Quantum Flow Control

Version 2.0
July 25, 1995

FCC-SPEC-95-1

ABSTRACT

This document describes Quantum Flow Control (QFC), a cell-relay protocol characterized by quantized buffer state feedback. QFC is a protocol mechanism that supports an Available Bit Rate (ABR) cell service that is distinguished for its ability to support instantaneous access to unused (available) network bandwidth with cell loss rates no greater than link bit error rates. The ABR service is designed for bursty applications where the application's bandwidth requirements are difficult to predict in advance. The service uses the QFC protocol to provide a feedback mechanism that controls the source rate to ensure zero cell loss regardless of network congestion. QFC supports virtual path and virtual channel services over point-to-point and point-to-multipoint topologies. Additionally, QFC operates over both physical links and tunnels. Utilizing buffer sharing techniques, the QFC protocol allows economical performance for links that are 1 meter to 10,000 kilometers or more in length and that operate at speeds up to many gigabits per second (Gbps).

General Release

Copyright Notification

Editors:

Mike Gaddis & Walter Kelt

Contributing Authors:

Mike Gaddis, Walter Kelt, Rick Bubenik, John Howard, Steve Hauser, Ken Brinkerhoff, Cüneyt Özveren, Bob Simcoe, Jim Schwalbe, Klaus Fosmark, Raj Nair, Doug Hunt

Record of Changes

Version	1.5	(June 19, 1995)	First official draft approved for NDA Release.
Version	1.8	(July 6, 1995)	Included Appendix F, used for Baseline.
Version	1.9	(July 7, 1995)	Included Appendix G, **Baseline QFC Text**.
Version	1.9.2	(July 20, 1995)	Changed Limits to 7 and 27 bit precision. Simplified Equation [5]. Some editorial clean-up.
Version	2.0	(July 25 1995)	Approved Draft for General Release.

Table of Contents

Appendices

Quantum Flow Control

Version 2.0
July 25, 1995 @ 9:41 pm

FCC-SPEC-95-1

ABSTRACT

This document describes Quantum Flow Control (QFC), a cell-relay protocol characterized by quantized buffer state feedback. QFC is a protocol mechanism that supports an Available Bit Rate (ABR) cell service that is distinguished for its ability to support instantaneous access to unused (available) network bandwidth with cell loss rates no greater than link bit error rates. The ABR service is designed for bursty applications where the application's bandwidth requirements are difficult to predict in advance. The service uses the QFC protocol to provide a feedback mechanism that controls the source rate to ensure zero cell loss regardless of network congestion. QFC supports virtual path and virtual channel services over point-to-point and point-to-multipoint topologies. Additionally, QFC operates over both physical links and tunnels. Utilizing buffer sharing techniques, the QFC protocol allows economical performance for links that are 1 meter to 10,000 kilometers or more in length and that operate at speeds up to many gigabits per second (Gbps).

1. Document Scope and Organization

This document is a protocol specification. It will eventually be used as a baseline document for future standards work relating to QFC.

The document is organized into three parts. The introduction, Sections 1 through 5, is intended for a general audience. The second part, Sections 6 through 10, is a detailed technical explanation of the protocol. Finally, the appendices present interpretive and explanatory material for the protocol.

2. Confidentiality

This document is NOT confidential. It may be freely distributed to any interested party. This document may not, however, be altered in any way without the express written permission of the Flow Control Consortium. This document should be distributed intact.

3. Point Of Contact

NOTICE: Flow Control Consortium companies have contributed to the preparation of this document. It is offered as a basis for discussion and is not binding on the contributing organizations. The material in this document is subject to editorial changes for clarification purposes. If you have any questions about or comments on this document, contact one of the company representatives listed in Table 1.

Table 1. Flow Control Consortium Company Points of Contact

Mike Gaddis Director, Software Development Ascom Nexion Inc. (314) 579-6511 gaddis@nexen.com	Tracy Earles Senior Product Marketing Engineer Brooktree Corporation (303) 543-2004 tearles@brooktree.com
Jim Schwalbe Director of Hardware Development CrossComm Corp. (508) 229-5129 schwalbe@crosscomm.com	Edward Sztuka Vice President Sales & Marketing Desknet Systems Inc. (914) 273-6232, extension 206 ajay@desknet.com
Cüneyt Özveren Principal Engineer Digital Equipment Corporation (508) 486-5995 ozveren@cuneyt.lkg.dec.com	Chase Bailey Chief Technical Officer Efficient Networks Inc. (214) 991-3884 chase@efficient.com
Thomas Daniel Director of System Engineering LSI Logic Corporation (408) 433-7852 thomasd@lsil.com	William N. Giudice President Maker Communications Inc. (617) 672-0622 guidice@maker.com
John Howard Senior Research Scientist Mitsubishi Electric Research Labs (617) 621-7521 howard@merl.com	Alex Joffe Vice President of Engineering MMC Networks, Inc. (408) 653-1810 extension 417 alex@mmcnet.com
Ken Brinkerhoff Director of Systems Architecture Standard Microsystems Corp. (714) 707-2437 ken_brinkerhoff@west.smc.com	

To be put on the Flow Control Consortium mailing list for revision releases of the protocol specification, contact:

Debi Folkerts
Senior Department Administrator
Ascom Nexion Inc.
1807 Park 270 Drive, Suite 350
Saint Louis, Missouri, 63141

folkerts@nexen.com
(Office) (314) 579-6510
(FAX) (314) 542-0495

4. Available Bit Rate (ABR) Service Overview

This section describes the ABR Service definition as defined by the Flow Control Consortium. Note that this definition is more comprehensive than that currently assumed by the ATM Forum. For example, a specific, very low cell loss rate is guaranteed.

4.1 ABR Application Characteristics

The Available Bit Rate (ABR) Service supports ATM connections that are characterized by an unpredictable and bursty flow of cells from a transmitter to one or more receivers. LAN Emulation and Classical IP over ATM are two examples of such applications. In such applications, a transmitter may require rapid access to all available bandwidth. Supporting such services is an important step to achieve the seamless integration of frame-based with circuit-based traffic. Bursty applications like LAN Emulation:

- are less sensitive to delay and delay variation than circuit-based services,
- require very small cell loss rates for good performance (i.e. no greater than 1 cell lost in 10^8 cells sent),
- have indeterminate bandwidth requirements, and
- are often able to use as much bandwidth as available when ready to transmit.

Meeting the needs of bursty, frame-based applications is the primary purpose of the ABR Cell-Relay Service.

4.2 ABR Service & Network Requirements

The ABR Service is defined to have cell loss commensurate to link aggregate statistical Bit Error Rate (BER). In other words, the ABR service guarantees that cells are not lost due to network congestion. Loss of cells may occur from transient network faults such as link bit errors, link loss of signal, or device faults. These faults are difficult to prevent but are assumed to be rare — on the order of one error event in 10^{10} for link BER or less. The ABR service may, therefore, be characterized as lossless under normal network operating conditions.

ABR increases network efficiency by allowing network providers to reuse bandwidth that is not being used by other connections, enabling 100 percent link utilization to be achieved on a link-by-link basis. Furthermore, unlike other Variable Bit Rate (VBR) services, all ABR data transported though the network end-to-end is "goodput" — throughput that is usable by the end user's application. The small cell size of ATM and the large frame sizes typical in high data rate applications intensify the goodput/badput problem in ATM networks when cells are lost. For this reason, supplying a low cell loss service that supports bursty applications is especially important for ATM.

If 100 percent network utilization is to be achieved, then any single ABR connection that has data queued to send at a network transmitter must be able to operate at Link Cell Rate instantaneously. Furthermore, if multiple connections have data to send, the ABR service should ensure that fair access to the bandwidth is achieved.

The ABR service, coupled with connections using the Constant Bit Rate (CBR) service, allow network providers to achieve the original vision of ATM statistical efficiency gain over traditional time division multiplexing (TDM) circuit services. With these two cell-relay services the important goal of integrating circuit- and packet-based traffic onto a single wire can be achieved.

4.3 ABR Service Connection Types & Topologies

ABR connection types include:

- ABR service on Virtual Channel Connections (VCCs),
- ABR service on Virtual Path Connections (VPCs),
- ABR VCCs carried within a CBR VPC, and
- ABR VCCs carried within an ABR VPC.

The connection types may be provided in either point-to-point or point-to-multipoint topologies for both virtual paths and virtual channels. Other topologies may also emerge (such as multipoint-to-multipoint or multipoint-to-point), but these are beyond the current scope of this document.

5. Quantum Flow Control Summary

Quantum Flow Control (QFC) supports the requirements of the ABR service definition by controlling the number of cells buffered at each node traversed by a connection. It is a hop-by-hop protocol, meaning that there is a separate control loop for each link traversed by a connection. It explicitly controls buffer occupancy levels and therefore completely avoids cell loss due to congestion, a fundamental requirement of the ABR service. As a hop-by-hop protocol, it is applicable to all types of connections, such as virtual circuit, virtual path, and point-to-multipoint connections. The hop-by-hop implementation of the protocol provides the shortest possible control loop, which provides the fastest feedback allowable by the physics of ATM.

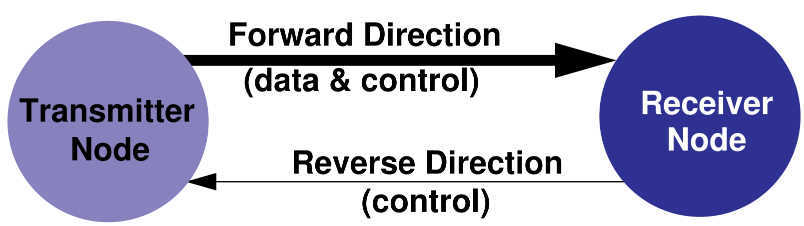


Figure 1. Protocol Reference Model (single hop)

Figure 1 summarizes the data forwarding part of QFC working over a single hop. A hop is a connection between a transmitter and a receiver on a virtual link. A virtual link is either a physical link or a *tunnel* through a VPC. The transmitter sends a cell whenever it has flow controlled cells queued and it knows that there is sufficient buffer space to receive that cell. The receiver may buffer the received cells and will, at some later time, forward the cells onto the next hop, thus freeing buffer space. It then informs the transmitter about the freed space using buffer state update messages. As buffer state update messages arrive, the transmitter sends more cells, completing the control loop.

Cells eventually arrive at the network destination, which is typically a Segmentation and Reassembly (SAR) device supporting a consuming application. The application will drain the cells (converted into frames by the SAR), completing the end-to-end transfer from the cell relay perspective.

In addition to the data forwarding protocol, QFC provides protocols to support buffer allocation, robustness against data and control cell loss, and initialization.

5.1 Quantum Flow Control Protocol Characteristics

Salient characteristics of QFC are that it:

- guarantees zero cell loss due to network congestion
- allows instantaneous access to available network bandwidth
- supports both Virtual Channel Connections and Virtual Path Connections
- supports ABR service over point-to-multipoint connections
- supports tunneling through CBR and ABR VPCs
- is robust against misbehaving traffic sources and cell loss
- is simple, allowing low cost implementations in both switches and end systems
- may be tested for conformance at any point in the network
- scales economically to all ranges of ATM link cell rates and distances
- limits control overhead to less than 7.5 percent of the link bandwidth
- allows latitude in the choice of memory size, memory management techniques, connection priorities, and cell scheduling policies
- configures automatically
- requires minimal ATM signalling
- does not require a signaled bandwidth specification

Two particularly appealing features of QFC are its intrinsic support for multipoint topologies and its minimal burden on signaling procedures for connection setup. Elegant solutions to these two problems have proven elusive to the designers of alternative ABR service mechanisms.

Since the QFC control loop is closed between a uni-directional transmitter and receiver on a single hop, constructing multipoint topologies is straightforward, each segment's resources can be locally managed without global network knowledge. At any point in a multipoint tree, a branch may be constructed on a connection allowing a multipoint leg to be easily added.

The QFC protocol requires no advanced bandwidth reservation or complex or negotiated parameters at call setup. Setting a single code point in an information element that already exists is the only required signaling extension. The bandwidth specification becomes optional (unlike current service offerings). Furthermore, the protocol does not require a complex Call Admission Control (CAC) algorithm.

In addition to its signaling simplicity, QFC also solves one of the perplexing problems of ATM signaling and traffic management. When should a calling or called party begin sending data after a call setup operation is completed? In current ATM services like Best Effort (Unspecified Bit Rate—UBR) and other proposed ABR service mechanisms, a race condition exists between the arrival of signaling messages to the two terminal endpoints of a call and the start of either terminal's data transmission. In these services, the terminals may lose their first data cells because the network nodes in the connection path or the terminal on the other end may not be ready to receive data. Since QFC terminals cannot transmit until given buffers by a downstream receiver, no race condition exists and data cannot be lost. **A QFC end-to-end connection may be setup segment-by-segment asynchronously with no loss of data**. QFC becomes the calling party-to-called party data transfer synchronization mechanism.

A QFC node may over-book the ABR service, allowing connections to gracefully share limited resources. With QFC, performance may be traded for cost in an elegant and predictable way. When making these trade-offs, the correctness of the QFC system is not compromised and performance degrades gracefully.

QFC is designed to "step aside without cell loss" when higher priority, real-time traffic like CBR or real-time VBR needs access to bandwidth. Yet, the service is "ready to go" whenever that bandwidth is unused. Furthermore, it can do all this without affecting the behavior of those higher priority services, achieving the ABR goal of utilizing bandwidth "when available." However, this also means that the bandwidth allocated to the ABR service is constantly varying. For this reason, managing the allocated ABR service buffers and available bandwidth carefully — as QFC does — is a fundamental requirement to meet the desired ABR service requirements. In essence, it is a matter of cell-relay physics, QFC→QED.

6. Protocol Reference Model

A protocol reference model is a set of system elements, counters, and messages used to define terminology and describe the objects and resources involved in a protocol. This section presents the reference model for Quantum Flow Control. Section 7 details the way the elements of this model interact.

6.1 Transmitter, Receiver, Link, and Connection Segments

The major physical components involved in Quantum Flow Control are the **logical link**, a **transmitter**, and a **receiver**. Instantiated on top of these physical components are **connections**. Connections are logical entities that are established between transmitters and receivers and traverse the link. These physical and logical components are shown in Figure 2.

A logical link may be either a physical link or a tunnel. A tunnel is implemented by a VPC between the transmitter and receiver and is treated as if it were a single link by QFC even if it passes through several other nodes or networks. A connection may be either a VCC or a VPC. VCCs can traverse both physical links and tunnels; VPCs can traverse only physical links.

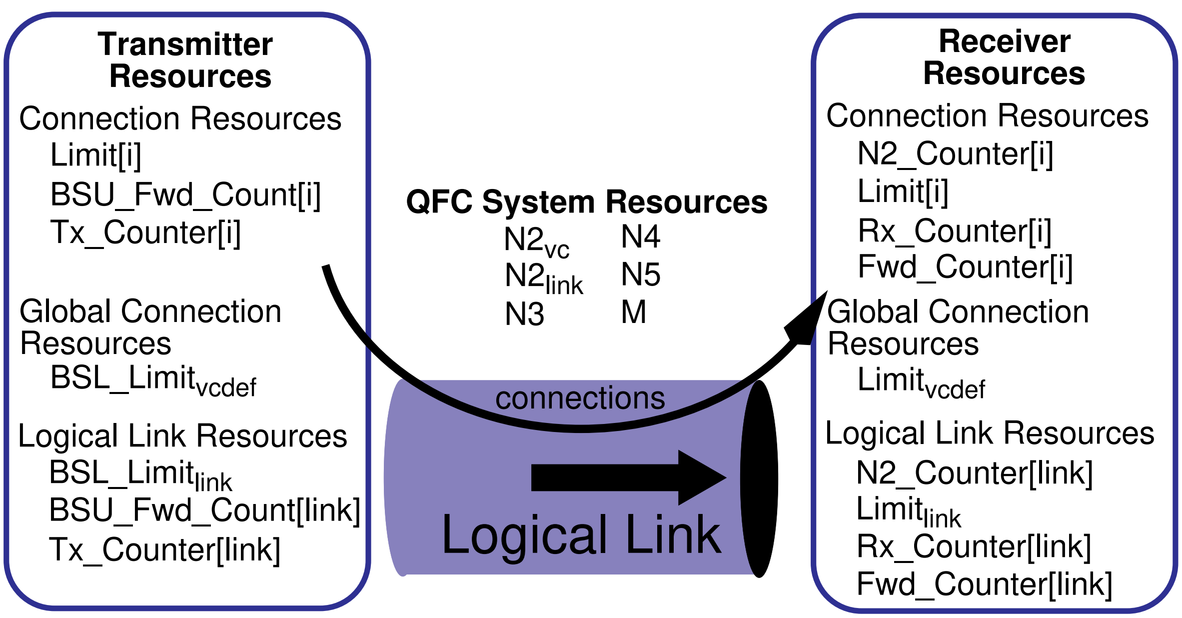


Figure 2. Transmitter, Receiver, Link and Connection QFC Resources

6.2 Transmitter and Receiver Resources

The protocol requires that the transmitter and receiver maintain a number of counters { ⊕ } and variables { ν }, as shown in Figure 2. All counters are monotonically increasing and wrap.

At the transmitter:

- { ⊕ } **Tx_Counter**[i] counts cells transmitted on the connection [i] (the [i] notation stands for an index that uniquely identifies a connection).
- { ν } **BSU_Fwd_Count**[i] contains a value received in an update message from the receiver that represents how many cells had been forwarded by the receiver at the time the message was composed for connection [i].
- { ν } **BSL_Limit**$_{vcdef}$ contains a global connection resource value (Limit$_{vcdef}$) received in the BSL limit message from the receiver that represents the default upper bound on the number of buffers allocated to a connection.
- { ν } **Limit**[i] contains the actual limit in use by connection [i] which will either be the BSL_Limit$_{vcdef}$ or a Limit override received in the signaling SETUP or CONNECT message at connection initialization.

At the receiver:

- { ⊕ } **Rx_Counter**[i] counts cells received from the transmitter for connection [i].
- { ⊕ } **Fwd_Counter**[i] counts how many cells have been forwarded by the receiver for connection [i].
- { ν } **Limit**$_{vcdef}$ contains a global connection resource value for the default upper bound on the number of buffers allocated to connections.
- { ν } **Limit**[i] contains the actual limit in use by connection [i] which will either be the Limit$_{vcdef}$ or a Limit override received in the signaling SETUP or CONNECT messages at connection initialization.
- { ⊕ } **N2_Counter**[i] contains the number of cells forwarded by the receiver since the last update message was initiated for connection [i].

In addition to the per-connection resources, there are several resources that apply to the link as a whole. The link resources are the same as connection resources, except that they apply to all QFC cells traversing the link rather than cells traversing an individual connection. These resources are identified with the notation [link] or $_{link}$ to represent a link resource versus a connection resource. In the case of the receiver's Limit$_{link}$ and the transmitter's BSL_Limit$_{link}$ there is a subtle difference between these link resources and their counterpart global connection resources for Limit$_{vcdef}$ and BSL_Limit$_{vcdef}$. Limit$_{link}$ and BSL_Limit$_{link}$, while arguably global resources, apply to only one entity — the logical link.

QFC System Resources—system invariants used by the transmitter and receiver:

(see Appendix B for more details)

- { ν } **N2**$_{\mathbf{vc}}$ represents the nominal number of cells that the receiver forwards before initiating an update message for a connection.
- { ν } **N2**$_{\mathbf{link}}$ represents the nominal number of cells that the receiver forwards before initiating an update message for a link. Generally, N2$_{link}$ is considerably larger than N2$_{vc}$.
- { ν } **N3** represents the measured Round-Trip-Time (RTT) of the logical link.

- { ν } **N4** represents the maximum number of Fwd_Counter updates ([i] or [link]) that can be packed into one update message sent from the receiver.
- { ν } **N5** represents the update message processing delay of the transmitter and receiver. (N5 is broken down into its two constituent components, $N5_{recv}$ and $N5_{trans}$, which represent the delay at the receiver and transmitter, respectively.)
- { ν } **M** represents an overhead bandwidth divisor that, when divided into the logical links cell rate, establishes how much bandwidth may be used for Fwd_Counter updates ([i] or [link]).

With these resources, information about the current state of the control system may be determined. For instance, the number of stored cells at the receiver can be determined through the simple equation:

- Rx_Counter[i] - Fwd_Counter[i] = stored cells for [i] at receiver[1].

Also, the number of cells in flight at any instant in time could be determined by the following simple equation:

- Tx_Counter[i] - Rx_Counter[i] = cells "in flight" on connection [i][1].

The transmitter and receiver are asynchronous systems. Consequently, the transmitter can never know — precisely — how many cells are on the wire or are queued in a receiver's buffers. The transmitter does have, however, specific knowledge about the maximum number of cells that *could* be in flight to the receiver or that *might* be stored in a receiver's buffers at any instance in time by subtracting BSU_Fwd_Count[i] from Tx_Counter[i]. This quantity is compared to the Limit[i] to make sure that the transmitter never sends more cells than the receiver has allocated buffers for.

Similarly, after making sure there are sufficient buffers available for the connection [i], the transmitter also ensures that there are sufficient buffers left on the logical link. This is important because the receiver is allowed to over-book its buffer space so that connections may share buffers in order to reduce cost.

The following constraints establish the fundamental required transmitter behavior that ensures zero cell loss due to congestion by QFC:[2]

- Tx_Counter[i] - BSU_Fwd_Count[i] ≤ Limit[i]
- Tx_Counter[link] - BSU_Fwd_Count[link] ≤ BSL_Limit_{link}

These constraints enforce the correctness of the QFC claim of zero cell loss due to congestion. Other ABR service proposals rely on less reliable statistical methods.

1 For simplicity these two equations have been shown as simple subtractions, but to be accurate, because the counters are monotonically increasing and wrap, the equations require modulo subtraction. The full equations will be detailed later.
2 Again, to be accurate, this requires a modulo subtraction.

6.3 Messages

The operation of the QFC protocol is defined in terms of messages that are sent from the transmitter to the receiver or vice versa and procedures relating to the receipt of those messages. (Section 10 shows how messages are actually carried in ATM cells, and Section 8 discusses the channels that the messages are sent over.) Protocol messages used by QFC are:

- Buffer State Update, **BSU**([i,link], Fwd_Counter[i,link]) - Sent from the receiver to the transmitter to update the value of BSU_Fwd_Count[i,link] in the transmitter.
- Buffer State Check, **BSC**([i,link], Tx_Counter[i,link]) - Sent from the transmitter to the receiver to verify the value of Rx_Counter and to compensate for lost QFC cells or lost BSU messages due to link bit errors.
- Buffer State Limit, **BSL**(**Indicate**, Limit_{vcdef} || Limit_{link})[3] - Sent from the receiver to the transmitter to establish an initial or new default BSL_Limit_{vcdef} or BSL_Limit_{link}, as appropriate, in the transmitter.
- Buffer State Limit, **BSL**(**Confirmation**, BSL_Limit_{vcdef} || BSL_Limit_{link}) - Sent from the transmitter to the receiver to confirm a new default BSL_Limit_{vcdef} or BSL_Limit_{link}, as appropriate.
- Buffer State RTT, **BSR**(**Indicate**, Tx_starttime) - Sent from the transmitter to the receiver to measure forward logical link propagation delay (LLPD).
- Buffer State RTT, **BSR**(**Confirmation**, Tx_starttime, Rx_receivetime, Rx_forwardtime) - Sent from the receiver to the transmitter to measure reverse LLPD.
- Auto-Configuration Protocol, **ACP**(**Indicate**, Revision ID, SL/LL Option, N4, $N2_{link}$, $N5_{recv}$, M) - Sent from the receiver to the transmitter to establish receiver auto-configuration parameters of QFC.
- Auto-Configuration Protocol, **ACP**(**Confirmation**, Revision ID, SL/LL Option, N4, $N2_{vc}$, $N2_{link}$, $N5_{trans}$, $N5_{recv}$, M) - Sent from the transmitter to the receiver to confirm the indicated auto-configuration parameters and to establish some additional parameters of QFC like $N2_{vc}$ and $N5_{trans}$.

The term Flow Controlled Virtual Connection (FCVC) is used to refer to the part of the Quantum Flow Control protocol that relates to managing a connection ([i]) type resource. Likewise, the term Flow Controlled Virtual Links (FCVL) is used to refer to the part of the protocol that relates to managing link ([link]) resources.

7. Procedural Behavior

7.1 QFC Cell Forwarding Operations

The following sections detail the required transmitter and receiver behavior for the Quantum Flow Control Model described in the previous section.

