基于60x总线的Lockstep处理器架构

周 啸,李 鹏,韩 强

(中航工业西安航空计算技术研究所,陕西 西安 710068)

摘 要: 在航空器、汽车等很多需要高可靠性计算的系统中,通常采用余度技术。随着余度系统带来的功耗、体积、重量、管理等方面的问题越来越严重,要求处理器达到很高的可靠性以降低系统余度。Lockstep处理器架构能够迅速监测处理器运行的错误,进行故障隔离,防止故障蔓延,在处理器级实现高可靠性。进而,Lockstep处理器架构作为新型高可靠计算系统的关键技术,可以实现信息处理的高完整性和高可用性。对Lockstep技术进行分析研究并进行设计实现,实现一种高可靠、高可用的计算处理架构。

关键词: 锁步; 高可靠; 容错; 纠错

中图分类号: TP336 **文献标识码:** A **文章编号:** 1671-654X(2015)01-0127-04

Architecture of Lockstep Processing System Based on 60x Bus

ZHOU Xiao, LI Peng, HAN Qiang

(*Xi'an Aeronautics Computing Technique Research Institute, AVIC, Xi'an* 710068, *China*)

Abstract: The high-reliable computing system in the vehicles improves the reliability by the technique of redundancies. As the problem of power, weight, size and management in the redundancies system becomes more and more serious, it needs to use the high-reliability processor to reduce the redundancies for solving the problem. The architecture of lockstep processing system can detect the fault between the processor and isolate the fault rapidly, which improves the reliability. Furthermore the lockstep system can reach the high-availability as the key technique of new computing system. This paper researches the technique of lockstep system and describes a lockstep architecture to perform the high-reliable and high-available processing system.

Key words: lockstep; high-reliability; fault tolerance; error-detection

引言

Lockstep技术作为一项安全关键的计算机技术,在20世纪80年代首次提出,经过这些年发展,国外大部分先进计算系统已经开始采用模块化分布式计算机设计技术[1]来保证处理器的运行正确,提高系统的可靠性。Lockstep架构处理器采用2套复制的处理器、存储器、比较逻辑、总线/背板接口,存在逻辑控制两套总线进行比较之后输出。这种设计能够有效提高处理器对于指令级错误的检测能力,Lockstep处理结构作为新型高可靠性计算机系统的主要处理器节点,可以实现信息处理的高完整性和故障高隔离率[2]。例如,利用Lockstep架构为系统处理模块提供一个镜像的检测系统,在每次处理器处理信息时进行检测,如果处理状态一致,则认为该处理结果可靠,并将其结果输出作为系统的输出,如果结果不一致时认为错误发生,并认为处理器节点不可靠[3]。通过对这种Lockstep同步监控方式的研究,可以提高处理器对于偶发的、细颗粒度错误的甄别能力。本文针对计算机系统的高可靠性需求,通过实现一种Lockstep系统架构,对这种高可靠性处理器架构进行研究和探索。

1 Lockstep处理器的架构实现方式

一个高效的Lockstep容错架构应当包括错误检测、故障隔离和错误恢复逻辑等功能,目前,国外的Lockstep的架构有很多方式实现:

1) 主从式处理器验证的方式;

2) 双处理器内部验证比较;

3) 定制双核处理器验证比较。

收稿日期: 2014-12-15 **修订日期:** 2015-01-02

基金项目: 航空科学基金项目资助(20121931002)

作者简介: 周 啸(1986-),男,河南荥阳人,工程师,主要研究方向为计算机结构、逻辑设计。

其中主从式处理器验证的方式简单易行，成本较低，通过处理器的硬件实现，为一种典型的 Lockstep 模型，如 PowerPC750GX 的 Lockstep 架构[4]。这种方式易于利用现有的处理器进行扩展，可以在处理器外部利用同步比较逻辑等进行同步比较。但是，处理效率不高，其结构示意图如图 1 所示。

