

ATM Interface Technologies for an ATM Switching System

Hong-Shik Park, Yool Kwon, Young-Sup Kim, and Seok-Youl Kang

CONTENTS

- I. INTRODUCTION
- II. REQUIREMENTS FOR UNI/NNI
- III. INTERFACE ARCHITECTURE
- IV. ATM INTERFACE TECHNOLOGIES
- V. CONCLUSIONS

REFERENCES

ABSTRACT

Realization of the economical, reliable, and efficient ATM interface block becomes an important key to development of the ATM switching system when we consider new issues raised recently. In this paper, we summarize requirements for the ATM interface block and present the UNI (User Network Interface)/NNI (Network Node Interface) architecture to meet these requirements. We also evaluate the performance of the multiplexer adopting the various multiplexing schemes and service disciplines. For ATM UNI/NNI interface technologies, we have developed a new policing device using the priority encoding scheme. It can reduce the decision time for policing significantly. We have also designed a new spacer that can space out the clumped cell stream almost perfectly. This algorithm guarantees more than 99 % conformance to the negotiated peak cell rate. Finally, we propose the interface architecture for accommodation of the ABR (Available Bit Rate) transfer capability. The proposed structure that performs virtual source and virtual destination functions as well as a switch algorithm can efficiently accommodate the ABR service.

I. INTRODUCTION

According to increasing user's demands for broadband services such as LAN interconnection service, video conference and video on demand (VOD) service, diverse broadband services are being provided through ATM-based equipments and ATM networks. Especially, proliferation of the LAN interconnection service for business users accommodated in private networks is remarkable. Due to this trend, even a public ATM switching system naturally has various private ATM user-network interfaces (UNI's) as well as public ATM UNI's. In the implementation view point of the ATM switching system, it needs to adopt a more efficient and flexible interface architecture than ever before. This is because it has to accommodate various kinds of existing terminals which do not comply with the ATM protocol. Moreover, due to high speed operation of the switch fabric, most complex functions such as cell header conversion, addition of a routing tag, and OAM (Operation And Maintenance) cell processing have to be implemented in the ATM interface block of the ATM switching system. Recently, as ITU-T and ATM Forum have recommended new ATM transfer capabilities, various ATM layer functions also need to be implemented in the ATM interface block [1], [2]. Considering the issues mentioned above, implementation of a more economical and efficient ATM interface block becomes an important key to development of the public ATM switching system. For ATM

UNI/NNI interface technologies, we have developed a new policing device using the priority encoding scheme and a new spacer adopting the cell departure time control scheme to resolve output contention.

Organization of the paper is as follows. In section 2, we summarize requirements for the ATM interface block focusing on the issues raised recently. We present the UNI/NNI (Network Node Interface) architecture to meet these requirements in section 3. We also evaluate the performance of the multiplexer adopting the various multiplexing schemes and service disciplines. In section 4, we present core ATM interface technologies for the UNI/NNI. We describe our policing device based on the VSA (Virtual Scheduling Algorithm). It can reduce the decision time for policing significantly using the priority encoding scheme. We also propose the new spacer that can space out the clumped cell stream almost perfectly. This algorithm guarantees more than 99 % conformance to the negotiated peak cell rate. Finally, we discuss implementation of the ABR (Available Bit Rate) transfer capability. The proposed structure that performs virtual source and virtual destination functions as well as a switch algorithm can efficiently accommodate the ABR service.

II. REQUIREMENTS FOR UNI/NNI

We summarize requirements for the physi-

cal and ATM layer of the ATM UNI/NNI in the following.

- Type of physical interface

Table 1 shows the types of physical interfaces of the ATM UNI recommended by the international standard bodies.

Table 1. Physical interfaces of the ATM UNI.

Range	ITU-T (public)	ATM Forum (private)
high speed	-STM-4c -STM-1	-STS-3c -155 Mbits/s multimode fiber, STP* and UTP** interface -E4 -100 Mbits/s multimode fiber
medium speed	-51.84 Mbit/s (S _B only) -25.6 Mbit/s (S _B only)	-51.84 Mbit/s UTP interface -DS3/E3 -25.6 Mbit/s
low speed	-1.544, 2.048 Mbit/s	-DS1

* STP: Shielded Twisted Pair

** UTP: Unshielded Twisted Pair

Besides the above interfaces, non-ATM interfaces such as Ethernet LAN, DS3, and DS1E interfaces need to be considered to accommodate existing terminals and circuit emulation services. As the speed of the interfaces ranges from a few Mbits/s to several hundreds of Mbits/s, it is necessary for the ATM interface block to adopt an effective multiplexing scheme.

