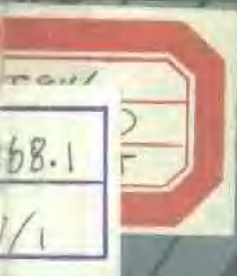


微处理器系统调试

〔英〕N·甘尼·E·法雷尔·著
朱文章·译
梅协英·唐裕亮·校



TP368.1
GN/1



微处理机系统

[英] N·登尼 著
E·法雷尔 著
朱文章 译
梅协英 校
唐焰亮 校



025465

0026824

人民邮电出版社



Microprocessor System Debugging

Noordin Ghani

Edward Farrell

RESEARCH STUDIES PRESS

1980

内 容 提 要

本书以实际工作为基础提出一套经济实用的微处理机系统调试方法,系统地介绍了较低级的控制面板调试系统和较高级的全屏幕调试系统的一般原理和具体实现。本书结合具体机器,描述了系统硬件和软件设计,并用流程图做详细说明。

本书可供从事计算机设计和应用的科技人员阅读,也可供大专院校有关专业的师生参考。

JS283/01

微 处 理 机 系 统 调 试

〔英〕 N·甘尼
E·法雷尔

朱 文 章 译

梅 协 英 校
唐 裕 亮

人民邮电出版社出版

北京东长安街27号

北京顺义兴华印刷厂印刷

新华书店北京发行所发行

各地新华书店经售

开本: 850×1166 1/32 1986年10月第一版

印张: 4 24/32 页数: 76 1986年10月北京第1次印刷

字数: 119 千字 印数: 1—4800 册

统一书号: 15045·总3284—有5481

定价: 1.00元

译者序言

七十年代以来,大规模集成电路的发展和微处理机芯片的出现,使计算机技术进入了新的发展阶段。微处理机具有体积小、重量轻、性能/价格比高等优点,为其应用、推广和普及开辟了广阔的前景,它已渗透到社会生活的各个领域。

引进、推广和应用微处理机,是我国新技术革命的一个经济而有效的途径。近年来,我国对微处理机的研究和应用有了较大发展。各个领域的科技人员积极学习微处理机新技术,并且根据自己的实际情况研究微处理机的应用。

根据特定应用设计出的系统,通常不是连上线路、加上电源和装进所编写的程序就能进行工作。所设计的系统必须经过查错、修正,即所谓“调试”过程才能使其正确地执行预定任务。设计者要方便、迅速地进行调试工作,必须有一套方法,以及所需要的调试设备。

为了加快应用系统的开发,使设计的系统尽快地投入运行,发挥它的经济效益和作用,就要根据具体情况建立经济实用的调试系统,即建立系统开发基地。为适应这种需要,将“Microprocessor System Debugging”一书翻译出版,供从事微处理机应用研究的科技人员以及大专院校计算机和自动控制专业的学生阅读,供计算技术和自动化实验室人员参考。希望它能对我国微计算机的推广应用有所助益,对读者建立适用调试系统有所启迪。

本书是作者及其同事在纽卡斯尔大学计算研究所数字系统实验室建立调试系统的经验和方法的总结。作者根据该实验室的目的和要求,对较低级的专用控制面板调试系统和较高级的全屏幕调试系统的一般性原理和具体实现,作了较为系统的叙述。以M6800在数字系统实验室建立调试系统的具体实现为例子,叙述了调试系统的

硬件和软件设计，并以流程图的形式作了详细说明。

作者希望不依赖昂贵的商业性开发系统，而是根据开发微处理机系统的实际需要建立适合自己使用的低成本调试系统。作者的这种想法，在我国微处理机应用和推广中是可取的。通过仔细地领会数字系统实验室建立调试系统的步骤和原理，读者可以顺着这个路子，根据实际条件建立实用而经济的调试系统。这样，读者不但为应用系统的开发建立了调试基地，而且会加深对微处理机的理解，使自己的设计思想更臻成熟。

由于译者水平有限，恳切希望广大读者对译文存在的错误和不妥之处予以批评指正。

译者

1984年4月

丛书编辑者前言

过去，计算机的性能是根据运算的速度、完成基准检测程序所花的时间和操作系统（管理程序）控制下运行一组混合程序所需的时间来衡量的。所有这些方法都没有顾及这样一个事实：大多数计算机通常在程序不断改变的环境里工作，即根据对计算机应用要求的改变和所需完成工作的改进，而不断地改变程序。

