doi:10.3969/j.issn.1000-436X.2012.10.016

交换结构中的可重构缓存机制

黄慧群[1,2]，刘勤让[1,2]，卜佑军[1,2]，张风雨[1,2]
（1. 解放军信息工程大学 信息工程学院，河南 郑州 450002; 2. 国家数字交换系统工程技术研究中心，河南 郑州 450002）

摘 要：大量研究集中在路由器对缓存区容量设置的理论需求上，可重构缓存则借助可重构系统按需分配的思想，试图从实现机制上提高缓存利用率以改善系统性能。现有固定缓存的交换结构，在大量缓存区空闲时仍可能有某些端口因较大突发流量而大量分组丢失。为此引入缓存的重构机制，打破端口对缓存单元的私有独占，按各端口的实际缓存需求量来实时重构，分析和实验结果表明，运用该机制可有效解决缓存资源浪费现象，在获得同等抗突发性能的条件下，节约大量存储单元，大大提高路由器交换系统中存储单元的利用率。

关键词：交换；可重构；缓存；FPGA
中图分类号：TP393 **文献标识码**：A **文章编号**：1000-436X(2012)10-0126-06

Reconfigurable buffer mechanism in switch fabric design

HUANG Hui-qun[1,2], LIU Qin-rang[1,2], BU You-jun[1,2], ZHANG Feng-yu[1,2]
(1. Information Engineering Institute of Information Engineering University of PLA, Zhengzhou 450002, China;
2. National Digital Switching System Engineering & Technological R&D Center, Zhengzhou 450002, China)

Abstract: Current research focused on the theoretic buffer size of routers. The reconfigurable buffer tried to improve the buffer utilization rate to optimize the system performance based on the on-demand design of the reconfigurable system. In fact, for current switching fabrics with fixed buffers, cell loss under burst traffics still could not be resolved at some ports despite abundant buffers were free. The reconfiguration mechanism was introduced into buffers, which broke the private occupancy of buffers by single ports and reconfigured buffers due to the real-time requirements of per port. Analysis and simulation results show that the reconfiguration mechanism could resolve the problem of resource consumption effectively, and improve the ultilization of buffers as well as save buffers with the same anti-burst performance.

Key words: switch; reconfigurable; buffer; FPGA

1 引言

基于分组的交换系统是实现路由器交换和大规模系统扩展的重要单元，交换结构按照排队缓存所在位置的不同，可分为输入排队（IQ，input queue）和输出排队（OQ，output queue）。输出排队交换结构中，到达各输入端口的分组直接经由交换单元到达输出端口，并在每个输出端口进行排队，如图 1 所示。输入排队则将到达分组在输入端口进行排队和缓存，经调度后由交换矩阵交换到系统输出端口[1,2]。文献[3]从分组丢失率、链路速率等性能需求入手，研究了核心路由器的缓存需求问题，并给出了核心路由器缓存设置法则。文献[4]重点针对路由器内部交换中采用简单 FIFO 算法产生的线头（HOL, head of line）阻塞现象，结合虚拟输出队列（VOQ, virtual output queue）机制，给出了一种消除数据转发过程中出现线头阻塞的 iSLIP 改进算法。文献[5]基于 TCP 协议模型对经验法则、斯坦

收稿日期：2011-11-30；**修回日期**：2012-04-18
基金项目：国家重点基础研究发展计划（“973”计划）基金资助项目（2012Cb315901）
Foundation Item: The National Key Basic Research Development Program(973 Program) (2012Cb315901)

福缓存法则和基于分组丢失率的缓存法则等各种缓存容量设计研究成果进行了比较分析。文献[6~10]给出了可重构路由器软硬件模型、ForCES 协议及体系结构方面的介绍。

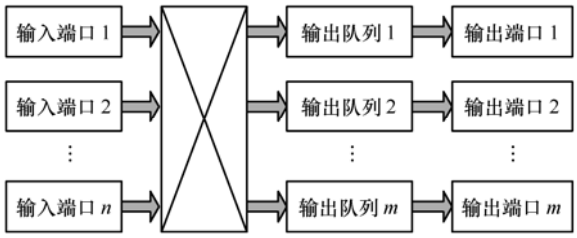


图 1　输出排队交换结构

大量研究集中在路由器对缓存区容量设置的理论需求上。可重构路由器则注重研究和实现有限资源高效利用的方法，也是可重构缓存机制研究的出发点。该机制在交换结构缓存队列中引入按需分配思想，将队列中闲置存储单元分配给有较大突发流量、需要更多缓存的端口，避免系统有大量缓存区空闲时仍有一些端口因较大突发流量而大量分组丢失。该机制可普遍适用于输入排队和输出排队 2 种交换结构，为使问题分析更加直观，以输出排队的交换结构为例。

