

基于国产服务器的国产 Linux 操作系统适配研究

付志鹏,冯 丹,陈 红,张晓梨

(鹏城实验室,广东 深圳 518055)

摘 要:完成国产操作系统对国产服务器的适配是建设国产计算平台及计算生态的基础。针对国产 ARM 服务器的适配需求,提出国产 Linux 操作系统在国产 ARM 服务器上的适配方案,从固件适配、处理器适配以及外设适配等方面综合考虑。针对固件适配中存在直接启动和间接启动两种不同方式,分别设置不同的启动加载内容、加载路径以及启动参数。在处理器适配方面,针对 GPIO 驱动模块适配,通过设置 IO 数据传输实现流程来实现;针对 GIC 中断驱动模块适配,通过建立中断操作和中断源的映射关系来实现;针对内存地址随机化模块适配,采用基地址加上随机数来实现。在外设适配方面,通过网卡注册、初始化以及数据发送接收设置等一系列步骤来实现网卡适配;通过设置渲染缓冲模块中动态多缓冲区和缓冲区池数量来实现显卡适配;通过改进命令服务层,建立不同外部存储和相应命令之间的识别对应来实现外部存储适配;通过建立系统对 RTC 时钟读写操作流程以及增加时钟漂移系数来实现 RTC 时钟芯片 DS1339 适配。采用该方案对银河麒麟操作系统在鲲鹏服务器上的适配验证表明,该适配方案能够满足国产 Linux 操作系统在国产 ARM 服务器上的适配需求。

关键词:国产;服务器;操作系统;适配;计算平台;计算生态

中图分类号:TP311.54

文献标识码:A

文章编号:1673-629X(2023)04-0089-07

doi:10.3969/j.issn.1673-629X.2023.04.013

Research on Adaptation of Domestic Linux Operating System on Domestic ARM Server

FU Zhi-peng, FENG Dan, CHEN Hong, ZHANG Xiao-li

(Pengcheng Laboratory, Shenzhen 518055, China)

Abstract: Completing the adaptation of domestic operating system to domestic server is the basis for the development of domestic computing platform and computing ecology. For the adaptation to domestic ARM server, we present an adaptation scheme of domestic Linux operating system, comprehensively considering the adaptation of firmware, processor and external devices. As the adapted firmware includes the direct startup mode and the indirect startup mode, we present different setting methods for content loading, loading directory and startup parameters. In terms of processor adaptation, the GPIO driver module is adapted by setting an IO data transmission implementation workflow; the GIC interrupt driver module is adapted by establishing a mapping table between interrupts and interrupt sources; the memory address randomization module is adapted by combining the base address and a randomized number. For the adaptation of external devices, the network card is adapted by establishing a workflow of network card registration, initialization, and settings of sending and receiving data; the graphics card is adapted by setting the dynamic multiple buffer and the number of buffer pools in the render buffer module; the storage is adapted by improving a command service layer to establish the corresponding identification between different types of storage and their commands; the DS1339 RTC chip is adapted by setting a read/write workflow for the system time and adding a clock drifting quotient. An adaptation test of the Kylin operating system on Kunpeng server shows that this adaptation scheme can meet the adaptation requirements of domestic Linux operating system on domestic ARM server.

Key words: domestication; server; operating system; adaptation; computing platform; computing ecology

0 引言

随着国内外环境的变化,建设自主可控的国产计算生态的必要性和紧迫性日益凸显。作为计算生态核

心的处理器和操作系统,除了 Wintel (Windows+Intel) 体系和 Intel+IOS 体系,AL (ARM^[1]+Linux) 体系由于其在移动端(手机、平板等)的优势地位而成为在服务

收稿日期:2022-05-13

修回日期:2022-09-15

基金项目:广东省重点领域研发计划资助(2020B010166001)

作者简介:付志鹏(1981-),男,博士,工程师,CCF 会员(C8151M),研究方向为人工智能、云计算、物联网;通信作者:冯 丹(1991-),女,硕士,工程师,CCF 会员(F0315M),研究方向为云计算、国产计算生态。

