CFTC－5

第五届
全国容错计算会议
论文集

1993.8.10-12 北京

主办单位：
中国计算机学会容错计算专业学会
中国空间技术研究院北京控制工程研究所

会议主席致词

经过一年多的筹备，中国第五届全国容错计算会议(CFTC-5)今天开幕了，在此我以大会主席的身份向所有来参加会议的各位代表表示热烈的欢迎。

容错计算技术是构成高可靠计算机的重要手段，虽然此技术的发展是在五十年代，但真正得到重视是在七十年代以后，至今亦已有二十多年的历史，因而在实际的高可靠计算机及复杂的电子设备中得到应用，从而使它们的有效使用率得到大大的提高。

这次会议的主题侧重于容错系统的应用，因此对容错在各种领域中的应用，其现状与未来，系统的设计与应用，特别是新的分布式网络系统结构，可降级的容错系统，以及在实际应用中对系统的容错性能的评价方法，故障注入和其它测试技术，系统的故障诊断等均加以讨论和交流，希望对今后在这一领域能够进一步向纵深发展。

这次会议是由中国计算机学会容错计算专业学会，和中国空间技术研究院北京控制工程研究所主办，并且得到了许多公司和研究所的支持。由于程序委员会成员的认真工作，会议得以今天在这里顺利开幕，我在此向他们表示真挚的谢意。

祝大会胜利开幕！

张翰英

CFTC-5会议主席

1993.8

General Chairman's Opening Speech

After more than one year's preparation, the 5th National Conference of Fault-tolerant Computing(CFTC-5) begins today. Please allow me, on behalf of the working group of CFTC conference extend a warm wellcome to all the participants of this conference.

Fault-tolerant computing technique is an important means to build a high reliable computer. Although the development of this technique began in the fifties, it was paid attention by scientist and engineers only after seventies, and up to now it has got more than twenty years wide application in building high reliable computers and complex electronic facilities, and then it increases the effective availability of the computers.

The main theme of this conference is application of the Fault-tolerant Systems. Therefore we paid more attention to the topics of the applications of Fault-tolerant techniques in different areas, the present status & prospects of FT; the design & applications of FT systems, particularly to those new system archietectures such as distributed systems & networks, degradable FT systems; the evaluation methods of the actual FT system, fault- injection techniques & other testing methods and fault diagnostic methods of the FT system etc. We hope that we can have vivid discusions in every area and they will promote our FT technique to a higher level.

This conference is orgnized by the FT Technological Committee of the China Computer Association and the Beijing Institute of Control Engineering of the China Academy of Space Technology, and it supported by many companies and institutes. Due to the careful works of members of the program committee, this conference holds the Opening ceremory today, therefore I would like to express my sincere thanks.

Congratulation to the conference!

Zhang Han-ying
General Chairmen of CFTC-5
1993.8

第五届全国容错计算会议程序按排

10日上午：全体会议

地　　点：大会议室

8：30——9：00　大会开幕式

主持人：张翰英

9：00——11：30　大会专题报告

主持人：张翰英

9：00——10：00　专题报告Ⅰ：容错在航天领域中的应用

报告人：梁思礼

何国伟

10：00——10：15　大会休息

10：15——11：30　专题报告ⅠⅠ：容错计算技术的现状与未来

报告人：闵应华

徐拾义

10日下午：分组报告

1：45——3：30　容错系统设计

主持人：王　镭

地　点：第一会议室

准三模冗余的容错计算机系统结构　　周志邦　员春欣　邓肃然

非相似余度容错计算机NFCS设计　　林　坚

具有通道自治能力的四余度容错机设计　　金惠华　郑　雄

高可靠A/D、D/A容错接口设计　　王英利　杨士元　童诗白

3：30——3：45　休息

3：45——5：15　故障注入技术

主持人：童诗白

地　点：第一会议室

故障注入技术初探　　田莉蓉

基于故障注入的容错机验证开发技术　　尹　彤　金惠华

故障注入方法及故障注入器　　吴智博　杨孝宗　陈继兰　刘柏岩

1：45——5：15　测试技术Ⅰ

主持人：李锦涛

地　点：第二会议室　（中间休息15分钟）

功能块级电路测试产生与故障模拟中的一些基本问题 魏道政
电路结构上一些特殊点和测试难易程度 杨志娟 梁业伟
临界路径跟踪--组合电路测试生成的一种算法 石 茵 魏道政
开关级差分及其在CMOS电路测试生成中的应用 胡 谋
A Global Design of Testbility Algorithm 晓 东 魏道政
Critical PathSensitisingFunctional FaultsCollapsing Algorithm
Peng Weidong Lu Zhenhua

