

# 基于多核处理器的非对称嵌入式系统研究综述

瞿 伟 余飞鸿

浙江大学光电科学与工程学院 杭州 310027

(quwei@zju.edu.cn)

**摘 要** 随着嵌入式系统的发展与不断分化,很多领域如工业控制、机器人、视频图像系统等对嵌入式系统的要求越来越高,这不仅需要良好的功能扩展性和维护性,还需要保证专有任务的特性(如实时性等)。基于多核处理器的非对称嵌入式系统是解决这些问题的重要发展方向。根据处理器各核心地位是否相同,多核处理器可以划分为同构和异构两种结构。基于同构或者异构多核处理器均可以实现非对称嵌入式系统,从硬件或者软件层面将多核处理器的核心进行划分,分别运行不同的任务,使得嵌入式系统可以兼顾良好的功能扩展性和实时性。文中总结并对比了基于多核处理器的非对称嵌入式系统的研究现状,同时归纳了其在科研和工程领域的应用,最后研究了基于多核处理器的非对称嵌入式系统未来可能的发展方向。

**关键词**:非对称多处理;多核处理器;双操作系统;嵌入式系统

**中图法分类号** TN316

## Survey of Research on Asymmetric Embedded System Based on Multi-core Processor

QU Wei and YU Fei-hong

College of Optical Science and Engineering, Zhejiang University, Hangzhou 310027, China

**Abstract** With the development and continuous differentiation of embedded systems, many fields such as industrial control, robotics, video and image systems, etc. have higher and higher requirements for embedded systems, which not only require good functional scalability and maintainability, but also need to ensure the real-time performance. Asymmetric embedded system based on multi-core processors is an important development direction to solve these problems. According to whether there is a primary and secondary distinction between processor cores, multi-core processors can be divided into two structures: homogeneous and heterogeneous. Asymmetric embedded systems can be realized based on both homogeneous and heterogeneous multi-core processors. The cores of multi-core processors are divided from the hardware or software level to run different tasks, so that the embedded system can balance good functional scalability and real-time performance. This paper summarizes and compares the research status of asymmetric embedded systems based on multi-core processors, and summarizes its applications in the fields of scientific research and engineering, and finally this paper studies the possible future development directions of asymmetric embedded systems based on multi-core processors.

**Keywords** Asymmetric multi process, Multi-core processor, Dual operating system, Embedded system

### 1 引言

嵌入式系统是计算机硬件和软件(可能还有机械或电子的其他部件)的组合,旨在执行专用功能<sup>[1]</sup>。嵌入式系统出现于20世纪70年代,起初应用于工业控制和国防系统等领域,对实时性和安全性有较高要求<sup>[2]</sup>。如今,随着计算机技术、集成电路技术等相关技术的发展,嵌入式系统的应用场景也得到了极大的拓展。目前嵌入式系统已经深入应用到各个行业,包括军事、通信、工业控制、手持移动设备、视频图像系统等。很多领域要求嵌入式系统不仅拥有实时性,同时也需要有良好的功能扩展性和可维护性,这对嵌入式系统的结构设计带来了新的挑战。传统的对称结构嵌入式系统已经无法满足这些要求,非对称结构的嵌入式系统成为了新的研究方向。

从多核处理器各个核心是否运行相同的任务出发,嵌入式系统可以分为对称结构和非对称结构。对称结构指多核处理器各核心地位相等,运行相同或者类似的任务;非对称结构

指多核处理器使用不同核心完成不同的任务<sup>[3-4]</sup>。

对于多核处理器而言,按照多核处理器各个核心架构是否相同,可以将多核处理器分为同构和异构两种类型。对于同构多核处理器,通常使用双操作系统的方式搭建非对称嵌入式系统,而对于异构多核处理器,各个核心在设计时即按照各自任务的特性进行设计,因此采用异构多核处理器即直接构成非对称结构。非对称结构的嵌入式系统的要点在于使用不同的核心完成不同的任务,发挥不同核心或不同操作系统的优势。