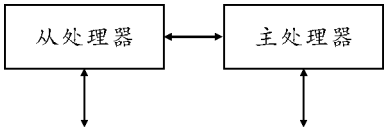

图 1 从处理器验证的方式

这种 Lockstep 架构一般通过设计一个 master 处理器正常运行输出，另一个处理器为 checker，负责监视 master 的工作情况，同步运行相同的任务，输出结果时主从结构逐步比较。当发现有不一致的情况出现时，表明出现了故障，系统停止工作。目前国外一些 Lockstep 架构采用的这种方式，设计简单，成本开销较小，例如 Freescale 的 PowerPC750GX 处理器。但是，这种 Lockstep 检测策略是不完善的，需要在处理器外额外扩展不同的同步比较架构进行同步比较，比较效率较低，导致处理器处理能力下降。

方式 2) 则是通过在双处理器内部验证比较。双处理器在内部进行同步比较，比较结果与外设总线通信，对于用户来说与单处理器相同。这种方式不仅可以减少系统板级的额外开销，同时还不需要额外的软件来协调同步。其结构示意图如图 2 所示。

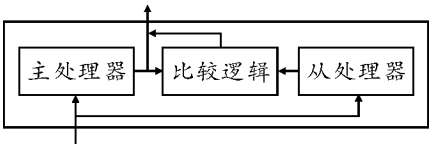

图 2 双处理器内部验证比较

但是这种方式同步检测只存在 CPU 与外部总线之间，而 CPU 外围的 memory、bridge 等外围资源缺乏对比控制，发生错误难以检测。与方式 1) 可以自由扩展同步比较方式相比，这种方式的 lockstep 架构不需要双处理器扩展，但是比较方式较为单一，仅在处理器内部进行比较。

方式 3) 则是定制双核处理器验证比较，通过用户定制同步双核处理器，可以在双核内部的流水线级进行同步比较。在指令级的同步比较可以迅速定位故障，进行隔离，其示意图如图 3 所示。

例如 Freescale 的微控制器 MPC5675K[5]，在处理器双核取指时进行指令级的 Lockstep 同步比较，结构对称，双核并行工作，这种双核处理器在取指时对 2 个核同步取指，并对每条取出的指令进行比较，如果指令不一致发生错误进行故障隔离，如果一致处理器继续运行。所以在可以在指令运行出错的第一时间立刻监测到故障，具有监测迅速，定位准确的特点，而方式 1)、方式 2) 则难以做到这一点，仅能在处理器与总线通信时监测到错误。与方式 1)、方式 2) 比较，指令级的 Lockstep 技术对比颗粒度小，其芯片实现方案可以简化应用设计，但是需要有自主设计的处理器芯片，自主流片等技术，定制较为复杂，验证不易，目前在国内较难实现。

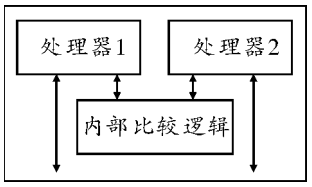

图 3 双核处理器内部验证比较

2 Lockstep 处理器的系统设计

经过研究，综合几种 Lockstep 架构的优缺点，本文设计一种新型的 Lockstep 架构的处理器系统，在处理器总线上进行比较监控，双处理器同源读取写回。这种 Lockstep 架构是通过在外部总线上设计逻辑比较 2 个处理器的处理结果，具有可靠性好，检测方式可以自由定制，故障检测率高以及处理器架构同步容易的特点，但是这种硬件架构需要增加硬件平台的复杂度。进一步的，可以通过这种 Lockstep 架构研究处理器指令级同步调试方法、应用功能和信息交换故障的实时隔离等，并进行可靠性分析与容错模型验证。该 Lockstep 架构的特点在于同时在 2 个松耦合的处理器之间通过共享的总线结构进行处理结果的对比，不仅仅在处理器内部比较，而且可以对比处理器总线以及桥上的数据，同时可以检测内存上的处理数据。