11日上午：继续分组报告

8：15--10：00 软件和网络容错

主持人：郦 萌

地 点：第一会议室

实时系统应用软件多样化容错设计 刘贵春 高 远
OOP程序设计方法及其在高可靠性软件设计中的应用 刘聚强 杨孟飞
一类容错网的特征分析及系统控制 冯斐玲 金林钢
物价网络信息容错计算机系统设计 代小红 李 庄

故障诊断和恢复

主持人：梁业伟

地 点：第二会议室

On Searching Strategy in Fault Diagnosis Expert System
Shiyi Xu Jianhua Gao Zheng Lei
分布式系统中的非预设恢复方法 许再越 吴荣泉 谢玉和
分布式系统中的断点释放算法 许佑辉 朱育清 王加红
The Minimum-Size Fault Diagnosis of Multiprocessor Systems
Xiaofan Yang Tinghuai Chen Liuding Zhou
ANeuaralNetworkMethodfortheFaultDiagnosisof Multiprocessor Systems
Xiaofan Yang Thighuai Chen

10：00--10：15 休息

10：15--11：30 建模与性能评价

主持人：袁贤铭

地 点：第一会议室

可降级系统性能可靠度的一种近似计算方法 田志宇 杨士元 童诗白
变化中的故障安全概念 员春欣
超立方体结构失效模型的可靠性研究 陆远明 陆建东

检错技术与编码

主持人：盛运焕

地　点：第二会议室

变长检错码　　张焕国　覃中平

K/2K码最小码字集合的研究　　许　波　孟永炎

ASIC芯片设计实例:检错纠错动态存贮控制器　　沈　理

11日下午　继续分组报告

1：45——　3：30　分布式系统

主持人：杨樱华

地　点：第一会议室

A Fault-tolerance Distributed Application Devolopment Environment

C.J Chen T.C Tsay S.N Yuan

990 DFTCS分布式计算机容错光纤网络系统的构成和实现　　韩　威　袁由光

分布式容错拓朴结构的可靠性分析与计算　　陆建东　周咏梅　陆为国

具有容错重构能力的多机系统的研究　　王申科　谢克嘉　郑守琪

3：30——　3：45　休息

3：45——　5：15　容错系统应用

主持人：王东盛

地　点：第一会议室

一个用于飞行控制的容错计算机和系统的设计与实现　　臧红伟　谢克嘉

数字式电传飞行控制系统容错计算机余度管理研究　　石立　王占林　姚一平

飞行控制计算机系统设计　　赵　刚

1：45——　5：15　测试技术II

主持人：魏道政

地　点：第二会议室　(中间休息15分钟)

时滞测试综述　　闵应华

可测试模拟集成电路的设计　　赵国南　郭裕顺

实现动态功能自测试的设计结构分析　　李晓维

分步旁路扫描设计　　黄维康　张美玉

程控数字交换机PCB的模块化测试　　李　立　徐星宁

A Testing System Based on Concurrent Mutiple-process Control

Xia Ling Wang Limin

目　　录

准三模冗余的容错计算机系统结构

周治邦　贡春欣　邓肃然
(上海铁道学院电信工程系,上海,200333)

摘要　本文介绍一种准三模冗余的容错计算机系统结构。系统的控制核心是双重微型计算机 X_1-X_2,它承担全部的控制任务。此外附加了一个模块M执行人机对话等功能。M不承担控制任务,只是在核 X_1-X_2 故障时才参与仲裁。由于M不是专为容错目的设置的,与标准的TMR结构相比,它节约了部分硬件,结构更加简单;而与双重系统相比,则极大地改善了系统整体的故障定位功能。

