形成,如何处理,如何达到一个统一、完整、有效、快速的作战主体,这都是操作系统要解决的问题。

阅读本书应具备的专业知识:

读者最好有一些自动控制专业的基础,从自动控制的角度来阅读、理解本书内容会比较容易些。另外,本书阐述的所有内容都是为工程实践服务的,从操作系统设计一直到底层的逻辑电路的实现。因此,难免与有些教材的内容发生冲突,希望各位读者仁者见仁,智者见智,使我国的操作系统事业全面地提升和发展。

1.2 体系结构的四个子系统

站在系统工程的学科角度上,计算机体系结构是最大的一个系统工程。如图 1.1 所示,计算机体系结构下设四个子系统:操作系统、指令系统、编译系统、人机界面系统。操作系统又是一个小的系统工程,下设三个子系统:自动控制原理、离散数学、数据结构。

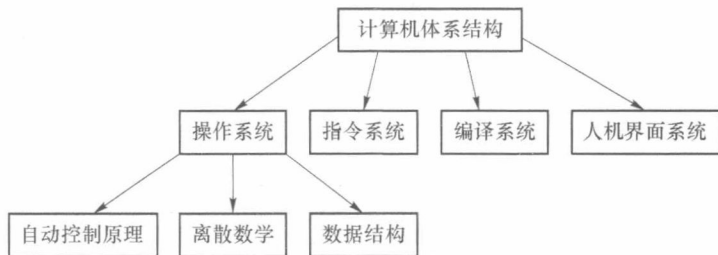


图 1.1 各学科间的因果关系

下面主要介绍两方面内容:一是操作系统、指令系统、编译系统、人机界面系统各自的功能作用是什么,主要解决哪些问题,包含哪些内容。这样,在层次上有个清晰的界定。二是操作系统、自动控制原理、离散数学、数据结构四者之间的拓扑关系。

