

How To Read the Output of the show controller frfab | tofab queue Commands on a Cisco 12000 Series Internet Router

Document ID: 18002

Introduction

Before You Begin

Conventions

Prerequisites

Components Used

Background

Buffer–Carving Algorithms

Receive Packet Memory

Packet Flow in the ToFab BMA

Transmit Packet Memory

Packet Flow in the FrFab BMA

Related Information

Introduction

This document explains how to read the output of the **show controller frfab queue** and **show controller tofab queue** commands. It also gives a detailed overview of the underlying architecture of the Cisco 12000 Series Internet Router related to these special queues.

Before You Begin

Conventions

For more information on document conventions, see the Cisco Technical Tips Conventions.

Prerequisites

There are no specific prerequisites for this document.

Components Used

The information in this document is based on:

- The Cisco 12000 Series Internet Router
- All versions of Cisco IOS© software

The information presented in this document was created from devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If you are working in a live network, ensure that you understand the potential impact of any command before using it.

Background

Each Line Card (LC) on a Cisco 12000 Series Internet Router has two types of memory:

- Route or processor memory (Dynamic RAM – DRAM): This memory enables mainly the onboard processor to run Cisco IOS software and store network routing tables (Forwarding Information Base – FIB, adjacency).
- Packet memory (Synchronous Dynamic RAM – SDRAM): Line card packet memory temporarily stores data packets awaiting switching decisions by the line card processor.

This document focuses exclusively on the Packet Memory which is divided into two banks: ToFab and FrFab (towards the Fabric and from the Fabric). The ToFab memory is used for packets that come in one of the interfaces on the LC and make its way to the fabric, whereas the FrFab memory is used for packets that are going out an interface on the LC from the fabric.

The *Tofab* and *FrFab* queues are the most important concepts to understand in order to efficiently troubleshoot ignored packets in the Cisco 12000 Series Internet Router. See [Troubleshooting Ignored Packets and No Memory Drops on the Cisco 12000 Series Internet Router](#) for details.

Note: "ToFab" (towards the Fabric) and "Rx" (received by the router) are two different names for the same thing, as are "FrFab" (From the Fabric) and "Tx" (transmitted by the router). For example, the ToFab Buffer Management ASIC (BMA) is also referred to as the Rx/BMA. This document uses the ToFab/FrFab convention, but you may see the Rx/TX nomenclature used elsewhere.

Access to packet memory is made through the Buffer Management ASIC (BMA). The BMA provides packet buffering and buffer queue management functions to the line card. All packets pass two times through the BMA – once coming in and once going out. In other words, packets arrive on a physical layer interface module (PLIM), spend a short amount of time in SDRAM buffers, and are then read out of the buffers and delivered to the Fabric Interface ASIC (FIA) module. Here, they are segmented into Cisco cells and transmitted to the switch fabric. The packets are then received from the switch fabric by the Fabric Interface ASIC on the egress line card. They are reassembled, go to SDRAM buffers, then to the PLIM, and finally sent on the wire.

Buffer–Carving Algorithms

Cisco IOS software implements a buffer–carving algorithm that divides SDRAM into various sized buffers. The GRP and other sources provide carving instructions to the line card, which then executes the instructions. There are different types of carves. For example, a simple carve creates a pool of same–sized buffers, while a complex carve creates multiple pools of different sizes, with each pool containing buffers of the same size.

All buffers of the same size are associated in one pool. One pool is always allocated for Inter–Process Communication (IPC) usage. Each associated Queue Static RAM (QSRAM) is updated with the queue head, tail, length, length threshold, associated buffer addresses in SDRAM, and the next queue element.

The following sequential conditions trigger a buffer carving on a line card:

- Bootload over the Maintenance BUS (MBUS) – Simple carve call to carve buffers to hold the Cisco IOS software image download.
- Cisco IOS software image in place – LC simple carve call to enable Inter–Process Communication (IPC) so that the GRP can use IPCs to give the LCs the initial carve specification. All the SDRAM available for carving is recarved.
- Once IPC is up – Using IPCs, the GRP can call an LC complex carve multiple times to dynamically recarve all SDRAM.