本文总结并对比了基于多核处理器的非对称嵌入式系统的研究现状,同时归纳了非对称嵌入式系统在科研和工程领域的应用。

### 2 多核处理器技术

信息社会时代对信息的处理速度要求逐渐提高,随着计算机硬件技术和软件技术的迅速发展,如何提升处理器的处

理能力成为了重要的研究方向<sup>[5]</sup>。

提升处理器性能的方法有3种。1)提高处理器主频,半导体行业一直坚持使用先进制程对芯片进行小型化并提升主频。但是,随着芯片晶体管数目越来越多,芯片电路线宽越来越小,芯片的功耗和散热成为了难以解决的问题。2)改造处理器结构,使用流水线、超标量、SIMD(Single Instruction Multiple Data)、超线程(Simultaneous Multithreading, SMT)等技术,让处理器在一个周期内执行多条指令,如Intel的Pentium处理器<sup>[6]</sup>。3)在单个处理器芯片上集成多个CPU核心,可以有效提升处理性能,并且带来更好的能效比<sup>[7]</sup>。多核处理器能在获得较强的并行处理性能的同时无需提升处理器主频<sup>[8]</sup>。

多核处理器也称为片上多处理器(Chip MultiProcessors, CMP)。1996年,斯坦福大学计算机系统实验室首次提出片上多处理器的思想,并首次设计了一种集成了4个MIPS R3000核心的处理器Hydra<sup>[9-10]</sup>。Hydra采用了共享二级缓存的同构对称设计和高速总线的核间通信方式,是一种典型的多核结构。在此之后,IBM,AMD以及Intel等公司也相继推出了双核、四核以及更多核心的处理器<sup>[11]</sup>。

目前,多核处理器的核心结构主要有同构和异构两种。同构结构采用对称设计,所有的核心均采用同一架构设计,原理简单,容易实现,但是存在数据一致性、核心存储以及I/O访问的性能限制、处理器功耗、多核间任务协调等问题。目前主流的桌面端多核处理器均采用同构结构。

异构结构则将不同架构的核心组织在一起,实现处理器性能的最佳化,有效降低了功耗。但是,多核任务分工、结构扩展性与数量限制、处理器指令系统设计等也成为了异构结构的实现难点<sup>[4,9,12-14]</sup>。异构结构的代表处理器有IBM的Cell处理器以及Xilinx的ZYNQ系列处理器。

相比单核处理器,多核处理器的优势有4点<sup>[15]</sup>:1)在单个时钟周期内可以执行的任务单元数更多;2)相对功耗低;3)通信延迟低;4)验证周期和设计时间较短。

多核处理器为很多应用领域提供了全新的解决方案,目前多核处理器已经成为提升处理器性能的主要途径,是非对称嵌入式系统的重要组成部分。

### 3 对称与非对称嵌入式系统

#### 3.1 对称嵌入式系统

按照嵌入式处理器的核心功能性,将嵌入式系统分为对称和非对称两种。对称嵌入式系统指嵌入式系统的处理器的各核心地位相同,运行相同或者类似的任务;非对称嵌入式系统的处理器各核心运行不同的任务。对称嵌入式系统的结构示意图如图1所示。

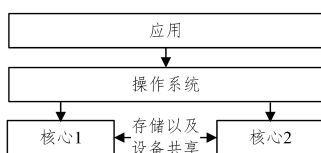


图1 对称嵌入式系统

Fig.1 Symmetric embedded system

对称嵌入式系统中,各个处理器核心对称地集成在系统总线上,内存中只存在一个操作系统镜像,每个处理器核心上

都运行同一操作系统副本,它们之间共享物理内存和外部设备,称为UMA(Uniform Memory Access)<sup>[16-17]</sup>。由于所有处理器都共享代码和数据,因此同步是对称嵌入式系统的关键问题。早期的对称嵌入式系统通过粗粒度锁定来简化同步问题,但是目前几乎所有的对称嵌入式系统都基于细粒度锁定来实现最大并行度<sup>[18-21]</sup>。

#### 3.2 非对称嵌入式系统

非对称嵌入式系统中,无论使用的多核处理器为同构还是异构,只要每个核心运行自己的特殊任务,则称其为非对称结构。在非对称系统中,每个核心都有自己的独立内存空间,多核通过共享内存、中断的方式进行核间通信(Inter Processor Communication Message, IPCM)。非对称嵌入式系统可以在满足特定应用程序性能要求的同时降低开发成本和资源消耗<sup>[22]</sup>,如图2所示。

