

数字计算机中的 错误检测逻辑

前 言

这本书是一部描述若干错误检测技术手册的扩充。这些手册是写给IBM公司的逻辑设计者使用的，把手册扩充为一本书是为了提供可以应用于数字计算机的一些错误检测方法，其目的在于能引起对涉及到计算机和专门逻辑设计有关问题的错误检测技术的兴趣。

本书可作为学习错误检测系统设计的教科书。同时，在各章中所讨论的计算机逻辑的专门范围内，它也将是一本关于检测错误方法的有益参考书。

书中的一些材料，作者在教学中曾使用过。予先具备一定的逻辑设计和布尔代数的知识时理解这本书的内容是必要的。此外，知道一些计算机设计的知识也是需要的。

各章是相互独立的。在阅读了第1、2和3章之后，就具备了阅读其他各章的教学基础，其他各章就可以任何次序阅读。

(略)

目 录

前 言

术语和符号	1
第一章 数字计算机中的错误检测	8
1.1 为保护用户的错误检测	10
1.2 着手重新启动的错误检测	14
1.3 为了矫正的错误检测	15
1.4 通过重新组合达到高可靠性的错误检测	18
1.5 失败型式假定	19
1.6 错误的种类	21
1.7 错误检测的理论状况	22
1.8 注释	24
第二章 数学基础	25
2.1 基本布尔差概念	25
2.2 用布尔差分析逻辑举例	28
2.3 求解布尔差的方法	31
2.4 二次布尔差	34
2.5 代码间距	36
2.6 全等和余数运算的基本概念	39
2.7 门限逻辑线路的简单介绍	40
2.8 注释。问题	43
第三章 一些错误检测代码	45
3.1 奇偶校验	46

3.2	剩余代码	50
3.3	定权代码	56
3.4	非法字符校验	58
3.5	代数错误检测代码	59
3.6	其他代码	65
3.7	注释、问题	67
第四章	错误校验线路	69
4.1	奇偶校验线路	69
4.2	定权代码校验线路和门限线路	76
4.3	门限线路	77
4.4	余数校验	87
4.5	注释、问题	96
第五章	数据通道校验	97
5.1	在数据通道中校验线路的安排	97
5.2	基于字或字节的校验	100
5.3	非法代码传输校验	102
	问题	
第六章	加法器错误特征	103
6.1	并行二进制加法器	103
6.2	加法器错误特征	108
6.3	注释、问题	113
第七章	奇偶校验加法器	114
7.1	半和校验	114
7.2	全和奇偶校验	115
7.3	用于加法器的奇偶校验	117

7.4	带有奇偶校验的双重进位 I	119
7.5	带有奇偶校验的双重进位 II	123
7.6	进位依赖于和的加法器	128
7.7	进位超前加法器	133
7.8	校验增量线路	138
7.9	二进制编码的十进制加法的校验	141
7.10	注释、问题	144
第八章	余数校验加法	
8.1	基本校验系统	
8.2	剩余代码的错误检测能力	
8.3	按字节运算部件的校验	
8.4	注释、问题	
第九章	各种校验方法的加法器	145
9.1	引言	145
9.2	校正的双重运算法	145
9.3	三电平和双路逻辑的错误检测	148
9.4	5 中取 2 代码反查表加法器	
9.5	二五进制码长加法器校验	
9.6	修正的反演二进制代码 (MRB 代码)	
9.7	MRB 加法器	
9.8	MRB 减法器	
9.9	注释、问题、附录	
第十章	除加法外的其他运算的错误检测	155
10.1	减法的错误检测	155
10.2	补数的错误检测	159

未译

未译

10.3	移位操作的错误检测	161
10.4	按位逻辑操作的错误检测	162
10.5	乘法的错误检测	163
10.6	除法的错误检测	173
10.7	注释、问题	
第十一章	计数器的校验	174
11.1	序	
11.2	奇偶校验的二进制计数器	174
11.3	环形计数器	} (未译)
11.4	定时计数器	
11.5	反馈移位寄存器计数器	
11.6	校验计数器的奇偶性的单位间距代码	
11.7	Kautz 单位间距错误校验码	
11.8	余数校验计数器	
11.9	用加法器作为计数器	
11.10	校验计数器和时序机	
11.11	注释	
第十二章	校验组合逻辑	112
12.1	错误检测的理论	} (未译)
12.2	错误检测的枚举	
12.3	校验组合逻辑的举例	
12.4	逐位网络中的错误检测	
12.5	用再生输入变量的错误检测	132
12.6	MLP 错误检测	} 未译
12.7	双向逻辑	

