文章编号：1001－9081(2008)10－2686－04

一种基于异构双核处理器的嵌入式操作系统构架设计

蒋建春[1,2]，曾素华[3]，岑 明[1]
(1.重庆邮电大学 自动化学院，重庆 400065；2.重庆大学 光电工程学院，重庆 400044；
3.重庆邮电大学 计算机科学与技术学院，重庆 400065)
(dennis_jjc@yahoo.com.cn)

摘 要：当前应用于异构多核处理器嵌入式操作系统主要是主从式结构。这种结构由于存在通信协议和结构的异构性，使操作系统在实现上和通信处理存在复杂性。根据异构双核处理器的特点，通过分析通信硬件的结构，在单核嵌入式操作系统构架的基础上，提出一种基于对称通信的双核处理器嵌入式操作系统构架，解决了异构双核处理器中的通信效率和共享内存的利用问题。该操作系统构架成功应用于TI OMAP5910处理器。

关键词：嵌入式操作系统；操作系统构架；异构双核处理器；对称通信
中图分类号：TP316.2 文献标志码：A

Architecture design of embedded operating system based on heterogeneous dual-core processor

JIANG Jian-chun[1,2], ZENG Su-hua[3], CEN Ming[1]
(1. College of Automatization, Chongqing University of Posts and Telecommunications, Chongqing 400065, China;
2. College of Opto-Electronic Engineering, Chongqing University, Chongqing 400044, China;
3. College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: The master-slave embedded operating system is widely used on heterogeneous multiprocessor. The architecture and communication protocol is heterogeneous, so it makes the realization and communication handling complicated. Through analyzing the structural characteristics of heterogeneous dual-core processor, a symmetric communication operating system architecture was proposed based on uniprocessor operating system, and resolves the problem how to make full use of the shared memory to communicate and to improve the efficiency of communication between processors in heterogeneous dual-core processor. This architecture operating system is used successfully on TI OMAP5910.

Key words: embedded operating system; operating system architecture; heterogeneous dual-core processor; symmetric communication

0 引言

随着微电子技术和计算机技术的发展，处理器从单核到双核，以及到多核，得到了飞速的发展。当前多核处理器主要有两种结构：一种是对称同构多核处理器，这种处理器中各个核具有相同的处理器结构，共享内存和外设，主要用于通用计算机领域；一种是非对称异构多核处理器[1-2]，它们具有不同结构的处理器内核，有各自的内存空间和外设，每个处理器核具有自己的系统时钟频率，通过处理器通信接口和共享内存进行通信，这种处理器主要用于嵌入式设备领域。根据处理器的结构，在多核处理器上的操作系统大体分成3类。一是以并行计算技术（Symmetric Multi-Processing，SMP）[3-5]为主的操作系统。SMP要求各个处理器结构必须要求一致，所有核心上运行单一的操作系统映像，共享SMP内核的数据和代码，通常用于对称多核处理器。二是基于非对称处理（Asymmetric Multi-Processing，AMP）技术的主从式操作系统[6-9]。在不同结构的处理器中运行具有不同功能不同构架的操作系统内核，主要应用于非对称异构多核处理器。第三种是分布式操作系统[10]。这种操作系统主要应用于分布式系统。

由于异构双核处理器在处理器结构、频率、共享内存的编址等不同，每个处理器可以在公共全局内存中有自己的分区，加上共享内存空间一般比较小，不足以运行操作系统。因此，SMP不适合非对称异构多核处理器。分布式操作系统由于不能利用共享内存，因此不适合异构处理器。针对这种情况，基于主从结构的多核操作系统被提出，在不同的处理器中采用不同结构和功能的操作系统内核。

但是，这种主从式操作系统采用非对称处理技术来设计操作系统构架，其主要特点是根据处理器的功能特点采用AMP技术将操作系统也分成主从式结构，主操作系统和从操作系统存在差异。操作系统间通信主要利用处理器间的通信接口，在不同的操作系统中定义不同的通信协议来实现通信，操作系统之间从构架和实现都是非对称。主从操作系统在实现上存在差异，并且没有过多结合共享内存来实现操作系统的功能，通常是在操作系统构架中加入相应的抽象层来管理处理器间的通信。如文献[9]在主从式操作系统中加入了一个抽象层来管理整个操作系统的任务。主操作系统的功能和从操作系统的功能和结构上存在差异，这种结构使得操作系

