



[美] D. P. 西沃赖克 著
R. S. 斯沃兹

可靠系统的设计理论与实践

上册

科学出版社

内 容 简 介

可靠性是系统设计最重要的目标之一。本书全面地介绍构成高可靠数字系统中所涉及的各种问题。

全书分上、下两册。上册为可靠系统的设计理论,下册为可靠系统的设计实践。上册共分六章。第一章介绍基本概念;第二章探究了失效的机理、故障的表现及其分布规律;第三、四章全面介绍了可靠性设计中的各种技术,重点讨论容错技术;第五章给出了可靠性的评价标准;第六章通过建立费用模型,讨论了系统可靠性与费用的关系及综合平衡等问题;附录给出了编码理论和可测试性设计等补充材料。

本书可供计算机专业的大学高年级学生、研究生以及从事系统工程设计、半导体器件可靠性分析的工程技术人员参考。

D. P. Siewiorek, R. S. Swarz
THE THEORY AND PRACTICE OF RELIABLE
SYSTEM DESIGN
Digital Press, 1982

可靠系统的设计理论与实践

上 册

[美] D. P. 西沃赖克 著
R. S. 斯沃兹

袁由光 曹泽翰 等 译

刘志模 陈以农

陈廷槐 陈以农 校

责任编辑 黄岁新

科学出版社出版

北京朝阳门内大街 137 号

中国科学院印刷厂印刷

新华书店北京发行所发行 各地新华书店经售

*

1988 年 9 月第 一 版 开本: 787×1092 1/16

1988 年 9 月第一次印刷 印张: 25 1/4

印数: 0001—2,530 字数: 581,000

ISBN 7-03-000428-0/TP·25

定价: 9.60 元

译 者 的 话

随着计算机的广泛使用,计算机系统的可靠性愈来愈受到人们的关注。容错技术是建造高可靠系统最有效的方法。在国外,容错计算机已广泛用于国防、工业控制、病人监护和银行事务处理等领域。我国航空工业部计算技术研究所从1979年开始研制机载容错计算机。中国船舶工业总公司数字工程研究所与重庆大学协作,于1983年开始了舰载分布式容错计算机系统研究。天津第一石油化工厂在航天工业部的帮助下已将三模冗余计算机应用于炉温控制,使可靠性增加了几个数量级,赢得了工人人们的信任。全国第一届容错计算会议已于1985年10月在北京召开,第二届会议又于1987年12月在上海召开,并出有会议论文集。然而,由于缺少一本这方面的综合性书籍,容错技术在国内至今还是鲜为人知。因此,本书中译本的公开出版,将促进国内容错计算的教育、研究与应用的发展。

本书在国外也是第一本容错计算的系统专著。在此之前,美国各大学也只是汇总许多论文作为教材,本书出版后,美国容错计算的教育大为改观。本书从1984年起即被重庆大学作为研究生的教材,内容十分丰富,能将研究生引到科研第一线。

重庆大学陈廷槐教授与本书作者西沃赖克教授1980年就已认识,以后又多次在国际容错计算会议上见面。西沃赖克高超的学术见解以及作为IEEE容错计算技术委员会主席所表现出的干练令人钦佩,他十分关心中译本的出版,还特别为此写了前言,在此谨表谢意。

本书的翻译是许多同行共同完成的。上册中,前言由陈廷槐译,第一章由黄开源译,第二章由陈以农译,第三、四章由袁由光译,第五、六章由曹泽翰译,附录A、B、C由刘志模译,附录D、E由陈以农译。第一至六章由陈以农初校,附录由诸义新初校。下册引言、第九、十、十一、十二、十三章由杨孝宗译,第七、八、十六章由袁由光译,第十四、十五、十七、十八章由陈以农译,名词索引由杨孝宗初译,刘志模参加了引言和十四章的翻译。下册由诸义新校。陈廷槐对全书作了最后校阅。

尽管译校者们都是从事容错计算研究的,特别是杨孝宗、诸义新曾在卡内基-梅隆大学听过此课、陈以农在重庆大学讲授此课,然而由于内容很新、专门术语太多,加之译校者水平有限,错误在所难免,敬希广大读者指正。