In case of NNI, it is desirable to accommodate the interfaces which comply with the Synchronous Digital Hierarchy (SDH) as well as

existing interfaces which are used to interface with the existing networks. The NNI architecture can be affected by the evolutionary strategy of the existing networks. For the virtual channel ATM switching system, STM-1 and STM-4 interfaces have been recommended up to the present.

- Operating speed of a switch fabric

The maximum speed of the interfaces that the ATM interface block can accommodate is limited by the operating speed of the switch fabric. Therefore, the operating speed of the switch fabric should be increased to at least 2.5 Gbit/s to enlarge diversity of the interface types. We can consider multiple connection set-up through the switch fabric for high speed service to overcome limitation of the operating speed of the switch fabric. But, this scheme may have difficulty in maintaining cell sequence integrity.

- Evolutionary strategy of the network

The interface structure may differ according to the ATMization strategy of existing networks. If we adopt the overlay network approach, interworking via the NNI's is desirable. For the replacement approach, we should consider interworking through the UNI's as well as NNI's. Here, interworking via UNI means accommodation of existing terminals via existing interfaces. In any case the interface structure may differ according to the locations where we implement the interworking functions within the network.

- CDV (Cell Delay Variation) and traffic shaping

CDV due to cell multiplexing within the ATM layer, insertion of the overhead in the physical layer and the slotted nature of the physical layer effects accurate operation of traffic control functions such as UPC (Usage Parameter Control) and CAC (Connection Admission Control). It also effects queueing performance of the switch fabric. Further, to increase the bandwidth utilization of the CAC, minimization of the CDV effect is desirable. As one of the methods to reduce the CDV effect, we can consider the use of traffic shaping in the switch node. It is required to add a traffic shaper to the transmitting part of the NNI interface of the switch node to guarantee the negotiated traffic parameters between two adjacent switch nodes. It is well known that if the worst case traffic due to the CDV occurs, performance of the output-buffer type of switch fabric severely deteriorates [3]. To improve performance of the switch fabric, we can also consider the use of a shaper in the ATM UNI interface block.

- Protection switching [4]

In order to avoid service interruption due to failure or performance degradation of links as well as switch nodes, protection switching capabilities are required. These capabilities need to be in the first place applied to permanent and semi-permanent VP/VC connections. Protection switching architecture, switching trigger mechanism, hitless protection switching, and

resource assignment method, etc. need to be well defined and prepared.

- ATM transfer capabilities

ITU-T has newly recommended ATM transfer capabilities such as DBR (Deterministic Bit Rate), SBR (Statistical Bit Rate), ABR (Available Bit Rate), UBR (Unspecified Bit Rate), and ABT (ATM Block Transfer) to specify a combination of QoS (Quality of Service) commitments and ATM traffic parameters that is suitable for a given set of applications and that allows for specific multiplexing schemes at the ATM layer [1]. To support these capabilities, an effective architecture for the ATM layer should be considered. Especially cost-effective implementation of the ABR transfer capability including a switch algorithm, a virtual source and a virtual destination is very important.

III. INTERFACE ARCHITECTURE

1. ATM UNI Interface Architecture

As the speed of the interfaces ranges from a few Mbits/s to several hundreds of Mbits/s, the ATM UNI interface block needs to have a modular and hierarchical multiplexing structure. The multiplexing structure mostly depends on the types of the interfaces and a cell delay due to queueing at each multiplexing level. A basic unit of a module is decided by the number of the links to be accommodated within a PBA