关键词:　准三模冗余,比较器,控制核心。

一、引　言

在重要场合下使用的计算机控制系统普遍地采用容错技术,这种技术可以明显地改善系统的可靠性能,使得系统即使在故障状态下仍能维持其全部的或部分的控制功能。如果撇开备用模块,容错计算机系统通常都采用N模($N\geqslant 2$)冗余结构,最通用的是双重或TMR结构。双重系统的特点是结构简单,易于进行故障检测,但故障定位困难,因为两个模块无从表决仲裁出故障块。而TMR系统的故障检测和定位都比较容易实现,但结构大为复杂,所有的资源都必须是三重的。

本文介绍一种准三模冗全的容错计算机控制系统,特点是:

①由两个微处理机 X_1 和 X_2 组成双重的控制系统核心。

②附加一个监控模块M,M不参与控制,仅承担次要的任务(如人机对话等)。一旦双微机核 X_1-X_2 发生故障,由M执行仲裁。

这种结构的优点是控制核心 X_1-X_2 不承担占时较多的次要任务,结构紧凑。而监控模块M则不承担控制任务,因此输出控制有关的全部电路免除。M只是核心故障时才参与仲裁,在此期间可能丢失的次要任务不致影系统的控制功能。由于M不是仅为容错目的设置的(稍为复杂的控制系统一般都设置一个处理模块为人机对话等功能服务),所以并不太多增加系统的开销。与双重系统相比,它具有良好的故障定位功能;而与TMR系统相比,则结构大为简化,系统的软硬件设计也较为单一。

二、准三模冗余系统结构

图1是准三模冗余系统的基本结构。

系统由三个基本模块组成：M、X_1 和 X_2，X_1-X_2 组成系统的双微机核。X_1-X_2 可以根据需要采用合适的同步方式，例如严格同步方式(机器周期或指令级同步)，或者松驰同步方式(校验点同步等)。在采用严格同步方式时，部分比较器可以省略，通过输出反馈由软件完成数据校验。在需要故障安全的场合，X_1-X_2 仍然可以采用标准的容错结构，只是需要附加 FS 输出口。由此可见，图 1 的结构非常适合于模块化设计。

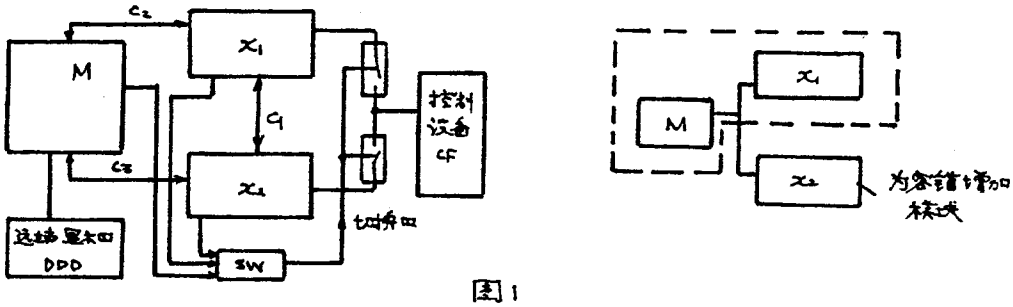


图 1

M 是监视器，它并不参与现场控制，所有控制全部由 X_1-X_2 承担。M 只是承担一些次要的任务，如人机对话，显示打印等等。M 的设计应该使其内部也形成一个核，使得显示器、磁盘等故障时，不致影响 M 执行 X_1-X_2 的仲裁任务。这个核实际上只是由 CPU，RAM，ROM 等部件及输入口组成，因而结构非常简单。

在正常运行的时候，X_1 和 X_2 产生相同的结果。如果单一模块发生故障，通过执行数据比较，可检测出故障。此后，M 参与仲裁，以定位故障，并根据需要切除故障单元(三取二表决)。M 自身故障并不影响系统整体的控制任务；而在 X_1 或 X_2 故障时，双微机核退化为单机，但输出结果的正确性仍可请求 M 校核。此时 M 必须丢弃一些次要任务，以跟踪退化了的单机运行。