中 译 本 序

1980年9月在陕西微电子研究所副所长黄敞教授和西安交通大学几位教授的安排下,七位世界知名专家组成的代表团在中华人民共和国陕西省临潼县陕西微电子研究所举办的一个为期四天的容错计算报告会上作了报告。有一百多位中国学者参加了这个报告会。作为代表团团长的我,对中华人民共和国在容错计算领域的强烈兴趣产生了深刻的印象。在这个报告会上进行了许多卓有成效的讨论并结识了许多朋友。在那段难忘的日子里,我会见了陈廷槐教授。报告会后不到两年,杨孝宗作为专攻容错计算的访问学者来到了卡内基-梅隆大学。

在本书(可靠系统的设计理论与实践)编写期间,容错计算已经明显地成为一个发展迅速,且具有商用潜力的领域。自1980年开始,容错计算的研究与发展步伐就加快了。美国计算杂志(IEEE 计算机学会的主要出版物)1984年8月号出版了容错计算专辑。Omri Serlin 在一篇文章里总结了不少于11个商业公司的产品经验。这些公司的主要产品是容错计算机。随着高性能微处理器的出现,它的性能是“超级小型计算机”的竞争对手,现在已有可能在单个箱体内组装有16到64个处理器的多处理器系统。将多处理器系统固有的冗余与当代的操作系统和容错概念相结合已产生出整个这种容错系统,其性能、可用性和从开始到失效的平均时间等均比超级小型计算机和主计算机要好几个数量级而成本只有它们的几分之一。

随着计算机的普及使用,原则上容错计算的重要性将会继续增长。我坚信中华人民共和国的研究人员和系统设计师将作出无数贡献并将把容错计算机设计的前沿向前推进。

D. P. 西沃赖克

1984年11月

作者简介

D. P. 西沃赖克 (Daniel P. Siewiorek) 于 1968 年在美国安阿伯的密执安大学获得电气工程科学学士学位, 并分别于 1969 年和 1972 年在斯坦福大学获得电气工程 (辅修计算机科学) 科学硕士和哲学博士学位。西沃赖克博士于 1972 年起在卡内基-梅隆大学的计算机科学与电气工程系任教。

在卡内基-梅隆大学, 西沃赖克博士协助开创并负责指导 Cm* 研究项目, 参加了美国陆军/海军的“军事计算机之家”研究课题, 而且还领导了 C.vmp 计算机的研制工作。

西沃赖克教授曾担任几个商业机构和政府机构的顾问, 而且还获得过好几种奖励。

他当前的研究方向包括计算机体系结构、可靠性模型、容错计算、模块设计以及设计自动化。西沃赖克博士还担任过 IEEE 容错计算技术委员会主席。

R. S. 斯沃兹 (Robert S. Swarz) 于 1967 年获得美国纽约大学电气工程科学学士学位, 1969 年获得伦塞勒尔工艺学院工程科学硕士, 1973 年获得纽约大学哲学博士, 并于 1981 年获得波士顿大学的 M.B.A 学位。从 1967 年至 1976 年, 斯沃兹博士在普拉特-惠特尼航空公司开发了专用测量设备系统。从 1976 年至 1981 年他在数字设备公司的研究开发部门担任某些职务。目前他受雇于普莱门计算机公司, 并在该公司的调查与高级系统部领导一个可靠性与可服务性工程小组。他还在沃斯特多科性工学院任教。