器和 PC 端具有全球发展潜力的另一种体系。随着多年的发展,国内在 AL 体系已逐渐形成了以飞腾^[2]+麒麟^[3-6]为代表的 PK (Phytium+Kylin) 体系和以华为鲲鹏+欧拉为代表的鲲鹏体系。近几年,飞腾服务器和鲲鹏服务器在国内市场飞速增长,基于国产飞腾服务器和鲲鹏服务器的生态适配显得日益迫切,这在关键领域加速完成国产化替代奠定基础,具有广阔的市场前景。

国产计算平台的核心部件,主要包括处理器、整机、操作系统、中间件等。将这些核心部件有效集成起来,形成能够支撑当前互联网信息时代的基础设施,进而建设国产计算生态,实现对国外生态的全面替代尤为重要。生态适配首先需要解决国产操作系统对国产服务器的适配。由于 Intel 服务器长期在国际国内的垄断地位,国产操作系统主要基于 Intel 服务器开发和运行,无需进行适配。在 AL 生态蓬勃发展的当下,国产操作系统需要首先完成对飞腾服务器和华为鲲鹏服务器的适配,才能以此为基础建设国产 AL 计算平台,并在此平台上发展 AL 应用生态。

国外关于操作系统对服务器适配的相关研究较少,主要原因在于 Wintel 体系和 Intel+IOS 体系在服务器和终端市场诞生的初期就天生结合在一起,Windows 和 IOS 操作系统的开发和调试主要在 Intel 系列服务器或终端设备上运行,具有天生的适配能力。

在国内,前期国产操作系统主要以在 Intel 系列服务器上运行为主。国内外环境改变后,计算平台国产化问题日益突出,国产操作系统在国产服务器上适配需求越来越迫切。在某些特殊领域,国产操作系统对处理器的适配已有相关研究。雷鸣等^[7]针对在龙芯平台上适配 Android 操作系统进行研究,通过编译环境适配、Linux 内核适配、文件系统适配等一系列改进,测试结果表明 Android 操作系统能够适配龙芯处理器。任占广等^[8]针对国内高性能通信需求,采用锐华国产嵌入式操作系统在自主研制的高性能通信处理器“网芯一号”上进行适配,突破了操作系统在实时通信方面的不足,完成了锐华操作系统在“网芯一号”上的板级升级包和相关驱动上的适配。测试结果表明,锐华操作系统完全能够满足国内高性能通信需求,解决了通信领域大量使用国外软硬件产品,存在安全隐患的矛盾。张静等^[9]针对锐华操作系统适配到国产飞腾 1500A/4 上的问题,给出了一个针对性的处理器支持包,该包支持锐华操作系统在飞腾 1500A/4 下的任务上下文切换、中断处理、多核同步与互斥等功能,有力解决了飞腾 1500A/4 和锐华操作系统在国产通信领域的嵌入式应用障碍。针对航空航天领域软硬件国产化的特殊需求,项涛等^[10]将国产自研的天脉 1 机载嵌

入式实时操作系统在国产自研飞腾 2000A 处理器上进行适配,测试结果表明天脉 1 嵌入式操作系统能在飞腾 2000A 上顺利运行。刘帜琦等^[11]将天脉 1 嵌入式操作系统适配到 T2080 处理器上,以满足机载计算平台对实时性的需求。基于 PK 体系,郭长国等^[12]针对近几年在国产办公信息系统的研发、集成以及国产化推进、实施等方面进行总结梳理,提出了若干建设 PK 体系生态需要注意的问题及相应的应对策略,为下一步国产计算生态建设积累了经验。针对 5G 时代下的物联网对新型移动操作系统“低延时、高带宽、大连接”需求,宋佳明等^[13]提出分别从硬件适配能力、软件兼容能力以及服务生态兼容性三个方面来构建新型操作系统适配验证体系的思路。

通过这几项相关研究,可以发现如下几个特点:

(1) 针对某些关键领域的国产化替代来进行,具有很强的针对性。

(2) 操作系统对处理器的适配主要在嵌入式领域,在服务器和 PC 领域没有涉及。

(3) 适配的对象主要是处理器,而不涉及到整机、外设等。

(4) 中国逐渐意识到生态建设的重要性并加大国产计算生态的建设力度。

针对国内通用 ARM 服务器领域的国产化替代来进行,适配内容除了处理器外,还包括固件、主板、外设等。其适配思路和方法,同样可以推广到国产 ARM PC 领域,具有一定的可推广性。

