

一种基于并行处理器的快速车道线检测系统及FPGA实现

李元金 张万成 吴南健

(中国科学院半导体研究所 北京 100083)

摘要: 该文提出了一种并行的快速车道线检测系统。该系统包含一个 32×32 的处理器单元(PE)阵列和双RISC子系统。PE阵列实现车道线图像像素级并行预处理,获取图像边缘特征,双RISC核子系统根据边缘特征实现两条车道线直线参数的并行检测,从而使得检测过程的每一步都是并行进行,显著提高检测速率。该系统用FPGA实现。实验结果表明本系统具有良好的鲁棒性且可达到每秒50帧的检测速率,满足了车道偏离预警系统实时性要求,具备重要的应用价值。

关键词: 图像处理; 车道线检测; 并行; FPGA; 精简指令集计算机(RISC)

中图分类号: TP391.41; TN432

文献标识码: A

文章编号: 1009-5896(2010)12-2901-06

DOI: 10.3724/SP.J.1146.2010.00111

A Fast Traffic Lane Detection System Based on Parallel Processors and FPGA Implementation

Li Yuan-jin Zhang Wan-cheng Wu Nan-jian

(Institute of Semiconductors, Chinese Academy of Sciences, Beijing 100083, China)

Abstract: This paper proposes a parallel fast traffic lane detection system. The system consists of a 32×32 Processing Elements (PE) array and a dual RISC core subsystem. The PE array performs pixel-parallel image preprocessing and outputs edge features, the dual RISC core subsystem performs two lanes parameters detection in parallel based on edge features. In this way, every step in the detection process is in parallel and the detection rate is rapidly increased. The system is implemented with FPGA. The experiment shows that it has good robustness and can reach up to 50 fps. This meets the demand of real-time for lane departure warning system and makes an important sense for practical application.

Key words: Image processing; Traffic lane detection; Parallel; FPGA; RISC (Reduced Instruction Set Computer)

1 引言

车道偏离预警系统是解决汽车驾驶安全的重要手段^[1,2],其研究的关键是如何快速和准确检测出车道偏离。传统的车道线检测都是基于PC机,采用软件实现,算法复杂,速度慢,无法满足实时性的要求。随着近年来信息技术的发展,尤其是像FPGA和DSP等可编程器件的快速发展,使得在小型器件上也能得以实现车道偏离预警,如Pankiewicz研究小组用FPGA实现了车道线的检测^[3]。而Hsiao Pei-yung研究小组开发出了用于车道线检测的芯片^[4]。然而这些研究工作主要是将传统的检测算法流程的一部分或者几个部分做成专用的硬件实现,图像处理过程在本质上还是逐个像素串行进行,导致其检测速率还是有限。基于图像处理的车道偏离检测速度依赖于图像的并行处理程度,并行度越高,则其检测速率越快。视觉芯片是一种采用2维的像素级并

行处理架构,能够实现对整个图像或图像块中的所有像素同时并行处理,从而实现每秒高达1000帧以上的高速图像处理^[5,6]。然而其缺乏处理高度复杂算法的能力,无法完成如直线检测等复杂的高级图像处理。

本文提出了一种基于视觉芯片构架的快速车道线检测片上系统。该系统主要包含一个 32×32 2维的处理单元(PE)阵列和双精简指令集计算机(RISC)子系统。PE阵列实现对车道线图像的像素级并行预处理,完成噪声去除和图像边缘特征提取等低级图像处理;而双RISC核子系统根据边缘特征实现对两条车道线直线参数的并行检测等高级图像处理。快速车道线检测片上系统通过PE阵列和双RISC核的两级并行图像处理,能快速和准确地检测出车道线,实现车道偏离预警。本文在以下的几节中将分别介绍并行快速车道线检测系统架构,主要模块设计与实现,检测过程图像处理的并行实现,FPGA实现及实验结果。