目 录

译者的话

中译本序

前言..... 1

第一部分 可靠系统的设计理论

第一章 基本概念..... 7

1.1 可靠性的重要性 7

1.2 数字系统的层次 8

1.3 系统寿命期的各个阶段 9

1.4 容错计算的特性及其定义 10

1.4.1 可用度 10

1.4.2 可靠度 10

1.5 制造阶段 11

1.5.1 设计成熟性测试 11

1.5.2 进料检验 12

1.5.3 工艺成熟性测试 15

1.6 运行阶段 16

1.7 拥有费用 17

1.8 模型系 18

1.9 可设计的参量 19

参考文献 20

第二章 故障及其表现..... 21

2.1 引言 21

2.2 故障的表现 23

2.2.1 物理缺陷 23

2.2.2 逻辑级故障的类别 29

2.2.3 系统级的抽象 29

2.3 故障的分布 32

2.3.1 概率复习 32

2.4 样本数据与数学分布的拟合 35

2.4.1 极大似然估计法 35

2.4.2 韦伯参数的极大似然估计 36

2.4.3 线性回归分析 36

2.4.4 置信区间 37

2.4.5 符合良度检验 37

2.5 永久故障的分布: MIL-HDBK-217 模型 40

2.5.1 寿命期测试和现场数据	40
2.5.2 永久失效数据的分析: 估计分布及其参数	47
2.6 自动失效率计算	52
2.7 瞬时错误和系统错误的分布	53
2.7.1 数据收集	53
2.7.2 图形化数据分析	54
2.7.3 参数的置信区间	61
2.7.4 符合检验	61
2.8 小结	61
参考文献	63
习题	63
第三章 可靠性和可用性技术 Steven A. Elkind	64
3.1 避错技术	68
3.1.1 环境变化	68
3.1.2 质量控制	71
3.1.3 元件集成度	75
3.2 故障检测技术	77
3.2.1 二模冗余	78
3.2.2 检错码	82
3.2.3 自校验、故障保险和失效-安全逻辑	101
3.2.4 监视计时器和超时	107
3.2.5 相容性检验和权力检验	108
3.3 屏蔽冗余	110
3.3.1 N 模表决冗余	110
3.3.2 纠错码	118
3.3.3 屏蔽逻辑	128
3.4 动态冗余	135
3.4.1 可重组的二模冗余	136
3.4.2 可重组的 NMR	140
3.4.3 后援备件	147
3.4.4 缓慢降级	151
3.4.5 重组	153
3.4.6 恢复	161
3.5 小结	166
参考文献	166
习题	167
第四章 可维护性和测试技术	174
4.1 生产阶段	175
4.1.1 参数测试	175
4.1.2 验收测试	177
4.1.3 可测试性设计	182
4.2 现场操作	186
参考文献	190

习题	190
第五章 评价标准 Stephen McConnel Daniel P. Siewiorek	191
5.1 评价标准概述	191
5.1.1 硬件评价	191
5.1.2 软件评价	196
5.2 模型技术	201
5.2.1 组合模型	201
5.2.2 马尔柯夫模型	235
5.2.3 系统可用性模型	264
5.2.4 建立冗余影响性能的模型	271
5.3 系统设计的综合分析	275
5.3.1 设计实例: PDP-8/e	276
5.3.2 实例分析	281
5.4 小结	285
参考文献	286
习题	286
第六章 财经考虑	295
6.1 引言和基本概念	295
6.1.1 定义	295
6.1.2 维护费用	296
6.1.3 用户拥有费用	298
6.2 现场服务概观和费用模型	300
6.2.1 维护费用模型	300
6.2.2 寿命期费用 LCC (Life-Cycle Cost) 模型	303
6.2.3 具有综合数据成分的 LCC 模型	307
6.3 结论	311
参考文献	311
习题	311
附录 A 差错控制的编码技术 D. T. Tang R. T. Chien	313
A.1 基本定义	313
A.1.1 冗余	313
A.1.2 源码	314
A.1.3 分组码	314
A.1.4 二元码	314
A.2 数字数据信道中的差错	314
A.2.1 传送与存储	314
A.2.2 源编码	314
A.2.3 调制与解调	315
A.3 差错源	315
A.3.1 差错统计	315
A.3.2 存储	316

A.3.3 信道模型.....	316
A.4 编码中的数学结构.....	316
A.4.1 线性分离码.....	317
A.4.2 多项式循环码.....	318
A.5 对编码与译码的一般要求.....	318
A.5.1 差错症候.....	318
A.5.2 条件极大似然译码.....	319
A.5.3 极大似然译码.....	319
A.5.4 最小距离译码.....	319
A.6 线性开关线路与移位寄存器.....	320
A.6.1 使用延迟算子 D 的多项式.....	320
A.7 编码器和译码器.....	323
A.8 差错控制码的功能分类.....	325
A.9 编码策略.....	325
A.9.1 差错检测.....	326
A.9.2 部分纠正.....	327
A.9.3 抹除.....	327
A.9.4 自适应编码方案.....	328
A.9.5 顺序译码法.....	328
A.10 某些差错控制的应用.....	328
A.10.1 数据通信.....	328
A.10.2 数据存储器.....	329
A.10.3 辅助存储器.....	329
A.10.4 数字多分支型差错控制.....	330
A.11 结束语.....	330
附录1 线性码的结构.....	331
附录2 多项式码的结构.....	332
附录3 求生成多项式的方法.....	333
附录4 特殊的差错控制码.....	337
附录5 循环冗余校验.....	343
参考文献.....	344