12.8	奇偶校验逻辑举例	} 未译	
12.9	注释、问题		
第十三章	时序线路中的错误检测		188
13.1	时序线路的错误检测		189
13.2	基本错误分析		190
13.3	同步时序逻辑的检验		192
13.4	异步时序逻辑检验		200
13.5	后 记		208
第十四章	记忆和存贮的错误检测		209
14.1	字地址的存贮的错误检测	} 未译	
14.2	磁带的错误检测		
14.3	延迟线记忆装置的错误检测		
14.4	磁盘和磁鼓存贮装置的错误检测		209
14.5	波形记录——(未译)		
14.6	用于线性反馈移位寄存器的		213
	错误检测		
14.7	注释、问题		
第十五章	系统错误检测问题		
15.1	关于系统任务的判定	} (未译)	
15.2	帮助检查校验的诊断		
15.3	关于避免错误陷阱的停机		
15.4	校验特殊线路		
15.5	控制错误信号		
15.6	两种逻辑校验范围的界线		
15.7	一个带故障机器的继续		

运行问题	_____	}
15.8 完成校验线路的测量	_____	
15.9 注释	_____	
文献	_____	
索引	_____	

术 语 和 符 号

本书介绍一些本书中所使用的专门技术术语和符号。其目的在于帮助读者了解一些不熟悉而作者经常用到的技术术语。在大多数情况下，这些术语在书内第一次使用时都给出定义。由于有几种标准和画逻辑的习惯不同，本书使用的图形符号在下面指出。

布尔差 (Boolean Difference)

布尔函数 $F = F(x_1, \dots, x_i, \dots, x_n)$ 对于 x_i 的布尔差表示如下。

$$\frac{dF}{dx_i} = F(x_1, \dots, x_i, \dots, x_n) \oplus F(x_1, \dots, \bar{x}_i, \dots, x_n)$$

位 (bit)

在二进制数中位取值 0 或 1

字节 (byte)

字节是二进制数的一个集合，例如由 4 个二进制数或 8 个二进制数组成一个字节。通常，一个计算机字由几个字节组成。

校验位 (check bit)

把一位或几位二进制数附加到一个二进制序列上组成一个新的序列，而在新的序列中的所有位遵从一个预先规定的法则。用这法则可以校验序列的正确性。

校验器 (checker)

校验器是用来校验一个代码或一个线路操作的正确性的一种线路。

代码 (code)

一个代码是一组以预先规定的法则组成的数字序列，而每一序列称为一

↑ 码字 (Code word),

1. 海明码 (Hamming Code)

一种由 R. W. Hamming 所设计的代码。一个码字中的全部单错误都可以被纠正。一组相关的代码纠正单错误并检测两个或偶数个错误。

2. 多项式代码或循环码 (Polynomial code or cyclic code)

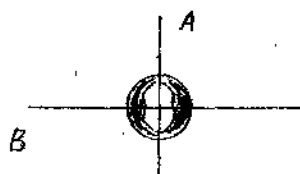
这种代码中的任一码字都可以表示为某一多项式 $f(x)$, 并且每一码字 $f(x)$ 都可以被一个公共的多项式 $g(x)$ 除尽。(a generator Polynomial $g(x)$)。

3. (n, k) 代码 ((n, k) code)

这种代码中, n 表示码字的长度, k 表示码字中信息位的个数。

冷子管或冷子元件 (Cryotron or Cryogenic Element)

冷子管是一种逻辑元件, 其逻辑功能类似一个密封的继电器。冷子管符号表示如下:



当电流通过导线 A 时, 便截断 B 导线中的电流, 当 A 中没有电流通过时, 电流可以流通 B。在温度接近于绝对零度情况下, 冷子管具有超导体的性质。

间距 (distance)

相同长度的两个二进制序列之间的间距等于二个序列对应位不同的位数。一种代码的最小间距是全部成对的码字之间的间距最小的间距。间距也被称为海明间距。

错误 (Error)

错误是指和正确结果不同的一个结果。

1. 固定错误 (Solid error)

是一种无限地持续下去的错误。

2. 瞬时错误 (Transient error)

在短期内发生并终止的错误。

3. 并发错误 (Burst error)

在一个二进制序列中几乎同时发生的几个错误。

异或 (Exclusive-OR)

异或是一种记为 $\oplus$ 的逻辑关系。它的定义为： $a \oplus b = a\bar{b} + \bar{a}b$ 。其逻辑图在关于符号那节中讨论。

故障 (Fault)

一个元件的一个或多个参数超出其特定的范围称为故障(元件)。

逆网络 L^{-1} (Inverse Network L^{-1})

网络 L^{-1} 是逻辑网络 L 的逆网络。如果 L 的输出是 L^{-1} 的输入，则 L^{-1} 的输出相同于 L 的输入。

卡洛图 (Karnaugh Map)

