

电子科技大学
UNIVERSITY OF ELECTRONIC SCIENCE AND TECHNOLOGY OF CHINA

专业学位硕士学位论文

MASTER THESIS FOR PROFESSIONAL DEGREE



论文题目 基于 VPX 总线的工件台运动控制系统研究与开发

专业学位类别 工程硕士

学号 201322080449

作者姓名 廖晨翔

指导教师 朱煜 教授

分类号 _____ 密级 _____

UDC ^{注1} _____

学 位 论 文

基于 VPX 总线的工件台运动控制系统研究与开发

廖 晨 翔

指导教师 朱 煜 教 授

电子科技大学 成 都

(姓名、职称、单位名称)

申请学位级别 硕士 专业学位类别 工程硕士

工程领域名称 机械工程

提交论文日期 2016.4.5 论文答辩日期 2016.5.12

学位授予单位和日期 电子科技大学 2016 年 6 月

答辩委员会主席 _____

评阅人 _____

注 1: 注明《国际十进分类法 UDC》的类号。

Research and Development of Motion Control System of Stage Based on VPX Bus

A Master Thesis Submitted to
University of Electronic Science and Technology of China

Major: **Master of Engineering**

Author: **Chenxiang Liao**

Supervisor: **Yu Zhu**

School: **School of Mechatronics Engineering**

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

作者签名：_____ 日期： 年 月 日

论文使用授权

本学位论文作者完全了解电子科技大学有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人授权电子科技大学可以将学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

(保密的学位论文在解密后应遵守此规定)

作者签名：_____ 导师签名：_____

日期： 年 月

摘 要

工件台系统是光刻机的关键子系统之一，工件台运动控制系统对实现光刻机性能指标具有至关重要的作用，因此研发工件台运动控制系统具有极其重要的工程应用价值。论文根据工件台控制系统必须具备的并行性、同步性和实时性等技术需求，建立了基于 VPX 总线的工件台运动控制系统硬件体系，并针对基于 C6678 共享内存的数据并行交互机制和系统精密同步测控策略开展了研究。

为解决系统中存在的多处理器之间数据传输实时性、处理器运算性能、光纤接口数量以及系统计算架构等问题，通过对 VPX 总线中多总线协议的研究，采用基于 RapidIO 总线的全网状星型拓扑结构，并结合集成有多核 DSP TMS320C6678 的运动控制卡 MC_4DSP_VPX 和光纤接口卡 FC_FPGA_VPX 等硬件，设计了一种基于 RapidIO 多处理器互连架构的新型工件台运动控制系统硬件体系结构。

工件台的多子系统决定了系统数据并行处理的特征，因此从并行性和实时性出发，根据系统硬件体系结构，设计了控制系统的数流，并以此建立了并行处理模型。根据并行模型的交互数据内容，规划了共享内存和核本地内存的存储空间，结合 C6678 的数据 Cache 与共享内存的数据一致性原则，实现了多核实时并行访问共享内存。

为了提高工件台控制系统性能，除了增强系统数据交互并行性，系统的同步性和实时性也至关重要。系统的同步主要分为多 DSP 同步和多核同步，为了实现实时同步中断多个 DSP，提出了两种同步测控方案，通过实验数据分析，基于 Direct I/O + GPIO 硬中断的同步测控方案比基于 Direct I/O + Doorbell 软中断的同步测控方案具有更好的同步性和实时性。为了实现 C6678 的多核同步，分别具体介绍了基于广播事件的多核同步技术和基于核间中断的多核同步技术两种方法，比较分析可知，基于广播事件的多核同步更能满足强实时系统设计的要求。

在上述研究分析的基础上，以双扫描硅片台、掩模台为控制对象，对多核 DSP 软件进行了整体框架设计。在现有的最小测试系统平台上，通过实验对比，验证了同步测控方案的可行性、软件设计的正确性以及控制系统的实时性和同步性。

关键词：工件台运动控制系统，VPX 总线，RapidIO 互连架构，共享内存，同步性与实时性

ABSTRACT

Stage system is one of the key subsystems of the lithography machine. Its motion control system plays a vital role in realizing lithography machine's performance and indicators. Therefore R&D of the system has very important value in engineering application. According to the system's technical requirements with parallelism, synchronization and real-time performance included, the hardware architecture of the stage's motion control system based on VPX bus is established, and also this thesis carries out research on system data parallel interaction mechanism based on TMS320C6678 shared memory and the precise synchronization measurement and control strategy.

In order to solve the problems existing in the system, such as real-time data transmission between multiprocessors, computing performance, the number of optical fiber interface and the system's computing architecture, through the study of the multi-bus protocol in VPX bus, the full mesh star topology based on RapidIO bus is adopted. A new hardware architecture base on VPX bus for this system is designed, which includes the motion control card MC_4DSP_VPX integrated with multi-core DSP TMS320C6678 and optical fiber interface card FC_FPGA_VPX.

The stage's multiple subsystems determine the characteristics of system data parallel processing. Therefore from the perspective of parallelism and real-time performance, on the basis of the system hardware architecture, the thesis shows the data flow design of the system and the parallel processing model. The storage space of the shared memory and local memory is planned in terms of the interactive data content. With the principle of data consistency between the data cache of C6678 and shared memory, the multi-core parallel access to shared memory is implemented.

In order to improve performance of the stage's control system, synchronization and real-time performance of the system are also crucial in addition to enhance the system data interactive parallelism. The system synchronization includes multi-DSP and multi-core synchronization. To realize the multi-DSP synchronous interrupt, the thesis puts forward two synchronous measurement and control schemes. By analyzing the experimental data, the scheme based on Direct I/O + GPIO hardware interrupt has better synchronization and real-time performance than the one based on Direct I/O + Doorbell software interrupt. To realize the C6678 multi-core synchronization, the thesis presents

the multi-core synchronization technology based on the broadcast event and kernel interruption respectively. And the comparative analysis shows that the one based on the broadcast event can meet more of the design requirements of strong real-time system.

On the basis of the above research and analysis, the thesis explains the overall framework design of multi-core DSP software whose control objects are the double scanning Wafer Stage and Reticle Stage. Through experimental comparison, the feasibility of synchronous measurement and control scheme, the correctness of the software design and real-time and synchronization performance of the control system is verified.

Keywords: Stage's Motion Control System, VPX Bus, RapidIO Interconnect Architecture, Shared Memory, Synchronization And Real-time Performance

目 录

第一章 绪论	1
1.1 课题背景及意义	1
1.2 光刻机及其工件台概述	2
1.3 多轴运动控制系统技术研究现状	3
1.3.1 多轴运动控制系统发展概况	3
1.3.2 工业计算机总线发展概况	4
1.4 工件台的多轴运动控制系统技术研究现状	6
1.4.1 工件台控制系统国内外研究进展	6
1.4.2 基于 VME 总线的工件台控制系统	7
1.4.3 基于 ATCA 总线的工件台控制系统	10
1.5 论文的研究内容	12
第二章 基于 VPX 总线的工件台运动控制系统总体方案设计	13
2.1 系统设计需求分析	13
2.1.1 工件台系统总体介绍	13
2.1.2 系统设计问题分析及需求提出	16
2.2 系统互连总线及关键硬件	18
2.2.1 互连总线选择	18
2.2.2 运动控制卡 (MC_4DSP_VPX)	20
2.2.3 光纤接口卡 (FC_FPGA_VPX)	22
2.2.4 主控卡 (HOST_CPU_VPX)	23
2.2.5 VPX 机箱背板	24
2.3 基于 RapidIO 总线的控制系统硬件体系结构设计	25
2.3.1 控制系统总体层次结构划分	25
2.3.2 基于 RapidIO 总线的控制系统架构	26
2.3.3 工件台控制系统硬件体系结构详细设计及功能分析	27
2.4 本章小结	30
第三章 基于多核共享内存的系统数据并行交互机制	31
3.1 系统数据流结构	31
3.2 基于并行处理模型的数据交互方式	32
3.2.1 系统并行处理模型	32

3.2.2 数据交互方式分析.....	33
3.3 C6678 的数据 Cache 一致性分析和维护.....	34
3.3.1 C6678 的数据 Cache 一致性问题.....	35
3.3.2 C6678 的数据 Cache 一致性维护操作.....	36
3.4 基于共享内存的数据交互.....	37
3.4.1 基于共享内存的数据存储空间规划.....	37
3.4.2 共享内存与本地内存 LL2 的数据地址对应关系.....	38
3.4.3 数据具体交互过程.....	39
3.5 本章小结.....	40
第四章 控制系统精密同步测控策略研究及技术实现.....	41
4.1 TMS320C6678 的中断子系统结构和中断映射原理.....	41
4.1.1 TMS320C6678 的中断子系统结构.....	41
4.1.2 片级中断控制器 (INTC).....	42
4.1.3 核级中断控制器 (CorePac Interrupt Control).....	43
4.2 工件台控制系统同步测控方案设计分析及分析.....	44
4.2.1 基于 Direct I/O + Doorbell 软中断模式的同步测控方案.....	45
4.2.2 基于 Direct I/O + GPIO 硬中断模式的同步测控方案.....	45
4.2.3 两种同步测控方案的比较分析.....	46
4.3 基于 GPIO 硬中断的多 DSP 同步技术实现.....	47
4.4 多核同步技术实现及分析.....	48
4.4.1 基于广播事件的多核同步触发实现.....	48
4.4.2 基于核间中断 IPC 的多核同步触发实现.....	50
4.4.3 两种多核同步方法的比较分析.....	51
4.5 系统同步时序最优化设计.....	51
4.5.1 最小测控模型建立.....	51
4.5.2 系统同步时序最优化详细设计.....	52
4.6 本章小结.....	53
第五章 工件台运动控制系统软件初步设计及实验验证.....	54
5.1 控制系统软件设计.....	54
5.1.1 工件台控制系统总体框架.....	54
5.1.2 工件台 DSP 伺服控制软件总体设计.....	54
5.2 控制系统测试与分析.....	58
5.2.1 测试系统平台简介.....	58

5.2.2 RapidIO 数据传输实时性测试.....	59
5.2.3 多核同步实时性测试.....	62
5.2.4 系统总体实时性测试.....	63
5.3 本章小结.....	64
第六章 总结与展望.....	65
6.1 研究内容总结.....	65
6.2 工作展望.....	66
致 谢.....	67
参考文献.....	68
攻读硕士学位期间取得的研究成果.....	71

第一章 绪论

1.1 课题背景及意义

随着电子信息产业的快速发展，移动互联设备、可穿戴设备、信息家电、信息交互设备、雷达装备、医疗器械、通信设备及工业控制器等各种智能设备的出现不仅改变了人们的生活方式，同时促进了半导体器件制造技术的突飞猛进。智能设备或多或少涉及到最基本的硬件单元——半导体集成电路（Integrated circuit, IC）芯片。自二十世纪七十年代第一块 IC 芯片诞生至今，IC 芯片的集成度和性能不断提升，一直遵循了著名的摩尔定律，不断挑战并推动着人类在超精密加工制造领域的技术极限^[1]。而在整个 IC 芯片生产制造流程中，IC 制造装备——光刻机起着至关重要的作用。

光刻机的研制涵盖了光学、机械、精密测量、电气、控制等多学科交叉技术，其由光学系统、对准系统和工件台系统组成，如图 1-1 所示^[2]。作为最重要的子系统之一，工件台子系统具有步进、同步扫描、对准扫描、上下片、定位、调平聚焦等超精密运动功能。工件台及其运动控制系统是提高光刻机曝光分辨率、套刻精度以及产率的关键技术，其高定位精度、高运动速度、良好的动态性能等直接影响了光刻机的系统性能。

由于光刻工艺不断提升，极大规模的电路集成在更小的芯片上，芯片功耗价格不断降低，促使智能设备越来越小型化、智能化、经济化。然而唯有国外的极个别大公司，如荷兰的 ASML、日本的 NIKON、CANON 三大厂商掌握着光刻机制造的核心技术，在光刻机技术上占有绝对的领先地位，并形成成长久的技术垄断，几乎占领了全部的市场份额^[3]，对我国半导体集成电路制造产业的发展造成了严重制约。因此，半导体集成电路制造被纳为国家的重大战略目标。

不断提升的光刻机性能指标对工件台控制系统的性能要求越来越高，与此同时，工件台控制系统的设计方法也随之发生了巨大的变化。在工件台控制系统中，工件台又可分为硅片台和掩模台，硅片台用于承载芯片，掩模台则承载着掩模，控制软件调度硅片台和掩模台的多轴按照预设的高精度轨迹协同运动。为了进一步优化轨迹定位、跟踪精度，并提高生产效率，可适当的缩减伺服周期。然而，由于整个工件台系统实时性要求极高，所需控制的运动轴数多，而且涉及的传感器信号众多。因此，工件台控制系统在数据并行计算技术、多传感器信息融合技术、复杂运动控制算法、实时性和低功耗等方面的技术有待得到进一步改进，不仅要求系统中相应的硬件在数据实时处理能力、数据存储容量、可靠性及能耗等方面具有更突

出的性能，而且对控制系统网络拓扑结构、网络通信带宽等提出了更为苛刻的设计要求。综上所述，研究光刻机工件台控制系统技术具有重要的意义。

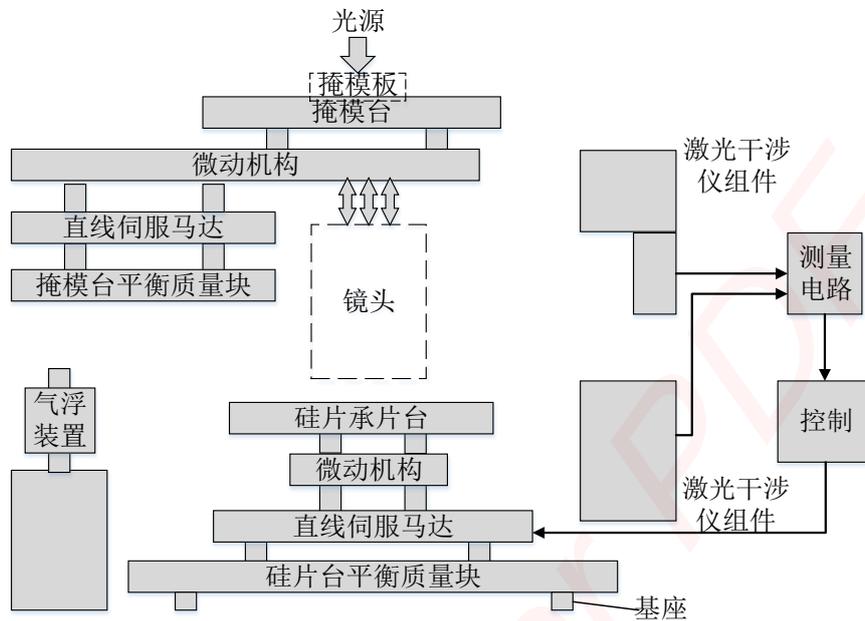


图 1-1 工件台系统组成框图

1.2 光刻机及其工件台概述

随着 IC 芯片的集成规模越来越大，光刻技术也经历了四十多年的发展历程，从早期的接触式光刻、接近式光刻、扫描投影式光刻、分步重复投影式光刻到步进扫描投影式光刻^[4]。

目前比较主流的光刻机均采用步进扫描投影式技术，其基本曝光工作原理即利用平行紫外线透过绘制有电路芯片构造的掩模，将掩模上的电路图像按照 4:1 的比例通过投影曝光复制到硅片上的曝光视场内，同时用来去除晶圆表面保护膜^[5]。根据投影物镜光学系统的设计原理，曝光区域为整个曝光视场的一条狭缝^[6]。硅片上包含多个曝光视场，曝光过程中物镜固定不动，硅片台做步进运动，待曝光的视场移动到物镜下，然后掩模台与硅片台沿 Y 方向以 4:1 的速度比做相反的不同步扫描运动，完成一场的扫描曝光，之后硅片台沿 X 方向做步进运动将硅片上的下一曝光视场移动到物镜下，如此循环完成整个硅片上的所有曝光视场的曝光^[7]。

在扫描曝光过程中，硅片台和掩模台的同步扫描运动精度、速度和加速度直接影响到曝光过程中图案的成像质量。同时，IC 芯片包含多层的复杂电路，需要多次曝光，层与层之间的图案需要保证一定的位置关系即套刻精度。因此对光刻机工件台的定位、同步运动精度等性能具有非常苛刻的要求^[8]。

对于工件台运动控制精度问题，国外公司 Canon 和 Nikon 可直接设计和制造

达到某种特定扫描精度的直线电机，通过这种直线电机构造具有单层结构的运动台，但这种结构却难以满足光刻工艺所需的大行程、高速、高精度等综合运动性能要求，而 ASML 则采用粗精动叠层的运动方式，并首先采用两套独立的硅片台系统，可以分别同时完成测量和曝光过程，这种设计方案大大提高了运动精度^[9]，也提升了工作效率。图 1-2 是 ASML 的 Twinscan 系统的基本结构^[10,11]。然而，在此方案中，双工件台结构却增加了所需控制的运动轴以及传感器数量，进而导致多轴协同运动控制系统设计方案的复杂度增加。一方面体现在工件台的数学模型、多轴的协调控制策略及算法等；另一方面，从硬件和软件设计层面来看，硬件的处理和通信性能要求提高。从光刻机市场上主流的光刻机产品以及专利检索结果来看，研究和改进双工件台技术及其控制系统技术已成为提升光刻机指标性能和生产效率的必要手段。

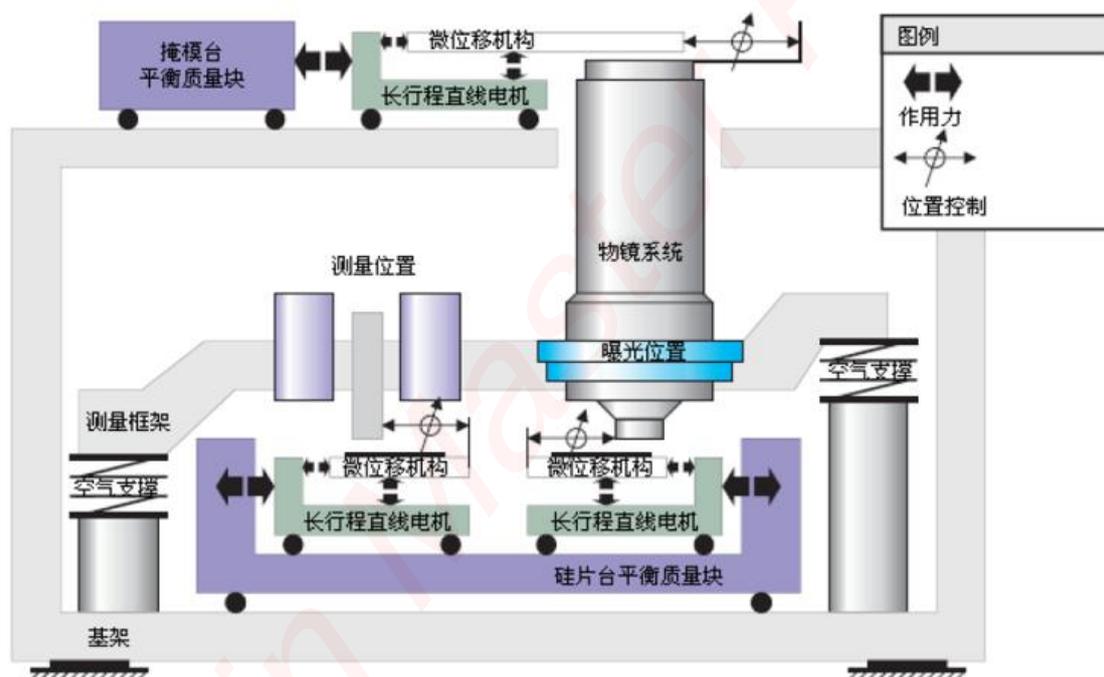


图 1-2 Twinscan 系统的基本结构

1.3 多轴运动控制系统技术研究现状

1.3.1 多轴运动控制系统发展概况

在诸多类似于光刻机工件台的高速高精度多轴运动控制过程中，运动台总体的运动精度不仅仅取决于结构方面的设计、加工制造以及安装精度，而且还受限于运动控制系统对多轴协同控制性能，以实现精密的轨迹运动。

随着控制技术突飞猛进，并且运动平台控制轴数及控制要求的不断提升，多轴

运动控制系统已经从单一的运动控制系统逐渐延伸到基于总线的网络化运动控制系统，而且在运动控制系统中，控制器也从以单片机或微处理器为核心发展到以专业芯片(ASIC)为核心；系统架构从之前的单处理器系统架构发展到如今的多CPU或DSP+FPGA多处理器架构。

针对多轴高精度同步控制，根据控制对象的复杂程度设计相应的控制系统网络架构，除了对控制系统的软硬件可靠性和可扩展性具有严格的要求外，同时，就控制性能而言，多轴运动控制系统需要具备两个最基本的特征^[12]：实时性和同步性。实时性是指在足够短且合适的伺服周期内，运动控制器以足够快的速度向运动执行机构发送控制指令，并从传感器节点获取反馈数据。同步性是指在多轴运动控制系统中，各驱动轴之间的运动应具有严格的同步关系。因此，高性能的运动控制器和网络总线是必须考虑的。

与高精度工业机器人、超精密数控机床等制造装备的多轴控制系统不同之处在于，光刻机工件台系统要求多达44轴的电机同步运动，控制要求及难度远远高于前两者。因此，要求工件台多轴运动控制系统需采用多卡分别控制不同的电机组，通过高速总线互连，实现多卡之间的数据实时传输。

1.3.2 工业计算机总线发展概况

在工业控制和计算机通信等领域中，嵌入式系统对总线传输速度及数据处理速度的要求越来越高，常用的总线形式已由传统的共享式并行总线构架逐渐发展成高速串行总线构架形式。对于目前大多数基于CPCI、VME总线构架的嵌入式系统而言，即使VME64X总线的带宽已达到320Mb/s，却依旧难以满足强实时性系统对大吞吐量、低延时的要求，因此，总线带宽已经严重制约了嵌入式系统处理能力的进一步提升。随着系统对实时性、高速计算、可靠性等方面越来越高的要求，迫切需要一种新体制的总线来满足嵌入式信号处理系统的需求^[13]，因此，新型高速串行总线标准如VPX、ATCA等开始崭露头角。

VPX (Verstatile Protocol Switch bus)^[14]总线构架是基于VME的缺陷和高速串行总线的不断应用而发展起来的，它满足军方、研究机构、一些高校企业等对强坚固、多核多CPU族计算、DSP数据处理、多种串行技术集成应用的环境^[15]。图1-3展示了VME到OpenVPX的发展历程，自从VME引入后的25年以来，VPX对VME总线构架做了重要的改进，主要变化是采用高速串行总线替代并行总线和多总线协议的融入^[16]，如VPX的VITA46规范针对互连协议定义了一系列子协议规范：RapidIO、PCI Express (以下称PCIe)和GbE (万兆以太网)。VPX多总线技术不仅提升了系统总线带宽，而且在背板上扩展了更多的I/O接口，使得系

统的格式布局更加灵活^[17]，并采用 RapidIO、PCIe、GbE 和 Advanced Switching Interconnect 等现代的工业标准的串行交换结构，可构建全网状、星形、双星、菊花链式通信架构，为嵌入式系统部署提供了多种灵活的互连方式。为了在系统集成过程中，各大厂商的 VPX 设备能相互兼容，OpenVPX 总线在 VPX 规范基础上定义了系统兼容框架，包括结构、协议等方面的完善。

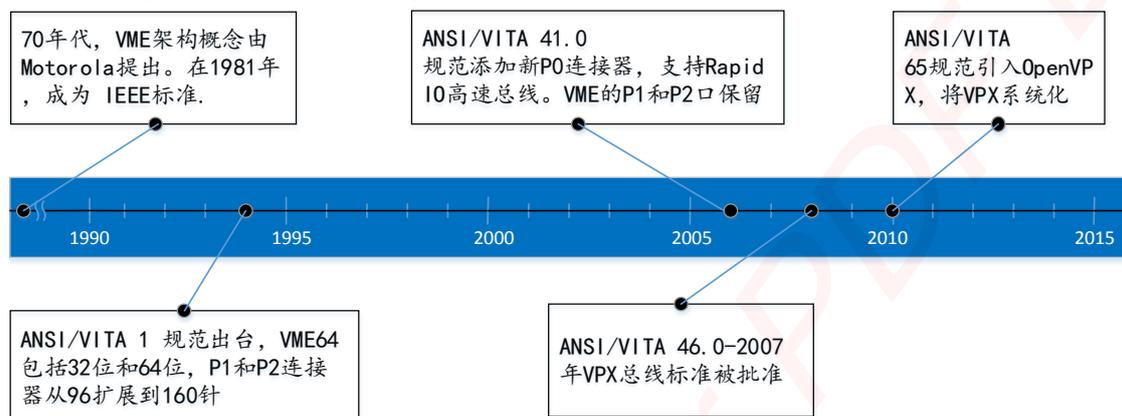


图 1-3 VME/VPX/OpenVPX 总线发展时间轴

ATCA (Advanced Telecom Computing Architecture)，即高级电信计算构架，是由全球 PCI 工业计算机制造组织 (PICMG) 提出的一种先进通信架构，在 2002 年底发表了 3.0 标准，即 ATCA 的核心规范，其主要变化是 CPCI 架构的低速并行总线逐渐被 ATCA 高速串行总线所替换，而且 ATCA 总线架构的网络拓扑结构多种多样。ATCA 的接口信道建立在 LVDS (低压差分信号) 差分对基础之上，采用点对点双工通讯以实现高速数据传输，其电气特性达到 3.125GHz。ATCA 构架可兼容多种网络总线协议，例如 GbE、PCIe 和 RapidIO 等，同时支持多种交换互连拓扑结构，如双星型、双—双星型和全网状。根据 ATCA 的可模块化和可配置特性，同一个平台上可以混合集成多种接口和不同技术的多功能模块(如 DSP, CPU、FPGA 和储存器)^[18]。由于 ATCA 系统在高总线带宽、强实时性、高可靠性以及灵活性等方面具有明显优势，电信设备厂商一直将其作为主要的通信架构平台^[19]。但 ATCA 的系统架构大、开发成本高、占用空间较大等缺点使得 ATCA 构架的使用范围具有片面性。虽然针对 ATCA 以上的几点缺点，PICMG 发布了 MicroTCA 技术标准，但一般只在大型网络(如电信运营商)中应用良好。

ACTA 和 VPX 构架中均包含 RapidIO、PCIe、GbE 等高速总线，RapidIO 总线在复杂嵌入式系统中应用更灵活，主要应用于 DSP、PowerPC、FPGA 等多处理器高密度系统中；PCIe 普遍应用于商用桌面电脑、高性能的单板计算机和协处理器中；而 GbE 是目前最为普遍的网络技术，一般作为平台间网络互连。

从表 1-1 可知,这三种总线不管是在数据传输速率还是在系统拓扑灵活性等方面,相对于 CPCI、VME 总线都具有明显的优势。虽然 ATCA 和 VPX 构架所涵盖的总线协议内容差不多,但 ATCA 长期以来面向于电信领域,主要服务于电信设备制造商,在工业控制领域可供使用的外围板卡很少;而 VPX 总线构架是在 VME 基础上发展起来的,在工业控制领域中 VPX 技术相对比较成熟,相应的外围板卡支持较多,且开发周期短。

表 1-1 多种总线的比较

属性	CPCI	VME320 (2eSST 协议)	VPX(RapidIO 1.x version)	VPX(PCIe 2.0version)	VPX(GbE)
数据 带宽	32bit/33MHz: 132MB/s 64bit/66MHz: 528MB/s	320Mb/s	10Gb/s(x4) @3.125Gb/s	16Gb/s(x4) @5Gb/s	10Gb/s(x1)
数据 通信 架构	PCI 并行总线	VME 并行总线	多种协议串行通信总线		
单板 供电	5V/35W 3.3V/33W 12V/12W	5V/90W 3.3V/66W	5V/120W 12V/383W 48V/768W		