附录B 算术差错码: 在数字系统设计中应用的代价和效果的研究 Algirdas Avižienis

.....	345
B.1 码评价方法论.....	345
B.1.1 问题的范围.....	345
B.1.2 代价准则.....	346
B.1.3 效果准则.....	346
B.1.4 逻辑故障分类.....	348
B.2 二进制算术处理器中的故障后果.....	349
B.2.1 并行算术运算中的基本故障.....	349
B.2.2 二进制处理器中的重复使用故障.....	351
B.3 低代价以2为基数的算术码.....	352

B.3.1	算术差错码的实现	352
B.3.2	低代价校验算法	353
B.3.3	故障效果: 一次使用故障	354
B.3.4	故障效果: 确定性重复使用故障	354
B.3.5	故障效果: 非确定性重复使用故障	355
B.3.6	剩余码中的重复使用故障	356
B.4	多重算术差错码	357
B.4.1	多重低代价码	357
B.4.2	多重码的“混合代价”形式	359
	参考文献	360
附录 C	可测试逻辑设计理论和实践的最新进展 R. G. Bennetts R. V. Scott	361
C.1	引言	361
C.2	理论方面的进展	362
C.2.1	组合线路	362
C.2.2	时序线路	370
C.2.3	重复阵列	377
C.3	可测试逻辑设计的实践情况	378
	结论	382
	参考文献	383
附录 D	MIL-HDBK-217B 可靠性模型梗概	384
	参考文献	387
附录 E	MIL-HDBK-217C 可靠性模型梗概	388
E.1	217C 模型	388
E.2	217C 1 号公报模型	389
	参考文献	392

目 录

第二部分 可靠系统的设计实践

C. vmp.....	395
商用计算机.....	395
DEC.....	395
IBM	395
UNIVAC	397
高可用性系统.....	399
Tandem 系列.....	399
ESS 处理器.....	400
Pluribus	404
宇宙飞船和航空电子系统.....	405
FTMP 和 SIFT	409
参考文献.....	410
第七章 C.vmp 表决多处理器	411
7.1 设计目标	411
7.2 系统结构	411
7.2.1 实际系统的构成	411
7.2.2 表决器的工作方式	413
7.2.3 外部设备	415
7.3 处理器同步问题	415
7.3.1 动态表决控制	415
7.3.2 总线控制信号的同步	416
7.3.3 系统时钟	418
7.4 性能量度	419
7.4.1 处理器执行/存储器读取时间	419
7.4.2 磁盘访问时间	421
7.5 运行经验	422
7.5.1 运行历史	422
7.5.2 C.vmp 系统可靠性	423
7.5.3 联机维护	424
参考文献	425
第八章 VAX-11 系列 (VAX-11/780 和 VAX-11/750) 中的 RAMP	426
8.1 VAX 结构.....	426
8.2 原始的 VAX-11 的实现	431
8.3 VAX-11/780 的实现	434
8.3.1 内部处理机寄存器	436
8.3.2 ID 总线寄存器	440

8.3.3 主存寄存器	442
8.3.4 控制台子系统	445
8.3.5 微诊断和宏诊断	450
8.4 VAX-11/750 的实现	452
8.4.1 设计改进	452
8.4.2 RAMP 特性	458
8.4.3 处理机寄存器	460
8.4.4 主存寄存器	462
8.4.5 诊断和修复	464
8.5 小结	466
参考文献	468
第九章 系统/360-系统/370 通过程序设计实现恢复	469
9.1 引言	469
9.2 恢复管理的目标	469
9.2.1 功能恢复	470
9.2.2 系统恢复	470
9.2.3 系统支持的再启动	471
9.2.4 系统修复	471
9.3 用户所涉及的问题	472
9.4 机构的简要描述	472
9.5 I/O 设备/部件恢复机构	473
9.5.1 IBM 标准错误恢复过程	473
9.5.2 可选的用户书写子程序	474
9.5.3 联机测试系统	474
9.6 通道检验管理机构 (CCH)	474
9.7 I/O 恢复管理支持机构	475
9.7.1 APR	476
9.7.2 DDR	476
9.8 CPU/处理机存储器恢复机构	477
9.8.1 机器检验管理机构 (MCH)	477
9.8.2 系统环境记录 (SER0 和 SER1)	478
9.9 系统相关的恢复机构	478
9.9.1 系统再启动	478
9.9.2 检测点/再启动	479
9.10 错误记录恢复机构	479
9.10.1 环境记录编辑和打印实用程序	479
9.10.2 系统环境记录、编辑和打印程序	480
9.11 RMS/65 与操作系统的关系	480
9.12 系统/370 的几点考虑	480
9.13 结束语	481
参考文献	481
第十章 SPERRY UNIVAC 1100/60 的可用性、可靠性和可维修性	482
摘要	482

