

基于双 PowerPC 7447A 处理器的嵌入式系统硬件设计

张中华^{1,2}

(1. 上海交通大学 电子信息与电气工程学院 上海 200030; 2. 中航雷达与电子设备研究院 江苏 无锡 214063)

摘要:随着雷达数据和信号处理需求的不断攀升,传统雷达数字处理系统的处理能力已渐显不足,因此有必要提高系统中每个处理单元的处理能力。鉴于此,设计一种基于 CPCI 标准总线和双 PowerPC 7447A 高性能处理器的通用处理单元硬件平台,并对部分功能单元的设计进行描述。硬件平台由双处理节点、双 PMC 接口和 CPCI 总线接口等组成,本地互连采用 PCI 总线,对外采用 CPCI 总线。该平台具有数据处理能力强、功能扩展性强、通用性强、维护方便等特点,有较高的应用价值。

关键词:PowerPC G4; 非对称多重处理; 全对称多重处理; 单指令多数据; 精简指令集计算机

中图分类号:TP368.4

文献标识码:B

文章编号:1004-373X(2008)24-009-05

Hardware Design of Embedded System Based on Dual PowerPC 7447A Processor

ZHANG Zhonghua^{1,2}

(1. School of Electrical and Information Engineering, Shanghai Jiaotong University, Shanghai, 200030, China;

2. Radar and Avionics Institute of AVIC, Wuxi, 214063, China)

Abstract: With the continuous requirement of the capability of data processing and signal processing in radar system, the capability of the traditional radar digital processing system is lacking gradually, so it is necessary to improve the capability of each unit. According to this, the paper proposes hardware design of an embedded system based on CPCI bus and dual PowerPC 7447A CPUs and describes some functional units. The hardware is made up of dual processing nodes, dual PMC interfaces and CPCI bus. It adopts PCI bus in the module and CPCI bus between the modules. It shows the characteristics of powerful processing capability, powerful developing function, great generality and convenient maintenance. It also has high application value.

Keywords: PowerPC G4; asymmetric multi-processing; symmetric multi-processing; single instruction multi-data; reduced instruction set computer

随着新一代战机的出现和战技指标需求的不断攀升,机载雷达作为航空电子产品中的重要一员,其功能和性能也要得到不断的完善和提高,但同时其重量和体积却要不断削减,以满足载机的作战需求。机载雷达的这种发展趋势给设计者带来了一定的困难,就其数字处理系统而言,功能的完备和性能的提高带来了大量数据的处理需求,增大数字处理系统的吞吐率和提高其处理能力成为要解决的首要问题,在体积重量相对缩减的前提下,完成高性能相关硬件的设计是设计者所面临的前所未有的挑战。

在目前的多功能雷达数字处理系统硬件设计中,通常采用 Intel80x86, AMD5x86 和 PowerPC 系列处理器作为数据处理的核心处理器,AD 公司和 TI 公司的 DSP 处理器作为信号处理的核心处理器。这种两类处理器共存的构架在提高了雷达性能的同时也带来了较大的问题:由于多方面原因,大多数雷达仍旧采用低性

能的处理器作为核心处理器,对于功能的增减,通过增减相关处理模块来解决,随之而来的是系统的笨重,硬件界线明显,体积和重量庞大,开发环境和调试手段繁琐复杂,研制周期长,维护困难;另外,由于内部通信效率低,虽然有些处理器内核运算速度快但数据吞吐量有限,影响性能指标的提高。

鉴于以上问题,本文设计一种基于 CPCI 标准总线和 2 个 PowerPC G4 高性能处理器 MPC7447A 的、采用非对称多重处理(ASMP)结构的通用嵌入式系统硬件平台,提高了单个处理单元的数据处理和吞吐能力,为快速构建和扩展多功能处理和通信系统缩短了研发周期,达到了系统紧凑高效、性能稳定可靠、扩展和维护方便快捷的目的。

1 系统概述

本文所描述的硬件平台基于 PowerPC G4 处理器技术,采用高性能的 MPC7447A 处理器和 MV64640 北桥控制器,通过 Altivec 矢量处理技术和高速 PCI 总

线互连技术,提供一个具有高速传输能力和高性能处理能力的硬件平台。

本硬件平台采用非对称多重处理(ASMP)结构设计,在单个模块中实现了2个分别进行高速数据处理的处理节点;利用互连的高速PCI总线,两个处理器可以互相通信和同步。当然,在多个这种模块间则可通过PMC子卡提供的高速串行接口进行高速数据交换,来方便地构建一个由多处理模块组成的级联或并行处理系统。