操作系统的选型是国产计算平台建设的基础,是进行国产 ARM 服务器适配的第一步。目前主流的国产操作系统主要有银河麒麟系列、欧拉系列以及统信操作系统。鉴于 PK 体系和鲲鹏欧拉体系在 AL 体系中的主流地位,采用银河麒麟操作系统来进行适配。该适配思路和方法可以同样扩展到欧拉或统信等其他操作系统上。

国产操作系统对国产 ARM 服务器的适配需要经过一系列流程。由于服务器主要由固件、CPU、主板、外设等几个大模块组成,因此相应的适配过程也主要分为如下四个步骤来进行:固件适配、处理器适配、主板适配、外设适配,其中每个步骤又可以细分为多个子步骤,适配整体方案如图 1 所示。

国产固件适配主要解决服务器加载并启动操作系统的问题,国产处理器适配主要针对国产处理器内置 SOC 控制器,包括 GPIO (General - Purpose Input/Output)、GIC (Generic Interrupt Controller)、内存地址随机化部件等进行适配。外设适配主要针对网卡、显卡、外部存储和 RTC (Real-Time Clock) 等模块,需要基于整体硬件平台构建操作系统的驱动模块,实现系

统对硬件的无缝兼容。主板适配主要是指操作系统在主板上运行时可能会遇到各种各样的问题,并针对这些问题进行排查、分析和解决。随着操作系统和国产主板几十年的发展,在主板适配上遇到的问题较少,因此这里主要阐述固件适配、处理器适配和外设适配。

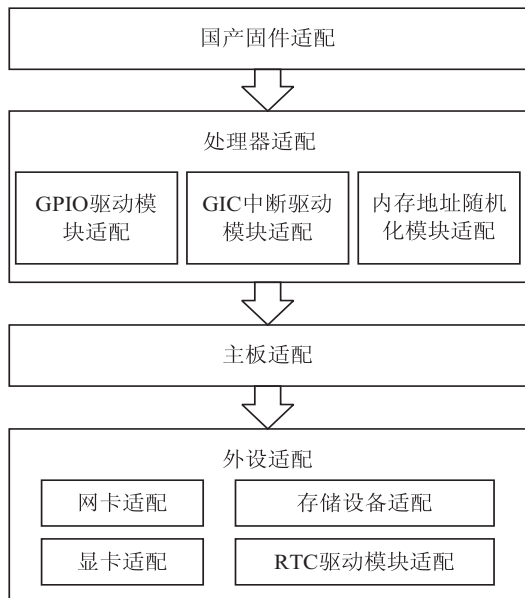


图1 国产操作系统对 ARM 服务器适配的整体方案

1 国产固件适配

固件是在启动操作系统之前首先在计算机上运行的一种软件,它的作用有两个,一个是检测并进行基本的硬件初始化,另外一个加载操作系统,然后把控制权交给操作系统。

国产固件适配是指让国产固件能够成功启动操作系统,目前主要支持两种启动方式:直接启动(固件直接加载操作系统内核并启动内核)和间接启动(固件先加载 grub 等 bootloader,然后通过 bootloader 启动操作系统),启动方式不同,其要做的工作也有所区别。固件适配要做的工作主要包括:

(1)约定好启动操作系统要加载的内容。直接启动,则要加载内核二进制文件、initrd 文件及设备树文件。间接启动,则还要加载启动加载器 grub。

(2)约定好所加载内容的加载地址,包括内核二进制文件加载地址、initrd 文件加载地址、设备树文件加载地址等。

(3)约定好固件传递给内核的启动参数。

当固件做完其硬件检查和配置工作,需要启动操作系统时,根据其启动方式不同,分别做相应的操作。当为直接启动时,使用约定的内核参数启动内核,内核在启动过程中需要解析设备树,解压并挂载 initrd 文件和文件系统进行内核和系统启动工作。当为间接启动时,通过已加载的 grub 参数加载指定的内核和设备

树,然后使用约定的内核参数启动内核,内核在启动过程中需要解析设备树,解压并挂载 initrd 文件和文件系统进行内核和系统启动工作。

2 国产处理器适配

国产处理器适配主要从 GPIO 驱动模块、GIC 中断驱动模块以及内存地址随机化模块三个方面进行。

国产处理器集成了 GPIO 模块^[14],主要用来实现 IO 数据的传输。GPIO 驱动模块适配需要根据约定的硬件采样机制,实现 GPIO 的接口操作封装,同时对系统内核通用关机和重启接口进行改造,实现系统下软关机和软重启功能。