收稿日期：2008－04－18；修回日期：2008－06－02。
基金项目：国家863计划项目（2006AA11A1C1）；重庆市教委科技研究项目（kj080501）。
作者简介：蒋建春（1975－），男，四川岳池人，博士研究生，主要研究方向：计算机控制、嵌入式系统；曾素华（1977－），女，重庆丰都人，讲师，硕士，主要研究方向：计算机控制、嵌入式系统；岑明（1972－），男，四川新津人，博士，主要研究方向：目标跟踪、数据融合。
?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

统在通信协议上也各不相同，构架也存在异构性。

而大多数应用在嵌入式领域的非对称异构多核处理器，特别是双核处理器中，除了具有相应的通信接口以外，同时还有存在共享内存。对于这种多核处理器，处理器之间除了可以通过通信接口进行少量数据通信以外，还可以利用共享内存实现大量数据的交换。主从式操作系统不能充分利用共享内存进行通信，通信速率和数据量都受到限制，整个处理器的性不能得到充分发挥。在另一方面，操作系统是用户程序和硬件的中间接口，大多数嵌入式操作系统设计成模块式，功能可裁剪[11-13]。因此，操作系统的设计中处理器的异构性可以不考虑。同时，在许多应用场合对称通信是必要的。本文主要从提高异构双核处理器的通信效率和简化操作系统的结构出发，提出一种新型的操作系统构架，称为 SMRTOS（Symmetric communication MultiProcessing Real-time OS）。另外，本文还将讨论共享内存通信。本文不涉及操作系统的调度和算法问题，这些已经比较成熟[14-16]。

1　异构双核处理器的特点

在异构双核处理器中，每个处理具有各自的结构特征，它们之间的通信是通过处理器间的通信模块和部分共享内存来实现的。消息邮箱是双核处理器的公共外设，具有双向的中断机制，通过中断来通知不同处理器，实现不同处理器的通信。

这种机制提供了一种处理器间非常灵活的软件协议，一般包括两个单向消息邮箱：一个用于处理器 1发送，处理器 2接收；一个用于处理器 1接收，处理器 2发送。处理器可以通过消息邮箱实现少量数据的快速传递。

共享存储是双核处理器在每个处理器核的片外存储器空间中预留了一段两个处理器都能访问的地址空间，每个处理器都可以根据自己的地址来访问。因此共享存储器空间可以作为双核处理器实现数据量较大的数据交换场所。

根据异构双核处理器的这种特点，通过设置共享存储空间，定义双核处理器访问共享存储空间的协议，很好地实现处理器间的通信。这种方式可以结合消息邮箱来建立一个正确访问共享存储器的同步握手中断。当通信的数据大于消息邮箱提供的通信量时，可以通过这种方式来实现处理器间通信。例如，相对于单个命令和指针，需要传递一个指针列表时，通过共享存储单元和消息邮箱，当接收到有处理器 1通过消息邮箱写入命令产生的中断时，处理器 2可以从共享区读指针列表。

2　对称通信异构双核处理器嵌入式操作系统

操作系统的主要功能就是对任务和硬件进行管理。传统的单核处理器嵌入式操作系统已经解决了任务间的通信问题，如通过消息队列、消息邮箱、事件来实现任务间数据的交换；通过对消息操作来触发与消息相关任务调度，如等待消息可以挂起任务、消息产生可以触发等待消息任务就绪；同时单核处理器操作系统也解决了由于外部事件产生而设置某个事件状态或者激活挂起任务就绪等工作，在中断服务程序中对任务通信数据结构的访问等问题，如 μCOS-II [17]在中断服务程序中可以进行任务激活、设置事件、消息邮箱等工作。不管处理器存在什么差异，操作系统的调度机制存在一致性。