2.1 Lockstep 硬件架构设计

在本系统中，处理器采用 60x 总线与桥进行通信，而作为 Lockstep 架构的核心，总线监控器采用 FPGA 设计实现，用于对两个处理器节点 60x 总线上进行指令级锁步监控比较。处理器外部存储器、PCI 总线及 I/O 设备总线扩展采用专用桥接器实现。

总线监控器通过在 60x 总线插入等待信号实现处理器的锁步工作。总线监控器逻辑对 60x 总线上双处理器进行同步，使得双处理器能够同时执行同一条指令，如果同步失败，即在 60x 总线上双处理器的信号时差大于一定时钟数，则会判为故障，进行故障处理；在总线监控器同步 60x 总线信号后，还会对 60x 总线上

的信号进行比较，如果 2 路 60x 总线信号不一致，表明双处理器运行的结果出错，总线监控逻辑就报告故障，通过桥接器向双处理器发送中断信号，并触发外部通信接口控制信号，终止 TTE 终端系统通信，防止故障蔓延到系统其他部分。

该 Lockstep 系统模块主要由以下几个功能块构成: 处理器 CPU 模块、FPGA 逻辑作为总线监控器、桥接器、SDRAM、FLASH、JTAG 接口电路、以太网接口、离散量接口、复位电路、时钟电路、电源转换电路和 TTE 终端，CPU 模块结构框图如图 4 所示。

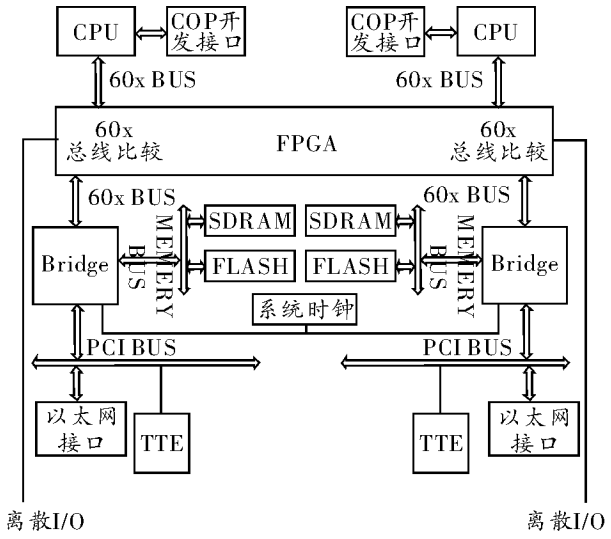

图 4　Lockstep 处理器的总线结构

Lockstep 机制主要采用同步 2 个 CPU 处理器，使其尽量运行在同一条指令。在 CPU 内部同步指令是非常困难且难以实现的，所以本系统考虑在 CPU 执行 L/S(Load/Store) 指令时通过 60x 总线检测 L/S 指令，在 CPU 之外通过 FPGA 设计 Lockstep 逻辑电路对处理器的运行进行控制。对处理器的 L/S 操作进行同步可以满足处理器检测的要求，PowerPC 处理器在运行时是基于寄存器堆的，基本的模式为内存→寄存器→运算→写回寄存器→写回内存，这也是基本的 RISC 架构处理器的处理方式。可以在处理器写回内存时检测处理器的运行状态，而在处理器操作内部的寄存器文件时不做检测，这样即使在运算或者写入寄存器文件时发生突发性的错误也会在写回内存时发现，可以满足检测机制的要求，同时不会频繁地发送双处理器的同步信号进行握手同步，从而减少同步机制引起的额外开销。