- A manual configuration or change of the MTU (Maximum Transmission Unit) on one interface causes the memory to be recarved. FrFab queues are carved up to the maximum MTU of the whole system, whereas the ToFab queues are carved up to the maximum MTU of the particular line card.

Note: We only recarve if we change the *maximum* MTU for the line card (ToFab queues), or if we change the *maximum* MTU for the whole system (FrFab queues). For instance, changing the MTU from 1500 to 4470 doesn't change anything if there is already an interface with MTU 4470 on that line card (ToFab queues) or on the whole system (FrFab queues).

Take a look at the following example:

```
Router#attach 1
Entering Console for 1 Port Packet Over SONET OC-48c/STM-16 in Slot: 1
Type "exit" to end this session
```

Press RETURN to get started!

```
LC-Slot1>enable
LC-Slot1#show controllers tofab queues
Carve information for ToFab buffers
SDRAM size: 268435456 bytes, address: 30000000, carve base: 30019100
268332800 bytes carve size, 4 SDRAM bank(s), 16384 bytes SDRAM
pagesize, 2 carve(s)
max buffer data size 4544 bytes, min buffer data size 80 bytes
262140/262140 buffers specified/carved
240637152/240637152 bytes sum buffer sizes specified/carved
```

Qnum	Head	Tail	#Qelem	LenThresh
----	----	----	-----	-----

4 non-IPC free queues:

115254/115254 (buffers specified/carved), 43.96%, 80 byte data size
1 201 115454 115254 262143
81202/81202 (buffers specified/carved), 30.97%, 608 byte data size
2 115455 196656 81202 262143
41910/41910 (buffers specified/carved), 15.98%, 1568 byte data size
3 196657 238566 41910 262143
23574/23574 (buffers specified/carved), 8.99%, 4544 byte data size
4 238567 262140 23574 262143

IPC Queue:

200/200 (buffers specified/carved), 0.7%, 4112 byte data size
30 131 130 200 262143

Raw Queue:

31 0 0 0 65535

ToFab Queues:

Dest Slot	Head	Tail	#Qelem	LenThresh
0	0	0	0	262143
1	0	0	0	262143
2	0	0	0	262143
3	0	0	0	262143
4	0	0	0	262143
5	0	0	0	262143
6	0	0	0	262143
7	0	0	0	262143

8	0	0	0	262143
9	0	0	0	262143
10	0	0	0	262143
11	0	0	0	262143
12	0	0	0	262143
13	0	0	0	262143
14	0	0	0	262143
15	0	0	0	262143
Multicast	0	0	0	262143

You can see that there have been two carves since this line card has been up and running, and that there are four pools: 80, 608, 1568, and 4544.

Now change the MTU on one interface belonging to this line card:

```
Router(config)#interface pos1/0
Router(config-if)#mtu ?
<64-18020> MTU size in bytes

Router(config-if)#mtu 2000
```

Now connect to the LC and check what has changed:

```
LC-Slot1#show control tofab queue
Carve information for ToFab buffers
  SDRAM size: 268435456 bytes, address: 30000000, carve base: 30019100
  268332800 bytes carve size, 4 SDRAM bank(s), 16384 bytes SDRAM
  pagesize, 3 carve(s)
  max buffer data size 4112 bytes, min buffer data size 80 bytes
  262142/262142 buffers specified/carved
  247054400/247054400 bytes sum buffer sizes specified/carved
```

Qnum	Head	Tail	#Qelem	LenThresh
----	----	----	-----	-----

4 non-IPC free queues:

91680/91680 (buffers specified/carved), 34.97%, 80 byte data size
1 202 201 91680 262143
65485/65485 (buffers specified/carved), 24.98%, 608 byte data size
2 91884 91883 65485 262143
49769/49769 (buffers specified/carved), 18.98%, 1568 byte data size
3 157366 207134 49769 262143
55008/55008 (buffers specified/carved), 20.98%, 2048 byte data size
4 207135 262142 55008 262143

IPC Queue:

200/200 (buffers specified/carved), 0.7%, 4112 byte data size
30 118 117 200 262143

Raw Queue:

31 206 205 0 65535
--

ToFab Queues:

Dest				
Slot				
0	0	0	0	262143
1	0	0	0	262143
2	0	0	0	262143
3	0	0	0	262143
4	0	0	0	262143

5	0	0	0	262143
6	0	0	0	262143
7	206	205	0	262143
8	0	0	0	262143
9	0	0	0	262143
10	0	0	0	262143
11	0	0	0	262143
12	0	0	0	262143
13	0	0	0	262143
14	0	0	0	262143
15	0	0	0	262143
Multicast	0	0	0	262143

There are now three carves and the maximum buffer size for the non-IPC queue is 2048 bytes instead of 4544.