GPIO 驱动模块适配流程分为五个过程:(1)探测设备并对片内 GPIO 控制器寄存器进行初始化;(2)获取设备树定义的 GPIO 属性配置信息进行 GPIO 设置;(3)构建 GPIO 控制器方法,包括定义 GPIO 方向输入方法、GPIO 值获取方法、GPIO 值设置方法并进行 GPIO 池的注册;(4)为了方便 GPIO 驱动和应用接口的良好交互,构建 sysfs 文件系统框架,将 GPIO 的方向输入接口、GPIO 值设置和获取接口提供给应用接口;(5)通过改造 poweroff/reboot 系统接口,按照约定的采样机制构建 GPIO 设置执行流程,实现系统下通过 GPIO 接口的软关机和软重启功能。

GIC 中断驱动模块^[15]适配,需要建立一个中断和中断源的映射关系,即一个设备发起一个中断,能够被系统及时察觉到并做相应的处理。如 PCIe 设备的 MSI 和 MSI-X 类型的中断,系统需要在内存中建立相应的结构,用于存储这种对应关系,从而确保这些中断能够及时被处理。适配完成后,所有外设触发的中断都能够在系统下正常被处理器响应和执行。系统下打开 /proc/interrupts 可以查看系统中各设备对应的中断关系,并查看这些中断的数目的变化,变化的幅度在正常范围内。

为了提供更安全的地址空间分配随机化功能(地址分配随机化增强功能),需要在 32 位系统达到 28-30 位时随机化,即 G bytes 级别;在 64 位系统要达到 ≥ 40 位时随机化,即 T bytes 级别。飞腾 FT1500A 芯片是 ARM64 架构,要满足 40 位以上的地址空间随机化,需要通过 mmap 基地址(mmap_base)映射代码段、数据段、堆段、栈段、mmap 段和匿名 mmap 段、vdso 段。内核在初始化 mmap 基地址的时候会判断当前内核是否支持地址空间随机化,若支持则会基于 mmap_base 加上经过一定处理的随机数 rnd 来保证后续每次访存得到的地址空间都随机变化,以达到提高系统安全性的目的。

适配完成的地址随机化功能,通过终端命令“cat

/proc/self/maps”,就可以查看系统下代码段、数据段、堆段、栈段、mmap 段和匿名 mmap 段、vdso 段等地址空间信息。重复输入该命令,对比可以发现同一地址空间(例如栈段)变动的位数,即当前系统实现的地址空间随机化位数。

3 外设适配

国产整机一般除了 CPU 上 SOC 集成的外设外,还会通过扩展总线(比如 PCIe 总线、USB 总线)扩展其他外设,这些外设主要包括网卡、显卡、存储设备等。

3.1 网卡适配

网卡适配主要是适对应网卡型号的网络设备驱动,并结合处理器和网卡驱动实现发送和接收环的定制操作。网卡驱动位于操作系统网络协议栈的数据链路层,上层接收网络协议中发出的数据,并传递给网卡硬件,下层从网卡接收数据,并传递给网络协议栈。

网卡适配分为四个部分,注册网卡驱动、网卡初始化、构建网卡基本操作接口和初始化发送/接收操作接口,如图 2 所示。

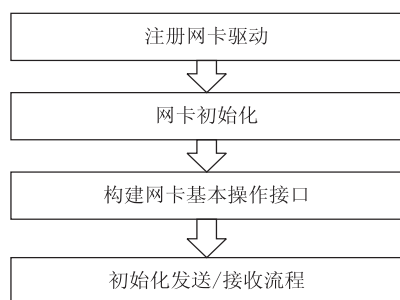


图 2 网卡适配流程

首先调用 PCIe 设备驱动注册函数注册网卡驱动,在匹配设备 ID 后,调用驱动探针函数对网卡进行初始化,包括寄存器、DMA 以及获取 EEPROM (Electrically Erasable Programmable Read Only Memory) 等信息,接着构建网卡基本操作接口,接口包括 open(命令 ifup 对应的内核接口),close(命令 ifdown 对应的内核接口)等;初始化发送/接收操作,包括发送接口函数的初始化以及 napi 的初始化(82574 网卡接收采用的是 napi 机制)以及发送接收中断等函数接口。