在单核处理器嵌入式操作系统中可以在内核中处理硬件中断服务程序，如 μCOS-II 中的系统时钟中断服务程序 OSTimeTick，只需要在硬件管理模块中提供一个时钟中断入口就可以。在中断服务程序中调用内核内核函数，如激活任务。这两种情况为在单核操作系统上实现多核处理器操作系统构架提供了一个很好的基础。但这些都是在单核处理器上的实现，都是基于对同一个内存的访问。在异构双核多处理器中的每个核有自己的存储空间，同时双核处理器间还有共享存储空间。处理器之间可以通过通信接口和共享内存进行通信。因此，单核处理器操作系统不能直接应用到异构双核处理器中，必须对其构架进行改造。

本文根据异构双核处理器对称通信的要求，设计一种应用于异构双核处理器上的对称结构的嵌入式操作系统。

2.1　对称通信异构双核处理器嵌入式操作系统构架设计

典型的嵌入式非对称异构双核系统将应用代码分成两个主要部分：控制与用户接口代码和不间断处理代码。控制用户接口部分运行在主微处理器上，不间断处理代码运行在其他微处理器内核上。运行在异构双核处理器上的操作系统除了各自运行自己独立的程序外，涉及操作系统之间任务的信息传递和交换，在硬件上主要是通过消息邮箱和共享内存来实现。对于操作系统，由于处理器的异构环境，不可能实现一个任务的并行处理，而是实现多任务的并行执行，相对一个任务的并行处理来说，在多任务的并行处理时，每个任务有自己独立的数据结构，任务间的通信数据量不大。因此，在异构双核处理器中操作系统内核间的任务不涉及大量数据的交换，可以采用消息和共享内存的方法来实现操作系统间的通信和同步。

操作系统主要是用来管理硬件资源，为用户提供一个方便快捷的编程接口，每个操作系统可以具备很多功能，如任务、调度管理、文件系统、网络的基本功能，并且可以根据具体需要进行功能裁减和系统时钟的设置。操作系统本身不存在主次之分，只存在功能的多少，嵌入式操作系统应该提供用户可裁减功能。在异构双核处理器中每个处理器核拷贝一个操作系统内核，按照自己的内存管理、任务调度、时钟频率工作，在需要进行数据交换的时候按照对称的协议进行通信。用户应该可以根据处理器提供的功能对操作系统进行裁减。因此，本文从异构双核处理器的通信角度，基于消息机制的通信方式提出一种用于非对称异构双核处理器上的对称嵌入式操作系统构架的设计，不分主次，以相同的构架实现异构处理器间的通信。

要使对称结构的嵌入式操作系统能在非对称异构双核处理器上正确地运行，必须保证通信协议的一致，包括消息结构和操作方式。主要包括三个方面：中断服务的处理、消息事件结构一致和处理器间基于消息的任务调度。综合考虑这三者之间的相互联系，在操作系统的内核构架中加入专门用于管理处理器间通信的模块，实现处理消息邮箱的中断、消息参数的传递、与消息相关的任务状态的设置、调度的触发等功能。

由于在双核处理器中，处理器核之间利用通信硬件接口的中断以及共享内存来实现通信，虽然它们的硬件实现形式可能存在不同，但通信的数据结构和协议存在一致性。利用这种一致性，将用于处理器间通信的硬件抽象层和相应的通信管理程序嵌入操作系统内核中，作为内核的一部分，给硬件抽象层留出一个接口。如图 1所示内核阴影部分，在不同处理器核中，操作系统构架是相同的。为了保证消息传递的实时性，对通信处理部分采用中断服务来进行管理。在通信模块中断服务程序中建立处理器通信的协议，并建立与之相对

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

应的消息结构和任务调度触发机制。也就是说,通信中断不仅传递通信相关的参数,还需触发调度使等待该消息的任务进入就绪。这就要求操作系统内核必须是抢占式。

图 1 异构双核处理对称通信操作系统内核结构

和单核处理器嵌入式操作系统相比,在硬件抽象层和操作系统内核中分别增加一个模块,用于管理处理器间通信。硬件抽象层中阴影部分主要针对双核之间通信的硬件而增加,主要负责通信硬件接口的读写,包括消息邮箱和共享内存。在单核操作系统内核增加一个专门用于处理处理器间通信的模块,该模块主要实现处理操作系统之间的通信协议,处理操作系统间的消息、事件、任务之间的关系,如触发对应事件标志、传递消息参数、置任务为就绪态、激活挂起任务、触发任务调度等。每个操作系统的通信部分结构都是对称的,也就是每个操作系统都可以对其他操作系统主动或被动地发送消息。