The FrFab queues remain unchanged:

LC-Slot1#show controllers frfab queues

Carve information for FrFab buffers

SDRAM size: 268435456 bytes, address: 20000000, carve base: 2039D100
264646400 bytes carve size, 4 SDRAM bank(s), 16384 bytes SDRAM

pagesize, 3 carve(s)

max buffer data size 9248 bytes, min buffer data size 80 bytes

251927/251927 buffers specified/carved

209883344/209883344 bytes sum buffer sizes specified/carved

Qnum	Head	Tail	#Qelem	LenThresh
----	----	----	-----	-----

6 non-IPC free queues:

123349/123349 (buffers specified/carved), 48.96%, 80 byte data size
1 210 209 123349 262143

75519/75519 (buffers specified/carved), 29.97%, 608 byte data size
2 123552 123551 75519 262143

37759/37759 (buffers specified/carved), 14.98%, 1568 byte data size
3 199069 236827 37759 262143

2516/2516 (buffers specified/carved), 0.99%, 2048 byte data size
4 236828 239343 2516 262143

7551/7551 (buffers specified/carved), 2.99%, 4544 byte data size
5 239344 246894 7551 262143

5033/5033 (buffers specified/carved), 1.99%, 9248 byte data size
6 246895 251927 5033 262143

IPC Queue:

200/200 (buffers specified/carved), 0.7%, 4112 byte data size
30 52 51 200 262143

Multicast Raw Queue:

29 0 0 0 62981

Raw Queue:

31 52 51 0 251928

Interface Queues:

0 210 209 0 262143

The maximum buffer size is 9248 bytes. Now, configure an MTU of 10000 on one other interface on another

card:

```
Router(config-if)#interface pos5/0
Router(config-if)#mtu ?
<64-18020> MTU size in bytes

Router(config-if)#mtu 10000
```

```
LC-Slot1#show contr frfab queues
Carve information for FrFab buffers
  SDRAM size: 268435456 bytes, address: 20000000, carve base: 2039D100
  264646400 bytes carve size, 4 SDRAM bank(s), 16384 bytes SDRAM
pagesize, 4 carve(s)
max buffer data size 10064 bytes, min buffer data size 80 bytes
257309/257309 buffers specified/carved
213496016/213496016 bytes sum buffer sizes specified/carved
```

Qnum	Head	Tail	#Qelem	LenThresh
------	------	------	--------	-----------

5 non-IPC free queues:

128556/128556 (buffers specified/carved), 49.96%, 80 byte data size
1 204 203 128556 262143
77133/77133 (buffers specified/carved), 29.97%, 608 byte data size
2 128758 128757 77133 262143
38566/38566 (buffers specified/carved), 14.98%, 1568 byte data size
3 205890 244455 38566 262143
7713/7713 (buffers specified/carved), 2.99%, 4544 byte data size
4 244456 252168 7713 262143
5141/5141 (buffers specified/carved), 1.99%, 10064 byte data size
5 252169 257309 5141 262143

IPC Queue:

200/200 (buffers specified/carved), 0.7%, 4112 byte data size
30 24 23 200 262143

Multicast Raw Queue:

29 0 0 0 64327
--

Raw Queue:

31 24 23 0 257310
--

Interface Queues:

0 205 204 0 262143
--

There are now four carves for the FrFab queues and the maximum buffer size has changed to 10064 bytes.

Note: On Packet Over Sonet (POS) line cards configured with Point-to-Point Protocol (PPP) encapsulation, Maximum Receive Unit (MRU) negotiation does occur, but it does not adjust the MTU size. Moreover, the PPP connections are not reset when the MTU is changed on the interface.