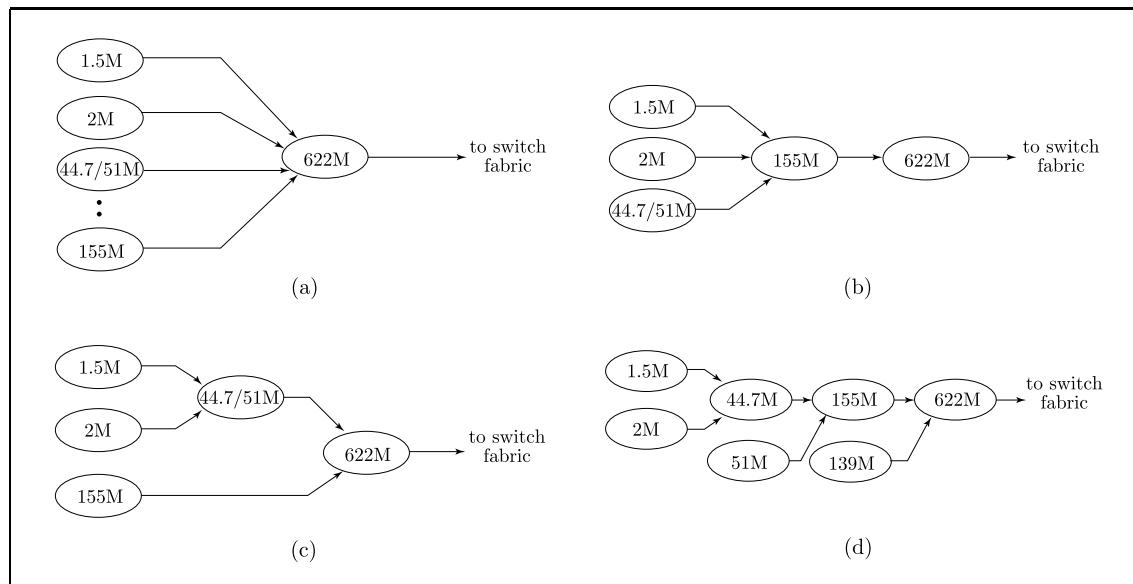


Fig. 1. Multiplexing structure.

(Printed Board Assembly) and traffic volume which a processor responsible for controlling the interface part can handle. If we assume that the speed of the input/output links of the switch fabric is 622 Mbits/s, we can consider several multiplexing structures. These are depicted in Fig. 1. Method (a) has a unique internal system interface and can easily accommodate a new interface. Expansion of the interface part is also simple. But it is very difficult to implement this structure because of limitation on the physical length of the internal system interface when the system size becomes large. Method (b) coincides with an improvement step of the switch fabric, that is, it can keep step with the gradual speed-up of the switch network easily. Method (c) aims to improve the difficulty in implementation of the internal high-speed system interface. This method multiplexes the

low-speed interfaces first to interface with the internal medium-speed system interface. This approach is easy to accommodate the medium-speed UNI. Method (d) can easily guarantee modularity and has merit in implementation. But, it suffers from many internal system interfaces and relatively large amount of cell delay.

Features of each method are compared and summarized in Table 2.

Fig. 2 shows an example of the ATM UNI interface block based on method (c). Low-speed interfaces are multiplexed first because of ease of implementation when the number of interfaces to be accommodated becomes large. For the multiplexing scheme, we can consider two methods. One method is to use the interrupt with the priority to inform the multiplexer the arrival of a new cell. The higher priority

Table 2. Features of each multiplexing structure.

method	a	b	c	d
item				
no. of internal system interface	small	normal	normal	large
implementation	very hard	normal	easy	easy
utilization of statistical multiplexing	good	good	good	good
cell delay	small	normal	normal	large
H/W complexity	low	normal	low	high
broadcasting	easy	normal	normal	hard
concentration	bad	normal	good	good

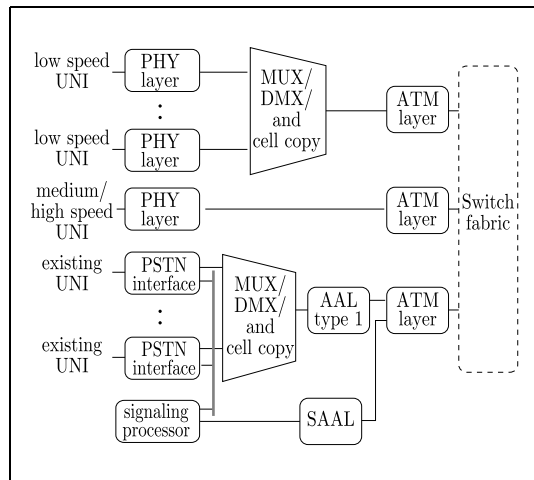


Fig. 2. UNI interface architecture.

is given to the high-speed interface to reduce the required buffer size. The other is to use a polling scheme. In this case, additional control overhead is required to guarantee the fairness between different interfaces.

Fig. 3 shows the cell loss and cell delay performance of the multiplexer adopting dif-

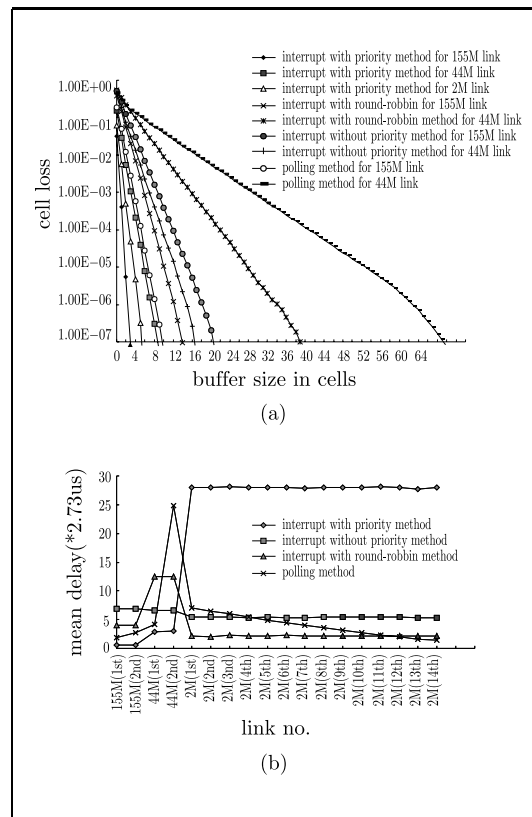


Fig. 3. Performance of the multiplexer, (a) cell loss, (b) mean delay.