10.1 引言	482
10.2 1100/60 的 ARM 基本原理	483
10.2.1 以前的 SPERRY UNIVAC 1100 系列中的 ARM	483
10.2.2 1100/60 中的 ARM——通用方法	483
10.3 ARM 的具体实现	484
10.3.1 系统特征	484
10.3.2 故障检测	485
10.3.3 错误纠正	486
10.3.4 故障隔离	487
10.3.5 错误恢复	487
10.3.6 故障注入	489
10.3.7 维修	490
10.4 ARM 的评价	491
10.5 小结	491
参考文献	492
第十一章 容错计算系统	493
摘要	493
11.1 引言	493
11.2 系统结构	494
11.2.1 系统组装	496
11.2.2 互连	497
11.3 处理器模块组织	497
11.3.1 CPU	498
11.3.2 主存储器	499
11.3.3 动态总线	501
11.3.4 输入/输出通道	503
11.4 输入/输出系统结构	504
11.4.1 双端口控制器	505
11.4.2 控制器缓冲器的几点考虑	506
11.4.3 磁盘控制器的几点考虑	507
11.4.4 Nonstop I/O 系统设计思想	508
11.5 电源、组装、联机维修	508
11.5.1 进一步组装和联机维修的考虑	509
11.6 小结	509
一个“不停机”的运行系统	510
摘要	510
背景	510
系统概述	510
系统设计目标	511
统一的硬件/软件设计	511
操作系统设计目标	511
操作系统结构	512
进程	512

消息	513
进程对	513
系统进程	515
应用进程接口	515
初始化和处理器重新加载	515
操作系统的错误检测	516
参考文献	517
第十二章 局域 ESS 处理器的容错设计	518
摘要	518
12.1 引言	518
12.2 系统停机时间的分配和原因	518
12.2.1 硬件可靠性	519
12.2.2 软件缺陷	519
12.2.3 恢复机制的缺陷	519
12.2.4 例行操作错误	519
12.3 双重结构	520
12.4 故障模拟技术	522
12.5 第一代 ESS 处理器	523
12.5.1 No.1 ESS 处理器	523
12.5.2 No.1 ESS 的运行结果	525
12.5.3 No.2 ESS 处理器	526
12.6 第二代 ESS 处理器	528
12.6.1 No.1A 处理器	528
12.6.2 No.3A 处理器	530
12.7 No.3A 处理器的维修设计	531
12.7.1 通用系统的描述	532
12.7.2 通用处理器的描述	533
12.7.3 检测技术	534
12.7.4 恢复技术	542
12.7.5 诊断硬件	545
12.7.6 修复	547
12.7.7 硬件实现	548
12.8 小结	549
参考文献	550
第十三章 Pluribus——一个实用的容错多处理器	551
摘要	551
13.1 引言	551
13.2 Pluribus 体系结构	552
13.2.1 主要的设计决策	552
13.2.2 系统概述	553
13.2.3 实际系统的结构	555
13.2.4 冗余技术	560
13.3 Pluribus 操作系统	560

13.3.1	操作系统的一般职能	561
13.3.2	STAGE 系统的分层结构	561
13.3.3	建立通信	562
13.3.4	协同机构	563
13.3.5	与应用相关的检验	564
13.4	应用可靠性的一个例子	564
13.5	Pluribus 容错方法的优点	565
13.6	近期的现场经验	566
13.6.1	处理器总线上的失效	567
13.6.2	公用存储器的错误及丢失	567
13.6.3	I/O 设备的丢失	568
13.6.4	关键硬件的丢失	568
13.6.5	内部软件错误	568
13.6.6	人为的病态条件	568
13.7	Pluribus 系统可维修性	569
13.7.1	报告机构	569
13.7.2	远程诊断与修复	570
13.7.3	划分	570
13.7.4	重新加载和下行线加载	571
13.7.5	维护经验	571
13.8	其他应用及扩充	572
13.8.1	信息系统	572
13.8.2	实时信号处理	572
13.8.3	通用分时系统	572
13.8.4	预定系统	573
13.8.5	过程控制	573
	参考文献	573