为了实现故障检测和定位，X_1，X_2 和 M 之间设置了通信口，以交换数据。邮箱是比较合适的构件。但设计实战表明，在松驰同步方式下，专用的邮箱会增加电路的复杂性。一种比较合理的解决办法是在内存贮器中划出一组存贮块，通过中断机构，将单元 U_i($U_i \in \{X_1, X_2, M\}$)传送的通信数据存入这组存贮块。在中断服务程序中，这些通信数据并不处理，它们只是在指定的存贮点上处理，这种方式使得中断服务处理极为简单，现场也不必全部保护，而中断响应的时间占用问题则由提高处理器钟频改善。

图 1 结构的最大特点是 M 并不是专为容错目的设置的，一些控制计算机系统都设置有这样的模块为人机对话等任务服务。与实际的单机系统相比，图 1 的结构只是增加了单元 X_2 及少量的附加电路，而实际实现的功能则接近于 TMR 系统。

三、系统结构的一些细节

(1)比较器

比较器可以采用一般小规模集成电路数值比较器，例如 74LS85。在松驰同步方式时，一种比较简单的方式是将比较器 cm_1、cm_2(图 2(a))的输出反馈至 $X_1(X_2)$的并行输入口，通过软件检测比较的结果。由于使用了两个比较器，单一比较器故障不会影响运行结果校验的正确性。在严格同步方式时，比较器输出可连接到系统的中断线上(图 2(b))，在比较不一致时，由中断服务程序完成故障处理。为了避免两个比较器同时失效，控制核 X_1-X_2 应周期性地对比较器进行校验，以检测比较器是否发生了故障。

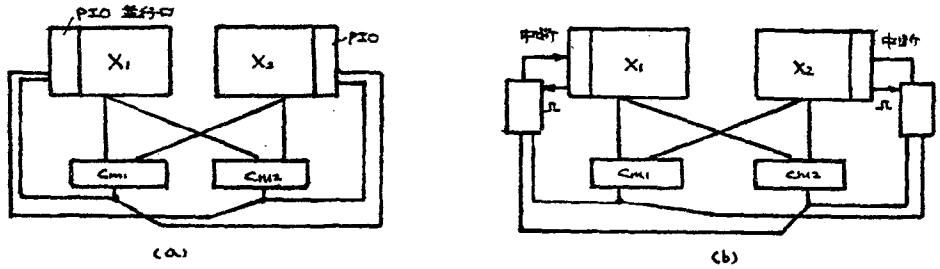


图 2

(2)时钟

时钟是系统的比较重要部件。松驰同步方式时，X_1 和 X_2 分别使用独立的时钟。但在严格同步方式时，必须采用双同步时钟，而 M 使用另一个独立的时钟源。双同步时钟的实现取决于实际使用的系统。一种可行的方式是采用上升沿同步，如图 3 所示。

图 3 在边的时序图晰地表明了同步的进程。这种方法的缺点是 φ 的有效高低电平宽度是变化的。因此要精心选择钟频，使得高低电平宽度变化不超出系统要求的限度。此外，这种时钟不适合于驱动实时时钟，因而系必须为实时时钟另设钟频器。

(3)基本的容错协议

在下面的讨论中，假设核 X_1-X_2 采用松驰同步方式。

在系统的程序流中，设有三类校验点 $Ch^i(j)$，$i=1,2,3$。i 是块验点的类型；$j=0,1,2,\cdots$，j 是每一类校验点的序号。三类校验点在程序流中的分布示于图 4。

在下列容错协议中，TYPE(ch)指校验点的类型 i，故障假设是系统中只存在单一模块故障。

```
while (tr=trve )
  begin CYCLE
    while (NOT ch_i)
```