网卡驱动的发送和接收流程是网卡适配难点,需要结合处理器以及网卡驱动实现发送和接收环的定制操作。以接收流程为例,每个接收描述符环含有 256(size)个描述符,一个头指针 Head,一个尾指针 Tail,如图 3 所示。

驱动初始化的时候 Head 和 Tail 指针都是指向 Base,软件控制 Tail 指针,初始化一个描述符,Tail 指针往后移动一个单位,当 Tail 指针移动到 Base+Size 的时候重置为 Base,当软件发现 Tail 等于 Head 的时

候,表示当前已经没有数据可供处理,需要等待网卡接收新的数据报文。硬件控制 Head 指针,网卡设备接收到数据包的时候,硬件移动 Head 指针,当 Head 指针移动到 Base+Size 的时候重置为 Base,当网卡设备发现 Head 等于 Tail 的时候,表示当前环的所有描述符都已经被填充,说明网卡接收速度快于驱动处理数据的速度,需要等待驱动处理数据并提供新的描述符。适配需要结合处理器以及网卡驱动实现发送和接收环的定制操作。

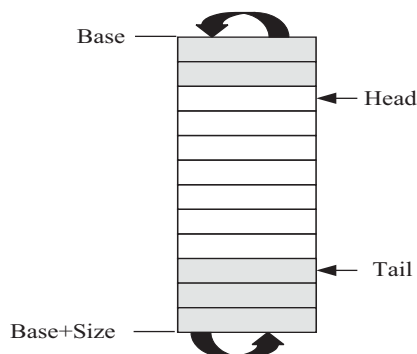


图 3 接受描述符环软件结构

3.2 显卡适配

显卡是显示设备,为计算机提供图形渲染与图形显示的功能。显卡一般由 GPU、显示控制器和显示接口等几部分组成。目前主流的显卡主要有两家厂商,ATI 的 A 系列显卡和英伟达的 N 系列显卡,而国产化的显卡在图形加速处理能力以及产品成熟度上跟主流商用显卡存在一定差距,急需解决国产化显卡在图形加速处理方面不足的问题。

早期的 3D 加速处理,渲染缓冲模块由一个屏幕前缓冲和一个屏幕后缓冲组成,所有的客户端和服务端必须共享这个单一的缓冲区,应用程序将直接渲染到这个屏幕后缓冲里,然后通过等待垂直同步信号的到来再交换到屏幕前缓冲中,这种机制下在下一轮渲染时需要等待当前渲染缓冲的释放,会带来阻塞开销,同时往返等待服务器发送渲染缓冲区的过程也会带来性能损失。

随着单位帧内渲染场景的复杂度增加,对渲染缓冲区的需求加大,现阶段 3D 处理提出了一种称为动态多缓冲区的机制,这种机制使得每一个客户端都使用自己私有的后缓冲,而不是使用单个共享的后缓冲。客户端通过这个私有的后缓冲关联自己的深度缓冲、模板缓冲、颜色缓冲,最后将渲染内容拷贝到屏幕。

该机制重新定义了一个与当前应用客户端关联的缓冲区池,当前应用需要缓冲区进行渲染时就会在该缓冲区池中获取一个新的缓存,通过轮询该缓冲区池中的每一个缓存,找到一个空闲的缓冲区。

动态多缓冲区机制的实现由原来一个缓冲增加到

现在的一个缓冲池。一个渲染客户端可以拥有一个或多个缓冲区,这些缓冲区的生命周期就是当前渲染上下文的生命周期。一旦当前渲染上下文销毁后,该缓冲池中的每一个缓存将被释放。

合理分配缓冲区池中缓存的数量,尽量避免3D引擎主动向服务器端轮询消息事件,有效控制往返等待消息事件的时间,使得渲染过程不会因为申请缓存的时间开销而增大渲染周期,给图形渲染带来性能损失。经过不同场景下对图形性能的测试发现,缓冲池中的缓存数量保持在2~3个为最佳。