ferent multiplexing schemes and service disciplines. In the simulation, we consider the multiplexer with 2 high-speed (155M) links, 2 medium-speed (44M) links and 14 low-speed (2M) links. We assume that all input traffic streams follow Bernoulli processes with the parameter $p_i = 0.2, 0.8, 0.8$, respectively. We note that the interrupt scheme has better cell loss performance than the polling scheme. Even in the cell delay view point, the interrupt scheme is better. In case of the interrupt with priority method, fairness in the cell delay point

of view is not so good because of its intrinsic nature. But, cell delay in 2M links is not so critical if we compare it with the cell transmission period ($212 \mu s$) of 2M links.

For demultiplexing, we can consider two methods. One is to use broadcasting. The demultiplexer broadcasts cells to all interfaces belonging to it and then each interface filters the received cells with its own address. The other scheme is to send cells only to the interface that the demultiplexer wants to send to. Identification of each interface in the method where the intermediate multiplexing stages are used can be done using a dedicated VPI (Virtual Path Identifier)/VCI (Virtual Channel Identifier) to each interface or an additional routing tag. To take system expansion into account, identification by the VPI/VCI may be preferred to use of the routing tag, because expansion of the routing tag seriously effects implementation of the system hardware. On the other hand, use of the routing tag has a benefit in realization of the cell copy function. In case of the multiplexing scheme where the intermediate multiplexing stages are used, the cell copy function is required to provide multicasting or broadcasting services. For this function, a bit mapping scheme can be used. In case of the multicast-ing, each bit in the routing tag is one-to-one mapped to each interface to identify the location of the interface. All interfaces whose corresponding bits in the routing tag are set receive the multicasting cells simultaneously and convert the VPI/VCI's to the new ones negotiated with the users.

2. ATM NNI Interface Architecture

The same architecture mentioned above can be adopted for the ATM-NNI. But for the ATM-NNI, implementation of the Signaling System Number 7 (SS No.7) should be carefully considered. To accommodate the SS No. 7, the Message Transfer Part Level 3 (MTP -3) as well as the underlying low layer ATM protocols are to be realized. The existing MTP-3 has been modified for the application to the ATM network [5]. Messages exchanged between signaling points (SP) when a change-over due to the failure of a signaling link (SL) occurs, have been changed to XCO (eXtended Change-over Order) and XCA (eXtended Change-over Acknowledgment) messages. Maximum length of the signaling message has also been changed from 273 bytes to 4K bytes. Because a SL uses a VCC (Virtual Channel Connection) in the ATM-NNI, QoS requirements on the SL need to be rigorously guaranteed. For this reason, a semi-permanent VCC for the SL is preferable. MTP-3 performs signaling message handling function (SMHF) and signaling network management function (SNMF). The SMHF transfers signaling messages to the proper SL's or users. The SNMF controls routing of signaling messages using predetermined data. It also controls reconfiguration of the network in case of the failure of the SL's. It is reasonable that the SMHF is implemented within each ATM-NNI interface part because it requires real-time operation. Instead, in case of the SNMF, we can