Receive Packet Memory

This memory is carved into different pools of packet buffers. To see how the receive memory is carved, you can **attach** to a Line Card and execute the **show controller tofab queue** command, as demonstrated below:

```
Router#attach ?
```

<0-15> slot number of linecard to connect
<cr>

Router#**attach 1**

Entering Console for 1 Port SONET based SRP OC-12c/STM-4 in Slot: 1

Type "exit" to end this session

Press RETURN to get started!

LC-Slot1>**enable**

LC-Slot1#

LC-Slot1#**show controllers tofab queues**

Carve information for ToFab buffers

SDRAM size: 33554432 bytes, address: 30000000, carve base: 30029100

33386240 bytes carve size, 4 SDRAM bank(s), 8192 bytes SDRAM pagesize, 2 carve(s)

max buffer data size 9248 bytes, min buffer data size 80 bytes

40606/40606 buffers specified/carved

33249088/33249088 bytes sum buffer sizes specified/carved

Qnum	Head	Tail	#Qelem	LenThresh
------	------	------	--------	-----------

5 non-IPC free queues:

20254/20254	(buffers specified/carved), 49.87%, 80 byte data size			
1	17297	17296	20254	65535

12152/12152	(buffers specified/carved), 29.92%, 608 byte data size			
2	20548	20547	12152	65535

6076/6076	(buffers specified/carved), 14.96%, 1568 byte data size			
3	32507	38582	6076	65535

1215/1215	(buffers specified/carved), 2.99%, 4544 byte data size			
4	38583	39797	1215	65535

809/809	(buffers specified/carved), 1.99%, 9248 byte data size			
5	39798	40606	809	65535

IPC Queue:

100/100	(buffers specified/carved), 0.24%, 4112 byte data size			
30	72	71	100	65535

Raw Queue:

31	0	17302	0	65535
----	---	-------	---	-------

ToFab Queues:

Dest	Slot	Head	Tail	#Qelem	LenThresh
	0	0	0	0	65535
	1	0	0	0	65535
	2	0	0	0	65535
	3	0	0	0	65535
	4	0	0	0	65535
	5	0	17282	0	65535
	6	0	0	0	65535
	7	0	75	0	65535
	8	0	0	0	65535
	9	0	0	0	65535
	10	0	0	0	65535
	11	0	0	0	65535
	12	0	0	0	65535
	13	0	0	0	65535
	14	0	0	0	65535
	15	0	0	0	65535
Multicast		0	0	0	65535

LC-Slot1#

The following list describes some of the key fields that are found in the previous example:

- **SDRAM size: 33554432 bytes, address: 30000000, carve base: 30029100** – The size of receive packet memory and the address location where it begins.
- **max buffer data size 9248 bytes, min buffer data size 80 bytes** – The maximum and minimum buffer sizes.
- **40606/40606 buffers specified/carved** – Buffers to be carved specified by Cisco IOS software and the number of buffers actually carved.
- **non-IPC free queues** – The non-IPC buffer pools are the packet buffer pools. Packets arriving into the line card are allocated a buffer from one of these buffer pools depending on the size of the packet. It is possible to have only three non-IPC free queues; if the board is Ethernet, you will not have the 4k pool, but only a pool up to 1.5k. This is because the ToFab queues are carved up to the maximum transmission unit (MTU) of that particular line card. The example output shows five packet buffer pools of sizes 80, 608, 1568, 4544, and 9248 bytes. For each pool, more details are given below:
 - ◆ **20254/20254 (buffers specified/carved), 49.87%, 80 byte data size** – 49.87% of the receive packet memory has been carved into 20254 80-byte buffers.
 - ◆ **Qnum** – The queue number.
 - ◆ **#Qelem** – The number of buffers that are currently assigned to that queue. If it is a free queue, then these buffers are available to the system. If it is a ToFab queue or a transmit queue, these buffers are not available to the system. This is the column to check to find out which queue is backed up.
 - ◆ **Head and Tail** – A head and tail mechanism is used to ensure the queues are moving properly.
- **IPC Queue** – Reserved for Inter-Process Communication messages from the LC to the GRP.
- **Raw Queue** – When an incoming packet has been assigned a buffer from a non-IPC free queue, it's enqueued on the raw queue. The raw queue is a First In, First Out (FIFO) processed by the LC CPU during interrupts. If you see a very large number in the #Qelem column of the "Raw Queue" row, you have too many packets waiting on the CPU and they will start getting ignored because the CPU can't keep up with the load. However, this is very rare.
- **ToFab Queue** – Virtual output queues; one per destination slot plus one for multicast traffic. The last portion of the previous example shows 15 virtual output queues. This is a 12012 router, which was originally designed as a 15-slot chassis; queues 13 through 15 are not used.