1.4 工件台的多轴运动控制系统技术研究现状

1.4.1 工件台控制系统国内外研究进展

为满足越来越高的运动控制性能、功能需求以及数据通讯速率要求,传统的工件台运动控制系统在信息处理能力、数据吞吐速度、实时性等面临更高的技术挑战。光刻机工件台的控制数据不断增加导致系统需要采用更高速率的总线来维持数据流的通畅,因而整个运动控制系统的硬件和软件复杂度增大,技术难度也不断上升。伴随着计算机技术和网络通信技术飞速发展,其技术和成本也正在逐渐降低,光刻机制造商不断地更新高速计算机总线来构建更复杂、更高性能的运动控制系统。

从近几年的国际半导体发展路线图 (ITRS) 来看,目前,特征线宽 65nm 以上的光刻机一般采用基于 VME 总线技术的运动控制系统。ASML 光刻机产品技术发展史^[20]中,如图 1-4 所示,从 PAS Stepper 产品才开始使用 VME 总线,且在 PAS Scanner、TWINSCAN AT、TWINSCAN XT^[21]等系列产品中一直沿用。与此同时,在 CARM^[22] (Control Architecture Reference Model) 体系中采用多处理器互连^[23]方

式，处理器的性能也从 20MHz 达到 800MHz，因此就 ASML 光刻机运动控制系统而言，伺服控制环执行时间也大大缩短^[24]。

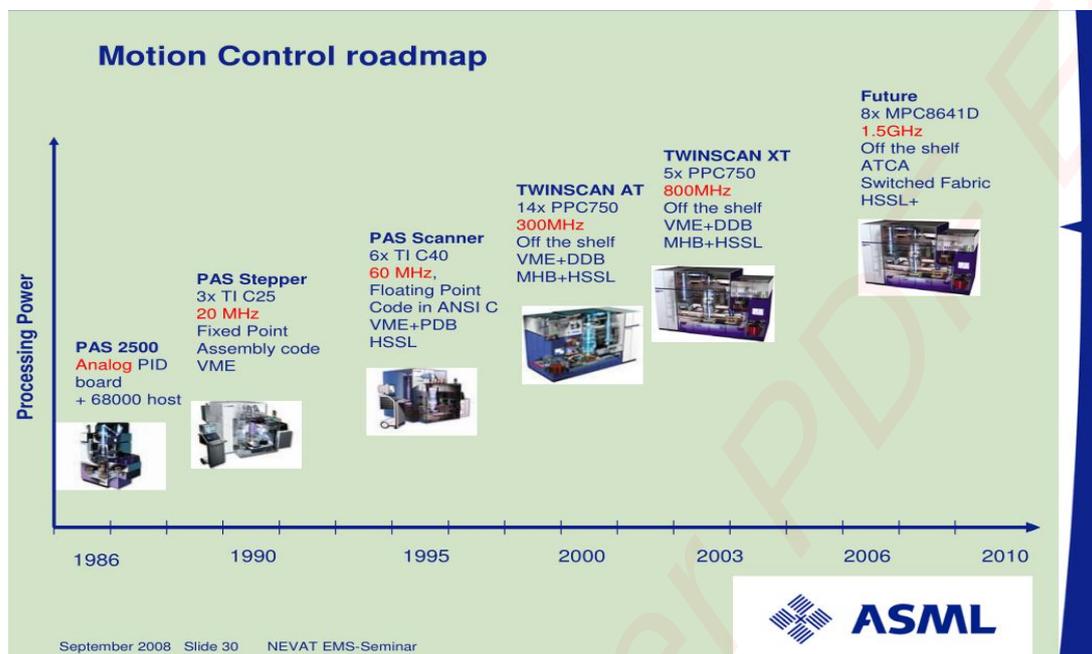


图 1-4 ASML 光刻机运动控制技术路线图

继 2003 年之后，为提高产品的市场竞争能力，ASML 在光刻机产品中采用 ATCA 高级电信计算构架来实现高速数据传输，并配备 1.5GHz 甚至更高的多核处理器，结合 FPGA 的使用，提高采样频率，减小 IO 延迟^[25]，对整个光刻机系统性能实现大幅度的提升。

国内相关领域的高校、研究机关、企业等也采用基于 VME 总线的 PowerPC+DSP 多处理器架构，并嵌入 VxWorks 操作系统来研究并开发工件台运动控制系统^[26]。其中，高校研究团队主要有华中科技大学、清华大学和哈尔滨工业大学团队，而国内首台先进封装光刻机整机由上海微电子装备有限公司 SMEE 成功研制^[27]。下面介绍国内外基于 VME 总线的控制系统和 ASML 基于 ATCA 的控制系统研制情况。

1.4.2 基于 VME 总线的工件台控制系统

1.4.2.1 SMEE 研制的 VME 控制系统

SMEE 的工件台控制系统硬件构架为集中式，系统由一台基于 VME 总线的工控机构成系统的基本框架，实现与上位机的通讯以及调度系统中其它功能模块，如图 1-5 所示。系统具体的运动控制由多块基于 VME 总线的专用运动控制卡 MC 实

现，MC 并行计算后将控制信号下传给电机驱动器，位置反馈信号由传感器接口板和激光双频干涉计数卡采集，MC 与 MC 之间、MC 与激光双频干涉计数卡之间的数据传输通过 PDB 总线（自定义的位置数据总线）。其中，PowerPC 单板作主控计算机，采用 TCP/IP 协议与上位机通讯，并通过 VME 总线与 MC 进行实时和非实时数据交换^[28]。

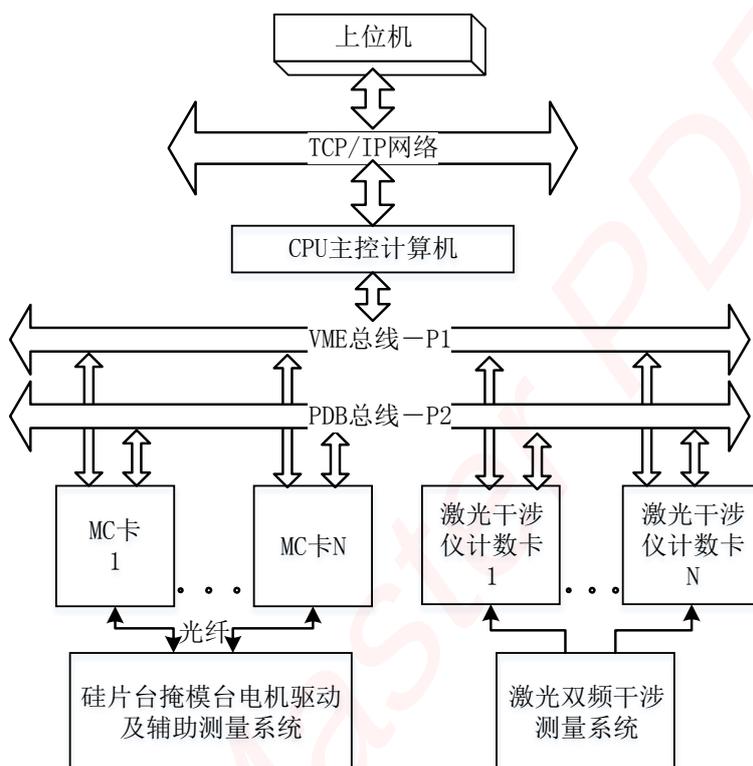


图 1-5 SMEE 研制的基于 VME 总线的控制系统架构

1.4.2.2 ASML 研制的 VME 控制系统

ASML 的工件台控制系统硬件框架为主从式结构，如图 1-6，由两台基于 VME 总线的工控机构成系统的基本框架，其中一台实现与上位机的接口和运动控制，另一台主要进行激光双频干涉仪反馈信号处理，ASML 公司之所以采用这种结构主要是由于双硅片台系统需要控制的对象大量增加，而且反馈信号也大量增加，由一台计算机已经无法容纳众多的控制卡和信号接收处理卡了。同时 ASML 的运动控制卡仅作数据处理而无控制 I/O，不能直接向电机驱动器下发信号，而是通过另外的光纤接口卡进行数据收发。激光干涉仪测量侧中的同步总线卡 SBC 采集并处理激光位置数据，通过光纤传输至运动控制侧的光纤接口卡中。光纤接口卡与运动控制卡之间通过总线通信。

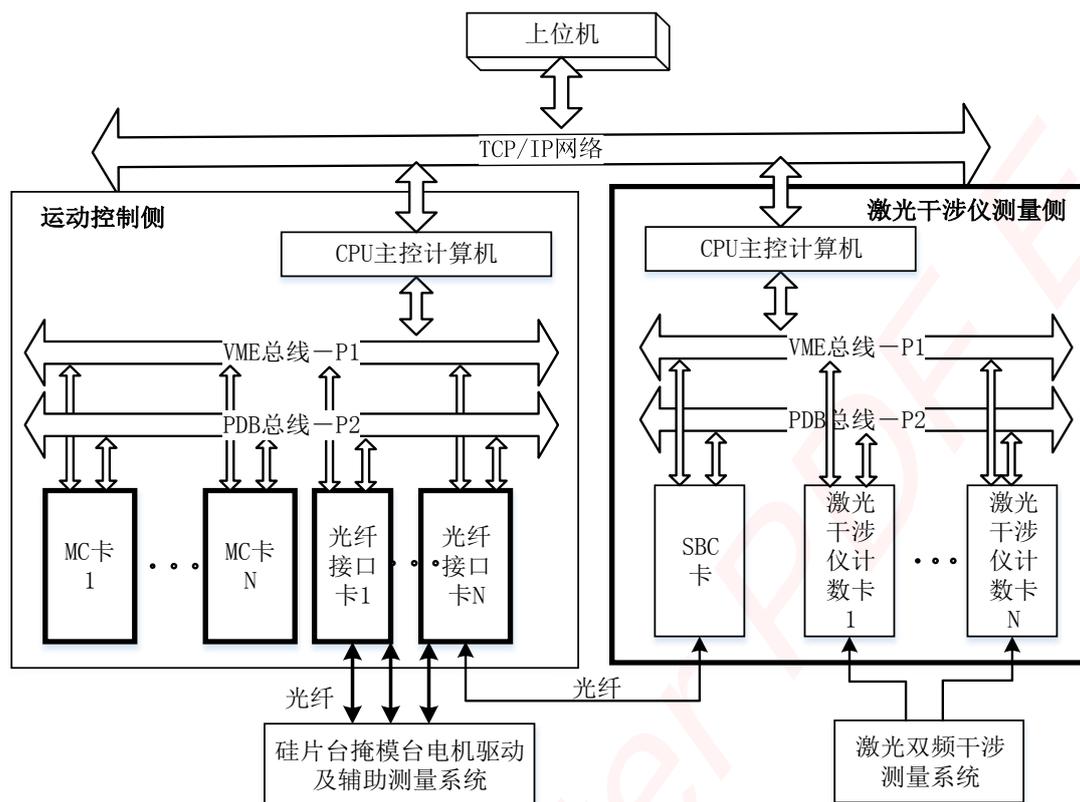


图 1-6 ASML 研制的基于 VME 总线的控制系统架构

1.4.2.3 华中科技大学等高校研制的 VME 控制系统

华中科技大学等高校研制的 VME 控制系统一部分沿袭了 SMEE 的技术方案，优点是成熟稳定可靠。如图 1-7，运动控制侧的 VME 系统完成运动控制功能，其中的运动控制卡集成有光纤接口，可直接向电机驱动模块传输控制信号。由于控制与测量信号大量增加，导致双工作台系统控制板卡很多，不能置于同一 VME 机箱，因此参考 ASML 的方案，将系统分为运动控制子系统和激光干涉仪测量子系统。在激光干涉仪测量子系统中，使用另外一套 VME 系统完成激光双频干涉信号的处理，它与运动控制侧 VME 系统之间采用高速光纤保证激光双频干涉信号实时传输到运动控制卡。在运动控制子系统中，通过带有光纤的同步总线控制卡 MBC 接收激光干涉仪测量子系统的数。同时，在 VME 总线和 PDB 总线上设计了对准系统和调平调焦测量系统的接口，使得它们的信息能够实时反馈到运动控制系统中。

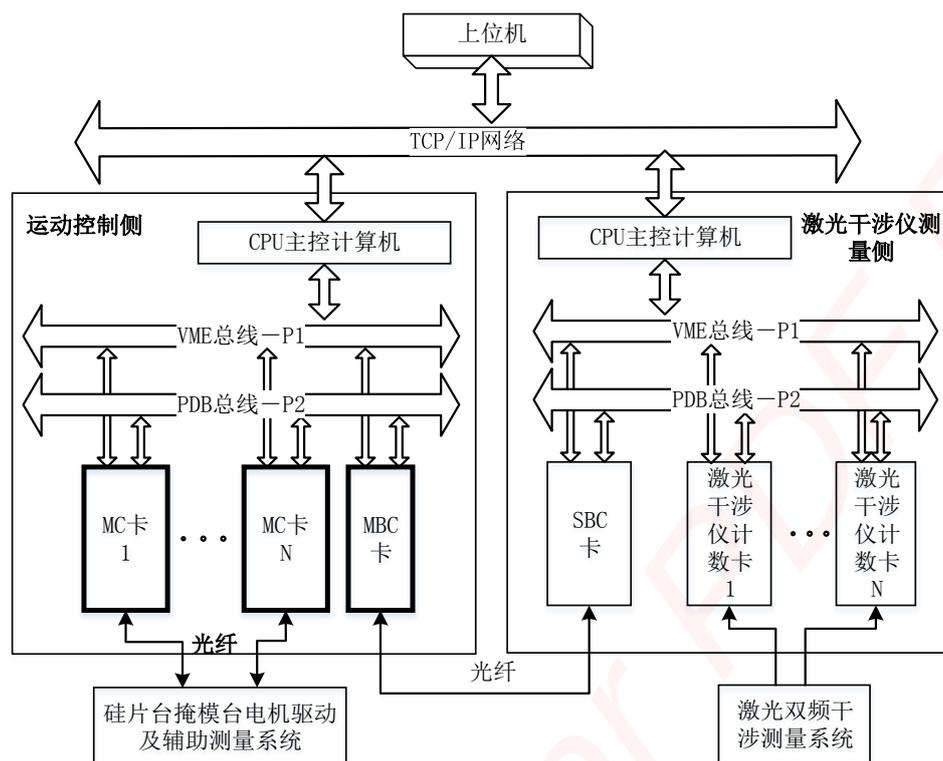


图 1-7 华中科技大学等高校研制的基于 VME 总线的控制系统架构

SMEE 的硬件结构简单，随着控制和信号大量增加，所需信号处理卡增多，单台 VME 机箱无法满足要求；ASML 方案中可直接利用通用板卡，运动控制卡 MC 仅作数据处理而无控制 I/O，硬件结构比较复杂，导致总线工作拥挤、传输效率低，占用较多时间，且控制卡数多；华中科技大学等高校的方案中，总线带宽理论上可达 320Mbit/s，但实际工作中使用效率非常低，整个控制的伺服频率可达 5KHz。

无论是 SMEE、ASML，还是华中科技大学等高校所研制的 VME 控制系统，采用的 PDB 并行总线结构导致是总线工作拥挤、传输效率低、传输稳定性与可靠性差，PDB 总线的带宽已成为高性能光刻机工件台控制系统的瓶颈。

1.4.3 基于 ATCA 总线的工件台控制系统

Delft 科技大学团队与 ASML 公司合作，对光刻机工件台控制系统提高实时性、缩短伺服周期等方面进行了较深入的研究^[29]。如图 1-8 所示，控制系统采用 ATCA 总线构架，整体层次与 1.4.2 节所述 VME 系统大体类似，Host 为主控单板计算机，运行 Linux 系统，主要负责各个板卡之间任务调度；各控制板卡（HPPC，High performance process control）负责运动控制；另外，如图 1-9 所示，还有数据采集（QHA）板卡负责数据采集。此系统与 VME 系统主要区别在于：

- 1、各控制板卡 HPPC 通过以太网（TCP/IP）协议与主控单板计算机相连。

2、各控制板卡 HPPC 与主控单板计算机均采用单核 PowerPC 作为核心处理器。

3、各控制板卡 HPPC 以及数据采集板卡均通过 RapidIO 总线协议网状互连。此系统采样频率可达 20KHz，较 VME 系统优势的地方主要体现在总线协议、核心处理器的应用、以及拓扑网络上。

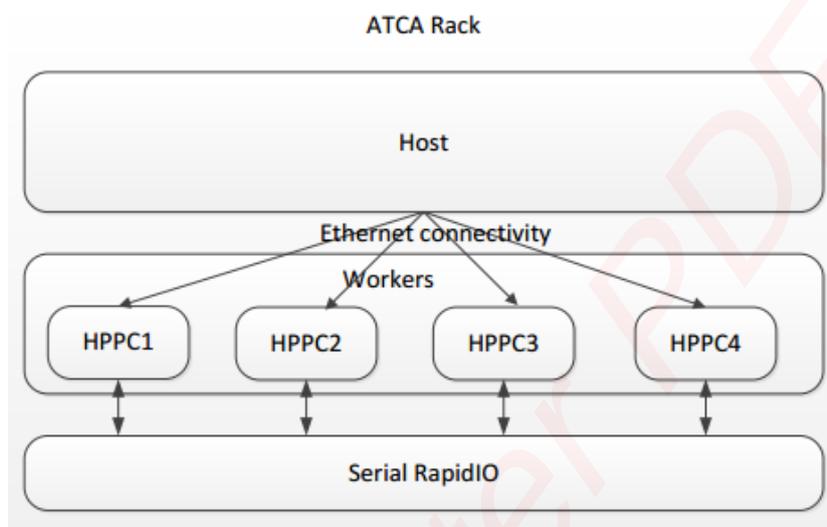


图 1-8 基于 ATCA 总线的多处理器架构图



图 1-9 基于 ATCA 总线的多处理器实物图

此文章还提出一种利用 FPGA 作为协处理器，与单核 PowerPC 协作进行计算控制，可以进一步提升采样频率。此外 Eindhoven 科技大学的学者 Raymond Mathias Wilhelmus Frijns 还提出专门利用专用指令集处理器（ASIP）作为光刻机运动控制的设计方法^[30]，以期通过这种方法达到 100KHz 的采用频率。

从 ASML 公司的光刻机运动控制技术路线以及 Delft 大学所研制的系统可知，高速总线互连高性能处理器已经是工件台控制系统的发展趋势。然而由于 VME 总

线在数据传输带宽或互连灵活性受限, ATCA 总线的应用瓶颈在于其技术或外围配件在运动控制领域中缺乏积累, 以致难以快速应用于工程产品中。因此, VPX 总线架构相比与以上两种均具有一定的优势, 更适合于光刻机工件台运动控制系统。

1.5 论文的研究内容

本文根据光刻机工件台运动控制需求, 研究基于 VPX 总线的运动控制系统架构, 结合高性能多核 DSP 处理器, 设计一种多核并行处理、强实时性以及高可靠性的光刻机工件台运动控制系统。论文的主要工作与各章的内容安排如下:

第一章介绍本课题研究背景及意义, 对光刻机及其工件台、多轴运动控制系统发展和工业计算机总线进行概述, 指出 VPX 总线相比于 CPCI、VME、ATCA 总线在工业控制领域中的优势, 然后对比和分析国内外光刻机工件台运动控制系统, 最后引出本课题的研究内容。

第二章首先对工件台驱动和传感及测量部分进行介绍, 描述当前基于 VME 总线的工件台控制系统所存在的问题和系统设计需求, 通过对比分析 VPX 体系中的多种总线, 明确多处理器互连总线及关键硬件类型, 建立以标准 VPX 总线构架为基础, 以 RapidIO 总线为多处理器互连方式的主从式运动控制系统体系结构。

第三章从工件台运动控制系统硬件体系结构出发, 详细设计系统数据流结构, 并在基础上提取并行处理模型, 分析归纳系统数据交互内容及其交互方式。为保证多核与外设正确的进行数据交互, 阐述 C6678 的数据 Cache 和共享内存的数据一致性原理。根据系统数据交互需求, 规划共享内存和多核本地内存的数据存储空间, 并结合一致性原理, 建立一套基于共享内存的数据并行交互机制。

第四章结合多核 DSP 中断子系统结构和中断映射原理, 根据系统同步要求, 设计并分析比较基于 Direct I/O + Doorbell 软中断和基于 Direct I/O + GPIO 硬中断两种测控方案的实时性。针对选定的测控方案, 结合部分硬件原理图及 C6678 中断映射原理, 详细介绍基于 GPIO 中断的多 DSP 同步方法以及两种实现 C6678 多核同步的具体实施方法, 将系统数据流结构简化成最小测控模型, 依据同步技术方法, 对系统进行时序最优化设计及分析。

第五章介绍工件台控制系统总体软件框架, 并以硅片台、掩模台为对象, 着重对 DSP 伺服控制软件进行初步开发, 包括多核任务规划、多核 DSP 程序设计流程。为了验证系统实时性和可行性, 以最小测试系统为实验平台, 对第四章中基于两种模式的同步测控方案进行实时性测试, 最后对多核同步响应延时以及伺服中断程序总体执行时间进行测试。

第六章是论文的研究结论和对下阶段研究的展望。

第二章 基于 VPX 总线的工件台运动控制系统总体方案设计

作为光刻机核心单元之一，超精密工件台主要负责实现快速扫描、上下片、精密定位、调平调焦等功能。目前，较为成熟的方案大多采用 VME 并行总线架构来建立超精密工件台控制系统，由于随着系统性能要求的提升，VME 总线以及相应的处理器已无法满足需求，所以必须设计一种新型工件台控制系统。

本章的工作主要是首先分析光刻机工件台的整体设计需求，在此基础上介绍互连总线及关键硬件类型，从多轴运动控制系统的实时性、同步性、并行性、可扩展性等要求出发，以 VPX 总线构架为基础，提出一种基于 RapidIO 总线互连架构的新型工件台运动控制系统方案。

2.1 系统设计需求分析

2.1.1 工件台系统总体介绍

如图 2-1 所示，步进扫描投影型光刻机的工件台主要由测量硅片台、曝光硅片台和掩模台三大部分构成，硅片台和掩模台之间为透视系统。在曝光过程中，控制系统需要对硅片台、掩模台通过激光干涉仪进行高精密度位置测量，并对多个自由度进行控制与调节，实现工件台精确定位运动。

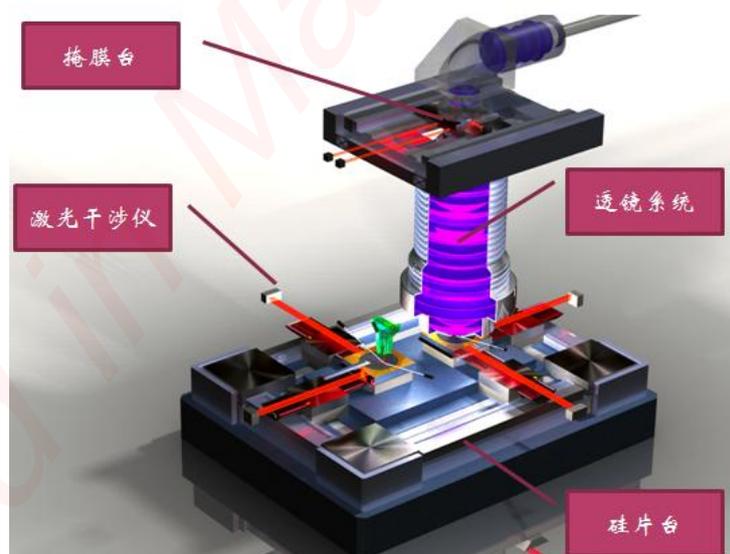


图 2-1 工件台模型图

图 2-2、图 2-3 分别为实验室 α 样机硅片台、掩模台的总体结构，均采用粗精叠层结构，由粗动台、微动台、线缆台以及平衡块组成。硅片台粗动台沿 X、Y 方

向以及绕 Z 方向运动的驱动力由 4 个相同线圈阵列供给，每个线圈阵列均由三个线圈单元组成；微动台驱动系统包括 2 个 X 向的动铁式音圈电机、2 个 Y 向的动铁式音圈电机以及 4 个 Z 向的动铁式音圈电机；平衡块四角上均安装了防漂移电机，据统计，测量硅片台和曝光硅片台系统所需驱动轴数共达 32 个。

同样，掩模台系统中微动台也由 2 个 X 向音圈电机、2 个 Y 向音圈电机和 4 个 Z 向音圈电机驱动；粗动台驱动系统包含 2 个 Y 向的直线电机；掩模台平衡块有 2 个防漂移电机。因此，掩模台系统共需驱动的轴数为 12 个。经过最终统计，工件台系统需驱动的轴数总共为 44 个，具体统计如表 2-1、表 2-2 所示。

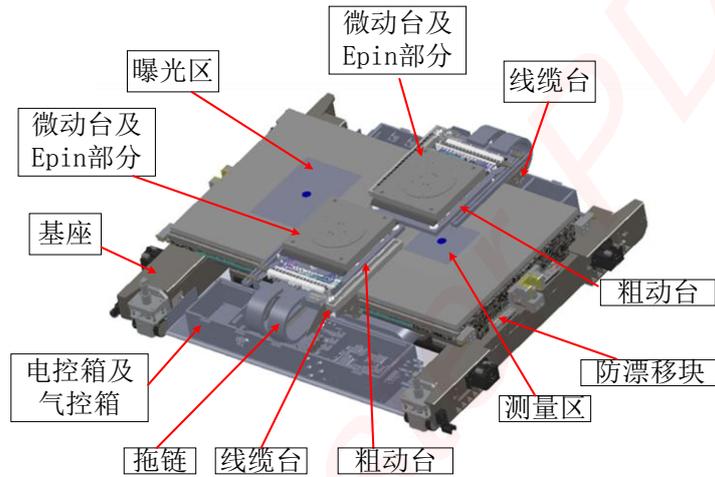


图 2-2 α 样机硅片台结构布局图

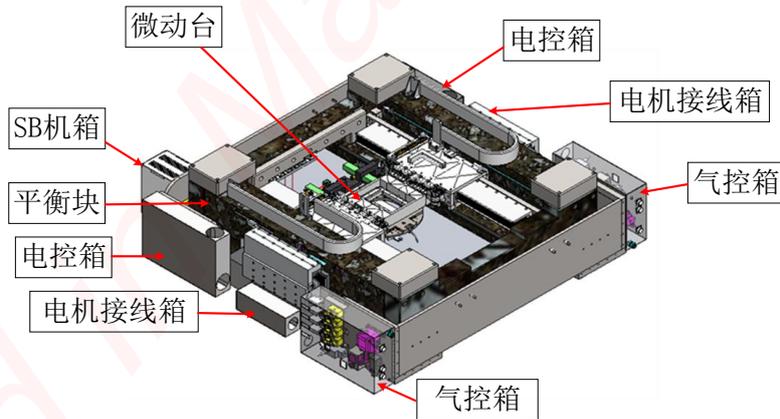


图 2-3 α 样机掩模台结构布局图

硅片台、掩模台的测量系统包括微动台绝对位置反馈测量、微动台与粗动台相对位置局部闭环测量、粗动台与线缆台相对位置测量、线缆台与平衡块相对位置测量、平衡块位置测量和相关的电气限位信号测量。在硅片台的测量系统中，微动台绝对位置反馈测量由 18 轴激光干涉仪实现；微动台与粗动台相对位置局部闭环测

量采用 3 个 PSD 传感器；粗动台与线缆台相对位置测量采用 2 个 PSD 传感器；线缆台与平衡块相对位置测量由 2 个光栅编码器采集完成；平衡块相对地面位置信息则由 4 个绝对光栅编码器采集；平衡块电气限位信号 8 个。

在掩模台的测量系统中，微动台绝对位置反馈测量由 9 轴激光干涉仪实现；微动台与粗动台相对位置局部闭环测量由 8 个电涡流传感器完成；粗动台与平衡块相对位置采用 2 个绝对编码器采集；平衡块相对地面位置通过 2 个绝对光栅编码器采集；粗动台电气限位信号和平衡块电气限位信号共 8 个。系统所需采集的信号数统计如表 2-3、表 2-4 所示。