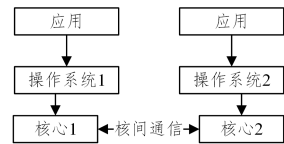


图2 非对称嵌入式系统

Fig.2 Asymmetric embedded system

对于非对称嵌入式系统而言,可以从软件和硬件层面实现。在软件层面,在每个处理器核心上运行独立的操作系统或者裸机程序;在硬件层面,通过使用异构多核处理器,让不同核心执行不同任务,也可以实现非对称嵌入式系统结构。

##### 3.2.1 使用双操作系统的非对称嵌入式系统

操作系统是嵌入式系统中的重要组成部分,它将嵌入式系统的软件和硬件连接,是嵌入式系统功能实现的重要基础。操作系统可以分为两类:通用操作系统(General-Purpose Operating System, GPOS)和实时操作系统(Real-Time Operating System, RTOS)。通用操作系统通常拥有良好的功能扩展性,主要的任务调度目标为在单位时间内进行尽可能多的计算。实时操作系统将时间作为关键参数,需要保证任务的可预测性、正确性和时效性。

在一些嵌入式应用场景中,在需要实时性的情况下也需要优良的功能扩展性,传统的单操作系统结构嵌入式系统无法满足这些需求,采用双操作系统结构成为了一种解决方案。双操作系统结构可以将GPOS和RTOS的优势结合,GPOS为嵌入式系统提供人机交互接口等非实时性功能,RTOS则用于保证系统的实时性<sup>[2]</sup>。

##### 3.2.2 使用异构多核处理器的非对称嵌入式系统

一般来说,使用异构多核处理器的嵌入式系统均构成非对称结构。对于异构多核处理器而言,通常将运行操作系统的核心称为通用处理核心或者主核心。而在通用处理核心之外,异构多核处理器一般会针对特定任务集成一些辅助核心,如DSP、FPGA、多媒体处理器等。这些特殊核心往往只能运行特殊设计的指令集,一般都需要通用处理核心进行管理和控制<sup>[8]</sup>。

### 4 同构多核处理器

对于同构多核处理器,通常使用双操作系统架构来实现非对称嵌入式系统。使用双操作系统实现非对称嵌入式系统的方法可以分为两种,第一种基于虚拟化技术,使用虚拟机管

理程序对双操作系统进行管理,第二种则让多核处理器的每个核心运行单独的操作系统<sup>[22]</sup>。

#### 4.1 虚拟化

虚拟化技术是在单核处理器上实现双操作系统架构的重要手段,指通过软件层面对系统的硬件资源进行虚拟化,引入虚拟机监控程序(hypervisor)或虚拟机监视器(Virtual Machine Monitor, VMM)对双操作系统进行管理,由于引入了虚拟机,因此这种结构可以实现硬件资源的复用,但是,同时也增加了任务调用开销,影响系统的实时性和性能<sup>[2]</sup>。

在多核处理器上,通过使用虚拟化技术也可以运行双操作系统。虚拟化提供了一种对多处理器系统进行分区的方法,虚拟化技术一般使用逻辑分区对硬件的负载进行平衡,但是各操作系统间的隔离不会在硬件级别进行,降低了多核处理器的优势<sup>[23]</sup>。对于多核处理器而言,各核心的任务无法做出区分,本质上依然是对称结构。

但是,基于一些特殊的虚拟化技术,也可以实现非对称结构。美国亚特兰大学提出的 DynOS SPUMONE 结构,使用在多核处理器内核上动态分配操作系统的方式,实现了在多核 CPU 上运行多个操作系统<sup>[24]</sup>。此设计可以动态控制处理器核心上运行的操作系统。初始状态下, GPOS 占用所有核心,检测到应用程序需要在 RTOS 上运行后,虚拟机指定一个核心运行 RTOS,并将正在运行的应用程序从此核心迁移到其他核心。迁移完成后,该指定核心上的 GPOS 相关数据被删除,并启动了 RTOS 及其应用程序。当 RTOS 端程序处理完成后,RTOS 关闭自身并重新调用 GPOS 占用核心。