从任一个输入端口进入交换矩阵的数据分组，其输出方向可能是输出端口 1~m 中的任意一个。设输出端口 i 的带宽是 P_i，当输出方向为 i 的输入分组的总和小于 P_i 时，所有该类数据分组均可无阻塞地送出；但是，事实上，由于网络中数据分组的随机性，目的端口为 i 的分组的总流量在某些时间段可能会大于其输出带宽。此时，则需要将未能及时送出的分组缓存在输出队列中。

如果目的端口为 i 的数据流持续大于其输出带宽，则输出队列中缓存数据持续增长，当超过其队列长度时，最终分组丢失无法避免。但对于突发的短时流量，只要在一个时间段内需求总带宽小于 P_i，输出缓存将暂时未能送出的数据缓存，则有效地避免了分组丢失。显然，输出缓存区的大小与其抗突发流量的能力成正比，即缓存区越大，抗突发能力越强。同时，由于突发流量的发生不局限于每一个输出端口，因而，只好将每一个缓存区均设置为一个较大的值，以此来防御可能发生的突发流量。

但是，事实上，相对于输出到某一个端口的突发流量，一个交换矩阵输入的总流量是固定的，则输出到所有端口 i 的流量的总和等于输入流量（且不大于输入带宽），这样，当某一个输出端 P_i 有突发大流量时，相对地，送到其他输出端 $P_j(j \neq i)$ 的流量就会变少，其需要的缓存区也就较小，而缓存区域是事先设置好的，这样，P_i 端口之外的其他缓存区大部分处于闲置状态。由于这个 P_i 是随机的，不可能预知从而减小其他输出端缓存的容量，系统设计时就必须为每一个输出端均设置一个较大的缓存区，从而造成巨大的存储资源浪费。

为削减这些闲置存储单元，本文为交换系统提出一种可重构缓存，首先将固定分配给每个输出端口的存储单元“公共化”，然后，根据系统实际流量特征以及相应的需求，将公共存储单元按需分配，实现了存储单元的实时重构，避免在大量缓存区空闲时仍有一些端口因较大突发流量而大量分组丢失，从而大大提升了资源利用率。

本文后面章节将基于可重构缓存的基本原理，深入探讨关键技术，最后给出具体实现方案以及性能分析和实验验证结论。

2　可重构缓存及调度算法

2.1　可重构缓存交换结构

基于 FPGA 实现的交换结构中，缓存区由 FPGA 内部的 BlockRAM 组成，Xilinx 公司的 FPGA 产品基本存储单元均为 18kbit 的 RAM[11]，Altera 公司 FPGA 内部的存储单元则主要是 4kbit 或者 512 bit 的 RAM 块。为提高抗突发能力，每个缓存区均需多个 BlockRAM，该存储区大小是固定的。

可重构缓存的目的是在同等抗突发能力的前提下，尽量减少缓存队列实际所需的存储资源数目，节省 FPGA 内部宝贵的存储资源。其基本指导思想是打破存储区依不同端口而专门设置的私有模式，通过引入大容量的公共缓存实现主要存储资源的按需分配。具体方法是，为每个端口设置一个较小的基础缓存，该缓存主要用于完成数据处理、输出控制等功能，属于各端口的“私有缓存”；同时将大量的存储资源块设置为公共缓存区，当流向某一个输出接口的突发流量超过其基础缓存时，向公共存储区申请得到更多缓存资源。