第一个主要意图旨在为设计师提供程序开发环境的，或许算是斯坦福大学的ARPA计划COPILLOT，但找到该报告是困难的，并且COPILLOT使用了非常昂贵的硬件。随着微计算机的出现，就有了不成熟的硬件/软件开发系统以及专门的显示方法。但是，它们之中很少有足够的用户指南，从未注意用人类工程学的观点去详细整理。

这本专著陈述在纽卡斯尔大学计算机科学系微计算机实验室所做的这方而的工作的重要细节。它表明了 in 程序开发期间控制和显示的可能性和不可能性，并且包括对各部件进行详细的硬件和软件的设计。我相信，此书对启发整个这方面的问题是有帮助的，并且对于想了解一个好的系统开发结构的任何人将是有用的。

F · G · 希思

作者前言

这本专著是根据1976~1977年在泰因河畔的纽卡斯尔大学计算机研究所, Eddie Farrell 和 Nik Kanellopoulos 在 Din Ghani的指导下所完成的实验工作写成的。

最初的目的是, 在计算研究所数字系统实验室的范围内, 为促进微处理机数字系统开发而进行技术研究。数字系统实验室(DLS)对数字系统设计的教学及研究提供了条件, 而硬件和软件在这中间起着重要作用。

在这一项技术研究工作中, 采用了两种不同的方法。一种方法是以常规的计算机控制面板的概念作为出发点, 问题在于如何为单片微处理机提供这样的面板装置, 以及为了得到较大的调试能力, 如何增强这些装置的功能; 为了便于调试, 另一种方法是探索用常驻调试监督程序开发存储器映象的字母数字TV显示灵活性的程度。

这本专著有两个主要目的。第一, 笔者希望说明不必依靠昂贵的商业性开发系统, 也可以在调试方面做一些事情; 第二, 笔者希望这里所提出的想法和提供的经验可以导至有相似想法的读者生成类似的或更好的系统。应记住, 本书所叙述的是在有限时间内实验活动的成果。

在写该书时, Nik Kanellopoulos和Din Ghani正在考虑实时环境的调试问题。

本书第四章和第五章所引用的报告成果是由Nik Kanellopoulos完成的。我们感谢他准许我们采用他的劳动成果, 以及编写这本专著时他给我们的帮助。这两章是根据他提交给计算研究所的理科硕士论文写成的。

第六章到第十一章是根据Eddie Farrell的理科硕士论文写成, 该论文也是Eddie Farrell提交给计算研究所的。由于计算

研究所使用了这份材料，编写本书时也采用了这份材料，因而该论文得到了肯定。

特别是Briam Bandel教授和John Givens积极地鼓励我们进行这一工作。还必须提及的是Mik Elpnick作为Eddie Farrell的主管所起的作用。在写这本专著中，Pete Lee、Paul Rautell-bach和John Aspden都给予了帮助。所论述的这些工作中，我们从Keith Heron和Vince Biltan的帮助中获益不浅。Kelvin Hanison设计和建立了TV显示系统的样机，该子系统样机作为全屏幕显示方法大有启迪。而Phie McLape、Mick Pgwel和Chris Boivin帮助生产出了更加轻便实用的系统。

我们感谢Mike Stoff为本书提供了照片。

Swan Hunter造船技师有限公司（英国造船业成员之一）给予了Eddie Farrell极大的支持。特别是R.I.Smith和其后的Alec Revel都给我们提出了合理的建议并给予极大的鼓励。SRC和Snan hunter通过工业奖学金给Eddie Farrell提供了财政支持。

最后，我们感谢Herriot-Watt大学的Fred Heath教授。近几年，他多方面参加我们的活动，为我们做了很多工作，使我们的工作得以沿着一条正确的道路进行。

N.甘尼

E.法雷尔

1979.8于泰因河畔的纽斯卡尔

目 录

第一部分 导 论

第一章 概述	(3)
1.1 调试问题	(3)
1.2 调试要求	(4)
1.3 低级工具	(5)
1.4 控制台控制	(6)
第二章 数字系统实验室的环境	(9)
2.1 导言	(9)
2.2 DSL硬件原理	(10)
2.3 DSL的M6800系统	(11)

