



(12) 发明专利申请

(10) 申请公布号 CN 103559166 A

(43) 申请公布日 2014. 02. 05

(21) 申请号 201310555626. 4

(22) 申请日 2013. 11. 11

(71) 申请人 厦门亿联网络技术股份有限公司
地址 361009 福建省厦门市软件园二期望海
路 63 号 402-502 单元

(72) 发明人 冯万健

(74) 专利代理机构 厦门市新华专利商标代理有
限公司 35203

代理人 朱凌

(51) Int. Cl.

G06F 15/167(2006. 01)

G06F 12/08(2006. 01)

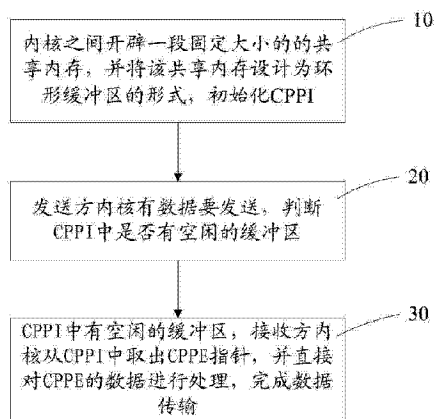
权利要求书1页 说明书3页 附图3页

(54) 发明名称

一种多核之间的高速数据传输的方法

(57) 摘要

本发明提出了一种多核之间的高速数据传输的方法。首先在内核之间开辟一段固定大小的共享内存,并将该共享内存设计为环形缓冲区的形式,初始化 CPPI,之后发送方内核有数据要发送,判断 CPPI 中是否有空闲的缓冲区,最后 CPPI 中有空闲的缓冲区,接收方内核从 CPPI 中取出 CPPE 指针,并直接对数据进行处理,完成数据传输。本发明能够减少数据拷贝及避免共享内存的数据读写冲突问题,在高数据率情况下提高数据的传输效率及传输稳定性。



1. 一种多核之间的高速数据传输的方法,其特征在于,包括:

步骤 10,内核之间开辟一段固定大小的共享内存,并将该共享内存设计为环形缓冲区的形式,初始化 CPPI;

步骤 20,发送方内核有数据要发送,判断 CPPI 中是否有空闲的缓冲区;

步骤 30,若 CPPI 中有空闲的缓冲区,接收方内核从 CPPI 中取出 CPPE 指针,并直接对数据进行处理,完成数据传输。

2. 如权利要求 1 所述的一种多核之间的高速数据传输的方法,其特征是:步骤 20 具体包括:

步骤 21,发送方内核有数据要发送;

步骤 22,发送方内核检查 CPPI 的写索引值进行自增或自减之后的值是否等于读索引值;

步骤 23,若不相等,发送方内核转载数据到 CPPE 并将 CPPI 的写索引值采用与所述步骤 22 一样的方式进行自增或自减,并通知接收方内核接收数据,否则返回步骤 21。

3. 如权利要求 1 或 2 所述的一种多核之间的高速数据传输的方法,其特征是:步骤 30 进一步具体为:

步骤 31,接收方内核接收通知,说明 CPPI 中有空闲的缓冲区,接收方内核从 CPPI 中取出 CPPE 的指针,并直接对数据进行处理,最后将读索引值采用与所述步骤 22 一样的方式进行自增或自减,数据传输结束。

4. 如权利要求 1 所述的一种多核之间的高速数据传输的方法,其特征是:所述初始化 CPPI 包括设置写索引值和读索引值的范围及初始值。

5. 如权利要求 1 所述的一种多核之间的高速数据传输的方法,其特征是:所述 CPPE 用于存放数据的地址指针及 BUF 的控制处理元素。

一种多核之间的高速数据传输的方法

技术领域

[0001] 本发明涉及无线通信领域,特别涉及一种多核之间的高速数据传输的方法。

背景技术

[0002] 在 VOIP 终端设备设计中经常会遇到有大量的网络音频数据需要实时传输,这种数据传输要求一般都比较,如稳定性、可靠性、低延时性都会比较苛刻。特别是在通信终端设备的 CPU 主频不高的情况下,这些音频数据在多核之间要求高速稳定的传输和存储,这无疑是一个困难的问题。