After the ingress line card CPU makes a packet switching decision, the packet is enqueued on the virtual output queue corresponding to the slot where the packet is destined. The number in the fourth column is the number of packets currently enqueued on a virtual output queue.

Packet Flow in the ToFab BMA

Step 1 – A packet comes into the physical layer interface module (PLIM). As the packet is received and processed, it is DMA'd (Direct Memory Access) into a small (approximately 2 x Maximum Transmission Unit (MTU) buffer) memory called the "First In, First Out (FIFO) burst memory". The amount of this memory depends on the type of LC (from 128 KB to 1 MB).

Step 2 – When the packet is completely in FIFO memory, an application-specific integrated circuit (ASIC) on the PLIM contacts the Buffer Management ASIC (BMA) and asks for a buffer to put the packet in. The BMA is told what size the packet is, and allocates a buffer accordingly. If the BMA cannot get a buffer of the right size, the packet is dropped and the "ignored" counter is incremented on the incoming interface. There is no fallback mechanism as with some other platforms.

Step 3 – While this is going on, the PLIM may be receiving another packet in the FIFO burst memory, which is why it is 2xMTU in size. If there is a free buffer available in the right queue, the packet is stored by the BMA in the free queue list of the appropriate size. This buffer is placed on the Raw Queue, which is examined by the Salsa ASIC or the R5K CPU, depending on the line card switching engine type.

Step 4 – On the engine 0 LC, the R5K CPU determines the destination of the packet by consulting its local Distributed Cisco Express Forwarding (dCEF) tables in DRAM. It then moves the buffer from the Raw Queue to a ToFabric queue corresponding to the destination slot. If the destination is not in the dCEF tables, the packet is dropped. If the packet is a control packet (for example, routing updates), it is enqueued to the queue of the GRP and is processed by the GRP. On a 12016 router, there are 17 ToFab queues (16 unicast, plus one Multicast).

Step 5 – The ToFab BMA enqueues the buffer into the proper ToFab queue. At this point, the #Qelem counter in the pool the buffer came from decreases by one, and the ToFab queue counter increases by one.

Note: There is one ToFab queue per line card (this includes the GRP). These queues are known as Virtual Output Queues (VOQs). These are important for avoiding head-of-line blocking.

Step 6 – The Fabric Interface ASIC (FIA) sees that an output queue is non-empty. The FIA is set up to segment the packet into 48-byte cells. An 8-byte header is added onto the packet and the 56-byte Cisco cell is sent across the switch fabric.

Transmit Packet Memory

Transmit packet memory stores packets coming from the switch fabric and awaiting transmission to the physical interface. This memory is also carved into pools of various sizes.

From the GRP, you can **attach** to a line card and execute the **show controller frfab queue** command to display the transmit packet memory. In addition to the fields in the ToFab output, the FrFab output displays an "Interface Queues" section. The output varies with the type and number of interfaces on the outgoing LC.

One such queue exists for each interface on the line card. Packets destined out a specific interface are enqueued onto the corresponding interface queue.

```
LC-Slot1#show controller frfab queue
===== Line Card (Slot 2) =====
Carve information for FrFab buffers
  SDRAM size: 16777216 bytes, address: 20000000, carve base: 2002D100
  16592640 bytes carve size, 0 SDRAM bank(s), 0 bytes SDRAM pagesize, 2 carve(s)
  max buffer data size 9248 bytes, min buffer data size 80 bytes
  20052/20052 buffers specified/carved
  16581552/16581552 bytes sum buffer sizes specified/carved
      Qnum      Head      Tail          #Qelem  LenThresh
      ----      -
5 non-IPC free queues:
  9977/9977 (buffers specified/carved), 49.75%, 80 byte data size
  1          101      10077         9977    65535

  5986/5986 (buffers specified/carved), 29.85%, 608 byte data size
  2          10078    16063         5986    65535

  2993/2993 (buffers specified/carved), 14.92%, 1568 byte data size
  3          16064    19056         2993    65535

  598/598 (buffers specified/carved), 2.98%, 4544 byte data size
  4          19057    19654         598     65535

  398/398 (buffers specified/carved), 1.98%, 9248 byte data size
  5          19655    20052         398     65535

IPC Queue:
  100/100 (buffers specified/carved), 0.49%, 4112 byte data size
  30       77       76           100     65535
```