[3] The || notation means -or-

7.1.1 Buffer State Update (BSU) Counters and Variables

The counters, Tx_Counter[i], Rx_Counter[i], Fwd_Counter[i] Tx_Counter[link], Rx_Counter[link], Fwd_Counter[link] and the variables, BSU_Fwd_Count[i], BSU_Fwd_Count[link], shall be initialized to zero when a connection segment is first established or a link is initialized. Therefore, when a Limit[i] (equal to the default Limit_{vcdef} or a connection limit override via signaling) or a BSL_Limit_{link} is first established for a connection or link, this number represents an initial allocation of buffers to that connection or link.

When the transmitter sends a cell, it shall increment Tx_Counter[i] and Tx_Counter[link] to ensure that the receive buffer reserved for that cell is accounted for by the transmitter. The transmitter shall not reuse that reserved buffer until an update message from the receiver indicates that the cell has been forwarded.

When a receiver accepts a cell from the forward direction link, it shall increment Rx_Counter[i] and Rx_Counter[link]. Later, when a receiver forwards a received cell downstream, it shall increment the Fwd_Counter[i], N2_Counter[i], Fwd_Counter[link], and N2_Counter[link].

The transmitter does not increment its BSU_Fwd_Count[i] or BSU_Fwd_Count[link] but instead uses the value of BSU([i], Fwd_Counter[i]) or BSU([link], Fwd_Counter[link]), as appropriate, from the BSU messages received from the receiver.

All counters and variables are unsigned integers.

In the Short Link (SL) option, all FCVC counters are unsigned modulo 2^8. In the Long Link (LL) option, all FCVC counters are unsigned modulo 2^{28}. In both options, SL and LL, the FCVL link counters are unsigned modulo 2^{28}.

QFC counters for a VCC or VPC using QFC include all cells that flow on the connection, except for QFC BSC message cells (which are a subset of RM cells (PTI 6), see Section 10.4). QFC counters for a logical link include the same cells that are included in QFC counters of all of the logical link's QFC VCCs or VPCs. Note that since other QFC control cells (BSU, ACP, BSL and BSR) are carried on non-QFC connections (see Section 8), they are excluded from QFC counters for the QFC VCC/VPC and the logical link, regardless of the control cell's PTI value. All non-QFC cells are excluded from the logical link counters.

All ATM Operations And Maintenance (OAM) cells on a QFC connection shall be treated as QFC cells. They shall be accounted for in the FCVC and FCVL counters.

A tunnel's VPC-level OAM cells (OAM F4 flows) are not included in the tunnel's QFC counters because the tunnel's VPC OAM cells flow on a non-QFC VCI. However, for a QFC VPC (on a physical link), VPC-level OAM cells are included in the (physical) link-level QFC counters.

Resource Management cells not associated with the QFC protocol (for instance, ATM Forum Explicit Rate RM cells) shall also be treated as QFC cells. Specifically, RM cells whose Protocol_ID field does not match the QFC protocol_ID shall be treated as QFC cells according to the rules established by this specification in Sections 6, 7, 9, and 10.

The call setup procedures in Section 9.2 outline the specific connection establishment and counter synchronization procedures that ensure that proper connection and logical link QFC cell accounting is performed.

7.1.2 Buffer State Update Records

Receivers send Buffer State Update (BSU) records contained in BSU messages on the reverse direction link to the transmitter using a reserved control channel. Each BSU message can contain from 1 to 7 BSU records. Each BSU record identifies a connection or the link and contains a snapshot from the receiver of the Fwd_Counter[i] or Fwd_Counter[link], as appropriate.

A receiver that has forwarded $N2_{vc}$ cells for a connection shall compose a BSU record that shall be packed into a BSU message to be sent to the transmitter. Likewise, a receiver which has forwarded $N2_{link}$ cells for all QFC connections carried on the logical link shall compose a BSU record for the link that shall also be packed into a BSU message to be sent to the transmitter[4]. The $N2_{vc}$ or $N2_{link}$ counters (N2_Counter[i] and N2_Counter[link]) are set to zero when the BSU record is composed.

Connection and link BSU records may be mixed in a BSU message. Also, more than one BSU record from the same connection or from the link may be stored in the same BSU message but, if so, the more recent BSU record (the one with the larger Fwd_Counter[i or link]) must be in the sequentially lower sector of the message so that any transmitter sequentially unpacking the message will get the more recent update last. However, it is recommended (but not required) that the receiver replace any "older" connection updates with any newer ones in order to increase BSU message packing efficiency. This will increase BSU update bandwidth efficiency.

It is recommended that the value of BSU record Fwd_Counter[i] and Fwd_Counter[link] be the actual Fwd_Counter[i or link] at the time the BSU record is composed by the receiver.

A node may compose a BSU record before forwarding $N2_{vc}$ (or $N2_{link}$) QFC cells as long as the transmitter remains in compliance with the maximum control bandwidth overhead constraint established by Table 10 in Section 9.3 and Appendix D.

The counters stored in BSU records are unsigned integers.

7.1.3 Cell Forwarding Criteria

Transmitters may send QFC cells, thus increasing Tx_Counter[i] and Tx_Counter[link], only when the result of doing so satisfies both constraints in Equation [1]. BSU_Fwd_Count[i] and BSU_Fwd_Count[link] are the values most recently received in BSU records from the receiver.

4 As a rule, $N2_{link}$ will generally be many times greater then $N2_{vc}$. Therefore, the update bandwidth overhead of link BSUs is typically much smaller then the aggregate connection BSU overhead.

The subtraction of the unsigned counters for the connection [i] in Equation [1] is unsigned Mod_{2^D} subtraction, where D is equal to 28 in the LL option and 8 in the SL option. The subtraction of the unsigned counters for the link ([link]) in Equation [1] is unsigned $Mod_{2^{28}}$ subtraction[5].

For each cell to be transmitted on connection [i], check:

$$Mod_{2^D}\ (Tx\_Counter[i] - BSU\_Fwd\_Count[i]) \le Limit[i]$$

$$Mod_{2^{28}}\ (Tx\_Counter[link] - BSU\_Fwd\_Count[link]) \le BSL\_Limit_{link}$$

Equation [1]: FCVC & FCVL Transmitter Constraints

The BSL_Limit_{link} number is established by the receiver and communicated to the transmitter via Buffer State Limit (BSL) messages defined in Section 10.5 that are managed via the BSL Protocol (BSLP) defined in Section 7.3. The Limit[i] number is established by the receiver as a default BSL_Limit_{vcdef} that is communicated to the transmitter via BSL messages defined in Sections 10.5 and 7.3. Optionally, in the Long Link QFC option, the receiver may override the automatic assignment of the BSL_Limit_{vcdef} default to a connection Limit[i] with an override value established through the signaling procedures defined in Section 9.2.4.

It is recommended that a receiver not allow the value of Fwd_Counter[i] or Fwd_Counter[link] decrease from one BSU record update to another (except, of course, when the corresponding Fwd_Counter wraps). However, the receiver may be forced to do so in response to BSC cells. In particular, a receiver may receive a BSC cell with a Tx_Counter value less than its Rx_Counter value. While this case may be very unlikely, the receiver is allowed to decrease the corresponding Fwd_Counter in this case. As a result of the BSU record generated in response to the BSC, the transmitter may observe a decrease in BSU_Fwd_Count. We allow a decrease in this case, provided that it is bounded by the corresponding Limit[i] or Limit[link].

Any QFC node may set the EFCI congestion indication in an ATM data cell (PTI = 0 or 2 to PTI 1 or 3, as appropriate). The policies for setting this bit are implementation dependant.

7.2 Quantum Flow Control Robustness Procedures—Buffer State Checks (BSC)

Buffer State Checks (BSC) are sent periodically to compensate for any discrepancy between the cell counters in the transmitter and receiver due to link transmission errors. Transmitters send connection segment BSC records, containing the transmitter's Tx_Counter[i] or Tx_Counter[link], to the receiver to allow the receiver to match this value against its forward and receive counters. For correctness, BSC records must remain in sequence with the QFC cell stream for the connection segment. Upon receipt of a BSC record, a receiver shall adjust the counters Rx_Counter[i] and Fwd_Counter[i] according to Equation [2] or

[5] Unsigned $Mod_m(x\text{-}y)$ is defined as the unsigned integral remainder of the operation (m+x-y)/m. The comparison of the modulo operation with the Limit[i] or BSL_Limit_{link} is an unsigned comparison operation.

Rx_Counter[link] and Fwd_Counter[link] according to Equation [3] as appropriate.

For each check received for connection [i], sequentially update:

$$\text{Fwd\_Counter[i]} \leftarrow \text{Mod}_{2^D}\,(\text{Fwd\_Counter[i]} + \text{Mod}_{2^D}\,(\text{BSC(Tx\_Counter[i])} - \text{Rx\_Counter[i]}))$$

$$\text{Rx\_Counter[i]} \leftarrow \text{BSC(Tx\_Counter[i])}$$

Equation [2]: Receiver Connection Buffer State Check (BSC) Update Equations

For each check received for the logical link, sequentially update:

$$\text{Fwd\_Counter[link]} \leftarrow \text{Mod}_{2^{28}}\,(\text{Fwd\_Counter[link]} + \text{Mod}_{2^{28}}\,(\text{BSC(Tx\_Counter[link])} - \text{Rx\_Counter[link]}))$$

$$\text{Rx\_Counter[link]} \leftarrow \text{BSC(Tx\_Counter[link])}$$

Equation [3]: Receiver Link Buffer State Check (BSC) Update Equations

The receiver shall respond to a BSC with a BSU record containing an up-to-date Fwd_Counter[i] or Fwd_Counter[link], as appropriate, even if the receiver has forwarded less than N2 cells since the last BSU record for the connection segment or the link.

Refer to Appendix D for a discussion about how often BSCs should be sent.

7.3 Buffer Management—Buffer State Limits (BSL)

The values of $\text{Limit}_{\text{vcdef}}$ and $\text{Limit}_{\text{link}}$ variables are determined by the receiver and communicated to the transmitter in a BSL Indication message described in Section 10.5. Copies of $\text{Limit}_{\text{vcdef}}$ and $\text{Limit}_{\text{link}}$ are stored in the $\text{BSL\_Limit}_{\text{vcdef}}$ and $\text{BSL\_Limit}_{\text{link}}$ variables at the transmitter. The transmitter acknowledges the limits by sending a BSL Confirmation message to the receiver according to the BSL Protocol described in this section.

$\text{Limit}_{\text{vcdef}}$, $\text{Limit}_{\text{link}}$, $\text{BSL\_Limit}_{\text{vcdef}}$, $\text{BSL\_Limit}_{\text{link}}$, and Limit[i] variables in both the transmitter and receiver shall have 1 bit less magnitude than the FCVC and FCVL counters. In the SL option, the $\text{Limit}_{\text{vcdef}}$, $\text{BSL\_Limit}_{\text{vcdef}}$, and Limit[i] shall be 7 bits and in the LL option shall be 27 bits. The $\text{Limit}_{\text{link}}$ and $\text{BSL\_Limit}_{\text{link}}$ shall be 27 bits. The reduced magnitude for QFC limits is required for robust behavior when counters wrap in Equation [1] in the presence of ATM cell bit errors that are undetected by the ATM cell header HEC.

During logical link initialization, a receiver shall generate a BSL record containing a value for $\text{Limit}_{\text{link}}$ and send the request to the transmitter in a BSL Indication message. Similarly, prior to the start of connection setup operations, a receiver shall generate a BSL Indication record containing a $\text{Limit}_{\text{vcdef}}$, and send the request to the transmitter in a BSL Indication message.

The $\text{Limit}_{\text{vcdef}}$ is a default limit for all connections [i] between a transmitter and a receiver. The $\text{Limit}_{\text{link}}$ is the limit for the forward direction logical link between a transmitter and a receiver. $\text{Limit}_{\text{vcdef}}$ and $\text{Limit}_{\text{link}}$ do not have to be the same value in both directions of a bi-directional logical link.

The receiver may change the default connection $Limit_{vcdef}$ or the $Limit_{link}$ values subsequent to initialization (including after QFC connections have been set up) if desired.

When setting the default connection or link limits, the conforming receiver shall set the Limit in the BSL Indication to be greater than, or equal to, the minimum cell buffers established in Equations [4] and [5], respectively. A receiver is considered non-conforming if it proposes a $Limit_{vcdef}$ or $Limit_{link}$ that is less than the $Limit_{vcmin}$ or $Limit_{linkmin}$ of Equations [4] or [5] as appropriate.

$$Limit_{vcmin} = N2_{vc} + 1$$

$$Limit_{vcmax} = \begin{bmatrix} 2^7 - 1 & \text{if SL option} \\ 2^{27} - 1 & \text{if LL option} \end{bmatrix}$$

Equation [4]: Minimum and Maximum Connection $Limit_{vcdef}$ Constraints

Receiver minimum $Limit_{link}$ constraint to avoid deadlock:

(For explanation of deadlock condition see Appendix E.)

C = Number of **established** connections

$$Limit_{linkmin} = N2_{link} + C + 1$$

$$Limit_{linkmax} = 2^{28} - 1$$

Equation [5]: Minimum and Maximum $Limit_{link}$ Constraint

The notation LLCR(forward) in Equation [5] indicates the LCR on the forward direction physical link or the forward direction VPC PCR bandwidth for a QFC tunnel depending on link type.

The number of connections C in Equation [5] is the number of connections that are **fully established**. A connection is fully established if the SETUP/CONNECT operations of the ATM signaling protocol have progressed to a point where cell flow operations are supported in the node's hardware (Section 9.2). The $Limit_{linkmin}$ in Equation [5] is established to avoid the N2 sharing deadlock problem described in Appendix E.

Upon receipt of a BSL Indication, the transmitter shall send a BSL Confirmation message to the receiver that contains the confirmation of the BSL_Limit_{vcdef} or BSL_Limit_{link}, as appropriate. When a receiver requests a lower $Limit_{link}$, the transmitter may optionally increase the BSL Indication's proposed $Limit_{link}$ to a larger number as long as the increased $Limit_{link}$ does not exceed the BSL_Limit_{link} currently established at the transmitter. At logical link initialization, the transmitter shall not increase the BSL Indication's proposed $Limit_{link}$. The transmitter shall not lower the BSL Indication's $Limit_{link}$. The conforming transmitter shall return BSL Confirmations in the same sequence as the BSL Indications arrive on the logical link.

The transmitter shall start using the new BSL_Limit_{vcdef} or BSL_Limit_{link} value immediately as the default for a connection or link, as appropriate. Furthermore, the transmitter shall replace Limit[i] with BSL_Limit_{vcdef} for all connections where Limit[i] was not overridden by the signaling procedures of Section 9.2.4.

In the case of buffer limit reduction, the transmitter shall ensure that no more cells will be transmitted on the connection or link, as appropriate, until the constraints in Equation [1] are satisfied.

Having received a BSL Confirmation, the receiver shall replace the current $Limit_{vcdef}$ and Limit[i] (as necessary, for connections that do not have connection limit overrides) or $Limit_{link}$ with new the BSL Confirmation's BSL_Limit_{vcdef} or BSL_Limit_{link} as appropriate. If the current limit is larger than the confirmation's BSL_Limit, then the receiver may reuse any buffers according to the reclamation policy established by Equations [7] and [8] and the receivers and transmitters state-transition tables (Tables [2] and [3] below), respectively. (Equation [6] is meant as an aid to understand Equations [7] and [8].)

In the case of buffer reduction, the receiver can also expect that — after the receipt of the BSL confirmation message — no more cells will arrive on the connection segment(s) or Link until the occupancy becomes less than the new $Limit_{vcdef}$ or $Limit_{link}$ as appropriate.

Connection Buffer Occupancy:

$$\mathrm{Mod}_{2^D}\,(\mathrm{Rx\_Counter[i]} - \mathrm{Fwd\_Count[i]})$$

Link Buffer Occupancy:

$$\mathrm{Mod}_{2^{28}}\,(\mathrm{Rx\_Counter[link]} - \mathrm{Fwd\_Count[link]})$$

Equation [6]: Determining Current Buffer Occupancy for a Connection or Link

$$\text{Reclaim Buffers for Connection [i]} \Leftrightarrow (\mathrm{Limit[i]} \le \mathrm{Mod}_{2^D}\,(\mathrm{Rx\_Counter[i]} - \mathrm{Fwd\_Count[i]}))$$

Equation [7]: Buffer Reclamation Policy When BSL_Limit_{vcdef} < $Limit_{vcdef}$

$$\text{Reclaim Link Buffers} \Leftrightarrow (\mathrm{Limit}_{link} \le \mathrm{Mod}_{2^{28}}\,(\mathrm{Rx\_Counter[link]} - \mathrm{Fwd\_Count[link]}))$$

Equation [8]: Buffer Reclamation Policy When BSL_Limit_{link} < $Limit_{link}$

The BSL Indication and Confirmation messages may be lost due to link bit errors. If the receiver has not received a response within 10 seconds after the Indication was sent, then it shall re-send a BSL Indication message. The conforming QFC receiver must not change the value of the $Limit_{vcdef}$ or $Limit_{link}$ in the re-sent Indication until the original BSL transaction is completed by acknowledgment from the transmitter through a BSL Confirmation. BSL Indication retransmission shall be repeated until a proper BSL Confirmation is received by the receiver or the link is declared out of service.

The state-transition tables in Table 2 (receiver) and Table 3 (transmitter) describe the transmitter and receiver portions of the BSL Protocol. The notations "Indicate" and "Confirm" are used to represent the BSL Indication and Confirmation, respectively. Limit_{new} and Limit_{cur} represent the new and current values of the Limit counters for a link or a connection segment.

Table 2. Buffer State Limit (BSL) Protocol Receiver Behavior

Current State	Events Causing Transition to New State		
	IDLE	**ACTIVE**	**RECLAIM**
IDLE	**Event**: (**duplicate same value**) Rcv confirm(BSL_Limit_{new}) if Limit_{cur} == BSL_Limit_{new} **Action**: ignore duplicate **Event**: (**duplicate out of range**) Rcv confirm(BSL_Limit) && [DUP. OUT OF RANGE ERROR][1] **Action**: ERROR—see Appendix C.	**STATE**⟨IDLE→ACTIVE⟩ **Event**: (**START**) Receiver wants to change Limit_{cur} to Limit_{new}. **Sequential Actions**: (1) Limit_{save}←Limit_{new} (2) Send Indicate(Limit_{new}), (3) TIMER←10sec.	**STATE**⟨IDLE→RECLAIM⟩ **Event**: (**duplicate, lowered limit**) Rcv confirm(BSL_Limit) [DUPLICATE LOWERED LIMIT][2] **Action**: $\text{Limit}_{saveBSL}$←BSL_Limit
ACTIVE	**STATE**⟨ACTIVE→IDLE⟩ **Event**: (**no reclamation required**) Rcv confirm(BSL_Limit) && [NO RECLAMATION NEEDED][3] **Sequential Actions**: (1) Stop TIMER (2) $\text{Limit}_{saveBSL}$←BSL_Limit (3) Limit_{cur}←$\text{Limit}_{saveBSL}$ **Event**: (**out of range**) Rcv confirm(BSL_Limit) && [OUT OF RANGE ERROR][4] **Sequential Actions**: (1) Stop TIMER (2) ERROR—see Appendix C.	**Event**: (**no response**) TIMER timed out. **Sequential Actions**: (**re-send Indicate**) (1) Send Indicate(Limit_{save}), (2) TIMER←10sec.	**STATE**⟨ACTIVE→RECLAIM⟩ **Event**: (**reclamation required**) Rcv confirm(BSL_Limit) && [RECLAMATION NEEDED][5] **Sequential Actions**: (1) Stop TIMER (2) $\text{Limit}_{saveBSL}$←BSL_Limit

Table 2. Buffer State Limit (BSL) Protocol Receiver Behavior

Current State	Events Causing Transition to New State		
	IDLE	**ACTIVE**	**RECLAIM**
RECLAIM	**STATE**⟨RECLAIM→IDLE⟩ **Event**: (**buffers reclaimable**) [ALL BUFFERS RECLAIMABLE][6] **Sequential Actions**: (1) **Reclaim**: ($Limit_{cur}$ - $Limit_{saveBSL}$) (2) $Limit_{cur} \leftarrow Limit_{saveBSL}$ **Event**: (**duplicate out of range**) Rcv confirm(BSL_Limit) && [DUP. OUT OF RANGE ERROR][1] **Sequential Actions**: ERROR—see Appendix C.	**STATE**⟨RECLAIM→ACTIVE⟩ No Transition	**Event**: (**duplicate, lowered limit**) Rcv confirm(BSL_Limit) [DUPLICATE LOWERED LIMIT][2] **Action**: $Limit_{saveBSL} \leftarrow$ BSL_Limit **Event**: (**duplicate same value**) Rcv confirm(BSL_Limit) && $Limit_{saveBSL}$ == BSL_Limit **Action**: ignore duplicate

1 [DUP(licate) OUT OF RANGE ERROR]:
($Limit_{save}$ > BSL_Limit) ‖ ($Limit_{saveBSL}$ > BSL_Limit)

2 [DUPLICATE LOWERED LIMIT]:
($Limit_{save}$ < BSL_Limit < $Limit_{saveBSL}$)

3 [NO RECLAMATION NEEDED]:
$Limit_{cur} \leq Limit_{save}$ (note: in this case, $Limit_{save}$ == BSL_Limit)

4 [OUT OF RANGE ERROR]:
($Limit_{save}$ > BSL_Limit) ‖ ($Limit_{save} < Limit_{cur}$ && $Limit_{cur}$ < BSL_Limit_{new})

5 [RECLAMATION NEEDED]:
$Limit_{save} < Limit_{cur}$ && BSL_Limit < $Limit_{cur}$

6 See Equations [7] and [8].

Table 3. Buffer State Limit (BSL) Protocol Transmitter Behavior

Event: (**new** or **duplicate** $\mathbf{Limit_{new}}$[]) Rcv Indicate($Limit_{new}$) **Sequential Action**:[1] (1) BSL_Limit⟵($Limit_{new}$ ‖ $Limit_{ovr}$) (2) Send Confirm(BSL_Limit)

1 The establishment of a new limit may cause data to be held at the transmitter depending on the values of Tx_Counter[] and BSU_Fwd_Count[]. See Equation [1].

8. Reserved Control Channels (RCC)

The Reserved Control Channel (RCC) is set aside as an out-of-band QFC control channel — out-of-band in the sense that QFC control messages are sent on a channel that is not one of the QFC (normal data) connections instantiated on the link. The RCC is used by most (but not all) QFC control messages (forward and reverse direction).

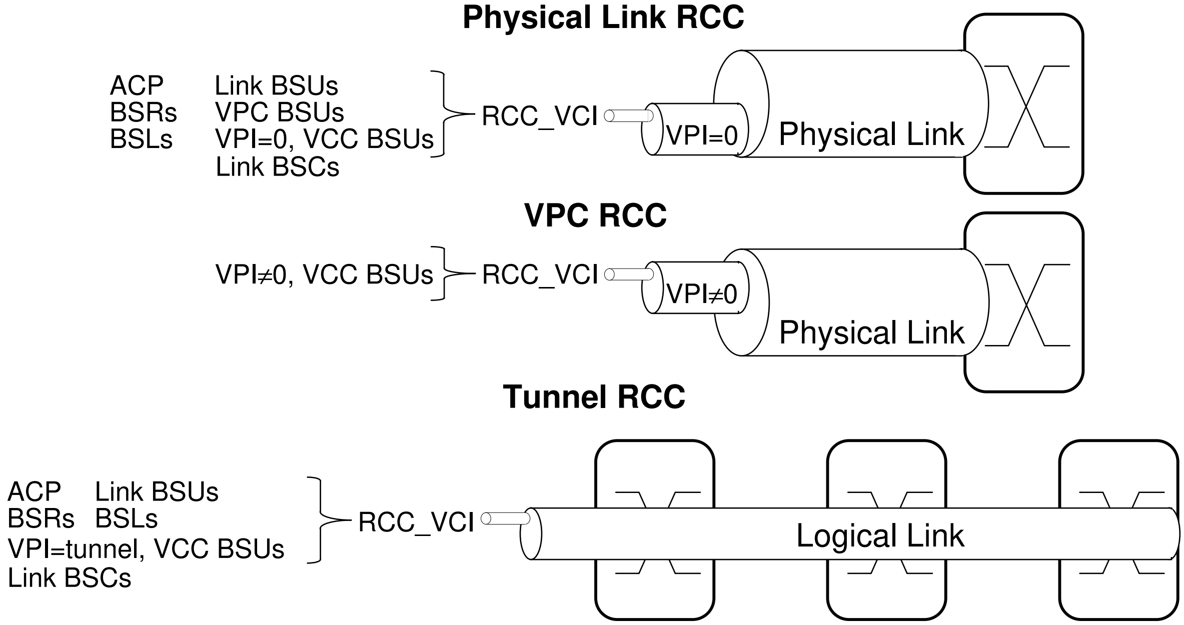


Figure 3. Reserve Control Channel Types

There are three types of RCCs: physical link RCCs (**PL-RCC**), tunnel RCCs (**T-RCC**) and VPC RCCs (**VP-RCC**). Each RCC type is distinguished by what QFC message types are transported over the RCC. Figure 3 shows these three types and lists the QFC messages that may flow over the RCC. Note that connection BSC messages are the only QFC message types that are not carried on an RCC channel. Connection BSCs are sent in-band with a connection's QFC cells as RM cells. See Section 10.4 for more information.