该机制的合理性在于，由于单播交换系统输入的总流量不大于其总带宽，当某个或者某些输出端口有大量突发流量时，其他端口流量则较小，只有

那些具有突发流量的端口申请到公共缓存区，流量较小的端口则不需要大缓存区，以此实现缓存区的按需分配，使得缓存资源得以高效利用。

可重构缓存对应的交换结构如图 2(a)。每一个输出端口的缓存队列均包括缓存调度、基础缓存以及输出控制 3 个基本单元，公共缓存块(Gbuffer, global buffer)可被每一个缓存队列调用。其中，缓存调度决定将待进入输出排队队列的分组写入哪个缓存块，输出控制则根据缓存重构的结构来控制读缓存顺序并将数据选择输出。

如图 2(b)所示，每个基础缓存(Bbuffer, basic buffer)由 2 部分组成，即：数据缓存区以及缓存结构指示链 FIFO，前者为本端口的数据缓存区，后者则指明本端口的输出缓存顺序地由哪些缓存块组成，用作数据输出时的读顺序控制以及数据复接时的使能。

2.2 缓存重构调度算法

缓存调度是可重构缓存的核心控制单元，本文给出一种对输出端口公平、对公共缓存块预设优先级的缓存重构调度算法。一个重构的缓存队列（也可视作一个逻辑缓存队列）由 1 个基础缓存和多个公共缓存块组成，当其中 2 个以上可选择调用时，选择最高优先级缓存块。所有缓存块的优先级预先设定，并应遵循以下 2 条原则：

1) 与公共缓存相比，基础缓存总是具有高优先级，以使尽量多的公共缓存块用于需要的地方；

2) 公共缓存块的优先级，可按照易于硬件实现的升序或者降序来指定。

为算法描述方便，对于任意一个逻辑缓存队列 X，定义其本地位置指针 LSP_X 为：已写入最新数据所调用的缓存块的编号。位置指针可能指向本地基础缓存或者所有公共缓存，因而二者统一编号，记为 $Buffer\_Num$，以−1 表示本地基础缓存，$0 \sim (n-1)$ 为公共缓存编号，其中，n 为公共缓存块的个数。

预设优先级的缓存重构调度算法如下。

初始时，LSP_X 设为−1。

运行过程中，每来一个新的目的端口为 X 的分组，则：

首先读取逻辑队列 X 的当前位置指针 LSP_X，并判断该缓存是否已满，若未满，则将新到来数据写入；同时，判断 LSP_X 指向的缓存块是否为空，若是，则向缓存指示链 FIFO 中写入 LSP_X，否则不变。

当检测到某个缓存块变为满时，则判断是否有其可调用的空闲块，若无，则将该分组的后续数据丢弃；若有，则找到优先级最高的空闲缓存块（设其 $Buffer\_Num=Y$）分配给队列 X，写入分组数据；同时，向该 X 端口的缓存结构指示链 FIFO 中写入 Y，并更新位置指针，使得 $LSP\_X=Y$。

在接口数据选择输出端，则根据缓存结构指示链 buffer 来控制读顺序。

初始状态：等待指示链 FIFO 变为非空时，并从中读一个数，即−1，并读基础缓存。

运行过程中：当检测到当前缓存块读空时，若指示链 buffer 未空，则从指示链 FIFO 中读取下一个缓存指示，然后根据该指示使能相关存储块的读控制，并将该存储块的输出送出。

2.3 抗突发性能与缓存区容量的关系

每个端口的输出带宽相等，均为 P，输入端口总带宽为 F。目的端口为 i 的流量速度记为 F_i。并定义端口 i 的突发流量 BF_i 为一段时间 Δt 内超出输出端口 i 输出带宽的流量的总和，记为 $\int (F_i - P)\mathrm{d}t$。则系统在 Δt 内的突发流量为 $\sum_i BF_i,\ i = 1 \sim M$。

设系统中每个可调用的缓存块的大小一致均为 C bit，则缓存容量可用缓存块的个数来表示。设每个基础缓存被分配 K 个块，公共缓存被分配 L 个块，则系统所用总存储块为 $S = MK + L$。