发明内容

[0003] 有鉴于此,本发明的目的是提供一种多核之间的高速数据传输的方法,能够提升系统的和速率性能。

[0004] 为了实现上述目的,本发明提出一种多核之间的高速数据传输的方法,其包括:

步骤 10,内核之间开辟一段固定大小的共享内存,并将该共享内存设计为环形缓冲区的形式,初始化 CPPI,所述 CPPI 为 Communications Port Programing Interface,即通讯端口程序处理接口;

步骤 20,发送方内核有数据要发送,判断 CPPI 中是否有空闲的缓冲区,所述 CPPE 为 Communications Port Programing Element,即通讯端口程序处理元素;

步骤 30,若 CPPI 中有空闲的缓冲区,接收方内核从 CPPI 中取出 CPPE 指针,并直接对数据进行处理,完成数据传输。

[0005] 进一步地,步骤 20 具体包括:

步骤 21,发送方内核有数据要发送;

步骤 22,发送方内核检查 CPPI 的写索引值进行自增或自减之后的值是否等于读索引值;

步骤 23,若不相等,发送方内核转载数据到 CPPE 并将 CPPI 的写索引值采用与所述步骤 22 一样的方式进行自增或自减,并通知接收方内核接收数据,否则返回步骤 21。

[0006] 进一步地,步骤 30 进一步具体为:

步骤 31,接收方内核接收通知,说明 CPPI 中有空闲的缓冲区,接收方内核从 CPPI 中取出 CPPE 的指针,并直接对数据进行处理,最后将读索引值采用与所述步骤 22 一样的方式进行自增或自减,数据传输结束。

[0007] 进一步地,所述初始化 CPPI 包括设置写索引值和读索引值的范围及初始值。

[0008] 进一步地,所述 CPPE 用于存放数据的地址指针及 BUF 的控制处理元素。

[0009] 本发明提出了一种多核之间的高速数据传输的方法,本发明避免了使用数据拷贝及共享内存的锁保护等效率不高的旧方法,同时也避免了数据流在多核之间的速度不均衡的情况发生。能够减少数据拷贝及避免共享内存的数据读写冲突问题,在高数据率情况下提高数据的传输效率及传输稳定性。

附图说明

[0010] 图 1 是本发明的多核间数据传输流程图。

[0011] 图 2 是本发明的多核间数据传输流程图。

[0012] 图 3 是本发明的多核间数据传输的示意图。

具体实施方式

[0013] 下面是对本发明一种多核之间的高速数据传输的方法进行具体说明。

[0014] 如图 1-3 所示,本发明一种多核之间的高速数据传输的方法的一种实施例方式包括:

步骤 10,内核之间开辟一段固定大小的共享内存,并将该共享内存设计为环形缓冲区的形式,这种设计方式可以使内存得到充分的使用,同时又不占用过多的内存,避免造成资源的浪费。并初始化 CPPI,所述 CPPI 为 Communications Port Programing Interface,即通讯端口程序处理接口;

步骤 20,发送方内核有数据要发送,判断 CPPI 中是否有空闲的缓冲区;

步骤 30,若 CPPI 中有空闲的缓冲区,接收方内核从 CPPI 中取出 CPPE 指针,所述 CPPE 为 Communications Port Programing Element,即通讯端口程序处理元素,并直接对数据进行处理,完成数据传输;

其中,步骤 20 具体包括:

步骤 21,发送方内核有数据要发送;

步骤 22,发送方内核检查 CPPI 的写索引值进行自增或自减之后的值是否等于读索引值;

步骤 23,若不相等,发送方内核转载数据到 CPPE 并将 CPPI 的写索引值自增或自减,并通知接收方内核接收数据,否则返回步骤 21;

其中,步骤 30 进一步具体为:

步骤 31,接收方内核接收通知,说明 CPPI 中有空闲的缓冲区,接收方内核从 CPPI 中取出 CPPE 的指针,并直接对数据进行处理,最后将读索引值做与写索引值相同的改变,数据传输结束。

[0015] CPPI 在初始化时会设置写索引值和读索引值的范围,例如最大值设置为 100,最小值设置为 0。在使用时,用一次,写索引值和读索引值就加“1”一次。当用到最大值 100 后就自动回归到最小值 0 开始使用,如此循环使用。采用 CPPE 接口方式,可以跨平台使用。

[0016] 例如:将这两个数据成员分别命名为 write_index 和 read_index。索引值的范围设计为 0—100。

[0017] 当 CPPI 在初始化时,会将 write_index 和 read_index 的值分别设置为 0。此时 write_index 与 read_index 的值相等,表示没有数据要接收,系统处于空闲状态。

[0018] 当发送方内核要发送数据时,将 write_index 的值加 1,因为是共享内存,所以接收方内核的线程会侦测到 write_index 与 read_index 不相等了,即 write_index = 1, read_index = 0,说明有数据要接收了。接收方内核会根据 read_index 值取出共享内存中相对应的数据(也就是内存区块 0 的数据),然后将 read_index 的值加 1。这时 write_index

与 read_index 的值又相等了,即 write_index = 1 , read_index = 1,系统又处于空闲状态。完成了一次数据传输。

[0019] 如果发送方内核发送数据太快了,而接收方内核接收数据比较慢,那么 write_index 的值可能为 100 了,而 read_index 的值才 90,这时 write_index 的值自动回归到 0,并将旧数据(即内存区块 0 的数据)覆盖了。因为 read_index 的值已经为 90,说明接收方内核在接收内存区块 90 的数据,而内存区块 0 的数据早已接收完成,所以覆盖内存区块 0 的数据也就不会有影响了。最终实现了循环使用内存的目的。

[0020] 与现有的技术相比较,本发明提出了一种多核之间的高速数据传输的方法,本发明避免了使用数据拷贝及共享内存的锁保护等效率不高的旧方法,同时也避免了数据流在多核之间的速度不均衡的情况发生。能够减少数据拷贝及避免共享内存的数据读写冲突问题,在高数据率情况下提高数据的传输效率及传输稳定性。

[0021] 以上所述仅是本发明的优选实施方式,应当指出,对于本技术领域的普通技术人员来说,在不脱离本发明原理的前提下,还可以作出若干改进,这些改进也应视为本发明的保护范围。