#### 4.2 每个核心独立运行操作系统

北京航空航天大学开发出基于多核处理器架构的双操作系统框架 RGMP(RTOS and GPOS Multi-Processor),使用 Linux 作为 GPOS,RTOS 则允许使用多种不同类型的 RTOS<sup>[25]</sup>。RGMP 利用同构多核处理器的 I/O 高级可编程中断控制器(I/O APIC)组件,不同设备产生的终端信号被分类分配到相应核心,由运行在该核心上的操作系统进行处理。如图 3 所示。

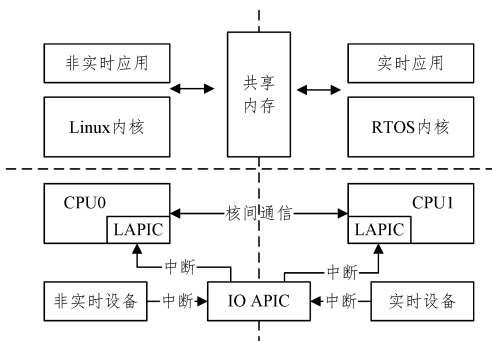


图 3 RGMP 结构

Fig. 3 Structure of RGMP

在此设计中,将设备分为实时和非实时设备,非实时设备由 Linux 控制,实时设备由 RTOS 控制。两种设备的主要区别在于其 I/O 中断发送到哪个操作系统。两个操作系统之间通过共享内存来实现双向消息先进先出(First In First Out, FIFO)。RGMP 使用两种方式实现了操作系统间的任务同步:互斥机制和同步机制。在互斥机制中使用自旋锁对共享变量上锁达到同步的目的。在同步机制中,假设任务 A 和 B 是在不同 OS 上运行的任务,两者构成了典型的生产者使用

者问题,即任务 A 写入 FIFO,任务 B 读取 FIFO。问题在于 A 和 B 将无法调用对方 OS 上的 sleep() 和 wakeup() 函数。解决方案是使用核间中断(Inter-Processor Interrupt, IPI)。以 wakeup() 函数为例,将此函数分为两个部分,任务 B 调用其 OS 上的 wakeup() 部分并将 IPI 发送给对方 OS,任务 A 则作为接受到终端后的处理程序被调用,这样的机制给操作系统代码带来了良好的隔离性。

西安电子科技大学利用 Zynq-7000 双核处理器设计了 Linux+ $\mu$ C/OS-II 的双操作系统架构,用于运动控制领域<sup>[26]</sup>。RTOS 采用了静态优先级调度算法,每个任务都拥有唯一的整型优先级数,数值越大,优先级越低,但是这种方法不够灵活,无法应对更复杂的实时系统的实时性需求。

Muir 等提出 AsyMOS(Asymmetric Multiprocessor Operating System)结构,采用了同构多核处理器,为不同的处理器分配了不同的功能<sup>[27]</sup>。它将处理器分为两个功能组,即设备处理器和应用程序处理器,每个设备处理器对应特定设备,如网卡、磁盘控制器等。应用程序处理器则运行本机内核。应用程序处理器上运行 Linux 内核,设备处理器上运行的则是 LDK(Lightweight Device Kernel)。LDK 只有两个功能:处理特定设备事件和与本机 OS 通信,因此它无需文件系统、终端、调度或流程管理功能,只运行单个任务,消除了切换内核时产生的开销。同时,LDK 以最高优先级运行,大大减少了在一些处理器架构上(如英特尔奔腾)的中断处理程序的调用。

## 5 异构多核处理器

如前文所述,异构多核处理器采用通用处理核心+辅助核心的结构,著名的异构多核处理器有 IBM 的 Cell、Xilinx 的 ZYNQ-7000、TI 的 OMAP 等<sup>[4]</sup>。在硬件层面上,异构多核处理器各个核心的用途已经确定,因此采用异构多核处理器的嵌入式系统均为非对称结构。

IBM Cell 处理器<sup>[28]</sup>是一个典型的异构多核处理器,采用一个 Power 架构的 64 位核心作为通用处理核心,8 个 32 位 SIMD 型核心作为辅助处理核心。通用处理核心负责进行任务调度,8 个辅助处理器对浮点处理进行特殊优化,用于高密度计算。