表 2-1 硅片台电机轴数统计表

名称	数量	功能
A 台微动台音圈电机	8 个	驱动 A 台微动台六自由度运动
A 台粗动台直线电机	4 个	驱动 A 台粗动台三自由度运动
A 台线缆台直线电机	1 个	驱动 A 台线缆台运动
A 台滑块旋转电机	1 个	驱动 A 台滑块运动
B 台微动台音圈电机	8 个	驱动 B 台微动台六自由度运动
B 台粗动台直线电机	4 个	驱动 B 台粗动台三自由度运动
B 台线缆台直线电机	1 个	驱动 B 台线缆台运动
B 台滑块旋转电机	1 个	驱动 B 台滑块运动
平衡块电机	4 个	驱动工件台平衡块运动
电机总数	32 个	——

表 2-2 掩模台电机统计表

名称	数量	功能
微动台音圈电机	8 个	驱动掩模台微动台六自由度运动
粗动台直线电机	2 个	驱动掩模台粗动台运动
平衡块直线电机	2 个	驱动掩模台平衡块运动
电机总数	12 个	——

表 2-3 硅片台传感器统计表

名称	数量	功能
A 台微动台 PSD 传感器	3 个	A 台微动台与粗动台相对位置测量
A 台粗动台 PSD 传感器	2 个	A 台粗动台相对线缆台、滑块位置测量
A 台绝对光栅编码器	2 个	A 台线缆台、滑块相对平衡块位置测量
B 台微动台 PSD 传感器	3 个	B 台微动台与粗动台相对位置测量
B 台粗动台 PSD 传感器	2 个	B 台粗动台相对线缆台、滑块位置测量
B 台绝对光栅编码器	2 个	B 台线缆台、滑块位置相对平衡块测量
平衡块绝对光栅编码器	4 个	平衡块相对地面位置测量
平衡块电气限位	8 个	粗动台、平衡块最大和最小行程限位
微动台激光干涉仪	18 轴	微动台绝对位置测量

表 2-4 掩模台传感器统计表

名称	数量	功能
微动台电涡流传感器	8 个	掩模台微动台与粗动台相对位置测量
微动台激光干涉仪	9 轴	掩模台微动台绝对位置测量
粗动台绝对光栅编码器	2 个	掩模台粗动台相对于平衡块位置测量
粗动台电气限位	4 个	掩模台粗动台左侧和右侧最大和最小行程限位
平衡块绝对光栅编码器	2 个	掩模台平衡块相对地面位置测量
平衡块电气限位	4 个	掩模台平衡块左侧和右侧最大和最小行程限位

2.1.2 系统设计问题分析及需求提出

为保证工件台控制系统的实时性，系统中数百个传感器反馈数据须实时传输到运动控制器，而运动控制器必须要在规定的伺服周期内完成 44 轴的位置解算、精密轨迹插补、位置-加速度误差补偿、电机解耦等控制算法以及误差处理、控制过程数据保存等任务，最后将伺服运算结果输送到各个电机的驱动器。

目前步进扫描光刻机工件台控制系统广泛采用 VME 总线构架，由于工件台控制系统所需控制的电机数量大量增加，而且反馈信号也大量增加，如果只采用一台工控机来处理所有的信号，导致系统数据负载量大、结构复杂，一台机箱无法容纳

众多的控制卡和信号处理卡。针对这种情况，控制系统被分为运动控制子系统和激光数据采集子系统，分别嵌入于运动控制侧与测量侧两个 VME 机箱中。运动控制子系统的 VME 机箱中插有多块运动控制卡 (MC)、单板计算机卡和一块带有光纤接口的主同步总线控制板 (MBC)。MC 卡之间通过 VME 机箱的 P2 口自定义的非复用同步位置数据总线 (PDB) 进行数据传输；MBC 接收来自激光数据采集子系统发过来的硅片台微动台、掩模台微动台的位置数据；VME 总线作为单板计算机与运动控制卡和 MBC 之间数据传输的通道，用以 MBC、MC 初始化，MC 固件下载和命令传输等。针对以上 VME 控制系统，归纳以下几点在进行系统设计中的问题及需求：

1) 多处理器之间数据传输实时性

VME 系统利用机箱背板的 P2 口自定义的非复用同步位置数据总线 (PDB) 作为多块 MC 运动控制卡之间高速数据传输的通道，用以在每个中断伺服周期 (200us) 内传输实时的伺服数据和状态信息。由于 PDB 总线的数据传输带宽理论带宽为 320Mbps，据统计，系统数据传输量在 800B-4KB 范围内，为了提高伺服系统采样频率，增强系统实时性，总线带宽必须有所提高。

2) 处理器运算性能

VME 系统中 MC 卡集成单核 TMS320C6713b 处理器，其主频性能只有 300MHz，现在 DSP 处理性能已超过 1GHz，因此提高处理器主频可使得控制算法计算时间缩短，进而提升系统的实时性。

3) 光纤接口数量

在 VME 系统中，每个 MC 卡集成了相应的光纤接口，都能进行独立的数据采集，由于一块 MC 卡上光纤口资源有限，导致出现以下复杂情况：

情况 i：例如，微动台的伺服计算由一块 MC 完成，由于单块 MC 卡光纤口有限，无法采集该微动台的所有传感器反馈信号，那么微动台某一个反馈信号则必须由另一块 MC 卡采集完成后经过 PDB 总线传输过来，增加了不必要的总线数据传输环节；

情况 ii：除了采集问题，一块 MC 卡在每个伺服周期内控制微动台的 8 个电机，光纤口的不足致使此块 MC 卡须将相应的电机控制量通过 PDB 总线传输至另一块 MC 卡，由另一块 MC 卡的光纤口将最终的电机控制量发送至驱动器。

虽然单 DSP 单光纤口具有模块化的功能，但所面临的系统环境复杂，无法更优的应用于当前系统中，使得系统数据流不合理、凌乱。

4) 系统计算架构

VME 运动控制系统在数据计算方面采用多个单核 DSP 并行处理模式，面对众

多控制对象和传感器信号，一台 VME 机箱无法容纳更多的控制卡和信号处理卡。随着多核 DSP 技术的快速发展，多个多核 DSP 并行计算模式有待应用，不仅可以精简整个控制系统，而且提高编程效率。

因此，为了提高光刻机工件台的定位、跟踪精度以及生产效率，控制系统必须在更短的伺服周期内完成大量的位置数据采集、数据传输、控制计算和控制量输出等任务。然而，基于 VME 总线的工件台运动控制系统瓶颈主要在于总线数据吞吐速率、网络拓扑、数据处理能力以及接口局限等多方面原因。因此亟待构建一种基于 VPX 总线的多核处理器并行处理架构，并以此为基础开发工件台控制系统的硬件和软件，试图将伺服周期缩短至 100us。

2.2 系统互连总线及关键硬件

2.2.1 互连总线选择

依据工业计算机总线的发展概况以及对现有的 VME 控制系统现存问题和需求的分析，与 ATCA 总线构架相比，将 VPX 总线构架应用于光刻机工件台控制系统中更具性价比。

2.2.1.1 多种总线比较分析

在 VPX 多总线体系结构中，RapidIO、PCIe、GbE 总线都能达到 10Gbps 的传输速率，但 RapidIO 总线比 PCIe、GbE 更适合组建大规模的多处理器互连系统，实现大量处理器之间的数据高速通信；PCIe 实际上已经成为核心处理器与外围设备之间数据高速传输应用的标准；而 GbE 是基于 IP 数据通信的标准。RapidIO 凭借总线协议的低开销、高兼容性等特性以及拥有更高的可靠性、更高的传输带宽和更快的总线速率等优势，可满足光刻机工件台控制系统对多处理器互连、强实时性及高数据吞吐率的需求。

2.2.1.2 RapidIO 总线协议介绍

RapidIO 协议体系结构包含逻辑层、传输层和物理层，如图 2-4 所示。逻辑层规范处于最顶层，主要负责对所有协议和数据包格式进行定义；传输层主要定义了端点设备之间包交换、路由和寻址机制等信息；物理层则定义了设备级接口细节，如包传输机制、电气特性、流量控制和低级错误管理等^[31,32]。

串行 RapidIO 支持全双工传输，可以独立收发信号。标准的 1x/4x 串行 RapidIO 接口可支持四个口，每一个串行 RapidIO 口由 4 根信号线组成，因此，共需 16 根信号线。这四个口可以分离为多个独立的接口以传输不同的数据，也可通过合并成

为一个接口以提高数据吞吐量^[33]。随着 RapidIO 标准协议的不断改进,目前 RapidIO 速率已从 RapidIO 1.x 标准支持的 1.25GHz 提升到 RapidIO3.x 标准所支持的 10.3125GHz。

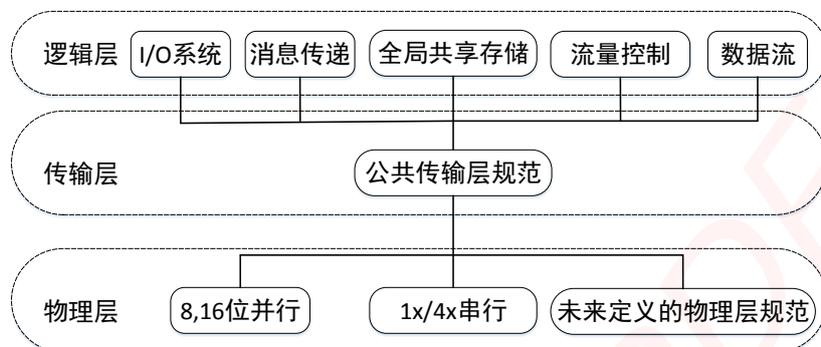


图 2-4 RapidIO 协议体系结构

2.2.1.3 RapidIO 网络拓扑结构

基于 RapidIO 总线的多处理器互连方式非常灵活, RapidIO 所支持的拓扑结构多种多样,如图 2-5 所示,多个处理器既可以直观点对点连接,可以通过交换机互连。

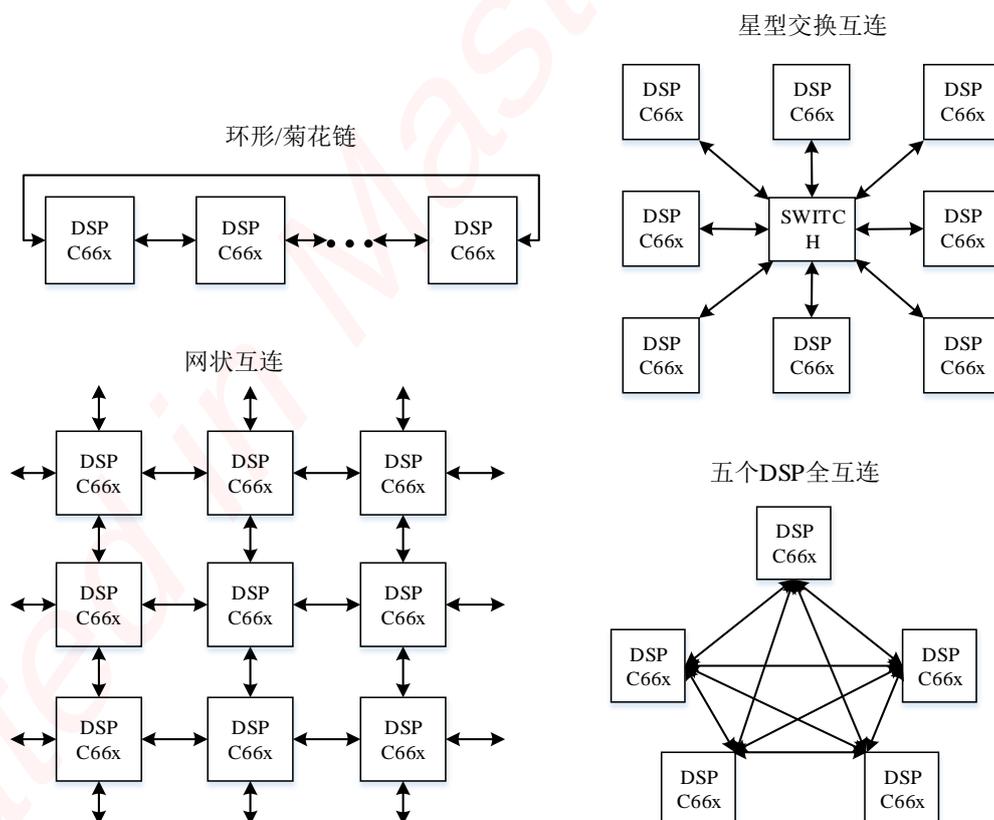


图 2-5 多种 RapidIO 互连拓扑结构

2.2.2 运动控制卡 (MC_4DSP_VPX)

除了总线带宽之外,运动控制卡核心处理器的性能也对工件台控制系统伺服周期影响较大,目前在工业控制领域中,单片机、ARM、DSP、FPGA 和 PowerPC 等一些微处理器比较常用。相比较而言,单片机主要面向于低端市场;ARM 在任务管理、软件系统支持等方面具有优势;FPGA 虽然在理论上可以达到更高频率和信号处理速度,可是编译复杂,开发周期长,由于 FPGA 具有超强的逻辑编程灵活性和并行处理资源,故只作为协处理器用在接收发送信号端;ASML 公司采用的单核 PowerPC 成本高且存在技术封锁问题,而且经过前期在 PowerPC 上的 RapidIO 带宽测试,通信延迟较大,不符合系统设计要求;DSP 具有杰出的大数据处理能力,而且随着处理数据量的增加,单核 DSP 芯片发展越来越不满足系统要求,传统嵌入式系统是增加并行处理板上单核 DSP 芯片数量^[34]或者增加单板数量,这样会导致单板面积增大、I/O 不足或机箱内板卡数过多等问题,因此多核 DSP 在运算速度、系统精简度和开发难度上有较好优势,并且依靠上一代单核 DSP 系统,具有一定的可移植性。通过对系统需求以及处理芯片的性能和价格等方面的比较和分析,本文选取 TI 公司的 TMS320C6678 这一款多核 DSP 作为运动控制板卡的核心处理器。

1) 多核 DSP 处理器 (TMS320C6678) 简介

C6678 是 TI 公司于 2010 年推出的 C66x 系列中的 8 核浮点型 DSP,集成了 8 个高性能定点(32MAC)/浮点(16FLOP) CPU 内核,每个内核最高能以 1.25GHz 的工作频率运行,整片 C6678 处理器的定点、浮点运算能力分别可达 320GMAC、160GFLOP。C6678 每个核具有 32KB 的 L1P (一级程序存储器)、32KB 的 L1D (一级数据存储器)和 512KB 的 LL2 (二级局部存储器),片内集成有 4M 共享 SRAM,通常用于核与外设、核与核之间的数据交互空间。除此之外,C6678 还提供 DDR3 接口,可挂载 DDR3 的容量最大可达 8GB。C6678 通过采用新一代多核处理器 KeyStone 架构高效地将 RISC、DSP 内核、协处理器和高速 IO 结构集成在一起,如图 2-6 所示,而且由于 KeyStone 架构提供了充足的内部带宽,DSP 内核、外设 (Hyperlink 接口、PCI Express Gen2、4x 的 RapidIO 及其他外设)、协处理器和 IO 接口之间实现高速访问^[35]。

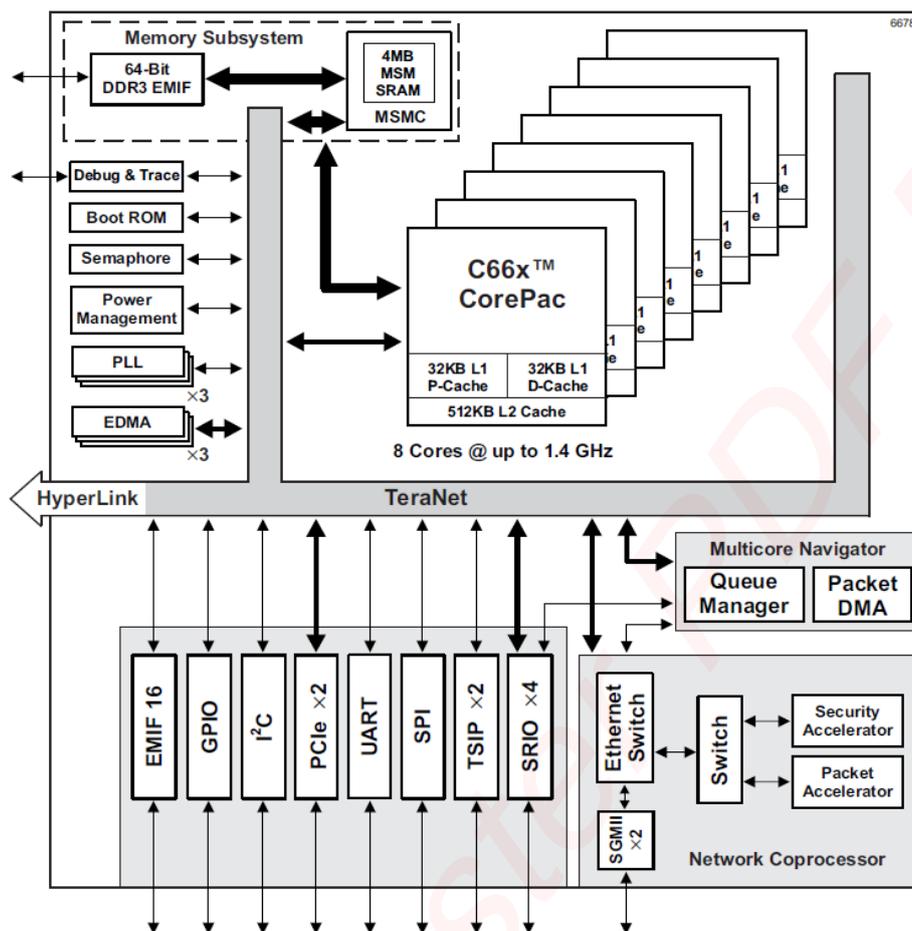


图 2-6 基于 KeyStone 架构的 TMS320C6678 内部框架

2) 运动控制卡 (MC_4DSP_VPX)

运动控制卡 MC_4DSP_VPX 是一块通用的 6U VPX 板卡。考虑到系统的扩展性，例如后期需要融入对准系统，曝光系统等子系统，单板采用集成 4 片 C6678 和 1 片 K7 FPGA 的方案，增加系统处理器的冗余度。每片 C6678 芯片可外挂 4GB 的 DDR3，单板总存储容量可高达 16GB，实现海量数据的存储，可使用于大规模实时信号处理。单板内部总体框图及实物图如图 2-7、图 2-8 所示。

运动控制卡 MC_4DSP_VPX 部分资源如下所述：

- 板载 4 片 C6678 处理芯片，工作频率最高为 1.25GHz，一般保守设置为 1GHz，2 片 C6678 芯片作为 1 个处理节点，其间通过互连总线 Hyperlink 实现通信；
- 每片 C6678 和 FPGA 与 RapidIO Switch 的互连分别通过 1 路 X4 RapidIO 接口和 2 路 X4 RapidIO 接口实现，RapidIO Switch 提供 4 路 X4 RapidIO 接口，与 VPX 背板的 P1 口相连，则多块板卡则通过背板的 RapidIO 总线和 P1 口实现数据传输。

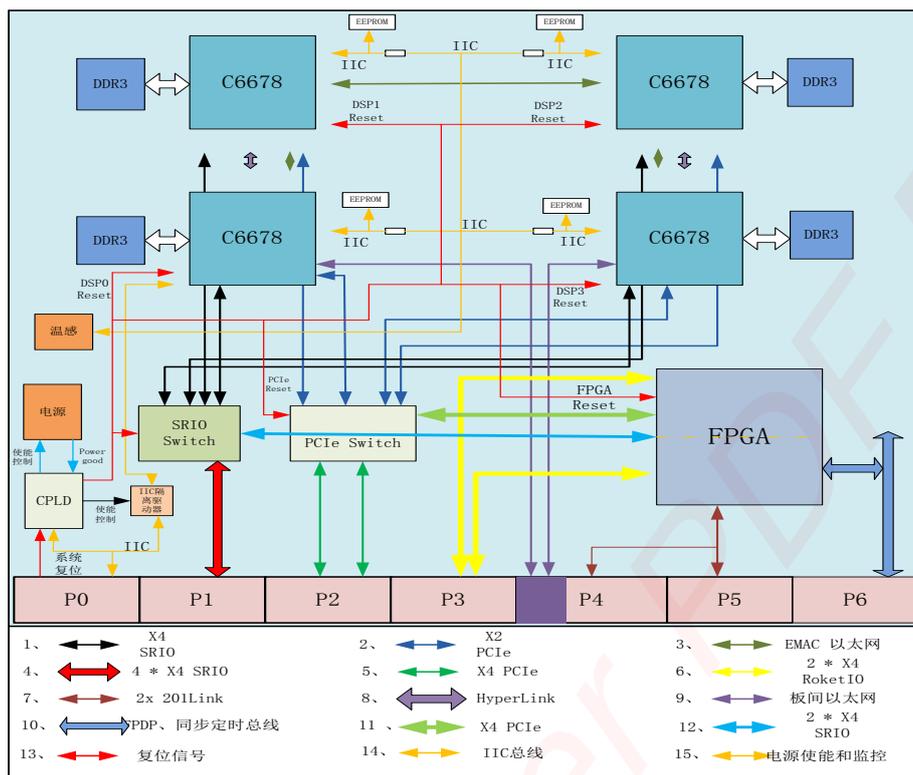


图 2-7 运动控制卡 MC_4DSP_VPX 内部总体框图



图 2-8 运动控制卡 MC_4DSP_VPX 实物图

2.2.3 光纤接口卡 (FC_FPGA_VPX)

正如 2.1.2 节中提到的光纤接口数量受限问题，本文方案拟将光纤接口与运动

控制处理卡分离，所有的光纤接口集成于单块板卡。光纤接口卡（FC_FPGA_VPX）由硬件公司根据实验室需求完全定制设计，板上使用 1 片 K7 FPGA 实现光纤接口的管理。常用接口包括：前面板 12 个协议可定义光口，同步信号输入输出接口、JTAG 接口等。参考方案框如图 2-9 所示：

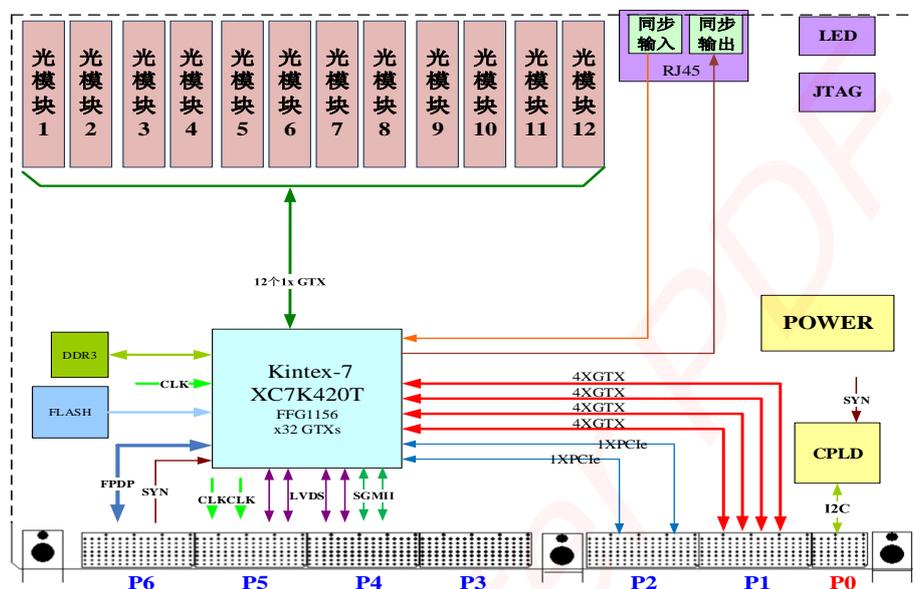


图 2-9 光纤接口卡 FC_FPGA_VPX 总体方案图

光纤接口卡 FC_FPGA_VPX 部分资源如下所述：

- P1: K7 FPGA 提供 4 个 x4 GTX;
- P2: K7_FPGA 提供 2 个 x1 PCIe;
- P4: K7 FPGA 提供多路 LVDS 差分对, 2 个 Ethernet;
- P5: K7 FPGA 提供多路 LVDS 差分对, 并根据要求定义;
- P6: K7 FPGA 提供系统 SYN 信号及 32bit 自定义总线;
- 前面板提供 12 路光纤, 采用 SFP 接口形式, 提供 LED 指示通断;
- 前面板提供 1 路输出同步定时信号, 1 路输入同步定时信号, 采用 RJ45 接口形式;

2.2.4 主控卡 (HOST_CPU_VPX)

主控卡 HOST_CPU_VPX 拟选用 PPC (TBD) 结构, 根据项目要求所提的部分资源需求描述如下:

- 主控单元
 - CPU 采用高性能 PowerPC;
 - 内存: 8 GB DDR3 (1600 MHz);

- 硬盘采用 256G 固态硬盘;
- 1 片 PCIE 转 RapidIO Switch;
- 支持 VxWorks6.8 操作系统;
- 对外接口
- 前面板 3 个 USB2.0 接口;
- 前面板 2 个 RJ-45 输出千兆以太网;
- 前面板 1 个 VGA 视频输出接口;
- 背板 2 个 4XRapidIO。

2.2.5 VPX 机箱背板

VPX 机箱背板是系统数据传输链路，是各功能模块之间的数据交互的基础，承载着系统全部功能模块，除了给各个模块供电，还为各模块间的数据高速传输提供通道，并为外接信号提供接口等。根据拓扑结构，VPX 背板可以是星型、双星型、菊花链型和孔网型等。

实验室所定制的 VPX 机箱采用 6U 加固式机箱，背板共有 6 个槽位，包括 1 个电源槽和 5 个处理模块槽，其中，3 槽为主控槽，1,2,4,5 槽为负载槽。背板各槽的连接方式如图 2-10 所示。

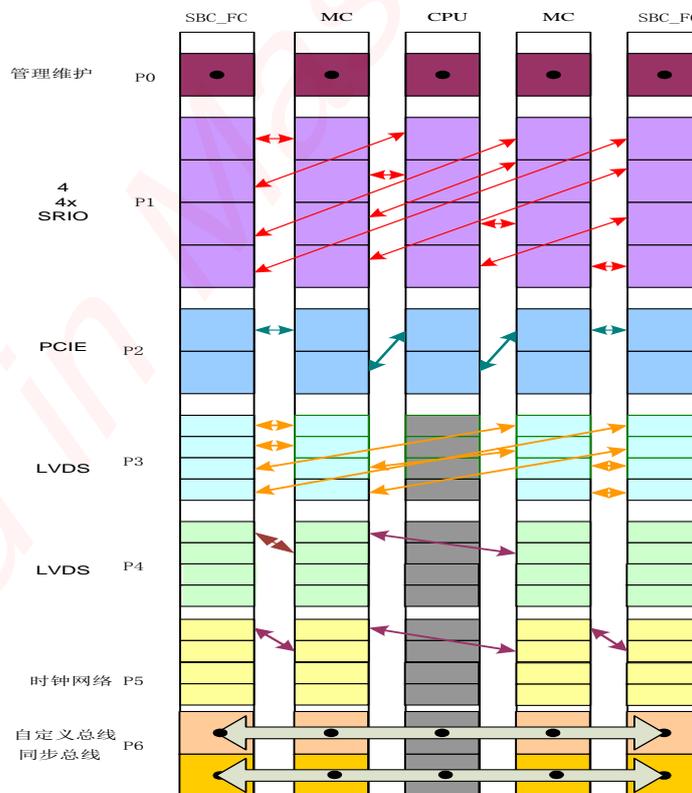


图 2-10 VPX 背板各槽连接方式

VPX 机箱背板部分资源描述如下所述:

- 主控槽 P0, P1, P2 与其它板卡互连, P3, P4, P5, P6 与其它板卡不互连, 预留为后插卡;
- P0: 主电源采用+12V 供电、+5V 供电、系统管理和复位等;
- P1: 4 个 4X RapidIO 组成 FULLMESH 拓扑;
- P4 上每个负载槽有 2 路千兆以太网;
- P4 和 P5: 可作为 4 个 4XTS201LINK 接口使用, 也可作为 32 对高速 LVDS 信号使用;
- P6 上定义了 32bit 自定义总线和 23bit 同步单端定时总线。