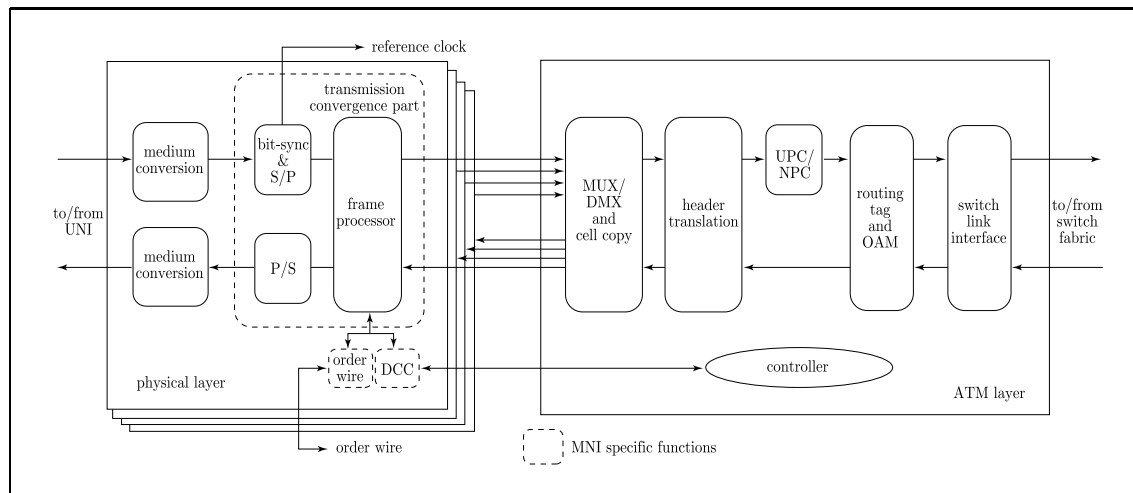


Fig. 4. Block diagram of UNI/NNI interface.

expect efficiency by centralization of it. As methods for accommodation of SL's, we can consider two approaches, that is, centralized and distributed accommodations. Centralized approach puts all SL's into one subsystem together. This method can use resources efficiently and management of SL's becomes easy. But, the amount of internal messages to be exchanged becomes large due to separation of MTP-3 and B-ISUP (B-ISDN User Part). The distributed approach has benefits in latency time and expansion of system capacity. Features of two approaches are summarized in Table 3.

IV. ATM INTERFACE TECHNOLOGIES

1. UNI/NNI Structure

The UNI/NNI interface block is divided

into three parts; the physical medium conversion part, the transmission convergence part, and the ATM layer part. The block diagram of the UNI/NNI module is shown in Fig. 4.

Table 3. Features of two approaches to accommodate SL's.

items	centralized	distributed
OAM of SL's	easy	relatively hard
message latency time	bad	good
expansion of system capacity	limited	easy
duplication of resources	small	many

In this section, we discuss the core ATM interface technologies such as UPC/NPC, spacer and implementation of ABR transfer capability.

2. UPC/NPC

To prevent the ATM network from reaching an unacceptable congestion level due to unexpected traffic variation or due to intentional excess of the negotiated parameters, UPC/NPC function monitors whether or not a traffic flow on every VCC conforms to the negotiated traffic parameters. ITU-T suggests two examples of algorithms, the Virtual Scheduling Algorithm (VSA) and the Continuous State Leaky Bucket Algorithm [1]. Here, we give an implementation example of the UPC/NPC using the VSA.

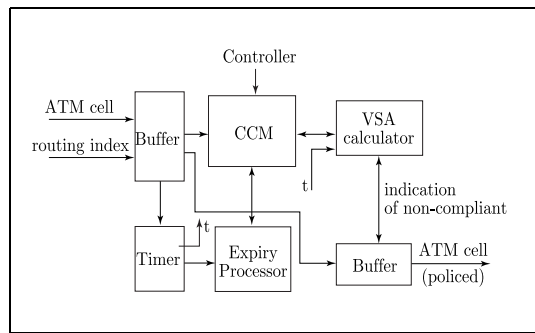


Fig. 5. Block diagram of the UPC part.

Fig. 5 shows a block diagram of the UPC/NPC. CCM (Connection Context Memory) stores four parameters; TAT (Theoretical Arrival Time), T (peak emission interval), τ (CDVT), and E (expiry bit) required for the VSA. It also stores the mode of operation for each connection. It provides four modes of operation according to the CLP (Cell Loss Priority) bit and the policing actions (tagging and discarding). CCM is accessed by the VSA calculator whenever multiplexed cells arrive at the device. The TAT value increases steadily as time passes until it reaches a maximum

value determined by the coding size and then returns to zero. This procedure makes it difficult to simply compare $TAT+T$ with t or $t+\tau$. To solve this problem, an expiry process is used on the TAT of every connection. All the TAT 's are scanned cyclically and are declared either 'valid' or 'expired' depending on the criterion built in the expiry process [7]. The VSA calculator performs the VSA through addition and comparison using TAT , T , τ and t . The VSA calculator processes three decision parts in the VSA in parallel instead of sequentially using the priority encoder so that the time to complete the algorithm can be reduced to a great extent. If we use a 23M clock, it takes only about 850 ns to police an incoming cell.