2 并行快速车道线检测系统架构

图1为本文提出的并行快速车道线检测片上系统的构架图。它包含一个 32×32 的PE阵列、阵列程序存储器ROM和阵列控制器、双RISC核子系统、仲裁器和多路选择器MUX等模块。PE阵列包含有阵列输入和输出端口。整个阵列构成一个并行单指令多数据(SIMD)系统,在阵列控制器的控制下,通过阵列程序存储器ROM供给指令,实现各种图像并行预处理。双RISC核子系统则有两个RISC处理器及相应的指令存储器ROM和数据存储器RAM构成,它们通过AHB总线互联。双RISC核通过仲裁器控制多路选择器MUX,实现和PE阵列以及外部的互连。双RISC核子系统从PE阵列获取图像预处理结果并行进行高级图像处理,并将处理结果通过多路选择器MUX输出。PE阵列控制器和双RISC子系统之间能够互相通信,实现PE阵列并行处理和双RISC核并行处理之间的同步,从而使整个系统协调一致工作。

这个系统充分考虑检测过程图像处理的并行特征。基于图像处理的检测过程,包含有大量数据量低复杂度的图像预处理和小数据量高复杂度的高级图像处理。图像预处理是由整块图像所有像素共同参与的过程,呈现出数据量的庞大,而每个像素的处理仅仅是和相邻像素之间按一定权重因子的加减运算,属于低复杂度的局部运算。因而图像预处理是一个适合于2维并行的处理。本系统中,采用PE阵列的2维并行结构来实现图像的预处理,从而显著提高图像预处理的速度。而高级图像处理实现检测的过程,是基于预处理结果获取的特征信息,因而数据量大为减小,且数据量之间相关性不大,需要做各种变换,涉及多次的乘法,查询,统计,排序等

高复杂度运算,所以由RISC核完成。然而两条车道线的检测,互相独立,互不相干,属于任务并行,所以用两个RISC核同步进行检测,从而实现高级图像处理中的并行。由此,本系统设计了双RISC核子系统,实现两条车道线的并行检测,从而进一步提高检测速度。

本系统正是基于在图像处理每一个阶段的最大化并行处理的结构设计,从而实现快速检测。其检测过程如下:首先感兴趣区域图像分块输入到PE阵列;其次,PE阵列并行完成图像的各种预处理,获取边缘特征信息;再次,双RISC核子系统从PE阵列中获取边缘特征信息,并行进行两条车道线直线参数的检测;最后,将检测结果输出。

3 主要模块设计与实现

3.1 PE及其阵列

如图2所示为PE的结构及其电路实现。PE是由一个1位的ALU和两个存储容量为256位的SRAM A和32位的SRAM B单元构成。PE中的算术逻辑运算单元(ALU)能完成1位的加法,与,或,非等操作。DATA1和DATA2为ALU的两个操作数,其中DATA1从两片SRAM的输入中选取,而DATA2从4个近邻PE和两片SRAM中通过多路选取器选择。ALU中还包含一个进位寄存器C_Reg,用来寄存每次加法的进位。由于引入了两片能同时读写的双口SRAM,PE能够在一个时钟周期内从SRAM中读取两个操作数,运算并将结果写回到SRAM,从而增强了处理能力。PE实现各种操作是在控制指令和SRAM的读写地址下完成的。虽然PE一个周期只能完成一位的运算,但是能够在多个周期内重复进行各种一位运算来完成任意位宽的复杂运算。如图3所示为PE多周期完成复杂运算示例及其所需周期