在本设计中,每个处理节点拥有独立的总线控制器、主存储器、FLASH存储器、以太网口、串行接口等多种通信接口和中断控制器等处理器外围电路。系统功能框图如图1所示。

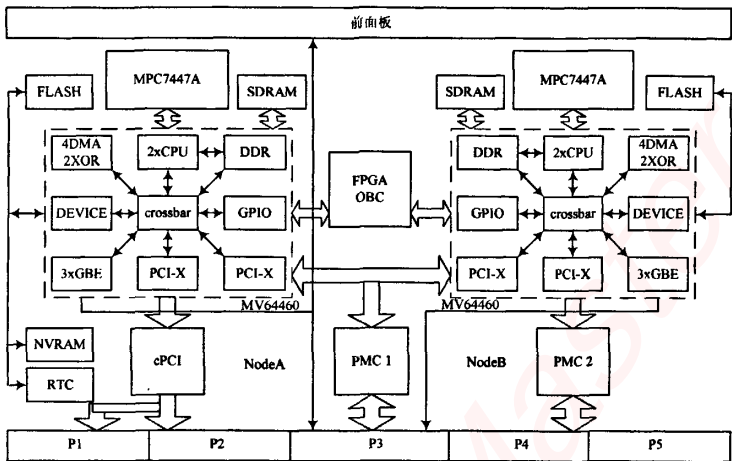


图1 系统功能框图

2 设计与实现

2.1 PowerPC 7447A 处理器性能简介

PowerPC 7447A 是 Freescale 公司推出的一款基于 PowerPC G4 技术的 32 b 高性能、低功耗超标量精简指令集计算机处理器,每个周期内可向 11 个独立的执行单元发送 4 个指令,在 64 kB 一级和 512 kB 二级 Cache、全对称多重处理技术(SMP)、133/166 MHz 64 b 总线接口和包含 128 b 执行指令的摩托罗拉领先的 AltiVec 单指令多数据(SIMD)矢量技术的支持下,可以完成高带宽数据处理和大密集度算法计算,当其内核主频工作在 1 167 MHz 时,功耗不超过 10 W,是一款具有超群性能和极低功耗的高效能处理器。

2.2 Marvell Discovery III MV64460 北桥性能简介

MV64460 北桥是 Marvell 公司推出的一款支持 PowerPC CPU 的高性能多总线接口的工业级主桥控制器,基于其高性能的 CROSSBAR 交换构架,提供端

口间的任意互连,并在 MPX 和 60x 总线模式下支持双处理器全对称多重处理(SMP)技术,通过集成多种独立的接口引擎来优化处理器对外设的访问频次。主要性能指标如下:

- (1) 主频 133/200 MHz, CROSSBAR 机构可提供 100 Gb/s 数据吞吐量;
- (2) 总线接口:1 个 64 b 200 MHz 处理器接口、1 个 72 b 200 MHz DDR SDRAM 接口、1 个 32 b 133 MHz 外设接口和 2 个 PCI/PCI-X 接口;
- (3) 集成外设和控制器:2 MB SRAM 存储器、3 个千兆以太网 MAC 控制器、2 个多协议串行控制器、2 个 XOR DMA 引擎和 4 个 IDMA 引擎。

2.3 处理节点设计

本硬件平台提供 2 个高性能处理节点(Node A, Node B),主要完成对数据的访问和处理,是本硬件平台的数据计算和交换中心。节点间通过 66/133 MHz 64 b PCI/PCI-X 总线互连,每个处理节点包括 1 个 MPC7447A 处理器,1 个 MV64460 总线控制器,1 GB DDR SDRAM,256 MB FLASH 和 16 MB 系统启动备份 FLASH,其中 Node A 为主处理节点,还配备有 128 kB NvRAM 和 RTC。主处理器节点(Node A)原理框图如图 2 所示。

MV64460 控制器作为处理节点的通信控制中心,为系统提供了丰富的控制器接口:2 个 PCI/PCI-X 总线设总线(Device Bus)接口、4 个 DMA 控制器、2 个

XOR DMA 控制器和 1 个可编程仲裁控制器等。设计中,外设总线(Device Bus)连接到 OBC(On Board Controller),用于对 FLASH, NvRAM, RTC、中断控制器等的逻辑控制;通过可编程仲裁控制器,可以对本地 DDR SDRAM 进行访问的各个设备的访问优先级进行调整,使对存储器的使用达到最优化;利用 DMA 控制器,可以在任意两个接口设备之间实现数据传输;使用 XOR DMA 控制器,则可以从最多 8 个源设备读取数据,进行位异或操作后将结果写入目标设备。

另外,节点间互连的高速 PCI/PCI-X 总线,是 PCI 设备之间的直接互连,不借助于任何 PCI-PCI 桥设备,只需要 1 次 PCI 总线仲裁,所以数据传输的延时非常小,从而很好地平衡了节点的处理能力和通信能力。

2.4 存储器单元设计

本单元包括 DDR SDRAM, FLASH, NvRAM 的设计。FLASH 和 NvRAM 存储单元原理框图见图 3。

(1) DDR SDRAM

每个处理节点包括 1 GB DDR SDRAM,带有 ECC 错误校验功能,工作时钟为 133 MHz,由于为双数据率存储器,其最大传输率可达 17 Gb/s,每次突发传输可达 128 B;SDRAM 被配置为 2 个区,每区 512 MB,可以被 MV64460 上的任何接口进行访问,在可编程控制器的控制下,可以被本地 CPU 总线和 PCI 总线直接访问,最多支持 8 页 SDRAM 页打开功能。

