

基于模块化开放式架构的CNI系统动态资源分配与容错重构方法研究

金第均 陈 健 王紫君 张 磊
陕西飞机工业有限责任公司 陕西 汉中 723000

摘 要: 本文聚焦于基于模块化开放式架构的综合通信导航识别(CNI)系统,深入研究了动态资源分配与容错重构方法。详细阐述了模块化开放式架构下硬件资源灵活配置与共享的实现方式,构建了多种动态资源分配策略并分析了其技术原理,同时提出了全面的容错重构方法。研究表明,这些方法显著提升了CNI系统的性能与可靠性,为航空电子系统的发展提供了关键技术支持。

关键词: 模块化开放式架构; CNI系统; 动态资源分配; 容错重构; 航空电子系统

1 引言

随着航空电子技术的飞速发展,飞机的通信、导航、识别(CNI)系统面临着日益复杂和多样化的任务需求。传统的CNI系统基于独立设备的联合式集成,存在体积大、重量重、功耗高、灵活性差等问题。基于模块化开放式架构的CNI系统应运而生,它通过采用标准化的通用功能模块,实现了硬件资源的灵活配置与共享,提高了系统的可扩展性和可维护性。而动态资源分配与容错重构方法则是确保这种复杂系统高效、可靠运行的核心技术。

2 模块化开放式架构的 CNI 系统概述

2.1 模块化开放式架构的特点

模块化开放式架构是一种基于标准化通用功能模块的系统设计方法。在CNI系统中,其特点体现在以下几个方面:

2.1.1 硬件资源的灵活配置与共享。各个功能模块采用统一的标准化接口,如物理接口(如PCI Express、RapidIO等)和逻辑接口(如ARINC 653等航空电子标准接口),这使得不同模块之间可以方便地进行连接和通信。例如,通信模块和导航模块可以通过标准化的数据总线接口进行数据交互,实现信息的共享。系统中的处理器、内存、通信带宽等硬件资源被抽象为一个资源池,系统管理软件可以根据任务需求动态地将资源分配给不同的功能模块。例如,当通信任务繁重时,系统可以为通信模块分配更多的处理器时间和通信带宽。

2.1.2 高可扩展性。当系统需要增加新的功能时,只需添加相应的功能模块即可。例如,如果需要增加一种新的卫星通信功能,只需开发并添加相应的卫星通信模块,而无需对整个系统进行重新设计。同时,当现有模块的性能不能满足需求时,可以通过升级模块来提高系

统的性能^[1]。模块化开放式架构通常采用分布式架构,各个功能模块可以分布在不同的物理位置,通过网络进行连接。这种分布式架构使得系统可以更容易地进行扩展,提高了系统的灵活性和可靠性。

2.1.3 高可维护性。每个功能模块都具有相对独立的功能和结构,当某个模块出现故障时,可以单独进行维修或更换,而不会影响其他模块的正常运行。例如,如果导航模块出现故障,维修人员可以将其从系统中取出进行维修或更换,而系统的通信和识别功能仍然可以正常工作。

2.2 CNI系统的功能需求

CNI系统作为飞机航空电子系统的重要组成部分,需要实现通信、导航和识别等多种功能。具体来说,CNI系统需要满足以下功能需求:

2.2.1 通信功能。实现机组人员与地面人员之间的实时语音交流,要求语音质量高、延迟低。例如,采用先进的语音编码算法(如AMR-WB)和抗干扰技术,确保在复杂的电磁环境下语音通信的清晰度和可靠性。传输飞行数据、维护数据等信息,要求数据传输速率高、误码率低。例如,采用高速的数据总线技术(如以太网)和差错控制编码技术(如LDPC码),确保数据的准确传输。

2.2.2 导航功能。为飞机提供准确的位置信息,包括经度、纬度、高度等。利用多种传感器和数据源,如全球定位系统(GPS)、惯性导航系统(INS)等,通过数据融合算法提高位置确定的精度。根据飞行计划为飞机提供导航引导,确保飞机沿着预定的航路飞行。

2.2.3 识别功能。识别飞机的身份,包括飞机编号、型号等信息。采用二次雷达应答机等技术,确保飞机能够被地面雷达准确识别。在军事应用中,识别飞机的敌

我属性,防止误伤。采用先进的敌我识别系统(IFF),通过特定的询问和应答信号进行识别。

3 动态资源分配方法研究

3.1 动态资源分配策略

3.1.1 基于任务优先级的资源分配。为不同的任务设置不同的优先级,例如,飞行安全相关的任务(如导航和识别功能)具有最高优先级,通信任务根据紧急程度设置不同的优先级。采用优先级调度算法,根据任务的优先级来分配资源。当系统资源不足时,优先满足高优先级任务的资源需求,低优先级任务可能需要等待或获得较少的资源^[2]。例如,当飞机遇到危险情况时,导航和识别任务会优先获得处理器和通信带宽资源,以确保飞行安全。