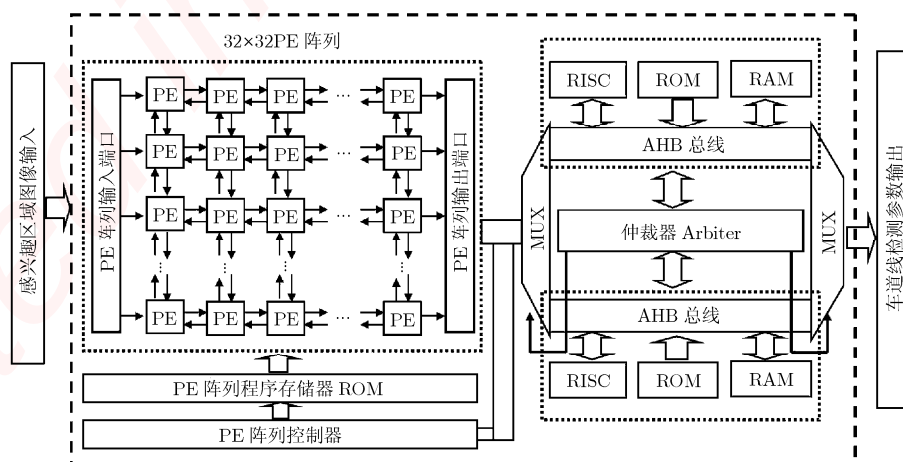


图1 并行快速车道线检测系统架构

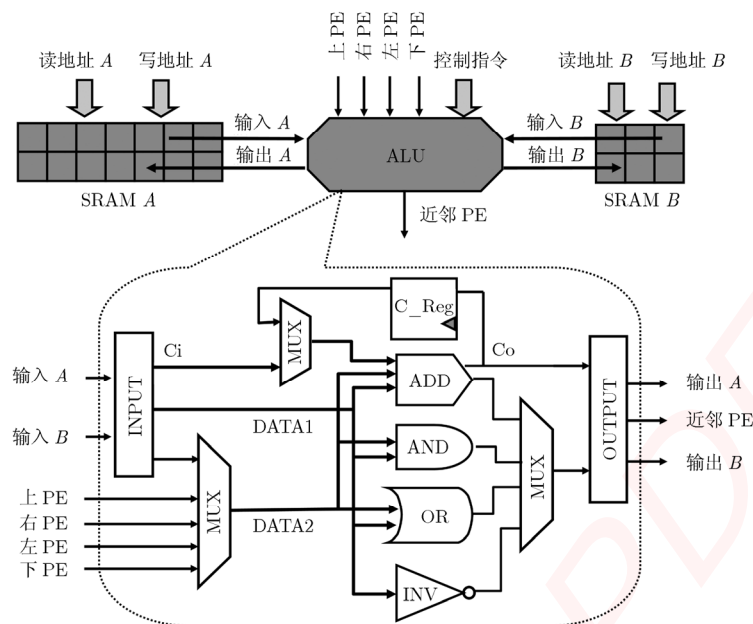


图2 PE及其电路实现

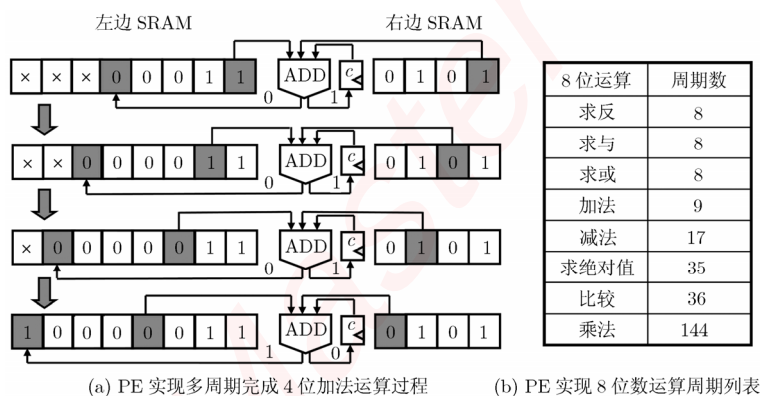


图3 PE多周期完成复杂运算示例及所需周期列表

数。其中图3(a)所示的是PE在多周期内完成4位加法过程。从低到高逐位相加，每次进位都寄存，以参与下次的加法，最终完成4位的加法。类似的多位运算，都可以划分为有限次数1位运算的组合，PE能够在多个周期内完成。图3(b)中列表是PE完成典型的8位运算所需周期数。

PE通过上下左右输入端口和近邻输出端口与近邻PE互连，构成PE阵列。1024个PE构成一个 32×32 的阵列。所有的PE接受共同的控制指令和SRAM读写地址，完成同样的操作，构成一个SIMD系统。PE阵列和图像之间的有多种对应关系。一个PE可以存储一个或多个像素，一个PE阵列可以对应一幅或多幅图像，所以PE阵列可以多次分块扫描处理高分辨率的图像。如对于一个 $N \times M$ 的PE阵列，要处理一幅 $N1 \times N2$ 的图像，如果 $N1, N2$ 大于 N, M ，则可以将图像分成 $N1 \times N2 / (N \times M)$ 块，分块扫描处理。