第十四章 自检测和自修复计算机 STAR——容错计算机设计理论与实践的一个研究报告

	摘要	574
14.1	引言: 研究过程及基本原理	574
14.2	STAR 计算机的体系结构	576
14.2.1	容错的方法	576
14.2.2	硬件系统的组织	576
14.2.3	标准操作	577
14.2.4	计算机字: 格式和编码	578
14.2.5	控制错误的检测	579
14.2.6	功能单元的性质	580
14.2.7	检测和修复处理器 (TARP) 及恢复方法	581
14.3	可靠性分析的比较	582
14.4	STAR 计算机的软件系统	585
14.5	STAR 技术向外围系统的扩展	586
14.6	TOPS 控制计算机的设计	587

14.7 现行研究	587
参考文献	588
第十五章 “旅行者”飞船中的故障自动保护	589
摘要	589
15.1 引言	589
15.1.1 使命	589
15.1.2 飞船	589
15.2 达到的可靠性	590
15.3 故障自动保护设计	591
15.3.1 要求	591
15.3.2 要求的硬件实现	592
15.3.3 要求的软件实现	592
15.4 命令计算机子系统的功能描述	592
15.4.1 CCS 例行程序结构	593
15.5 故障保护软件	594
15.5.1 CCS 中的故障保护	594
15.6 设计验证	603
15.7 飞行中的经验	604
15.7.1 失效和降级	604
15.7.2 环境因素	604
15.7.3 序列错误	605
15.8 结论和建议	605
参考文献	605
第十六章 SIFT: 飞行控制容错计算机的设计与分析	606
摘要	606
16.1 引言	606
16.1.1 动机	606
16.1.2 背景	607
16.2 SIFT 的容错概念	608
16.2.1 系统概述	608
16.2.2 故障隔离	609
16.2.3 故障屏蔽	610
16.2.4 调度	610
16.2.5 处理器同步	611
16.2.6 可靠性预测	613
16.3 SIFT 硬件	614
16.4 软件系统	618
16.4.1 应用软件	618
16.4.2 SIFT 执行软件	618
16.4.3 故障检测	622
16.4.4 模拟器	623
16.5 正确性证明	624
16.5.1 概念	624

16.5.2 模型	624
16.5.3 可靠性模型	625
16.5.4 分配模型	625
16.5.5 今后的工作	628
16.6 结论	628
附录: SPECIAL 说明的实例	629
参考文献	630
第十七章 FTMP——一个用于飞机的高可靠容错多处理器	631
摘要	631
17.1 引言	631
17.1.1 背景与由来	632
17.1.2 FTMP 方法的基本原理	633
17.2 FTMP 的理论	633
17.2.1 标定组织	633
17.2.2 冗余组织	635
17.2.3 同步	639
17.2.4 失灵管理	640
17.3 FTMP 的一个工程样机的描述	643
17.3.1 冗余总线结构	647
17.3.2 LRU 与总线系统的对接	647
17.3.3 系统控制单元	648
17.3.4 主要故障限制区域	649
17.3.5 主电源	652
17.4 FTMP 的生存与分配概率模型	652
17.4.1 生存概率模型	652
17.4.2 间歇性故障的影响	656
17.4.3 FTMP 计算机的分配可靠性	659
17.5 实验结果	660
17.5.1 故障诊断能力	661
17.5.2 软件经验	662
17.6 结论	663
17.6.1 FTMP 设计的关键区域	663
17.6.2 小结	664
参考文献	664
第十八章 高可靠性系统的设计方法论——Intel 432	665
18.1 高可靠性系统的设计方法论	665
18.1.1 定义系统目标	665
18.1.2 限制范围	666
18.1.3 定义故障处理的层次	667
18.1.4 定义重组和修复边界	668
18.1.5 设计故障处理机构	669
18.1.6 识别硬核	669
18.2 工艺的影响	670