If we use 24 bits to code the TAT , 22 bits for T (where, 14 bits are used to represent the integer part), 12 bits for τ , and 24 bits for t , the peak cell rate P_c is given as

$$P_c = 10^6 / 2.726(k + m/256) \quad (1)$$

$$\text{for } 0 \leq k \leq 2^{14} - 1, 0 \leq m \leq 2^8 - 1,$$

$$m + k \neq 0.$$

From (1), we know that the bit rate to be handled by the device ranges from 9.493 Kbits/s to 39.82 Gbits/s. We can also formulate the relative variation between two consecutive values as

$$\Delta P_c / P = 1.0694 \times 10^{-8} P_c. \quad (2)$$

This relationship tells us the ATM peak cell rate granularity. For example, two nearest values to 100 Mbits/s are 99.79 Mbits/s and 100.16 Mbits/s, and the relative variation is 0.371 %.

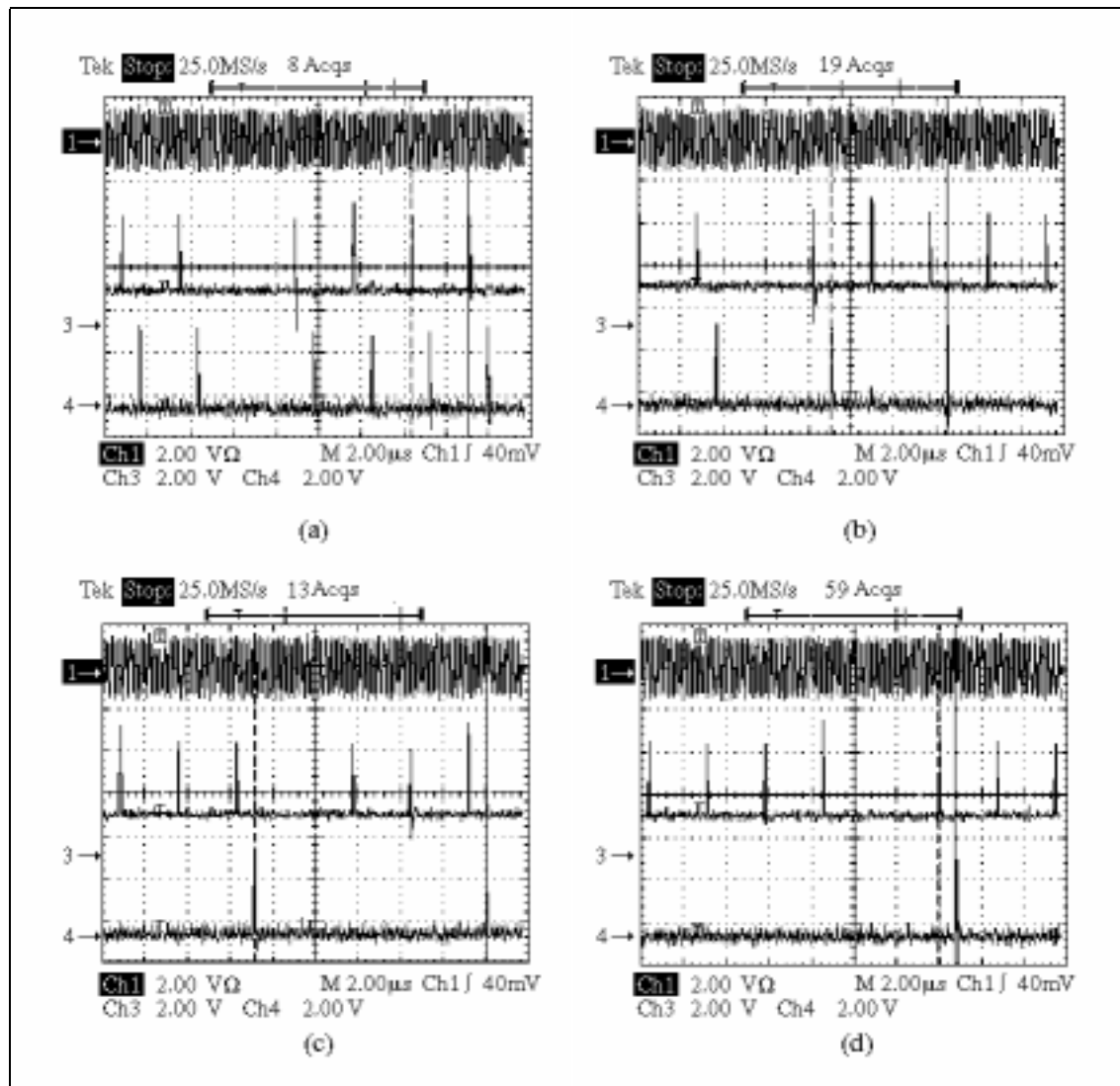


Fig. 6. Snapshot of waveforms.