3.1.2 基于负载预测的资源分配。通过对系统历史负载数据和当前状态的分析,建立负载预测模型。例如,采用时间序列分析方法(如ARIMA模型)或机器学习算法(如神经网络)来预测未来一段时间内系统的负载情况。根据负载预测结果,提前分配资源以满足未来的任务需求。例如,如果预测到未来一段时间内通信任务的数据传输量将大幅增加,系统可以提前为通信模块分配更多的通信带宽和处理器资源。

3.1.3 自适应资源分配。系统管理软件实时监测各个模块的资源使用情况和任务执行状态,根据反馈信息动态调整资源的分配。例如,当检测到某个模块的负载过高时,系统可以自动将部分任务转移到其他负载较低的模块上执行。

3.2 动态资源分配的实现

3.2.1 资源监测。(1)硬件资源监测:通过传感器和监测接口,实时监测处理器、内存、通信带宽等硬件资源的使用情况。例如,监测处理器的占用率、内存的剩余容量、通信带宽的利用率等。(2)任务状态监测:监测系统中各个任务的执行状态,包括任务的进度、优先级、资源需求等。

3.2.2 任务管理。(1)任务创建与调度:根据系统的需求创建新的任务,并根据任务的优先级和资源需求进行调度。例如,采用优先级队列来管理任务,确保高优先级任务能够优先执行。(2)任务执行与结束处理:负责任务的执行过程管理,当任务执行结束时,释放任务所占用的资源。

3.2.3 策略执行。(1)策略解析:解析预设的动态资源分配策略,根据系统的实时状态和任务需求确定资源分配方案。(2)资源调整:根据资源分配方案,通过硬件接口和软件指令对资源进行动态调整。例如,调整处理器

的频率、分配内存空间、调整通信带宽的分配等。

4 容错重构方法研究

4.1 容错重构策略

4.1.1 硬件冗余备份。在系统中设置冗余的硬件模块,如通信模块、导航模块等。当主模块出现故障时,可以切换到备份模块来继续执行任务。例如,采用双冗余的通信模块,当主通信模块出现故障时,系统可以自动切换到备份通信模块,确保通信的连续性。除了模块冗余外,还可以对关键资源进行冗余备份,如电源、时钟等。当主资源出现故障时,切换到备份资源,保证系统的正常运行。

4.1.2 软件容错算法。采用数据冗余技术,如奇偶校验码、海明码等,对关键数据进行冗余存储。当数据出现错误时,可以通过冗余数据进行纠错。采用循环冗余校验(CRC)等错误检测算法检测数据中的错误,并采用相应的纠正算法进行纠正^[3]。例如,在数据传输过程中,接收端通过CRC校验检测数据是否出现错误,如果出现错误,则请求发送端重新发送数据或采用前向纠错算法进行纠正。

4.1.3 系统重构。当系统出现故障或任务需求发生变化时,根据新的任务需求重新配置系统资源,调整任务的执行顺序和分配方式。例如,当某个通信任务失败时,系统可以重新分配通信资源,尝试重新执行该任务。在系统出现故障时,通过重新配置硬件资源和软件算法来恢复故障模块的功能。例如,当导航模块出现故障时,系统可以通过重新加载导航算法、调整传感器参数等方式来恢复导航功能。

4.2 容错重构的实现

4.2.1 故障检测与诊断。利用传感器实时监测系统的运行状态,如温度、电压、电流等参数。当这些参数超出正常范围时,可能表明系统出现故障。采用故障诊断算法(如基于模型的故障诊断、基于数据的故障诊断等)对监测到的异常数据进行分析,确定故障的类型和位置。例如,通过建立系统的数学模型,将实际监测数据与模型预测数据进行比较,当两者存在较大偏差时,判断系统出现故障。

4.2.2 重构决策。根据故障检测与诊断的结果,评估故障对系统的影响程度。如果故障导致系统部分功能失效,但不影响飞行安全,可以采用软件容错算法进行纠正;如果故障导致系统关键功能失效,需要进行硬件冗余切换或系统重构。根据故障评估结果,选择合适的重构策略。例如,当主通信模块出现故障时,选择硬件冗余备份策略,切换到备份通信模块;当数据出现错误