The PL-RCC and T-RCC are very similar and may be grouped as types of Logical Link RCCs (**LL-RCC**s). When convenient, the term LL-RCC will be used to refer to either the PL-RCC or T-RCC whenever the behavior is consistent between the two RCC types. The PL-RCC and T-RCC both carry ACP logical link initialization messages, BSR RTT measurement messages, BSL default limit messages, connection, link BSUs and link BSCs. The primary difference between them is that the PL-RCC supports BSU message transfer for switched VPCs whereas the T-RCC cannot support switched VPCs (only VCCs). A switched VPC at a physical link is treated by the QFC protocol just like a switched VCC. The BSU records for the switched VPCs (where the VPI $\neq$ 0) are sent on the PL-RCC because the RCC reserved VCI channel number (RCC_VCI) on the switched VPC may not be used by the QFC link because it is reserved for the end-to-end VPC.

The VPC RCC only carries VCC connection BSU messages. To distinguish the VCCs on different VPIs from one another, the VCC BSU's must be sent on a channel that is associated with the VPI, the VP-RCC. If they were all sent on the same PL-RCC, the VCI channel numbers could "collide." In most cases, VCCs will be assigned to VPI=0 and VP-RCCs will not be required.

The RCC_VCI is the same channel number for all RCC types. It is recommended that implementations support a programmable RCC_VCI within the range of 0-63 until ATM Forum and ITU-T standardization is completed. **The current RCC_VCI is assigned the value VCI=30_{10}=$1E_{16}$**.

Auto-configuration must be accomplished prior to QFC operations on a logical link. For this reason, the QFC-compliant node shall automatically open the LL-RCC in both the forward and reverse direction at logical link initialization when the node is ready to begin QFC operations. The compliant node shall also open any VP-RCCs as needed on a physical link, in both the forward and reverse direction, prior to establishing the first VCC with a VPI $\neq$ 0.

9. Initialization and Tear-Down Procedures

9.1 Logical Link Initialization and Reset Procedures

When two nodes are connected with a logical link (i.e., a physical link or tunnel), the nodes shall perform QFC Auto-Configuration prior to accepting any QFC connection setup requests on the link. Auto-configuration consists of seven configuration events: (1) determining the QFC Revision level, (2) determining the LL or SL option, (3) determining the N4 packing value, (4) establishing receiver $N2_{vc}$ and $N2_{link}$, (5) measuring the RTT, (6) establishing the $Limit_{link}$ and (7) establishing the default connection limit, $Limit_{vcdef}$. The first four configuration events, (1) through (4) above, are accomplished by the Auto-Configuration Protocol (ACP) RM messages. The fifth event is accomplished through a special mechanism to measure RTT, and the final two events are done with BSL messages. When these seven configuration events have been completed, the nodes may initiate the setup of the first connection segment.

9.1.1 Logical Link Auto-Configuration Protocol

Until this point, the focus of the specification has been on a uni-directional model of QFC. Auto-configuration is performed in the same uni-directional manner from a protocol perspective, but there are some parameters that are required to be adopted as the same value for both directions. Specifically, the Revision_ID and SL/LL options are configured the same for both uni-directional sides of a bi-directional logical link. The auto-configured values for N4, N5, $N2_{vc}$, $N2_{link}$, $Limit_{link}$ and $Limit_{vcdef}$ for a bi-directional logical link do not necessarily have to be the same for each uni-directional half.

The Auto-Configuration Protocol (ACP) Indication message is initiated by either node’s receiver attached to the bi-directional ATM link. This will be followed by an ACP Confirmation message returned from the transmitter.

When the ACP Confirmation message is sent by a transmitter, the values in the Confirmation message represent the final configured values of the uni-directional half of the QFC logical link. **To complete the auto-configuration procedures of the logical link, the receivers of both sides must execute the ACP protocol**.

The receiver shall select its preferred choices for the SL/LL Option and N4 and will set the highest QFC Revision_ID level that the node can support in the ACP Indication message.

The receiver will also select the desired BSU Bandwidth Overhead Divisor M, the $N5_{rcv}$ BSU processing delay, and the $N2_{link}$ value for the link or tunnel and place it in the Indication message. The receiver will format the ACP Indication message according to the formats described in Section 10.6 and send the message to the transmitter on the LL-RCC of the logical link.

The transmitter will select the highest Revision_ID level it can support or the Revision_ID level of the ACP Indication, whichever is lowest. This Revision_ID level will be used in all operations thereafter by the nodes, including the interpretation of the rest of the ACP message. All subsequent QFC messages must be marked with the selected Revision_ID level. The transmitter will place this Revision level into the ACP Confirmation message.

The transmitter will select the highest SL/LL option it can support or the value from the ACP Indication, whichever is lowest. The SL option is consider "lower" than the LL option.

The transmitter will select the highest N4 value that it can support or the value from the ACP Indication, whichever is lower. The allowable range of values is defined by Table 31 in Section 10.6.2. For physical links, this is fixed at seven and requires that any QFC-compliant transmitter be able to support the unpacking of at least seven BSU records. The auto-configured N4 is placed into the ACP Confirmation message. It applies only to the uni-directional reverse direction BSU updates for the transmitter to receiver half of the bi-directional QFC logical link.

Having selected the appropriate value of N4, the transmitter determines the $N2_{vc}$ value. The $N2_{vc}$ value has to be computed in such a way that the nominal BSU cell overhead defined in Table 10 can be maintained under nominal BSU record load. This requires that N4 and M be large enough to accommodate BSU record load due to:

- BSU records for a connection when $N2_{vc}$ cells have been forwarded,
- BSU records for the logical when at least $N2_{link}$ cells have been forwarded, and
- BSU records generated in response to BSC cells,

and still remain in compliance with the overhead constraint established by Table 10[6].

Given the nominal spacing of M cells between successive BSU cells (Section 9.3), each of the above cases contributes the following fraction of records to each BSU cell:

- $M/N2_{vc}$ records per BSU cell, since M forward direction QFC cells are nominally transmitted during a BSU interval and to sustain a rate of M QFC cells, up to $M/N2_{vc}$ BSU records may have to be generated.
- $M/N2_{link}$ records per BSU cell, for reasons similar to above[7].

[6] Extraneous BSU records, not covered by the three listed cases, may be injected into a BSU message if there are less than N4 records in a BSU message and the receiver has no other BSU records to send.

[7] For any QFC cell forwarded from a transmitter to a receiver, the cell is counted twice, once in the connection counters and once in the link counters. Therefore, if N4 =1, $N2_{vc}$=1 and $N2_{link}$=1 the reverse direction update bandwidth would be 2 x LCR.

- M/100 records per BSU cell, since the upstream transmitter may send as many as one BSC message per one hundred QFC cells (see Table 10), and the downstream receiver is required to generate one BSU **record** for each BSC **message** received.

Since each BSU cell contains up to N4 records, $N2_{vc}$ needs to be chosen such that N4 is large enough to sustain the load described above, that is: $N4 \geq M/N2_{vc} + M/N2_{link} + M/100$.

Solving this constraint for N2, we get: $N2_{vc} \geq M/(N4 - M/N2_{link} - M/100)$. The optimal value of $N2_{vc}$ is the smallest integer that satisfies the above constraint. Therefore, the transmitter computes $N2_{vc}$ using Equation [9].

$$N2_{vc_{cells/update}} = \left\lceil \frac{M_{cells/BSU}}{\left(N4_{updates/BSU} - \frac{M_{cells/BSU}}{N2_{link_{cells/update}}} - \frac{M_{cells/BSU}}{100_{cells/update}}\right)} \right\rceil$$

Equation [9]: Auto-configuring $N2_{vc}$ Given N4, M and $N2_{link}$.

M and $N2_{link}$ are supplied by the receiver in the ACP Indication message and may not be modified by the transmitter. The transmitter is constrained by the values of M and $N2_{link}$ and Equation [9] and may not increase change $N2_{link}$, M or $N2_{vc}$. The auto-configured $N2_{vc}$, the receiver's echoed M, and echoed $N2_{link}$ are placed into the ACP Confirmation message.

$N2_{vc}$, M, and $N2_{link}$ apply only to the uni-directional reverse direction BSU updates for the transmitter to receiver half of the bi-directional logical link.

Table 4 shows, for a minimum value of M, the range of allowable N4, and for two different values of $N2_{link}$, what the auto-configured $N2_{vc}$ would be. The first $N2_{link}$ value is designed to show what the minimum $N2_{vc}$ can be, given minimum M and N4. The second $N2_{link}$ value shows what $N2_{vc}$ would be if a policy was set such that $N2_{link}$ should be roughly ten times the value of $N2_{vc}$ (i.e. $N2_{link} \sim 10 \times N2_{vc}$). The reader may calculate other values of M,

$N2_{link}$ and $N2_{vc}$ using Equation [9]. For more information about $N2_{vc}$, $N2_{link}$, N4 and M, see Appendix B[8].

Table 4. Range of $N2_{vc}$ Given M, N4 and $N2_{link}$

Link Type	Range of M[1]	N4	$N2_{link}$[2]	$N2_{vc}$[3]	$N2_{link}$[4]	$N2_{vc}$[5]
	Minimum		Minimum $N2_{vc}$ Rule		$N2_{link}$ ~ 10 x $N2_{vc}$ Rule	
Tunnel	17	1	831	21	188	23
		2	131	10	60	11
		3	43	7	43	7
		4	40	5	40	5
		5	30	4	30	4
		6	105	3	11	4
Physical Link or Tunnel		7	15	3	15	3

1 Maximum 6% BSU requires ⌈100/6⌉ = 17 minimum M.
2 This value of $N2_{link}$ is designed to yield the minimum $N2_{vc}$ given M and N4.
3 This is the smallest $N2_{vc}$ can be, given M and N4.
4 This value of $N2_{link}$ is designed to yield an $N2_{vc}$ where $N2_{link}$ ~ 10 x $N2_{vc}$.
5 This is the smallest $N2_{vc}$ can be made given the policy of $N2_{link}$ ~ 10 x $N2_{vc}$.

The receiver puts its BSU processing delay ($N5_{recv}$) in the ACP Indication message, and the transmitter puts its BSU processing delay ($N5_{trans}$) in the ACP Confirmation message. The transmitter also echoes (but shall not modify) the $N5_{recv}$ of the receiver in the Indication message.

Having auto-configured all parameters of the ACP, the transmitter will send the ACP Confirmation to the receiver on the forward direction LL-RCC. The Revision_ID, SL/LL Option, N4, $N5_{rcv}$, $N5_{trans}$, $N2_{vc}$, $N2_{link}$ and M parameters become invariant while the link is in oper-

8 The symbols ⌈ ⌉ in Equation [9] mean: take the integer *ceiling* of the quotient.

ation. Table 5 summarizes the ACP message parameters and where each is sent or echoed by the receiver or transmitter, as appropriate.

Table 5. ACP Message Parameters

Sent by Receiver	Echoed by Transmitter	Sent by Transmitter
Receiver's Revision_ID		Logical Link Revision_ID
Receiver's SL/LL Option		Logical Link SL/LL Option
Receiver's "Best" Forward Direction N4		Auto-Configured Forward Direction N4
		Auto-Configured Forward Direction $N2_{vc}$
Forward Direction $N2_{link}$	$N2_{link}$	
Forward Direction M	M	
Forward Direction $N5_{recv}$	$N5_{recv}$	
		Forward Direction $N5_{trans}$

The ACP Indication and Confirmation messages may be lost due to link bit errors or because the transmitter either is not ready to receive QFC ACP messages or does not support QFC. A compliant QFC node shall re-send an ACP Indication message no more often than once every second and no less often than every 60 seconds until an ACP Confirmation message is received from the transmitter.

All ACP parameters are generally deterministic for two nodes at any instant in time. Therefore, while duplicate ACP Confirmation messages cause the receiver to start the QFC auto-configuration process over on the logical link, the outcome should be the same.

9.1.2 Measuring Link RTT

Knowledge of the RTT allows for optimal sizing of per-connection receiver buffers. Appendices B and E describe the relationship of RTT (N3), receiver buffer provisioning, and maintaining LLCR transmission rates.

The components of RTT include, for a logical link, link propagation delay (LPD) in the forward and reverse direction between the transmitter and receiver. RTT does not include N5 delay (Appendix B). QFC implementations should strive to measure RTT such that, as much as practical, variable cell pipeline and processing delay is minimized or eliminated in the RTT measurement process. Cell pipeline and processing delay for a logical link interface that is constant for all cells transmitted or received on a logical link may be included in RTT measurements.

9.1.2.1 Buffer State Round Trip Time Measurement Mechanism

The Round Trip Time (RTT) measurement operation is initiated by the transmitter by sending a Buffer State RTT (BSR) measurement message to the receiver which will be returned at some later time. The BSR message is time-stamped whenever it is injected onto, or removed from, a physical link in both directions in its round-trip loop from transmitter to receiver and back. When the BSR is returned to the transmitter, sufficient information will exist to deduce the transmitters RTT. Both QFC nodes must initiate this measurement so that each may determine RTT (independently). The measurement of RTT shall be accomplished prior to initializing $Limit_{link}$ and $Limit_{vcdef}$ and before the start of call setup operations by either node. (RTT may be measured periodically, if desired, to check that RTT has not changed significantly.)

The RTT measurement process has four resources: Tx_starttime, Rx_receivetime, Rx_forwardtime and Tx_stoptime. Figure 4 shows the juxtaposition of the four time-stamp resources in the QFC model. RTT time-stamps are implemented as asynchronous counters in the transmitter and receiver that have a time resolution of 250 nano-seconds.

The transmitter measures the absolute elapsed time with its Tx_starttime and Tx_stoptime and then subtracts the receiver's BSR hold time measured by the Rx_receivetime and Rx_forwardtime. With these resources, a transmitter's measured RTT can be determined by Equation [10]. Because the receiver time-stamps the BSR received time and the BSR forwarded time, the RTT process can be implemented in hardware or software, allowing implementation flexibility. The sequence number is used to provide robustness under BSR loss due to link bit errors.

At link initialization, just prior to transmission of a BSR to the receiver, the transmitter shall include a time-stamp in the message in the Tx_starttime. The BSR is sent to the receiver on the LL-RCC of the forward direction logical link.

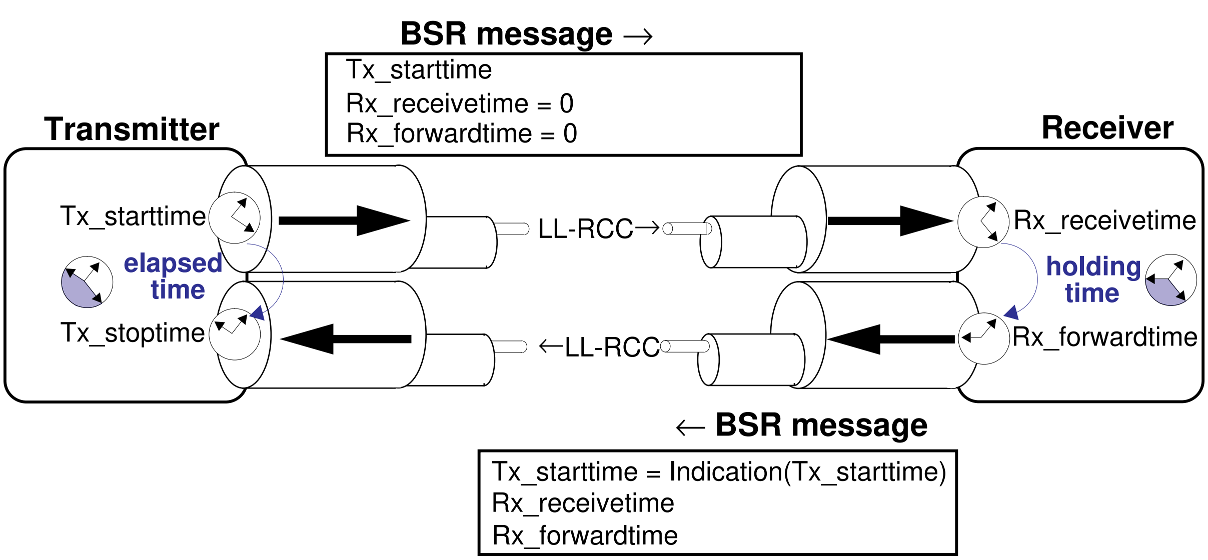


Figure 4. RTT Time-stamps

Receiver Hold Time:

$$\mathrm{Mod}_{2^{28}}(\mathrm{Rx\_forwardtime} - \mathrm{Rx\_receivetime})$$

Transmitter Elapsed Time:

$$\mathrm{Mod}_{2^{28}}(\mathrm{Tx\_stoptime} - \mathrm{Tx\_starttime})$$

$$\mathrm{RTT}_{nsecs} = \mathrm{Mod}_{2^{28}}(\mathrm{Tx\_stoptime} - \mathrm{Tx\_starttime}) - \mathrm{Mod}_{2^{28}}(\mathrm{Rx\_forwardtime} - \mathrm{Rx\_receivetime})$$

Equation [10]: Transmitter's Measured RTT

When the BSR Indication message arrives at the receiver, it shall be immediately time-stamped by the receiver in the Rx_receivetime field of the BSR. The BSR Indication shall be converted into a BSR Confirmation message by adjusting its Msg_Type field to indicate a confirmation message. At some later time, the receiver shall forward the BSR Confirmation message to the transmitter on the reverse direction LL-RCC. The receiver, just prior to transmission of the BSR to the transmitter, will time-stamp the BSR in the Rx_forwardtime field. When the BSR arrives at the transmitter, the transmitter will record the Tx_stoptime and complete the BSR/RTT measurement process. (Space has been reserved for the transmitter to put the Tx_stoptime time-stamp in the BSR Confirmation message, if desired.)

The transmitter shall set the Rx_receivetime and Rx_forwardtime time-stamp fields of the BSR Indication to zero. Since these fields are set to zero, the receiver may implement a fast turn-around of the BSR in hardware and avoid the receiver time-stamp overhead. The evaluation of Equation [10] is valid when using fast turn-around, so long as the receiver does not inject significant hardware processing delay when performing the loopback. For example, when using the fast turn-around mode, adding no more than 2 link cell-times (link cell time = 1/LLCR) to the RTT is considered a good, fast turn-around mechanism.

The conforming receiver shall not hold the BSR more than 10 seconds. If a transmitter receives a BSR Confirmation message where the transmitter's elapsed time (Equation [10]) is greater than 10 seconds, the BSR Confirmation shall be discarded and a BSR Indication shall be resent. If, after 10 seconds, a BSR Indication has not been confirmed by a BSR Confirmation, the transmitter shall resend the BSR according to the procedures set out above.

If a transmitter detects a receiver hold-time (Equation [10]) that is greater than the transmitters elapsed time, the transmitter shall discard the BSR Confirmation cell and resend the BSR according to the procedures set out above. After five such failures, the receiver shall be declared non-conforming. See Appendix C for the proper error response.

9.1.3 Initializing Limit_{link} and Limit_{vcdef}

When a link first goes into service and before the Auto-configuration Protocol is initiated, QFC nodes shall set Limit_{link}, Limit_{vcdef}, $\mathrm{BSL\_Limit}_{vcdef}$, $\mathrm{BSL\_Limit}_{link}$, and the receiver and transmitter's Limit[i] for all connections to zero. Then, after both sides have completed the execution of the ACP protocol and RTT measurement, both receivers will initialize Limit_{link}, Limit_{vcdef}, $\mathrm{BSL\_Limit}_{vcdef}$, $\mathrm{BSL\_Limit}_{link}$ through the procedures established by the BSL Protocol in Section 7.3. A receiver and transmitter's Limit[i] will be established during QFC connection setup operations as described in Section 9.2.

9.1.4 Link Reset

The receipt of an ACP Indication message after a logical link has established active QFC connections requires that QFC operations on a link be restarted. That is, the nodes will tear down any ABR connections using QFC procedures on the logical link and restart ABR operations as if the link were newly initialized.

If a physical link has lost physical synchronization, frame synchronization or incurs a high BER or for any other reason is declared out-of-service, the QFC operations on the logical link shall be re-started. The length of time of an outage required for a physical link to be declared out-of-service is not standardized. Likewise, if an OAM RDI or AIS is received on a tunnel or the tunnel VPC has a call clearing operation initiated, the QFC operations on the tunnel shall be re-started.

When the logical link is declared in-service at some later time, the receiver at either node may begin QFC operations again by sending out an ACP Indication message.

9.2 Connection Establishment and Tear-Down Procedures

This section describes modifications to the UNI 3.1 signaling specification required to support Quantum Flow Control (QFC) connections. The **signaling modifications are very simple** and restricted to:

• A minor augmentation to the Broadband Bearer Capability information element (BBC_IE) to specify creation of a QFC connection (see Section 10.8 for the BBC-IE code point).

• Extensions to the procedures to indicate when the various QFC initialization and tear down steps should occur relative to the normal UNI 3.1 call setup and call clearing procedures. The UNI 3.1 procedures themselves are unchanged.

• The addition of an optional information element (IE) to allow a signaling entity to set a Limit[i] override of $Limit_{vcdef}$ for a specific connection when using the Long Link option (see Section 10.8 for a description of the IE).

No new messages, call states, timers or cause codes are required.

9.2.1 Broadband-Bearer Capability Information Element

To signal the creation of a QFC connection, the QFC traffic type is specified in the Broadband Bearer Capability information element (BBC-IE). Section 10.8 delineates the actual code point value for the BBC-IE to indicate a QFC connection request.

9.2.2 ATM Traffic Descriptor Information Element

When signaling for QFC connections, the Peak Cell Rate CLP=0+1Traffic Parameter is used to indicate the desired maximum QFC bandwidth rate. If any other Traffic Parameters are specified, they may be ignored by the network or terminal.

When configuring the QFC algorithms, the Peak Cell Rate may be used to dimension buffers (see Appendices B and E). In some cases, such as long haul links, buffers may be restricted to limit the maximum flow of cells on the connection to a sustained rate equal to PCR.

If the maximum available bandwidth is desired on all links traversed by the QFC connection (LLCR at each hop), the Peak Cell Rate should be coded to its maximum specifiable value which is equivalent to specifying no bandwidth (unspecified bandwidth rate—UBR).

9.2.3 QFC Connection Limit_{vcdef} Override Information Element

The optional connection Limit_{vcdef} override Information Element (Limit-IE) supports a 28 bit unsigned integer used to specify a Limit[i] override of the default Limit_{vcdef} for a QFC connection. The IE is optional for the calling party (SETUP) or called party (CONNECT) but, if included, must be accepted by the called (SETUP) or calling party (CONNECT), as appropriate. The Limit-IE applies to a single direction only and has local significance only.

When the Limit-IE is included in a SETUP message, it indicates an override to be used in the **calling party** (receiver) to **called party** (transmitter) QFC forward[9] direction of the connection's logical link (marked as 1 in Figure 5). When specified in a CONNECT message, it indicates an override to be used in the **called party** (receiver) to **calling party** (transmitter) QFC forward direction of the connection's logical link (marked as 2 in Figure 5). The Limit-IE directionality is shown in Figure 5 for the case of a terminal initiated call (calling party to network) and a network initiated call (network to called party).

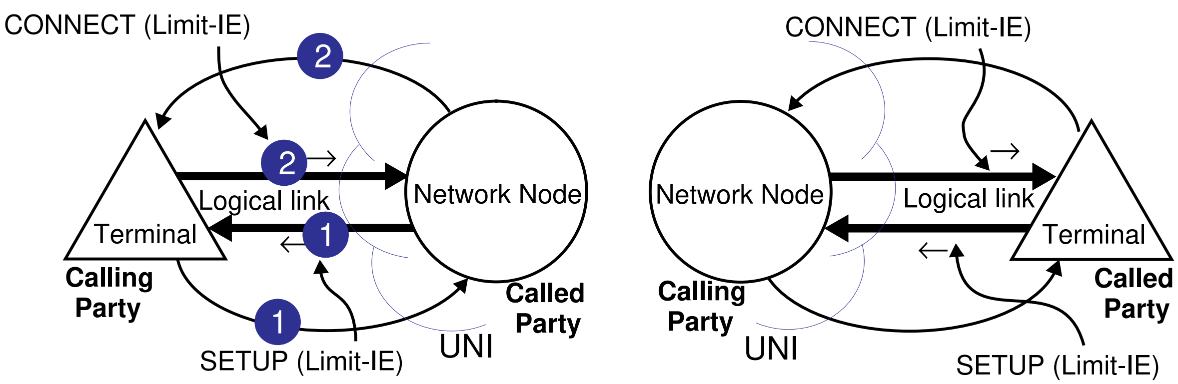


Figure 5. Limit-IE Directionality for SETUP and CONNECT

The purpose of providing the Limit_{vcdef} override is to allow a receiver to set individual connection limits. In a long-haul link, Limit_{vcdef} overrides can be used to enforce a sustained PCR by decreasing the number of N3 buffers assigned to the connection. By decreasing N3 — when N3 is sufficiently large — the network can enforce specific sustained peak rates[10] known as the dynamic bandwidth limit (DBL). For more information on the DBL, see Appendix E.