1.2.1 计算机体系结构下设的四个子系统

1. 操作系统

操作系统是控制许多离散设备,为设备打造数据通道,创建数据平台,制定统一的通信标准。根据系统工程的需求,将不同属性的设备集成控制在同一个数据平台上。

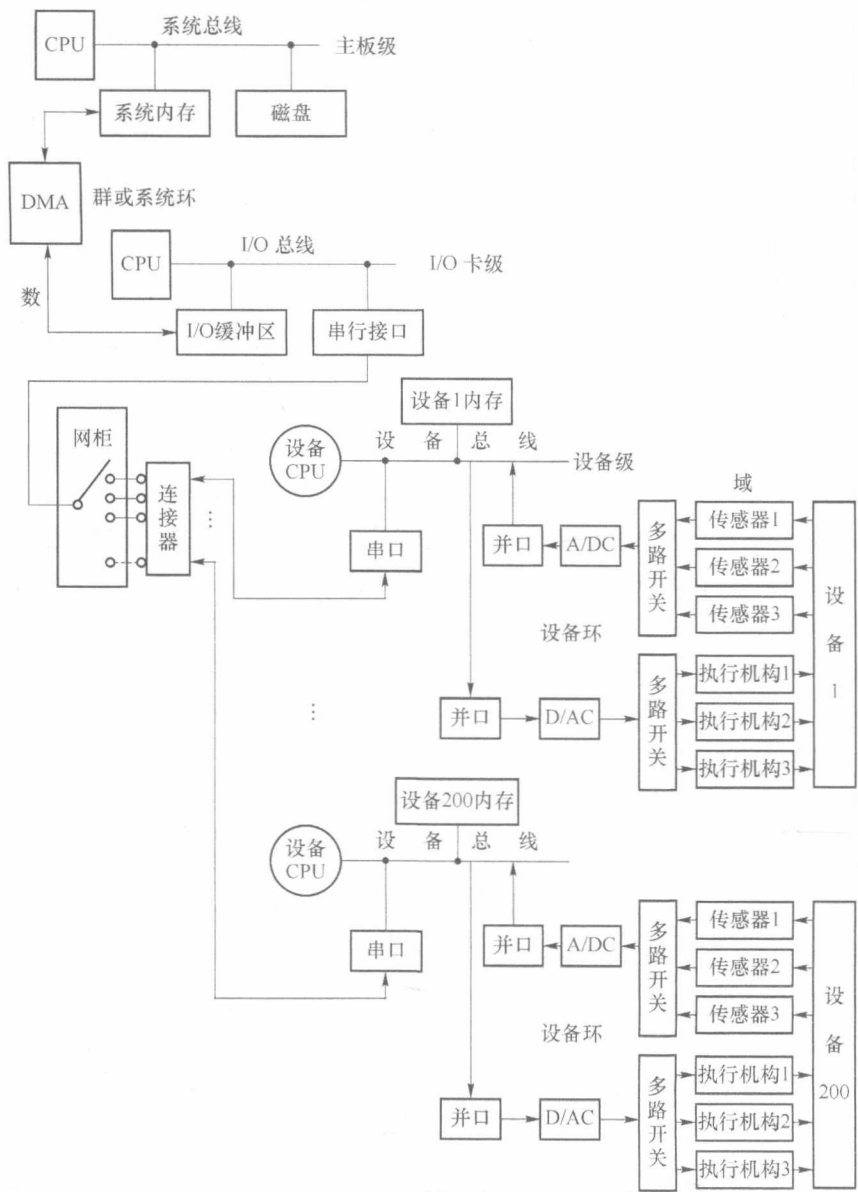


图 1.2 系统结构示意图

如图 1.2 所示,操作系统的核心是给设备分配内存空间和磁盘空间,并打造数据通道,数据通道的核心是 DMA。其中 DMA、内存、总线是数据通道的重要组成部分。

操作系统是一个控制系统,因为操作系统在整个数据链的形成过程中各模块上 CPU 的功能定义不同。操作系统是系统级设计,包括四个及四个以上功能模块,各模块都有自己的

CPU、内存及接口,数据链的形成就是对各模块上内存队列的划分,首指针、尾指针的赋值。

操作系统要想保证数据链的形成,首先要解决的是四大模块上的内存分配问题。操作系统通过数据结构中的表、树、图完成对各级内存的分配及队列首、尾指针的赋值。进程调度和通信机制来保证哪台设备、哪个文件、哪个域名、哪个逻辑块号对应数据链的形成。数据链的操作是对各模块上内存队列的操作,在各自模块之内为数据流,各模块之间为数据链,数据链把各模块队列的首、尾指针链接到一起。所以说,操作系统是以内存分配为核心的,即为各个设备提供的数据链条分配各自不同的内存空间,即大数据平台。

操作系统没有军用、民用和商用之分,操作系统就是个控制系统,根据系统工程的需求,由设备属性决定操作系统的性质。

没有通用的操作系统,也没有通用的 CPU,操作系统设计是面向对象的设计,面向对象即面向设备,设备的属性不同,操作系统也会不同。例如,磁盘设备的属性要求将磁盘内存的空间转换成台面号、柱面号、盘面号、扇区号,从而找到扇区地址;显卡设备的属性要求将显卡内存的空间转换成块号、行号、列号,从而找到像素点地址。操作系统要满足设备的工艺流程的需求,满足系统工程的需求。

所谓没有通用的操作系统,是因为操作系统本身对设备的管理分为两部分:一部分是基本块,如磁盘、显示器、打印机等,也叫基本 BIOS;另一部分是操作系统的扩展部分,即面向对象,如卫星、雷达、电机等。因面向对象不同及系统工程的需求不同,支持该系统的设备属性也就不同,由设备属性提供的数据链条也不同,所以,没有通用的操作系统。

没有通用的 CPU,是因为在操作系统整个数据链的形成过程中各模块上 CPU 的功能定义不同。操作系统包括四个功能模块:CPU 模块、通道模块、I/O 卡模块和设备模块。各模块都有自己的 CPU,因为各模块 CPU 的功能不同,数据结构也就不一样,表、树、图也不一样。因为,各模块上的 CPU 必须要满足进程调度和通信机制的需求和支持。