2.3 基于 RapidIO 总线的控制系统硬件体系结构设计

2.3.1 控制系统总体层次结构划分

由于工件台整机系统非常复杂, 涉及的传感器和电机轴特别多, 因此通过采用主从结构, 对系统进行分层设计, 分析各部分的关联程度, 将不同的部分合理部署在相应的层次中。如图 2-11 所示, 光刻机工件台整体运动控制系统共规划了五层, 从上之下依次是总控制层、分系统层、运动控制层、IO 接口层和传感器执行器层, 各层控制结构的具体功能^[36]如下:

1) 总控制层即上位机监视器, 不仅提供人机交互界面, 进行运行操作、测试诊断、数据记录与存储、数据分析等工作, 而且通过与分系统层进行数据交互, 实现对各个分系统的工作指令调度及协调管理, 完成工件台的工作流程。

2) 分系统层主要包含工件台运动控制分系统、对准分系统等各个分系统的主控制器, 负责对各个分系统的软硬件管理及任务调度等。在工件台运动控制分系统, 主控卡为 Power PC 板卡, 其工作在主模式, 对工件台运动控制层的从设备进行管理与调度。通过以太网 (Ethernet) 与上位机通信, 实现命令接收、命令解析、任务调度以及上传数据等任务。

3) 运动控制层包含有工作台运动控制卡、光纤接口卡和激光计数卡, 相对于主控卡 HOST_CPU_VPX, 工作在从模式, 主要负责各控制台运动轴的闭环伺服控制。其中, 运动控制卡主要实现相关伺服运算以及与分系统层通讯完成命令解析与功能执行; 光纤接口卡作为数据交互中心, 为各种数据提供的外围接口, 负责位置数据的接收, 数据的传输, 计算结果的输出; 激光计数卡则同步采集激光数据。

4) IO 接口层由功率放大器控制卡 (驱动器) 和传感器卡组成。功率放大器控制卡通过高速光纤接收运动控制层的控制指令, 并转化成电流, 实现对电机的驱动, 而传感器卡将采集的多种传感器信号通过光纤上传至运动控制层。

5) 传感器执行器层的设备主要是各种传感器和电机，负责实现物理信号（位置、温度、力等）的采集以及电机的驱动。

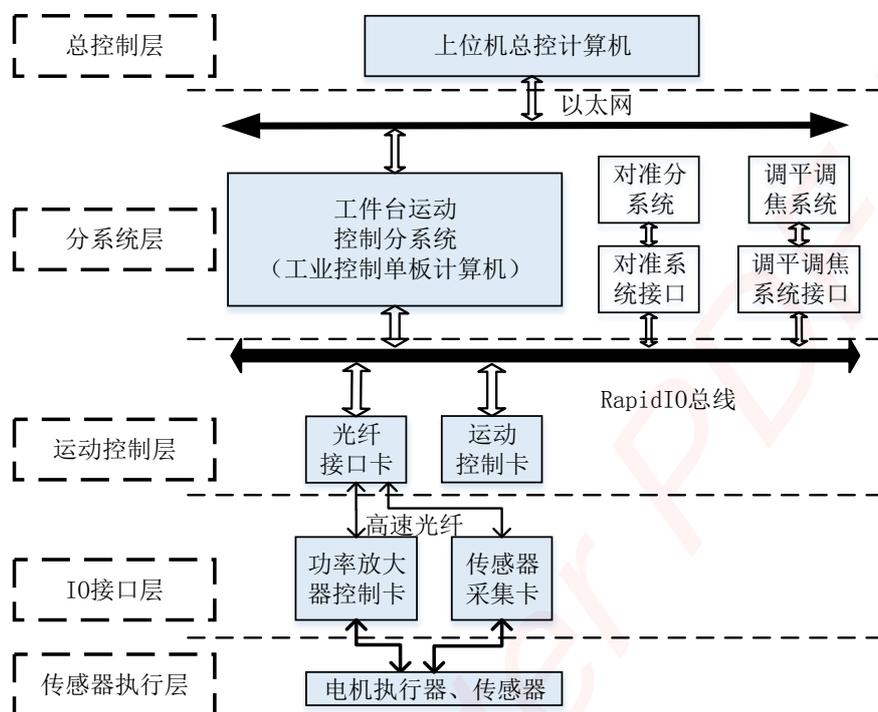


图 2-11 光刻机工件台控制系统分层结构

2.3.2 基于 RapidIO 总线的控制系统架构

考虑到目前 DSP 性能的极大提高，特别是保持与前期工作的兼容性，充分利用已有成果经验，避免较大幅度的方案变动，总控制层、IO 接口层和传感器执行器层基本沿用了 VME 系统的部分技术方案，依旧使用一台 VME 系统完成激光双频干涉信号的处理，与运动控制侧之间采用高速光纤连接。

运动控制子系统则根据控制系统分层结构，并针对所提出的问题和需求，采用如图 2-12 所示的结构方案。方案中运动控制侧机箱采用 VPX 系统，并结合 ASML 和华中科技大学等高校研制的基于 VME 总线的运动控制系统架构的优点，将 RapidIO 总线作为运动控制卡 MC_4DSP_VPX 和光纤接口卡 FC_FPGA_VPX 互连通信网络，具体优点体现在以下几点：

- 1) 运动控制卡 MC 采用高性能的多核处理器 TMS320C6678，一方面提高了主频，另一方面多 DSP 多核大规模并行处理模式极大增强了系统的运算能力。
- 2) 运动控制卡 MC 采用 DSP 处理器与光纤接口模块分离式设计，独立的光纤接口卡增加了光纤接口数量以及提高了连线灵活性，避免了系统数据流不合理现象。
- 3) 将 VME 总线和 PDB 总线替换成 RapidIO 总线和 UDB 总线，运动控制卡和光

纤接口卡之间采用 RapidIO 互连通信，由于总线带宽极大提升以及，无须担心总线拥堵等问题，而且通过 UDB 总线自行构建合理的时钟网络。

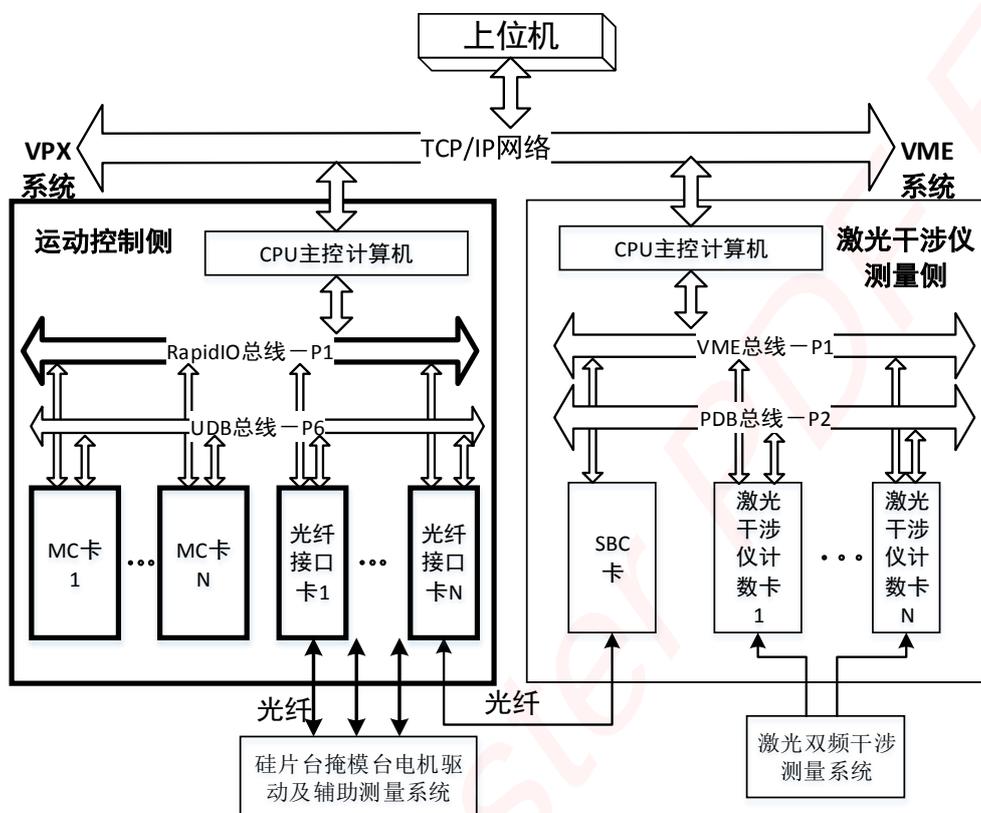


图 2-12 基于 RapidIO 总线的工件台控制系统架构

2.3.3 工件台控制系统硬件体系结构详细设计及功能分析

基于 RapidIO 总线架构的光刻机工件台系统总体硬件体系结构如图 2-13 所示。在运动控制侧，VPX 机箱背板共 5 个槽位，部署了 1 块 HOST_CPU_VPX 卡、2 块 MC_4DSP_VPX 卡和 2 块 FC_FPGA_VPX 卡，满足了目前系统大运算量、大吞吐量、灵活性等需求。每一个槽位作为一个节点，利用 VPX 背板 P1 口物理介质，与其他 4 个节点实现 4X 全双工互连通信，构成 RapidIO 全网状星型拓扑结构，为板间通信提供数据传输通道，其背板数据通道互连如图 2-14 所示。每块 MC_4DSP_VPX 卡板内置有 RapidIO 交换器，通过 RapidIO 交换器和背板 RapidIO 物理链路可实现 DSP 与 DSP、DSP 与 FPGA 之间的数据通信和命令传输等。

分系统层的主机为 HOST_CPU_VPX 卡，以 PowerPC 处理器为核心，具有强大的任务管理等功能和软件系统支持，负责上位机命令的解译和分配，其通过 TCP/IP 网络与上位机进行通信；并采用 RapidIO 总线互连 MC_4DSP_VPX 卡、FC_FPGA_VPX 卡。

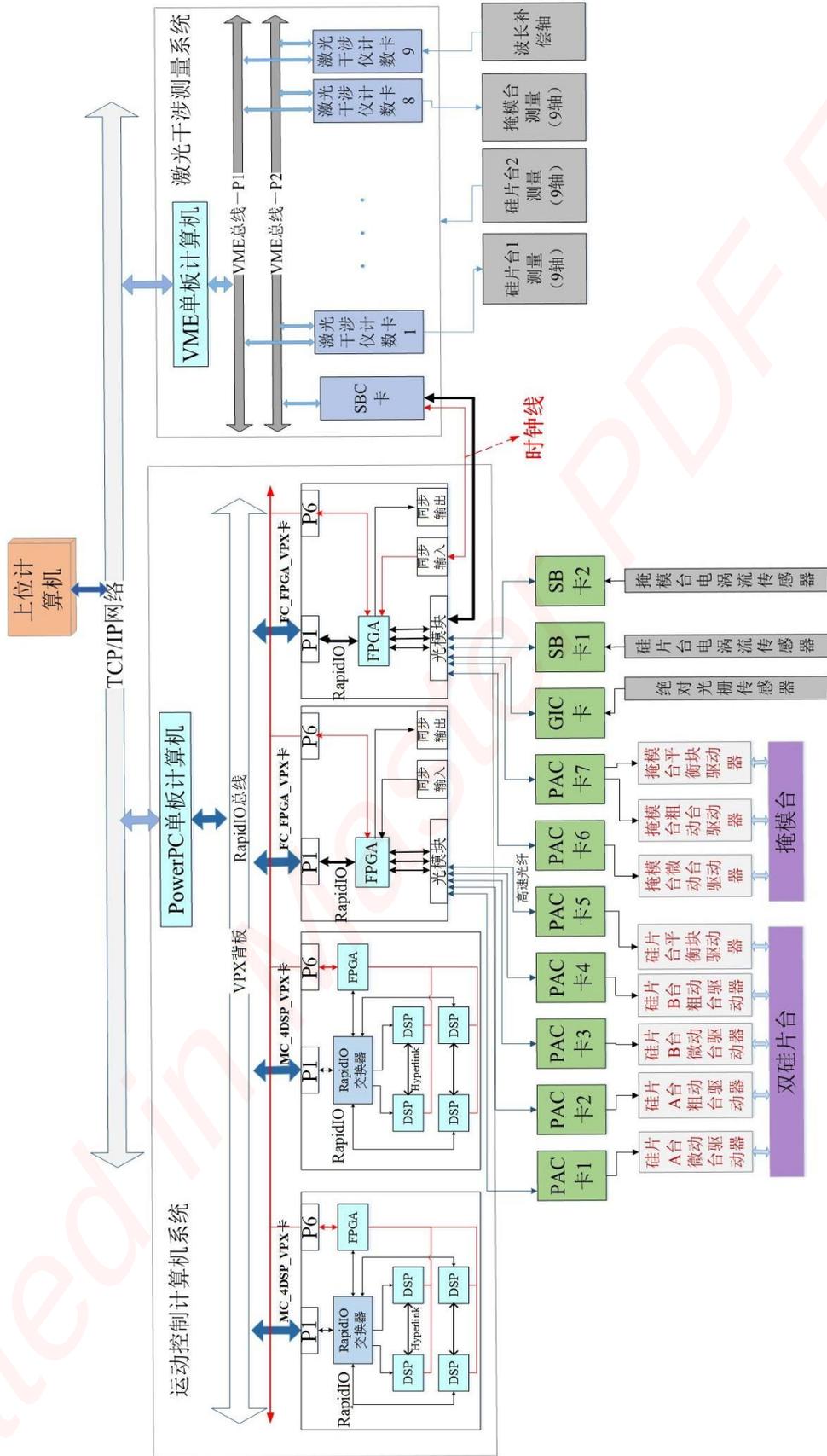


图 2-13 基于 RapidIO 总线的工件台控制系统总体硬件体系结构

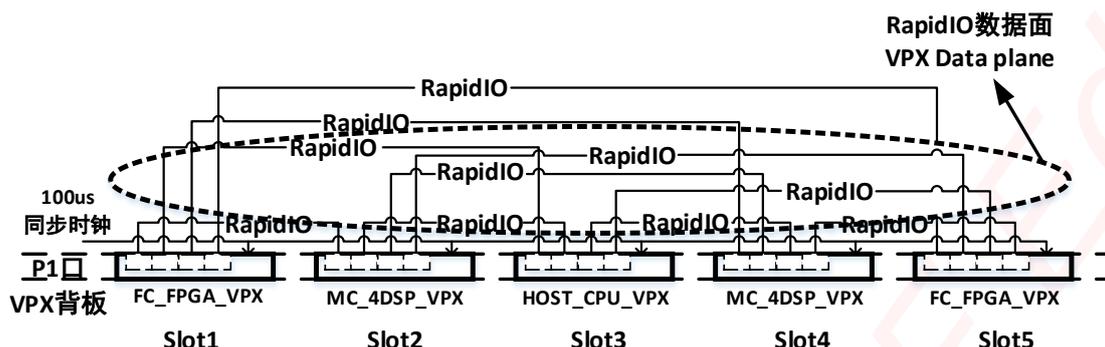


图 2-14 VPX 机箱背板 RapidIO 数据通道互连图

运动控制层主要部署了运动控制卡 MC_4DSP_VPX 和光纤接口卡 FC_FPGA_VPX，MC_4DSP_VPX 卡用来控制光刻机工件台的驱动轴，每个核可控制一组轴，多核同步进行伺服运算，可实现对多个复杂控制对象的逻辑控制、伺服调制及保护等功能。MC_4DSP_VPX 卡无光纤接口，不直接向 IO 接口层收发信号，而是通过光纤接口卡 FC_FPGA_VPX 进行数据输入输出。由于在运动控制过程中会进行大量的插补运算和实时数据存储，而每片 DSP 只有 4M 共享内存，远远不能满足存储需求，因此，每片 DSP 外挂 4GB DDR3 存储空间。

光纤接口卡 FC_FPGA_VPX 在系统中扮演着“九州通衢”的角色，成为系统大量数据输入输出通道，各种数据传输协议、数据封装与解析、收发、存储均由基于 FPGA 硬件实现，从而减少了 MC_4DSP_VPX 卡中 DSP 用于数据传输的软件开销，节约了 DSP 资源。FC_FPGA_VPX 卡通过 FPGA 实现 RocketIO（即 RapidIO）总线接口，与 MC_4DSP_VPX 卡进行实时数据传输。每块 FC_FPGA_VPX 卡包含 12 个光纤接口模块，为系统 IO 接口层的传感器数据采集卡、功率放大器及驱动器等设备提供了足够的通道。光纤接口模块则通过 FPGA 实现光纤数据的收发功能。FC_FPGA_VPX 卡与 IO 接口层设备之间的光纤模块通过高速光纤链路进行数据传输，光纤链路是点对点全双工的串行传输通道，传输速率可达 1.25Gbps。

FC_FPGA_VPX 卡除了与外围设备通讯以外，系统中多处理器的时钟同步机制也依据其与 VPX 背板 P6 口物理介质形成的同步定时网络。实现方式有两种，一种是基于外部同步时钟，如激光干涉测量系统中的总线同步控制卡 SBC 的定时器，并以之作为本地系统同步信号源，并通过时钟差分线与运动控制侧 FC_FPGA_VPX 卡相连；第二种是在没有外接时钟情况下，本地系统中光纤接口卡 FC_FPGA_VPX 输出同步信号，然后将同步采样触发信号通过 VPX 背板 P6 口分发给其他各个处理板卡，主导本系统中同步 50us，100us，200us，同步时间可由系统编程设定。

2.4 本章小结

本章对工件台驱动和感知部分进行了介绍,指出了当前基于 VME 总线的工件台控制系统所存在的问题和系统设计需求,通过对比分析 VPX 体系中的多种总线,明确了多处理器互连总线及关键硬件类型,设计了以标准 VPX 总线构架为基础,以 RapidIO 总线为多处理器互连方式的工件台运动控制系统总体硬件结构方案。

第三章 基于多核共享内存的系统数据并行交互机制

工件台控制系统作为光刻机的核心子系统，包含硅片台和掩模台等多个子系统，在每个伺服周期内，系统会产生大量的数据，各子系统需要对相应的数据进行并行同步计算。因此，需要考虑系统数据并行交互机制及数据交互的实时性。

本章首先以工件台控制系统硬件体系结构为基础，结合运动控制系统设计原理，详细设计系统数据流结构；其次，从数据流结构中提取基于多核共享内存的并行处理模型，对并行处理系统的数据交互需求及其交互方式进行归纳总结及分析；再次，介绍 C6678 的共享内存和 Cache 的数据一致性原理，为多核与外设进行数据正确交互提供基础；最后，针对系统数据交互需求，详细规划 4M 共享内存以及多核本地内存的数据存储空间，并在此基础上，建立基于共享内存的多核并行交互机制。

3.1 系统数据流结构

通常设计运动控制网络来解决运动控制器、电机驱动器及传感器间的数据流问题^[37]，根据第 2.3.2 节的硬件总体架构，采用 RapidIO 互连技术为光刻机工件台系统设计了一种具有低延时性的数据流结构。如图 3-1 所示，实线表示实时通信，虚线代表弱实时传输，其中 1、2、7、8 属于非实时传输，3、4、5、6 属于实时部分，具体的数据流描述如下：

① 上位机监视系统通过以太网接口将控制指令发送给主控卡 HOST_CPU_VPX；

② HOST_CPU_VPX 卡将指令解析并通过 RapidIO 网络发送到 FC_FPGA_VPX 卡；

③ 系统的同步采样信号由激光干涉测量系统中总线控制器 SBC 周期性产生，触发 FC_FPGA_VPX 卡读取电涡流采集卡 SB、光栅采集卡 GIC 和激光计数采集卡中的反馈数据。反馈数据主要是光纤数字信号，由一些高频率、高精度的位置模拟信号转换而来；

④ FC_FPGA_VPX 卡将原始光纤数据整理、打包，通过 RapidIO 网络并行分发给 MC_4DSP_VPX 卡上对应的处理节点 C6678；

⑤ 处理节点 C6678 则根据相应的控制指令对原始光纤数据进行伺服运算，并将数据结果返回至 FC_FPGA_VPX 卡；

⑥ FC_FPGA_VPX 卡将处理后的电机控制量通过光纤发送到电机驱动器，完

成电机控制量的 DA 输出；

⑦HOST_CPU_VPX 卡通过 RapidIO 总线读取 FC_FPGA_VPX 卡中的控制系统状态信息；

⑧HOST_CPU_VPX 卡以一定的周期通过以太网接口将系统状态信息发送至上位机显示。

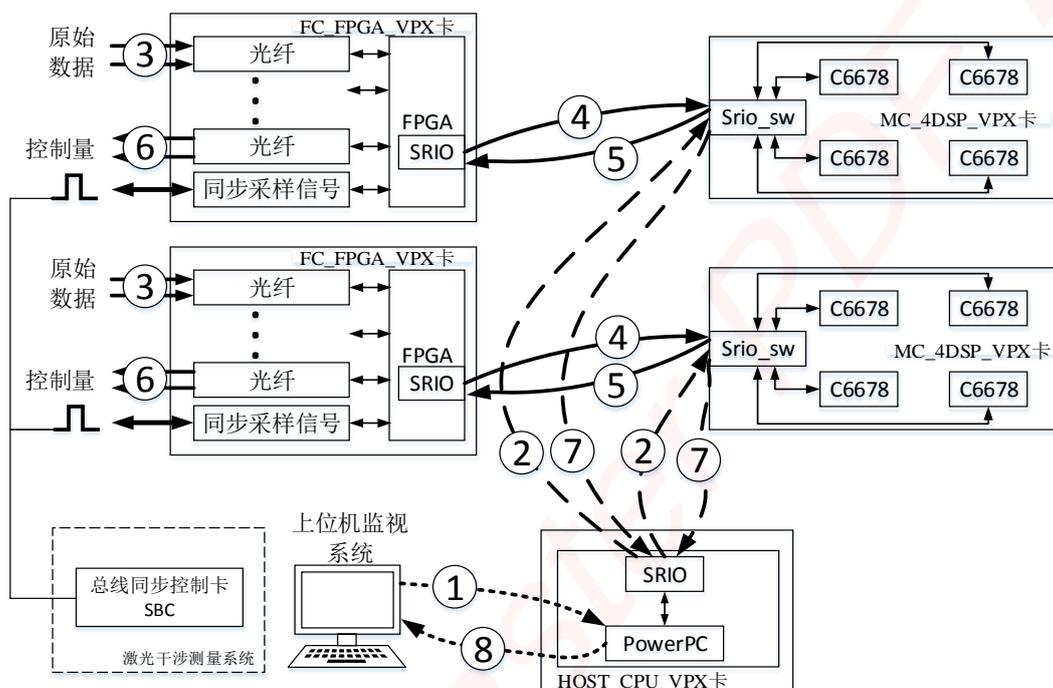


图 3-1 系统数据流结构

3.2 基于并行处理模型的数据交互方式

3.2.1 系统并行处理模型

由于每个伺服周期系统数据流都是确定不变的，包括数据源地址、目的地址、数据包大小、数据包传输的先后顺序等，而且大量的交互数据都汇入相应的 DSP 运动控制卡中进行运算。为保证大量数据的高效计算，降低数据传输延时，实现数据流的最优化控制，设计了多核 DSP 并行处理模型以便进行分析。

如图 3-2 所示，并行处理模型的并行功能是基于串行 RapidIO 交换器实现的。RapidIO 交换器主要实现 DSP、处理器 FPGA、其他交换器或任何基于串行 RapidIO 的器件的路由功能，还可以用于串行 RapidIO 背板交换，同时支持从其任意输入端口到任意输出端口的串行 RapidIO 包交换，包括单播、组播和广播^[38]。

因此，基于 RapidIO 组播方式可实现 FPGA 并行发送数据至多 DSP，而非流水线式依次发送，并基于多核同步访问共享内存，进而实现 DSP 多核同步接收、

发送数据，可大大提升系统整体的并行性和实时性。

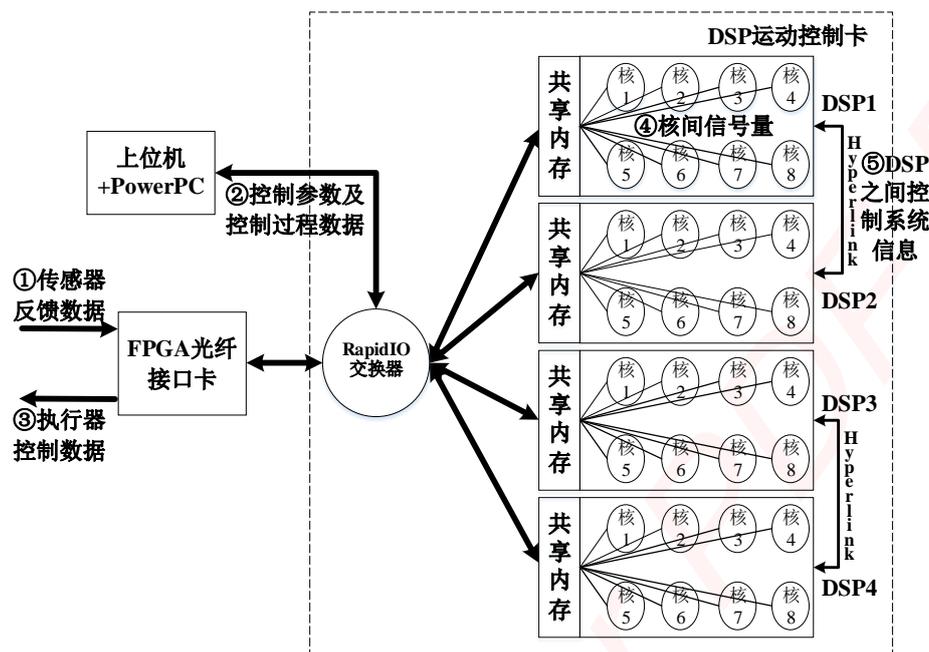


图 3-2 系统并行处理模型

根据多核 DSP 并行处理模型可知，数据交互的主要内容如下：① 测量系统对传感器反馈数据的同步采样，并向其关联的多核 DSP 运动控制器的实时传输；② PowerPC 将板卡配置参数及运动控制参数向多核 DSP 运动控制器的传输，以及 PowerPC 接收 DSP 产生的控制过程数据，如位置误差数据、反馈控制器输出数据等；③ DSP 计算的执行器控制数据向相关执行系统的传输；④ 不同核之间的信号量传递；⑤ 两个 DSP 之间的控制系统信息互传等需求。

3.2.2 数据交互方式分析

系统中以 DSP 为数据交互中心，以上交互内容可归纳为以下几种交互类型：DSP 与 FPGA 之间的数据交互、DSP 与 PowerPC 之间的数据交互、DSP 与 DSP 之间的数据交互、DSP 片内核间的数据交互。DSP 与 FPGA 之间的数据交互和 DSP 与 PowerPC 之间的数据传输采用 RapidIO 技术实现，而 DSP 与 DSP 之间的数据交互由 Hyperlink 总线完成，DSP 片内核间数据交互则采用核间通信技术。

常用的 RapidIO 数据传输方式主要包含直接传输（Direct I/O）、消息传递（Message Passing）和门铃（Doorbell）三种方式。Direct I/O 是最简单使用的传输方式，在该模式下，当 CPU 向外设单元发送或者从外设单元中读取信息时，CPU 需要将包含 DSP 内存地址、目标设备 ID、目标地址、优先级等信息提供给 RapidIO；Message Passing 模式下，由于发送设备对接收设备内部结构和存储器映射地址不

明确，因此，源设备通过 Message Passing 可直接发送数据至目标设备；Doorbell 包格式仅为一个 16 位数据负载，该负载内容可以使用户自定义的中断信息，是一种特殊的轻量级消息传递事务类型，一般用于发送门铃中断通知目标设备^[39]。通过对 RapidIO 几种传输模式的对比，Doorbell 的数据传输量较小，Message Passing 模式的协议复杂，系统软件开销加大，导致其传输速度要小于 Direct I/O 模式，也即直接传输（Direct I/O）更具有传输实时性。由于工件台运动控制系统的伺服周期短、伺服数据量大，为了满足运动控制强实时性、高吞吐率要求，RapidIO 通信协议开销应该尽可能小，尽量缩小通信时间延时，因此，Direct I/O 传输模式最适用于 DSP 与 FPGA 之间、DSP 与 PowerPC 之间的数据交互。