[9] The term forward in this section is used from the perspective of the QFC model and not from the perspective of the calling to called party forward and reverse directions.

[10] The sustained PCR is quantized by N2 and N4 and is not strictly paced.

9.2.4 Call Setup

When signaling for QFC connections, the procedures of Sections 5.5 and 5.6 of UNI 3.1 are used. Table 6 shows the QFC initialization steps that occur over the UNI between the **calling party** and the network. The QFC calling and called parties shall follow the QFC resource assignment procedures and the signaling message sequencing of Table 6.

The network opens up its receive side first, allowing the calling party to inject cells into the network after the CONNECT message is received by the calling party. The network shall not transmit QFC cells to the calling party terminal until it has received the CONNECT ACK message. The calling party and network take their initial QFC buffer allocation for the link from either: 1) Limit_{vcdef}, if no QFC Limit_{vcdef} override IE is specified or 2) from the QFC Limit_{vcdef} override, if the IE is specified.

Table 6. Calling Party Initiates QFC Call at Calling Party's UNI to Network

CALLING PARTY		**Message Sequences**	NETWORK UNI	
UNI Terminal Actions for connection [i]			UNI Network Actions for connection [i]	
Transmitter	Receiver		Transmitter	Receiver
The calling party does not yet know VPI/VCI		**SETUP** (to A from B) (QFC & PCR & (O) Limit_{ovr}) →	The network is unaware of pending call	
Closed TX side	**Closed** RCV side		SETUP Message transit time	
			(R) Select bi-directional VPI/VCI	
CALL PROCEEDING Message transit time		← **(O) CALL PROCEEDING** (VPI/VCI)	Tx_Counter=0 BSU_Fwd_Count=0 Limit[i]=0 (note [1])	Rx_Counter=0 N2_Counter=0 Fwd_Counter=0 Limit[i]=0 (note[1])
(O) reserve VPI/VCI				
Optional— Tx_Counter=0 BSU_Fwd_Count=0 Limit[i]=0 (note[1])	Optional— Rx_Counter=0 N2_Counter=0 Fwd_Counter=0 Limit[i]=0 (note[1])			
Closed TX side	**Closed** RCV side		**Closed** TX side	**Closed** RCV side
			Execute network algorithms to setup up QFC connection to called party.	
			Closed TX side	Limit[i] = Limit_{vcdef} || CONNECT(Limit_{ovr}) **Open RCV side**
CONNECT Message transit time		← **CONNECT** (VPI/VCI (O) Limit_{ovr})		Opened RCV side
(R) allocate VPI/VCI				
Tx_Counter=0 BSU_Fwd_Count=0 Limit[i] = BSL_Limit_{vcdef} || CONNECT(Limit_{ovr}) **Open TX side**	Rx_Counter=0 N2_Counter=0 Fwd_Counter=0 Limit[i] = Limit_{vcdef} || SETUP(Limit_{ovr}) **Open RCV side**			
Opened TX side	Opened RCV side	**CONNECT ACK** →	CONNECT ACK Message transit time	
			Limit[i] = BSL_Limit_{vcdef} || SETUP(Limit_{ovr}) **Open TX side**	Opened RCV side
Call Setup Complete				

[1] Setting the Limit[i] to 0 at the transmitter shuts off data forwarding operations (see Equation [1]). Limit[i] set to 0 at the receiver enforces transmitter compliance with the setup procedures (see Table C.2 in Appendix C). In both cases, until a Limit[i] value is established for a connection, any cells arriving on the connection segment shall be discarded without modifying the connection and logical link counters.

Table 7 shows the QFC initialization steps that occur at the network to **called party** interface. The sequence of events is similar to the calling party-to-network UNI interface except that the network allocates the VPI/VCI in advance, allowing each side — called party and network — to initialize the QFC counters and variables earlier and open their transmitter and receiver side interfaces aggressively. The called party opens its receiver and transmitter interfaces prior to sending the CONNECT message back to the network. The called party should be prepared to receive QFC cells from the network at any time after the CONNECT message is sent by the called party to the network.

ATM Forum and ITU-T signaling procedures require that the called party wait until the CONNECT ACK message is received before sending cells on the connection to the network. It is possible that data cells from the called party will be processed more quickly than the CONNECT message and even arrive at a switch or the calling terminal before the call is completely set up. QFC avoids this problem by buffering cells at each stage until the CONNECT ACK message arrives. QFC guarantees that up to Limit[i] cells can be sent to the network without cell loss.

The network, upon receipt of the CONNECT message from the called party, shall open its transmit and receive side interfaces.

Table 7. Network Initiates QFC Call at Network's UNI to Called Party

NETWORK UNI		**Message Sequences**	CALLED PARTY	
UNI Network Actions for connection [i]			UNI Terminal Actions for connection [i]	
Transmitter	Receiver		Transmitter	Receiver
(R) Select bi-directional VPI/VCI			Called Party not aware of call	
Tx_Counter=0 BSU_Fwd_Count=0 Limit[i]=0	Rx_Counter=0 N2_Counter=0 Fwd_Counter=0 Limit[i] = 0			
Closed TX side	Closed RCV side	**SETUP** (to A from B) (QFC & PCR & (O) Limit$_{ovr}$) (VPI/VCI) →	SETUP Message transmit time	
			(R) allocate VPI/VCI	
			Tx_Counter=0 BSU_Fwd_Count=0 Limit[i] = BSL_Limit$_{vcdef}$ ∥ SETUP(Limit$_{ovr}$) **Open TX side**	Rx_Counter=0 N2_Counter=0 Fwd_Counter=0 Limit[i] = Limit$_{vcdef}$ ∥ CONNECT(Limit$_{ovr}$) **Open RCV side**
CONNECT Message transmit time		← **CONNECT** ((O) Limit$_{ovr}$)	Opened TX side	Opened RCV side
Limit[i] = BSL_Limit$_{vcdef}$ ∥ CONNECT(Limit$_{ovr}$) **Open TX side**	Limit[i] = Limit$_{vcdef}$ ∥ SETUP(Limit$_{ovr}$) **Open RCV side**			
Opened TX side	Opened RCV side	**CONNECT ACK** →	CONNECT ACK Message transmit time	
			Opened TX side	Opened RCV side
Call Setup Complete				

Figure 6 demonstrates how the procedures of Tables [6] and [7] might be used in PNNI symmetric signaling. The ∅ symbol in Figure 6 indicates that both the transmitter and receiver side interfaces are closed. The ← or → symbols indicate that a receiver side interface has been opened (with directionality), and the ⇔ symbol indicates that both the transmitter and receiver side have been opened. The SETUP, CONNECT and CONNECT ACK messages are indicated in their relative juxtaposition in the call setup sequence end-to-end.

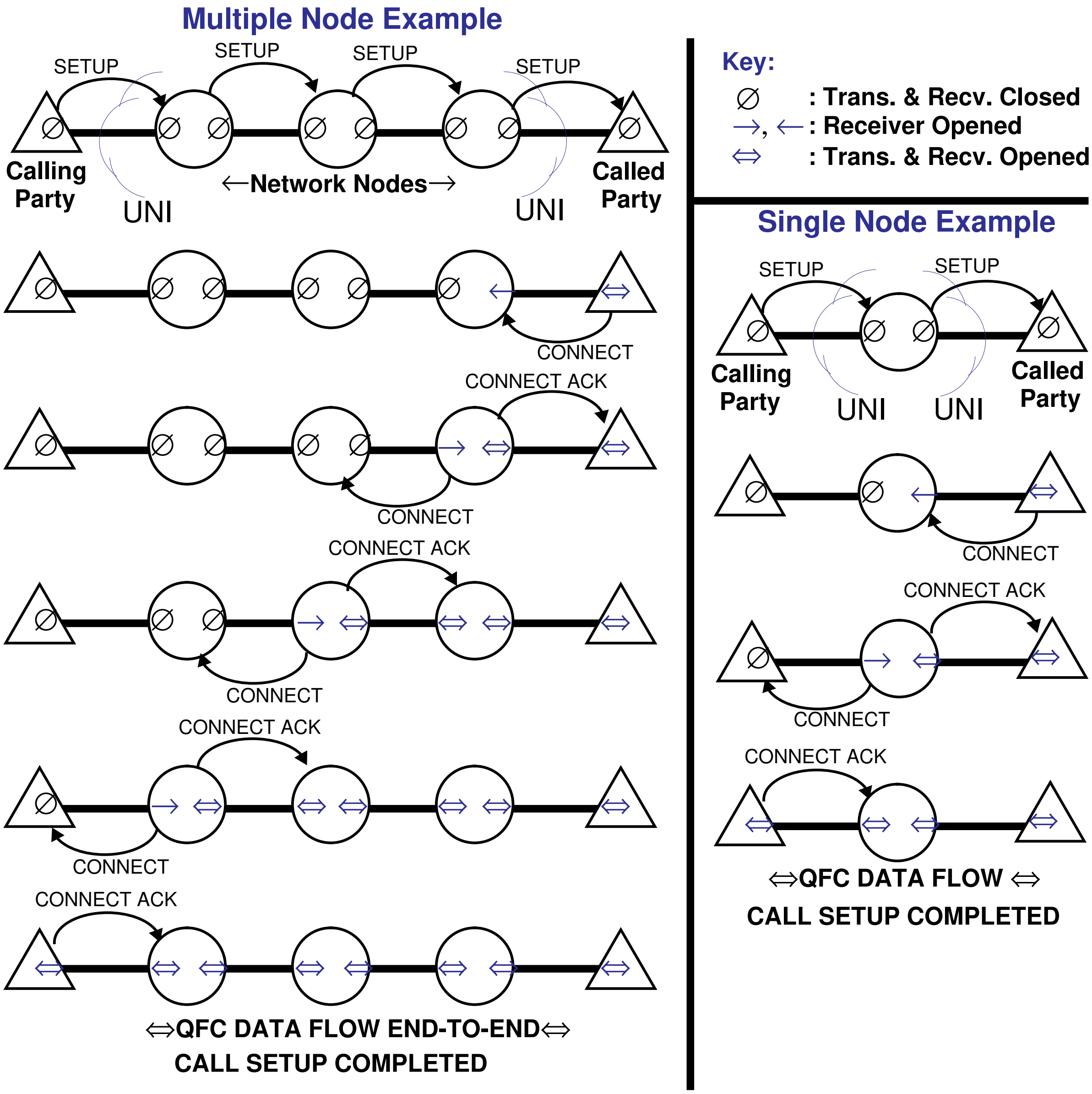


Figure 6. Call SETUP Sequencing With QFC Initialization End-to-End

9.2.5 Call Setup Error Handling

If an error occurs at any point during the establishment of a QFC connection, the UNI 3.1 error procedures should be followed. Prior to sending a RELEASE or RELEASE COMPLETE message, the QFC tear down steps indicated in Section 9.2.6 shall be completed.

9.2.6 Call Clearing

QFC call clearing occurs relative to the UNI 3.1 signaling messages as indicated below (Refer to Table 8):

- Prior to sending a RELEASE message (including as part of all error handling procedures), the User or Network shall:
  1) Stop transmitting and receiving cells on the QFC connection and discard all subsequent incoming cells received on the connection.
  2) Flush all buffered cells for the connection. Increment Fwd_Count[link] by the number of buffers freed so that they are available for other connections on the link (this is accounted for as if the flushed cells had been forwarded).
  3) Reclaim for the receiver and transmitter side, as appropriate, QFC resources.
- Upon receiving a RELEASE message and prior to sending the RELEASE COMPLETE message, the User or Network shall perform steps (1) through (3) above (if not already completed).

Table 8. Call Clearing Operation With Release Complete

Call Releasing Party		Message Sequences	Released Party	
UNI Call Releasing Actions for connection [i]			UNI Call Releasing Actions for connection [i]	
Transmitter	Receiver		Transmitter	Receiver
Tx_Counter = **r** BSU_Fwd_Count = **s** “transit” cells = (r-s)	Rx_Counter = **x** Fwd_Counter = **z** buffered cells = (x-z)		not aware of call release pending, data up to Limit[i] may be sent	
Close TX side	**Close RCV side**	RELEASE →	RELEASE Message transmit time	
BSU_Fwd_Count[link] = (Tx_Counter-BSU_Fwd_Count) Limit[i] = 0 Tx_Counter = 0 BSU_Fwd_Count = 0	Fwd_Counter[link] = (Rx_Counter - Fwd_Counter) Limit[i] = 0 Rx_Counter = 0 N2_Counter = 0 Fwd_Counter = 0	← RELEASE COMPLETE	**Close TX side**	**Close RCV side**
			BSU_Fwd_Count[link] = (Tx_Counter-BSU_Fwd_Count) Limit[i] = 0 Tx_Counter = 0 BSU_Fwd_Count = 0	Fwd_Counter[link] = (Rx_Counter - Fwd_Counter) Limit[i] = 0 Rx_Counter = 0 N2_Counter = 0 Fwd_Counter = 0
Call Teardown Complete				

If no RELEASE COMPLETE message is received by the User or Network in response to a RELEASE message, or if a RELEASE message is received in response to a RELEASE message (as indicated in Section 5.5.4.5 of UNI 3.1), no special QFC tear down actions are required since all tear down steps have already been taken (Refer to Table 9).

Table 9. Call Clearing Operation With Simultaneous Release

Call Releasing Party		Message Sequences	Simultaneous Call Releasing	
UNI Call Releasing Actions for connection [i]			UNI Call Releasing Actions for connection [i]	
Transmitter	Receiver		Transmitter	Receiver
Tx_Counter = r BSU_Fwd_Count = s “transit” cells = (r-s)	Rx_Counter = x Fwd_Counter = z buffered cells = (x-z)	RELEASE RELEASE	Tx_Counter = r’ BSU_Fwd_Count =s’ “transit” cells = (r’-s’)	Rx_Counter = x’ Fwd_Counter = z’ buffered cells =(x’-z’)
Close TX side	**Close RCV side**		**Close TX side**	**Close RCV side**
BSU_Fwd_Count[link]= (Tx_Counter-BSU_Fwd_Count) Limit[i] = 0 Tx_Counter = 0 BSU_Fwd_Count = 0	Fwd_Counter[link] = (Rx_Counter - Fwd_Counter) Limit[i] = 0 Rx_Counter = 0 N2_Counter = 0 Fwd_Counter = 0	RELEASE COMPLETE (optional) RELEASE COMPLETE (optional)	BSU_Fwd_Count[link]= (Tx_Counter-BSU_Fwd_Count) Limit[i] = 0 Tx_Counter = 0 BSU_Fwd_Count = 0	Fwd_Counter[link] = (Rx_Counter - Fwd_Counter) Limit[i] = 0 Rx_Counter = 0 N2_Counter = 0 Fwd_Counter = 0
Call Teardown Complete				

Figure 7 demonstrates how the procedures in Table 8 might be used in PNNI symmetric signaling. The ∅ symbol in Figure 7 indicates that both the transmitter and receiver side interfaces are closed. The ⇔ symbol indicates that both the transmitter and receiver side remain open. The RELEASE and RELEASE COMPLETE messages are indicated in their relative juxtaposition in the call teardown sequence end-to-end.

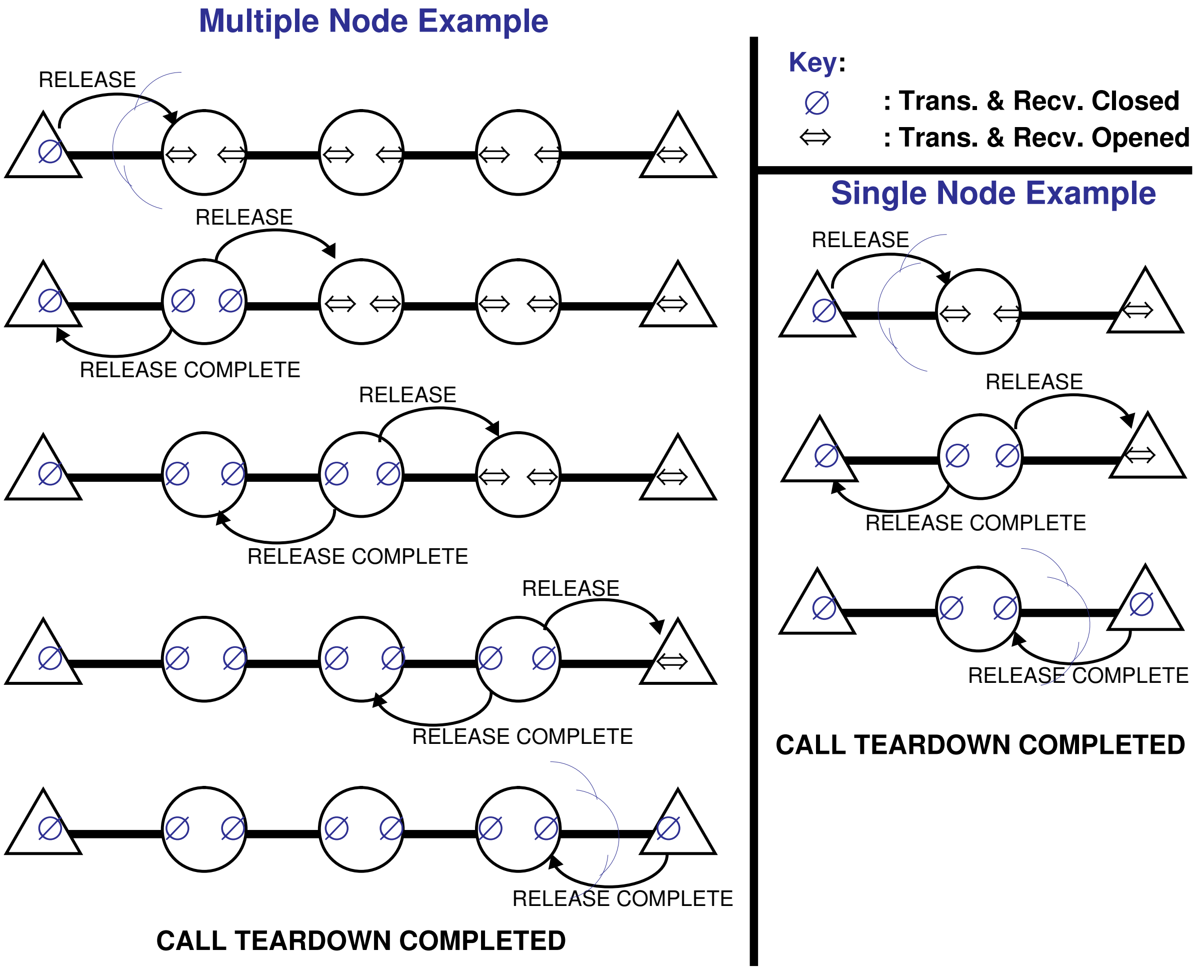


Figure 7. Call Teardown Sequencing with QFC De-allocation End-to-End

9.3 QFC Message Bandwidth Limits

The QFC protocol requires only a small percentage of the link bandwidth to communicate protocol information between a transmitter and receiver. Any QFC entity that exceeds the maximum sustained bandwidth limitations defined in Table 10 is considered non-conforming. Conformance bandwidth restrictions of Table 10 shall be measured as sustained rates over a period of 1 second.

The receiver, when sending BSU updates to the transmitter, shall not send (put on the wire) two BSUs with less than a minimum N4 logical link cell spacing interval. On a physical link, the cell spacing interval is contiguous cells at LCR[11]. On a tunnel (if on CBR) the cell spacing interval is N4 times the nominal PCR cell spacing interval. The nominal cell spacing interval will be M (of LCR or PCR as appropriate for a physical link or tunnel) to achieve the

[11] For this discussion we ignore the cell spacing perturbations caused by link framing techniques.

sustained bandwidth criteria of Table 10. If, at the end of an M interval, the receiver has a partially filled BSU cell, the receiver shall send the BSU cell to the transmitter.

Table 10. Protocol Bandwidth Limitations for QFC Messages

Message_Type	Nominal % of LCR or Tunnel PCR Bandwidth
ACP Indication	very small number
ACP Confirmation	
BSL Indication	≤ 0.5%
BSL Confirmation	
BSU	≤ 6.0%
BSC	≤ 1.0%
BSR Indication	very small number
BSR Confirmation	
Vendor Defined	Vendor < 7.5% - (Sum of BSL, BSU, BSC)

10. QFC Control Messages

QFC messages are contained in single 53-byte ATM control cells. This section describes all of the QFC control cells used in the protocol.

10.1 Cell Byte and Bit Ordering Rules

All QFC control cells use the same byte and bit ordering rules used by the ATM Forum and ITU-T. All multi-byte fields in cells are ordered big-endian. That is, the highest ordered byte appears in the first byte and the lowest ordered byte appears in the last byte of the field. Likewise, all bits in a byte or partial byte field are ordered big-endian such that the highest ordered bit appears to the left of the lower ordered bits. For notational purposes, the bits are labeled left-to-right as 8-to-1 and the bytes as 0-52 in cell descriptions in this section. This is in alignment with ITU-T and ATM Forum documentation.

10.2 Generic Control Cell Format

Table 11 illustrates the generic format of the QFC control cell. All control cells required to support the QFC protocol utilize this basic cell structure. The contents of specific fields may vary and the record formats may be different for different control cells. Specific QFC control cell formats and record formats will be the subject of subsequent sections.

Table 11. Generic Control Cell Format

offset	8	7	6	5	4	3	2	1	ID
0	GFC/VPI				VPI				ATM Cell Header
1	VPI								
2	VCI								
3					PT			CLP	
4	HEC								
5	Protocol_ ID								Payload Hdr
6	Message_Type				Revision_ID				
7	Reserved								
8	Reserved								
9-50	Record Data								Records
51	Reserved								Trailer
52	CRC-10								

10.2.1 ATM Cell Header

The values contained in the VPI, VCI, and PT fields of the control cell's ATM header are determined by the function of the cell. The PT field in the QFC control cell's ATM header is set to the value indicated in Table 12 for each type of control cell. The CLP bit is always set to 0_2 to indicate a non-CLP-marked cell. Also, when the cell is a UNI cell format, the GFC field is set as described in ITU-T recommendation I.371. Subsequent sections will elaborate on the determination of the VPI and VCI for each cell type.

Table 12. PT Field For Control Cell ATM Header

PT Field			QFC Control Cell Type
Binary	Decimal	Type	
000	0	Data	Auto-Configuration Protocol (ACP) Indication
000	0	Data	Auto-Configuration Protocol (ACP) Confirmation
000	0	Data	Buffer State Limit Indication (BSL Indication)
000	0	Data	Buffer State Limit Confirmation (BSL Confirmation)
110	6	RM	Buffer State Update (BSU)
110	6	RM	Buffer State Check (BSC)
110	6	RM	Buffer State RTT (BSR) Indication
110	6	RM	Buffer State RTT (BSR) Confirmation

10.2.2 Payload Header

The QFC control cell payload header contains three fields:

- Protocol_ID
- Message_Type
- Revision_ID

The Protocol_ID is used to uniquely identify this cell as a QFC control cell. This 8-bit field is always set to 00001010_2, (10_{10}). It is recommended that prior to ATM Forum and ITU-T standardization, that implementors of the QFC protocol keep the Protocol_ID value programmable. The Message_Type field identifies the type of QFC message the control cell contains.

This field may be set to one of the values in Table 13. The format of the control cells listed in Table 13 will be explained in later sections.

Table 13. QFC Control Cell Message_Type Values

Message_Type		QFC Control Cell Type
Binary	Decimal	
0000	0	Auto-Configuration Protocol (ACP) Indication
0001	1	Auto-Configuration Protocol (ACP) Confirmation
0010	2	Buffer State Limit Indication (BSL Indication)
0011	3	Buffer State Limit Confirmation (BSL Confirmation)
0100	4	Buffer State Update (BSU)
0101	5	Buffer State Check (BSC)
0110	6	Buffer State RTT (BSR Indication)
0111	7	Buffer State RTT (BSR Confirmation)
1xxx	8-15	Reserved for Vendor Specific Usage

The Revision_ID describes the version of the QFC protocol in use by the logical link end-points. After auto-configuration is complete, the Revision_ID is fixed and must be the same value in all subsequent messages during the operational life of a QFC logical link. Table 14 describes the legal values of the Revision_ID.