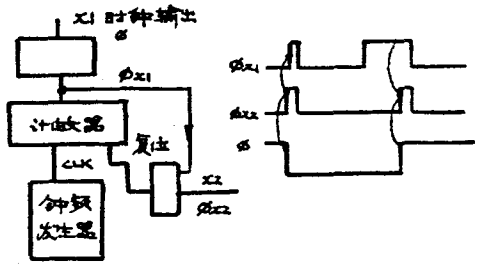
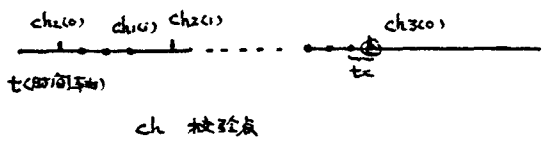


图3



ch 校验点

图4

begin EXE

M：记录随机的输入数据

X_1 或 X_2：执行校验点间的程序段

end EXE

if (TYPE (ch)=ch_i) do:CH1

else

begin

if (TYPE (ch)=ch2) do :CH3

else do:CH3

end

end CYCLE

CH_1：X_1 或 X_2： 等待同步标志并执行数据校验

if (异常或错误发生)

begin

请求 M 仲裁

在不可恢复故障时跳出 CYCLE 循环并作故障处理

否则故障恢复

end

RET

CH_2：X_1 或 X_2：do CH1

M:等待同步标志

更新全部新生成的非输入数据

RET

CH3：M:执行 tx 段控制程序,执行结果传送给 X_1 和 X_2 仲裁

X_1 或 X_2：do CH1

M：等待同步标志

更新全部新生成的非输入数据

RET

上述容错协议中，CH1、CH2、CH3，仅适用于正常运行至首次故障发生的时间段。在某个单元(X_1，X_2 或 M)失效后，CH1、CH2 或 CH3 需作修正。在需要较高可靠性能的场合，应该撤消 CH3 校验点，而全部 CH2 校验点由 CH3 替代。

四、实验说明

本文介绍了准三模冗余系统的结构和特点。采用这种结构的微机控制系统正在实验中。由于实验样机是低速的实时系统，全部省略了比较器，由程序执行校验算法，然后选择校验结果广播给其它两个单元。第三节介绍的基本容错协议曾在 SUNWORK STATLON 上用 C 语言模拟过(小输入数据样本)，证明算法是可行的。模块 M 除了执行一些较为繁重的次任务外，在 X_1 或 X_2 故障时，还承担仲裁任务。因此 M 的 CPU 应采用较高档次的微处理器芯片，并适当提高系统的时钟钟频，以改善响应时间。

参 考 文 献

[1] D. P. Siewiorek, "Fault Tolerance in Commericial Computers",Computer, , vol. 23. No. 7 ,July 1990,PP26—37

[2] P. K. Lala Fault tolerant and fault testable hard ware design , Prentice Hall International, Inc, London ,1985

非相似余度容错计算机系统NSFCS设计

林　坚
(航空航天部航空计算技术研究所，西安，710068)

摘要　在现有的相似余度容错系统中，容错的基本方法是采用资源重复的技术，即硬件和软件的资源重复，这种容错方法可以有效地避免系统中出现的非共性的随机故障，但对于共性故障，如硬件设计或软件编程中的错误，这种方法会导致系统崩溃。非相似余度系统研究目的在于克服系统中的共性故障，使之保持正常运行。

关键词　非相似余度，N-版本，同步，交叉检查，版本重构。

1、非相似余度计算机系统

非余度计算机或单机系统的特性是，一个程序(单个软件1S)，在一个硬件通道上(单次硬件1H)运行，一次执行(单次时间1T)，简计为1T/1S/1H，对于这样的系统，一旦出现一次软件或硬件的故障，则将引起系统的失效，对于一些安全关键系统，这是不能允许的。为了提高系统的可靠性，应从软件硬件功能设计上采取措施，通常为解决硬件随机故障，采用了余度技术，即同一个程序在N个相同的重复的硬件通道上运行，这种方法一般记为(1T/1S/NH)。这对于提高硬件可靠性是非常有效的，但如果在一个容错系统的实现过程中，存在下列任何一种错误，则用简单的资源重复的系统是达不到容错的目的的。

a. 系统规范错误；
b. 软件规范错误；
c. 硬件规范错误；
d. 软件设计错误；
e. 软件实现错误；
f. 硬件设计错误；

以上的任何一种错误，都会导致在同一个时刻，对同一个操作对象或工作单元产生相同的故障结果。所以用一般的各个余度之间交叉通道监控的方法来检测隔离这种错误是不可能的。而且系统本身会因为各个结果相同而认为所有的余度都是正确的，从而可能导致灾难性的后果。

非相似余度系统设计的基本思想是采用非相似的N-版本(N>2)的硬件和软件，即采用多个版本程序运行在多个硬件通道上，以实现对软硬件故障的容错。N-版本系统称之为(1T/NdS/NdH)系统.