2 2 操作系统通信管理模块的设计

通信管理模块是构架设计的核心部分,该模块实现了各个处理器核任务间通信和同步。由处理器各核之间的任务间通信产生的所有操作都是通过该模块直接或间接处理。通信管理模块相当于集成到内核中的通信接口的中断服务程序,处理器间通信接口产生中断后,通过中断入口地址进入通信管理模块,中断管理模块根据通信内容触发相应动作。

保证不同处理器核之间的任务正常的通信和同步是该模块设计的重点。本文中采用消息来实现处理器核间的任务通信,通过事件建立消息与任务的联系,然后根据通信机制对消息进行处理和事件参数进行设置,并触发相应的操作。

根据传递数据的长度不同消息可以分成两类:一类是符号型消息,消息数据少,通过消息邮箱就可以一次全部传输;另外一种是数据型消息,通信数据量较大,通过消息邮箱需要进行多次传输。由于需要消息的任务可能处于挂起状态,不能实时读取消息,需要中断写入到消息队列中进行保存,增加了输入写入过程,数据传输效率低,这时就利用共享内存来实现大量数据的交换,而消息邮箱负责传递消息相关的参数,中断处理将该参数写入与任务相关的事件中,消息的数据被写入共享内存对应消息位置,操作系统任务只需在对应的事件中读取消息的参数,然后在共享存储区读取消息值,不会因为任务还未进入运行态而丢失数据。由于共享内存的数据单元长度对每个处理器都是一样的,只要保证操作系统通信的消息结构完全一致,不同处理器分时进行访问,就可以正确进行数据交换。

由于各个核间的任务可能存在不同步,消息需要保存以便于在任务执行时读取。因此,除了单核操作系统具有的事件模块外,在不修改现有操作系统功能的情况下,需增加一个事件模块来关联任务和来自其他处理器核的消息。事件主要包括,事件标志,用来标示消息是否有效;消息类型,用来描述该消息是符号型消息还是数据型消息;消息指针,消息的指针指向存放消息的地址,符号型消息指向消息队列,数据型消息指向共享内存该消息的首地址;消息长度,用来保存消息的长度,不同的数据型消息其长度可能不相同,在事件初始化时,通过定义的消息长度来设置;等待任务控制块指针,该指针指向该消息关联的任务控制块,通过该指针用于读取等待消息的任务状态,在通信管理模块中当消息到来时,通过该指针将该关联任务设置为就绪态,并调用调度函数。一个消息对应一个事件,事件在创建时采用数组来和消息一一对应。

在单核操作系统中,任务间的通信都是在同一个内存中实现,同一个内存地址是唯一的,操作系统对同一地址的访问也是分时,不存在访问冲突问题。而在非对称异构双核处理器中,各个处理器按照自己的时钟周期去访问共享内存,共享内存地址对不同处理器其地址也是不一样,存在不同处理器同时访问而发生冲突的问题。要利用消息邮箱和共享内存来实现操作系统消息的传递,必须对操作系统消息结构和共享内存的地址分配作统一规定。对于消息结构,操作系统采用统一的结构,然后设置相同的参数,经过编译器编译指向同一个地址。为了防止不同处理器同时访问同一个地址,采用环形消息队列,每个队列只能是一个操作系统写,一个操作系统读。任务对消息读写直接在共享内存中进行,不需要数据的传递,如图 2中步骤①。