Table 14. QFC Revisions Defined by Revision_ID

Revision_ID		Revision Description
Binary	Decimal	
0000	0	Reserved
0001	1	First version of the QFC protocol.
0010-1111	2-15	Reserved

10.2.3 Record Packing

The maximum number of payload records in the generic control cell format is 7. For different Message_Types, however, this level of record packing may not be allowed. Table 15 summarizes the allowed record packing options for each Message_Type.

Table 15. Maximum QFC Control Cell Record Packing for Message_Types

Message_Type	Maximum Records	
	Physical Link	Tunneled Link
ACP Indication	N/A	
ACP Confirmation	N/A	
BSL Indication	1	
BSL Confirmation	1	
BSU	7	1-7
BSC	1	
BSR	N/A	
BSR	N/A	
Vendor Defined	1-7	

10.2.4 Control Cell CRC-10

A CRC-10 is generated for each QFC control cell covering the entire control cell payload field minus the last ten bits, which is the CRC-10 field. This CRC follows the same modality as ITU-T standard I.371. The control cell receiver must check this CRC and shall discard any QFC control cells if a CRC error is found.

10.2.5 Reserved Fields

Reserved fields may be set to all zeros; however, non-zeroed reserve fields are considered conforming.

10.3 Buffer State Update (BSU) Control Cells

BSU control cells consist of Indication cells only. They are sent from a receiver to a transmitter in the reverse direction on an appropriate RCC. The format of the BSU is detailed in Table 16. The same 6-byte BSU record format is used by all BSU types. The BSU control cell may have as few as one valid (non-null) record per cell or it may have multiple records up to an auto-configured N4 maximum defined by Table 15. All non-null records should appear in the topmost record locations. The first null record encountered in the cell shall terminate the processing of any additional records. If a specific QFC connection has more than one BSU record in a BSU cell, the most recent BSU record(s) shall be ordered in the BSU cell after the older BSU record. For example, if an older BSU record for connection [i] is in record position 2, the more recent records must be placed in record positions 3-7 as appropriate. The transmitter shall unpack the BSU cell in order of record 1 through 7 in sequence as appropriate. A receiver is considered non-conforming if it packs a cell beyond the auto-configured N4 limit. A transmitter may ignore any record in a cell that is packed beyond the N4 limit.

Table 16. Buffer State Update (BSU) Control Cell Format

<table>
<tr><th>offset</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>ID</th></tr>
<tr><td>0</td><td colspan="4">GFC/VPI</td><td colspan="4">VPI</td><td rowspan="5">ATM Cell Header</td></tr>
<tr><td>1</td><td colspan="4">VPI</td><td colspan="4"></td></tr>
<tr><td>2</td><td colspan="8">VCI</td></tr>
<tr><td>3</td><td colspan="4"></td><td colspan="3">PT = 110_2</td><td>0_2</td></tr>
<tr><td>4</td><td colspan="8">HEC</td></tr>
<tr><td>5</td><td colspan="8">Protocol_ID = 00001010_2</td><td rowspan="4">Payload Hdr</td></tr>
<tr><td>6</td><td colspan="4">Msg_Type=0100_2</td><td colspan="4">Revision_ID</td></tr>
<tr><td>7</td><td colspan="8">Reserved</td></tr>
<tr><td>8</td><td colspan="8">Reserved</td></tr>
<tr><td>9</td><td colspan="4">BSU_Type</td><td colspan="4">(record (1))</td><td rowspan="7">BSU Records</td></tr>
<tr><td>10</td><td colspan="8" rowspan="3">Fwd_Counter[i] or Fwd_Counter[link]</td></tr>
<tr><td>11</td></tr>
<tr><td>12</td></tr>
<tr><td>13</td><td colspan="8" rowspan="2">Connection ID [15:0]</td></tr>
<tr><td>14</td></tr>
<tr><td>15-50</td><td colspan="8">Records 2-7</td></tr>
<tr><td>51</td><td colspan="6">Reserved</td><td colspan="2"></td><td rowspan="2">Trailer</td></tr>
<tr><td>52</td><td colspan="8">CRC-10</td></tr>
</table>

10.3.1 BSU ATM Cell Header

The VPI/VCI of the ATM cell header is determined by Table 17 depending on whether the buffer state update is for a connection or a link.

Table 17. Determining VPI/VCI for a Buffer State Update (BSU) Control Cell

Update Type	Type	BSU Channel	
		VPI	VCI
Connection	VCC	VCC's VPI	RCC_VCI
	VPC	0	
Link	Physical	0	
	Tunnel	Tunnel's VPI	

10.3.2 BSU Record

Table 18 defines the values for the BSU_Type field which may relate to a connection segment Fwd_Counter[i] for a VCC or VPC, or a link Fwd_Counter[link] for a physical or logical link.

Depending on the auto-configuration option chosen — Short Link or Long Link — the Fwd_Counter[i] field will be 8 bits or 28 bits wide, respectively. This only applies if the BSU_Type field is a connection segment. If the BSU_Type field is a Link or Tunnel, then the Fwd_Counter[link] is always 28 bits wide. When the Fwd_Counter[i] is an 8 bit field, the 8 bits will reside in the low ordered 7:0 region of the Fwd_Counter field. Furthermore, when using 8 bit Fwd_Counter[i], the higher-order bits, 27:8, shall be maskable by the transmitter and receiver, that is, they do not have to be set to zero.

Table 18. Buffer State Update (BSU) Record BSU_Type Field Values

BSU_Type		Relates to:	Notes
Binary	Decimal		
0000	0	NULL Record	Marks a record as unused.
0001	1	VCC	Summation of bandwidth overhead for connections and Logical Links updates must be less than the maximum specified in Table 10 of Section 9.3.
0010	2	VPC	
0011	3	Physical Link	
0100	4	Tunnel	

Table 18. Buffer State Update (BSU) Record BSU_Type Field Values

BSU_Type		Relates to:	Notes
Binary	Decimal		
0101	5	Reserved	
0110	6	Reserved	
0111	7	Reserved	
1xxx	8-15	Vendor Defined	If used, combination of vendor defined plus rest of BSU must use bandwidth less than the maximum specified in Table 10 of Section 9.3.

The Connection ID field shall be set to the value appropriate for the update type listed in Table 19.

Table 19. Determining The Connection ID Field for BSU Control Cells

Update Type	Type	Connection Identifier
Connection	VCC	VCI of VCC
	VPC	VPI of VPC (padded left)
Link	Physical	0
	Tunnel	

10.4 Buffer State Check (BSC) Control Cells

BSC control cells consist of Indication cells only. They are sent from a transmitter to a receiver in the forward direction on the same channel as the connection or link cells rather than the RCC because the correct operation of the protocol requires that the BSC be in sequence with the QFC data cells. The format of the BSC is detailed in Table 20. The same 4-byte BSC record format is used by both BSC types.

Table 20. Buffer State Check (BSC) Control Cell Format

<table>
<tr><th>offset</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>ID</th></tr>
<tr><td>0</td><td colspan="4">GFC/VPI</td><td colspan="4">VPI</td><td rowspan="5">ATM Cell Header</td></tr>
<tr><td>1</td><td colspan="4">VPI</td><td colspan="4" rowspan="2">VCI</td></tr>
<tr><td>2</td><td colspan="4">VCI</td></tr>
<tr><td>3</td><td colspan="4">VCI</td><td colspan="3">PT = 110_2</td><td>0_2</td></tr>
<tr><td>4</td><td colspan="8">HEC</td></tr>
<tr><td>5</td><td colspan="8">Protocol_ID = 00001010_2</td><td rowspan="4">Payload Hdr</td></tr>
<tr><td>6</td><td colspan="4">Msg_Type=0101_2</td><td colspan="4">Revision_ID</td></tr>
<tr><td>7</td><td colspan="8">Reserved</td></tr>
<tr><td>8</td><td colspan="8">Reserved</td></tr>
<tr><td>9</td><td colspan="4">Check_Type</td><td colspan="4"></td><td rowspan="4">BSC Record</td></tr>
<tr><td>10</td><td colspan="8" rowspan="3">Tx_Counter[i] or Tx_Counter[link]</td></tr>
<tr><td>11</td></tr>
<tr><td>12</td></tr>
<tr><td>13-50</td><td colspan="8">Reserved</td><td></td></tr>
<tr><td>51</td><td colspan="6">Reserved</td><td colspan="2"></td><td rowspan="2">Trailer</td></tr>
<tr><td>52</td><td colspan="8">CRC-10</td></tr>
</table>

10.4.1 BSC ATM Cell Header

The VPI/VCI of the ATM cell header is determined by Table 21 depending on whether the buffer state update is for a connection or a link.

Table 21. Determining VPI/VCI for a Buffer State Check (BSC) Control Cell

<table>
<tr><th rowspan="2">Check Type</th><th rowspan="2">Type</th><th colspan="2">BSC Channel</th></tr>
<tr><th>VPI</th><th>VCI</th></tr>
<tr><td rowspan="2">Connection</td><td>VCC</td><td>VCC’s VPI</td><td>VCC’s VCI</td></tr>
<tr><td>VPC</td><td>VPC’s VPI</td><td>6_{10} (see note [1])</td></tr>
<tr><td rowspan="2">Link</td><td>Physical</td><td>0</td><td rowspan="2">RCC_VCI</td></tr>
<tr><td>Tunnel</td><td>Tunnel’s VPI</td></tr>
</table>

[1] This is the VP RM channel.

10.4.2 BSC Record

Table 22 defines the values for the BSC_Type field which may relate to a connection segment Tx_Counter[i] for a VCC or VPC, or a link Tx_Counter[link] for a physical link or tunnel.

Depending on the auto-configuration option chosen — Short Link or Long Link — the Tx_Counter[i] field will be 8 bits or 28 bits wide, respectively. This only applies if the BSC_Type field is a connection segment. If the BSC_Type field is a Link or Tunnel then the Tx_Counter[link] is always 28 bits wide. When the Tx_Counter[i] is an 8 bit field, the 8 bits will reside in the low ordered 7:0 region of the Tx_Counter field. Furthermore, when using 8 bit Tx_Counter[i], the higher-order bits, 27:8, shall be maskable by the transmitter and receiver, that is, do not have to be set to zero.

Table 22. Buffer State Check (BSC) Record Check_Type Field Values

Check_Type		Relates to:	Notes
Binary	Decimal		
0000	0	Reserved	
0001	1	VCC	Summation of bandwidth overhead for connection segments and Links must be less than the maximum specified in Table 10 of Section 9.3.
0010	2	VPC	
0011	3	Physical Link	
0100	4	Tunnel	
0101	5	Reserved	
0110	6	Reserved	
0111	7	Reserved	
1xxx	8-15	Vendor Defined	If used, combination of vendor defined plus rest of BSC must use less bandwidth than the maximum specified in Table 10 of Section 9.3.

10.5 Buffer State Limit (BSL) Control Cells

The BSL control cells consist of both Indication and Confirmation cells. Both types have the same format as Table 23.

Table 23. Buffer State Limit (BSL) Indication and Confirmation Format

<table>
<tr><th>offset</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>ID</th></tr>
<tr><td>0</td><td colspan="4">GFC/VPI</td><td colspan="4">VPI</td><td rowspan="5">ATM Cell Header</td></tr>
<tr><td>1</td><td colspan="4">VPI</td><td colspan="4"></td></tr>
<tr><td>2</td><td colspan="8">VCI</td></tr>
<tr><td>3</td><td colspan="4"></td><td colspan="3">PT = 000_2</td><td>0_2</td></tr>
<tr><td>4</td><td colspan="8">HEC</td></tr>
<tr><td>5</td><td colspan="8">Protocol_ ID = 00001010_2</td><td rowspan="4">Payload Hdr</td></tr>
<tr><td>6</td><td colspan="4">Msg_Type $(2 \text{ or } 3)_{10}$</td><td colspan="4">Revision_ID</td></tr>
<tr><td>7</td><td colspan="8">Reserved</td></tr>
<tr><td>8</td><td colspan="8">Reserved</td></tr>
<tr><td>9</td><td colspan="4">BSL_Type</td><td>Res.</td><td colspan="3"></td><td rowspan="5">BSL Record</td></tr>
<tr><td>10</td><td colspan="8" rowspan="3">Limit_{vcdef} or Limit_{link}</td></tr>
<tr><td>11</td></tr>
<tr><td>12</td></tr>
<tr><td>13-50</td><td colspan="8">Reserved</td></tr>
<tr><td>51</td><td colspan="6">Reserved</td><td colspan="2"></td><td rowspan="2">Trailer</td></tr>
<tr><td>52</td><td colspan="8">CRC-10</td></tr>
</table>

10.5.1 BSL ATM Cell Header

The BSL Indication and Confirmation cells are sent on the LL-RCC. The VPI/VCI of the ATM cell header is determined by Table 24 depending on whether the limit is for a connection or a link.

Table 24. Determining VPI/VCI for a Buffer State Limit (BSL) Control Cell

<table>
<tr><th rowspan="2">Limit Type</th><th rowspan="2">Type</th><th colspan="2">BSL Channel</th></tr>
<tr><th>VPI</th><th>VCI</th></tr>
<tr><td rowspan="2">Connection</td><td>VCC</td><td>VCC's VPI</td><td rowspan="4">RCC_VCI</td></tr>
<tr><td>VPC</td><td>0</td></tr>
<tr><td rowspan="2">Link</td><td>Physical</td><td>0</td></tr>
<tr><td>Tunnel</td><td>Tunnel's VPI</td></tr>
</table>

10.5.2 BSL Record

Table 25 defines the values for the BSL_Type field which may relate to a connection segment $Limit_{vcdef}$ for a VCC or VPC, or a $Limit_{link}$ for a physical link or tunnel. Depending on the auto-configuration option chosen — Short Link or Long Link — the $Limit_{vcdef}$ field will be 7 bits or 27 bits wide, respectively. This only applies if the BSL_Type field is a connection segment. If the BSL_Type field is a Link or Tunnel, then the $Limit_{vcdef}$ is always 27 bits wide. When the $Limit_{vcdef}$ is an 7 bit field, the 7 bits will reside in the low ordered 6:0 region of the Limit field. Furthermore, when using 7 bit $Limit_{vcdef}$, the higher-order bits, 27:7, shall be maskable by the transmitter and receiver, that is, they do not have to be set to zero.

Table 25. Buffer State Limit (BSL) Record BSL_Type Field Values

<table>
<tr><th colspan="2">BSL_Type</th><th rowspan="2">Relates to:</th><th rowspan="2">Notes</th></tr>
<tr><th>Binary</th><th>Decimal</th></tr>
<tr><td>0000</td><td>0</td><td>Reserved</td><td></td></tr>
<tr><td>0001</td><td>1</td><td>VCC</td><td rowspan="4">Summation of bandwidth overhead for connection segments and Links must be less than the maximum specified in Table 10 of Section 9.3.</td></tr>
<tr><td>0010</td><td>2</td><td>VPC</td></tr>
<tr><td>0011</td><td>3</td><td>Physical Link</td></tr>
<tr><td>0100</td><td>4</td><td>Tunnel</td></tr>
<tr><td>0101</td><td>5</td><td>Reserved</td><td rowspan="3"></td></tr>
<tr><td>0110</td><td>6</td><td>Reserved</td></tr>
<tr><td>0111</td><td>7</td><td>Reserved</td></tr>
<tr><td>1xxx</td><td>8-15</td><td>Vendor Defined</td><td>If used, combination of vendor defined plus rest of BSL must be less than the maximum specified in Table 10 of Section 9.3.</td></tr>
</table>

10.6 Auto-Configuration Protocol (ACP) Control Cells

The Auto-Configuration Protocol (ACP) control cells consist of both Indication and Confirmation cells. The format of the Indication cell is shown in Table 26 and the Confirmation is shown in Table 27.

Table 26. ACP Indication Control Cell Format

offset	8	7	6	5	4	3	2	1	ID
0	GFC/VPI				VPI				ATM Cell Header
1	VPI				VCI				
2	VCI								
3	VCI				PT = 000_2			0_2	
4	HEC								
5	Protocol_ ID = 00001010_2								Payload Hdr
6	Msg_Type = 0_{10}				Revision_ID				
7	Reserved								
8	Reserved								
9	ACP_Type				Reserved				ACP Record
10	SL/LL Option				Receiver N4				
11	Reserved								
12-13	Receiver $N2_{link}$								
14	BSU Bandwidth Overhead Divisor M								
15	Reserved				Receiver $N5_{recv}$ Delay [27:0]				
16	Receiver $N5_{recv}$ Delay [27:0]								
17	Receiver $N5_{recv}$ Delay [27:0]								
18	Receiver $N5_{recv}$ Delay [27:0]								
19-22	Reserved								
23-50	Reserved								
51	Reserved						CRC-10		Trailer
52	CRC-10								

Table 27. ACP Confirmation Control Cell Format

<table>
<tr><th>offset</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>ID</th></tr>
<tr><td>0</td><td colspan="4">GFC/VPI</td><td colspan="4">VPI</td><td rowspan="5">ATM Cell Header</td></tr>
<tr><td>1</td><td colspan="4">VPI</td><td colspan="4"></td></tr>
<tr><td>2</td><td colspan="8">VCI</td></tr>
<tr><td>3</td><td colspan="4"></td><td colspan="3">PT = 000_2</td><td>0_2</td></tr>
<tr><td>4</td><td colspan="8">HEC</td></tr>
<tr><td>5</td><td colspan="8">Protocol_ ID = 00001010_2</td><td rowspan="4">Payload Hdr</td></tr>
<tr><td>6</td><td colspan="4">Msg_Type = 1_{10}</td><td colspan="4">Revision_ID</td></tr>
<tr><td>7</td><td colspan="8">Reserved</td></tr>
<tr><td>8</td><td colspan="8">Reserved</td></tr>
<tr><td>9</td><td colspan="4">ACP_Type</td><td colspan="4">Reserved</td><td rowspan="14">ACP Record</td></tr>
<tr><td>10</td><td colspan="4">SL/LL Option</td><td colspan="4">Auto-Configured N4</td></tr>
<tr><td>11</td><td colspan="8">Auto-Configured $N2_{vc}$</td></tr>
<tr><td>12-13</td><td colspan="8">Echoed Receiver $N2_{link}$</td></tr>
<tr><td>14</td><td colspan="8">Echoed Receiver M</td></tr>
<tr><td>15</td><td colspan="4">Reserved</td><td colspan="4"></td></tr>
<tr><td>16</td><td colspan="8" rowspan="3">Echoed Receiver $N5_{recv}$ Delay [27:0]</td></tr>
<tr><td>17</td></tr>
<tr><td>18</td></tr>
<tr><td>19</td><td colspan="4">Reserved</td><td colspan="4"></td></tr>
<tr><td>20</td><td colspan="8" rowspan="3">Transmitter $N5_{trans}$ Delay [27:0]</td></tr>
<tr><td>21</td></tr>
<tr><td>22</td></tr>
<tr><td>23-50</td><td colspan="8">Reserved</td></tr>
<tr><td>51</td><td colspan="6">Reserved</td><td colspan="2"></td><td rowspan="2">Trailer</td></tr>
<tr><td>52</td><td colspan="8">CRC-10</td></tr>
</table>

10.6.1 ACP ATM Cell Header

The ACP Indication and Confirmation cells are sent on the LL-RCC channel for the logical link (Section 8). The VPI/VCI of the ATM cell header is determined by Table 28 depending on whether the QFC link is a physical link or tunnel.

Table 28. Determining VPI/VCI for an ACP Control Cell

Link Type	ACP Channel (**LL-RCC**)	
	VPI	VCI
Physical Link	0	RCC_VCI
Tunnel	Tunnel VPI	

10.6.2 ACP Record

Table 29 defines the values for the ACP_Type field which may relate to a physical link pair or tunnel.

Table 29. Auto-Configuration Protocol (ACP) Record ACP_Type Field Values

ACP_Type		Relates to:	Notes
Binary	Decimal		
0000	0	Link	
0001	1	Tunnel	
0010-0111	2-7	Reserved	
1xxx	8-15	Vendor Defined	For auto-configuration extensions.

The valid ranges for the Revision_ID are listed in Table 14. The SL/LL Option values are listed in Table 30 and the allowable N4 values in Table 31. The procedures related to choosing these three values are explained in Section 9.1.1.

Table 30. Values for Short Link (SL) / Long Link (LL) Option

SL/LL Option		Option	Notes
Binary	Decimal		
0000	0	Short Link	FCVC counters are limited to 7 bits and the $Limit_{vcdef}$ is default limit for all connections. FCVL counters and limits are 27 bits.
0001	1	Long Link	Both FCVC and FCVL counters and limits are 27 bits wide.
0010-1111	2-15	Reserved	

Table 31. Valid BSU Packing (N4) Values.

Link Type	N4		Notes
	Binary	Decimal	
	0	0	Illegal
Tunnel	0001-0110	1-6	Transmitter must be able to unpack up to 1-6 BSU records from a BSU cell to be compliant.
Physical Link or Tunnel	0111	7	Maximum N4, transmitter must be able to unpack 7 BSU records from one BSU cell.
	1xxx	8-15	Illegal

The Receiver $N2_{vc}$ field is an unsigned 8-bit variable that establishes the N2 value for all connections established for the receiver. This field is assigned by the transmitter based on the auto-configured value of N4 and the receiver provided value of M and $N2_{link}$. The procedures are described in Section 9.1.1.

The Receiver $N2_{link}$ field is an unsigned 16-bit variable that establishes the link N2 value for the receiver. The Confirmation echoed value for $N2_{link}$ shall equal the value in the Indication cell. That is, the transmitter shall not modify the field.

The Receiver BSU Overhead Bandwidth Divisor M field is an unsigned 8-bit variable that establishes the target amount of overhead to be used on the reverse direction RCC(s) of the logical link for BSUs cells. The Confirmation echoed value for M shall equal the value in the Indication cell. The transmitter shall not modify M. The relationship of M and BSU overhead bandwidth is explained in Sections 9.1.1 and 9.3.

The Receiver $N5_{recv}$ delay field is an unsigned 28-bit timer variable that establishes the nominal BSU processing delay at the receiver. $N5_{recv}$ delay field timer is a counter with a time resolution of 250 nano-seconds (ns). The Confirmation echoed value for Receiver $N5_{recv}$ delay shall equal the value in the Indication cell. The transmitter shall not modify $N5_{recv}$.

The Transmitter $N5_{trans}$ delay field is an unsigned 28-bit timer variable that establishes the nominal BSU processing delay at the transmitter. $N5_{trans}$ delay field timer is a counter with a time resolution of 250 nano-seconds (ns).

10.7 Buffer State RTT (BSR) Measurement Control Cells

The Buffer State RTT (BSR) measurement control cells consist of an Indication and Confirmation measurement cell. The format of the BSR Indication cell is shown in Table 32 and the Confirmation cell in Table 33.