2、非相似三余度计算机系统(NSFCS)的设计

NSFCS是一个N-版本(N=3)的非相似余度的容错计算机系统，它由三个独立通道的计算机FTP1、FTP2、FTP3构成，每个通道使用非相似的硬件实现，且使用非相似软件来编程。系统的总体方案原理图如下图1所示：

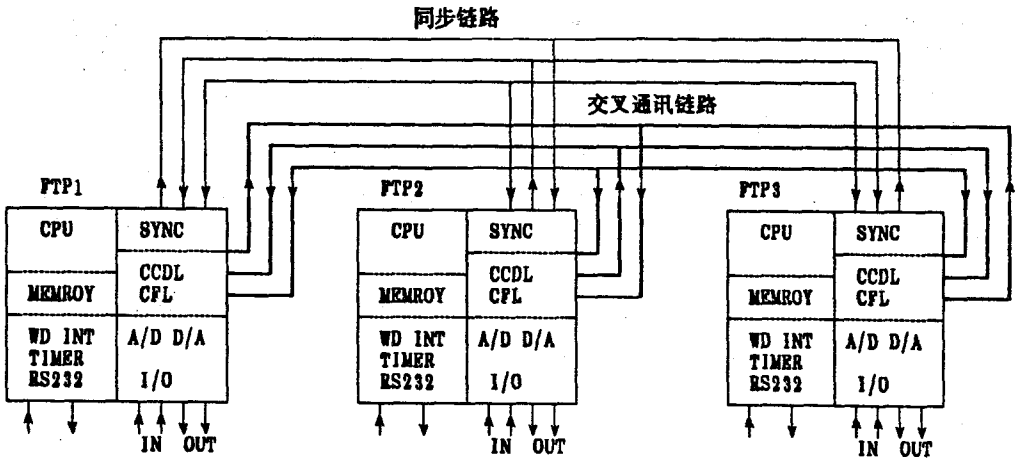


图1 NSFCS硬件系统框图

NSFCS硬件中采用N=3个不同系列的32位中央处理机芯片，构成三余度非相似容错计算机系统；这样由于三个CPU 的指令系统不同，在指令这一级上避免相同机器指令的共性故障；另外，由于三种CPU的结构不同，设计中分三个组分别进行，因而从使用的芯片到计算机的结构也不尽一样，用非相似性保证硬件系统中的共性故障发生的概率达到最小程度。

软件系统同样也采用多版本设计，除需求规范以外，软件分三个组分别对应三个计算机进行多版本的完全独立的相异性设计，即：软件概要设计采用三个版本设计，软件详细设计采用三个版本设计，软件编码采用三个版本设计，且编程语言采用三个种不同的语言；软件编程使用不同的语言，这不仅由于语言的不同而带来系统的非相似性，避免了编译和连接系统错误引起软件共性故障，而且因为各个语言的特性强迫程序员朝不同的实现方法努力，从而使软件的共性故障率大大降低。

三个通道的计算机中央处理机和所使用的语言为：

Intel 80386+80387/ADA;

Intel 80960/C;

Inmos T800/OCCAM;

三个通道之间用同步链SYNC、交叉通信链CCDL控制完成通道间同步运行和数据的传输，I/O用于控制外界信号的输入/输出，每个通道都自带有独立的电源系统。

3、通道计算机FTPi的组成

非相似余度计算机包括三个通道的数字计算机．每一个通道包括二个模块，即中央处理机模块(CP)，输入输出模块(IO)，其中CP模块控制系统的IO模块，实现计算机管理，余度管理，自检测，应用程序的功能，．IO模块完成系统各个模拟量到数字量的转换，完成数字量到模拟量的转换，离散量输入和输出，同步功能电路交叉通道通讯及通道故障逻辑．