考察以下 3 种情形。

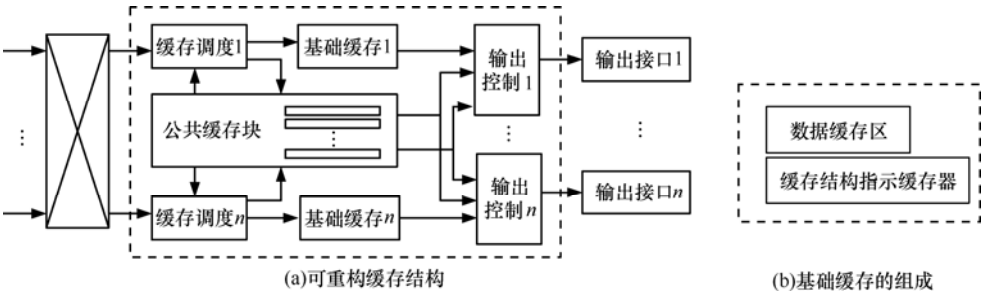


图 2 可重构缓存结构及基础缓存的组成

1) 单个输出端口的重构队列获得最大突发流量的情形：只有一个输出端有突发流量，当其基础缓存以及所有公共缓存块均满时达到最大值

$$\max(BF_i)=(K+L)C=[S-(M-1)K]C \quad (1)$$

2）系统容纳最多突发流量的情形：只有一个输出端流量小于输出带宽 P，其他 $M-1$ 个端口均有大于 P 的流量，则当这 $M-1$ 个端口的基础缓存以及所有公共缓存块均为满时，系统容纳突发流量达到最大值，此时

$$\max(\sum_i BF_i)=\left[(M-1)K+L\right]C=(S-K)C \quad (2)$$

3) 系统可容纳突发流量最少的情形：考虑某一时刻 t 开始，有流向端口 j 的突发流量 *Flow_j*，并且此时所有公共缓存块均已被 t 时刻之前的突发流量占用，则 *Flow_j* 进入基础缓存区 Bbuffer_*j* 等待，在 Bbuffer_*j* 变为满之后，若还没有可用的 Gbuffer，则后续流量将被丢弃直到有被释放的 Gbuffer 为止。t 时刻之前被占用的 Gbuffer 中，最先被释放的是 $\max_{i\neq j,i\in[1,m]}(P-F_i)$ 对应的 F_i 在时刻 t 被调用的存储块。

因：$F_1+F_2+,\cdots,+F_m\leqslant F=MP$

故有：$F_j-P\leqslant\sum_{i\neq j,i\in[1,M]}(P-F_i)$

等式右边 $M-1$ 项中最大的那一项将首先释放缓存块从而使得 *Flow_j* 获得可用缓存，但最坏情况下，这 $M-1$ 项均相等，即有

$$\max_{i\neq j,i\in[1,M]}(P-F_i)=(F_j-P)/(M-1)$$

且 t 时刻所有端口 $i=\{i\mid i\neq j,i\in[1,M]\}$ 的输出队列当前正输出的是其基础缓存区的数据，则为获得第一个空闲 GBuffer，*Flow_j* 在基础缓存满之后需要等待的时间为

$$\max(T_{\text{wait}})=\frac{(K+1)C}{\max_{i\neq j,i\in[1,M]}(P-F_i)}-\frac{KC}{F_j-P}=\frac{(M-1)(K+1)-K}{F_j-P}C$$

$$=\frac{(M-2)K+1}{F_j-P}C \quad (3)$$

在 $\max(T_{\text{wait}})$ 到来之前的一刻，系统获得最少可容纳突发流量

$$\min(\sum_i BF_i)=\left[K+L-M+1\right]C=[S-(M-1)(K+1)]C \quad (4)$$

考察式(1)、式(2)和式(4)，K 越小则突发流量容纳性能越强；根据式(3)，K 越小，最坏情形下分组丢失时间则越短。当 K 达到最大值 $K=S/M$ 时，即为不使用重构缓存的情况。

因实际工程实现需要，本文选取 K=1。

3　性能分析以及算法优化

3.1　算法实现以及硬件资源分析

以 Xilinx 公司的 FPGA 产品系列为例，其基本存储单元为 BlockRAM，每一个均为 18kbit。通过上述分析可知，基础缓存区只要不小于一个公共缓存块的容量，其本身的大小对于抗突发流量性能无影响，因而基础缓存只需一个 BlockRAM 即可。

输出接口缓存区结构指示 buffer 只需要按顺序记录当前该接口占用的公共存储单元序号或者自有基础缓存，若公共存储单元为 N 个，则需区分 $N+1$ 个缓存块，需 $\log(N+1)$ bit，并且其深度不超过 $N+1$，因而容量为 $(N+1)\log(N+1)$ bit，当 $N=15$ 时，该值为 64bit，可以用分布式 RAM 来实现。