Table 32. The BSR Indication Control Cells

<table>
<tr><th>offset</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>ID</th></tr>
<tr><td>0</td><td colspan="4">GFC/VPI</td><td colspan="4">VPI</td><td rowspan="5">ATM Cell Header</td></tr>
<tr><td>1</td><td colspan="4">VPI</td><td colspan="4" rowspan="2">VCI</td></tr>
<tr><td>2</td><td colspan="4" rowspan="2"></td></tr>
<tr><td>3</td><td colspan="3">PT = 110_2</td><td>0_2</td></tr>
<tr><td>4</td><td colspan="8">HEC</td></tr>
<tr><td>5</td><td colspan="8">Protocol_ ID = 00001010_2</td><td rowspan="4">Payload Hdr</td></tr>
<tr><td>6</td><td colspan="4">Msg_Type = 6_{10}</td><td colspan="4">Revision_ID</td></tr>
<tr><td>7</td><td colspan="8">Reserved</td></tr>
<tr><td>8</td><td colspan="8">Reserved</td></tr>
<tr><td>9</td><td colspan="4">Reserved</td><td colspan="4" rowspan="4">Tx_starttime [27:0]</td><td rowspan="13">BSR Records</td></tr>
<tr><td>10</td><td colspan="4" rowspan="3"></td></tr>
<tr><td>11</td></tr>
<tr><td>12</td></tr>
<tr><td>13</td><td colspan="4">Reserved</td><td colspan="4" rowspan="4">Rx_receivetime [27:0] = 0</td></tr>
<tr><td>14</td><td colspan="4" rowspan="3"></td></tr>
<tr><td>15</td></tr>
<tr><td>16</td></tr>
<tr><td>17</td><td colspan="4">Reserved</td><td colspan="4" rowspan="4">Rx_forwardtime [27:0] = 0</td></tr>
<tr><td>18</td><td colspan="4" rowspan="3"></td></tr>
<tr><td>19</td></tr>
<tr><td>20</td></tr>
<tr><td>21-50</td><td colspan="8">Reserved</td></tr>
<tr><td>51</td><td colspan="6">Reserved</td><td colspan="2" rowspan="2">CRC-10</td><td rowspan="2">Trailer</td></tr>
<tr><td>52</td><td colspan="6"></td></tr>
</table>

Table 33. BSR Confirmation Control Cell

offset	8	7	6	5	4	3	2	1	ID
0	GFC/VPI				VPI				ATM Cell Header
1	VPI				VCI				
2	VCI								
3	VCI				PT = 110_2			0_2	
4	HEC								
5	Protocol_ ID = 00001010_2								Payload Hdr
6	Msg_Type = 7_{10}				Revision_ID				
7	Reserved								
8	Reserved								
9	Reserved				Tx_starttime[27:0] ← Indication(Tx_starttime)				BSR Records
10-12	Tx_starttime[27:0] ← Indication(Tx_starttime)								
13	Reserved				Rx_receivetime [27:0]				
14-16	Rx_receivetime [27:0]								
17	Reserved				Rx_forwardtime [27:0]				
18-20	Rx_forwardtime [27:0]								
21-24	Reserved for Transmitter (Tx_stoptime)								
25-50	Reserved								
51	Reserved						CRC-10		Trailer
52	CRC-10								

10.7.1 BSR ATM Cell Header

The BSR Indication and Confirmation cells are sent on the LL-RCC channel for the logical link (Section 8). The VPI/VCI of the ATM cell header is determined by Table 34 depending on whether the QFC link is a physical link or tunnel.

Table 34. Determining the VPI/VCI for a BSR Control Cell

Link Type	BSR Channel	
	VPI	VCI
Physical Link	0	RCC_VCI
Tunnel	Tunnel VPI	

10.7.2 BSR Records

The three time-stamp fields of the BSR record are unsigned 28-bit counters with a time resolution of 250 nanoseconds[1]. The transmitter shall set Rx_receivetime and Rx_forwardtime to zero prior to sending the Indication to the receiver. The procedures for assigning and using these time-stamp fields are defined in Section 9.1.2

10.7.3 BSR Control Cell CRC-10

A CRC-10 shall be generated by the transmitter after it puts the time-stamp in Tx_starttime. The CRC-10 shall be checked by the receiver when it receives the BSR Indication prior to time-stamping the Rx_receivetime. The receiver shall generate a new CRC-10 after it injects the Rx_forwardtime into the BSR Confirmation cell prior to sending it to the transmitter. The transmitter shall check the BSR Confirmation cell's CRC-10 when it receives the cell.

10.8 Signaling Coding Requirements

10.8.1 Modifications to SETUP Message

Table 35 shows the extensions for the SETUP message required for the optional connection Limit_{vcdef} override for QFC connection establishment. Table 35 is an extension to the format shown in Figure 5-7 of the ATM Forum's UNI 3.1 signaling specification.

Table 35. SETUP Message Extension for Optional Limit_{vcdef} Override

Message Type: SETUP

Significance: local

Direction: reverse

Information Element	Reference	Direction	Type	Length
QFC Limit_{vcdef} Override	10.8.4	reverse	O[1]	7

[1] Only allowed when creating a QFC connection (as requested in the Broadband bearer capability information element, Section 10.8.3). When included, overrides the default Limit_{vcdef} for the called party to calling party forward direction of the QFC connection.

[1] The transmitter may use a time-stamp resolution other than 250 ns, if desired.

10.8.2 Modifications to CONNECT Message

Table 36 shows the extensions for the CONNECT message required for the optional connection Limit_{vcdef} override for QFC connection establishment. Table 36 is an extension to the format shown in Figure 5-3 of the ATM Forum's UNI 3.1 signaling specification.

Table 36. CONNECT Message Extension for Optional Limit_{vcdef} Override

Message Type: CONNECT

Significance: local

Direction: reverse

Information Element	Reference	Direction	Type	Length
QFC Limit_{vcdef} Override	10.8.4	reverse	O[1]	7

[1] Only allowed when creating an QFC connection (as requested in the Broadband bearer capability information element, Section 10.8.3). When included, overrides the default global Limit_{vcdef} for the calling party to called party forward direction of the QFC connection.

10.8.3 Modification to Existing Broadband Bearer Capability Information Element

In order to support QFC signaling, the Broadband Bearer Capability information element must be extended to allow signaling of the creation of an QFC connection. This is accomplished by adding an additional code point to the Traffic Type parameter of the Broadband Bearer capability:

Traffic Type (octet 5a)

Bits 5 4 3	Meaning
0 0 0	No indication
0 0 1	Constant bit rate
0 1 0	Variable bit rate
1 1 1	QFC[1]

[1] When the QFC indication is given, the Timing Requirements should be set to "End-to-end timing not required" (binary 10). Implementors should allow the specific Traffic Type value for QFC to be programmable until ATM Forum and ITU-T standardization is complete.

10.8.4 Connection Limit_{vcdef} Override Information Element (Limit-IE)

Table 37 contains the information element coding needed for the new QFC connection Limit_{vcdef} information element. Implementors should allow the specific connection Limit-IE code set value for QFC to be programmable until ATM Forum and ITU-T standardization is complete.

Table 37. Connection Limit$_{vcdef}$ Override IE Identifier Coding (Code set 0)

Bits 8 7 6 5 4 3 2 1	Information Element	Section Reference	Max Length	Max no. of occurrences
1 1 1 1 0 0 0 0	QFC Connection Limit$_{vcdef}$ Override	10.8.4	6	1

The QFC Limit information element is an 8-byte element that may be used with the SETUP and CONNECT message. Table 38 shows the format of the information element.

Table 38. Connection Limit$_{vcdef}$ Override Information Element Format

<table>
<tr><td colspan="8" align="center">Bits</td><td></td></tr>
<tr><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>Octets</td></tr>
<tr><td colspan="8" align="center">QFC limit
1 1 1 1 0 0 0 0
Information element identifier</td><td>1</td></tr>
<tr><td>1
ext</td><td colspan="2">Coding Standard</td><td colspan="5">IE Instruction Field</td><td>2</td></tr>
<tr><td colspan="8" align="center">Length of QFC limit contents</td><td>3</td></tr>
<tr><td colspan="8" align="center">Length of QFC limit contents (continued)</td><td>4</td></tr>
<tr><td colspan="5">Reserved</td><td colspan="3">Limit$_{vcdef}$ Override</td><td>5</td></tr>
<tr><td colspan="8" align="center">Limit$_{vcdef}$ Override (continued)</td><td>6</td></tr>
<tr><td colspan="8" align="center">Limit$_{vcdef}$ Override (continued)</td><td>7</td></tr>
<tr><td colspan="8" align="center">Limit$_{vcdef}$ Override (continued)</td><td>8</td></tr>
</table>

Coding Standard (octet 2)

Bits 7 6	Meaning
0 0	ITU-T standardized

IE Instruction Field (octet 2)

Bits 5 4 3 2 1	Meaning
1 0 0 0 0	Clear call

$Limit_{vcdef}$ (octets 5 through 8)

The $Limit_{vcdef}$ is used to specify an connection $Limit_{vcdef}$ override for an QFC connection. It is coded as an unsigned binary integer, with bit-3 of the first octet being the most significant bit and bit-1 of the forth octet being the least significant. Depending on the auto-configuration option chosen — Short Link or Long Link — the $Limit_{vcdef}$ field will be 7 bits or 27 bits wide, respectively. When the $Limit_{vcdef}$ is an 7 bit field, the 7 bits will reside in the low ordered 6:0 region of the $Limit_{vcdef}$ field.

Appendix A: Glossary

ABR	Available Bit Rate.
ACP	Auto-Configuration Protocol.
BER	Bit Error Rate.
BSC	Buffer State Check Messages (control cells).
BSL	Buffer State Limits Messages (control cells).
BSR	Buffer State RTT Messages (control cells).
BSU	Buffer State Update Messages (control cells).
CBR	Constant Bit Rate.
Control Update Pipeline Delay (N5)	One of the four delay components of the one-hop, closed loop control system regulated by the QFC protocol. Also referred to as N5. A summation of all of the receiver and transmitter control system control cell processing delays.
FCVC	Flow-Controlled Virtual Connection. Term used to refer to the parts of QFC protocol related to the management of the connection resources.
FCVL	Flow-Controlled Virtual Link. Term used to refer to the parts of QFC protocol related to the management of the link resources.
Goodput	Throughput that is usable by the end-user's applications.
Hop	A connection between a transmitter and a receiver on a logical link.
LCR	Link Cell Rate of a Physical Link.
LLCR	Logical Link Cell Rate. Equals the LCR of a physical link or the PCR of a Tunnel VPC.
LLPD	Link Propagation Delay of a logical link.
LPD	Link Propagation Delay of a physical link.
Logical Link	A physical link or a tunnel through a VPC.
LL	Long Link Option. An option of the QFC protocol that supports 28 bit counters to support any size link. See SL.

M BSU Bandwidth Overhead Divisor.

N2 See Quantum count.

N3 See RTT.

N4 See Quantum packing factor.

N5 See control update pipeline delay.

QFC Quantum Flow Control. A protocol mechanism to offer an Available Bit Rate (ABR) service that can support instantaneous access to unused network bandwidth with cell loss no greater than link BER.

Quantum Count (N2) One of the four delay components of the one-hop, closed loop control system regulated by the QFC protocol. Also referred to as N2. Quantum count allows the receiver to delay sending a forwarded cell update to transmitter until N2 data cells associated with a connection segment have been forwarded.

Quantum Packing Factor (N4) One of the four delay components of the one-hop, closed loop control system regulated by the QFC protocol. Also referred to as N4. Quantum packing factor allows the receiver to pack up to N4 quantum count updates into an update message.

RCC Reserved Control Channel.

RM Resource Management

RTT Round Trip Time, one of the four delay components of the one-hop, closed loop control system regulated by QFC protocol. It is also referred to as N3. This is the sum of the forward and reverse Link Propagation Delay between the transmitter and the receiver.

SL Short Link Option. An option of the QFC protocol that allows 8 bit counters to reduce cost. See LL.

TDM Time-Division Multiplexing.

Throughput Data transported through the network. See Goodput.

Tunnel A VPC between the transmitter and receiver that passes through several nodes, but is treated as if it were a logical QFC link.

VBR Variable Bit Rate

VCC Virtual Channel Connection

VPC Virtual Path Connection.

Appendix B: Technical Tutorial on N2, N3, N4 and N5

This appendix presents the background for the parameters N2, N3, N4 and N5 used in the description of the QFC protocol. This appendix should be used to gain a wider understanding of the QFC control loop and the dynamic ranges of these four parameters.

B.1 The Quantum Flow Control Closed-Loop Control System

The QFC control system is a distributed, unsynchronized, potentially lossy, closed-loop control system. Furthermore, the dynamic range of this control system can vary widely for different link distances and different workload ranges.

For basic operations, data is sent in the forward direction from a transmitter to a receiver, and control information is sent back from the receiver to inform the transmitter about the downstream receiver state. In QFC, as in any closed-loop control system, care must be taken to ensure that control feedback information flows back and forth in such a way that stability and performance can be assured.

QFC control system feedback delay is defined as the time a data cell is sent by a transmitter (puts the first bit of a transmitted cell on the "wire") to the time an acknowledgment from the receiver that the cell was forwarded downstream is received and processed by the transmitter. This one-hop, closed-loop control system has four delay components: quantum-count delay, transmission delay (Round Trip Time), quantum update packing delay and control update pipeline delay. We also refer to these delay components as N2, N3, N4 and N5, respectively.

B.1.1 Quantum-Count Delay (N2)

The quantum-count delay (N2) is an auto-configured delay factor used to reduce the reverse direction QFC BSU bandwidth overhead to 1/N2. Quantum-count allows the receiver to delay sending a forwarded cell update to the transmitter until N2 data cells associated with a connection have been forwarded from the receiver's buffers to a downstream destination (or destinations, if multipoint). The receiver may send BSUs to the transmitter more often than N2 as long as the receiver does not exceed the maximum reverse direction control bandwidth overhead limit of QFC.

B.1.2 Round Trip Time (RTT) Delay (N3)

RTT (N3) is defined as the sum of the forward and reverse Link Propagation Delay (LPD) between the transmitting and receiving node. During RTT, the transmitter may have sent as many as RTT_{usec} x $LLCR_{cells/usec}$ cells. This number is defined as N3. RTT is determined at link bootstrap via RTT measurement mechanism defined in Sections [9.1.2] and [10.7]. During the period of cell operations on a physical link, the RTT is generally invariant. (In most cases, LPD will be symmetric, but the QFC control loop is not dependant on link symmetry.) In a short-haul link, N3 may be as small as a fraction of a cell. In this case, N2, N4 and N5 delay dominate the QFC cell equations. When QFC operates over a long-haul link, N3 may dominate the buffer requirements of N2, N4 and N5.

B.1.3 Quantum Update Packing Delay (N4)

The quantum update packing delay (N4) is caused by packing more than one update record into a BSU message. Like N2, this packing reduces the reverse direction QFC BSU bandwidth overhead that flows from the receiver to the transmitter to 1/(N2 x N4). In order to pack the cell to its capacity, however, the receiver may have to hold up to N4 updates before sending the BSU cell. Since an update may not be constructed except for every N2 cell times, the worst case BSU delay for an individual connection segment may be N2 x N4 if a connection segment is operating at LLCR.

B.1.4 Control Update Pipeline Delay (N5)

The control update pipeline delay (N5) is a summation of all of the receiver and transmitter control system pipeline processing delays. N5 will differ for every QFC device implementation depending on the efficacy of the implementation of BSU processing. N5 is a pipeline delay metric and, as such, is measured in cell times of forward direction LLCR.

B.1.5 Control System Delay — Meeting Instantaneous Bandwidth Requirement.

Figure B.1 contains a diagram of the best case QFC control system feedback delay for an individual connection segment transmitting at a sustained rate at LLCR. The cumulative delay expressed by Figure B.1 determines the delay buffering required for a connection segment in order to meet the 100 percent link occupancy goal desired for the ABR service definition. That is, at any given time, the ability for one connection segment to consume up to full LLCR of available bandwidth.

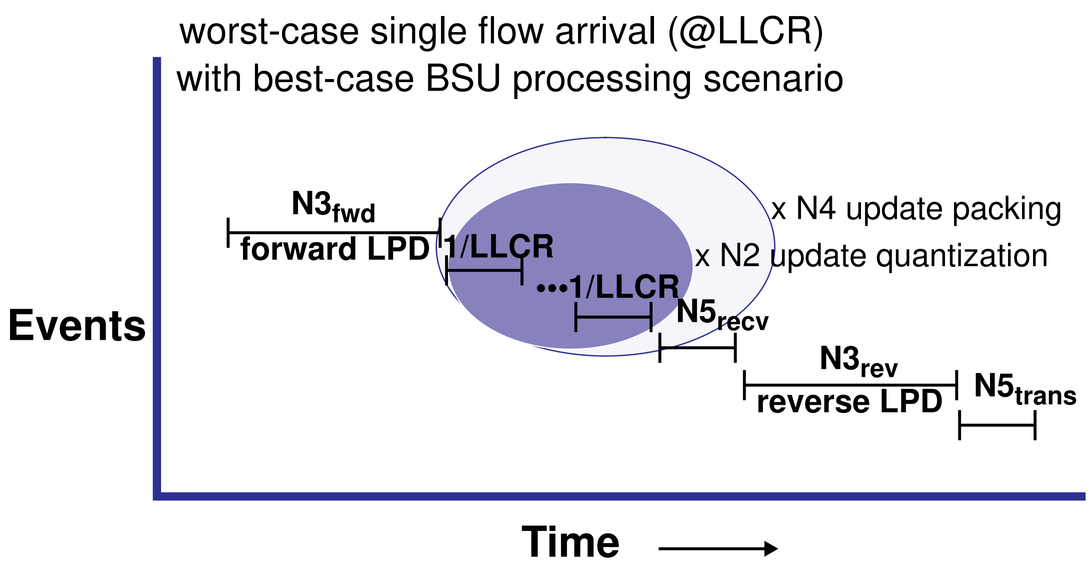


Figure B.1. Best-Case FCVC Control System Feedback Delay

The four delay components — N2, N3, N4 and N5 — are broken down in the graph in Figure B.1 in relative juxtaposition to their appearance in time within the control feedback pipeline. The graph shows the forward component of N3 as the first delay event (equal to forward LPD). The next event is the receiver having to wait for N2 data cells before an update may be composed and sent back to the transmitter. Since cells cannot be forwarded until they arrive, the minimum N2 delay is determined by the LLCR and is equal to N2 x (1/LLCR), where 1/LLCR is the “cell time” of the forward link.

N4 defines the BSU packing density. Therefore, it is necessary — in this scenario of a single transmitter running at LLCR — to pack N4 of the N2 events in the BSU before sending the BSU to the transmitter. Until this happens, the transmitter cannot be informed that the receiver has forwarded any cells. This means that the total delay incurred for the N2 and N4 parameters as described in Equation [B.1] is factored by N2 x N4.

The first N5 component, called $N5_{recv}$ in Figure B.1, is the receiver BSU processing pipeline delay incurred by the receiver when it forwards the cell to the transmitter. $N5_{recv}$ is measured as pipeline delay; it may be converted to cells in the same way that N3 (RTT) is: that is $N5_{usec}$ x $LLCR_{cells/usec}$. After the cell has been transmitted by the receiver, it will take reverse direction LPD to arrive at the transmitter. This delay event is the other half of the N3 delay. Finally, when the cell arrives at the transmitter, the transmitter will also have some non-zero processing pipeline delay. This N5 delay component, called $N5_{trans}$ in Figure B.1, is the other half of N5. At the end of this example feedback loop the transmitter would get refreshed with N2 x N4 buffers.

$$\text{Max N4/N2 Delay per BSU cell}_{\text{usec}} = (\text{N2}_{\text{cells/update}} \times \text{N4}_{\text{update/BSU}}) \times \left(\frac{1}{\text{LLCR\_FWD}_{\text{cells/usec}}}\right) \times 1_{\text{BSU}}$$

Equation [B.1]: Maximum Connection Delay for BSU Cell Packing (N4)

The previous analysis of the control feedback delay assumes the best case drain rate (LLCR) so as to place the most burden on the control system delay problem. That is, to eliminate the variability of the receiver drain rate from the control loop delay calculations. In reality, the receiver connection drain rate will vary from as high as LLCR down to a blocked state. This is normal QFC (and ATM) behavior.

If a receiver gives a transmitter sufficient buffers to keep the transmit pipeline open ($Limit_{LLCRmin}$), computed by Equation [B.2] below, then the transmitter may sustain data forwarding while waiting for the receiver to refresh the transmitter’s buffer counts. If all connection segments terminated at a receiver have Equation [B.2] buffers assigned to them, then the dynamic response of the QFC protocol may be provisioned to be arbitrarily precise (see the description of Model A in Appendix E). Provisioning Equation [B.2] buffers allows any connection segment to transmit whenever data is available on the connection segment and available bandwidth is free on the link.

$$\text{Receiver Buffers for Delay}_{\text{cells}} = (\text{N3}_{\text{usec}} + \text{N5}_{\text{usec}}) \times \text{LLCR-FWD}_{\text{cells/usec}}$$

$$\text{Receiver Buffers for Delay per BSU}_{\text{cells}} = (\text{N2}_{\text{cells/update}} \times \text{N4}_{\text{updates/BSU}}) \times 1_{\text{BSU}}$$

$$\text{Limit}_{\text{LLCRmin}_{\text{cells}}} = \text{Receiver Buffers for Delay}_{\text{cells}} + \text{Receiver Buffers for Delay per BSU}_{\text{cells}}$$

Equation [B.2]: FCVC Receiver Buffers Required to Maintain LLCR Transfer

Reducing the number of buffers computed in Equation [B.2] that are assigned to a connection will affect a connection's ability to sustain LLCR but will not create incorrect FCVC or FCVL protocol behavior.

Given Equation [B.2], which is the closed-loop control system delay measured in cells, the maximum delay loop for a connection segment flow can be calculated by converting cells into cell-times at forward LLCR. The formula for this conversion is contained in Equation [B.3].

$$\text{Maximum Control Response Delay}_{\text{usec}} = \frac{\text{Limit}_{\text{LLCRmin}_{\text{cells}}}}{\text{LLCR-FWD}_{\text{cells/usec}}}$$

Equation [B.3]: Worst Case Control Response Delay for an FCVC Flow

Appendix C: Protocol Robustness

This appendix delineates the recommended QFC protocol response to protocol compliance errors.

C.1 Error Handling for Non-compliant Quantum Flow Control Behavior

There are three phases of QFC protocol behavior where compliance errors can occur: initialization, normal operations and teardown. The initialization phase relates to physical or logical link initialization as well as connection initialization. Normal operation relates to connection segment data forwarding operations as well as protocol resource reallocations that may occur occasionally (such as a BSL Limit_{link} change). Finally, connection teardown procedural errors can occur.

C.1.1 Initialization Compliance Errors

Table C.1 describes, for possible initialization errors, the appropriate response behavior for the conforming QFC entity.

Table C.1. Incorrect Initialization Behavior with Appropriate Response

Behavior	Response
A receiver or transmitter replies with a Long Link (LL) option when it should have chosen the Short Link (SL) option according to the procedures established in Section 9.1.1.	The conforming protocol entity may continue to send ACP Requests to the non-conforming entity until a correct response is received according to Section 9.1.1. The conforming protocol entity will not allow QFC operations unless this behavior is corrected. An error will be recorded against the non-conforming entity.
A receiver or transmitter replies with an incorrect N4 BSU packing factor according to the procedures established in Section 9.1.1.	
ACP Confirmation Cell arrives with one or more of the following variables equal to 0: Receiver $N2_{vc}$, Receiver $N2_{link}$, Receiver M, $N5_{rcv}$ Delay, Transmitter $N5_{trans}$ Delay	
ACP Confirmation Cell arrives with parameters that allow BSU bandwidth > 6%	

C.1.2 Incorrect Operational Behavior

Table C.2 describes, for possible operational errors, the appropriate response behavior for the conforming QFC entity.

Table C.2. Incorrect Data Forwarding Behavior with Appropriate Response

Behavior	Response
A receiver or transmitter receives a QFC RM cell with a Revision_ID that is not equal to the Auto-Configured Revision_ID.	The conforming protocol entity will send an ACP Request to the non-conforming entity and restart the QFC Service. The conforming entity will record an non-conformance error on the protocol entity that sent the errored control cell.
The transmitter forwards a cell for which the receiver does not have a connection buffer available (i.e. exceeds Limit[i]).	The receiver will discard the cell and record the error. Data forwarding may continue to operate normally, the Rx_Counter[i,link] and Fwd_Counter[i,link] shall be updated as if a cell had been received and forwarded and the receiver shall generate a BSU cell for the connection or Logical Link as appropriate and send it to the transmitter. The receiver may optionally reset the QFC logical link with an ACP message.
The transmitter forwards a cell for which a receiver does not have a shared link buffer available (exceeds Limit$_{link}$).	
The transmitter *apparently* does not forward Buffer State Check (BSC) cells.	This could be caused by link BER but if sustained long enough may be declared as a non-conformance error on the transmitter. The receiver may continue or terminate QFC operations.
The receiver *apparently* does not send Fwd_Count[i] or Fwd_Count[link] in BSU cells.	This could be caused by link BER but if sustained long enough may be declared as a con-conformance error on the receiver. The transmitter should abort QFC operations.
The Buffer State Confirmation cell contains a BSL_Limit$_{vcdef}$ or BSL_Limit$_{link}$ that is less then the Limit placed in the Buffer State Limit Indication cell or is higher then the previous instantiated BSL_Limit$_{vcdef}$ or BSL_Limit$_{link}$.	The receiver should record a compliance error on the transmitter and may chose to resend the BSL to establish the proper limit (cells may be lost during this time) or restart QFC on the link.

Appendix D: Protocol Overhead Bandwidth Management

This appendix presents the background information concerning QFC control cell bandwidth management. It provides useful equations for determining how often BSC checks should be sent while maintaining compliance with the control bandwidth overhead limitations.