对于 DSP 与 DSP 之间的数据交互，由于传输的数据量非常小，根据板卡各 DSP 总线互连关系，两 DSP 之间采用 Hyperlink 总线。相比于 RapidIO 总线，Hyperlink 是一种更高速度、更低延时、和具有更少的信号线的通信接口。

核间通信包括数据通信和状态通知，数据通信即大量数据的传输，其一般应用于低实时性等系统中；而状态通信所传输的数据量极小，但要求传递延时短、实时性高。工件台控制系统中核间的数据通信主要是状态通信，如急停信号量等。核间通信方法多种多样，可采用共享内存、使用特定的存储空间和存储空间转换等核间数据通信方法^[40]，也可以通过核间中断、IPC、OpenMP^[41]等方法实现。除了以上几种方法，也可以通过基于全局寻址的方式来实现核间通信。其原理如下：

在 TMS320C6678 多核平台中，LL2 存储器既可分配成 Cache，也可设置为 SRAM，这里全配置成 512KB 的 SRAM。C6678 的所有核都有相同“别名地址”，其起始地址一样，可以通过“别名地址”直接访问自己的内存空间，而不能相互访问。但在 LL2 上所有核也可通过全局寻址互相读写数据^[42]。例如，地址 0x10800000 是 C6678 核 0 的 LL2 存储器的全局基地址，核 0 能够通过访问地址 0x10800000 或 0x00800000 来访问该地址，而其他核只能通过访问 0x10800000 来读写该地址空间，0x00800000 可以被其他核用来访问各自的 LL2 存储器。因此，若数据变量 A 在核 0 的 LL2 存储器中，其他核则只需读全局地址 $Addr_n = \&A + 0x10000000 + 0x01000000 * n$ ($n=1, 2, \dots, 7$) 来获取该地址中的数据，从而实现核间数据交互。

3.3 C6678 的数据 Cache 一致性分析和维护

多核在并行进行数据运算时，不可避免会出现多核、多外设共同访问同一个共享内存的情况。由于在核（Core）和内存之间增加了缓存（Cache），任何外设对内存的修改并不能保证 Cache 中的内容也得到同样的更新，同样 Core 对 Cache 中内

容的修改也不能保证内存中的数据得到更新。在 TMS320C6678 中，各核的 L1D Cache 和 LL2 存储空间的数据一致性由硬件维护，硬件保证了 LL2 中更新的任何数据都能加载到 L1D Cache 中，而共享内存与 L1D Cache 之间的数据一致性由人工手动维护^[43]。

3.3.1 C6678 的数据 Cache 一致性问题

Core 在与共享内存进行数据交互时，如果没有对 Cache 进行适当的操作，将可能产生以下数据不一致等问题，导致向外设传输数据时或 Core 取数进行计算时出现错误。

1) Core 读一致性问题

当 Core 从共享内存读数据时，由于 Core 第一次对共享内存进行读操作的时候，数据不在 L1D Cache 中，硬件会直接将共享内存中的数据读到 L1D Cache。而此后，RapidIO 总线每个伺服周期都会更新共享内存中的数据，Core 再次读共享内存时，发现数据已经存在 L1D Cache 中，但依旧是上一个伺服周期的数据，导致 Core 不能得到最新的数据，就会出现 Core 读一致性问题，如图 3-3 所示。

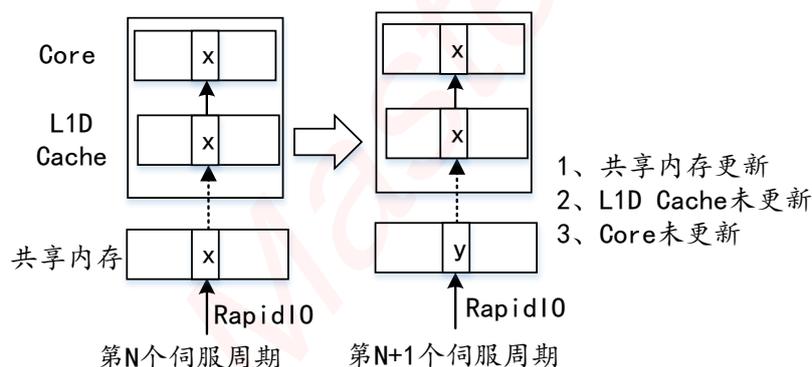


图 3-3 Core 读一致性问题

2) Core 写一致性问题

当 Core 向共享内存写数据时，Core 处理完后会将数据更新到 L1D Cache 中，当 RapidIO 总线从共享内存读数据的时候，由于 L1D Cache 中新的数据未被及时更新到共享内存中，则在下一个伺服周期中，共享内存依旧是原来的数据，RapidIO 读取的并不是更新后的数据，就会出现 Core 写一致性问题，如图 3-4 所示。

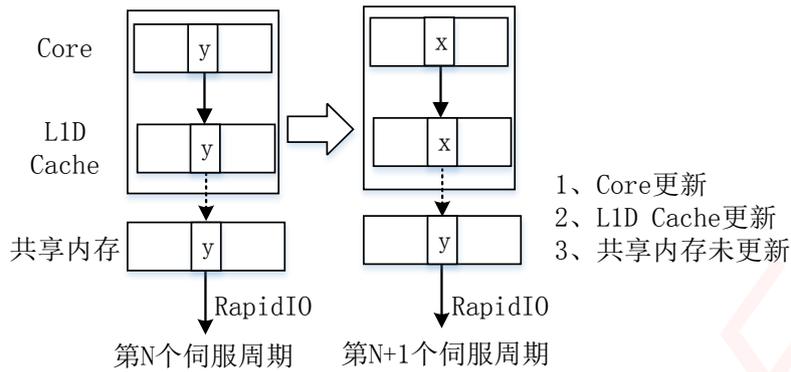


图 3-4 Core 写一致性问题

3.3.2 C6678 的数据 Cache 一致性维护操作

为了保证共享内存和 Core 以及 L1D Cache 之间进行正确的数据传输，须对 L1D Cache 进行正确的操作。根据 TI 或第三方提供的 API 函数，针对以上两种情况分别作出如下决策：

1) Core 读共享内存一致性维护

图 3-5 描述了 Core 读取共享内存中的数据进行处理的情况。当下一个伺服周期时，RapidIO 总线刷新共享内存对应的内容之后，而在 Core 读取之前，可先对此部分数据进行失效 (invalidate) 操作，保证了共享内存中新的数据会被重新加载到 L1D Cache 中，最终 Core 获取实时更新的数据。

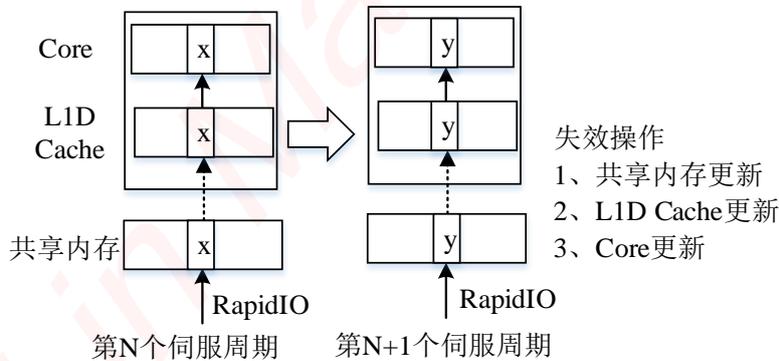


图 3-5 Core 读共享内存一致性维护

2) Core 写共享内存一致性维护

图 3-6 描述了 Core 写数据至共享内存进行 RapidIO 传输的情况。Core 在每个伺服周期计算完成后，更新的数据都会保存在 L1D Cache 中。为了保证 RapidIO 能够正确从共享内存中读取数据，需要在 RapidIO 读取数据之前，通过写回 (writeback) 操作将 L1D Cache 中的数据同步更新到共享内存中。

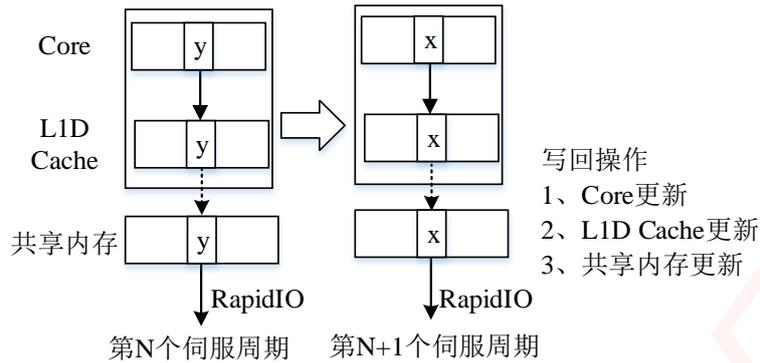


图 3-6 Core 写共享内存一致性维护

3.4 基于共享内存的数据交互

3.4.1 基于共享内存的数据存储空间规划

由于在多核 DSP 并行处理架构中，系统存在大量而频繁的数据交换，采用共享内存建立全局物理内存来共享数据，为每种交互数据类型规划了可独立编址的统一数据存储空间，同时也为每个并行核设计了具备透明存储接口的分布式本地内存，由系统周期性数据流刷新共享内存中的数据。

四种交互数据类型中，大量数据传输集中在 DSP 与 FPGA 之间、DSP 与 PowerPC 之间，而 DSP 与 DSP 之间以及 DSP 片内核间的通信数据量较小，因此，在共享内存中规划了两段独立地址作为这四种类型的数据缓存区间。如下图 3-7 所示，C6678 DSP 提供了 4M Byte 大小的共享内存，其起始地址为 0x0c000000。第一部分区间 MSMCSRAM0 起始地址设置为 0xc010000，总长度为 0x2f0000，该区段用于 Hyperlink 数据收发端以及多核共享变量的数据缓存空间，可实现 DSP 与 DSP 之间、DSP 片内核间的数据缓存；将第二部分区间 MSMCSRAM1 设为 RapidIO 数据发送方和接收方的固定起始位置 SrioFixAddr，地址为 0x0c300000，总长度为 0x100000，能容纳 30801392 个字节。

在工件台运动控制系统中，由于共享内存的数据传输存在着复杂数据源和目的地址，即使是 RapidIO 数据传输，其数据源也有 PowerPC 端和 FPGA 端，而 PowerPC 端与 DSP 端交互的主要是命令包和控制过程数据包，FPGA 端和 DSP 端传输的数据包括传感器数据包和执行器控制数据包，这些数据包涵盖多个控制台的数据。如果对每个控制台的相关数据在共享内存中划分一片存储区间，每个控制台的命令包、传感器数据包等在存储空间上是分离的，导致无法通过 RapidIO 总线将数据包统一发送或接收。考虑到传输的数据在共享内存中分布统一性和连续性，即相同物理量的数据处于同一片地址段，在 MSMCSRAM0 空间中分别设置了 DSP

与 DSP 之间交互数据缓冲区 DSPtoDSP_Buf、核间交互数据缓冲区 COREtoCORE_Buf; 在 MSMCSRAM1 空间内定义了四大缓冲区, 分别为控制参数缓冲区 CmdBuf, 控制过程数据缓冲区 ConprogBuf, 传感器反馈数据缓冲区 SensorBuf, 执行器控制数据缓冲区 ActconBuf。各缓冲区均以数组变量形式存在, 以便后期可以通过修改数组大小来扩增缓冲区容量, 从而可以为增添的控制台提供数据存储空间。

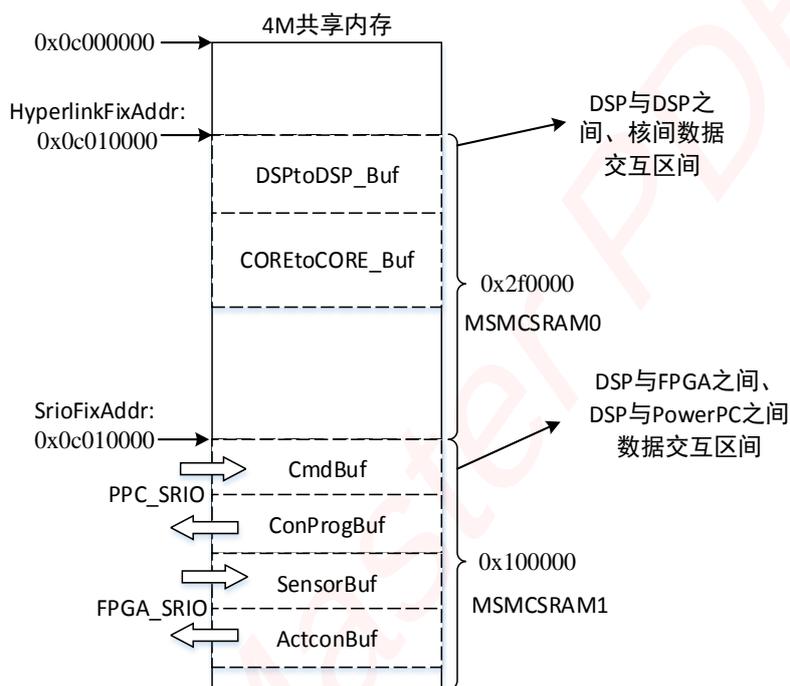


图 3-7 基于共享内存的数据存储空间规划

3.4.2 共享内存与本地内存 LL2 的数据地址对应关系

在共享内存中, 同一类型的缓冲区间包含多个控制台的数据, 不同控制台运动计算由相应的 Core 实现。如图 3-8 所示, SensorBuf 缓冲区包含 Core0、Core1、Core2 所需的传感器反馈数据, 当三核同步访问共享内存同一片缓冲区 SensorBuf 时, 可采用交叉复制策略, 即将该缓冲区的不同内容同步交叉复制到对应核的本地内存空间 LL2 SRAM 的相应地址单元中。为了便于 RapidIO 数据存储及交互, 在 Core LL2 SRAM 中, 规划与共享内存一样顺序的缓冲区, 也就是 LL2 中也按照 DSPtoDSP_Buf、COREtoCORE_Buf、CmdBuf、ConprogBuf、SensorBuf、ActconBuf 顺序进行规划。同理, 各缓冲区的数据也可以采用同样原理复制到各核中, 与此同时, 在软件维护 Cache 一致性的前提下, 也可将各核的数据交叉复制到共享内存相应的缓冲区。

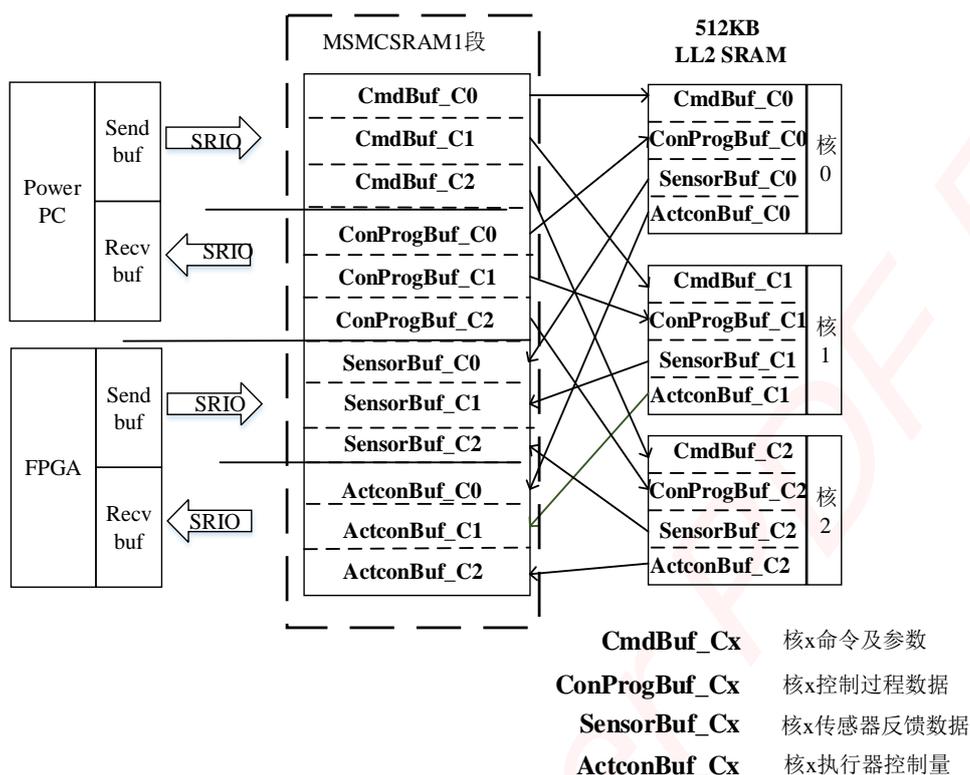


图 3-8 共享内存与多核本地内存的数据地址对应关系图

3.4.3 数据具体交互过程

运动控制系统具有明确的插补周期，过于频繁刷新共享数据虽然能保证处理器获取的传感器反馈数据更实时、更有效，但是受限于总线带宽以及共享内存访问速率，因此，系统数据交互频率与伺服频率保持一致即可，也即每个并行计算周期中完成一次共享内存地址单元的数据更新。

为了更好的展示具体交互过程，将多核并行访问共享内存框图简化成如图 3-9 所示的数据具体交互过程模型，共享内存中的所有不同收发缓存被定义为接收缓存 RX_BUF 和发送缓存 TX_BUF，通过定时及 RapidIO 总线数据传输来驱动数据流，并刷新共享内存收发缓冲区的数据内容，通过对 L1D Cache 失效操作，将共享内存中的内容交叉复制至各核的 LL2 SRAM 中，周期性地刷新了 LL2 SRAM 中的数据。由于 L1D Cache 与 LL2 SRAM 中的数据一致性由硬件保证，Core 能准确的获取 LL2 SRAM 中的数据进行伺服运算，并将控制过程数据和计算出的执行器控制数据临时保存至 LL2 SRAM 相应缓冲区中，再次通过数据 Cache 一致性维护中的写回操作，将各核需交叉复制的数据写回到共享内存中。RapidIO 总线发送相关的内容至 FC_FPGA_VPX 卡和 HOST_CPU_VPX 卡，完成执行器控制数据和控制过程数据的传输。

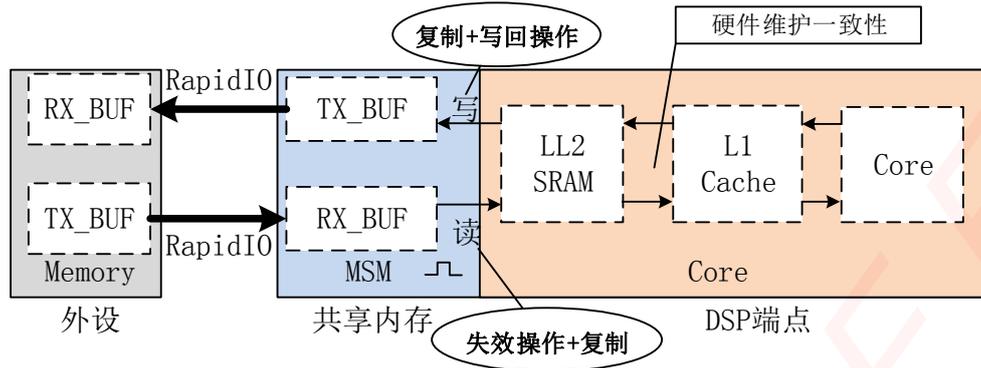


图 3-9 数据具体交互过程模型

3.5 本章小结

在基于 RapidIO 总线的工件台控制系统硬件结构的基础上，设计了系统的数据流结构，并建立了多核 DSP 并行处理模型以及分析了数据需求和交互方式。为了保证多核与外设的数据交互一致性，引出了 C6678 数据 Cache 和共享内存的数据一致性，包括 C6678 的数据 Cache 一致性问题和维护操作。根据工件台系统数据需求，规划了共享内存的数据存储空间及与多核本地内存 LL2，在上述基础上，详细展示了基于共享内存的数据并行交互具体过程。

第四章 控制系统精密同步测控策略研究及技术实现

建立合理的同步测控策略是进行工件台运动控制系统详细设计的重要基础，一般针对多轴运动控制系统的同步性问题，常规的多个单核 DSP 运动控制器不能进一步提升系统，从而使用多个多核 DSP，而多 DSP 多核同步技术成为系统同步的重要技术手段。

本章分为以下五个部分，第一部分回顾多核 DSP 中断子系统结构和中断映射原理；第二部分根据系统同步要求，分别设计基于 Direct I/O + Doorbell 软中断模式和基于 Direct I/O + GPIO 硬中断模式的同步测控方案，并对两种方案的同步性和实时性进行比较分析；第三、四部分则针对选定的测控方案二，结合部分硬件原理图及 C6678 中断映射原理，详细介绍基于 GPIO 中断的多 DSP 同步技术实现以及 C6678 多核同步的具体实施方法；第五部分根据最小测控模型，对系统进行最优时序设计及分析。

4.1 TMS320C6678 的中断子系统结构和中断映射原理

4.1.1 TMS320C6678 的中断子系统结构

TMS320C6678 的中断子系统^[44]由芯片级中断控制器 (INTC) 和 DSP 核 (以下简称 CorePac) 级中断处理控制器 (CorePac Interrupt Control) 两部分构成。如图 4-1, 整个 Soc 片内中断处理子系统框图中共有四个 INTC, 分别为 INTC0、INTC1、INTC2、INTC3。其中, Core0 到 Core3 的控制主要由 INTC0 负责, C4 到 C7 的控制由 INTC1 完成, EDMA3 的 TPCC1 和 TPCC2 由 INTC2 负责控制, EMDA3 的 TPCC0 以及 Hyperlink 的中断控制则由 INTC3 实现。在 C6678 中, 8 个 CorePac 内部都有自己的核级中断控制器, 通过中断映射原理等, 将外部事件转换成 CorePac 内部的中断信号。

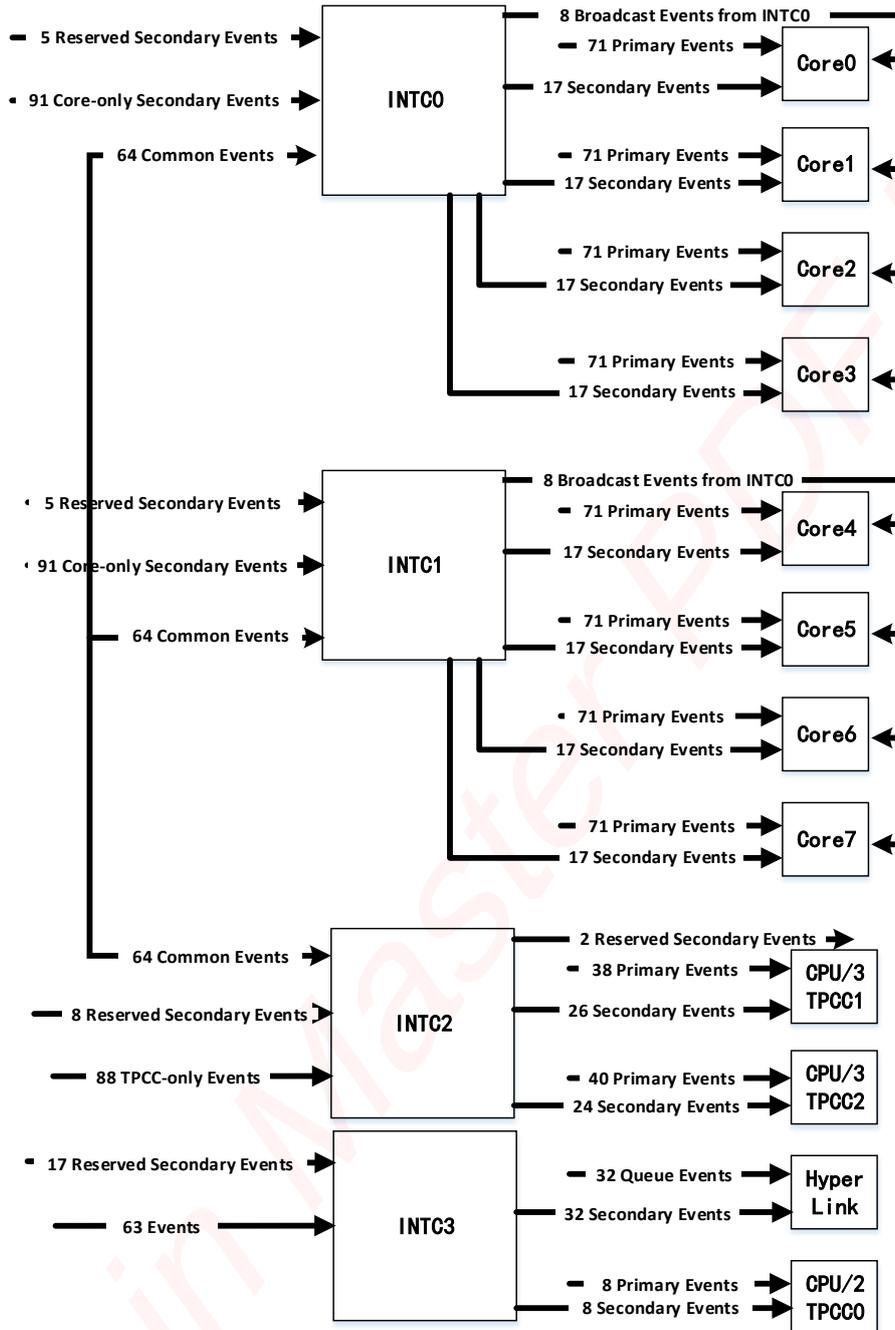


图 4-1 C6678 片内中断处理子系统框图

4.1.2 片级中断控制器（INTC）

INTC 的主要功能为将系统事件（system interrupt）映射为 CorePac 内部中断控制器可以处理的主机事件（host interrupt）。由于 CorePac 中断控制器最多只能处理 128 个输入事件，那么 INTC 的一个重要的功能就是事件映射功能，即将多个系统事件映射为多个或者单个主机事件。

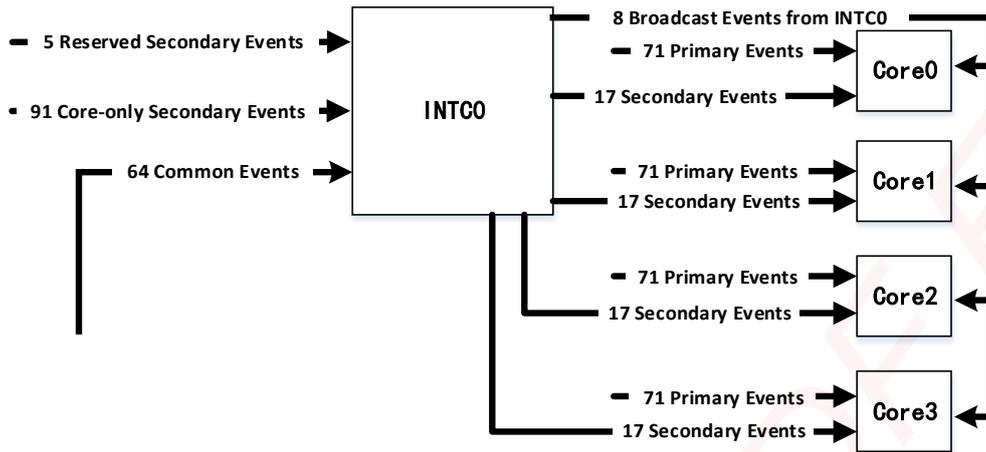


图 4-2 INTC0 的事件输入输出图

如图 4-3 所示，INTC 整个事件映射分为两部分，系统事件映射和信道映射，系统事件映射就是将系统事件映射到 channel 中，然后通过信道映射功能，将每个 channel 又映射到主机事件中，从而能被 CorePac 直接处理。因此，通过 INTC0 的映射配置，就可以将左边 160 个系统事件，有选择性转换为右边的 17+8 个主机事件。