3.3 存储设备适配

存储设备是指存储控制器,为计算机提供访问外存的控制设备。存储包括许多不同的种类,SATA (Serial Advanced Technology Attachment) 控制器用于扩展 SATA 磁盘,RAID (Redundant Array of Independent Disk) 控制器用来搭建 RAID,光纤控制器用来扩展 iSCSI (Internet Small Computer System Interface) 磁盘阵列等。存储设备驱动位于 IO 栈的底层,其上层是通用块设备层,下层就是具体的存储设备。存储设备驱动接收来自通用块设备层的 IO 请求,根据 IO 请求的内容和方向,发送数据给磁盘或从磁盘读取数据传递给上层。

SCSI 实现了一种客户机/服务器风格的通信架构。发起者向目标设备发送命令请求,该目标处理此请求并向发起者返回响应。发起者可以是托管计算机中的一个 SCSI 设备,而 SCSI 目标则可以是一个磁盘、光盘和磁带设备或特殊设备(比如箱体设备)。

目前,国产平台选用的 SATA 控制器对命令的支持程度是有局限性的,有一些 SCSI 命令无法被硬件所识别并响应,特别是针对光驱的一些命令,如读取光盘信息等,这就导致了刻录光盘时无法得到空光盘的信息,进而无法刻录等一系列问题。因此,存储设备适配重点解决 SATA 控制器对 SCSI 命令无法被硬件识别的问题。为此,首先找到无法被识别的 SCSI 命令,并根据 SCSI 命令手册找到这些命令的作用。然后,把无法识别的命令在驱动命令服务层中进行截流改造,如图4所示,让硬件可以正常返回。

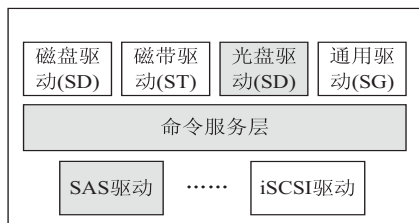


图4 系统命令服务逻辑框架

3.4 RTC 驱动模块适配

国产平台主要采用 DS1339 型号的 RTC 芯片^[16],

挂载在 I2C0 总线下,其作用是为用户提供可靠的时间,当系统处于关机的情况下,RTC 记录操作系统时间,并可在电池供电情况下继续正常工作,当系统正常启动后,系统可从 RTC 读取时间信息,来确保关机到开机后时间运行连续性。

为了实现系统对 RTC 时间的读写操作,需要构建基于 DS1339 芯片的 RTC 驱动模型,并且实现对 RTC 接口的封装以及文件系统下设备节点的创建。用户可以通过操作系统中的设备节点来实现读写 RTC 时间,具体驱动设计框架如图5所示。

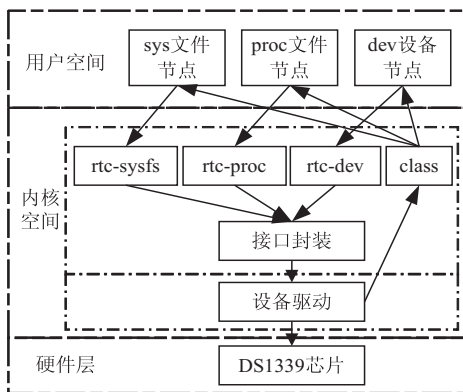


图5 RTC 驱动设计框架

主要工作针对图5中的内核空间部分展开。首先,进行设备驱动的注册和初始化工作,包括根据数据手册对 DS1339 芯片寄存器的初始化;其次,创建设备文件节点,通过 class 封装函数分别在文件系统下创建 sys 文件节点,proc 文件节点,dev 设备节点;最后,针对设备文件节点的操作函数接口封装,让用户通过操作相应的设备文件节点就可以完成对 RTC 的读写。

由于每次将时间写入 RTC 芯片的过程中会产生一个时间延迟漂移,根据各芯片特性不同漂移系数会有差别,这是一个隐性问题,比较难以发现。在操作系统的关机过程中,会有一次将系统时间写入 RTC 的操作,在系统开机加电时再将 RTC 时间读出,写入系统时间。故而在重启一定次数后,系统时间将会产生比较大的误差,原因是漂移延迟进行了累加。解决办法是在写入 RTC 过程中加上该芯片的漂移系数。

适配完成的 RTC 驱动,可以通过 date 命令设置当前系统时间,通过命令 hwclock -w 将系统时间写入 RTC 芯片,通过命令 hwclock -r 读取当前 RTC 时间,通过命令 hwclock -s 将 RTC 时间覆盖系统时间,观察是否正确。