ZYNQ-7000 系列处理器采用双核 Cortex-A9+FPGA 的三核设计。两颗 ARM 核心作为通用处理核心,FPGA 作为辅助核心。ARM 核心与 FPGA 核心可以独立运行也可以协同运行。ARM 提供了丰富的外设匹配和灵活的开发方式,FPGA 的优势则在于配置灵活。ZYNA-7000 已经应用于很多嵌入式领域,包括小型蜂窝基站、多摄像头驾驶员辅助系统、工业自动化机器视觉、医疗内窥镜和 4K2K 超高清电视<sup>[29]</sup>。

OMAP 系列处理器采用 ARM+DSP 的结构,以 OMAP137 为例,此处理器集成了一个 ARM9 核心和一个 C674x 系列 DSP,DSP 核心用于数据运算,而 ARM 核心负责通用控制<sup>[30]</sup>。

这些异构多核处理器多针对专业任务设计,通用处理核心上一般运行有操作系统,辅助核心上可以运行操作系统或者裸机程序,通过各个核心之间的搭配来达到能效最大化。

## 6 多核处理器的非对称嵌入式系统应用

基于多核处理器的非对称嵌入式系统在工业控制以及视频图像处理等领域均有广泛的应用,下文对其应用进行举例说明。

### 6.1 工业控制

非对称嵌入式系统在工业控制的重要应用之一是机器人。Sun 等设计了一款基于 ZYNQ-7000 双核处理器的 AMP 架构机器人控制器,其采用 Linux 和 FreeRTOS 系统<sup>[31]</sup>。RTOS 用于运动控制和紧急操作,GPOS 用于人机交互、机器人路径规划和其他任务。Chen 等同样基于 ZYNQ-7000,搭建了一个基于 Linux 和 Bare-Metal 双系统的 AMP 架构运动控制器<sup>[32]</sup>。Dong 等利用 RGMP 架构,在双核处理器上采用了 Linux+NuttX 架构实现机器人控制<sup>[33]</sup>。

### 6.2 视频图像编解码系统

视频图像编解码是嵌入式系统的热门应用领域之一。Peng Li 等用 OMAP5910 处理器搭建了一个 AMP 架构多媒体系统<sup>[33]</sup>,ARM 核心运行 Linux 系统,DSP 核心运行 DSP 任务。两个核心采用 3 种方式进行交互:1)允许 ARM 端通过微处理器接口(Microprocessor Unit Interface,MPUI)端口访问 DSP 的所有内存空间;2)ARM 端和 DSP 端的内存管理单元(Memory Management Unit,MMU)会在外部内存映射同一段虚拟地址,双方通过此共享内存进行交互;3)ARM 核心和 DSP 核心之间通过邮箱中断的方式进行交互。当用户应用程序希望 DSP 进行一些计算时,ARM 端的 DSP 驱动程序将控制代码和应用程序数据地址写入邮箱,以告诉 DSP 用户应用程序的需求,然后 DSP 会在邮箱获取控制代码和应用程序数据地址,产生邮箱中断。DSP 完成数据处理后,通过写入邮箱对 ARM 端做出响应,之后 ARM 端的 DSP 驱动程序处理邮箱中断,并将计算结果提供给用户应用程序。Tseng 等利用 TI 的 DM6446 处理器搭建了一个 AMP 架构的 H.264 编码系统<sup>[34]</sup>,采用了类似的结构,他们使用了一个 DSP 管理器管理共享内存并进行动态任务分配。Erdos 等<sup>[35]</sup>基于 Xilinx 的 Xtensa LX 处理器实现了一个非对称 JPEG 加速系统。

**结束语** 本文研究了基于多核处理器的非对称嵌入式系统发展现状,总结了各种非对称嵌入式系统的实现原理。从多核处理器各个核心是否运行不同的任务出发,嵌入式系统可以分为对称结构和非对称结构。对称结构中各核心地位相等,运行相同或者类似的任务,而非对称结构中不同核心完成不同的任务。非对称嵌入式系统可以发挥各个核心或运行于核心上的操作系统的优势,在满足任务的特殊要求的同时拥有良好的功能扩展性。基于同构多核处理器和异构多核处理器均可以实现非对称结构。异构多核处理器为特殊任务设计了辅助核心,采用异构多核处理器的嵌入式系统均构成非对称结构。同构多核处理器则可以采用 GPOS+RTOS 双操作系统的方式构建非对称系统。