Fig. 6 shows the snapshot of the master clock, the cell sync input and the cell sync output signal of a connection that are observed by an oscilloscope. The channel 1, 3 and 4 in the figure represent the master clock, the cell sync input and the cell sync output signal, re-

spectively. The master clock is used by all signals in the device for synchronization purposes. The cell sync input signal represents the starting time of the arriving cell at the device. The cell sync output signal indicates the starting time of the policed cell by the device.

Fig. 6 (a), (b), (c), and (d) represent the measured waveforms for four cases where the negotiated peak cell rate is 155, 155/2, 155/4 and 155/8 Mbits/s. For all cases, the input of the connection is VBR (Variable Bit Rate) traffic with the peak cell rate of 155 Mbits/s, the average cell rate of 70 Mbits/s and the CDV of $5.44 \mu\text{s}$. Fig. 6(a) shows that the device passes all the incoming cells without any cell dropping since the user sends the cells at the same rate as the negotiated peak cell rate of 155 Mbits/s. From Fig. 6(b), we see that the device passes the half of the incoming cells since the user emits the cells at two times the negotiated peak cell rate of 155/2 Mbits/s. Fig. 6(c) and (d) show that the device passes a quarter ((c)) and one eighth ((d)) of the incoming cells since the negotiated peak cell rate in each figure is 155/4 Mbits/s and 155/8 Mbits/s.

3. Shaper

If the UPC function considers the CDV, the generation of the worst case traffic (WCT) is inevitable. The WCT is defined as cell clusters that are embedded in the traffic flow at the link transmission speed and can pass a policing device transparently. This WCT can induce severe degradation in queuing performance of the ATM network. To increase performance of the switch fabric, we can consider the use of a shaper in the ATM UNI interface block. It is also necessary to add a shaper to the transmitting part of the NNI interface to guarantee the negotiated traffic parameters between adjacent

switch nodes.

An implementation example of the shaper based on the linked list mechanism is given in Fig. 7.

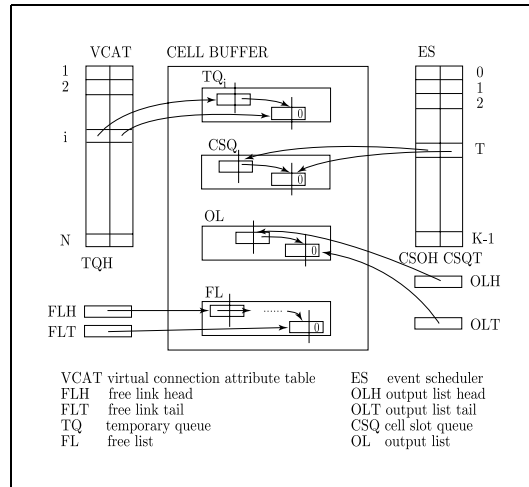


Fig. 7. Structure of the peak rate spacer.

K is determined by

$$K = \min(\delta, \max(Pi_i) + 1), \quad 1 \leq i \leq N, \quad (3)$$

where δ denotes maximum value of the CDV, Pi_i is the peak interval of the i th connection and N is the number of VCC's accommodated together in the input link. Our shaper can guarantee the negotiated peak cell rate almost perfectly even though contention of cells in the output link of the shaper occurs. For detailed explanation, refer to [8]. If we let δ be a CDV bound based on the 10^{-10} quantile of the cell delay distribution, the maximum cell buffer size of the shaper is the sum of the total TQ size and the CSQ size. The total TQ size n_{TQT}

is given as

$$\begin{aligned} n_{TQT} &\leq \sum_{i=1}^N \{[\delta/(PI_i - 1)] - [\delta/PI_i(PI_i - 1)]\} \\ &\leq \sum_{i=1}^N \delta/PI_i \\ &\leq \delta, \end{aligned} \quad (4)$$

using the boundary condition for N

$$0 < \sum_{i=1}^N \frac{1}{PI_i} < 1. \quad (5)$$

Maximum CSQ size is given as δ for a large K . Therefore, the maximum cell buffer size becomes 2δ . If we use the right 10^{-10} quantile value of the cell delay distribution of the M/D/1 queue when the utilization is 0.8 as a δ value, the required maximum cell buffer size becomes at most 108. Therefore, we can economically implement the shaper used by all connections multiplexed together within the UNI/NNI link with the reasonable cell buffer size. Fig. 8 shows the distribution of the cell interarrival and interdeparture time of the shaper with heterogeneous CBR sources. We consider 10 multiplexed sources; one CBR source with the peak interval $PI = 5$, three CBR sources with $PI = 10$, and six CBR sources with $PI = 20$.

We find that more than 99 % of departing cells conform to the negotiated peak interval. We note that there is no cell which departs the shaper faster than the negotiated peak cell rate. Therefore, we can establish a clear bandwidth allocation strategy and also expect more efficient bandwidth utilization.