第二部分 专用控制面板方法

第三章 硬件控制台	(15)
3.1 动机的形成	(15)
3.2 控制面板部件	(16)
3.3 M6800的前面板功能	(17)
3.4 M6800面板接口	(20)
3.5 硬件控制台的限制	(24)
第四章 增加了软件的硬控制台	(26)
4.1 一般原理	(26)
4.2 M6800增加了软件的控制台	(28)
4.3 缺点	(35)
4.4 为存储器复写的硬件扩充	(36)
第五章 为了灵活性而增加固件	(38)

5.1	导言	(38)
5.2	具有可编程能力的硬件组织	(40)
5.3	状态时序	(44)
5.4	微程序的生成	(50)
5.5	固件方法的优缺点	(51)

第三部分 全屏幕方法

第六章	导言	(55)
6.1	一般概念	(55)
6.2	目的	(55)
6.3	BEDBUG的组织	(56)
第七章	BEDBUG的硬件	(60)
7.1	输入端口	(60)
7.2	输出端口	(63)
7.3	专用硬件	(65)
第八章	软件结构	(69)
8.1	程序的组织	(69)
8.2	监督程序的工作	(71)
8.3	调试命令	(74)
第九章	表的管理	(79)
9.1	符号表	(79)
9.2	断点表	(84)
第十章	一般命令	(88)
10.1	显示命令	(88)
10.2	修改命令	(98)
10.3	输入/输出命令	(106)
第十一章	指令执行控制	(113)
11.1	控制技术	(113)

11.2	执行控制	(114)
11.3	IEC的实现	(117)
11.4	测试IEC软件	(129)
附录A	BEDBUG的NMI控制器	(131)
附录B	存储器保护控制器	(134)
附录C	BEDBUG命令快速参照表	(137)

第一部分

导 论



第一章 概 述

1.1 调试问题

大规模集成电路（LSI）技术的迅速发展和LSI器件的价格／性能急剧地下降，对现代世界产生了很大的影响。最明显的影响在于数字系统的设计方法；其次，是现今对于非数字问题采用数字解的可行性逐渐增加。这种可行性是由于可以利用便宜而具有极强可编程能力的微电子器件而得到。

然而，可编程能力本身也提出了它自己的问题。一个系统最重要的元器件就是大量的“1”和“0”的集合，这些“0”和“1”隐藏在微细的单片器件内，这就意味着要生产一个工作系统不是那样简单容易的。开发和调试（特别是软件部分的开发和调试）所花费的力量可能是LSI有效成本降低的主要障碍。

观察LSI器件内部动作很困难，这是调试中的主要问题之一。越是复杂的器件，困难就越大。即便是容易看到器件内部正在进行的动作，由于存储程序式计算机最基本的性质是很多观察到了的动作不是直接与整个系统功能（例如，取指令、中断内务处理操作）相关，这样，用已有的方法来解释观察到的动作就相当困难。

另一个问题是这种硬件很难象通用计算机系统那样可靠。原因是：①复杂元器件的激增使得它本身就难于测试；②每一个新的应用系统的某些部分（如果说不是整体的话）将重新组合的机率很高。所有这些常常导致难以区分硬件和软件故障。硬件故障可以引起软件的可疑动作，而软件故障又可以表现为一些硬件元器件的错误动作。

在调试以微处理机为基础的系统，其它的问题是：系统经常

涉及实时动作[7]，如果系统功能不是时间临界的话，也是依赖于时间的。所以调试工具的引入必须不与这些时间的相依性冲突。

本书侧重提及的第一个问题，即帮助用户观察系统内部，而不关心为软件开发[9,11]提供支持这类比较一般化的问题。本书只介绍在教学环境中提供低成本调试手段的想法和经验，而不做概括性的叙述。这个教学环境将在第二章叙述。它旨在表明：用少量专门化的硬件能够做些什么事情。