时,选择软件容错算法进行纠正。

4.2.3 重构执行。如果选择硬件冗余备份策略,系统管理软件通过硬件接口控制切换到备份模块。例如,通过控制开关电路将信号从主模块切换到备份模块。如果选择软件容错算法或系统重构策略,系统管理软件需要重新配置软件算法和系统资源。例如,重新加载导航算法、调整任务分配等。

5 实验与验证

5.1 实验设计

5.1.1 实验环境。搭建一个基于模块化开放式架构的CNI系统实验平台,包括多个通用功能模块(如通信模块、导航模块、识别模块等)、系统管理软件、故障注入设备等。开发相应的系统管理软件,实现动态资源分配和容错重构算法。

5.1.2 实验任务。(1)正常任务场景:模拟正常的通信、导航和识别任务,观察系统的资源分配情况和性能表现。(2)故障任务场景:通过故障注入设备模拟系统模块的故障情况,观察系统的容错重构能力和故障恢复时间。

5.1.3 实验指标

系统响应时间:从任务发起到系统响应完成的时间。

资源利用率:处理器、内存、通信带宽等资源的使用效率。

任务成功率:成功完成的任务数量与总任务数量的比值。

故障恢复时间:从系统检测到故障到恢复功能的时间。

5.2 实验结果与分析

5.2.1 动态资源分配实验结果

(1)资源分配合理性。在模拟的通信任务场景中,初始时系统有10个通信任务,其中5个为高优先级任务,5个为低优先级任务。系统处理器资源总量为100个单位,通信带宽资源总量为1000Mbps。

采用基于任务优先级的资源分配策略后,高优先级任务平均分配到处理器资源60个单位,通信带宽600Mbps;低优先级任务平均分配到处理器资源40个单位,通信带宽400Mbps。实验结果显示,高优先级任务能够优先获得足够的资源,保证了通信的及时性和可靠性。

(2)系统性能提升。在负载预测实验中,通过对历史负载数据的分析,预测到未来一段时间内通信任务的数据传输量将增加50%。系统提前为通信模块分配了额外的20个单位处理器资源和300Mbps通信带宽。实验结果表明,采用基于负载预测的资源分配策略后,系统的响

应时间从原来的平均200ms缩短到150ms,资源利用率从70%提高到85%。自适应资源分配策略根据系统的实时运行状态动态调整资源分配,使得系统在不同负载情况下都能保持较高的性能。

5.2.2 容错重构实验结果

(1)故障恢复能力。在故障注入实验中,模拟通信模块故障10次,导航模块故障8次,识别模块故障5次。

采用硬件冗余备份策略后,通信模块故障时,系统能够在平均50ms内自动切换到备份通信模块,确保通信不中断;导航模块故障时,通过重新加载导航算法和调整传感器参数,在平均80ms内恢复导航功能;识别模块故障时,在平均60ms内完成故障恢复。

(2)故障恢复时间缩短。对比采用容错重构方法前后,通信模块故障恢复时间从原来的平均200ms缩短到50ms,导航模块故障恢复时间从原来的平均300ms缩短到80ms,识别模块故障恢复时间从原来的平均250ms缩短到60ms。这表明容错重构方法能够显著提高系统的故障恢复能力。

5.3 综合性能评估

在综合实验中,共执行100个任务,其中正常任务80个,故障任务20个。采用动态资源分配与容错重构方法前,任务成功率为75%;采用该方法后,任务成功率提高到92%。通过长时间运行实验,系统在连续运行100小时的过程中,未出现因资源分配不合理或故障处理不及时导致的系统崩溃情况,表明系统的稳定性得到了显著增强。

结语

本文研究了基于模块化开放式架构的CNI系统动态资源分配与容错重构方法。通过模块化设计,系统实现了硬件资源的灵活配置与共享,提高了系统的可扩展性和可维护性。动态资源分配方法能够根据任务需求实时调整资源的分配,确保了系统的高效运行。容错重构方法则通过硬件冗余备份、软件容错算法和系统重构策略提高了系统的可靠性。实验结果表明,该方法能够显著提升CNI系统的性能与可靠性,为航空电子系统的发展提供了有力支持。

参考文献

- [1]张轶.面向动态需求的软件模块化结构评价研究[D].重庆大学.
- [2]欧先诚,陈林凤.基于模块化的嵌入式软件设计研究[J].电子世界,2020(21):2.
- [3]周晓伟,潘清眉,陈威.基于Spring框架的模块化移动机器人控制软件模型设计与实现[J].科学大众:科技创新,2020.