所以,游离于操作系统之外的 CPU 设计都没有任何物理意义。

操作系统设计应包括核、高、基、理论的研究,大规模集成电路设计,网络设计和数控四个方面,它们是四位一体的关系,是一个统一的整体,不要再把这个统一的整体划分成不同的研究课题。

操作系统是一个控制系统,最终目标是:管数。通信机制和进程调度都是为数服务的。

数据结构的最终目标是:解决数在内存中的存储问题,为数据文件分配内存地址。

计算机体系结构的最终目标是:根据系统工程的需求定义网架结构的拓扑结构,网架结构指的是路径,是拓扑结构。

操作系统形成的数据链包括通信链和调度链,通信机制是一系列的表展开过程,由原语保障实施。通信链如图 1.3 所示。

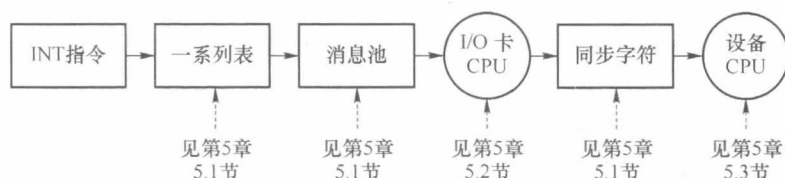


图 1.3 通信路径示意图

进程调度是队列操作,核心是过桥理论,即谁管 DMA。系统 CPU、I/O 卡、IOP 都可以管

DMA。调度链如图 1.4 所示。

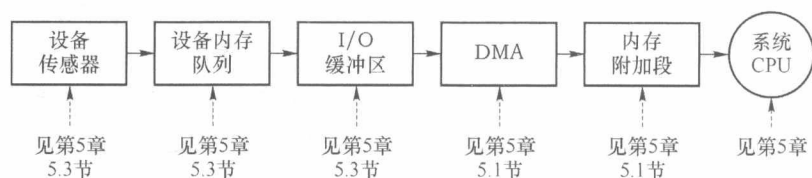


图 1.4 进程调度路径示意图

2. 指令系统(指令框架结构集)

操作系统解决了数的问题后,指令系统解决的问题是如何处理这些数据。指令系统是围绕数怎样处理来进行指令的定义和设计,没有数据,指令也就没有任何物理意义。先有数据再有指令,指令是为数服务的。指令系统与操作系统的接口是设备管理指令(INT 指令)。前提是操作系统必须先有此设备,才能有相应的设备管理指令,编程过程不可能凭空产生,因此,指令系统设计也是需要面向对象的。

指令系统是用户级的,指令的设计由方框语言来描述,一条汇编指令对应着一段微程序,一条微指令对应着若干种微操作,按 CPU 周期来划分。

指令系统的本身要解决的问题是将内存数据段的数对 CPU 内部各寄存器进行分配,如何分配由指令的寻址方式决定。指令系统的内容应包括两大部分:用户级指令和系统级指令。用户级指令包括:①高级语言指令;②宏汇编;③汇编指令;④伪指令;⑤微程序、微指令、微操作。系统级指令就是原语指令,如进程控制类原语、资源管理类原语等。原语是属于操作系统语言,对用户是不透明的。

一条高级语言指令(如 C 语言指令)对应一段宏汇编,是一段汇编指令的集合;一条汇编指令对应一段微程序,是一段微指令的集合;一条微指令是一段微操作的集合,微程序对应着计算机可以执行的机器码,是一个控制块,决定数据该进行何种操作。

指令系统的核心是微程序控制器,整个计算机的运作是由指令控制的,不管什么指令,最后都是微操作级的。因此,要想设计指令系统,需要对 CPU 的体系结构以及 CPU 内的各个寄存器非常熟悉才可以设计指令系统。整个时序进行分配时,在哪一个 CPU 周期里面,各个寄存器要达到怎么样的目的,实现什么功能等一系列问题都需要由微操作来控制。