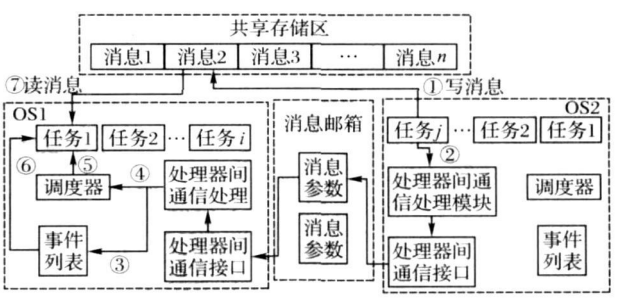

图 2 消息传递过程

在对共享内存的消息传递时,对于数据型消息,任务需要传递消息参数,通过图 2中步骤②。由于共享存储区在每个处理器中的地址不一样,如果传递消息起始地址需要进行变换才能保证正确读出数据。根据这种情况,对于共享存储区消息的传递的方式有两种。一种是传递消息的起始地址,那么需要将一个处理器里的消息起始地址转换成另一个处理器的地址,由于结构一致,所以就能正确读出数据。另外一种是在编译器中通过地址分配表给消息在内存中固定分配地址空间,直接设置消息的起始地址指向共享内存的同一地址,各个消息依次排列,在消息传递时,消息邮箱只需要传递消息的标号,操作系统根据消息编号对消息数据进行操作。

根据这种构架,在操作系统中用一种事件来建立任务与消息的联系,在通行管理模块中设置对应任务事件标志。各个操作系统在任务建立时配置与任务相关的事件、消息的参数,建立任务、事件、消息的关系。在消息传递时只需要传递消息的编号,通信管理模块根据消息参数在对应事件结构体中设置事件的状态标志、消息的参数、激活对应任务为就绪等操作,如图 2中的步骤③④,然后触发任务调度⑤。当等待该消息的任务处于运行态时,首先检测与之相对应的事件状态标志⑥,如有效,查看消息参数,然后在共享内存中读取消息值⑦。

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

3 结语

这种构架的操作系统可以在不改变结构的情况下运行在双核处理器的每个核中。不存在主从之分，每个操作系统内核结构和通信协议相同。该操作系统构架可以通过单核操作系统构架增加处理器间通信处理模块来快速实现，不同处理器端可以根据功能需求在不改变操作系统构架的情况下进行裁减。这种结构操作系统已成功地应用于 OMAP5910。

参考文献：

[1] WOLF W. The future of multiprocessor systems-on-chips[C]// Proceedings of the 41st Annual Conference on Design Automation. New York, NY, USA: ACM, 2004: 681—685.

[2] KUMAR R, TULLSEN D M, JOUPPI N P, et al. Heterogeneous chip multiprocessors[J]. IEEE Computer, 2005, 38(11): 32—38.

[3] LEHEY G. Improving the FreeBSD SMP implementation[C]// Proceedings of the FREENIX Track: 2001 USENIX Annual Technical Conference. Berkeley, CA, USA: USENIX Association, 2001: 155—164.

[4] CARBONE J. A SMP RTOS for the ARM MPCore multiprocessor[J]. ARM Information Quarterly, 2005, 4(3): 64—67.

[5] MUIR S, SMITH J. Functional divisions in the piglet multiprocessor operating system[C]// Proceedings of the 8th ACM SIGOPS European Workshop on Support for Composing Distributed Applications. New York, NY, USA: ACM Press, 1998: 255—260.

[6] GOBLE G H, MARSH M H. A dual processor VAX 11/780[C]// Proceedings of the 9th Annual Symposium on Computer Architecture. New York, NY, USA: ACM, 1982: 291—298.

[7] KAGSTROM S, LUNDBERG L, GRAHN H. A novel method for adding multiprocessor support to a large and complex uniprocessor kernel[C]// Proceedings of 18th International Parallel and Distributed Processing Symposium. Washington, DC: IEEE Computer Society, 2004: 60—60.

[8] MUIR S, SMITH J. AsymOS— an asymmetric multiprocessor operating system[C]// Proceedings of Open Architectures and Network Programming. [S.l.]: IEEE, 1998: 25—34.

[9] SEO M, KIM H S, MAENG J C, et al. An effective design of master-slave operating system architecture for multiprocessor embedded systems[C]// Proceedings of 12th Asia-Pacific Conference. LNCS 4697. Berlin: Springer, 2007: 114—125.

[10] INDRESAN A. A distributed real-time operating system[J]. IEEE Software, 1992, 9(5): 58—68.

[11] FRIEDRICH L F, STANKOVIC J, HUMPHREY M, et al. A survey of configurable, component-based operating systems for embedded applications[J]. IEEE Micro, 2001, 21(3): 54—68.