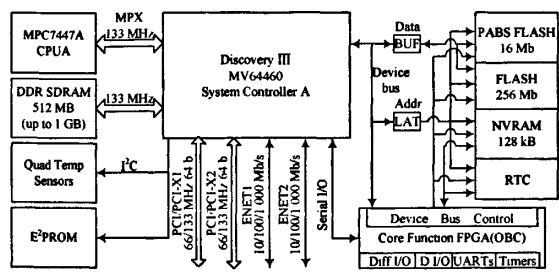


图 2 主处理器节点(Node A)原理框图

本设计采用 Micron 公司的 DDR266 SDRAM MT46V64M16 芯片,单片存储量为 1 Gb,由于单芯片的数据宽度选择为 16 位,因此需要 4 片并联以构成 64 位数据总线。

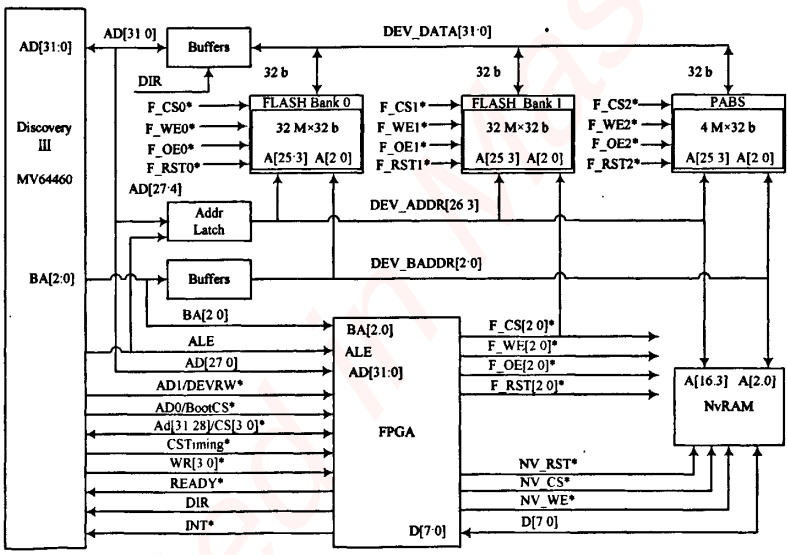


图 3 FLASH 和 NvRAM 存储单元原理框图

(2) FLASH

每个处理节点包括 256 MB 应用程序 FLASH 存储器和 16 MB 系统启动备份 FLASH 存储器,都连接于外设总线(Device Bus)上,可进行字节、半字、字的读操作和对齐字的写操作。所有 FLASH 分成 3 个区,应用程序 FLASH 分为 2 个区,每个区 128 MB,共使用 4 片 AMD 公司的 S29GL512N 芯片实现;系统启动备

份 FLASH 存储器独占 1 个区,使用 1 片 AMD 公司的 AM29PDL128G 芯片实现,用于系统在从应用程序区启动失败后的系统维护,该区的使用与否通过跳线或插件上相应管脚进行设置。本设计中 FLASH 的典型可重写次数为 100 000 次,典型数据保持时间为 20 年。为了保证数据安全,可以通过 FPGA 控制,对 FLASH 进行写保护。

(3) NvRAM

在主处理器节点(Node A)上提供 128 kB 数据宽度为 8 b 的自动存储非易失存储器,由 1 颗 Simtek 公司的 STK14CA8 芯片组成。该存储器连接于外设总线(Device Bus)上,可进行无限次读写访问,在系统电源故障时,其通过内部的电容网络来保护快速 SRAM 中的数据不丢失;在系统电源恢复时,保存的重要数据自动被系统调用。另外,通过软件控制,也可以调整该存储器的数据存储和读出周期数。在本设计中,还采取写保护位和写保护跳线 2 种方式来保护其存储的数据。

2.5 本地 PCI 总线单元设计

本单元主要包括 PCI 本地总线、PCI - cPCI 桥和 PMC 接口的设计。

(1) PCI 本地总线

平台中设计了 3 个本地 PCI 总线,均通过 MV64460 控制器的 PCI 桥进行互连和访问控制,其工作速率和类型为 66/133 MHz 64 b PCI/PCI - X 总线,拓扑形式如图 1 所示:主处理器节点(Node A)中的第一个 PCI 接口连接 cPCI 桥,第二个 PCI 接口连接 PMC1 接口和从处理器节点(Node B)中的第二个 PCI 接口;从处理器节点(Node B)中的第一个 PCI 接口连接 PMC2 接口。

在本设计中,对 PCI 总线号的分配采用静态和动态相结合的方式,3 个本地 PCI 总线中每一个 PCI 总线都被指定了一段总线号,而在每一个 PCI 总线上扩展的其他 PCI 总线,将采用动态分配总线号的形式,分配预定总线号段中的 1 个。在