卡洛图是表示布尔函数的一种形式。图中一项对应输入到逻辑函数的每一种可能的状态的记录。如果逻辑函数是1，则对应于输入变量的状态的值记录为1。它也称为Vietch图。

维修 (Maintenance)

维修是矫正和调整执行一项继续原有操作的动作。

模 m (Mod m)

$y = x \text{ Mod } m$ 的范义为 $y = k \cdot m + x$ ，其中 k 为某些整数。

模 m 计算装置 (Mod m calculator)

计算输入数 x 被模数 m 除后的余数的电路。

模数 (Modulus)

使用于预先规定的情况下的数 m 称为模数。

MRF (修正的反射二进制代码。Modified reflected binary code)

表示相邻数字的代码字之间的间距为2的一种代码。它是奇偶代码的一种特殊情况,将在第9章中讨论。

奇偶位 (Parity bit)

如果数 N 表示如下:

$$N = \{a_0, a_1, a_2, \dots, a_n\}$$

则 N 的奇偶位 $P(N)$ 是

$$P(N) = \sum_{i=0}^n a_i \bmod 2$$

其中 2 与数 N 的数字的基数相同。在二进制系统中, $2=2$ 并且一个序列的奇偶二进制数是一个加到序列上从而使序列中的1的个数成为偶数的二进制数。一个奇数的奇偶二进制数意味着结果序列1的个数是个奇数。

奇偶校验线路 (Parity checker)

奇偶校验线路是这样一种线路即: 如果二进制数的序列呈现出不正确的奇偶性, 校验线路就指出一个有错误的输入。

只读存储器 (ROS)

只读存储器是一种不破坏读出存储器, 有时用于计算机的控制部分。

重新组合 (Reconfiguration)

应用于计算机中的重新组合是指改变计算机系统中各部件之间的相互连接。把故障部件排出系统之外后, 当正在修理时允许系统工作。

余数 (Residue)

一个数 N 的余数表示为 $R(N)$ 。 $R(N)$ 取以 m 为模的 $R(N) = N$ 中的最小的正数。

余数校验 (Residue checker)

应用于校验一个字的余数的线路。

重试 (Retry) (重新执行, 再执行)

在一个计算中, 返回到一个已知的正确的点并重新开始计算的功能。术语“重试”通常含有上述功能是自动控制的, 而当这种能包括了人工干预时, 则常常使用“重新启动”这个术语 (restart)。

存储器 (Storage)

在一个计算机系统中, 存储器是一些装置, 这些装置保存了按序访问的几个状态之一。同时, 它也是保存程序和指令信息的部件的名称。后者有时也称为“记忆 (memory)。

门限线路 (Threshold circuit)

每当条件:

$$\sum_{i=1}^n I_i w_i \geq T$$

满足时, 门限线路的输出总是 1。其中 I_i 是第 i 个输出 0 或 1 的值, w_i 是预先指定的权, T 是一个预先指定数。在第二章和第四章将较详细地讨论门限线路。

双向逻辑 (Two-rail Logic)

双向逻辑是一种二进制逻辑, 在那里, 每一个布尔变数用两种状态来表示, 而这两种状态总是相反。它有时也称为双轨逻辑。(two-wire Logic)。

权 (Weight)

权包含两种不同的概念:

1. 如果 $N = \sum_{i=0}^n a_i r^i$, 则 N 的第 i 位的权为 r^i 。

2. 在二进制序列中, 权表示 1 的总个数。

字 (word)

在计算机中，一个字是一个二进制序列。通常，规定有关那些算术指令的须序。例如，“加A到B”，一般，A和B都是字。对于随机存取存储器，字是存储在一个地址中的一组二进制数。

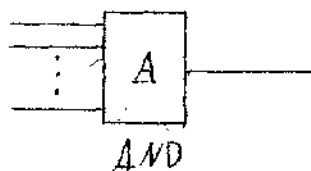
图形符号说明

本书中所用的一些图形符号及说明是按照美国标准
ASAY32.14 画的。这些不熟悉符号在下边作一些简短的规定。

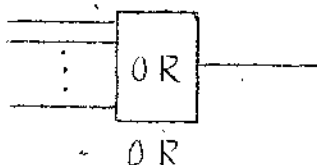


Time delay

一个信息进入时间延迟元件，要经过确定的时间周期之后，才有输出信号。



当且仅当所有的输入都假定为1状态时，“与”门的输出才是1状态。



只要有一个或几个输入为1状态，“或”门的输出就是1状态。



Invertor

当输入为1状态时，其输出相反为0状态，或者均反之。（本书中也用于同其他逻辑符号连用。）小三角是符号的组成部分。

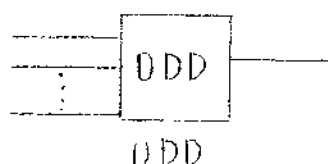


AND invertor

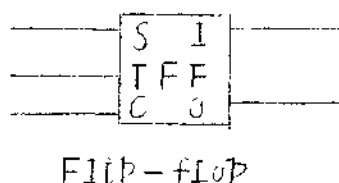
与非的输出类似于与门后边连接一个反相器。本书的所有逻辑框图，带有小三角形的输出都指明其输出是“正常”输出的布尔反向，否则，不带三角形。本书中，小三角形仅用于符号的输出端。