[12] ZHU LI-XIN, WANG FEI-YUE. Component-based constructing approach for application specific embedded operating systems[C]// Proceedings of Intelligent Transportation Systems. Washington, DC: IEEE, 2003, 2: 1338—1343.

[13] GABBER E, SMALL C, BRUNO J, et al. The pebble component-based operating system[C]// Proceedings of the USENIX Annual Technical Conference. Berkeley, CA, USA: USENIX Association, 1999: 20—20.

[14] CARPENTER J, FUNK S, HOLMAN P, et al. A categorization of real-time multiprocessor scheduling problems and algorithms[C]// LEUNG J Y-T, ed. Handbook of Scheduling: Algorithms, Models, and Performance Analysis. Boca Raton, USA: Chapman Hall/CRC Press, 2004.

[15] LOVE R. Linux Kernel Development[M]. 2nd ed. USA: Novell Press, 2005.

[16] BACH M J. Design of the UNIX Operating System[M]. Englewood Cliffs: Prentice Hall, 1986.

[17] LABROSSE J J. μC/OS-II——源码公开的实时嵌入式操作系统[M]. 邵贝贝，译. 北京，中国电力出版社，2001.

[18] CLARK R, O QUIN J, WEAVER T. Symmetric multiprocessing for the AIX operating system[C]// Proceedings of the 40th IEEE Computer Society International Conference in Compcon. Washington, DC: IEEE Computer Society, 1996: 110—115.

[19] LIEDTKE J. On micro-kernel construction[C]// Proceedings of the 15th ACM Symposium on Operating Systems Principles. New York, NY, USA: ACM Press, 1995: 237—250.

（上接第 2669页）

4 结语

本文旨在帮助人类专家对签名样本真伪进行粗筛选或将未知类别缩小到一个范围。在认证主体中为每个签名者构造一个独立的分类器，针对认证和识别功能采用两套不同的训练集。采用信息融合手段，提高鉴别准确率，能对未知类别签名作拒识别处理（签名样本不在已知类别内）。当新增或减少签名类别时，只需添加或删除相应的子网络。试验证明签名识别效果不会随签名类别的增加而降低。本文提出的集成神经网络结构可以应用于其他多分类问题，根据实际需要，还可将子分类器中的神经网络换成决策树、KNN等其他分类方法。

参考文献：

[1] NAKANISHI I, SAKAMOTO H, NISHIGUCHI N. Multimatcher on-line signature verification system in DWT domain[C]// Proceedings of 2006 IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences. Oxford, England: Oxford University Press, 2006: 178—185.

[2] NANNI L, LUMINI A. A novel local on-line signature verification system[J]. Pattern Recognition Letters, 2008, 29(5): 559—568.

[3] FIERRREZ J, ORTEGE-GARCIA J, RAMOS D. HMM-based on-line signature verification: Feature extraction and signature modeling[J]. Pattern Recognition Letters, 2007, 28(16): 2325—2334.

[4] MOZAFFARI P, FAEZ K, FARADJI F. One dimensional fractal coder for on-line signature recognition[C]// Proceedings of the 18th International Conference on Pattern Recognition. Washington, DC: IEEE Computer Society, 2006: 857—860.

[5] PLAMONDON R, SRIHARI S A. On-line and off-line handwriting recognition: A comprehensive survey[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(1): 63—84.

[6] MAGID A, TOTMAN S R, WEISS A M. Picture thresholding using an iterative selection method[J]. IEEE Transactions on Systems, Man and Cybernetics, 1990, 20(5): 1238—1239.

[7] HU M K. Visual pattern recognition by moment invariants[J]. IEEE Transactions on Information Theory, 1962, 8(1): 179—187.

[8] 孙即祥. 模式识别中的特征提取与计算机视觉不变量[M]. 北京：国防工业出版社，2001.

[9] HARALICK R M, SHANMUGAN K, DINSREN I. Textural features for image classification[J]. IEEE Transactions on Systems, Man and Cybernetics, 1973, 3(6): 610—621.

[10] DEMPSTER A P. Upper and lower probabilities induced by a multivalued mapping[J]. Annals of Mathematical Statistics, 1967, 38(2): 325—339.

[11] MITCHELL T M. Machine Learning. 北京：机械工业出版社，2003.

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net