1.2 调试要求

这一节介绍调试工具应该满足的主要要求。但是，应该指出，本书中所叙述的工具不一定都完全满足这些要求。调试的主要要求如下：

(1) 监视

调试工具应该帮助用户容易看见测试下的系统(SUT)正在运行的情况，使用户能够检测到处理机寄存器和存储器单元的内容：检查程序代码、数据区、栈等等。信息显示应该是有选择的，而且是广泛的，并且所需的信息应该是以某种易于为人们接受的形式出现。

(2) 修改

用户必须能够修改系统状态。他可能需要改变某些寄存器或存储器单元的内容。例如，改变程序中的某一标志以强迫执行特定的一段代码。

(3) 控制

调试工具应该允许用户完全控制程序的执行。他应该能够在任意点开始和停止程序的执行，并且能够单步通过程序。根据他使用

的机器类型和语言，可以有不同级的单步执行。

(4) 透明性

对于试验下的系统，调试工具的引入不应该对系统有妨碍。这要从两方面来考虑：

(a) 静态透明性。静态透明性就是考虑：在引进调试工具时，要避免对SUT（测试下的系统）的硬件或软件做任何改动。

(b) 动态透明性。动态透明性就是考虑：引进调试工具后必须避免SUT的运行状态有任何意外的改变，它只能是用户命令的直接结果。

1.3 低级工具

作为最低级的工具，电压表和示波器仍然是微处理机系统调试中有用的工具。至少，它们可以被用来验证某些系统基本参数（比如电源的电平和时钟信号）的正确性。在检查外围接口和总线的相互连接中，示波器确实是有用的，但它不是单独地使用。一般说来，在这样的应用场合，要求示波器具有控制微处理机系统的运行和修改其状态的能力。例如，为了调试硬件接口，可能必须使微处理机执行一个试验程序，该程序应该以有规律的方式使接口运行。可以用示波器来监视接口电路的动作。如果没有试验程序，示波器也就不能完成任何有意义的监视作用。

如果把示波器看作是在时间域里工作的仪器，那么逻辑分析器〔13,14〕就可以被认为是数据领域的仪器。这样的仪器适用于面向数字数据的微处理机环境。逻辑分析器常常提供广泛的监视能力，包括以某个工作速度获得一定数量的信息，随后以各种形式显示这些信息的能力以及使用复杂的方法触发获得信息的器件的能力。

一个典型的逻辑分析器应该提供许多根并行的输入探针（比如

说16根)，这些探针可以由用户连接到系统中的任何一种信号。在微处理机的系统调试中，这些输入趋向于连接到地址总线的引线上（如果有足够多的输入，也可以连接到其他总线上）。输入可以依靠逻辑分析器异步地取样，或者能够从SUT获得取样时钟来取样。每一个取样的输入字被存储在一个快速的循环缓冲器里。

同时，所提供的某种形式的触发机构应根据数据比较而不是根据电压门限来工作。在送入的取样和一组前面板开关之间，通常应采用布尔比较形式，这常用来在某一个所希望的点停止取样，以便把最新的取样显示出来。有时提供额外的输入用来动态地接通和断开取样。这个特点可用于像滤掉操作数地址（只要SUT能够提供用于区别指令和操作数地址的信号）这样的场合。

显示格式可以包括以各种数制获得的字形成的简明表格，模仿通常由示波器显示的伪定时图以及“位图”显示（在位图里，每一个所获得的字被当作在两维矩阵中的一个点的坐标来解释），位图显示提供有趣的抉择。在16位机里，每次取样被当作 256×256 矩阵中的一个点来显示，结果就得到一个相互连接起来的点阵图案。从这个图里，可以一目了然地看出存储器访问的模式。虽然不太可能挑选出单个的地址，但是利用位图显示能够相当近似地观察处理机动作全貌，它可以帮助决定程序中需要更加细致地观察的地方。此外，还有更专门型式的仪器，这类仪器把从数据总线获得的信息当作指令和数据来解释，并以汇编的格式显示这些信息。

所以逻辑分析器提供了较完善的监视，但是仍然没有提供交互式的控制。在程序调试中，逻辑分析器是比示波器更为有用的仪器，但仍需要控制微处理机系统的其他方法。

1.4 控制台控制

在使用示波器或逻辑分析器这样的工具时，少量的控制台可以由操作员控制台〔8〕提供。按照传统方式，这样的控制台应该作