2.2　60x 总线协议

本系统中处理器采用 60x 总线与外部通信，60x 总线协议传输主要包括 3 个工作状态: 仲裁周期、地址周期和数据周期[6]。其中在仲裁周期中主处理器通过 BR 信号进行请求，收到 BG 信号之后，占用总线传输数据，之后进入地址周期，通过 TS 标示地址周期的开始，开始传输 ADDR 等地址信号，在传输完成后接收到 AACK 反馈应答信号结束读写地址的地址周期。在地址周期之后进行数据周期的传输，通过 DBG 标示数据的传输，再传输完 DATA 等数据信号后，接收到 TA 反馈信号完成一次数据传输。如图 5 所示。60x 总线上的地址周期和数据周期是分开的，可以在完成第一个地址周期传输之后立刻进行第二个地址周期的传输，而不必等待第一个地址周期对应的数据周期，这样在 60x 总线上存在一个乱序传输的过程，在系统中需要设计同步比较在 60x 总线上的地址与数据。

2.3　60x 总线上地址和数据周期的同步设计

由于 60x 总线上地址和数据的分开传输，需要对 60x 总线上的地址和数据周期进行同步设计。同时 Lockstep 系统中处理器与桥之间的 data 线为双向逻辑设计，而在总线控制逻辑 FPGA 中需要将 data 线分开为 data_in 与 data_out 两种，然后在 FPGA 中通过三态门来实现，这样在 60x 总线上需要根据地址周期的属性来判断之后数据周期中数据的方向，如图 5 所示，在本系统中采用一个 FIFO 结构来保证地址周期中的属性结构。

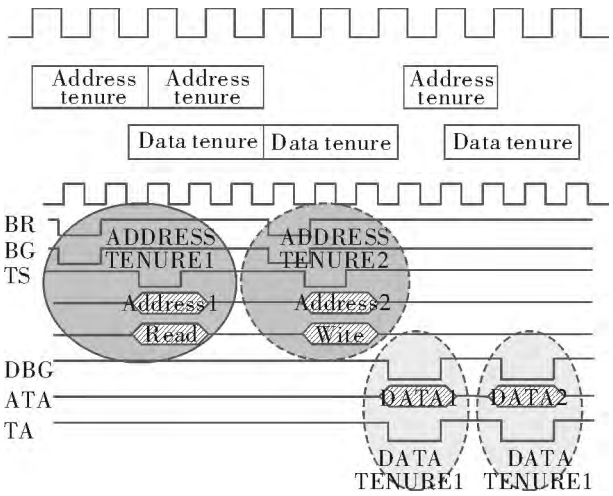

图 5　60x 总线上总线仲裁的设计

如图 5 所示，地址周期 1 时，通过 TS 信号将地址总线上传输地址与属性写入 FIFO 中，在传输地址周期 2 时，将地址周期 2 的地址属性写入 FIFO 中，在传输数据周期 1 时，通过 DBG 读出 FIFO 中的地址周期 1 保存的属性与地址，保证在数据周期 1 与地址周期 1 对应。同理将地址周期 2 与数据周期 2 对应。在地址周期与数据周期同步对应后，通过译码逻辑译出地址周期中的读写属性，并根据该属性控制三态门，选择出数据周期中的数据流向，如图 6 所示。

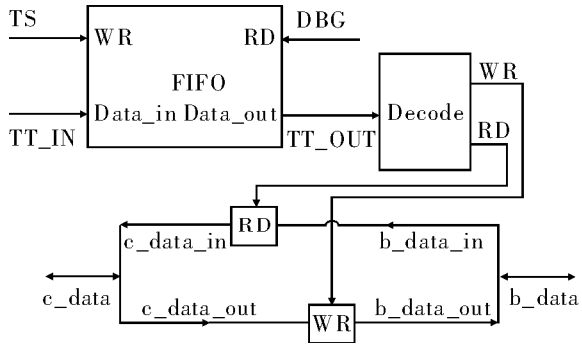

图 6 FIFO 结构的数据线仲裁

采用这种方式，可以在 FPGA 中对 60x 总线上的地址与数据周期进行同步对应，保证 lockstep 处理器的正常运行。

2.4 Lockstep 系统的同步比较

lockstep 系统同步双处理器通过 FPGA 总线监控同步 60x 总线仲裁信号实现，如图 7 所示，在 lockstep 处理器进行一次 L/S 操作时，首先某一个处理器进行请求 60x 总线，发送 BR_C0 信号，当有总线请求时总线监控控制器将该请求信号保留，而不直接发送给桥接器，直到另一个处理器的 BR_C1 信号有效才同时将两个处理器的 BR 信号有效，将 BR_B0，BR_B1 发送给桥接器，实现 Lockstep 机制下的一次同步的总线仲裁，达到总线同步的目的。