系统复位后,系统会启动代码自动枚举 PCI 设备,并将 PCI 总线号配置给每个 PCI 设备。

(2) PCI - cPCI 桥

平台提供 66 MHz 64 b CompactPCI 接口,采用 PCI6540 芯片实现系统 PCI - X 总线到 PCI 总线的扩展,设计中配置 PCI6540 芯片工作在 Universal 工作模

式,在无需硬件配置的情况下,当模块插入到 cPCI 总线的系统槽或外设槽时,可自适应系统槽和外设槽。并且在系统槽中时,该芯片配置自己工作在透明模式,为系统提供电气上的隔离,以便多个 66 MHz 64 b 的 PCI 外设添加到 PCI-X 插槽中;在外设槽时,该芯片配置自己工作在非透明模式,为系统提供寻址和电气上的隔离,以便创建多处理器系统。另外,PCI6540 芯片还支持异步工作方式,以便芯片两边的端口工作在不同的时钟域内,不会因同步慢速设备而降低快速设备的工作效率,以获得更高工作效能。

(3) PMC 接口

本平台可同时安装 2 个 PMC 子卡,以达到迅速扩展系统功能的目的。2 接口均支持 66/133 MHz 64 b PCI/PCI-X 接口标准,传输速度峰值可达 1 GB/s,以支持数据传输率较大的光纤、显卡等高速设备。设计中 PMC 接口均被映射到每一个处理节点,以使每个处理节点都能从 PMC 接口上获得高速数据交换。每一个 PMC 接口提供 4 个 PCI 中断源,通过 OBC 控制,用户可以指定任意一个处理节点对任意一个中断源进行处理。

另外,在布线上考虑到通用性,2 个 PMC PN4 接插件上的 I/O 信号线分别通过平台上的 JNP4 连接到下面板 3 号和 4 号接插件上,且 I/O 信号线以差分信号线对的方式引线,在设计中还对其阻抗和线长进行了严格控制,以获得数据在 I/O 信号线的高速传输。

2.6 OBC 单元设计

平台提供 OBC(On Board Controller)控制器,设计中使用 XILINX 公司的 1 颗 XC3S1000 芯片实现各功能外设的控制逻辑,例如复位、中断、计数器、看门狗定时器、串口、GPIO 寄存器、差分 I/O 寄存器、处理节点通信、外设的片选译码(如 FPGA 寄存器、FLASH, PABS, NvRAM, RTC 等)等。每个处理节点通过 MV64460 的外设总线(Device Bus)接口连接到 OBC,进行访问控制。设计中采用 1 个并行 PROM 为 OBC 存储逻辑代码,并在系统上电时将代码加载到 OBC 中。OBC 功能框图见图 4。

2.7 复位控制单元设计

系统复位控制单元主要完成系统中所有功能单元复位信号的产生,由 OBC 中的复位控制器实现。通过外设总线(Device Bus),每个处理节点可以对复位控制器中控制寄存器进行操作,以实现对特定硬件的复位。在设计中严格控制了所有复位信号的产生次序,以保证系统正常稳定的工作。此外,为保证在 OBC 初始化配置后,以下复位信号按照设计时序要求依次置为复位无

效状态(MV1_PCI0RSTn, MV1_PCI1RSTn, MV2_PCI0RSTn, MV2_PCI1RSTn, PMC1_RSTn, PMC2_RSTn, MV1_P0RSTn, MV1_P1RSTn, MV2_P0RSTn, MV2_P1RSTn, MV1_FLASHRSTn, MV2_FLASHRSTn, MV1_SYSRSTn, MV2_SYSRSTn)。这里还在这些信号上接入下拉电阻,以保证在 OBC 正常工作之前所有复位信号输出复位有效状态。

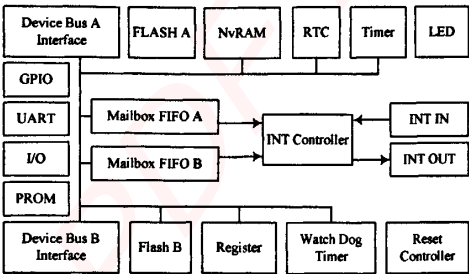


图 4 OBC 功能框图

2.8 中断控制单元设计

在本系统中主要中断源包括 PMC, cPCI 总线、以太网等,设计中所有中断源均被连接到 OBC 上一个软件可配置的中断控制器上。通过对这个中断控制器的配置,每个处理单元均可以灵活地选择管理所有中断源。该功能单元原理框图如图 5 所示。