4 测试验证

为了对上述适配方案进行验证,采用银河麒麟操作系统在鲲鹏服务器上进行,其测试验证平台的软硬件配置如表1所示。

表 1 测试验证环境配置

服务器配置	操作系统	软件配置
TaiShan200-2280v2;鲲鹏 920-4826 CPU:48 核心/48 线程,2600MHZ * 2, 内存:海力士 DDR4,32 GB * 16;硬盘:三星 MZ7LH480HAHQ-00005,SSD 450 GB * 1	kylin-374.1 9.90-23.6. v2101.ky10.aarch64	unixbench5.1.3 stream5.9.1 iozone3.43

测试验证的内容主要针对麒麟操作系统通过上面的适配方案在鲲鹏服务器上适配后,其固件、处理器、外设等是否适配成功,即相关的固件、处理器、外设是否能够正常使用。

测试方法主要是输入与适配部件相关的命令,如果能够正常查看并使用相关部件,可以认为适配成功并能正常使用,相关测试内容及结果如表 2 所示。

表 2 适配测试及结果

测试项	测试目标	测试步骤	测试结果
固 件 适 配 测试	测试固件是否能正常引导系统关闭	打开终端,输入 poweroff	系统顺利关闭
	测试固件是否能正常引导系统重启	打开终端,输入 reboot	系统顺利重启
	测试是否能正常查看处理器信息	打开终端,输入 lscpu	正确显示 CPU 信息
处 理 器 适 配测试	测试是否能正常查看处理器中树状进程信息	打开终端,输入 pstree	正确显示树状进程信息
	测试是否能实时显示处理器的资源占用情况	打开终端,输入 top	正确显示资源占用情况
	测试是否能正常查看处理器中运行的进程信息	打开终端,输入 ps -ef	正确显示进程信息
中 断 模 块 适配测试	测试中断模块是否是否适配成功并可查看系统中断信息	打开终端,输入 cat /proc/interrupts	正确显示中断信息
内 存 模 块 适配测试	测试内存模块是否适配成功并可查看系统内存信息	打开终端,输入 cat /proc/meminfo,	内存运行正常,信息正确显示
	测试内存模块是否适配成功并可查看内存地址随机化信息	打开终端,输入 cat /proc/self/maps,	正确显示内存中各类型地址空间信息
外 设 适 配 测试	测试外设适配是否成功并查看成功适配的外部硬件信息	打开终端,输入 dmidecode-q	正确显示所有硬件信息
网 卡 适 配 测试	测试网卡是否适配成功并可查看网卡信息	打开终端,输入 ifconfig	正确显示网卡信息
	测试网卡是否适配成功并查看网络状态信息	打开终端,输入 netstat	正确显示网络状态信息
	测试网卡是否正常工作并测试网络连通性	打开终端,输入 ping www.baidu.com	正确显示百度连通信息
	测试网卡是否正常工作并查看路由表信息	打开终端,输入 route	正确显示路由表信息
显 卡 适 配 测试	测试显卡是否适配成功并查看显卡信息	1. 设备识别:lspci -v grep VGA 检查操作系统能否正确检测到显卡设备硬件信息	正确显示显卡运行正常,信息正确
		2. 驱动对应性:lspci -v grep VGA -A14 查看 VGA 部分在用的驱动	
存 储 设 备 适配测试	测试外部存储设备是否适配成功并可查看存储磁盘相关信息	打开终端,输入 df	正确显示磁盘信息
	测试存储设备是否正常工作并查看已加载的文件系统	打开终端,输入 cat /proc/mounts	正确显示文件系统信息
	测试存储设备是否正常工作并查看已挂载的分区列表	打开终端,输入 df -h	正确显示挂载的分区信息
	测试存储设备是否正常工作并查看某目录已经使用的磁盘空间	打开终端,输入 du -sh /home	正确显示 home 目录下已用磁盘信息
	测试存储设备是否正常工作并备份磁盘内容到一个文件	打开终端,输入 dd if=/dev/sdc1 of=/tmp/file1	正确显示备份成功
	测试存储设备是否正常工作并创建文件夹	打开终端,输入 mkdir abc	正确显示创建文件夹成功
RTC 驱动模块适配测试	测试 RTC 驱动模块是否适配成功并正常工作,查看操作系统时间	打开终端,输入 date	显示系统时间