当且仅当其一个输入端为1状态时，“异或”门的输出才是1状态。

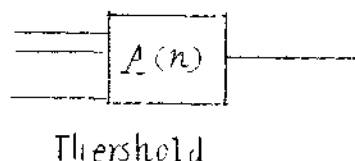


当且仅当其输入的1状态的个数是奇数时，“ODD”的输出才是1状态。如果它只有两个输入端，这个线路等同于“异或”线路。



把1信号加于S端，引起触发器处于“置位”状态。把1信号加于C端则产生“清位”状态。1信号加于I端，则反转触发器状态。在“置位”状态下的触发器，它的1输出端具有1状态信号而0输出端具有0状态信号。

在本书中，假定正好在I端的触发脉冲返回0状态之后，触发器的输出端改变。



当且仅当输入端1态的数目达到或超过函数符号的括号内所给的数目，其门假线路的输出才为1态。注意，这是与A S A标准不同的，在A S A中，不使用括号。

第一章 数字机中的错误检测

本章介绍关于在数字机中错误检测使用的情况。讨论由错误检测所实现的基本功能。同时还讨论各种故障类型的假设。有关错误的一些概念在各节的叙述中适当的给予回答。

错误检测 是检测信号的过程，这些信号不是一个正常运行的系统中应出现的信号。

一个失效 (Faulty Component) 元件是指这样的元件，它的一个或几个参数超出它的规定的范围之外。例如一个短路的二极管就是一个失效元件。故障 (Fault) 就是一个失效的元件。

错误是指不同于一个正常运行线路的标准输出的逻辑输出。值得注意的是，有些故障并不一定会引起错误，而只有一定的输入通过含有失效元件的逻辑部件时才会引起输出错误。

错误检测设备是用于检测错误的电路系统。本书中所叙述的错误检测方案主要是检测数字机和组合的数字装置的逻辑电路系统的错误。

书内所述的大部份技术属于硬设备方案。在1.1节中讨论了为保护用户的错误检测。同时也讨论了硬设备技术比起程序错误检测技术的一些优越性。

一些程序错误检测技术在计算机中已经使用 4.8 30、31、42、97、100。以我们的看法，硬设备技术较之程序技术的优越性是足以证明上述讨论是合理的。

在检测错误之后，一个计算机系统可以执行几种功能。其中有如下情况。

1. 给正在处理数据的用户一些讯号。这些讯号可以包含错误讯号，或者相反。使用户知道系统是在正常运行的讯号。

2. 由于错误而被中断的运算着手重新启动。

3. 为了较有效的诊断故障的情况。要给维修人员提供发生错误时机器状态的信息。在矫正之后，错误检测系统对检查机器也是有帮助的。

4. 以某种方式给系统提供错误信息，使机器本身能够重新组合。而在重新组合之后，直到失效元件被校正之前，系统将不再使用这一失效元件。这种能力是设计高可靠性计算机的基础。

在开始系统设计之前，要明确的规划。依照错误检查的结果而由系统所执行的功能。

在设计已经完成之后，而要系统提供所希望的功能往往是困难的。如果系统对错误响应在系统设计中是不清楚的，那么在完成系统方面，提供的希望的系统对错误的必须的先决条件是不够充分的。

应该强调的一点是，如果在错误被检测之后，用户或系统没有能力采取适当的措施，那么错误检测系统等于没有起作用。

反之也对，如果错误检测系统不能够胜任引起适当的响应即使系统对错误的处理具有相当的能力也只有很小的价值。

例如，如果错误检测的目的是为了从某一点重新启动被错误所终止的操作，而这一点恰好处在错误发生之前，这样，重新启动系统就需要关于系统处在错误发生时的操作点的信息，以及由于错误的结果而被破坏了的数据的信息。如果设计排除提供这些信息，那么上述要求的功能就不能实现。因此，在设计错误检测中一定要实际地达到功能的主要目标。作为一个基本设计的要求，上述所讲的都是极重要的。

设计指导(Design Guidelines)

1. 错误检测系统所具有的检测错误的能力和对错误的正确的响应至