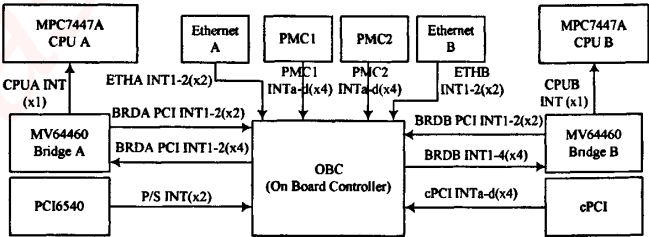


图 5 中断控制单元功能框图

工作时,处理节点通过外设总线将各个中断信号的屏蔽信息同时写入中断屏蔽寄存器中,该屏蔽信息与中断请求寄存器(显示当前所有中断输入管脚的当前状态)中的相应位进行操作后,即可得到当前所有有效的中断源的中断信息,并存入中断服务寄存器中。当中断产生时,如果信号是非屏蔽信号,而且系统又没有把所有中断源屏蔽掉,处理器就会接收到中断信号,并通过外设总线和寄存器控制逻辑来读取中断信息,以确定中断源的向量号,来调用相应中断处理程序进行处理。在读取中断信息后,系统将自动清除相应的中断信息。中断控制器逻辑功能图见图 6 所示。

2.9 其他单元设计

本系统提供 4 个 10/100/1 000 Mb/s 自适应以太网接口,均符合 IEEE802.3 标准,每个处理节点 2 个,

利用 MV64460 中 MAC 控制器实现控制,其外部的 PHY 使用 Marvell 公司的 88E1111 芯片实现。设计中每个节点均有一个网络接口引到前面板和后面板。

系统也提供 4 个通用串行接口,每个处理节点 2 个,利用 MV64460 中 2 个 MPSC(多协议串行控制器)控制器实现控制,通过软件对 OBC 中的相关逻辑进行选择,以外接不同的接口芯片来实现符合 EIA - 232 或 EIA - 422/485 标准的串行接口。

I²C(Intelligent Interface Controller)是系统提供的另外一种总线接口,利用 MV64460 中集成的控制器实现相关控制,在该总线上,连接了 4 个温度传感器和 1 个 E²PROM 芯片。温度传感器采用 MAX6634 芯片,分别放置在重要器件的旁边采集工作环境的温度;E²PROM 是 1 颗 AT24C04 芯片,用于存储 MV64460 芯片的配置信息和系统其它相关信息,其在 I²C 总线上被分配到 0 号地址上。

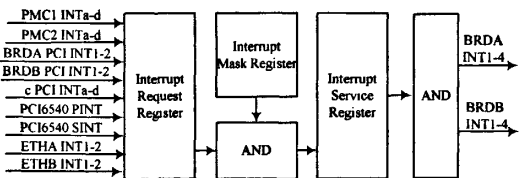


图 6 中断控制器逻辑功能框图

RTC(Real Time Clock)由 1 颗 Dallas 公司的 DS1501 芯片实现,连接到主处理节点的外设总线上,为系统提供日期、时间信息,以及提供 RTC 和看门狗定时、上电复位、电池监控、256 B 的 NvSRAM 和 32.768 kHz 的时钟输出等功能。

3 结 语

本文结合高性能 MPC7447A 处理器和 MV64640 北桥控制器的特性,设计了一种采用非对称结构技术、

G4 AltiVec 矢量处理技术、高速 PCI 总线互连技术和高速串行总线互连技术的双处理节点硬件平台。由于该平台具有数据和信号处理能力强、数据传输速度快、功能扩展方便快捷、通用性强等特点,有利于快速构建雷达数字处理系统和其他高性能数据、信号处理系统,以减少模块种类和数量,降低系统研发成本和人力资源开销,缩短系统研发和维护周期,降低设计风险,提高产品质量。本设计应用前景广阔,为雷达数据和信号处理系统提供了一种新的设计方法和实现途径,具有很高的应用价值。

参 考 文 献

[1] 李承伟,杜娟,李波,等.多 PowerPC 7400/7410 处理器体系架构研究[J].计算机工程,2005,31(6):189-191.
[2] 林鹏.基于多机处理的雷达数据处理系统[J].舰船电子对抗,2005,28(5):52-55.
[3] Freescale Semiconductor Inc. MPC7447A RISC Microprocessor Hardware Specifications[S]. 2006.
[4] Freescale Semiconductor Inc. MPC7447A RISC Microprocessor Family Reference Manual Specifications[S]. 2005.
[5] Marvell Semiconductor Inc. MV64460, MV64461, and MV64462 System Controller for PowerPC Processors Hardware Specification[S]. 2005. Doc. No. MV-S101286-01, Rev. D.
[6] Marvell Semiconductor Inc. MV64460, MV64461, and MV64462 System Controller for PowerPC Processors User Manual[S]. 2006. Doc. No. MV-S101286-00, Rev. C.
[7] PLX Technology Inc. PCI 6450CB Data Book[S]. 2004.
[8] 马鸣锦,朱剑冰,何红旗,等. PCI, PCI-X 和 PCI Express 的原理及体系结构[M]. 北京:清华大学出版社,2007.
[9] Edward Solari, George Wills. PCI & PCI-X Hardware and Software Architecture & Design[M]. Fifth Edition. San Diego: Annabooks, 2001.