目前,非对称嵌入式系统已经应用于工业控制、视频图像处理等多个领域。而嵌入式系统的发展方向会随着社会的发展而不断变化,下文对嵌入式系统的发展趋势进行了展望:

嵌入式系统也会朝着智能化方向发展,目前已经有很多嵌入式处理器集成了卷积神经网络等智能化功能,智能化在今后会是嵌入式系统的重要研究方向。

(2) 互联化。随着 5G 网络等低延时网络技术的发展,未来的嵌入式系统将不再是一个个的孤立个体,各个嵌入式系统相互配合,构建万物互联的物联网,嵌入式系统的互联化也正成为热门的研究方向。

## 参考文献

- [1] MICHAEL BARR A M. Programming Embedded Systems: With C and GNU Development Tools[M]//Sebastopol:O'Reilly, 2006:15.
- [2] ZHANG M Y,ZHANG Q Y,MENG Z Q. A Survey of Research on Real-Time Dual-OS Architecture for Embedded Platform[J]. Acta Electronica Sinica, 2018, 46(11): 2787-2796.
- [3] KONG X Z. Research and Implementation of Embedded Real-time Operating System for SMP Architecture DSP[D]. Xi'an: Xidian University, 2013.
- [4] SUN C. Research on Multi-core Communication System Based on AMP Architecture[D]. Baoding: North China Electric Power University, 2019.
- [5] LIU J G, LIANG M G. The Development and the Software System Architecture of Multi-core Multi-threading Processor[J]. Microprocessors, 2007(1): 1-3, 7.
- [6] ZHENG F. Superscalar and Superpipeline Hybrid Microprocessor Pentium[J]. Microprocessor, 1994(4): 1-5.
- [7] XU L L. Research and Implementation of Embedded Real-time Operating System Supporting Symmetric Multi-core Processor[D]. Beijing: China Academy of Launch Vehicle Technology, 2016.
- [8] HUANG Z B. Cooperative Scheduling Method with Hardware and Software for Heterogeneous Multi-core in Embedded System[D]. Hangzhou: Zhejiang University, 2007.
- [9] HUANG G R,ZHANG P,WEI G B. Key Techniques of Multi-core Processor and Its Development Trends[J]. Computer Engineering and Design, 2009, 30(10): 2424-2418.
- [10] OLUKOTUM K. The Case for a Single-Chip Multiprocessor[J]. Proc 7th Int Conf on Architectural Support for Programming Languages Operating Systems, 1996, 30(5): 2-11.
- [11] KELTCHER C N,MCGRATH K J,AHMED A, et al. The AMD Opteron processor for multiprocessor servers[J]. Micro IEEE, 2003, 23(2): 66-76.
- [12] SHI L W, FAN X Y, et al. Research of Chip Multiprocessor. [J]. Application Research of Computers, 2007, 24(9): 46-49.
- [13] LIU B W, CHEN S M, et al. Survey on Advance Microprocessor Architecture and Its Development Trends[J]. Application Research of Computers, 2007, 24(3): 16-20.
- [14] XIE X H, HU S T, LI H L. Multi-core/Many-core Processor and Its Influence on Computer Architecture Design[J]. Journal of Frontiers of Computer Science and Technology, 2008, 2(6): 641-650.
- [15] WANG K. Research and Design of Embedded Operating System Based on Multi-core Processors[D]. Nanjing: Nanjing University of Aeronautics and Astronautics, 2010.

