

一种嵌入式平台下提高 SCTP 偶联带宽利用率的方法

A Method to Improve SCTP Association Bandwidth Utilization

黄东明 郑善贤 (湖南大学 电气与信息工程学院 湖南 长沙 410012)

摘要: 提出一种在平均模式下,提高 SCTP 偶联带宽利用率的方法及其实现方式。该方法使用共享带宽策略,对偶联带宽进行了集中的管理和分配。通过这种策略,突破了嵌入式系统资源限制对偶联产生的影响,并同时提高了整个带宽资源的利用率。

关键词: 嵌入式系统 SCTP 偶联 带宽

当前,随着 IP 技术的发展,逐渐克服了传输过程中遇到的 QoS 问题,而且相比 ATM 传输,IP 具有网络资源利用率高、运营简单、节省成本、组网灵活等优点,使得越来越多的网络开始向全 IP 方向发展^[1]。

流控制传输协议(Stream Control Transmission Protocol, SCTP)^[2,3]是一种传输层协议,它在两个端点之间提供可靠、稳定的数据传递服务。SCTP 发展了 TCP 和 UDP 两种协议的长处,一方面增强了 UDP 业务,提供了数据报的可靠传输;另一方面,继承了 TCP 的诸多优点,并提出了多宿主、多流等全新概念,增强了数据传输的可靠性,因而成为可靠 IP 网络传输的首选^[4]。SCTP 建立的连接称作偶联,数据传输都是承载在偶联之上的,因此偶联的设计指标成为了整个 IP^[5]传输网络的关键。目前,SCTP 主要是在传输要求比较高的通信网络中应用。然而,通信网络多为嵌入式系统,实现 SCTP 协议时往往会受到存储空间等资源的限制,反应在具体的偶联上就是带宽资源的争夺,那么如何才能有限资源的条件下,做到偶联带宽的最大化利用呢?

由此,本文提出了一种在嵌入式平台下提高带宽利用效率的实现方法,用来克服嵌入式系统的带宽限制。

1 方案设计与实现

1.1 设计思路

IP 网络的一个重要指标就是传输带宽,而 SCTP 的传输带宽主要体现在存储区的利用上,那么相对于存储资源有限的嵌入式系统而言,这部分存储带宽的利用将直接影响系统的性能。

SCTP 在建立偶联的时候需要分配一定的带宽,当有多条偶联在系统上运行时这部分有限的带宽该如何分配呢?比较简单的实现方案就是系统启动时为每条偶联预留固定带宽,偶联运行时直接占用这部分带宽。然而这种方案会带来一个问题,系统的偶联容量受到了限制(偶联数=总带宽/单条偶联带宽),不利于系统的扩展。而且在设计之初也无法准确估计系统所需偶联数目,如果估计少了就会造成带宽资源的浪费;估计多了又会造成数据传输的带宽限制。由此可见设计的主要问题集中在带宽的合理分配和系统的扩展性上。

这里提出一种带宽均分剥夺的思想,让所有偶联既对等的获取带宽同时也对等的剥夺带宽。思路如下:为系统设计一共享带宽区,在偶联数较少的情况下,带宽资源比较充足,每条偶联将从共享区分配到足够的带宽(即满配带宽),让其带宽限制降为最小。当偶联数目较多时,带宽资源有限,但是每条偶联都必须

① 收稿时间:2008-11-23

投入服务,设计采取的方法是平均策略,共享带宽对等的进行再分配,此时偶联带宽计算如下:偶联带宽=总带宽/偶联数。多偶联情况下系统共享带宽可能已经分配完了,那么新偶联的带宽从何而来呢?设计使用了剥夺式方法,思想是剥夺已有偶联的空闲带宽并重新分配给新偶联,从而使那些空闲的带宽也能得到有效的利用。剥夺方案可以实现系统的扩展性,偶联的数目可以自由的增长,但是最终会受到最小带宽的限制(即没有空闲带宽可以剥夺)。

反之,当某条偶联释放后,如果其它偶联的带宽不是满配,则会将释放后的带宽资源平均返还给每条偶联,每条偶联所得带宽=释放的偶联带宽/当前偶联数。

设计的偶联带宽利用方法如下图所示:

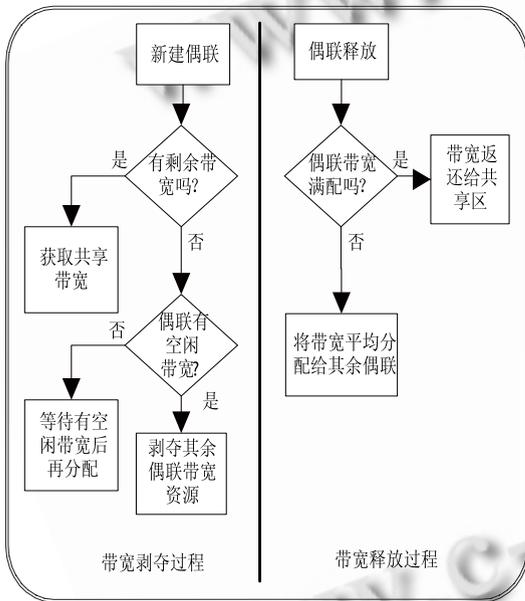


图 1 偶联带宽的分配和释放过程

1.2 偶联带宽均分剥夺策略的设计和实现

针对上述设计思想,需要在不破坏当前偶联数据传输的情况下,实现带宽资源的剥夺。这里主要涉及两部分功能,共享带宽区的管理和偶联带宽区的管理。

1.2.1 共享带宽区的设计。

共享带宽区是所有偶联共享的,系统启动时将总带宽分配给 SCTP,大小可根据实际系统条件设定的,

假定为 $(\text{Sum_BandWidth})M$ 。为了提高分配效率,设计将共享区按块划分,每块大小为 `chunk_unit`,系统分配带宽时是以块为单位进行的,则共享区可分配的总块数 $N = \text{Sum_BandWidth} / \text{chunk_unit}$ 。

共享区使用一链表结构来管理,称为空闲列表,表单元指向未分配的共享区块结构。分配和释放带宽时都从表尾开始操作,依次增加或删除 `pFreeTail` 单元。共享区管理结构带宽分配实例如图 2 所示, `PFreeHead=NULL` 时表明共享区剩余带宽分配完。

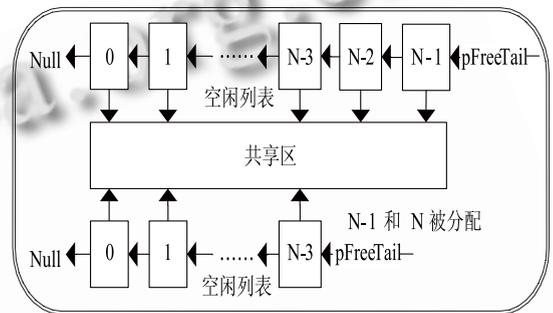


图 2 共享区结构和带宽分配过程

1.2.2 偶联带宽区的设计

偶联带宽区指向共享区,是共享区的一部分,用于偶联数据的收发缓存。为了防止偶联数据的串扰,设计每条偶联都独占一个带宽区。而对偶联带宽区的管理主要实现的是带宽资源的获取和释放以及数据的发送和接收。为了满足上述要求,设计使用了环形队列来管理偶联带宽区。

环形队列设计首先要考虑队列的大小以及队列与带宽的关系问题。设计的最终目的是要实现带宽的剥夺,但是在带宽资源足够的情况下,显然要尽量满足偶联的带宽需求,即偶联有一个足够使用的最大带宽,假设为 $(\text{Max_AssocBW})M$ 。由上知,带宽的分配是以块为单位的,那么一条偶最大块数 $\text{assoc_max_chunk} = \text{Max_AssocBW} / \text{chunk_unit}$ 。由此设计了队列 `pQueue[assoc_max_chunk]` 形成一个如下图(假设 $\text{assoc_max_chunk} = 16$)逆时环形结构,每个 `pQueue[]` 需要向共享区申请才会获得带宽,大小和共享区的块结构一致。

环形队列需要完成带宽的获取和释放,设计使用

了队尾指针 $pTail$, 队头读指针 $pRead$, 写指针 $pWrite$ 来实现这部分功能, 具体的操作过程如下。下图为带宽的强行剥夺过程, 假设满配块数为 16, 其中标数字的块表示带宽已分配, 空白部分则表示还未分配。

1) 计算可分配的带宽, 假设当前偶联数为 $AssocNo$, 则按平均策略新偶联需要的带宽为 $Sum_BandWidth/(AssocNo+1)$ 。

2) 判断是否有剩余带宽, 有则从共享空闲队列中获取, 否则剥夺其余偶联的带宽。按均摊原则, 每条偶联需强制剥夺的带宽为 $Sum_BandWidth/(AssocNo+1)/AssocNo$

3) 获取到的带宽按逆时针方向加入队列尾部, $pTail$ 逆向后移。如果 $pTail=pRead$ 则表示队列的带宽满配。

4) 带宽的剥夺和获取过程相反, 从队尾 $pTail$ 顺时针方向释放, 假如 $pTail=pWrite$ 则表示偶联没有空闲带宽了。

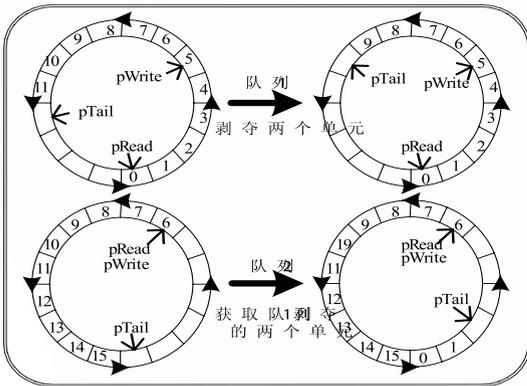


图 3 环形队列实现带宽强制剥夺

1.3 偶联数据收发实现

SCTP 提供数据的可靠传输, 可以实现数据的分片重组、SACK 确认、顺序传输等机制。因而在 SCTP 的实施过程中, 必然要用到重组队列和失序队列。而且实现采用带宽剥夺, 可能会使环形队列带宽不满配, 为了不影响数据传输, 必须做队列迁移。参照图例, 环形队列、重组队列、失序队列协调完成数据传输的过程如下所示:

1) 偶联带宽分配完成后, SCTP 向上层上报偶联建立成功。上层利用该偶联开始发送数据。

2) 数据发送。首选判断发送队列是否已满, 如果满则通知上层发送失败。再者判断数据是否需要分片, 如果是则先将数据进行分片处理。构造 DATA 块结构, 将需要发送的数据放入 DATA 净荷, 然后将 DATA 存入发送队列等待发送。发送模块发送数据时, 构造 SCTP 的 packet 结构, 而后通过 IP 层将数据发送给对端。这个过程中写数据和读数据都要做队列调整, 写数据时需要获取 $pWrite$ 所在的空闲块, 然后将数据逆时存入队列。 $pRead$ 指向数据存储区, 发送数据时通过 $pRead$ 获取存储的 DATA。当发送的数据得到了对端的接收确认, $pRead$ 所在的 DATA 区需要清空。此时, 为了防止队列带宽空间压缩, 在不满配的情况下, 需要做队列的迁移。采取的措施是首尾相补法, 将队头清空的块直接补到 $pTail$ 区, 图例是将 0 和 1 的带宽直接赋给 12 和 13。

3) 数据接收。顺序传输的情况下, 如果收到的数据失序, 则放入失序队列, 等待对端重发丢失的数据。如果收到的是分片数据, 则放入重组队列, 等待重组完成后存入接收队列。上层读取接收队列中的数据, 如果接收队列带宽不是满配, 需要做队列迁移, 方法同发送队列。

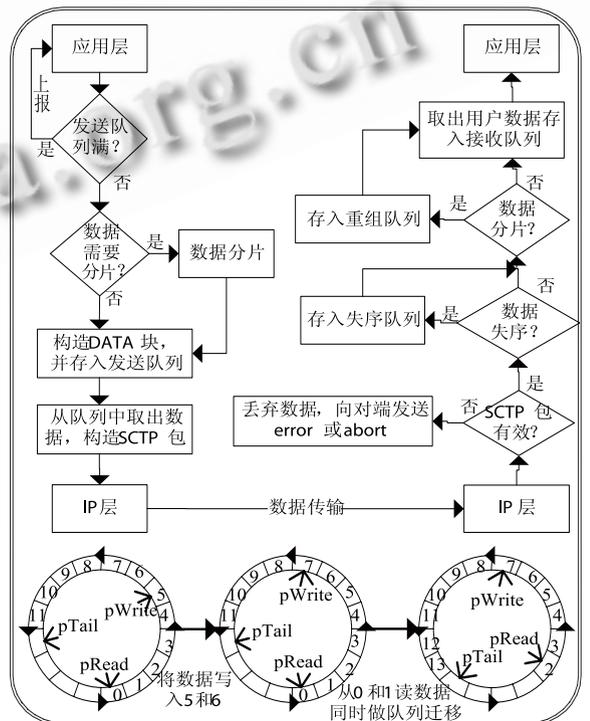


图 4 SCTP 的数据收发及队列的迁移过程

2 测试

测试是借助 3G 传输平台进行的,在两台 RNC(无线网络控制器, Radio Network Controller)实验机上实现 SCTP 协议,建立 Iur 口的偶联连接。

测试方法如下:在两台 RNC 之间建立大约 40 条偶联,一台 RNC 上实现模拟发送程序,另一台上实现模拟接收程序。发送程序同时向 40 条偶联做数据灌包,数据包的大小(如 128, 512, 1500, 1628 等)和发送频率(如 10 包/s, 100 包/s 等)依据测试规程做调整。接收端主要负责接收包的数目统计。通过分析偶联的运行状态、cpu 占用率,丢包率等参数便得出设计的方案在系统上运行的性能。

测试表明,在无剩余带宽的情况下新偶联能够完成带宽的剥夺,并成功建立,偶联释放时其余偶联也能成功完成带宽的增长。同时数据发送的时延、吞吐量、以及丢包率等均能达到系统运行的要求。

3 结语

本文结合具体的工程实际需要,在嵌入式 © 中国科学院软件研究所 <http://www.c-s-a.org.cn>

实现了一种 SCTP 偶联带宽有效利用的方案,突破了嵌入式系统存储资源对偶联数目的限制,提高了系统的扩展性。同时基于空闲带宽剥夺的思想,使带宽资源得到了更加充分的利用。测试表明系统运行稳定,具有很好的扩展性,满足实际应用的性能需求。

参考文献

- 1 龚双瑾,刘多.下一代电信网的关键技术.北京:国防工业出版社,2003.
- 2 Stream Control Transmission Protocol.IETF RFC2960, 2000.
- 3 中华人民共和国信息产业部发布.流控传输协议(SCTP).YD/T 1194-2002.
- 4 中华人民共和国信息产业部发布.NO.7 信令与 IP 互通的技术要求. YD/T 1127-2001.
- 5 IETF draft.Stream Control Transmission Protocol (SCTP)Dynamic Address Reconfiguration, March