3.2 双RISC核子系统

每一个RISC核子系统包含有一个32位的RISC核，程序存储器ROM和数据存储器RAM。它们之间通过AHB总线互连。RISC核带有32位的ALU，能够单周期内实现加法，与或非，判断跳转等多种基本运算，另外还有 32×8 位的乘法器单元，能够快速实现乘法运算。RISC核从ROM中获取指令，从RAM中获取数据，实现程序设定的各种操作，并将结果存放到RAM中。两个RISC核之间通过仲裁器控制多路选择器MUX的连接关系，从而实现和PE阵列以及外部的互通信。

4 车道线图像并行处理的实现

4.1 车道线图像分析

图4展示的是一张典型的车道线图像。根据余天洪^[4]等人的研究成果，只需要选择感兴趣区域进行车



图4 典型的车道线图像分析

道线检测即可。另外车道线检测的目的是应用于车道偏离预警系统，所以只需要检测当前车道上的左右两条车道线，确保车行驶在当前车道而不偏离。因此，本文根据以上结论，选择以QVGA格式的图片为处理对象，感兴趣区域选择为整个图片处于下方的2/5区域，且分割成左右两条车道线的两个区域，这样每个区域的大小为：160×96像素。

4.2 图像预处理的并行实现

4.2.1 图像边缘提取的并行实现 边缘提取一般都是基于模板运算完成。模板运算都是局部相邻像素间按一定权重因子的相互加减，因而能够在整块图像上并行进行。如图5所示为利用3×3的拉普拉斯模板算子进行边缘提取的图示。如式(1)所示，通过上下左右4个相邻像素的相加，再减去中间像素的4倍，就得到该点的边缘值。而这处理过程在PE阵列里可以如下实现：首先所有的PE依次和其上下左右4个PE相加，相加之和存放在当前的PE；其次，当前PE对应的像素值向左移两次，实现4倍操作；再次，相加之和减去移位后的像素值，获得边缘值。图5只示意了其中两个PE对应的像素边缘提取的过程，实际所有的PE均完成一样的操作。

$$I_d^*(i, j) = \{I_d(i+1, j) + I_d(i-1, j) + I_d(i, j+1) + I_d(i, j-1)\} - 4I_d(i, j) \quad (1)$$

4.2.2 图像边缘分割的并行实现 在经过边缘提取后，需要进行边缘分割以明确区分边缘和背景。进行边缘分割的关键是找出最佳分割阈值。当像素的灰度值大于阈值时，认为是边缘点，分割为灰度值是255的白点，用1表示；反之认为是背景，弱化为灰度值是0的黑点，用0表示，如式(2)所示。本文借助PE阵列的快速并行运算，设计了一种基于循环迭代来确定最佳分割阈值和分割结果的算法。通过合理设定分割图像中白点的个数作为循环终止条件，

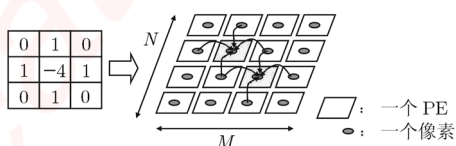


图5 边缘提取拉普拉斯模板PE阵列实现示例

给出初始阈值，进行并行比较和比较结果的并行相加，统计白点的个数，进行循环判决，如不满足，增大阈值，开启新一轮循环。如此循环直到终止，完成分割任务。图6展示的是PE阵列进行边缘分割一次循环的过程。首先输入阈值，接着进行并行比较，如果当前像素灰度值比阈值大，则其结果为1，反之则为0，再通过PE阵列的行相加和列相加实现比较结果统计，最后输出统计结果，进行循环判决。

$$F_d(i, j) = \begin{cases} 1, & I_d^*(i, j) \geq Th \\ 0, & I_d^*(i, j) < Th \end{cases} \quad (2)$$