1 CP 模块

中央处理机CPU 管理和控制整个通道的资源；存储器中包括程序存储器EPROM，数据存储器RAM和非易失存储器NVM，NVM是用于保存系统中的一些有关故障信息，如加电BIT结果、故障原因和系统恢复参数等；看门狗定时器是用于故障分析的一个手段；看门狗定时器的时间要在系统初始化阶段设置，以后看门狗将定时器应需要定时刷新，否则看门狗将发出中断，指示系统出错；中断控制器、定时器是为系统提供必要服务功能；RS-232通讯接口是提供使用PC-386 调试与开发FTPi软件的。

2 IO 模块

IO 模块中，同步链路SYNC是用于控制系统的同步运行，同步链路使用全双工的通信结构向每个FTPi提供出三个离散量位，称之为同步指示器，它用于登记和驱动本地通道向其它通道发送同步信号，以及登记和接收其它通道发来的同步信号；每个FTPi通过对同步指示器置1和置0 完成三个FTPi之间的双握手同步；

交叉通道链路CCDL完成三个通道相互之间通过点对点高速串行数据通路完成信息的传输，每个FTPi通过交叉通道独立地广播式发送数据到其它FTPi，每个FTPi可以独立地连续地接收从其它FTPi发送的数据，在每个FTPi中都有一个故障表，通过故障表，FTPi可以对故障的FTPi所发送的数据不予理睬，而故障的FTPi可以连续地广播发送数据；

通道故障逻辑是根据本通道的在线监控和通道之间的相互评价来判断本通道和其它通道的有效性，并根据硬件多数表决控制本地通道的输出，完成本地通道的故障隔离。

A/D、D/A和离散量I/O是提供用于控制应用对象的接口。

4、软件

1 N-版本程序

N 版本程序设计能够容错的基本原理基于这种假设：即相互之间独立形成的软件版本出现相关软件错误的概率非常小，在同一时刻出现相关的软件故障的概率极小，其它软件故障可以用交叉通道监控、检测、隔离直至屏蔽；这样就可以降低甚至消除软件共性错误。

同一般软件不同，N-版本软件需要特殊的支持机理，它们是：同步机制、交叉检测点个数和位置、交叉检测向量的格式和内容、版本间的通讯、表决算法、表决门限及版本中任务的调度和故障版本的处理；

2 N-版本 (N=3) 软件的结构

FTPi 通道计算机系统软件包括：支持软件，计算机BIT，执行软件，余度管理；

支持软件主要是提供FTPi的监控程序和软件的调试开发环境；

计算机BIT是用于在使用系统前对FTPi进行的自检测；在加电阶段，要完成全部的BIT，将其结果存储在NVM中，同版本的故障历史一起，建立通道可用状态，为系统余度管理提供必要的信息；

执行软件包括系统的初始化，硬件资源的管理，任务的调度，输入/输出处理，同步和交叉传输；系统的任务的调度是可以根据不同任务的实时性要求来按排，但要使各版本都产生同一时刻的交叉通道检查向量。

同步机制是非相似系统中的关键因素之一，一般在非相似容错系统中，同步机制主要有二种方式：帧同步，事件同步；帧同步是将任务分在几个帧中，系统对任务定时实行周期性调度及有规律的同步时钟，这种同步技术是使用一个时钟来预测输出结果可以比较的时间，它不允许执行时间的不同，即不能预计通讯中的延迟，优点是可以减少同步开销；一般事件同步是选诸如输入，输出，中断，例外，多版本之间通讯等为主，版本间使用同步协议保证表决的结果是出自每个版本的同样交叉检测点，它的优点是可以及时发现软件故障，缺点是同步开销比较大；同步机制的选择还应能对超前和滞后的通道进行有效的检测和隔离。