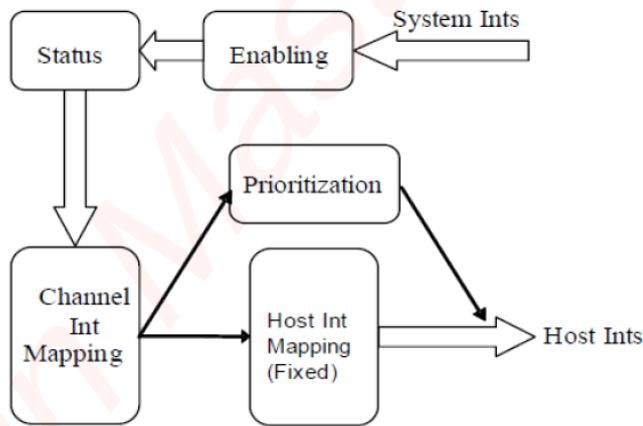


图 4-3 INTC 的事件映射和信道映射原理图

4.1.3 核级中断控制器（CorePac Interrupt Control）

如图 4-4 所示，CorePac 的中断控制器有 3 个输入来源，RESET，NMEVT 和 EVT。分别是复位信号、不可屏蔽中断信号和普通事件信号。其中普通事件总共有 128 个，这 128 个事件可分为 3 个部分，第一部分是组合事件，即 EVT0- EVT3，这部每个事件可由剩余的 124 个事件组合而成，相当与一个逻辑的或门；第二部分 INTC 信道映射过来的主机中断；剩下的为第三部分。在 CPU 内，可供处理的

信号为 INT[15:4]，即只有 12 个可供 CPU 处理的中断信号。因此，CorePac 中断控制器的主要工作就是通过中断映射将 128 个 EVT 转换为 12 个 INT。

中断映射指的是从 128 个中断事件中选择 12 个映射到 CPU 侧的 INT 中。中断映射需要用到 CorePac 内部的中断复用寄存器 (Interrupt Mux Register)。每个 CorePac 内部有 3 个 Mux 寄存器，每个 Mux 可以配置 4 个 INT，因此总共配置 12 个 INT。

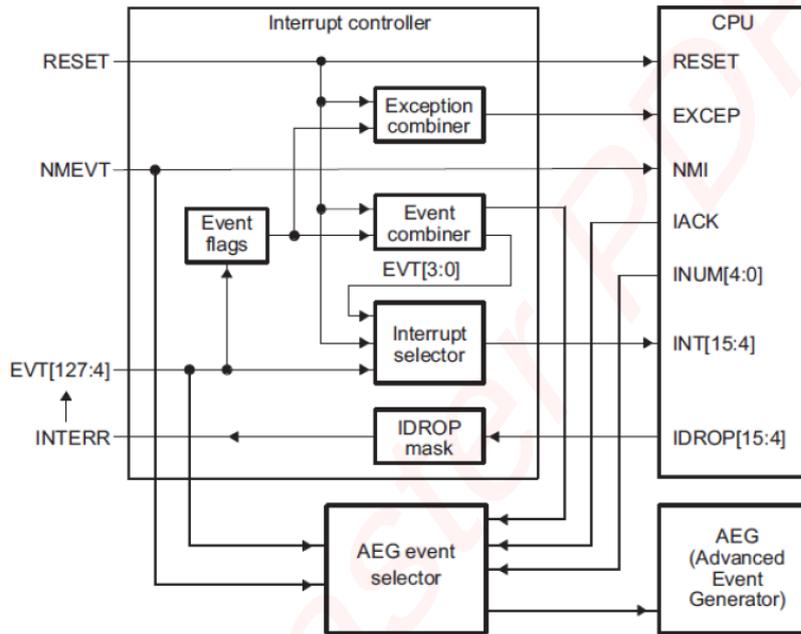


图 4-4 核级中断控制器内部框图

4.2 工件台控制系统同步测控方案设计及其分析

在硅片曝光时，光刻机工件台要求在高加速度、高速度下，达到更高精度的同步运动精度，除了必须保证硅片台系统、掩模台系统中各子系统的数据并行计算，还需要确保各子系统的时序一致性，若各子系统在时序上存在微小的同步时间误差，将会引起较大的同步位置误差，进而影响工件台整体的运动控制精度。因此，需要建立一套具有强实时性的精密同步测控方案。

系统有两块光纤数字信号采集卡，系统的同步性和实时性要求数字采集卡必须在同一个时钟沿开始启动采集，即共享同一个时钟源，并在相同的伺服周期内完成光纤信号输出。而且 FC_FPGA_VPX 卡和 MC_4DSP_VPX 卡的时钟系统相对独立，两卡之间如何协调工作是整个系统的关键技术之一，其具体表现在中断控制线和 RapidIO 数据线之间如何协调。本节选取 Direct I/O 数据传输方式，主要就基于 Direct I/O + Doorbell 软中断模式、基于 Direct I/O + GPIO 硬中断模式的同步测控

方案进行研究及分析。

4.2.1 基于 Direct I/O + Doorbell 软中断模式的同步测控方案

在 RapidIO 多种数据传输包中，门铃（Doorbell）包具有软中断特点。为了实现接收端 DSP 快速接收数据，采用数据线和中断控制线统一模式，即数据发送和中断发送使用同一个物理传输通道。如图 4-5 所示，时钟源由外部定时器或 FC_FPGA_VPX 卡的内部定时器提供，通过背板定时通道口 P6 实现多 FC_FPGA_VPX 卡的同步定时。同步采样信号，也即时钟源信号，周期性触发 FC_FPGA_VPX 卡采集光纤数据后，通过 RapidIO 交换器（SW）组播数据到 MC_4DSP_VPX 卡的多个 DSP 后，但由于 Doorbell 无组播功能，需立即向目标设备依次发送 Doorbell 中断，告知相应 DSP，在其共享内存中的 RapidIO 缓冲区已刷新完毕。在 Doorbell 中断服务程序中，使用 DSP 多核同步技术，实现 DSP 各核的伺服中断程序同步。

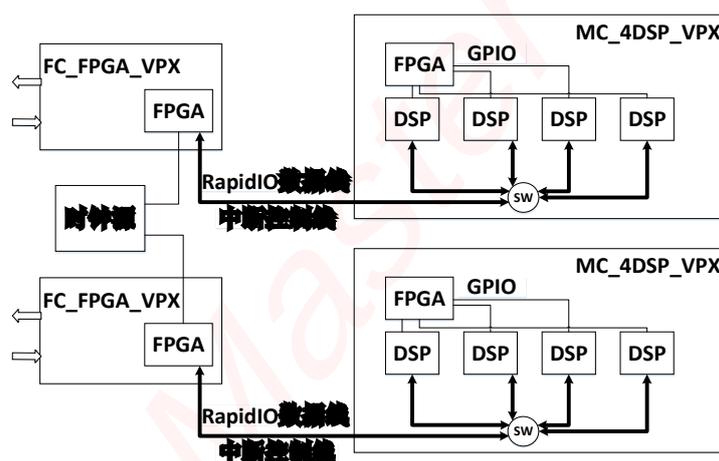


图 4-5 基于 Direct I/O + Doorbell 软中断模式的同步测控框图

4.2.2 基于 Direct I/O + GPIO 硬中断模式的同步测控方案

除了采用方案一中 Direct I/O 传输方式配合软中断模式，也可以将数据线和中断控制线分离，使用 Direct I/O 传输方式与 GPIO 硬中断技术。如图 4-6 所示，FC_FPGA_VPX 卡的同步采样信号由时钟源生成，而且 MC_4DSP_VPX 卡通过内部 FPGA 接收时钟源的定时中断信号，之后通过 GPIO 同步中断技术实现板上多 DSP 同步。由于 FC_FPGA_VPX 卡的同步采集信号和 MC_4DSP_VPX 卡的中断信号同步，因此，当 DSP 接收到周期性的中断触发信号时，RapidIO 总线还未发送当前伺服周期的反馈数据至 DSP，即在 DSP 共享内存中，RapidIO 缓冲区的位置数据并未得到刷新，而伺服中断任务也在同步读取 RapidIO 缓冲区中的位置数据。因

此，读取的数据依旧还是上一次伺服采样周期的数据，导致伺服计算及输出总滞后于位置采样一个周期，造成较大的位置误差。

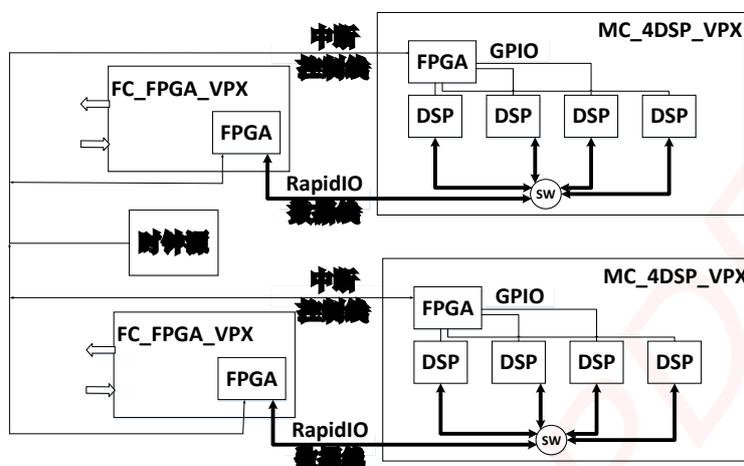


图 4-6 基于同步时钟源的同步测控框图

针对上述存在的缺陷，采用异步时钟技术，即触发 DSP 同步的中断源信号以时钟源为基准，频率与时钟源保持一致，但须通过 FC_FPGA_VPX 卡中的 FPGA 延迟中断源信号，延迟时间至少大于 Direct I/O 单向传输延时(具体延时见第 5.2.2.1 节)。方案原理如图 4-7 所示，FC_FPGA_VPX 卡同步采样完成后，通过 Direct I/O 组播数据，MC_4DSP_VPX 卡等待定时生成的中断源信号触发，通过 GPIO 硬中断同步技术实现多 DSP 同步实时中断。

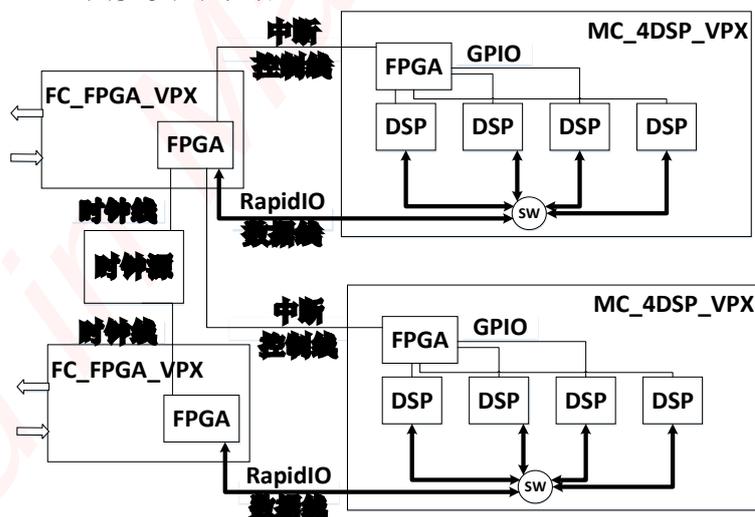


图 4-7 基于异步时钟源的同步测控框图

4.2.3 两种同步测控方案的比较分析

经过实验验证，基于 Direct I/O + Doorbell 模式的同步测控方案中 Doorbell 中断

响应延时约4.46us左右（具体实验数据结果见第5.2.2.2节），所消耗的时间在传输总时间中所占比重较大，不容忽视；而且Doorbell不能组播中断，严格意义上的多DSP同步中断不存在。基于Direct I/O + GPIO模式的同步测控方案虽增加了系统软件复杂度，不仅给FC_FPGA_VPX卡上的FPGA增加了同步中断源信号生成模块，而且DSP也需加载对应的GPIO口中断模块，但GPIO硬中断响应速度极快，考虑到系统强实时性，基于Direct I/O + GPIO模式的同步测控方案是最优选择。

4.3 基于 GPIO 硬中断的多 DSP 同步技术实现

在 MC_4DSP_VPX 卡上，4 个 DSP 与 FPGA 的 GPIO 引脚连接原理图如图 4-8 所示，通过 FPGA 的并行处理特点实现引脚 AM31、AK32、G28 和 A28 上的信号同步输出，由各 DSP 的 Core 0 的核级中断控制器来接收相应的定时中断信号。

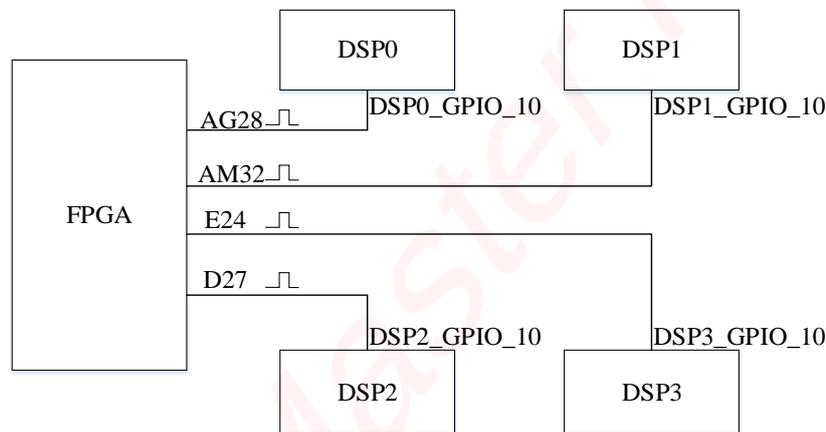


图 4-8 FPGA 与 DSP 引脚连接原理图

根据 4.1 节中的 C6678 片内中断处理子系统框图，GPIO 中断源属于 Core 0 的一级事件输入，由核级中断控制器处理该中断事件。每个 DSP 的 GPIO_10 作为中断接收引脚，如图 4-9 所示，该引脚的中断事件号为 84，为 GPIO_10 定义一个实际的中断向量号 5，然后将该中断向量号和一个自定义的中断服务函数入口地址相关联。

Input Event Number	Interrupt Event	Description
83	GPINT9	Local GPIO interrupt
84	GPINT10	Local GPIO interrupt

图 4-9 中断事件号与中断事件关系图

使用端口 GPIO_10 进行中断操作的具体流程如下：

- 1) 初始化 GPIO 模块 C6678_Gpio_Init();
- 2) 将 GPIO_10 设置为输入 C6678_Gpio_SetDirection(10,GPIO_IN);
- 3) 使能对 CPU 的 GPIO 中断 C6678_Gpio_EnableGlobalInterrupt();
- 4) 设置为检测 GPIO_10 上升沿触发 C6678_Gpio_SetRisingEdgeInterrupt(0);
- 5) 初始化核中断控制器 C6678_CoreInt_Init();
- 6) 挂中断服务程序 C6678_CoreInt_Set(84,5,test_isr_handler_Gpio,NULL)。

4.4 多核同步技术实现及分析

4.4.1 基于广播事件的多核同步触发实现

C6678 提供了大量的系统中断事件，如图 4-10 所示，中断事件分为三种：一级中断事件、二级中断事件和广播事件。一级中断事件直接给各核级中断控制器产生中断，而不经片级中断控制器。二级中断事件经由片级中断控制器映射给核中断事件，再通过核中断控制器触发核中断。广播事件是系统中断事件经片级中断控制器映射到广播通道后广播给 8 个或 4 个核，其中，片级中断控制器 0 广播给 0 号，1 号，2 号，3 号共 4 个核，片级中断控制器 1 广播给 4 号，5 号，6 号，7 号核共四个核，通过分别配置片级中断控制器 0 号和 1 号以及 8 个核的核级中断控制器，即可将某系统中断事件广播给 8 个核，将中断事件广播给 8 个核。

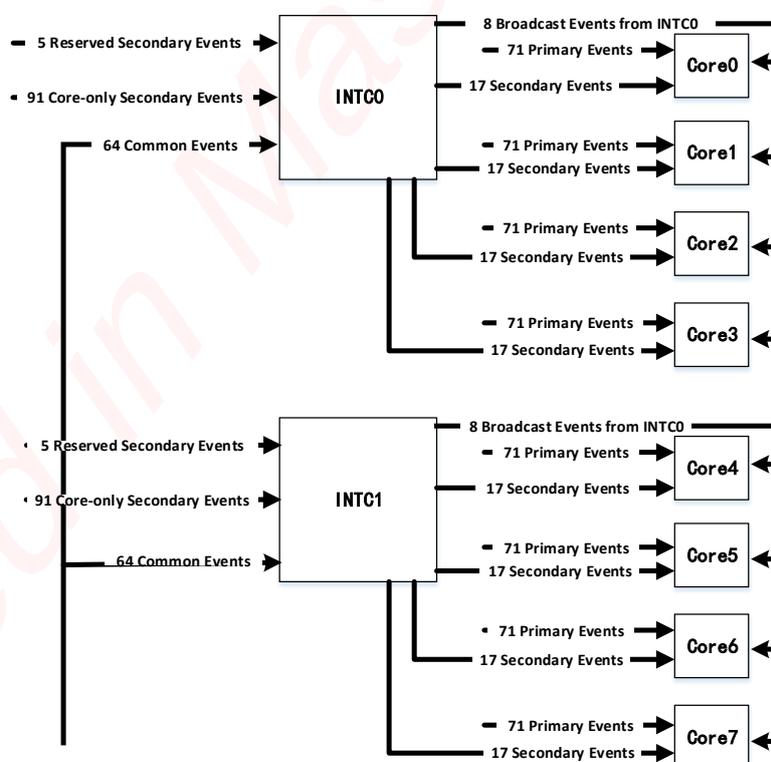


图 4-10 片级中断控制器 0、1 与核级中断控制器事件输入输出框图

以上三种中断事件中,唯独只能通过广播事件可驱动多核同步并行运行,如图 4-11,核 0 为主核,管理着整个芯片的外设,如 GPIO 口中断,因此通过核 0 调用 C6678_CoreInt_Init (Uint8 ChipIntNum),初始化核级中断控制器,并通过 C6678_CoreInt_Set()挂载 GPIO 中断服务程序,接收 FPGA 定时触发 DSP 的 GPIO 中断事件。在 GPIO 中断服务子程序中,手动触发片级中断控制器 0、1 上的某个二级中断事件,如 58 号中断事件 I2C,然后调用 C6678_ChipInt_Set (Uint8 ChipIntNum, Uint16 EventId, Uint16 ChanId)函数将 58 号中断事件映射到广播事件 (Broadcast) 通道 0,由文献[42]中核级中断事件表可知,102 号核级中断事件是通道 0 的输出事件,通道 0 是广播通道,片级事件映射到广播通道后将事件广播给 4 个核,如果片级中断控制器 0、1 都使能,则可以映射给 8 个核,进而实现 8 核定时同步。

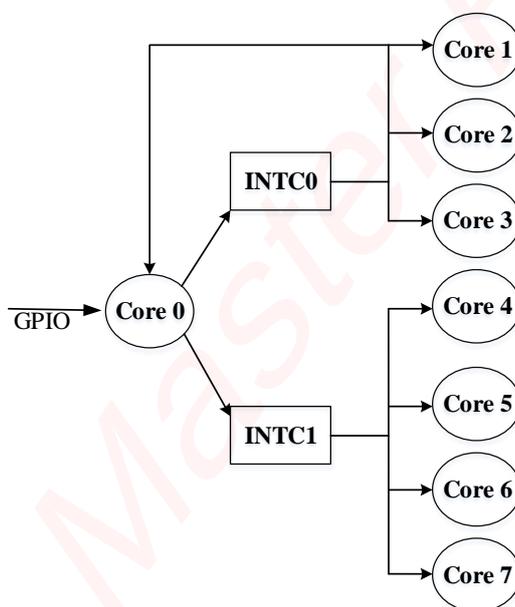


图 4-11 基于广播事件的多核同步原理图

该方法的具体操作流程如图 4-12 所示:

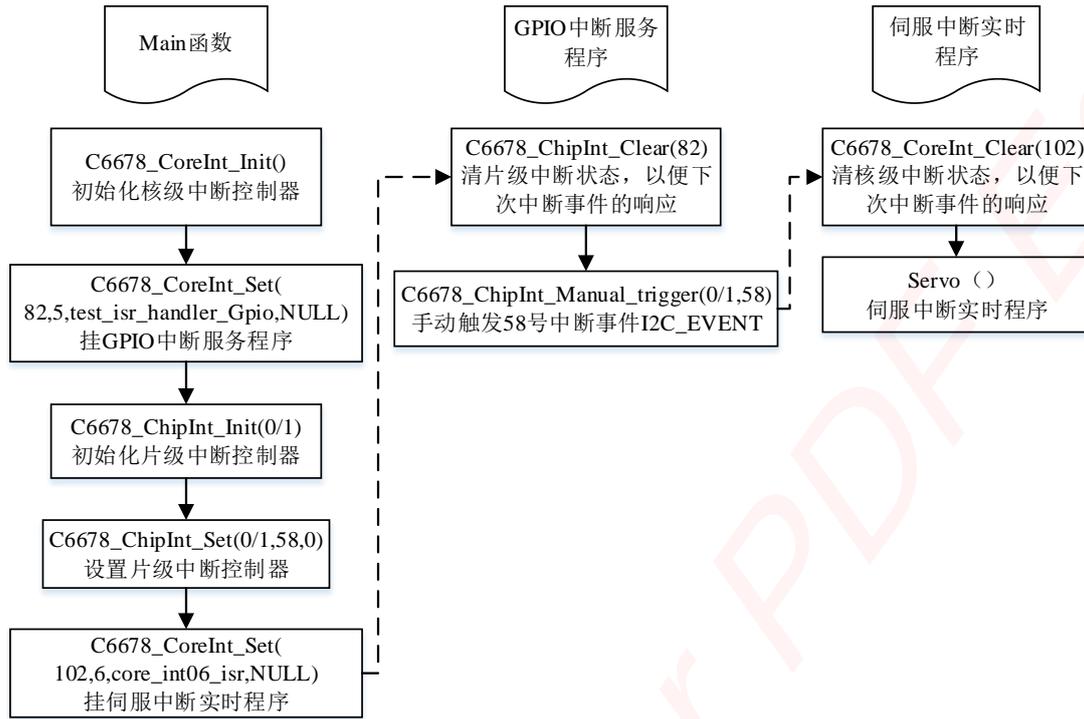


图 4-12 基于广播事件的多核同步流程图

4.4.2 基于核间中断 IPC 的多核同步触发实现

为了实现多核同步，也可使用核间中断 IPC 触发其它核中断，在 C6678 中，由于核 0 可以通过事件控制器产生事件 (Event)，触发核间中断与其他核心进行通信。当中断发生时，其他核就会由事先配置好的中断向量表跳入中断服务程序 (ISR)，完成核间通信，实现同步，如图 4-13 所示。

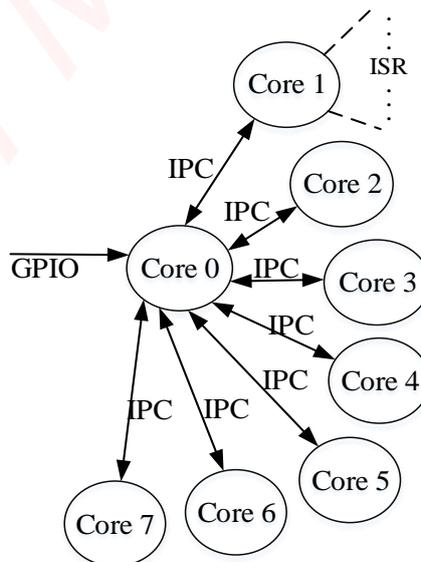


图 4-13 基于核间 IPC 中断的多核同步原理图

现以核 0 为主核，核 1-7 为从核，基于核间中断 IPC 的多核同步流程如下：

- 1) 核 0 通过配置 GPIO 口中断参数，实现对 FPGA 触发的多核同步周期信号的接收；
- 2) 核 0 接收到多核同步周期信号后，多次调用 C6678_IPC_Intc_Send()函数依次触发其余从核的 IPC 中断；
- 3) 各个从核通过函数 C6678_IPC_Intc_HookUp()挂载中断服务程序，接收到相应的 IPC 中断后立即进入伺服中断子程序，进行相应的伺服运算任务。

4.4.3 两种多核同步方法的比较分析

以上两种方式均基于硬中断完成多核的同步，由于基于核间中断 IPC 的 DSP 多核同步技术在触发从核 IPC 中断时，需要主核按照流水线的方式依次向下一个从核发送核间中断，不能严格同步多核。而且实验表明，核间中断 IPC 的响应延时为 6.4us 左右（具体实验数据见 5.2.3.2 节），而基于广播事件的多核同步的中断响应只需 576ns \ll 6.4us。可见，基于广播事件的多核同步技术在中断响应实时性和多核同步性等方面更具优越性。

4.5 系统同步时序最优化设计

4.5.1 最小测控模型建立

根据基于 Direct I/O + GPIO 硬中断模式的同步测控方案，可以从系统数据流结构中提取出一个最小化的测控模型，如图 4-14 所示，该模型由传感器单元、光纤接口单元、控制单元、执行器单元四部分组成。传感器单元和执行器单元均与光纤接口单元均通过一根双向传输的光纤线连接，控制单元与光纤接口单元之间通过 RapidIO 总线建立一个双向通信的 LVDS 链路。

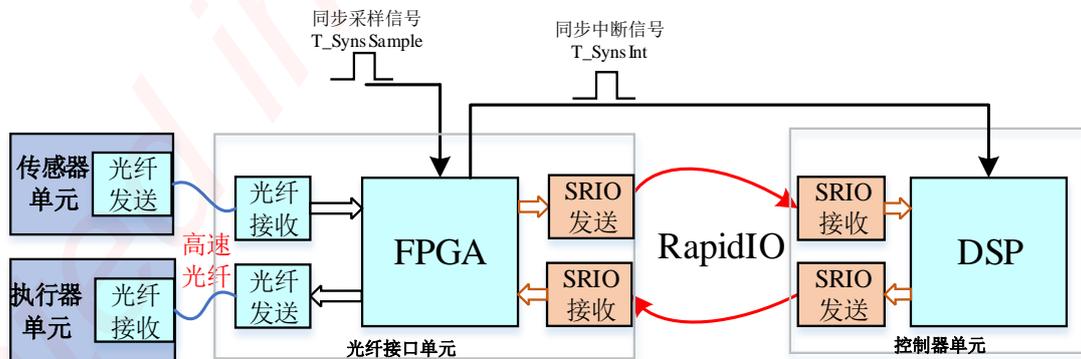


图 4-14 最小测控模型

4.5.2 系统同步时序最优化详细设计

在高速、高精度的运动控制中，不仅需要非常短的伺服周期，还需要精确的时序。假设系统中最高的运动速度为 1400 mm/s，时序上 1us 引起的采样位置误差为 $\delta = 1400 \text{ mm/s} * 1\text{us} = 1.4\mu\text{m}$ ，因此，为了达到 nm 级运动精度，必须保证 ns 级别的时序精度，否则将引入较大的位置误差，无法对控制和运动性能进行评价。

为了获得相对严格的时序，通过系统时钟（外部定时器或 FC_FPGA_VPX 卡内定时器）产生一个标准且唯一的 10KHz 的时钟源，作为系统同步采样信号和同步中断源信号的生成基准，同步采样信号用于 FC_FPGA_VPX 卡的光纤数据同步采集，同步中断源信号负责定时触发多个 DSP 中断。

如图 4-15 所示，在标准的时钟沿到来之后，即在 t_0 时刻，同步采样信号立即触发光纤接口单元中的 FPGA 光纤控制器发送读光纤数据命令（READ_TRIG），捕获传感器单元最近锁存的位置光纤数据。为了保证当前采样周期内的所有光纤数据能够被读取，添加一个光纤数据更新完成标志位 Flag_Sfp_Over，用来判断当前采样周期内光纤接口单元中的位置信息是否为更新完成。若 Flag_Sfp_Over=0，则继续等待数据采集完成；若 Flag_Sfp_Over=1，表示位置光纤数据已完全更新，随后将光纤数据填充至 RapidIO 缓冲区，并调用 RapidIO 发送模块传输数据至相应的 DSP，之后清标志位 Flag_Sfp_Over=0，此时刻记为 t_1 。