D.1 Forward Direction BSC Control Bandwidth Management

BSC records should be sent at 10 times the rate at which the logical link is expected to lose or corrupt ABR data cells. For the purpose of BSC correction to QFC counters, a corrupted ABR data cell is one which has a cell header corruption — causing HEC discard. (Cell payload corruption of ABR data cells will not affect QFC counters.) For example, if a link's Bit Error Rate (BER) is measured as 1 chance in 10^{-10} then we should expect a lost ABR data cell with the probability of 1 chance in (40 cell header bits) x 10^{-10} = 4 x 10^{-9}. BSC records may be sent more frequently the 10 times the error rate so long as their total contribution to overhead is less then that prescribed by Table 10 in Section 9.3. Equation [D.1] defines the maximum BSC overhead bandwidth guidelines.

$$\text{Err. Data Cells}_{\text{err cells/sec}} = (\text{BER-FWD}_{\text{errors/bit}} \times (40)_{\text{bits/hdr}}) \times 1_{\text{hdr/cell}} \times \text{LLCR-FWD}_{\text{cells/sec}}$$

$$\text{Err. QFC Cntrl Cells}_{\text{err cells/sec}} = (\text{BER-FWD}_{\text{errors/bit}} \times (424)_{\text{bits/cell}}) \times \left(\frac{7.5_{\text{cntrl cells}}}{100_{\text{cells}}} \times \text{LLCR-FWD}_{\text{cells/sec}}\right)$$

$$\text{BSC Overhead}_{\text{BSCs/sec}} = C \times 10_{\text{BSCs/errored cell}} \times (\text{Err. Data Cells}_{\text{err cells/sec}} + \text{Err. QFC Cntrl Cells}_{\text{err cells/sec}})$$

$$\text{BSC Overhead}_{\text{BSCs/sec}} \le \left(\frac{1_{\text{BSC}}}{100_{\text{cells}}} \times \text{LLCR-FWD}_{\text{cells/sec}}\right)$$

Equation [D.1]: Buffer State Check (BSC) Maximum Overhead Constraint

Case Study: SONET OC-3 with LLCR of 353,207 and BER of 10^{-10} could have as many as .015 errored data cells per second and .001 errored QFC control cells per second. If the link were supporting 3000 established connection C, the BSC Overhead would be approximately 3000 x 10 x .016 = 480 BSCs/second. This is well below 1% of the LLCR equal to 3532 cells/second. In this example, the BSCs could be generated once every second for each connection C with 500 link BSCs and still remain under target overhead bandwidth.

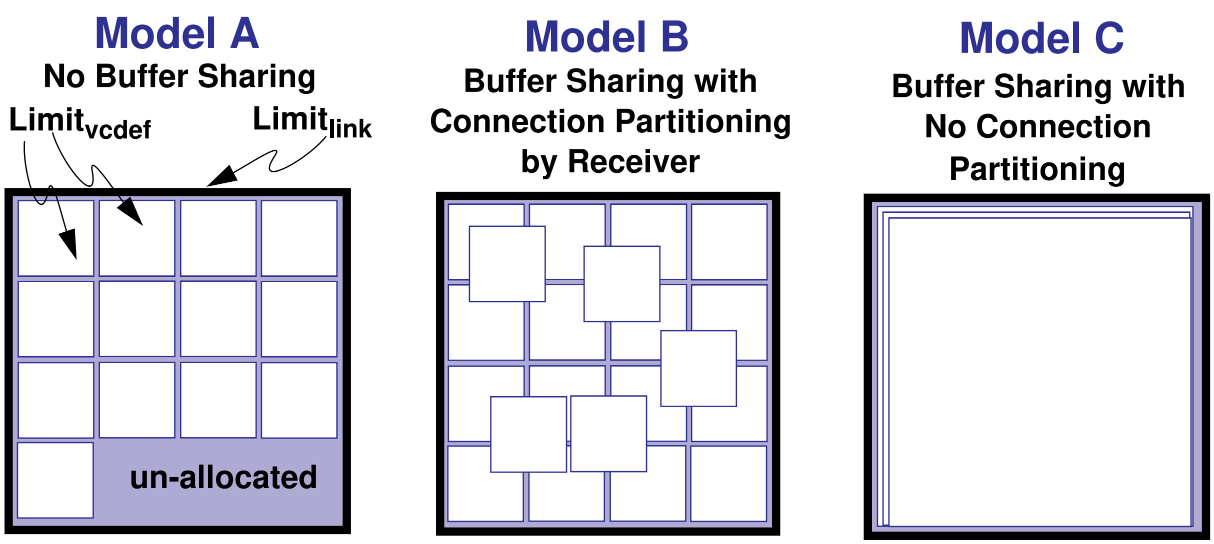
Appendix E: Buffer Sharing

The operation and benefit of Quantum Flow Control (QFC) is intuitively easy to understand in the context of a few connections over a short logical link. This appendix explores upward scaling in terms of the number of supported connections and the length of links. The bottom line issue, of course, is the amount of buffer memory required since, in a large scale system, memory can become a very significant cost component.

The QFC protocol inherently supports the sharing of buffers among the connections passing through a logical link, enabling cost effective solutions. In order to choose parameters appropriate for a particular target range of maximum connections, average number of active connections, performance, and cost, it is necessary to understand some fundamental properties of QFC buffer sharing. This appendix will explore performance in relation to available memory.

E.1 Managing QFC Memory

There are three general buffer management configurations supported by QFC. These are pictorially represented in Figure E.1 as Models A, B and C. Differentiation between the three models depends on the magnitude of, and relationship between, two receiver buffering parameters, Limit_{vcdef}[1] and Limit_{link}. Each model is robust in being able to guarantee zero cell loss. They vary in their cost versus performance characteristics.



$C \times \text{Limit}_{vcdef} \leq \text{Limit}_{link}$ | $C \times \text{Limit}_{vcdef} > \text{Limit}_{link}$, $\text{Limit}_{vcdef} \ll \text{Limit}_{link}$ | $\text{Limit}_{vcdef} = \text{Limit}_{link}$

Figure E.1. Receiver Limit_{link} Management Policies

In Model A, there is no buffer sharing. The buffer allocation per connection is such that the aggregate allocation of all connections is less than or equal to the total allocation for the link.

[1] We will not consider the Limit[i] override capability available through signaling for this discussion.

This is the simplest but least efficient memory model. It will support instantaneous and sustained throughput at LLCR if $\text{Limit}_{\text{vcdef}}$ is sufficiently large for all connections [i].

Model B is the typical buffer sharing configuration. In this model, there is a level of buffer over-booking. This means that the sum of all of the per-connection buffers is greater than the number of buffers allocated to the logical link. This model expects the number of simultaneously active connections to usually be less than the maximum supported connections. During times that the number of simultaneous connections is greater than anticipated, some connections may get less than their usual share of LLCR. There is a calculable minimum buffer configuration which will protect this model from serious problems under any conditions.

As shown, Model C is not a practical configuration. It allows any individual connection to use up all the buffers for the entire logical link. A number of activity scenarios can be conceived that would result in unfairness or blocking. A receiver in this configuration expects the transmitter to take full responsibility of managing the connections within the logical link. While the operation of such transmitter control is not addressed in this appendix, the ability of the QFC protocol to shift buffer management responsibility from the receiver to the transmitter simply by setting two parameters is an elegant mechanism for product differentiation.

In all models, active connections with data to send and for which downstream buffers are available share the ABR available LLCR bandwidth fairly.

E.2 Analysis of Model A (Non-Sharing Logical Link Buffers)

In Model A, the number of buffers allocated to each connection is such that the sum of buffers allocated to all the connections is less than, or equal to, the total number of buffers allocated to the logical link. It is not possible for the activity or status of any connection to impact the ability of another connection to forward cells. In the most extreme example, if all connections except one had volumes of data to send but they were all blocked downstream, the one remaining connection would still be able to send instantaneously at full rate. Full rate, in this model, would be LLCR as long as the per-connection buffer allocation satisfies the LLCR minimum ($\text{Limit}_{\text{LLCRmin}}$) shown in Equation [B.2] in Appendix B and LLCR bandwidth is available to the ABR service.

If all per-connection buffers are set to $\text{Limit}_{\text{LLCRmin}}$ by the receiver, enough buffering will exist to ensure that a BSU message reporting forwarded cells will be received by the transmitter before the per-connection limit of cells has been exhausted at the transmitter. With this minimum per-logical link memory ($C \times \text{Limit}_{\text{LLCRmin}} \leq \text{Limit}_{\text{link}}$), theoretical maximum link performance can be achieved.

The minimum number of per-connection buffers, $\text{Limit}_{\text{vcmin}}$, that will ensure functional operation is shown in Equation [4] in Section 7.3.

E.2.1 Managing Sustained Bandwidth Using Connection Limits

As stated earlier, memory configurations sufficiently large to provide $\text{Limit}_{\text{LLCRmin}}$ buffers for every established connection provide instantaneous access to the LLCR rate. It is useful to understand the effect of per-connection allocations below $\text{Limit}_{\text{LLCRmin}}$ on sustained bandwidth throughput. There are two cases to consider — a short-haul link where N3 is small and is dominated by (N2 x N4) + N5 in the $\text{Limit}_{\text{LLCRmin}}$ equation (see Equation [B.2] in Appendix B) and a long-haul link where N3 dominates $\text{Limit}_{\text{LLCRmin}}$.

To understand how a connection's throughput could be limited in the short-haul link, suppose $\text{Limit}_{\text{vcdef}} < \text{Limit}_{\text{LLCRmin}}$ cell buffers were allocated to each connection. If only one connection becomes active, cells are sent by the transmitter to the receiver for that connection only. After sending the first $\text{Limit}_{\text{vcdef}}$ cells, the transmitter has to stop sending cells. After the receiver forwards the first N2 cells, the receiver will form a BSU record. If no other connection were forwarding cells at the receiver, then it could take as many as N2 * N4 cell times for the receiver to send a BSU message with forwarded buffer information. After the reverse direction logical link delay, the BSU message will reach the transmitting node and enable it to resume sending. The result is that over the time that $\text{Limit}_{\text{LLCRmin}}$ cells could have been sent, approximately $\text{Limit}_{\text{vcdef}}$ cells were sent instead.

The worst case analysis for the short-haul link above only accounts for a singular connection where the N4 BSU packing delays the forwarding of a connection's N2 quantum BSU update (the factor (N2 x N4) in Equation [B.2] in Appendix B). In the nominal case, where there are many active connections, it is possible that a connection's N2 roll-over event will coincide with an N4 BSU forwarding event. In these cases the maximum BSU delay is less than N2 x N4. Also, when many QFC connections are sharing the LLCR, throughput limitations are a normal artifact of fair share access to available bandwidth. This analysis highlights the quantum nature of QFC. Therefore, in short-haul links, limiting the per-connection $\text{Limit}_{\text{vc-def}}$ is an imprecise metric for controlling access to available QFC bandwidth and may have no measurable effect on real sustained throughput.

In long haul links, where N3 dominates the $\text{Limit}_{\text{LLCRmin}}$, the quantum effects of N2 counting and N4 packing are dampened. Also, unlike N2 and N4 quantization delay, N3 delay is invariant. Therefore, for long haul links, a connection's access to bandwidth may be limited by limiting the N3 buffers assigned to a connection. This is called the **dynamic bandwidth limit** (**DBL**). Using DBL a connections *sustained* access to QFC available bandwidth[2] may be arbitrarily constrained. **DBL provides a rate management mechanism for long-haul logical links**. Equation [E.1] shows how restricting N3 buffers can affect a DBL rate limitation for long-haul links. The resulting DBL sustained bandwidth is roughly proportional to the N3_{DBL}. Also, DBL precision is proportional to the size of N3_{RTT} is.

[2] A DBL may be established as a per-connection Limit[i] override using the signaling procedures of Section 9.2.

Sustained bandwidth when using DBL:

$$DBL_{cells/sec} \approx LLCR\text{-}FWD_{cells/sec} \times \left(\frac{N3_{DBL_{cells}}}{N3_{RTT_{cells}}}\right)$$

DBL Buffers:

$$Limit_{DBL_{cells}} \approx N3_{DBL_{cells}} + (N5_{usec} \times LLCR\text{-}FWD_{cells/usec}) + (N2_{cells/update} \times N4_{updates/BSU}) \times 1_{BSU}$$

Where:

$$Limit_{vcmin_{cells}} \le Limit_{DBL_{cells}} < Limit_{LLCRmin_{cells}}$$

$$N3_{RTT_{cells}} \gg (N5_{usec} \times LLCR\text{-}FWD_{cells/usec}) + (N2_{cells/update} \times N4_{updates/BSU}) \times 1_{BSU}$$

Equation [E.1]: DBL Throughput When $Limit_{DBL} < Limit_{LLCRmin}$

E.2.2 Summary of Model A

Summarizing the analysis of Model A, at least $Limit_{vcmin}$ buffers must be allocated per-connection for it to function. Per-connection $Limit_{vcdef}$ allocation in the range of $Limit_{vcmin}$ up to $Limit_{LLCRmin}$ results in full functionality but with limited throughput proportional to the number of per-connection allocated buffers on long haul links. Allocation of $Limit_{LLCRmin}$ for all connections results in instantaneous access to LLCR operation.

E.2.3 Model B Cost/Performance Trade-offs

The question of a cost/performance design point and CAC criteria for Model A can now be theorized. A CAC and resource management algorithm may place an approximate ceiling on the maximum number of connections expected for a QFC logical link. This ceiling, multiplied times the $Limit_{LLCRmin}$, dictates the amount of logical link buffers that should be provisioned for the link. In operation, if the number of requested calls exceeds that ceiling, there are two possible responses, calls can be rejected or the per-connection buffer allocation $Limit_{vcdef}$ can be decreased.

Decreasing $Limit_{vcdef}$ results in a gradual degradation of a logical link's QFC connection's ability to access instantaneous LLCR bandwidth. Since the per-connection limit reduction is only made when a large number of connections are present, the expected number of active connections should increase as the number of established connections increase — ultimately increase the likelihood of QFC maintaining 100% LLCR goodput.

There is one caveat to the above buffer management scheme. If a connection is accepted and has available to it $Limit_{LLCRmin}$ buffers, but becomes blocked downstream, it can consume up to its allocated number of receive buffers. Later, if more connections are accepted and the system chooses to decrease the per-connection allocation, consumed buffers are still consumed. This means that the connection now has more in-use buffers than the current per-connection allocation. Effectively, the total buffer pool is diminished, at least temporarily and this must be taken into account in the decision to accept additional calls by decreasing the per-connection allocation.

E.3 Analysis of Modal B (Shared Logical Link Buffering)

In Model B each connection is allocated less buffers than the link, but the sum of per-connection $Limit_{vcdef}$ allocations (C x $Limit_{vcdef}$) is greater than the link allocation ($Limit_{link}$). The concept is not so different from mapping a moderate amount of real memory into a large virtual address space. Only active connections[3] require real buffers. Also, it is likely that the nominal number of active connections $AC \subseteq C$ is small. Therefore, reusing buffers from inactive connections allows for a more economical QFC implementation when C or $Limit_{LLCRmin}$ are large.

This model can be analyzed along the same lines as Model A except that active connections take the place of defined connections. However, there is one significant difference. With Model A, a CAC decision was made at call setup time whether there were sufficient buffer resources to accept the call. In Model B, assumptions are made about the probability of simultaneous call activity. Model B CAC is based on this assumed probability[4].

In Model B, the system will operate at instantaneous LLCR as long as each active connection has $Limit_{LLCRmin}$ buffers available to it from the shared link pool of buffers at the time it is ready to burst. If this condition is always true, then an observer would not be able to measure a performance difference between a Model A or B managed system.

E.3.1 Peer-Blocking

In Model B, where buffers are shared, downstream connection blocking can cause the shared logical link buffer pool to shrink temporarily. Due to this phenomenon, connection peer-blocking exists for Model B. Peer-blocking is the performance trade-off cost of reducing link buffer expense. Peer-blocking is entirely avoidable by adding more memory to a QFC link at the discretion of the network manager. Model A does not suffer peer-blocking artifacts because each connection has sufficient buffers to run at LLCR independent of the state of another connection.

Formally, a connection segment is peer-blocked if its unacknowledged forward count of transmitted cells is less than its $Limit_{vcdef}$, (the connection has buffers available), but the link's unacknowledged forward count is equal to the $Limit_{link}$, (the link is blocked). The peer connections that are holding the $Limit_{link}$ buffers may be blocked as a normal result of QFC flow control downstream.

E.3.2 N2 Orphaning

N2 orphaning can occur when the receiver forwards less than N2 (greater than 0), cells after forming its last BSU record for the connection. Connection cell forwarding at a receiver may halt at any time because of downstream connection blocking or because the transmitter has

3 Active connections are those with queued cells in the transmitter or receiver.

4 The actual AC can be easily measured by a transmitter or receiver node. The shared buffer assumption then becomes an expectation that the measured AC is stable over time — a much better assumption base.

no more data to send. Because the receiver has not yet forwarded an N2 quantum of cells, it is prevented from sending a BSU message to the transmitter to update its state. Because of the quantization effect of N2, it is statistically likely that when either of the two connection forwarding halting events occur, some amount of orphaned N2 buffers exist.

N2 orphaning can affect both Model A or B. Model A or B connections can be affected by N2 orphaning if less than $\text{Limit}_{\text{LLCRmin}}$ allocation is given to a connection. Also, even if a Model B connection has $\text{Limit}_{\text{LLCRmin}}$ allocated buffers per connection, shared buffering can reduce the available buffers for a connection to less than $\text{Limit}_{\text{LLCRmin}}$ for some period of time.

In order to avoid N2 orphaning, the receiver, when implementing buffer sharing, shall return a Fwd_Count[link] every N2_{link} times to the transmitter. The Fwd_Count[link] will update the receiver for **any** forwarded cells, including BSUs that are being held by a connection segment waiting to send a Fwd_Count[i] due to N2 orphaning. The effect of this mechanism is to occasionally refresh the $\text{Limit}_{\text{link}}$ shared pool of buffers. **This avoids N2 orphaning, and by doing so, avoids N2 orphaned induced peer-blocking**.

Again, a buffer allocation of $\text{Limit}_{\text{LLCRmin}}$ takes potential N2 orphaning into account and ensures that it will have no impact on a connection's access to instantaneous bandwidth. N2 orphaning, like peer-blocking, is entirely avoidable given sufficient buffering.

E.3.3 N2 Starvation

N2 orphaning is a concern because there is no deterministic way to predict when downstream congestion will clear. If the congestion is not cleared or it becomes a steady state condition (i.e., the sustained nominal receiver drain rate is less than the forward direction LLCR), this leads to a condition that we call **N2 starvation**. N2 starvation occurs when some number of receiver buffers are N2 orphaned over a sustained period. This has the effect of semi-permanently removing the orphaned N2 buffers from the receiver's logical link buffer pool. The sustained aggregate of these N2 orphaned connections is the amount of N2 starvation for the logical link.

E.3.4 N2 Starvation Avoidance and Deadlock Prevention

If the $\text{Limit}_{\text{link}}$ is sufficiently restricted, certain pathological conditions exist such that N2 starvation can equal $\text{Limit}_{\text{link}}$ for a QFC link. Under certain pathological conditions, this can occur network-wide and can lead to QFC **deadlock**. QFC deadlock is a pathological, network-wide state where no data can be forwarded because each receiver's N2 starvation causes each transmitter to reach a link blocked state.

The logical link shared pool replenishment (using the Fwd_Count[link] mechanism) makes QFC deadlock a very remote possibility. Nonetheless, there exists a pathological situation where, if the number of buffers is small enough for an active group of connections C, network-wide N2 starvation and QFC deadlock may still occur.

To ensure that starvation and deadlock are highly unlikely and to insure that QFC deadlock cannot persist, the receiver and transmitter are required to implement two behaviors whenever buffer sharing is used. First, the receiver and transmitter shall allocate enough buffers to the $\text{Limit}_{\text{link}}$ to meet the buffer requirements established by Equations [4] and [5] of Section 7.3. This limit insures that, combined with the Fwd_Counter[link] replenishment mechanism, N2 starvation is unlikely.

Finally, to insure that deadlocks are prevented, one of the four necessary conditions for a resource deadlock must be made false. These conditions are: (1) mutual exclusion, (2) hold and wait, (3) circular wait and (4) no preemption. In the QFC protocol, the fourth condition is made false and cell buffers held by a stuck flow for longer than a time-out period are pre-empted by the system and released to other flows. Since, at least one of the four conditions are invalid, the protocol is proved to be free of permanent resource deadlocks. The time-out period may be large and the property still holds.

To mitigate effects of transient deadlocks, a detection scheme based on a buffer occupancy threshold may be used. Once a deadlock is impending, the system may reallocate buffer resources preferentially to certain flows and negate the second necessary condition for deadlocks. This will allow flows to drain in some predetermined order and mitigate the effects of transient deadlocks.

E.3.5 Model B Cost/Performance Trade-offs

The question of a cost/performance design point and CAC criteria for Model B can now be theorized. An approximate ceiling on the maximum number of established connections (C) at any point in time and the probable fraction of active connections $AC \subseteq C$ would be estimated of a QFC logical link. The minimum buffers required is defined by Equation [5] in Section 7.3 as roughly 1 buffer per connection plus some room for link BSU updates. To ensure no performance degradation, $\text{Limit}_{\text{LLCRmin}}$ buffers for each of the simultaneously active connections can also be provisioned. This last statement may be overkill, however, since, if AC is high, connections will not get access to full LLCR anyway.

If more connections become active, the only possible response is to decrease the per-connection $\text{Limit}_{\text{vcdef}}$ allocation, increase $\text{Limit}_{\text{link}}$, allow a higher degree of logical link buffer pool sharing or reject new ABR call requests on the logical link[5]. Since this condition only occurs when many connections are simultaneously active, per-connection access to LLCR bandwidth is less relevant because the logical link bandwidth is already substantially divided by the need to provide the active connections fair access to the available QFC bandwidth. Like the Model A case, it is likely that the QFC logical link can still operate at 100 percent LLCR goodput.

E.4 Analysis of Model C

As noted earlier, Model C is not a practical configuration as shown. The efficacy of Model C depends on the transmitter node taking control of the QFC protocol. Under this assumption, Model C would become functionally equivalent to Model A or B depending on the policies established by the transmitter. The analysis of those section would then apply.

5 In this latter case, routing avoidance mechanisms or crank-back routing can provide alternate routes for ABR connections.

E.5 Buffer Sharing

It has been shown that the QFC protocol inherently supports distributed allocation of memory buffers among the connections across a logical link. Figures [E.2] and [E.4] illustrate the required memory size for a variety of design points. With these figures, a qualitative parameter called the shared factor (SharedFac), is introduced which indicates the extent to which buffers are shared. This parameter is indicative of the percentage of the total connections that share any particular link buffer. The lowest value, SharedFac = 0, indicates no sharing (Model A). Buffers for the link are evenly and exclusively divided among the connections. The highest value, SharedFac = 100, indicates complete sharing. This provides an entire spectrum of design points which balance memory size (cost) and performance. Equation 3 defines the required number of buffers given the choice of the maximum supported established connections, C, and SharedFac.

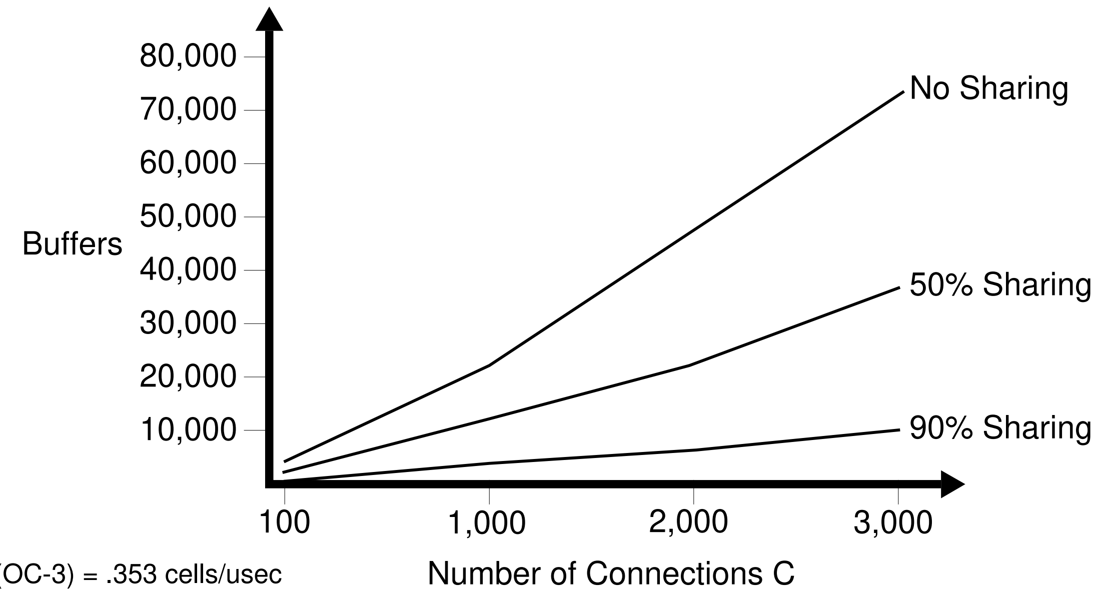
$$\text{Buffers (Model A)} = C \times (N3 + N5 + (N2 \times N4))$$

$$\text{ReductionFactor} = \begin{pmatrix} 1 & \text{if SharedFac} = 100 \\ 1 - \dfrac{\text{SharedFac}}{100} & \text{if SharedFac} < 100 \end{pmatrix}$$

$$\text{Buffers (Model B)} = \text{ReductionFactor} \times \text{Buffers (Model A)}$$

Equation [E.2]: Number of Buffers for Model B Given C and SharedFac

Figure E.2 analyzes the amount of buffering required for a range of connections for a 100 meter link running at an OC-3 LCR. Figure E.2 is a typical LAN case and shows the value of buffer sharing when C is large even in short-haul links. In a short-haul link, N3 is typically small, 1-3 cells. Therefore, N3 is dominated by the N2 and N4 values.