交叉传输要在版本的交叉检测点 (CCP) 上进行，每个版本都应在同一规定的时间限制内到达同样的交叉检测点上；交叉检测点的数目可以根据系统的要求而定，对实时性要求较强的系统，交叉检测点的数目可以少一些；

a. 硬件输入检查点：在硬件输入之后进行的软件交叉检查，这时比较门限由主要由硬件特性决定，除此之外，非相似软件的异步运行也对门限的选择有影响，监控/表决算法用多版本表决；

b. 应用软件计算检查点：整个应用软件计算分为若干个模块，在一个模块算法完成之后，设置一个交叉检查点，用以决定在这一个软件模块运算的结果的正确性，这时其交叉通道检查向量

为这一个模块的输出结果和中间结果，这时要求用多版本表决算法。

c. 输出命令检查点：这是最终控制计算结果的表决，这时要求对于系统计算出的模拟控制面命令进行多版本表决。

交叉检查点设置的数量决定着对软件故障的检测隔离程度，设置的数量越多，则软件故障定位的更准确，但是带来的问题是软件版本表决算法和同步开销增大，从而实时性受到影响；

在每一个交叉通道检查点上，要求对检查向量进行多版本的表决，所以首先要根据系统软件和应用软件的各个软件模块和软件需求规范的内容来定义这些数据，形成一个交叉通道检查向量。交叉检查向量是一个版本程序运行状态的一个子集，.一般情况包括以下几类数据：

a. 比较变量的描述，如类型，数目等；

b. 各个通道处理器的状态，主要包括软件自检测和硬件自检测的结果；

c. 具体的数据.

由于各个版本使用的语言不同，为了比较各版本的结果，因而需要设计一种接口来处理不同语言使用的不一致的数据表示，即规定在每个交叉检测点上的交叉检 测向量 (CC) 的格式和内容，交叉检测向量的格式和内容应包括所有机器和语言的特征，这种处理技术被称之为软件总线；

余度管理是通过交叉传输的数据，监控和表决系统的输入/输出数据和版本的计算结果，负责故障的定位和隔离；故障的检测和隔离是基于通道本身的在线监控，交叉比较和BIT的结果，根据多数表决的原则定位和隔离故障通道。

由于多个不同的软件版本是对同一个输入参数进行处理，但由于处理器不同，指令系统不同，算法不同，最后的结果也有差别，与之有关的就有决定多版本软件结果是否正确的多版本表决算法．多版本表决算法要求很高的可靠性，所以必须简单有效，所以一般情况下出现2:1情况下判定少数为故障，当出现无法判断时要借助于硬件的在线监控和软件的运行时的检测进行．

版本表决算法的输入参数是三个在不同的中央处理机上运行不同的软件版本的结果，而对于不同的中央处理机运行软件模块结果的表示形式也就有所差别，所以各个判定算法对输入参数要求先进行预处理，将其数据的表示形式统一化，然后才能进入版本表决算法，最后的结果也要进行处理以适应于不同的处理机，为防止系统降级太快，每一个版本中重要的模块分别有一个主要的软件模块和若干 '冷备份' 软件模块，在一个时刻一个软件版本只有一个软件模块在运行，这种N版本软件中的某个版本的某个软件模块如果在交叉检查点上被判定为故障，则这一个软件模块以后将不能参加运行，这时可以利用前向恢复的方法恢复这一个版本的运行，本次运行内系统将不减少余度，从下一次运行开始，利用本次运行软件版本表决监控的结果，就可以控制隔离这一个版本的软件模块，将相应的软件模块 '冷备份' 切入计算。

5、小结

三余度非相似容错计算机系统NSFCS可以有效的消除了软、硬件设计有关的共性故障；并能将软、硬件引起的一次故障定位到通道；也可以发现软、硬件引起的二次故障；

另外，在系统设计中反映出'永不放弃'的原则，即在系统中只剩下有两个通道工作，但表决结果又不一致的情况下，要保证系统连工作，通道故障逻辑可以给出单通道工作方式，充分利用可用的资源；

NSFCS 是用硬件和软件的非相似性来获得系统的容错，系统设计是从实际应用出发，用检测、定位故障和系统重构避免系统的失效，给实时关键控制系统提供了一个重要的容错手段．