Raw Queue:				
31	0	82	0	65535
Interface Queues:				
0	0	0	0	65535
1	0	0	0	65535
2	0	0	0	65535
3	0	0	0	65535

The following list describes some of the key fields that are found in the previous example:

- **Non-IPC free queues:** These queues are packet buffer pools of different sizes. When a packet is received over the fabric, an appropriate sized buffer is taken from one of these queues, the packet is copied into it, and the buffer is placed on the appropriate output interface queue.

Note: There are as many pools as needed for the whole router. As a consequence, FrFab queues are carved up to the maximum MTU of the whole system. This is different for the ToFab queues which are carved up to the maximum MTU of the particular line card.

- **IPC queue:** Reserved for Inter-Process Communication messages from the GRP to the LC.
- **Interface queues:** These queues are for the interfaces, not for the slot numbers. The last number (65535) is the TX-queue-limit. This number controls the maximum length of any queue and can be tuned by the **TX-queue limit** command on the Engine 0 line card. If you experience some congestion, this command can be used to prevent the egress LC from buffering more than the configured number of packets on the interface queue for that specific port. Make sure that you configure this number low enough so that it does not contain all the FrFab queues for this interface. However, this tuning provides no control over which packets get dropped on the outbound LC. See Troubleshooting Ignored Packets and No Memory Drops on the Cisco 12000 Series Internet Router for details.

Packet Flow in the FrFab BMA

At this point, the Cisco cells have been transmitted over the switch fabric by the FIA.

Step 1 – These Cisco cells are DMA'd into FIFOs on the FrFab FIAs, and then into a buffer on the FrFab BMA. The FrFab BMA is the one that actually does the reassembly of cells into a packet.

How does the FrFab BMA know in which buffer to put the cells before it reassembles them? This is another decision made by the incoming line card switching engine. Since all queues on the entire box are the same size and in the same order, the switching engine tells the transmitting LC to put the packet in the same number queue from which it entered the router.

The FrFab BMA SDRAM queues can be viewed with the **show controller frfab queue** command on the LC.

Step 2 – This step is basically the same as the ToFab BMA output. Packets come in and are placed in packets that are dequeued from their respective free queues. These packets are placed into the FrFab queue, and enqueued on either the interface queue (there is one queue per physical port) or the rawQ for output processing. Not much happens in the rawQ: per-port multicast replication, Modified Deficit Round Robin (MDRR) – same idea as Distributed Weighted Fair Queuing (DWFQ), and output Committed Access Rate (CAR). If the transmit queue is full, the packet is dropped and the output drop counter is incremented.

Step 3 – The FrFab BMA waits until the TX portion of the PLIM is ready to send a packet. The FrFab BMA does the actual Media Access Control (MAC) rewrite (based, remember, on information contained in the Cisco Cell header), and DMA's the packet over to a small (again, 2xMTU) buffer in the PLIM circuitry. The PLIM does the Asynchronous Transfer Mode (ATM) segmentation and reassembly (SAR) and Synchronous

Optical Network (SONET) encapsulation, where appropriate, and transmits the packet.

Related Information

- **[Troubleshooting Ignored Packets and No Memory Drops on the Cisco 12000 Series Internet Router](#)**
 - **[Troubleshooting Input Drops on the Cisco 12000 Series Internet Router](#)**
 - **[How To Read the Output of the Show Controller fia Command](#)**
 - **[Technical Support – Cisco Systems](#)**
-

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2008 – 2009 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

Updated: Jul 07, 2005

Document ID: 18002