4.3 Hough变换的并行实现

图7(a)展示的是Hough变换检测直线的原理图。在直角坐标系下的一条直线方程 $y = kx + b$ ，对应于变换空间的一点 (r, θ) ，而直角坐标系下一点 (x, y) 对应于变换空间一条曲线 $r = \rho(\theta)$ 。同属于一条直线上的点，在变换空间对应的各条曲线必定相交于同一点，这一点的坐标值即为直线参数。通过在 (r, θ) 空间投票统计各曲线相交于同一点的次数，来确定对应直角坐标系下的直线段分布。次数越多，对应的直线段越长。次数最多的点对应直角坐标系下最长的直线段。其映射关系如式(3)所示。

$$r = x \cos \theta + y \sin \theta \quad (3)$$

在车道线检测过程中，车道线所在直线段就是所在平面最长的直线段，即对应变换空间各曲线相交次数最多的点，所以只要找出这个点，就检测出了车道线的直线参数。图7(b)是本文中涉及到的图像坐标和直角坐标之间的关系。其中 i, j 是图像坐标，而 x, y 是直角坐标。由此可得其坐标关系如式(4)：

$$\begin{cases} x = i - 160, & 0 \leq i < 320 \\ y = 96 - j, & 0 \leq j < 96 \end{cases} \quad (4)$$

将式(4)代入式(3)得到本文的Hough变换公式如下：

$$r = (i - 160) \cos \theta + (96 - j) \sin \theta \quad (5)$$

另外根据余天洪^[1]的研究结果， θ 的取值范围右边为 $15^\circ \sim 75^\circ$ ，左边为 $105^\circ \sim 165^\circ$ ，并且在量化的过程中，步长为 1° 。

图8是进行Hough变换并检测出车道线直线参数的算法流程图。首先将变换空间所有 (r, θ) 点所对应的统计计数数目置为0，其次逐个像素按照图像坐标系顺序去获取分割后的图像，如果是白点，就按照式(5)进行Hough变换，并且将变换结果对应点的统计数目加1，如此循环直到所有的点都变换完，最后统计变换空间内数目最多的点，并将该点对应的变换空间的坐标作为当前车道线所在的直线参数。因