指令系统设计也是面向对象(设备)的设计。例如,建立一个动态的、智能化的、高等级的大型数据库,也必须要有底层数据链的支持及指令之间的相互组合。因此,它也必须面向对象的,与操作系统一样,也没有通用的数据库。数据库的建立也需要多台设备来支持它的文件组织结构。在保证数据链的执行过程中,不存在兼容不兼容的问题。

指令系统是编译系统的底层,没有指令系统就没有编译系统。

3. 编译系统

首先要加以区分的是:编译系统是编译系统,支撑软件是支撑软件。支撑软件是面向对象的,如 VC++6.0、CVF 6.6、Oracle Database 11g 等。

用户可以直接调用各种设备,实质上真正的高级语言可以调用操作系统所定义的各种设备,编译系统的核心是如何对操作系统的底层设备进行编译,包括大型数据库,数据库编程能直接调用哪些数据,在调用数据时就是直接调用哪些设备。大型数据库的编译系统将支持对 200 台设备,1 024 个域名直接进行调用的数据库语言。

编译系统要想将用户程序变成可执行的,则必须为用户程序分配内存空间和磁盘空间,将用户级程序的逻辑地址转换成可执行的代码级的物理地址。该物理地址给 CS、DS、SS 和 ES 寄存器赋值。

编译系统与原语级操作有关,因为编译系统涉及资源分配。原语操作对用户是不透明的,它取决于状态,分配内存空间就必须要用到原语级的操作。有限的自动机制和半自动机制都跟原语有关,原语是用来赋值的,为各种表、树、图来赋值。

构建操作系统的大数据平台,就是构建大型数据库。

编译系统是面向对象的,此对象指的是高级语言,是对大型数据库进行编译的过程,数据库语言就是高级语言。编译系统是个过程,这个过程应该包含对操作系统的调用过程。一个大的数据平台往往由不同类型的大型数据库构成,许多台设备组成一个大型数据库,大型数据库如何调用这些设备,将是编译系统要解决的重大问题之一。只有这样,才能使大型数据库管辖的各种数据动态起来,智能起来。编译系统的核心功能是把大型数据库和操作系统连接起来,从而使大型数据库面向操作系统,面向设备,面向对象,成为真正意义上大数据平台,使操作系统各设备调用进来的数据和用户级的需求达到了完美的结合和统一。

4. 人机界面系统

编译系统和指令系统之上是人机界面系统,人机界面系统包括一些键盘指令、鼠标指令等系统级命令字,用户编程时可以直接调用这些命令字。

人机界面系统就是一个壳,是外包,很多人把人机界面系统命令字分为内部命令字和外部命令字。常用的内部命令字有:MD 建立子目录、CD 改变当前目录、RD 删除子目录命令等。常用的外部命令字有:DELTREE 删除整个目录命令、FORMAT 磁盘格式化命令、DISK-COPY 整盘复制命令等。

内部命令随着人机界面系统的启动同时被加载到内存且长驻内存,因此,只要启动了人机界面系统,用户就可以使用内部命令;外部命令是储存在磁盘上的可执行文件,执行这些外部命令需要从磁盘将其文件调入内存,因此,外部命令只有该文件存在时才能使用。

现今的 Windows Server 2003、Windows XP 系统、Windows 7 系统等是作用在人机界面系统之上的软件外包,它们将人机界面系统上的命令字进行软件外包,对用户屏蔽掉其具体的实施细节,是在人机界面系统之上披上一层软件外衣。这样,使用户操作变得更加方便,界面更加友好。