若实现一个 4×4 的重构缓存队列 (RQ, reconfigurable queue)交换系统，则需要的 FPGA 资源总数为。

1) 输出控制部分。4 路输出缓存，每一路占用的资源为：一个 18bit 的 BlockRAM，用作基础缓存区，一个分布式 RAM 用作接口缓存区结构指示 buffer；以及一个数据复接电路。

2) 缓存调度部分，具体到 FPGA 实现中。事实上是 N 个公共缓存块以及 4 个基础缓存块的写控制电路。

本文在现有的 4×4 交换系统上实现了上述可重构的缓存。综合结果显示，上述逻辑电路占用的资源很有限，不足 1 000 个 LUT，而这些资源却是 FPGA 所富余的。FPGA 内紧张的存储资源，却可用于提高系统性能。本系统中，设置的公共缓存块个数为 8。

3.2　性能测试与算法优化

使用固定缓存的交换系统 FQ(fixed queue)，所使用缓存块总数与 RQ 相等，其输出队列长度为 3 个 BlockRAM，即 54kbit。依据可重构缓存算法升级的交换系统，已应用到自主研发的可重构路由器中。为便于比较二者的抗突发性能，本文设计了如下测试方案。

采用安捷伦测试仪作数据源，4 个端口卡 Port1~Port4 的输出连接路由器 4 个线卡的输入，交换系统的 4 个输出则分别送回给 4 个端口卡，实现闭环。测试仪各端口设置如下。

表 1　　Port1 4 个突发数据流的参数设置

Bflow	目的端口	突发长度(packet)	分组速率(packet/s)	最大带宽
Bflow1	Port2	40	1 815 272.73	10%
Bflow2	Port3	50	1 815 272.73	10%
Bflow3	Port4	110	1 815 272.73	10%
Bflow4	Port4	120	1 815 272.73	10%

表 2　　单端口突发流量时的分组丢失率统计

系统	序号 1		序号 2		序号 3		序号 4	
	Flow1	Bflow1	Flow2	Bflow2	Flow3	Bflow3	Flow4	Bflow4
RQ	0	0	0	0	0	0	0.002	0.017
FQ	0	0	0.002	0.020	0.021	0.203	0.024	0.250

对 Port1，设置 6 个突发数据流，参数设置如表 1 所示。各数据流的五元组均不相同，并通过路由器对转发表的配置来规定其目的输出端口。突发流的模式为周期性突发，分组长为固定的 128byte，突发长度以及分组速率设置如表 1 所示。

对于 Port2~Port4，分别设置数据流 Flow1~Flow3，对应的目的端口分别为 port2~Port4，分组长均为均匀分布，带宽均为 90%。

表 2 反映了系统容纳单端口突发流量的能力对比。在 flow1~flow3 同时发送的条件下，表中序号 1~4 分别表示突发流量为 Bflow1~Bflow4 的情形。与 FQ 相比，重构缓存的交换系统，容纳单端口突发流量的能力大大增加。

将 Bflow2 和 Bflow3 设置为与 Bflow1 相同，同时发送 flow1~flow3 以及 Bflow1~Bflow3，即 3 个端口均有突发流量时的分组丢失率如表 3 所示。表 3 分别给出了突发长度为 40 个分组和 50 个分组时的统计数据，由于这 3 组数据流的分组丢失率相同，表中不再区分各数据流。由表可见，容纳多端口突发流量的能力 RQ 比 FQ 仍略有改善。

表 3　系统容纳多端口突发流量时的分组丢失率统计

系统	序号 1(*size*=40)		序号 2(*size*=50)	
	Flow	Bflow	Flow	Bflow
FQ	0	0	0.002	0.020
RQ	0	0	0.002	0.017

由于存储单元数目毕竟是有限的，当超过其存储能力时分组丢失率将瞬时大大增加，造成链路的不稳定。

4　结束语

通常，交换系统通过增加输出缓存队列容量的方式来获得更好的容纳突发流量性能，为每个输出队列设置一个固定的大缓存区将耗费大量宝贵的存储资源。然而，由于系统输入总带宽有限，在某些端口有突发流量需要大量缓存、并因缓存不足而大量分组丢失时，另外那些端口则流量不足，设定的缓存处于闲置状态，造成严重的资源浪费。