LCR(OC-3) = .353 cells/usec

RTT(100m) = .5usec

$\lceil N3 \rceil$ = LCR (OC-3) x RTT(100m) = 1

N2 = 3, N4 = 7, N5 = 2, M = 21 (M dominates N3)

Figure E.2. LAN (Short-Haul) Buffer Requirements with Buffer Sharing

Buffer sharing allows a receiver to either reduce cost by having fewer buffers or, for a fixed amount of buffering, to accept more calls. A simple example of an elastic buffer sharing model is shown in Figure E.3. For the figure, assume that the receiver only has 12,500 buffers to assign to the ABR service. As C expands from 100 to 3000, we have to accept a greater degree of buffer sharing (SharedFac) because our total buffer pool is fixed at 12,500. In the example, additional buffers are added to the pool as needed until all 12,500 buffers are assigned (happens at approximately 1,000 connections with a SharedFac of 48%). After all of the buffers are assigned, we have to either reject ABR calls or allow a higher SharedFac. The example algorithm allows up to 2,000 additional calls but keeps the SharedFac below 85%. If the nominal number of active connections, AC, is lower than 17% of C (17% is the SharedFac when C = 3,000), transient peer-blocking artifacts will not be generally felt.

Since most receivers will have a relatively static pool of buffers to assign to the ABR service, it makes sense to consider allowing the amount of buffer sharing to increase as the number of connections grows. A dynamic buffer sharing function of this type can be based on real-time measured or empirical values of AC.

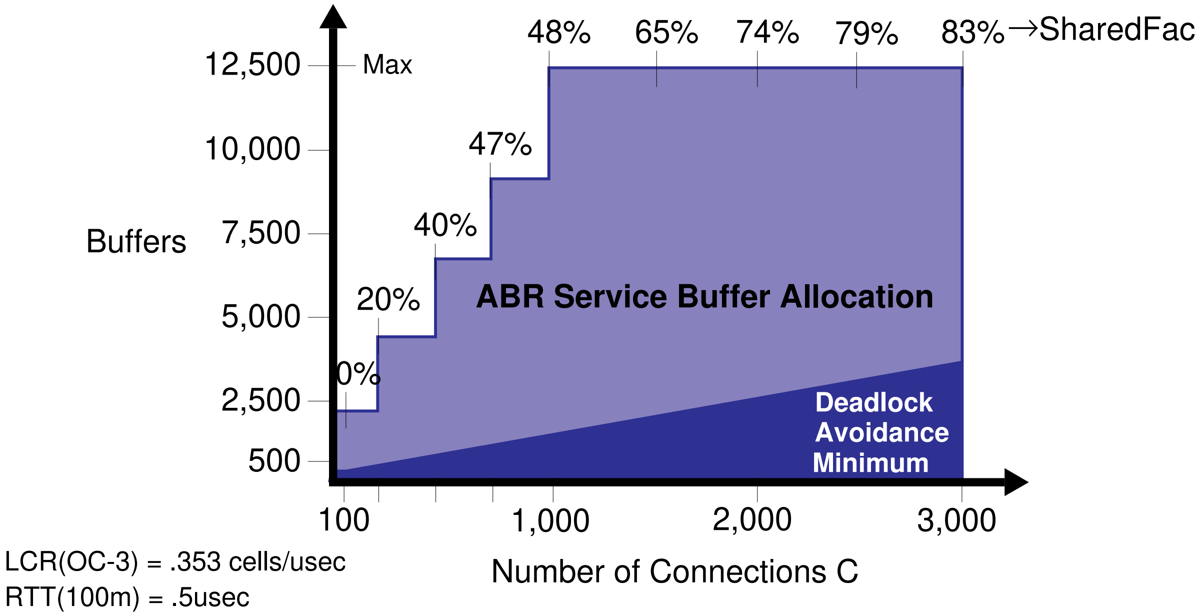
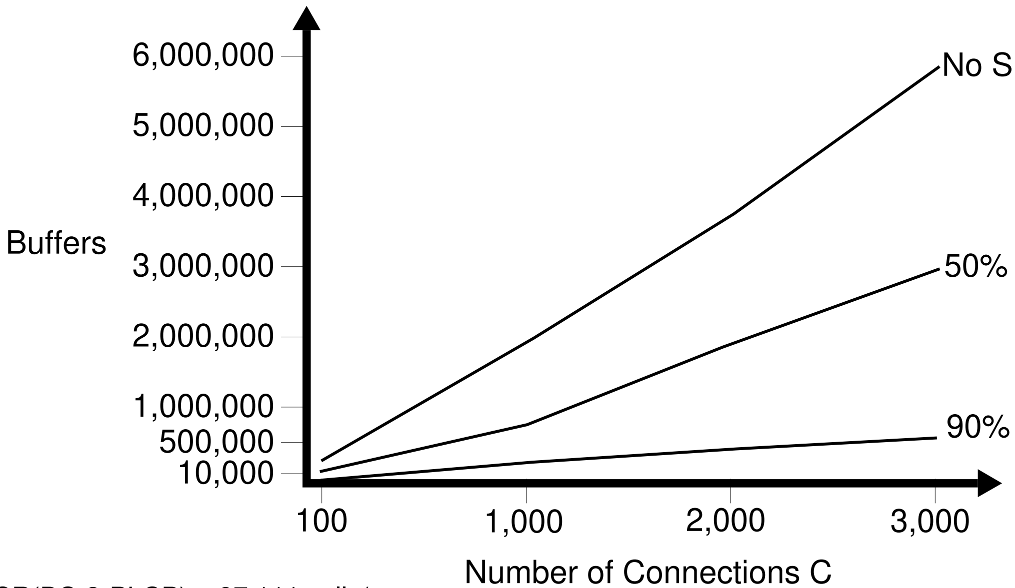


Figure E.3. Example Shared Buffer Allocation with Fixed Resources

The shared buffer allocation problem for a long-haul link is different than a short-haul link because the N3 value, based on RTT, becomes an overriding factor. (Remember that N3 was insignificant in the example of Figure E.2.) In the long-haul case, the N2/N4 sharing issues can be discounted because the total number of buffers is large for even small C.

The long-haul link example of Figure E.4 is similar to the short-haul example of Figure E.2 except that the link is 2,000 kilometers instead of 100 meters and the line speed is a DS-3 instead of a an OC-3. It is clear by Figure E.4 that N3 buffer sharing is very important. Without buffer sharing, when 3,000 ABR connections are accepted on the DS-3 link, the total buffering requirements exceed 312 MBytes of storage. With a SharedFac of 80%, this number can be reduced to 62.5 MBytes and for 90%, down to 31.26 MBytes. At first glance, 31 MBytes seems like a lot of memory. However, in view of the effectiveness of QFC, (that is, 100% goodput at LLCR) and the high monthly cost of a cross-country DS-3 service, the economic benefit is clear.



LCR(DS-3-PLCP) = 97.111 cells/ms

RTT(2,000km) = 20ms

$\lceil N3 \rceil$ = LCR (DS-3-PLCP) x RTT(2,000km) = 1943

N2 = 3, N4 = 7, N5 = 2, M = 21 (N3 dominates M)

Figure E.4. WAN (Long-Haul) Buffer Requirements with N3 Sharing

Appendix F: Rate ABR Support in QFC Domains

QFC adaptation of an Explicit-Rate (ER) based service (termed ER in the rest of this appendix) may be accomplished by termination of the ER service at the QFC node through a Virtual Source and Virtual Destination (VS/VD) control loop termination. This effectively terminates the ER span of control and couples two connection segments, one ER and one QFC, to form a larger connection segment. This option for QFC-ER adaptation is fairly obvious and will not be addressed in this appendix. Rather, an alternative method will be described — one that is arguably less expensive than VS/VD termination.

A QFC-capable node may support per-connection cell rates and may generate ER RM cells in full compliance with the ER algorithms for an adjoining ER node. Also, since QFC is required to forward source generated ER RM cells through a QFC logical link (Section 7.1.1), end-to-end ER is supported. Therefore, a QFC device can interface another device that only supports an ER service with complete transparency. The zero loss and best link utilization that are characteristic to pure QFC links along a path containing ER devices will enhance the effectiveness of the ER algorithm. Transparently embedded QFC links will make the end-to-end connection more responsive, minimize feedback time and facilitate effective control.

This type of adaptation is termed Transparent ER. There are many ways that a QFC node could support Transparent ER. One such mechanism is outlined in the rest of this appendix. The robustness of QFC is used as a guard against unplanned congestion, while the ER algorithms manage an individual connection's access to link and node bandwidth.

F.1 Transparent ER — An Implementation Example

F.1.1 Signaling

The signaling procedures of ER are not yet devised[1]. However, it is safe to assume that these procedures will be in place before ER connections are used in any wide-spread manner. (An ER connection is defined as a connection that contains at least one ER device along its established path end-to-end between two terminals.) It is safe to surmise that (1) it will be possible to discern if a connection is an ER connection or not and (2) that certain ER parameters like Initial Cell Rate, Minimum Cell Rate etcetera, will be defined as part of the signaling procedures. For the rest of this example, it is assumed that these signaling capabilities already exist. That is, a calling party may signal for a QFC connection and, optionally, provide the requisite ER parameters to traverse any ER hop encountered in the connection path setup en-route to the called party.

It is assumed that a calling party will know whether its attached link is QFC and/or ER capable. In this example, a calling party may request (1) an ER call (which may include one or more QFC hops internally but is ER at the calling party's edge), (2) a QFC call (which must be QFC end-to-end) or (3) a QFC/ER hybrid call (local access is primarily QFC but source/

[1] Unfortunately, because the ER signaling extensions to UNI 3.1/4.0/5.0 appear to be extensive but are still largely undetermined, it is difficult to claim true signaling interoperability at this time. However, since QFC signaling is so trivial, it is possible to make some generalizations now regardless. This section will be updated as ER signaling solidifies.

destination behavior of ER is also followed). Refer to Figure F.1 below. Case 1 of Figure F.1 is an example of (1) above, where both terminals are ER-only capable and a QFC node capable of supporting ER is transparently contained in its path. Case 2 of Figure F.1 is an example of (3) above, where both terminals can support QFC and ER and where there is an ER-only hop in the connection path.

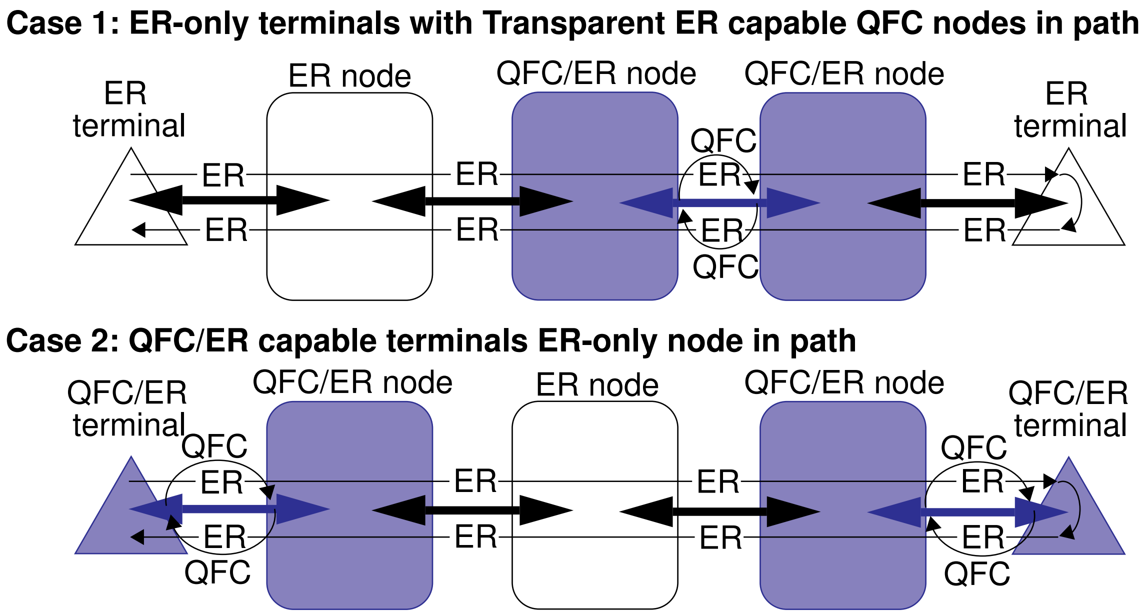


Figure F.1. Example Uses of Transparent ER Using QFC

When a calling party signals for a QFC connection and includes ER parameters (QFC/ER hybrid), it is implicitly agreeing that it may be routed through an ER hop. If a calling party signaling for a QFC connection does not include ER parameters (QFC only), it may not be routed through an ER hop. However, a QFC-only calling party may re-issue a call request if the QFC call cannot be routed in a pure QFC path.

A calling party that does not signal for a QFC connection (whether capable of QFC on the attached link or not) and includes ER parameters is implicitly allowing the use of QFC hops that support Transparent ER as long as the ER behavior is not compromised at the calling party's point of access. In this case, the burden is on QFC to have minimal burden on the calling party procedures. Since QFC is hop-to-hop and does not require end-to-end call setup procedures (Section 9.2) or traffic management behavior, this is trivially accomplished by QFC. In this later case, since the calling party does not know that there is a QFC segment in its connection path, no special requirements exist for the source ER behavior.

F.1.2 QFC/ER Capable Terminal Source Behavior

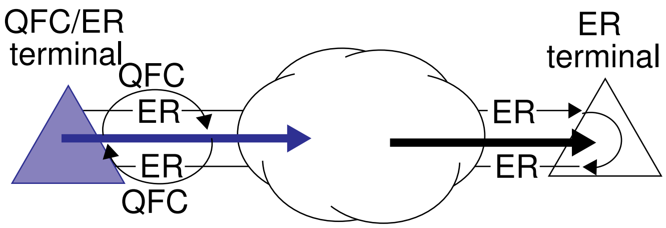


Figure F.2. QFC/ER Capable Terminal Source Behavior

At call setup, if the calling or called party, discovers (in SETUP for called party and CONNECT for calling party) that there is at least one ER device along the path of the connection[2], then that connection is characterized as an ER connection. In this case, both QFC and ER procedures will apply to the connection setup, maintenance, cell forwarding operations and teardown. Alternatively, in the same call setup scenario, if no ER hops were encountered, then the connection is characterized as a QFC-only connection and only QFC procedures are used.

For ER connections, the QFC/ER terminal source generates and processes RM cells in compliance with ATM Forum specifications. The cells for the ER connection are paced according to the rate allocation for that connection, but they are also subjected to QFC limits on the outgoing link. In other words, a cell that belongs to an ER connection can be transmitted if it can be transmitted within the confines of the rate allocation for that connection **and** if the credit allocation for that connection is not exhausted based on the QFC specification in this document.

If the connection's QFC buffer allocations were dimensioned to allow full LLCR transfer and were not shared, QFC flow control would only kick in (limiting the ER rate) if unexpected congestion were experienced in the network. Without QFC flow control in this scenario, ER may have dropped those cells downstream.

There are no modifications required for the processing requirements of cells that belong to QFC connections.

[2] The determination of an ER hop in a connection segment would have to be an extension to the SETUP and CONNECT signaling procedures.

F.1.3 QFC/ER Capable Terminal Destination Behavior

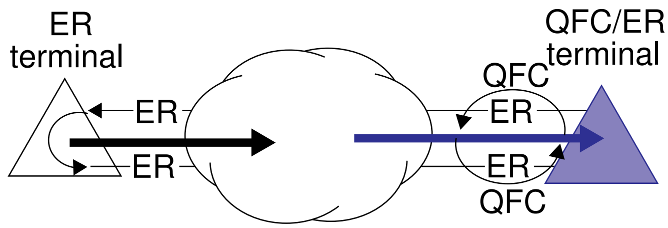


Figure F.3. QFC/ER Capable Terminal Destination Behavior

The destination processes RM cells in compliance with ATM Forum specifications and, like the source case, operates both the rate management algorithms of ER and the flow control algorithms of QFC. Again, if the QFC receiver buffers are dimensioned for LLCR transfer and are not shared, only real terminal congestion will cause a cell to be held due to QFC flow control.

F.1.4 Transparent ER Switch Behavior

For this implementation example, let each QFC switching device capable of supporting ER allocate an aggregate bandwidth (**Brate**) for each one of its outgoing links and guarantee Brate for the ER connections using that link. The rest of the link capacity, as well as any allocated bandwidth not used by ER or other ATM service classes (like CBR), may be used by QFC traffic, thanks to the excellent link bandwidth utilization characteristics of QFC ABR service. Figure F.4 demonstrates the Brate Concept.

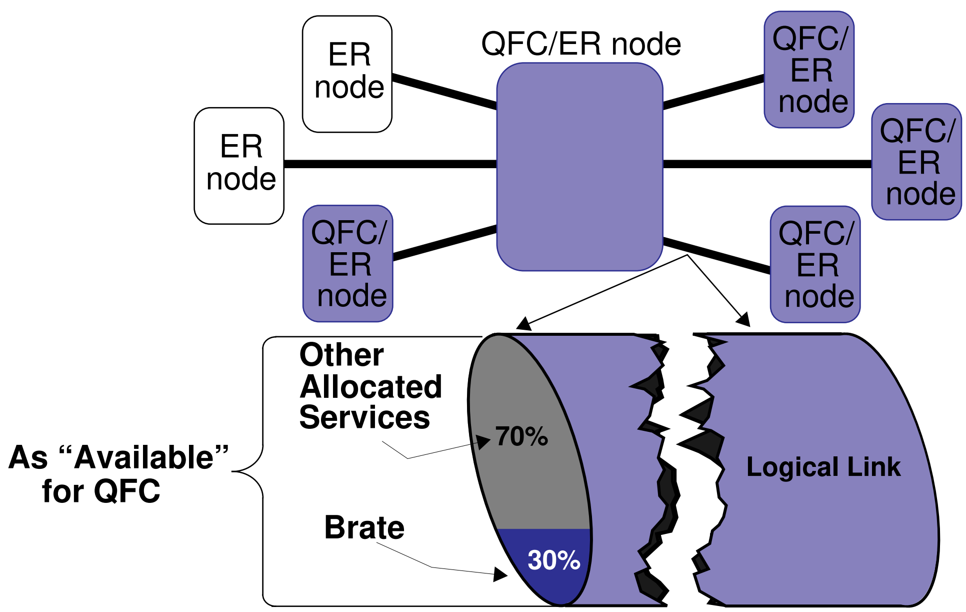


Figure F.4. Example Bandwidth Partitioning for Allocated ER Bandwidth (Brate)

In this example, network administrators may define preferences for a range for Brate consisting of a minimum rate, BrateMin, and a maximum rate, BrateMax. QFC devices may adjust Brate within those limits dynamically in response to changes in ER and QFC traffic loads observed on a link, due to variations in the type of established connections over time or due to dynamic rate requests from an ER source/destination via ER RM cells.

The QFC/ER switch processes RM cells in compliance with ATM Forum Specifications. Rate allocations for ER connections using a specific link will need to be computed based on Brate allocated for ER on that link. The distinction between ER connections and QFC connections are completely transparent to the link protocol for QFC devices defined in this document. In other words, a switch's logical link treats all QFC/ER connections equally from the perspective of the QFC protocol transmitting a cell belonging to a QFC connection based on the buffer allocation for that connection. In the case of an ER connection using QFC, there is the additional step where the cell must be checked against its current ER rate. The ER connections using QFC would get prioritized access to Brate bandwidth over any *pure* QFC connections.

Appendix G: Automatic Synchronization of QFC PVCs

This appendix describes a simple mechanism to automatically synchronize a QFC PVC between two nodes without signaling or network management station (NMS) interaction. The hop-by-hop nature of QFC supports this appealing feature in a fully distributed manner, requiring no end-to-end synchronization mechanism.

A simple link mechanism, as described below, can be used to activate QFC for a PVC on a link. These procedures use the same QFC protocol building blocks described in Sections 6, 7, 9, and 10. No new protocol mechanism or procedures are required.

As the PVC is configured in each node in the path, each node will initialize the QFC links to its adjacent node(s) before sending any data. The mechanism described in this appendix guarantees the zero cell loss and best link utilization characteristics of QFC for the PVC connection. Like the signaling procedures of Section 9.2, these procedures can be used to ensure that data integrity is maintained while the segments of the end-to-end PVC are brought up. This can be done automatically and without human intervention, making this mechanism useful for QFC PVCs that are automatically brought up at node bootstrap or for any PVC (automatic PVCs or NMS generated PVCs) during automated fault recovery (recovery without NMS support).

G.1 QFC Rules for Automatic PVC Synchronization

The mechanism for activating QFC on a PVC simply uses BSC cells sent by the transmitter to initiate the activation process with standard BSU responses by the receiver to complete the initialization.

Each device is responsible for maintaining correctness with respect to the QFC specification before the transmission of the BSC cells and BSU responses. The sequence of steps taken in this activation process are described in detail in the following sections.

G.1.1 QFC Synchronization Initiation Events

A QFC node is triggered to initiate the process of activating QFC on its forward direction by one of the following events:

- basic QFC PVC setup is completed (i.e., the virtual channel is opened to send and receive cells) after QFC node power-up (automatic PVC mode),
- The downstream device has initiated an activation on the reverse direction or QFC activation has already been established in the reverse direction or
- the NMS has requested an activation and QFC is not already activated on the forward direction (NMS PVC setup mode).

G.1.2 Initiation Requirement

A QFC node shall ensure that it has enough resources to activate QFC in the forward and reverse directions of the PVC. If it lacks such resources, the PVC setup fails.

G.1.3 Transmitter Initiation

In order to initiate the synchronization process, a QFC device:

- sets Limit[i] to the global BSL_Limit_{vcdef}[1,2]
- sets Tx_Counter[i] and BSU_Fwd_Count[i] to values that do not allow cell transfer, that is, ($Mod_2{}^D$(Tx_Counter[i] - BSU_Fwd_Count[i]) > Limit[i]))
- sends a BSC cell.

G.1.4 Receiver Response

Upon initialization and prior to receiving the BSC from the transmitter the receiver:

- sets Limit[i] to the global $Limit_{vcdef}$[1,2]
- sets Rx_Counter[i] and Fwd_Counter[i] to values that do not allow cell reception, that is, Rx_Counter[i] = Limit[i] and Fwd_Counter[i] = 0, and
- waits for a BSC cell (discards any cell that is not a BSC, turns off QFC error trap for cell discard).

The receiver may accept a transmitter initiated QFC synchronization request after the receipt of a BSC message from the transmitter by:

- setting the Rx_Counter[i] to the received BSC(Tx_Counter[i]),
- setting the Fwd_Counter[i] to the received BSC(Tx_Counter[i]),
- setting Limit[i] to the global $Limit_{vcdef}$[1,2],
- setting N2_Counter[i] to zero, and
- sending a BSU record for that PVC.

G.1.5 Completing the Synchronization

When a BSU response is received from the receiver, the transmitter updates its QFC counters, as described in Section 7. **In this case, QFC is successfully activated on the PVC**.

The transmitter may send multiple BSC cells in order to activate QFC. However, if no BSU response is received from the receiver for some number of attempts, the QFC transmitter may stop trying.

---

1 The BSL_Limit_{vcdef} must be established by the QFC autoconfiguration procedures of Section 9.1.3 prior to setting up the first PVC (or SVC).

2 Optionally, like the Limit[i] IE override of the signaling procedures of Section 9.2, the NMS or node defaults may have established a Limit[i] override value not equal to BSL_Limit_{vcdef}.