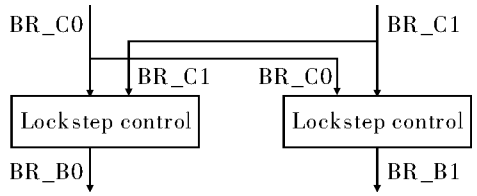

图 7 Lockstep 控制器示意图

这样，只有在 2 个处理器均发起 BR 请求总线时才将 2 个 BR 信号传送给桥接器进行总线传输，而在只有一个 BR 信号时桥接器不会收到，从而不会回执反馈信号 BG，使得处理器处于等待状态。其他 60x 总线上的信号也可以同理采用这个机制来实现同步。

在 Lockstep 机制下，同步之后的 60x 总线信号比较可以直接通过 FPGA 中比较完成。如果比较结果相等，认为两个处理器运行同一条指令正确，让处理器继续运行，如果比较结果不同，认为处理器节点不可靠，可以进行故障隔离或者进一步进行故障恢复。这样，通过总线同步比较可以控制双处理器同步的运行，并在比较结果不一致时迅速进行故障隔离，从而达到高可靠性的要求。

3 实验结论与展望

目前该 Lockstep 系统架构可以正常运行 Vxworkes5.5 操作系统，并运行应用程序，与正常处理器一致。进行测试时可以模拟在恶劣环境中内存数据突然翻转出错的情形，通过修改 2 个 CPU 节点中的内存数据等方式进行故障注入，该模块可以立即进行错误报告，达到高可靠性的要求，进一步的可以针对故障情况进行一系列的故障处理，甚至进行错误恢复，达到高可靠性计算。

目前的多余度容错设计技术在实际工程中更多地是在系统级、模块级的容错，即通过专用网络完成系统内各个模块之间的信息交换，然后进行比较、监控和表决。对于更细的颗粒度上的容错技术，更多地是进行理论分析和仿真验证研究工作。目前，国内还未见 Lockstep 的相关资料和产品，作为一项安全关键的计算机技术，本文对 Lockstep 技术进行分析研究和实践，实现了 Lockstep 系统架构的同步运行，检测比较，故障隔离等，进一步将实现先进高可靠计算机系统的要求。

参考文献：

[1] Ganesh C Gopalakrishnan, Narayana S Mani, Venkatesh Akella. Parallel Composition of Lockstep Synchronous Processes for Hardware Validation: Divide - and - conquer Composition [J]. Lecture Notes in Computer Science, 2005, 407: 374 - 382.

[2] Reorda M S, Sterpone L. Fault Injection-based Reliability Evaluation of SoCs [C]//Proc. IEEE EuroTest Symp. 2006: 75 - 82.

[3] Casey M Jeffery, Renato J O Figueiredo. A Flexible Approach to Improving System Reliability with Virtual Lockstep [J]. IEEE Transactions on Dependable and Secure Computing. 2012, 9: 2 - 15.

[4] PowerPC 750GX Lockstep Facility User Manual [EB/OL]. http://www.ibm.com, 2008 - 3.

[5] MPC5675K User Manual [EB/OL]. http://www.freescale.com, 2010 - 8.

[6] PowerPC Microprocessor Family: The 60x Bus Interface for 32 - Bit Microprocessors [EB/OL]. http://www.ibm.com. 2000 - 10.