作者简介 张中华 男,1977 年出生,在职硕士研究生。研究方向为嵌入式系统研究。

多晶硅产业重新洗牌政企联动提升竞争力

近几年,伴随着太阳能光伏产业的发展,多晶硅市场需求猛增,造成全球多晶硅价格飙升。然而面对全球金融危机的冲击,国外订单迅速减少,多晶硅价格痛遭“腰斩”,太阳能光伏行业的冬天提前到来。但业内专家指出,太阳能光伏行业提前过冬未必是一件坏事,它将促进多晶硅行业重新洗牌,更好地迎接下一轮黄金时期的到来。

全球金融危机的蔓延,对中国太阳能光伏行业,尤其是对处于上游的多晶硅材料行业的发展产生巨大的冲击。在多晶硅价格方面,今年 9~10 月期间,由于受多晶硅产能和产量陆续扩大和预期价格下跌的影响,多晶硅现货从上半年的 400~500 美元 1 kg 已下降到 250~300 美元 1 kg。而目前更是降到了 200 美元以下。

(摘自中国电子报)

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](#)
17. [基于 VPX 总线的工件台运动控制系统研究与开发](#)
18. [PCI Express 流控机制的研究与实现](#)
19. [UART16C554 的设计](#)
20. [基于 VPX 的高性能计算机设计](#)
21. [基于 CAN 总线技术的嵌入式网关设计](#)
22. [Visual C 串行通讯控件使用方法与技巧的研究](#)
23. [IEEE1588 精密时钟同步关键技术研究](#)
24. [GPS 信号发生器射频模块的一种实现方案](#)
25. [基于 CPCI 接口的视频采集卡的设计](#)
26. [基于 VPX 的 3U 信号处理平台的设计](#)
27. [基于 PCI Express 总线 1394b 网络传输系统 WDM 驱动设计](#)
28. [AT89C52 单片机与 ARINC429 航空总线接口设计](#)
29. [基于 CPCI 总线多 DSP 系统的高速主机接口设计](#)
30. [总线协议中的 CRC 及其在 SATA 通信技术中的应用](#)
31. [基于 FPGA 的 SATA 硬盘加解密控制器设计](#)
32. [Modbus 协议在串口通讯中的研究及应用](#)
33. [高可用的磁盘阵列 Cache 的设计和实现](#)
34. [RAID 阵列中高速 Cache 管理的优化](#)

35. [一种新的基于 RAID 的 CACHE 技术研究与实现](#)
36. [基于 PCIE-104 总线的高速数据接口设计](#)
37. [基于 VPX 标准的 RapidIO 交换和 Flash 存储模块设计](#)
38. [北斗卫星系统在海洋工程中的应用](#)
39. [北斗卫星系统在远洋船舶上应用的研究](#)
40. [基于 CPCI 总线的红外实时信号处理系统](#)
41. [硬件实现 RAID 与软件实现 RAID 的比较](#)
42. [基于 PCI Express 总线系统的热插拔设计](#)
43. [基于 RAID5 的磁盘阵列 Cache 的研究与实现](#)
44. [基于 PCI 总线的 MPEG2 码流播放卡驱动程序开发](#)
45. [基于磁盘阵列引擎的 RAID5 小写性能优化](#)
46. [基于 IEEE1588 的时钟同步技术研究](#)
47. [基于 Davinci 平台的 SD 卡读写优化](#)
48. [基于 PCI 总线的图像处理及传输系统的设计](#)
49. [串口和以太网通信技术在油液在线监测系统中的应用](#)
50. [USB30 数据传输协议分析及实现](#)
51. [IEEE 1588 协议在工业以太网中的实现](#)
52. [基于 USB30 的设备自定义请求实现方法](#)
53. [IEEE1588 协议在网络测控系统中的应用](#)
54. [USB30 物理层中弹性缓冲的设计与实现](#)
55. [USB30 的高速信息传输瓶颈研究](#)
56. [基于 IPv6 的 UDP 通信的实现](#)
57. [一种基于 IPv6 的流媒体传送方案研究与实现](#)
58. [基于 IPv4-IPv6 双栈的 MODBUS-TCP 协议实现](#)