同时,通过表 3 所示的 48 小时稳定性测试,来检验适配后的服务器是否能稳定运行。

表 3 稳定性测试结果

测试项	测试目标	测试步骤	测试结果
整机 48 小时稳定性	测试鲲鹏服务器适配成功后,能否开机后正常运行 48 小时无宕机	1. 解压 Performance_v1.23_20201221-desktop 压缩包 2. 进入解压目录执行: bash ltp.sh sstress 48 3. top 命令监控系统资源占用,48 小时之后系统资源是否正常释放	48 小时无宕机,资源正常释放

测试验证结果表明,银河麒麟服务器操作系统 V10 在 TaiShan200-2280v2 服务器上适配成功,能满足正常使用需求。

5 结束语

建设国产计算平台是信创建设的基础,是实现国家关键领域国产化替代的关键环节,它首先需要解决国产操作系统在国产服务器上的适配。针对银河麒麟在国产 ARM 服务器上的适配问题,该文从固件、处理器、网卡、显卡、外部存储、时钟等方面详细阐述,测试验证结果表明,该适配方案能够很好地满足国产操作系统对国产 ARM 服务器的适配。

下一步将基于国产计算平台建设基于 AL 平台的国产计算生态,重点在数据库、中间件、关键行业应用软件等方面进行迁移适配。针对国产昇腾 AI 芯片,下一步也将做相应的适配,满足国内日益增长的人工智能计算需求。通过一系列的数据库、中间件、应用软件适配,希望丰富壮大国产计算生态,有力解决当前关键技术受制于人的被动局面。

参考文献:

[1] SEAL D. ARM architecture reference manual[M]. 2nd ed. [s. l.]:Addison-Wesley Professional,2000.

[2] HALFHILL T R. Phytium samples 64-Core ARMv8[J]. Microprocessor Report,2016,30(9):9-11.

[3] 麒麟软件有限公司. 银河麒麟高级服务器操作系统 V10 产品白皮书[M/OL]. 2020. <https://www.kylinos.cn/support/document/54.html>.

[4] 李震宁. 银河麒麟操作系统开源生态实践[J]. 软件和集成电路,2021(6):40-41.

[5] PEI S,LI Y,ZHOU J,et al. Design of time unification software based on Kylin operating system[C]//Proceedings of the 2017 2nd international conference on image, vision and computing (ICIVC). Chengdu:[s. n.],2017:1040-1043.

[6] GAO Y,YU X. Design and implementation of trusted plug-in based on Kylin operating system platform[J]. Journal of Physics Conference Series,2020,1544(1):012042.

[7] 雷鸣,靳婷,徐晓亮,等. 面向龙芯平台的 Android 系统移植研究[J]. 计算机工程与应用,2012,48(22):70-73.

[8] 任占广,张学敏. 基于网芯一号处理器的锐华操作系统适配[J]. 计算机与网络,2020,46(22):58-61.

[9] 张静,李娟. 基于飞腾 1500A/4 的嵌入式操作系统适配技术[J]. 电子技术与软件工程,2018(7):200-201.

[10] 项涛,吕大鹏,董延军. FT2000A 处理器嵌入式操作系统适配技术研究[J]. 信息通信,2019(3):110-112.

[11] 刘帆琦,万欣宇,李智慧. 基于 T2080 处理器嵌入式操作系统适配研究[J]. 长江信息通信,2022,35(1):92-94.

[12] 郭长国,谢劲松,韩鹏,等. 国产办公信息系统若干关键问题研究[J]. 信息技术与网络安全,2018,37(9):17-21.

[13] 宋佳明,马霁阳,曾晨曦. 5G 时代新型移动操作系统适配验证体系研究[J]. 质量与认证,2021(4):66-68.

[14] 刘进,肖克亮,邓睿. 基于 ARM9 的 Linux 嵌入式字符驱动 GPIO 设计[J]. 电子设计工程,2015(15):115-117.

[15] 陆俊. 基于 ARMv8 架构的中断控制器设计[D]. 南京:东南大学,2015.

[16] 朱英杰. 串行时钟芯片 DS1339 在 RTU 中的应用[J]. 仪器仪表用户,2003,10(5):39-40.