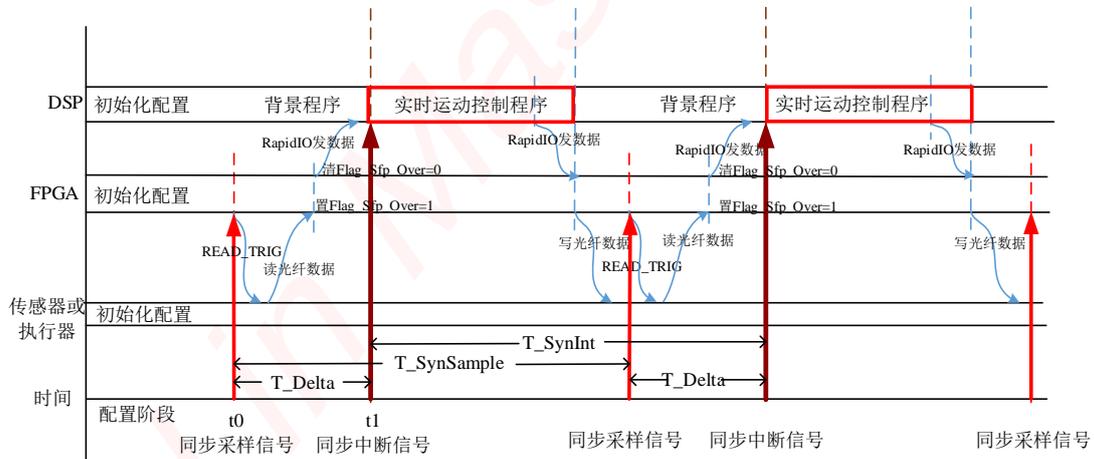


图 4-15 系统同步时序图

从以上过程可知，光纤接口单元在同步采样信号触发之后，将采集的光纤位置数据传输给 DSP 所用的时间为 $T_Delta = t_1 - t_0 > 7.28\mu\text{s}$ 。由于设计的同步中断源信号落后于同步采样信号，为了确保 DSP 在接收到同步中断的触发信号后能完整获取 RapidIO 位置数据，因此，至少在同步采样信号延迟 T_Delta 时间后，触发一次同步中断源信号。通过 GPIO 同步中断技术，实现多个 DSP 的同时进入 GPIO 中

断。在该 GPIO 中断服务子程序中，设置多核同步的触发源，依据基于广播事件的多核同步技术，实现 DSP 各核的中断实时程序同步执行。

对于多核 DSP，每个核上部署的程序均采用背景循环程序和中断实时程序两部分构成，因此，DSP 整体时序相对比较单一，但其中断实时程序具有非常严格的实时性，且需有立即响应能力。

4.6 本章小结

针对工件台运动控制系统的同步要求，并在此基础上根据 C6678 中断映射原理，提出了两种同步测控方案。经过比较，相对于基于软件中断的方案，基于硬中断的方案使系统更具实时性。针对基于 Direct I/O + GPIO 模式的测控方案，结合 MC_4DSP_VPX 卡 GPIO 引脚连接原理图，详细介绍了基于 GPIO 的多 DSP 同步中断技术，然后对基于广播事件的多核同步技术和基于核间中断的多核同步技术的具体实现方法进行了介绍及比对分析。为了方便进行系统同步时序设计，将数据流结构简化成最小测控模型，并利用以上的技术手段，对系统同步时序进行了最优化设计及分析。

第五章 工件台运动控制系统软件初步设计及实验验证

工件台运动控制系统软件设计主要在于多核 DSP 伺服控制软件初步开发，包括多核任务规划、多核 DSP 软件程序流程设计等。然后，在现有测试平台上，第一阶段测试基于 Direct I/O + Doorbell 模式和基于 Direct I/O + GPIO 模式的测控方案中的数据传输延时，对其实时性进行分析；第二阶段集中于多核同步实时性测试，最后阶段联合测试系统的总体实时性，通过实验验证方案的可行性。

5.1 控制系统软件设计

5.1.1 工件台控制系统总体框架

依据工件台控制系统硬件体系结构和数据流方案，VPX 工件台软件系统总体框架主要分为上位机软件、中间层 PowerPC 主控软件和底层 DSP 伺服控制软件，如图 5-1 所示。三部分负责不同的任务并且相互分工协调，实现一个大而细的系统功能，层与层之间通过网络 TCP/IP 和 RapidIO 进行数据信息交互。其中，PowerPC 主控软件、上位机软件分别用 VxWorks 操作系统开发和 C# 开发。本文仅对工件台底层 DSP 伺服控制软件设计进行详细介绍。

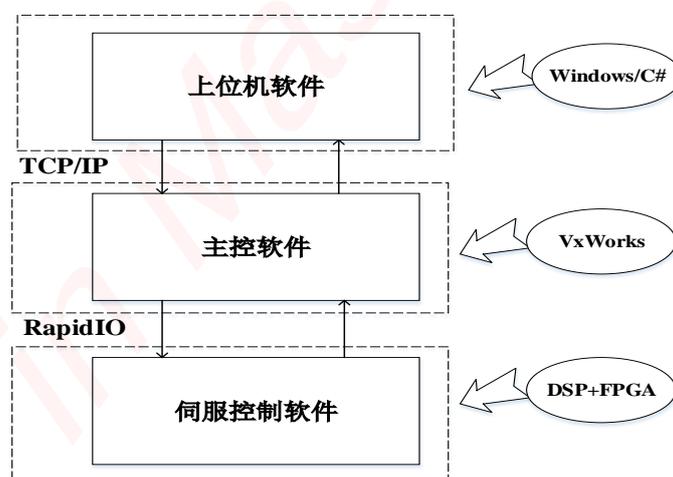


图 5-1 VPX 工件台控制系统软件总体框图

5.1.2 工件台 DSP 伺服控制软件总体设计

在进行软件设计之前，首先需要在多 DSP 上对工件台的任务进行详细、合理的部署。针对工件台多达 44 轴的运动控制，综合分析了各个运动轴间的计算耦合性，以及各模块间的共享数据传递需求等因素，将多轴的伺服控制任务划分到若干

个 DSP 核中。根据第二章中光刻机工件台控制系统设计需求分析以及对传感器和执行器统计情况,结合各板卡所能处理的信号,工件台控制系统拟采用两个 C6678 处理芯片,共 8 个 DSP 核并行完成双硅片台和掩模台的运动控制任务。

5.1.2.1 多核任务规划

1) 掩模台任务规划

如图 5-2 所示,MC_4DSP_VPX 运动控制卡#1 中 DSP #11 负责完成掩模台的控制计算。掩模台微动台的电涡流传感器信号、激光干涉仪信号以及粗动台和平衡块的光栅传感器信号均通过 FC_FPGA_VPX 光纤接口卡#1 采集,并通过 RapidIO 总线传输给 DSP#11。DSP #11 的核 0 用来控制掩模台微动台,驱动微动台 8 个电机动作;核 1、核 2 分别完成粗动台 2 个电机、平衡块 2 个电机的伺服驱动;其余核备用。

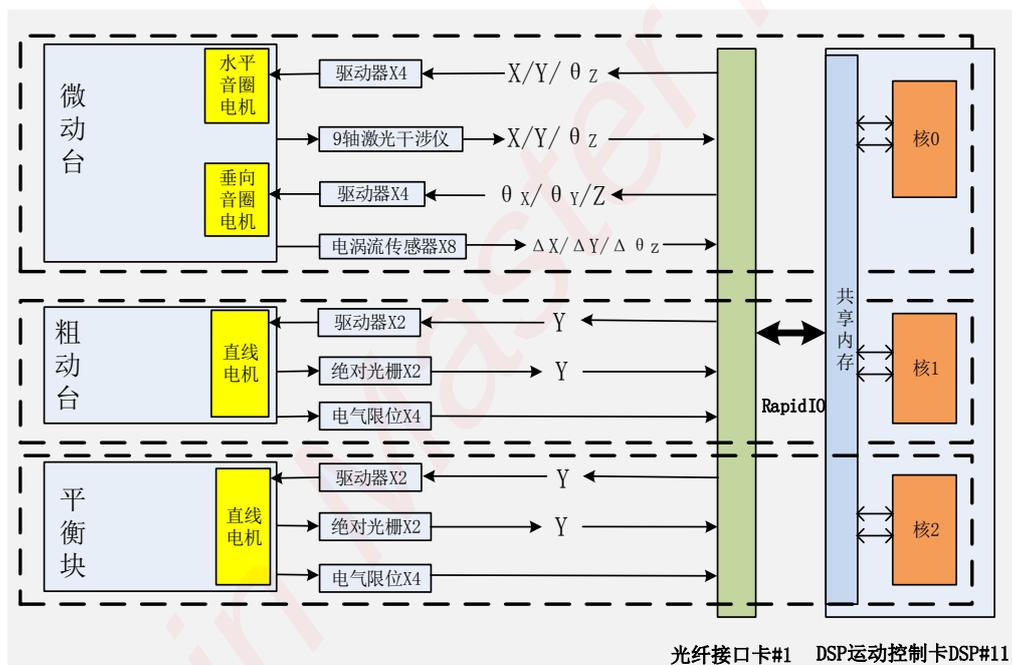


图 5-2 掩模台任务规划

2) 硅片台任务规划

MC_4DSP_VPX 卡#1 中 DSP #12 用来控制双硅片台,如图 5-3 所示,其中,核 0、核 1 分别负责 A 硅片台的微动台位置伺服计算和粗动台、线缆台以及滑块位置伺服计算;核 2、核 3 分别完成 B 硅片台的微动台和粗动台、线缆台以及滑块的控制;硅片台平衡块的运动控制计算由核 4 实现;同时考虑到系统冗余度,核 5、6、7 保留备用。硅片台所有传感器信号由 FC_FPGA_VPX 卡#2 来采集,之后通过

RapidIO 总线将数据进行传输至 DSP#12 的共享内存中，通过自定义的多核共享内存数据交互机制，实现数据能被核 0 至核 4 获取。

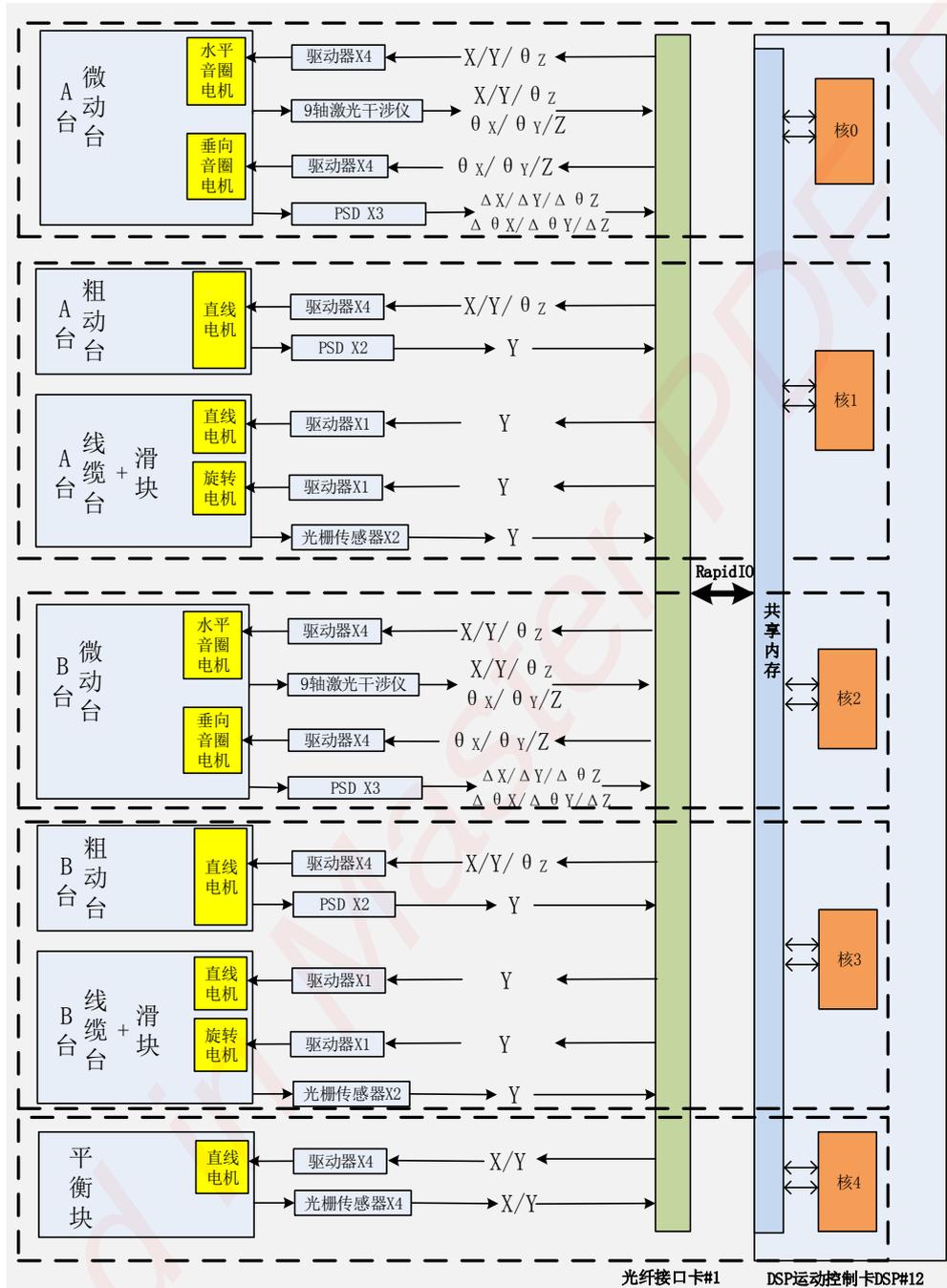


图 5-3 硅片台任务规划

5.1.2.2 多核 DSP 软件程序流程设计

多核 DSP 伺服控制软件以 CCS5.3 为开发环境，软件整体设计采用单镜像 (Single Image) 软件多任务模式，即在同一个软件工程下，DSP 多个核使用同一

份应用代码和相同的数据段，程序根据 DSP 核 ID 号对不同的控制台进行分支同步处理，实现不同的任务。

以工件台掩模台为例，其伺服控制软件具体流程如图 5-4 所示，程序首先对 DSP 设备初始化，如电源、时钟锁相环、核级中断控制器、片级中断控制器、GPIO 和 RapidIO 外设等初始化。通过函数 CSL_chipReadReg() 获取各核号，根据核号执行相应分支中的控制台伺服程序，每个分支的伺服程序均由平台参数初始任务、背景循环任务和伺服中断任务三部分组成，自行构建一个简单的实时控制系统。平台参数初始任务主要包括相应平台运动模式、轨迹参数、测量系统参数、控制器参数执行器参数初始化等。背景任务负责各 DSP 与 PowerPC 的非实时 RapidIO 通信以及非实时数据处理。准备工作就绪后，必须使能中断，以接收 100us 定时中断信号。100us 伺服周期信号由外部定时控制器统一生成，经过多 DSP、多核同步中断技术实现周期性触发各核进入伺服中断服务程序中。工件台粗、微动台和平衡块伺服轴的调制解算由伺服中断任务实现，包括从共享内存中的 RapidIO 数据缓冲区读反馈数据、滤波、位姿解算、控制量计算、电机解耦、通过 RapidIO 发送电机控制量到 FPGA 和发送控制过程数据到 PowerPC。

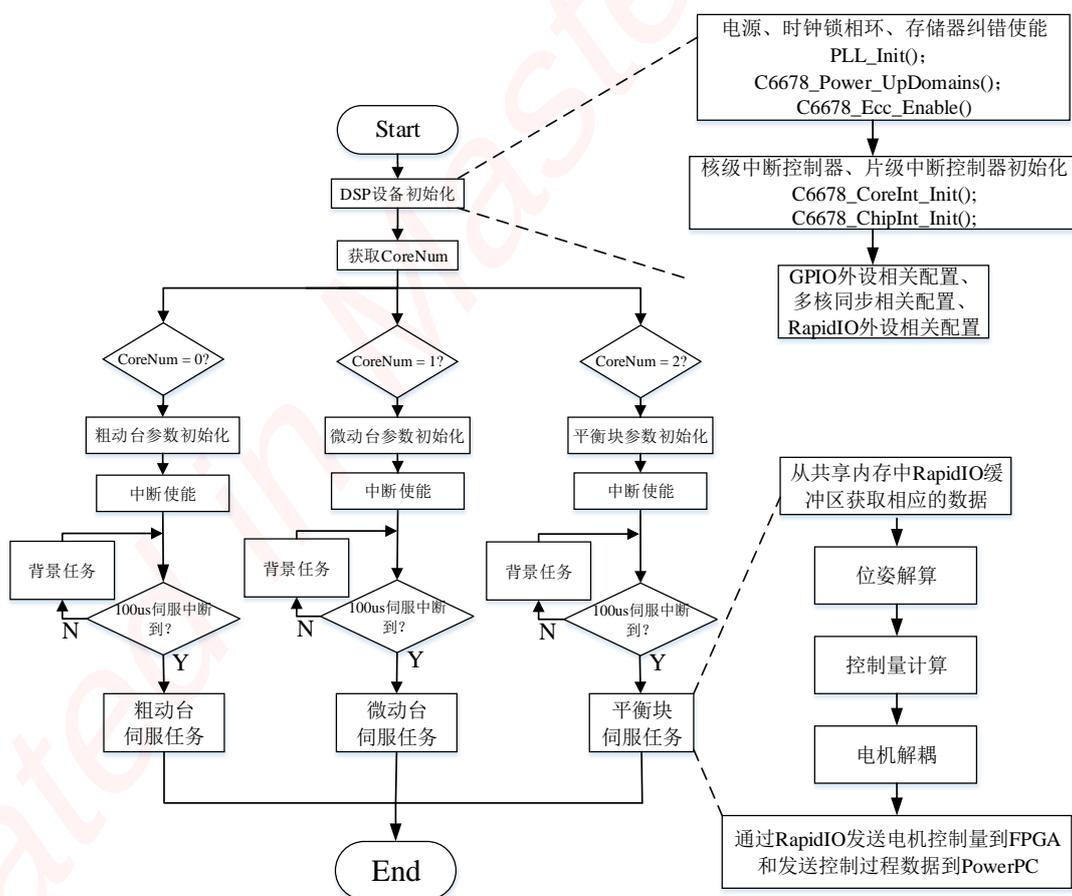


图 5-4 多核 DSP 伺服控制软件程序流程图

5.2 控制系统测试与分析

5.2.1 测试系统平台简介

为了验证运动控制系统的硬件可靠性以及软件的实时性、可行性，由于 HOST_CPU_VPX 卡、FC_FPGA_VPX 卡均处于开发阶段，只能根据部分需求搭建相应的测试系统平台，平台采用现有的运动控制卡 MC_4DSP_VPX 来进行部分功能验证，该卡由 4 片 C6678、1 片 K7 FPGA 以及少量光纤接口组成。其中，FPGA 和光纤接口可以验证 FC_FPGA_VPX 卡的功能。图 5-5 为最小化测试系统平台，包括：

- 1) 硬件环境：VPX 机箱、MC_4DSP_VPX 运动控制卡、电源、PC 机；
- 2) 软件环境：CCS5.3；
- 3) 测试工具：Wintech XDS560 V2 系列 DSP 仿真器。

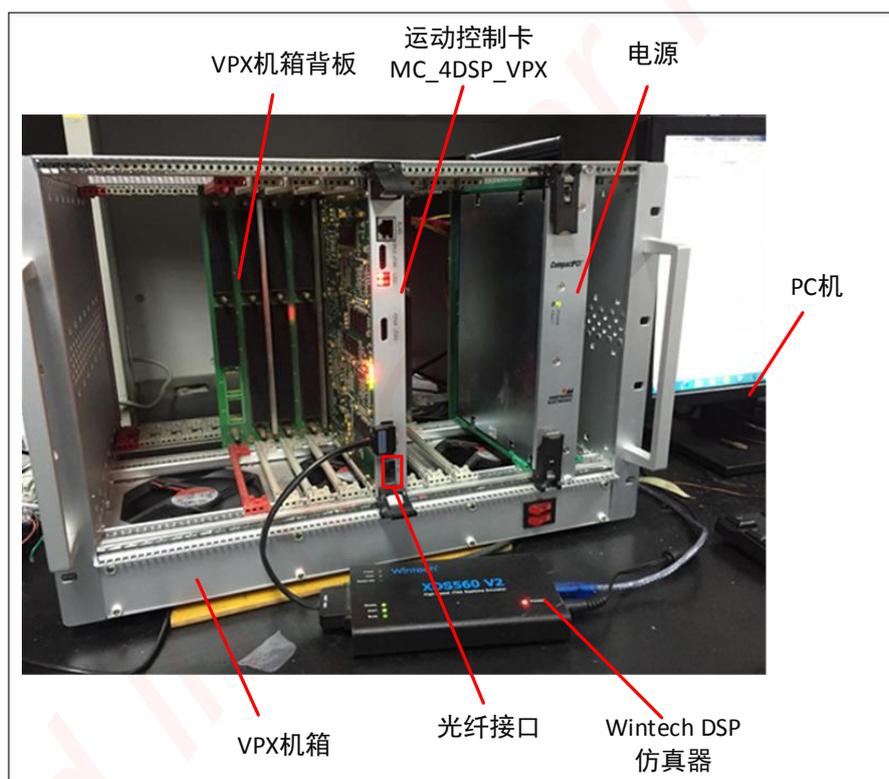


图 5-5 最小测试系统平台实物图

由于硬件条件有限，目前只能对基于 Direct I/O + Doorbell 模式和基于 Direct I/O + GPIO 模式的同步测控方案进行实时性测试，对 DSP 多核同步响应延时以及程序总体执行时间进行测试，进而根据所测试的实验数据，分析整个系统的实时性是否得到满足。

5.2.2 RapidIO 数据传输实时性测试

根据 4.3 节所设计的方案, 针对 FPGA 和 DSP 之间的 RapidIO 数据传输实时性, 分别进行了基于 Direct I/O + Doorbell 模式的数据传输延时测试和基于 Direct I/O + GPIO 模式的数据闭环传输延时测试。

5.2.2.1 两种模式下的数据传输延时测试

1) 测试方法

根据所调研的文献, 由于 RapidIO 总线主要用于传输大量数据的场合, 如雷达系统、基站系统等, 然而在运动控制系统中, 数据量级相对小很多。因此, 实验会根据工件台运动控制系统的数据交互需求, 选取合适的数据包进行测试。

DSP0 分别通过 Direct I/O+Doorbell 模式和 Direct I/O +GPIO 模式向目标端点 FPGA 分别发送 64B、128B、256B、512B、1KB、2KB、4KB、8KB、32KB、64KB、128KB、256KB、512KB、1MB 等多个不同大小数据包, 在发送数据包之前通过时间戳来记录当前时刻 t_0 , 然后再向 FPGA 发送 Doorbell 软中断或 GPIO 硬中断, FPGA 检测到中断信号后, 将接收到的数据包返回至 DSP0, 并发送中断通知 DSP0 数据发送完毕, 记录此时时刻 t_1 , t_1-t_0 即为 FPGA 与 DSP 的闭环传输时间。

2) 数据闭环传输延时结果

针对不同大小的数据包, 从每组测试中随机采集 1000 个测试数据点, 分别计算出不同数据量的平均闭环传输延时, 记基于 DirectIO + Doorbell 模式的数据闭环传输延时为 $t_delay_close_DD$, 基于 DirectIO + GPIO 模式的数据闭环传输延时为 $t_delay_close_DG$, 结果如表 5-1 所示。

表 5-1 不同数据量下两种模式的 RapidIO 传输延时结果

模式 \ 数据量	64B	128B	256B	512B	1KB	2KB	4KB
DD 闭环传输延时(us) $t_delay_close_DD$	19.168	19.197	19.257	19.520	20.645	21.222	23.481
DG 闭环传输延时(us) $t_delay_close_DD$	10.248	10.277	10.337	10.600	11.723	12.302	14.561
模式 \ 数据量	8KB	32KB	64KB	128KB	256KB	512KB	1MB
DD 闭环传输延时(us) $t_delay_close_DD$	29.357	87.868	165.685	321.33	632.63	1255.23	2500.4
DG 闭环传输延时(us) $t_delay_close_DG$	20.437	78.948	156.765	312.42	623.72	1246.3	2491.5

图 5-6 为两种模式下的闭环传输延时对比, 由于系统要求传输数据量在 800B-

4KB 之间，当传输数据量从 64B 增长到 4KB 的时候，数据量变为原来的 16 倍，而闭环传输延时却只增加了 4.31us，说明在此段数据范围内，传输延时变化较小。

实验比较可知，采用 Direct I/O+GPIO 硬中断（延时一般小于 1us）传输方式时，其数据传输只需 14.56us，因此同步中断源信号至少要比时钟源信号延迟 $14.56/2 = 7.28us$ 。而基于 Direct I/O+Doorbell 模式的数据传输延时最大约为 23.48us，其中 Doorbell 双向中断响应所占用的时间就达 8.92us，则其单向中断响应延时为 4.46us，相对于 100us 的伺服采样周期，所造成的影响较大。Doorbell 中断响应延时测试结果如图 5-7。

此外，由于 Doorbell 无法实现广播中断给多个 DSP，其需要依次发送门铃中断至不同 DSP，导致多 DSP 不能同步接收数据，不满足工件台强实时性同步系统的设计要求，严重制约了系统伺服采样频率的提升。因此，基于 Direct I/O+GPIO 模式的数据传输实时性更强于基于 Direct I/O+Doorbell 模式。

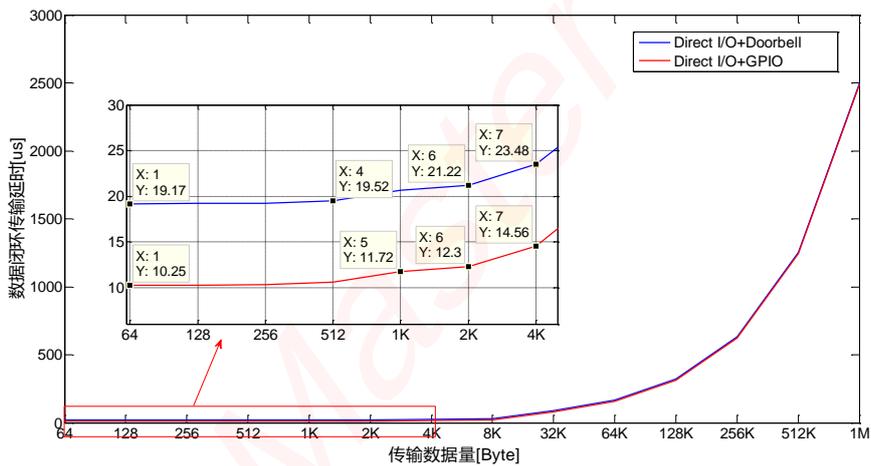


图 5-6 不同数据量时两种模式下闭环传输延时对比

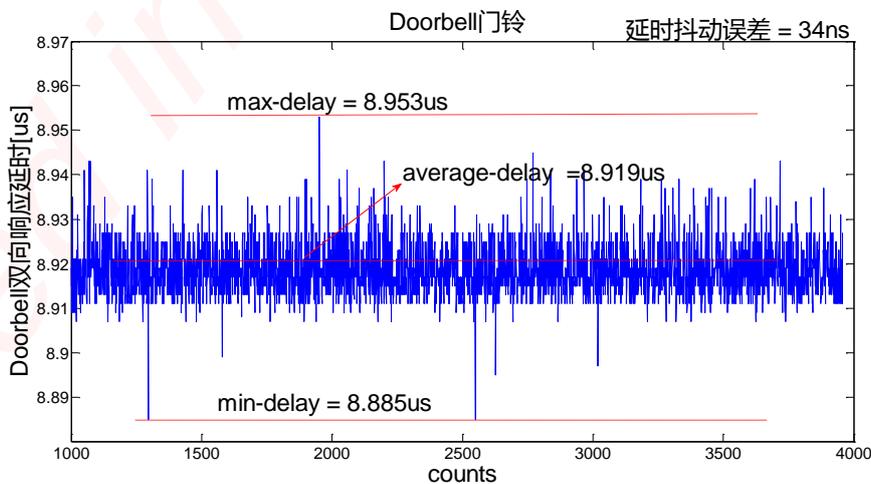


图 5-7 门铃 Doorbell 中断双向响应延时

5.2.2.2 数据传输稳定性分析

系统的数据传输稳定性保证了运动控制卡与光纤接口卡的长时间稳定数据传输交互，若在某一时刻，数据传输稳定性误差较大，可能导致运动控制系统在该伺服周期内出现数据流不稳定或共享内存中的数据交互出现问题。因此数据传输稳定性对于运动控制系统至关重要。由于系统要求数据传输量在 800B-4KB 之间，分别取 1KB、2KB、4KB 的数据传输包，针对基于 Direct I/O+GPIO 模式的同步测控方案，测得相应数据的闭环传输延时稳定性，结果如图 5-8、5-9、5-10 所示。实验表明，三种情况下的延时抖动误差均小于 50ns，误差率分别为 0.37%、0.39%、0.31%，完全符合数据传输稳定性要求。