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](#)
16. [基于 VxWorks 嵌入式系统的中文平台研究与实现](#)
17. [VxBus 的 A429 接口驱动](#)
18. [基于 VxBus 和 MPC8569E 千兆网驱动开发和实现](#)
19. [一种基于 vxBus 的 PPC 与 FPGA 高速互联的驱动设计方法](#)
20. [基于 VxBus 的设备驱动开发](#)
21. [基于 VxBus 的驱动程序架构分析](#)
22. [基于 VxBus 的高速数据采集卡驱动程序开发](#)
23. [Vxworks 下的冗余 CAN 通讯模块设计](#)
24. [WindML 工业平台下开发 S1d13506 驱动及显示功能的实现](#)
25. [WindML 中 Mesa 的应用](#)
26. [VxWorks 下图形用户界面开发中双缓冲技术应用](#)
27. [VxWorks 上的一种 GUI 系统的设计与实现](#)
28. [VxWorks 环境下 socket 的实现](#)
29. [VxWorks 的 WindML 图形界面程序的框架分析](#)
30. [VxWorks 实时操作系统及其在 PC104 下以太网编程的应用](#)
31. [实时操作系统任务调度策略的研究与设计](#)
32. [军事指挥系统中 VxWorks 下汉字显示技术](#)
33. [基于 VxWorks 实时控制系统中文交互界面开发平台](#)
34. [基于 VxWorks 操作系统的 WindML 图形操控界面实现方法](#)
35. [基于 GPU FPGA 芯片原型的 VxWorks 下驱动软件开发](#)
36. [VxWorks 下的多串口卡设计](#)
37. [VxWorks 内存管理机制的研究](#)
38. [T9 输入法在 Tilcon 下的实现](#)
39. [基于 VxWorks 的 WindML 图形界面开发方法](#)
40. [基于 Tilcon 的 IO 控制板可视化测试软件的设计和实现](#)

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](#)

9. [Android 应用的反编译](#)
10. [基于 Android 行为的加密应用系统研究](#)
11. [嵌入式 Linux 系统移植步步通](#)
12. [嵌入式 C++ 语言精华文章集锦](#)
13. [基于 Linux 的高性能服务器端的设计与研究](#)
14. [S3C6410 移植 Android 内核](#)
15. [Android 开发指南中文版](#)
16. [图解 Linux 操作系统架构设计与实现原理（第二版）](#)
17. [如何在 Ubuntu 和 Linux Mint 下轻松升级 Linux 内核](#)
18. [Android 简单 mp3 播放器源码](#)
19. [嵌入式 Linux 系统实时性的研究](#)
20. [Android 嵌入式系统架构及内核浅析](#)
21. [基于嵌入式 Linux 操作系统内核实时性的改进方法研究](#)
22. [Linux TCP IP 协议详解](#)
23. [Linux 桌面环境下内存去重技术的研究与实现](#)
24. [掌握 Android 7.0 新增特性 Quick Settings](#)
25. [Android 应用逆向分析方法研究](#)
26. [Android 操作系统的课程教学](#)
27. [Android 智能手机操作系统的研究](#)
28. [Android 英文朗读功能的实现](#)
29. [基于 Yocto 订制嵌入式 Linux 发行版](#)
30. [基于嵌入式 Linux 的网络设备驱动设计与实现](#)
31. [如何高效学习嵌入式](#)
32. [基于 Android 平台的 GPS 定位系统的设计与实现](#)
33. [Linux ARM 下的 USB 驱动开发](#)
34. [Linux 下基于 I2C 协议的 RTC 驱动开发](#)
35. [嵌入式下 Linux 系统设备驱动程序的开发](#)
36. [基于嵌入式 Linux 的 SD 卡驱动程序的设计与实现](#)
37. [Linux 系统中进程调度策略](#)
38. [嵌入式 Linux 实时性方法](#)
39. [基于实时 Linux 计算机联锁系统实时性分析与改进](#)
40. [基于嵌入式 Linux 下的 USB30 驱动程序开发方法研究](#)
41. [Android 手机应用开发之音乐资源播放器](#)
42. [Linux 下以太网的 IPv6 隧道技术的实现](#)
43. [Research and design of mobile learning platform based on Android](#)
44. [基于 linux 和 Qt 的串口通信调试器调的设计及应用](#)
45. [在 Linux 平台上基于 QT 的动态图像采集系统的设计](#)
46. [基于 Android 平台的医护查房系统的研究与设计](#)
47. [基于 Android 平台的软件自动化监控工具的设计开发](#)
48. [基于 Android 的视频软硬解码及渲染的对比研究与实现](#)

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](#)
17. [Windows CE 环境下无线网卡的自动安装](#)
18. [基于 Windows CE 的可视电话的研究与实现](#)
19. [基于 WinCE 的嵌入式图像采集系统设计](#)
20. [基于 ARM 与 WinCE 的掌纹鉴别系统](#)
21. [DCOM 协议在网络冗余环境下的应用](#)
22. [Windows XP Embedded 在变电站通信管理机中的应用](#)
23. [XPE 在多功能显控台上的开发与应用](#)
24. [基于 Windows XP Embedded 的 LKJ2000 仿真系统设计与实现](#)
25. [虚拟仪器的 Windows XP Embedded 操作系统开发](#)

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](#)