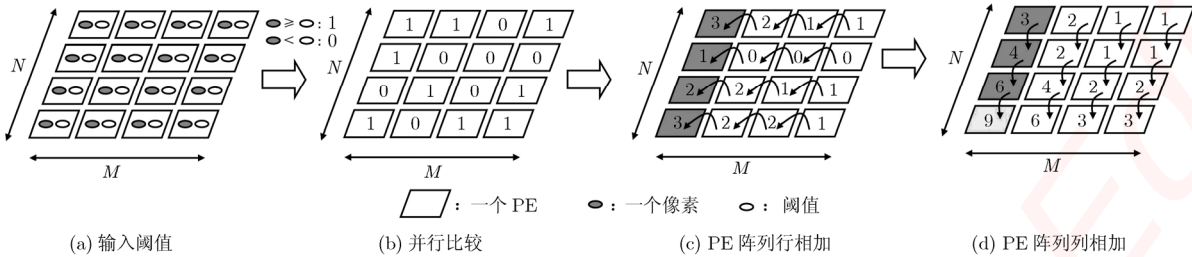


图6 PE阵列实现快速边缘分割算法过程示意图

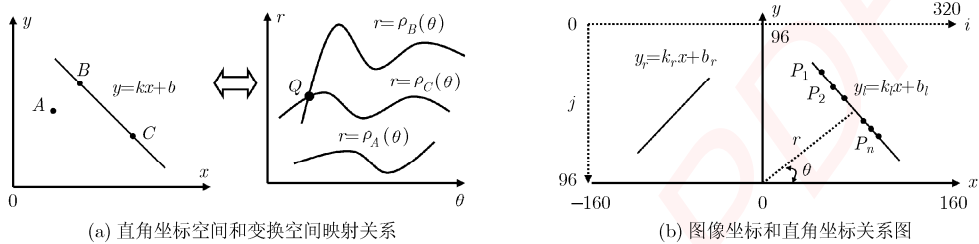


图7 基于Hough变换的直线检测原理及坐标变换关系

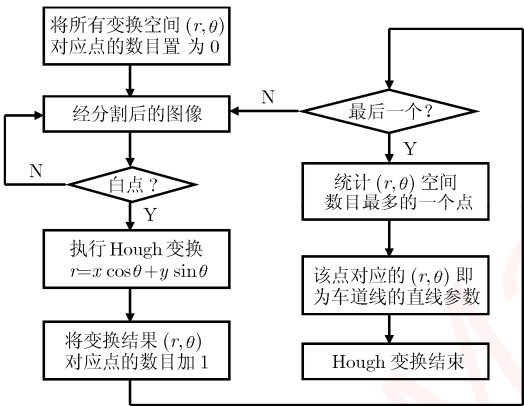


图8 Hough变换算法流程图

为两条车道线的检测是相互独立的，所以可以利用双RISC核分别对左右车道线实施如上所述的Hough变换检测车道线，从而实现Hough变换的并行进行。

5 FPGA实现及其实验结果

FPGA采用的是Altera公司的Stratix II

EP2S180，资源消耗见表1。

图9是检测识别实验结果。图9(a)，9(b)，9(c)，9(d)取自各种不同光强情况下的典型情况，均能正确检测，可见本文提出的并行检测系统具有很强的鲁棒性和适应性。表2是在最坏情况下检测过程的时间开销分布。由于引入了2维并行的PE阵列，将图像预处理的时间大幅降低，从无并行结构中的60%降低到并行结构中的4%，同时两个RISC核子系统的并行处理，使Hough变换时间减小一半，从而达到了每秒50帧的检测速率。

表1 FPGA消耗资源表

FPGA	Stratix II EP2S180
最大频率(MHz)	100 MHz
ALUTs	18528/143520(13%)
Registers	5553
Memory Bits	1867776/9383040(20%)
PLLs	1/12 (8%)

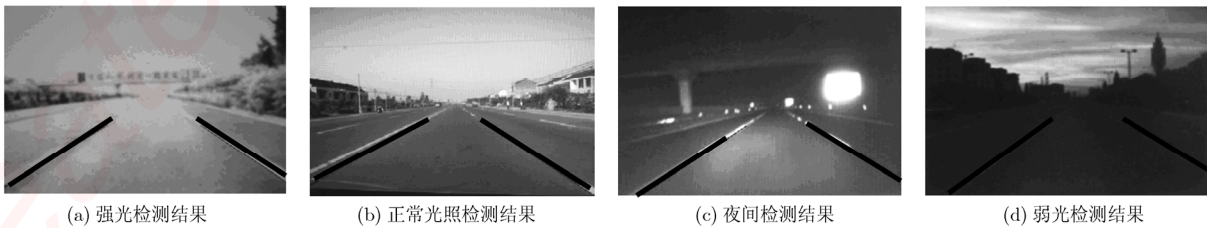


图9 实验结果

表2 检测过程时间开销对比

处理过程	并行架构		单个RISC		并行/ RISC
	时间(ms)	百分比(%)	时间(ms)	百分比(%)	提高倍数
边缘提取	0.0492	0.24	6.758	7.3	137.4
边缘分割	0.76	3.84	49	51.8	64.5
Hough变换	19	95.92	38	40.9	2
总的时间	19.8	100	92.8	100	4.5
帧率(帧/s)	50		11		4.5

表3给出了本文和已有工作的对比，由此可见，本文提出的快速检测系统，不仅具有良好的鲁棒性和可靠性，而且极大的提高了检测效率，具备良好的实时性。

表3 结果对比

研究成果	架构	硬件	检测速率(fps)
文献[7]	-	PC	<15
文献[8]	并行+流水线	FPGA+ARM	25
本文	并行	FPGA	50

6 结论

本文提出了一种并行快速车道线检测系统，实现检测过程的每一步图像处理都并行进行，从而显著提高检测速率。该系统用FPGA实现，实验结果表明本系统具有良好的鲁棒性和可靠性，并且达到了每秒50帧的检测速率，满足车道偏离预警系统等安全辅助驾驶系统的实时性要求，具有重大的现实意义。

参 考 文 献

[1] 余天洪. 基于机器视觉的车道偏离预警系统的研究[D]. [博士学位论文], 吉林大学, 2006.
Yu T. Study on vision-based lane departure warning system [D]. [Ph.D. dissertation], Jilin University, 2006.

[2] 王荣本, 余天洪, 郭列. 基于机器视觉的车道偏离预警系统研究综述[J]. 汽车工程, 2005, 27(4): 463-464.
Wang R, Yu T, and Guo L. A survey on research of vision-based lane departure warning system [J]. *Automotive Engineering*, 2005, 27(4): 463-464.