下面重点介绍的是第二部分——操作系统设计。

1.2.2 各学科之间的结构关系

操作系统是一个控制系统,是现代控制理论的核心与基础。什么是操作系统,操作系统解决的问题是什么,如果解决不了这个问题,就根本没法解决大数据问题。

操作系统是为设备打造数据通道,创建数据平台,制定统一的通信标准。根据系统工程的需求,将不同属性的设备集中控制在一个数据平台上。

① 数据平台:根据系统工程的需求选择不同属性的设备,设备以数据文件的形式存在。

② 数据通道:从 INT 指令进来,一直到设备出去。

③ 通信标准:通信机制是通信标准的最底层,解决三个问题。一是怎么找到设备。二是如何给设备分配各级内存地址空间。三是该设备进入数据平台的先后次序,即进程调度。

自控原理的开环、闭环,离散数学的群、环、域,数据结构的表、树、图都是为了给设备打造

数据通道,创建数据平台。

1. 自动控制原理与操作系统

现代控制论的核心是计算机,计算机的核心价值体现在控制系统。控制系统包括两大部分:自动控制原理(数的处理过程)和操作系统(数据的传输过程)。操作系统是一个数据控制平台,它将数据从外部设备的传感器传送到系统 CPU(控制中心)。之后,系统 CPU 运用 PID 调节算法完成对数据的处理并反馈给外部执行机构,PID 调节是自动控制原理的核心内容。

PID 调节是用户通过编程,对该设备采集来的数据进行加工处理,此过程需要有运算器的支持。

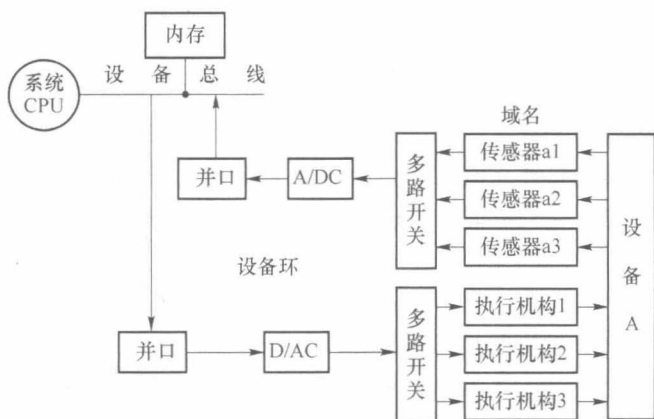


图 1.5 PID 调节数据环示意图

控制理论分为经典控制论和现代控制论。

经典控制论的核心是 PID 调节,PID 调节包括开环调节和闭环调节。如图 1.5 所示,设备上的每个传感器表示一个域名,传感器将采集到的数据经过 A/D 转换后传送给系统 CPU(控制中心)。PID 调节是一个负反馈调节,信号经过编码器反馈给 CPU,CPU 利用一些复杂的 PID 调节算法计算、处理后,经过 D/A 转换向执行机构(如伺服电机)发送相应的指令以达到控制外部设备的目的。

当 PID 调节是一个闭环调节时,设备环 a1 的数据路径是:设备 A→传感器 a1→多路开关→A/D 转换器→并口→系统 CPU→并口→D/A 转换器→多路开关→执行机构 1→设备 A。

当设备环 a1 没有反馈时,设备环 a1 是一个开环,数据路径是:设备 A→传感器 a1→多路开关→A/D 转换器→并口→系统 CPU。

自控原理的环和域就是离散数学的环和域,离散数学是工程数学,是现代控制论的基础理论之一。设备的每个传感器对应一个域名,一个域名对应一个设备环,一个设备环对应设备的一个控制参数。每台设备包含多个设备环。

经典控制论只能做到控制环和域,并没有出现群的概念,因此经典控制论只能控制离散的、单一的属性设备,而不能将各种不同属性的设备整合到一起,无法做到对群的控制,无法构成一个由所需多种设备支撑的大数据平台。

群就是系统环,一个群对应着一个系统环。群是许多设备的整合,即许多计算机的整合。

20 世纪 80 年代以前,经典控制论主导整个控制理论领域。经典控制理论的研究对象是单输入、单输出的自动控制系统,特别是线性定常系统,限于对标量的控制。随着控制对象扩展为多输入多输出、非线性、时变、离散的系统,则必须站在现代控制论的基础上来分析、研究。