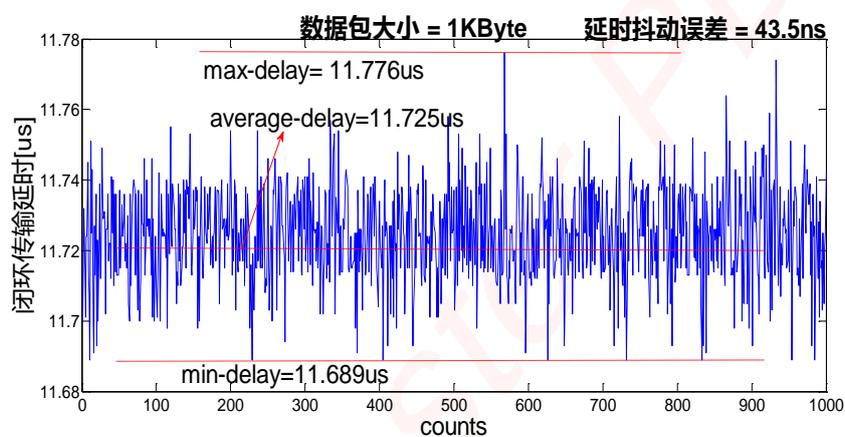


图 5-8 数据量为 1KByte 时 RapidIO 双向传输延时稳定性

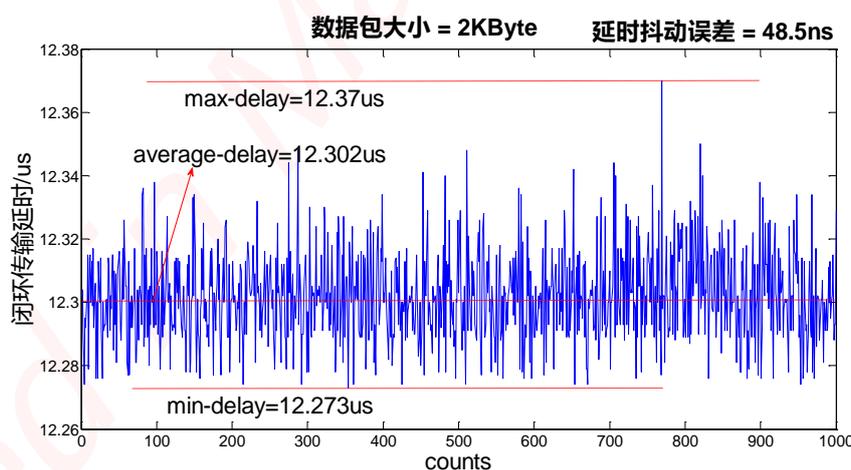


图 5-9 数据量为 2KByte 时 RapidIO 双向传输延时稳定性

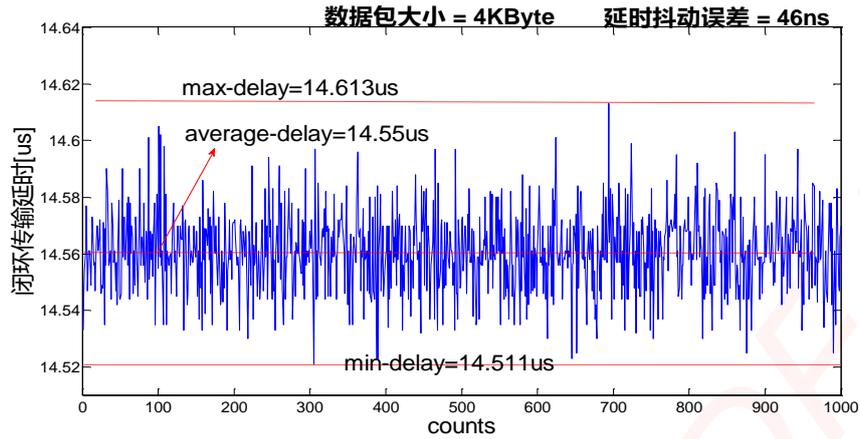


图 5-10 数据量为 4KByte 时闭环传输延时稳定性

5.2.3 多核同步实时性测试

5.2.3.1 精确系统时钟测试

精确的时钟源是实现稳定伺服周期的基础。在采用多核同步中断技术测试多核的伺服中断周期之前，首先需要有一个精确的时钟源。

图 5-11 为 MC_4DSP_VPX 卡内定时器产生的理论定时周期为 100us 时，所记录的 1000 次实际定时周期值，从图上可以看出，最大、最小的定时周期分别为 100.036us 和 99.970us，相对于 100us 理论定时周期，时钟抖动误差在 -30ns~+36ns，误差率为 -0.03%~+0.036%，对系统同步中断造成的影响非常小，可以忽略不计。

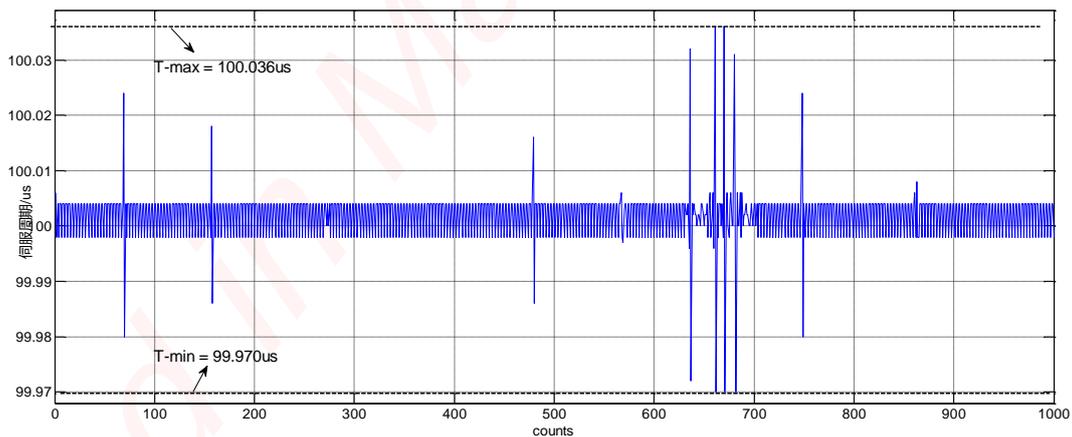


图 5-11 100us 实际定时周期值

5.2.3.2 多核同步中断实时性测试

1) 基于广播事件的多核同步实时性测试

在精准的系统时钟的基础上，按照基于广播事件的多核同步技术实现方法，对

该方法的实时性进行了试验。测试结果如图 5-12，从 Core0 接收到定时同步中断到其他几个核响应时所耗时间为 $576\text{ns} < 1\mu\text{s}$ 。

g_uiTime2	unsigned int[1000]	0x0080EE80
[0 ... 99]		
(0)= [0]	unsigned int	576
(0)= [1]	unsigned int	576
(0)= [2]	unsigned int	576
(0)= [3]	unsigned int	576
(0)= [4]	unsigned int	576
(0)= [5]	unsigned int	576
(0)= [6]	unsigned int	576
(0)= [7]	unsigned int	576
(0)= [8]	unsigned int	576
(0)= [9]	unsigned int	576
(0)= [10]	unsigned int	576
(0)= [11]	unsigned int	576

图 5-12 基于广播事件的多核同步响应延时结果

2) 基于 IPC 中断的多核同步实时性测试

由于不同核之间的时域不一样，因此，跨核时域的测试必须以一个核的时钟为基准。为了保证实验的有效性，在 Core0 上使能本地计时器，发送 IPC 中断至 Core1 之前记录时间戳为 t_0 ，Core1 接收到中断后，立即向 Core0 返回 IPC 中断，并记下时间戳 t_1 ，则两次中断响应总延时为 $t_{\text{Delay}} = t_1 - t_0$ 。通过多次往返发送，实验测得核间中断 IPC 双向中断响应延时约为 $12.8\mu\text{s}$ ，测试结果如图 5-13 所示。从测试结果可知，基于核间中断 IPC 的多核同步的单向中断响应延时约 $6.4\mu\text{s}$ ，相比基于广播事件的多核同步 576ns ，延时太大，影响 C6678 多核实时性及同步性。因此，基于广播事件的多核同步技术更优于基于核间中断 IPC 的多核同步技术。

```

4DSPTargetConfiguration.ccxml:CIO
! Core0 to Core1 time is 12.682000 us
! Core0 to Core1 time is 12.656000 us
! Core0 to Core1 time is 12.690000 us
! Core0 to Core1 time is 12.555000 us
! Core0 to Core1 time is 12.659000 us
! Core0 to Core1 time is 12.774000 us
! Core0 to Core1 time is 12.486000 us
! Core0 to Core1 time is 12.689000 us
! Core0 to Core1 time is 12.621000 us
! Core0 to Core1 time is 12.679000 us
! Core0 to Core1 time is 12.727000 us
! Core0 to Core1 time is 12.374000 us
! Core0 to Core1 time is 12.734000 us
! Core0 to Core1 time is 12.735000 us

```

图 5-13 基于核间中断 IPC 的多核同步响应延时结果

5.2.4 系统总体实时性测试

为了表征系统总体实时性，选取伺服运算量最大的程序代码，由于微动台为六

自由度平台，其位置解算、算法计算、电机解耦等比较复杂，因此，移植原 VME 系统中的微动台程序至 VPX 系统中，进行系统实时性测试。如图 5-14、图 5-15 所示，在 VME 系统中，该程序的中断任务最长执行时间约 144us 左右，而在 VPX 系统中仅仅只需 26.66us，加上数据传输延时 14.56us 及中断响应延时 576us，所耗时仅为 41.796us<100us，可满足系统 100us 的实时性要求。

正弦/四...	8	0	0	
正弦/四... 0	9	0	0	
微动台轨迹	10	0	0	
	11	0	0	
在线/离... 1	Y	50431	50419	VME系统测试
正弦/四... 2	motoFY	0	0	伺服中断程序耗时约
曝光监控	motoFZ	0	0	144us
	motoFX	0	0	
微曝光... 0	报错	12	12	
	时间	143940	143100	

图 5-14 VME 系统中伺服中断程序执行时间

```

4DSPTargetConfiguration.ccxml:CIO
the cycle time is 26657 ns,26.657000 us
the cycle time is 26659 ns,26.659000 us
the cycle time is 26657 ns,26.657000 us
    
```

VPX系统测试

伺服中断程序耗时约 26.66us

图 5-15 VPX 系统中伺服中断程序执行时间

5.3 本章小结

本章以双硅片台、掩模台等多控制台为对象，针对多核 DSP 进行了工件台伺服控制软件框架总体设计，包括多核任务规划、多核 DSP 软件程序流程设计。并搭建了最小测试系统平台上，分别对基于 Direct I/O+Doorbell 模式、基于 Direct I/O+GPIO 模式的数据传输实时性和基于广播事件、基于核间中断 IPC 的多核同步响应延时以及系统总体实时性等进行了测试，实验表明基于 Direct I/O+GPIO 模式的数据闭环传输延时仅约 14.56us 左右，基于广播事件的多核同步响应 576ns，总体执行时间为 41.796us，远远小于伺服周期 100us，符合系统要求，验证了方案、软件设计的正确性和系统的实时性、同步性。

第六章 总结与展望

6.1 研究内容总结

论文以光刻机工件台样机为对象,以实现工件台测控系统的并行性、同步性和实时性为目标,提出了一种基于 VPX 总线的工件台运动控制系统总体架构方案,围绕基于共享内存的系统数据交互机制和系统精密同步测控策略开展了较深入的研究与分析。具体工作总结如下:

(1) 基于 VPX 总线的工件台运动控制系统总体架构方案的提出

针对基于 VME 总线的工件台运动控制系统存在的问题,提出一种基于 VPX 总线的工件台运动控制系统设计方案。由于 RapidIO 总线具有高传输带宽、互连灵活等特点,采用 RapidIO 全网状星型拓扑结构,并结合 TMS320C6678 多核运动控制卡 MC_4DSP_VPX 和 FPGA 光纤接口卡 FC_FPGA_VPX 等硬件,对系统进行了硬件体系架构设计。该系统可大幅提高多处理器之间的数据吞吐率以及系统的可扩展性和实时性。

(2) 基于共享内存的数据交互机制的研究

根据系统硬件体系结构,对控制系统的数流结构进行了设计,在此基础上,针对系统要求建立并行处理模型,并归纳总结出系统数据交互内容;针对工件台数据交互需求,建立了基于 C6678 多核共享内存的数据交互机制。该机制采用 4M 共享内存作为数据缓冲通道,通过遵循 C6678 的数据 Cache 和共享内存的数据一致性原理,并根据交互数据内容,对共享内存和核本地内存进行存储空间划分,进而实现了多核并行访问共享内存,进一步提升了传感器反馈数据与控制器计算数据的传输实时性和可靠性。

(3) 控制系统精密同步测控策略研究

为了实现多 DSP 实时同步,提出并比较了两种同步测控方案,理论分析和实验证明,基于 GPIO 硬中断的同步测控方案相比于基于门铃 Doorbell 软中断的同步测控方案更具同步性和实时性;在基于 GPIO 硬中断的同步测控方案中,对其基于 GPIO 的多 DSP 同步技术具体实施方法进行了详细介绍;为了实现 C6678 的多核同步,具体阐述了两种多核同步技术方法,比较分析可知,基于广播事件的多核同步技术更能满足强实时系统设计的要求。

(4) 控制系统软件的初步设计与测试

以双硅片台、掩模台等多控制台为对象,对多核 DSP 软件进行了整体架构设计。搭建了最小测试系统平台,对系统中 RapidIO 的 Direct I/O 数据传输实时性、

门铃 Doorbell 中断响应实时性、多核同步响应延时以等进行了测试，得出 Direct I/O 双向传输延时约 14.56us 左右，多核同步响应 576ns，总体执行时间为 41.796us，远远小于预定的伺服周期 100us，满足系统要求，进而验证了方案、软件设计的正确性和系统的实时性、同步性。

6.2 工作展望

本文完成了基于 VPX 总线的工件台系统的初期方案设计，并重点对 C6678 多核访问共享内存和同步测控策略进行了分析、实现和测试。但由于项目实验条件受限以及研发时间不够，本课题所提出的方案仍然存在不足，一些技术问题需要进一步得到解决，主要包括以下几个方面：

(1) 本文仅对 RapidIO 组播方式对光纤接口卡同步并行发送反馈数据至多 DSP 进行了分析，未对多 DSP 如何将控制过程数据返回至 FPGA 的方式进行研究，该过程涉及到系统的实时性，有待进一步分析。

(2) 对于多核任务规划，本文一个核负责控制一个运动平台，如掩模台微动台或粗动台，没有充分利用 C6678 的八个核，如何减小一个运动平台的控制程序各个模块的耦合性，并将其部署至 C6678 多个核中，增强 C6678 利用率对提高系统实时性还有待深入研究。

(3) 目前的控制卡上由于 C6678 的八个核共用一个 RapidIO 外设，仅用于多核同时通过 RapidIO 传输数据的场合，可以考虑不同的核如何分时利用 RapidIO 外设，并提高 RapidIO 外设的互斥性和灵活性。

致 谢

衷心感谢导师朱煜教授提供了优异的科研平台和良好的学术氛围，在学习过程中，给了我精心的指导，他的言传身教将使我终生受益。

感谢杨开明老师深切的关怀，不管是在课题研究中还是在论文写作上，都给予了我很多指导和建议，帮我克服了很多难题，并树立了正确的科研态度。

感谢 IC 装备实验室尹文生老师、胡金春老师、穆海华老师、张鸣老师、徐登峰老师、李鑫老师、杨进老师、余东东、蒋毅、陈涵、李敏、黄涛、鲁森，祁利山、吴亚风、吴双、陈金水、代艳鹏、袁翠平、肖培、喻强、申飞、季钊澎、石淼、王希远等老师和同学以及苏哲欣、梁志敏等工程师的热情帮助和支持。

感谢参加论文评审和论文答辩的各位老师。

参考文献

- [1] 朱贻玮.我国 IC 芯片制造线目前状况分析[J].半导体光伏行业,2010(01):3-4
- [2] 王公峰.双工件台控制系统设计及单自由度试验研究[D]. 哈尔滨:哈尔滨工业大学,2013
- [3] C. Mack. Milestones in Optical Lithography Tool Suppliers [EB/OL]. http://www.lithoguru.com/scientist/litho_history/milestones_tools.pdf, 2005
- [4] M. Quirk, J. Serda. Semiconductor manufacturing technology[M]. Beijing: Publishing House of Electronics Industry, 2006
- [5] 王锐延.半导体晶圆自动清洗设备[J].电子工业专用设备, 2004(12): 8-12
- [6] 王波,董申,赵万生.超精密定位中的几项相关关键技术[J].航空精密制造技术, 1998, 34(3): 13-16
- [7] 杨一博.粗精动平台的系统辨识与解耦控制研究[D].北京: 清华大学,2009
- [8] 董柏民.光刻技术层间套准精度的技术改进研究[D].上海: 上海交通大学,2012
- [9] 毕延帅.面向双工件台的 VxWorks 实时嵌入式系统设计与优化[D].哈尔滨: 哈尔滨工业大学,2013
- [10] B. Sluijk, T. Castenmiller, R. C. Jongh, et al. Performance Result of New Generation of 300mm Lithography Systems[C]. Proceedings of SPIE, 2001. 43(46): 544-557
- [11] R. Rubingh, Y. V. Dommelen, S. Tempelaars, et al. Performance of a high productivity 300 mm dual stage 193 nm 0.75 NA TWINSCAN at: 1100B system for 100 nm applications [J]. J. Micro/Nanolith. MEMS MOEMS, 2003, 2(1):8~18
- [12] 王宝仁.网络化运动控制系统多轴协同关键技术研究[D].山东: 山东大学,2008
- [13] 英国并行技术公司.MicroTCA 和 AMC 模块综述[A].中国计算机行业协会[C].第五届中国 PICMG 技术年会论文汇编.中国计算机行业协会,2006
- [14] A. P. Rodrigues, L. Pereira, T. I. Madeira, et al. Real-time multi-DSP based VME system for feedback control on the TCV Tokamak[J]. IEEE Transactions on Nuclear Science, 2006, 53(3):845-848
- [15] VPX 高可靠性军用加固通讯计算平台[J].国防制造技术,2011,03:11
- [16] 蒙佳.VPX 抗恶劣环境通信计算平台关键技术研究[J].测控技术,2012, 3: Vol 31,No.10
- [17] 王丽,付月生,陈思思.基于 VPX 总线的系统主控模块的设计与实现[J].电子设计工程, 2014, 14:117-119
- [18] 盛骁.基于 MicroTCA 平台的多核 MAC 子卡设计[D].北京: 北京邮电大学, 2014
- [19] 贺信.基于 MTCA 架构和软交换体系的 ISDN 网关设计[D].南京: 南京航空航天大学, 2012

- [20] 王正华.ASML 的创新之路[J].中国集成电路,2006,15(6):58-62
- [21] W. D. Boeij, G. Dicker, M. D. Wit, et al.Extending KrF lithography beyond 80nm with the TWINSCAN XT:1000H 0.93NA scanner[C].Lithography Asia 2008. Part one of two parts.2009:71401B:1-71401B:11
- [22] M. F. Ozturk. An FPGA based motion controller[D]. Eindhoven University of Technology .2011, 14-20
- [23] V. Camelo. Multi-core CPU Exploration for CARM Host in ASML Technology[D]. Master thesis Eindhoven University of Technology, 2012
- [24] D. S. P. Hernandez. Design-Space Exploration for High-Performance Motion Control[J]. 2013
- [25] A. E. E. Gozek, J. P. M. J. Voeten, R. M. W. R. Frijns, et al. Task Execution Time Prediction for Motion Control Applications[J]. 2013
- [26] 毕延帅.面向双工件台的 VxWorks 实时嵌入式系统设计与优化[D].哈尔滨: 哈尔滨工业大学,2013
- [27] 王春.国产首台先进封装光刻机投入使用[N].中国技术市场报,2010-08-06001
- [28] 苏哲欣, 廖凯.工件台系统 $\alpha 1$ 样机控制系统设计评审[R].北京: 清华大学内部资料
- [29] S. W. Jambekar. Performance Improvement of Motion-control Applications Using Multi-ASIP in FPGA[D]. Delft University of Technology, 2014:42-44
- [30] R. M. W. Frijns. Platform-based Design for High-Performance Mechatronic Systems[D], Eindhoven University of Technology. 2015
- [31] 何玉红, 赵琨.基于 FPGA 的 RapidIO 总线接口设计与实现技术[J].计算机与网络, 2012, 13:57-59
- [32] 汪星宇. RapidIO 技术在信号处理系统中的应用与研究[D].南京: 南京理工大学, 2009
- [33] 欧阳益民.高速数字图像并行处理系统[D].成都: 电子科技大学, 2008
- [34] 李鑫, 姜明.多核 DSP 高速实时信号处理系统设计[J].光学技术. 2012, 38 (1) :116-120
- [35] 杨方.基于 TMS320C6678 的多核 DSP 并行处理应用技术研究[D].北京: 北京理工大学,2014
- [36] 王耀.基于 VME 总线的多处理器运动控制卡[D].武汉: 华中科技大学, 2012
- [37] 陈冰, 陈幼平, 谢经明, 等. 运动控制网络的时钟同步[J]. 中国机械工程, 2007, 2(18): 331-334
- [38] IDT 推出全新 Serial RapidIO Gen2 交换器系列[J].电子与电脑, 2010, 07:89
- [39] 张颖川.多核数字信号处理平台的设计与实现[D].西安: 西安电子科技大学,2013
- [40] 丁有源, 汪安民.基于多核任务并行处理的 DSP 软硬件设计[J].单片机与嵌入式系统应用, 2012, 05:43-45

- [41] 周佩.基于多核 DSP 并行调度机制的实现[D].成都: 中国科学院研究生院(光电技术研究所), 2014
- [42] TMS320C6678 Multi-core Fixed and Floating-Point Digital Signal Processor[EB/OL].
<http://www.ti.com.cn/cn/lit/ds/symlink/tms320c6678.pdf>, March, 2010
- [43] 吴灏.多核 DSP 操作系统关键技术研究及实现[D].长沙: 国防科学技术大学, 2012
- [44] 陈玉生.KeyStone 多核 DSP 中断系统原理及实现[J].电子世界, 2015, 13:23-25+35

攻读硕士学位期间取得的研究成果

- [1] 朱煜, 杨开明, 成荣, 廖晨翔等.一种基于 VPX 总线的工件台同步运动控制系统及方法[P].
北京:专利号:201510885509.3

嵌入式资源免费下载

总线协议:

1. [基于 PCIe 驱动程序的数据传输卡 DMA 传输](#)
2. [基于 PCIe 总线协议的设备驱动开发](#)
3. [CANopen 协议介绍](#)
4. [基于 PXI 总线 RS422 数据通信卡 WDM 驱动程序设计](#)
5. [FPGA 实现 PCIe 总线 DMA 设计](#)
6. [PCI Express 协议实现与验证](#)
7. [VPX 总线技术及其实现](#)
8. [基于 Xilinx FPGA 的 PCIE 接口实现](#)
9. [基于 PCI 总线的 GPS 授时卡设计](#)
10. [基于 CPCI 标准的 6U 信号处理平台的设计](#)
11. [USB30 电路保护](#)
12. [USB30 协议分析与框架设计](#)
13. [USB 30 中的 CRC 校验原理及实现](#)
14. [基于 CPLD 的 UART 设计](#)
15. [IPMI 在 VPX 系统中的应用与设计](#)
16. [基于 CPCI 总线的 PMC 载板设计](#)

VxWorks:

1. [基于 VxWorks 的多任务程序设计](#)
2. [基于 VxWorks 的数据采集存储装置设计](#)
3. [Flash 文件系统分析及其在 VxWorks 中的实现](#)
4. [VxWorks 多任务编程中的异常研究](#)
5. [VxWorks 应用技巧两例](#)
6. [一种基于 VxWorks 的飞行仿真实时管理系统](#)
7. [在 VxWorks 系统中使用 TrueType 字库](#)
8. [基于 FreeType 的 VxWorks 中文显示方案](#)
9. [基于 Tilcon 的 VxWorks 简单动画开发](#)
10. [基于 Tilcon 的某武器显控系统界面设计](#)
11. [基于 Tilcon 的综合导航信息处理装置界面设计](#)
12. [VxWorks 的内存配置和管理](#)

13. [基于 VxWorks 系统的 PCI 配置与应用](#)
14. [基于 MPC8270 的 VxWorks BSP 的移植](#)
15. [Bootrom 功能改进经验谈](#)

Linux:

1. [Linux 程序设计第三版及源代码](#)
2. [NAND FLASH 文件系统的设计与实现](#)
3. [多通道串行通信设备的 Linux 驱动程序实现](#)
4. [Zsh 开发指南-数组](#)
5. [常用 GDB 命令中文速览](#)
6. [嵌入式 C 进阶之道](#)
7. [Linux 串口编程实例](#)
8. [基于 Yocto Project 的嵌入式应用设计](#)

Windows CE:

1. [Windows CE.NET 下 YAFFS 文件系统 NAND Flash 驱动程序设计](#)
2. [Windows CE 的 CAN 总线驱动程序设计](#)
3. [基于 Windows CE.NET 的 ADC 驱动程序实现与应用的研究](#)
4. [基于 Windows CE.NET 平台的串行通信实现](#)
5. [基于 Windows CE.NET 下的 GPRS 模块的研究与开发](#)
6. [win2k 下 NTFS 分区用 ntldr 加载进 dos 源代码](#)
7. [Windows 下的 USB 设备驱动程序开发](#)
8. [WinCE 的大容量程控数据传输解决方案设计](#)
9. [WinCE6.0 安装开发详解](#)
10. [DOS 下仿 Windows 的自带计算器程序 C 源码](#)
11. [G726 局域网语音通话程序和源代码](#)
12. [WinCE 主板加载第三方驱动程序的方法](#)
13. [WinCE 下的注册表编辑程序和源代码](#)
14. [WinCE 串口通信源代码](#)
15. [WINCE 的 SD 卡程序\[可实现读写的源码\]](#)
16. [基于 WinCE 的 BootLoader 研究](#)

PowerPC:

1. [Freescale MPC8536 开发板原理图](#)
2. [基于 MPC8548E 的固件设计](#)
3. [基于 MPC8548E 的嵌入式数据处理系统设计](#)
4. [基于 PowerPC 嵌入式网络通信平台的实现](#)
5. [PowerPC 在车辆显控系统中的应用](#)
6. [基于 PowerPC 的单板计算机的设计](#)
7. [用 PowerPC860 实现 FPGA 配置](#)

ARM:

1. [基于 DiskOnChip 2000 的驱动程序设计及应用](#)
2. [基于 ARM 体系的 PC-104 总线设计](#)
3. [基于 ARM 的嵌入式系统中断处理机制研究](#)
4. [设计 ARM 的中断处理](#)
5. [基于 ARM 的数据采集系统并行总线的驱动设计](#)
6. [S3C2410 下的 TFT LCD 驱动源码](#)
7. [STM32 SD 卡移植 FATFS 文件系统源码](#)
8. [STM32 ADC 多通道源码](#)
9. [ARM Linux 在 EP7312 上的移植](#)
10. [ARM 经典 300 问](#)

Hardware:

1. [DSP 电源的典型设计](#)
2. [高频脉冲电源设计](#)
3. [电源的综合保护设计](#)
4. [任意波形电源的设计](#)
5. [高速 PCB 信号完整性分析及应用](#)
6. [DM642 高速图像采集系统的电磁干扰设计](#)
7. [使用 COMExpress Nano 工控板实现 IP 调度设备](#)

Created in Master PDF Editor