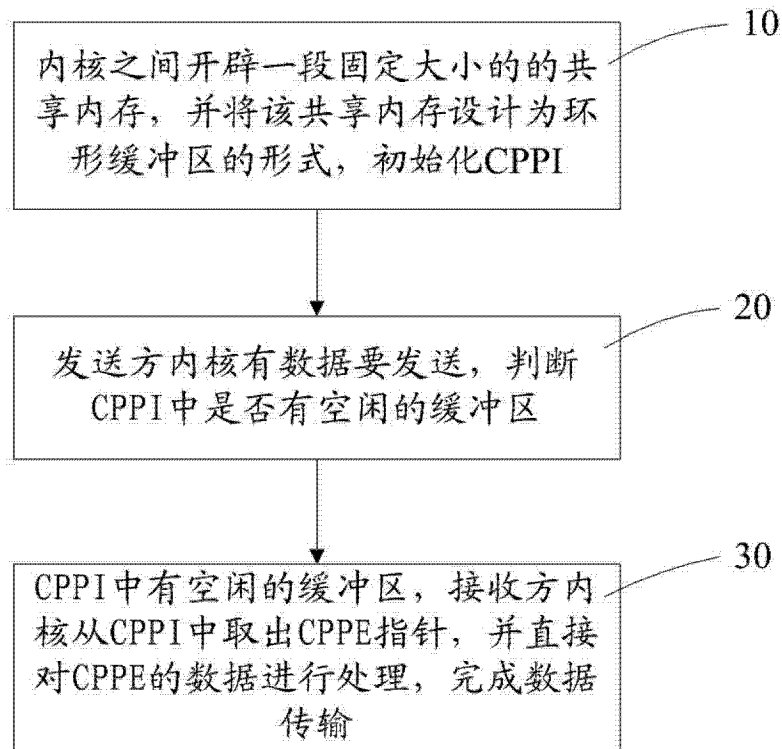


图 1

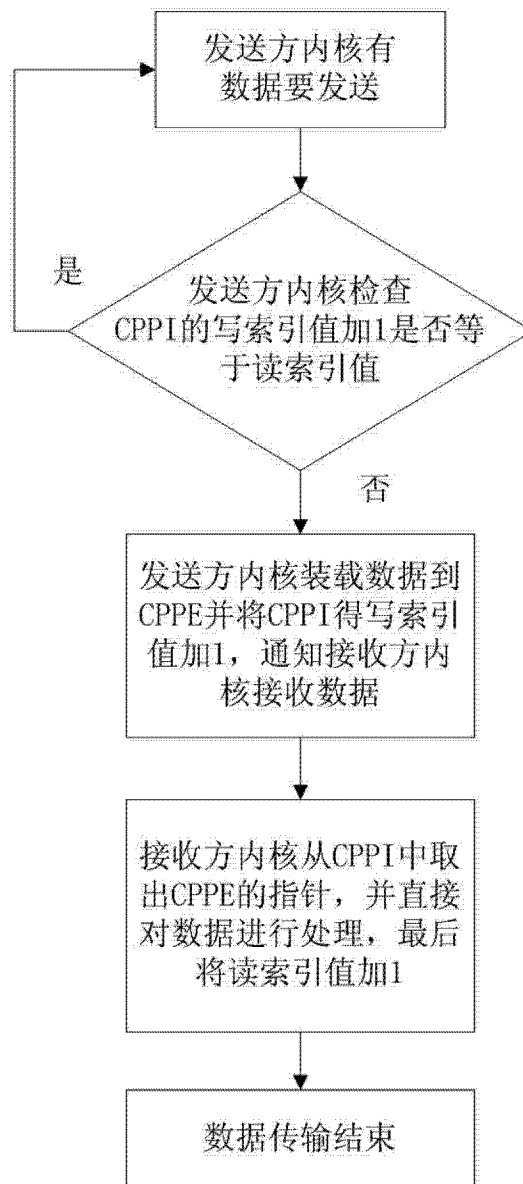


图 2

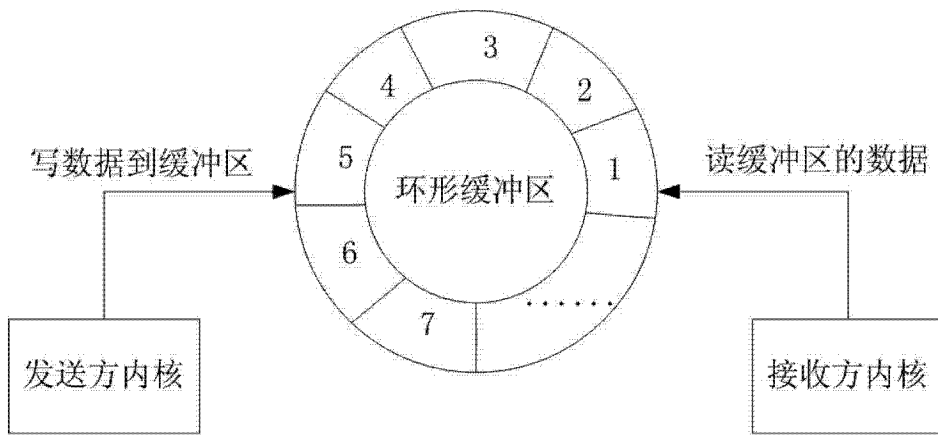


图 3