[3] Pankiewicz P, Powietowski W, and Roszak G. VHDL

implementation of the lane detection algorithm [C]. *Mixed Design of Integrated Circuits and Systems*, Poland, June 19-21, 2008: 581-584.

[4] Hsiao Pei-yung, Cheng Hsien-chein, and Yeh Chun-wei. Automobile lane detection system-on-chip integrated with mixed signal mode CMOS image sensor[C]. *Consumer Electronics*, Macau, June 14-16, 2005: 320-323.

[5] Miao W, Lin Q Y, Zhang W C, and Wu N J. A programmable SIMD vision chip for real-time vision applications[J]. *IEEE Journal of Solid-State Circuit*, 2008, 43(6): 1470-1479.

[6] Lin Q Y, Miao W, Zhang W C, Fu Q Y, and Wu N J. A 1000 frames/s programmable vision chip with variable resolution and row-pixel-mixed parallel image processors [J]. *Sensors*, 2009, 9(8): 5933-5951.

[7] 田炳香, 郑榜贵, 吴晴. 高速公路车道线检测与跟踪算法研究[J]. 现代电子技术, 2008, 9(31): 180-183.
Tian B, Zheng B, and Wu Q. Research on lane detection and tracking algorithm of highway[J]. *Modern Electronic Technique*, 2008, 31(9): 180-183.

[8] Huang Pei-yung, Yeh Chun-wei, and Huang Shih-shinh. A portable vision-based real-time lane departure warning system based on embedded calculating technique [C]. *Vehicular Technology Conference*, Australia May 7-10, 2006, 6: 2982-2986.

李元金: 男, 1984年生, 硕士生, 研究方向为高速图像处理及其集成电路设计.

张万成: 男, 1985年生, 博士生, 研究方向为单电子及其电路、高速图像处理及其集成电路设计.

吴南健: 男, 1961年生, 博士, 研究员, 目前的研究方向包括数模混合集成电路、射频集成电路、视觉芯片、RFID、单电子电路等.

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](#)
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](#)
59. [RS485CAN 网关设计与实现](#)
60. [MVB 周期信息的实时调度](#)
61. [RS485 和 PROFINET 网关设计](#)
62. [基于 IPv6 的 Socket 通信的实现](#)
63. [MVB 网络重复器的设计](#)
64. [一种新型 MVB 通信板的探究](#)
65. [具有 MVB 接口的输入输出设备的分析](#)
66. [基于 STM32 的 GSM 模块综合应用](#)
67. [基于 ARM7 的 MVB CAN 网关设计](#)

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](#)
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](#)
41. [基于 VxWorks 的通信服务器实时多任务软件设计](#)
42. [基于 VXWORKS 的 RS485MVB 网关的设计与实现](#)
43. [实时操作系统 VxWorks 在微机保护中的应用](#)
44. [基于 VxWorks 的多任务程序设计及通信管理](#)

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](#)
49. [基于 Android 移动设备的加速度传感器技术研究](#)
50. [基于 Android 系统振动测试仪研究](#)
51. [基于缓存竞争优化的 Linux 进程调度策略](#)

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](#)
26. [基于 EVC 的嵌入式导航电子地图设计](#)
27. [基于 XPEmbedded 的警务区 SMS 指挥平台的设计与实现](#)
28. [基于 XPE 的数字残币兑换工具开发](#)

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 IO 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](#)
24. [基于双 PowerPC 7447A 处理器的嵌入式系统硬件设计](#)
25. [基于 PowerPC603e 通用处理模块的设计与实现](#)
26. [嵌入式微机 MPC555 驻留片内监控器的开发与实现](#)
27. [基于 PowerPC 和 DSP 的电能质量在线监测装置的研制](#)
28. [基于 PowerPC 架构多核处理器嵌入式系统硬件设计](#)

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](#)
33. [基于 ARM11 的电源完整性分析](#)
34. [基于 ARM S3C6410 的 uboot 分析与移植](#)
35. [基于 S5PC100 移动视频监控终端的设计与实现](#)

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 CPUGPU 异构计算的混合编程模型](#)
32. [面向 OpenCL 模型的 GPU 性能优化](#)
33. [基于 GPU 的 FDTD 算法](#)
34. [基于 GPU 的瑕疵检测](#)
35. [基于 GPU 通用计算的分析与研究](#)

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](#)