- [16] NÁCUL A C, REGAZZONI F, LAJOLO M. Hardware Scheduling Support in SMP Architectures [C] // proceedings of the Design, Automation & Test in Europe Conference & Exhibition, F. 2007:1-6.
- [17] TANEBAUM A S. Modern Operating Systems [M]. Beijing: China Machine Press, 2017:292.
- [18] SEO M, KIM H S, MAENG J C, et al. An Effective Design of Master-Slave Operating System Architecture for Multiprocessor Embedded Systems [C] // Advances in Computer Systems Architecture (ACSAC 2007). Lecture Notes in Computer Science, 2007:114-125.
- [19] LEHEY G. Improving the FreeBSD SMP Implementation [C] // proceedings of the Freenix Track; Usenix Technical Conference. 2001:155-164.
- [20] CLARK R, QUIN J O, WEAVER T. Symmetric multiprocessing for the AIX operating system [C] // proceedings of the Digest of Papers COMPCON'95 Technologies for the Information Superhighway. 1995:110-115.
- [21] KLEIMAN S, VOLL J, EYKHOLT J, et al. Symmetric multiprocessing in Solaris 2.0 [C] // proceedings of the Compeon Spring '92 Thirty-Seventh IEEE Computer Society International Conference, Digest of Papers. 1992:181-186.
- [22] BIEMAN J M, OTT L M. Measuring Functional Cohesion [J]. IEEE Transactions on Software Engineering, 1994, 20(8):644-57.
- [23] SMITH, JIM, NAIR, et al. Virtual Machines: Versatile Platforms for Systems and Processes (The Morgan Kaufmann Series in Computer Architecture and Design) [M] // San Francisco: Morgan Kaufmann Publishers Inc. 2005:445-496.
- [24] AALTO A. Dynamic management of multiple operating systems in an embedded multi-core environment [D]. Helsinki: Aalto University, 2010.
- [25] YU Q, WEI H, LIU M, et al. A novel multi-OS architecture for robot application [C] // Proceedings of the 2011 IEEE International Conference on Robotics and Biomimetics. 2011:2301-2306.
- [26] ZHANG L. Research and Implement of Transplanting Dual Operating System Based on Dual-core ARM Processor Platform [D]. Xi'an: Xidian University, 2013.
- [27] MUIR S, SMITH J. AsyMOS-an asymmetric multiprocessor operating system [C] // Proceedings of the Open Architectures & Network Programming. 1998:25-34.
- [28] PHAM D C, AIPPERSPACH T, BOERSTLER D, et al. Overview of the architecture, circuit design, and physical implementation of a first-generation cell processor [J]. IEEE Journal of Solid-State Circuits, 2005, 41(1):179-96.
- [29] Zynq-7000 All Programmable SoC [OL]. [https://china.xilinx.com/support/documentation/product-briefs/c\\_zynq-7000-product-brief.pdf](https://china.xilinx.com/support/documentation/product-briefs/c_zynq-7000-product-brief.pdf).
- [30] OMAP-L137 C6747 DSP+Arm 处理器 [OL]. <https://www.ti.com.cn/product/cn/OMAP-L137#tech-docs>.
- [31] SUN Y, LI E, YANG G, et al. Design of a Dual-core Processor Based Controller with RTOS-GPOS Dual Operating System [C] // Proceedings of the 2019 IEEE International Conference on Mechatronics and Automation. 2019:1859-1864.
- [32] CHEN X, GU Y, WANG C, et al. Asymmetric multiprocessing for motion control based on Zynq SoC [C] // Proceedings of the 2016 International Conference on Field-Programmable Technology. 2017:315-318.
- [33] LI P, LU Y, WEI H, et al. Realization of embedded multimedia system based on dual-core Processor OMAP5910 [C] // The Proceedings of the Multiconference on Computational Engineering in Systems Applications. IEEE, 2006:101-105.
- [34] TSENG C, CHEN Y. Dynamic Dispatching Tasks Management for H. 264 Encoder on Heterogeneous Dual-Core Embedded System [C] // Proceedings of the 2010 International Conference on Communications and Mobile Computing. 2010:253-257.
- [35] ERDOS A, PARAMESWARAN S, SHEE S L. Heterogeneous multiprocessor implementations for JPEG; a case study [C] // proceedings of the Proceedings of the 4th International Conference on Hardware/Software Codesign and System Synthesis. 2006:217-222.



**QU Wei**, born in 1997, postgraduate. His main research interest includes embedded system.



**YU Fei-hong**, born in 1964, professor. His main research interests include optical design and image processing.