8. [基于 MPC8247 嵌入式电力交换系统的设计与实现](#)
9. [基于设备树的 MPC8247 嵌入式 Linux 系统开发](#)
10. [基于 MPC8313E 嵌入式系统 UBoot 的移植](#)
11. [基于 PowerPC 处理器 SMP 系统的 UBoot 移植](#)
12. [基于 PowerPC 双核处理器嵌入式系统 UBoot 移植](#)
13. [基于 PowerPC 的雷达通用处理机设计](#)
14. [PowerPC 平台引导加载程序的移植](#)
15. [基于 PowerPC 嵌入式内核的多串口通信扩展设计](#)
16. [基于 PowerPC 的多网口系统抗干扰设计](#)
17. [基于 MPC860T 与 VxWorks 的图形界面设计](#)
18. [基于 MPC8260 处理器的 PPMC 系统](#)
19. [基于 PowerPC 的控制器研究与设计](#)
20. [基于 PowerPC 的模拟量输入接口扩展](#)
21. [基于 PowerPC 的车载通信系统设计](#)
22. [基于 PowerPC 的嵌入式系统中通用 I/O 口的扩展方法](#)
23. [基于 PowerPC440GP 型微控制器的嵌入式系统设计与研究](#)

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](#)
11. [基于 S5PV210 的频谱监测设备嵌入式系统设计与实现](#)
12. [Uboot 中 start.S 源码的指令级的详尽解析](#)
13. [基于 ARM9 的嵌入式 Zigbee 网关设计与实现](#)
14. [基于 S3C6410 处理器的嵌入式 Linux 系统移植](#)
15. [CortexA8 平台的 \$\mu\$ C-OS II 及 LwIP 协议栈的移植与实现](#)
16. [基于 ARM 的嵌入式 Linux 无线网卡设备驱动设计](#)
17. [ARM S3C2440 Linux ADC 驱动](#)
18. [ARM S3C2440 Linux 触摸屏驱动](#)
19. [Linux 和 Cortex-A8 的视频处理及数字微波传输系统设计](#)

20. [Nand Flash 启动模式下的 Uboot 移植](#)
21. [基于 ARM 处理器的 UART 设计](#)
22. [ARM CortexM3 处理器故障的分析与处理](#)
23. [ARM 微处理器启动和调试浅析](#)
24. [基于 ARM 系统下映像文件的执行与中断运行机制的实现](#)
25. [中断调用方式的 ARM 二次开发接口设计](#)
26. [ARM11 嵌入式系统 Linux 下 LCD 的驱动设计](#)
27. [Uboot 在 S3C2440 上的移植](#)
28. [基于 ARM11 的嵌入式无线视频终端的设计](#)
29. [基于 S3C6410 的 Uboot 分析与移植](#)
30. [基于 ARM 嵌入式系统的高保真无损音乐播放器设计](#)
31. [UBoot 在 Mini6410 上的移植](#)
32. [基于 ARM11 的嵌入式 Linux NAND FLASH 模拟 U 盘挂载分析与实现](#)

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](#)
8. [基于 COM Express 架构的数据记录仪的设计与实现](#)
9. [基于 COM Express 的信号系统逻辑运算单元设计](#)
10. [基于 COM Express 的回波预处理模块设计](#)
11. [基于 X86 平台的简单多任务内核的分析与实现](#)
12. [基于 UEFI Shell 的 PreOS Application 的开发与研究](#)
13. [基于 UEFI 固件的恶意代码防范技术研究](#)
14. [MIPS 架构计算机平台的支持固件研究](#)
15. [基于 UEFI 固件的攻击验证技术研究](#)
16. [基于 UEFI 的 Application 和 Driver 的分析与开发](#)
17. [基于 UEFI 的可信 BIOS 研究与实现](#)
18. [基于 UEFI 的国产计算机平台 BIOS 研究](#)
19. [基于 UEFI 的安全模块设计分析](#)
20. [基于 FPGA Nios II 的等精度频率计设计](#)
21. [基于 FPGA 的 SOPC 设计](#)
22. [基于 SOPC 基本信号产生器的设计与实现](#)
23. [基于龙芯平台的 PMON 研究与开发](#)

24. [基于 X86 平台的嵌入式 BIOS 可配置设计](#)
25. [基于龙芯 2F 架构的 PMON 分析与优化](#)
26. [CPU 与 GPU 之间接口电路的设计与实现](#)
27. [基于龙芯 1A 平台的 PMON 源码编译和启动分析](#)
28. [基于 PC104 工控机的嵌入式直流监控装置的设计](#)
29. [GPGPU 技术研究与实现](#)
30. [GPU 实现的高速 FIR 数字滤波算法](#)
31. [一种基于 CPU/GPU 异构计算的混合编程模型](#)
32. [面向 OpenCL 模型的 GPU 性能优化](#)

Programming:

1. [计算机软件基础数据结构 - 算法](#)
2. [高级数据结构对算法的优化](#)
3. [零基础学算法](#)
4. [Linux 环境下基于 TCP 的 Socket 编程浅析](#)
5. [Linux 环境下基于 UDP 的 socket 编程浅析](#)
6. [基于 Socket 的网络编程技术及其实现](#)
7. [数据结构考题 - 第 1 章 绪论](#)
8. [数据结构考题 - 第 2 章 线性表](#)
9. [数据结构考题 - 第 2 章 线性表 - 答案](#)