目前，在路由器缓存区设置的问题上，大量研究集中在容量与性能的理论分析，而未有试图从实现机制上减少资源浪费来提高系统性能。可重构缓存机制从提高资源利用率的角度出发，在以 FPGA 实现的交换结构中引入缓存块的按需分配思想，打破交换结构中端口对缓存单元的私有独占，按各端口的实际缓存需求量来实时重构各缓存区大小，使 FPGA 中有限的缓存资源得以充分利用，避免在有缓存区空闲时仍有一些端口因较大突发流量而大量分组丢失。采用可重构缓存技术的交换系统，为获得同样的抗突发流量性能需要的存储单元数目大大下降，或者说，同样数目的存储单元可获得更好的抗突发流量性能。

参考文献：

[1] KAROL M J, HLUCHYJ M G, MORGAN S P. Input versus output queuing on a space-division packet switch[J]. IEEE Trans Com, 1987, 35(12): 1347-1356.

[2] 扈红超.分组交换网交换结构与调度策略关键技术研究[D].郑州:国家数字交换系统工程技术研究中心,2010.

HU H C. Key Technology Research on Swith Fabric and Scheduling Policy of Packet Switching Network[D]. Zhengzhou: China National Digital Switching System Engineering & Technological R&D Center,2010.

[3] 李春泉. 路由器缓存需求的研究[D].长沙:中南大学,2009.
LI C Q. Rsearch on Need for Buffer of Router[D]. Changsha:Central South University,2009.

[4] 樊晓桠. 基于 FPGA 的网络路由器报文交换算法及实现[D]. 西安: 西北工业大学, 2007.
FAN X Y. Message Exchange Algorithm and Realizaing of Network Router Based On FPGA[D]. Xi'an: Northwestern Polytechnical University, 2007.

[5] 张博,颜金尧.路由器缓存容量的分析[J].中国传媒大学学报自然科学版, 2009, 16(4):44-50.
ZHANG B, YAN J Y. Analysis and study on router buffer sizeing[J]. Journal of Communication University of China Science and Technology, 2009, 16(4):44-50.

[6] 张小平,刘振华,赵有健.可扩展路由器[J].软件学报, 2008, 19(2): 1452-1464.
ZHANG X P, LIU Z H, ZHAO Y J. Scalable router[J]. Journal of Software, 2008, 19(2): 1452-1464.

[7] WANG W M, LIGANG D, BIN Z. Analysis and implementation of an open programmable router based on forwarding and control elements separation[J]. Journal of Computing Science and Technology, 2008, 23(5): 769-779.

[8] FERREIRA R, LOURE M, BECK A C, *et al*. A low cost and adaptable routing network for reconfigurable systems[A]. IPDPS 2009[C]. Rome, Italy, 2009. 1-8.

[9] YUAN F M, DONG L G, LI C H. Service mapping in open and reconfigurable routing and switch node[A]. Proc of Computer Science and Information Technology (ICCSIT). 2010 3rd IEEE International Conference[C]. Chengdu, China, 2010. 199-203.

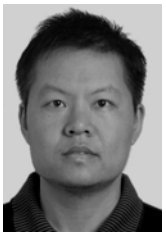
[10] ZHANG L, ESTRIN D, BURKE J, *et al*. Named Data Networking (NDN) Project[R]. Research Techmcal Report, 2010.

[11] Virtex-5 family overview[EB/OL]. http://www.xilinx.com/support/documentation/ data_sheets/ds100.pdf.

作者简介：

黄慧群（1973-），女，江苏海门人，解放军信息工程大学博士生，主要研究方向为路由交换与分发技术的研究。

刘勤让（1975-），男，河南睢县人，博士，解放军信息工程大学副教授、硕士生导师，主要研究方向为网络业务识别与控制。



卜佑军（1978-），男，河南获嘉人，博士，解放军信息工程大学讲师，主要研究方向为网络交换与路由、高速模式匹配。

张风雨（1975- ），男，河南叶县人，解放军信息工程大学讲师，主要研究方向为网络流量分析。