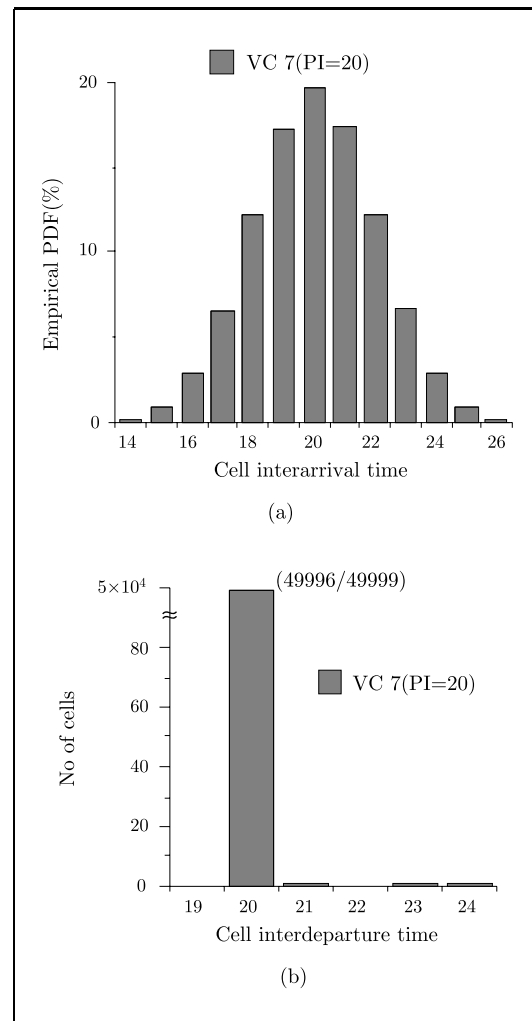


Fig. 8. Distributions of the cell interval and interdeparture time for the $PI = 20$ CBR source in the heterogeneous input case. (a) PDF of the cell interarrival time for the $PI = 20$ CBR source, (b) The cell interdeparture time distribution for the $PI = 20$ CBR source.

4. Implementation of ABR Transfer Capability

An ABR service is intended for data ap-

plications that can adopt to time-varying bandwidth and tolerable unpredictable end-to-end cell delays. The ABR traffic has access to bandwidth only when no CBR/VBR traffic is waiting for transmission. Thus the ABR traffic is allowed to use bandwidth that would be unused, increasing the link utilization without effecting the QoS of CBR/VBR connections.

The end-to-end rate-based control mechanism is used for the flow control of the ABR service to allow much more flexibility in switch implementation. Here, we give a brief explanation on the ABR flow control mechanism. A source sends a forward RM (Resource Management) cell to the network following *Nrm* data cells with the EFCI (Explicit Forward Congestion Indication) bit = 0 in a cell header. A source rate is not changed before the source receives a backward RM cell. A switch node calculates an ER (Explicit Rate) value or sets CI (Congestion Indication) bit of the RM cell according to network congestion state. Receiving the forward RM cell, a destination sends a backward RM cell to the source. On receiving the backward RM cell, the source changes or holds the transmission rate by feedback information.

An ABR connection may be segmented into two or more separately controlled ABR segments. Each ABR control segment is sourced by a virtual source (VS). A VS assumes the behavior of an ABR source endpoint. Each ABR control segment is terminated by a virtual destination (VD). A VD assumes the behavior of an ABR destination end-

point.

To accommodate the ABR service efficiently, suitable switch algorithm, VS and VD functions should be implemented. Though many algorithms such as EFCI, EPRCA (Enhanced Proportional Rate Control Algorithm), ERICA (Explicit Rate Indication for Congestion Avoidance), and CAPC2 (Congestion Avoidance using Proportional Control) [9] have been proposed for the switch algorithm, the performance of the algorithms has not been proven perfectly yet and they seem to be suitable for the private switch if we consider interoperability between ATM transfer capabilities in the public ATM switch. Development of switch algorithms suitable for the public switch remains for further study. Whatever we use, there can be many ways to implement the ABR transfer capability. To minimize duplication of functions within the switch and to simplify control actions related to the switch algorithm, VS and VD, it is reasonable that functions for the ABR transfer capability be implemented in the egress UNI/NNI interface blocks considering the output-buffer type of ATM switch fabric. In this case, ABR cells and related RM cells received at the ingress UNI/NNI interface blocks are transferred to the switch fabric transparently without any processing in the ingress UNI/NNI interface blocks. Fig. 9 shows the block diagram when the UNI/NNI interface block operates as the egress for the ABR service.

Cells received from the source via the switch fabric are queued in the ABR buffer.

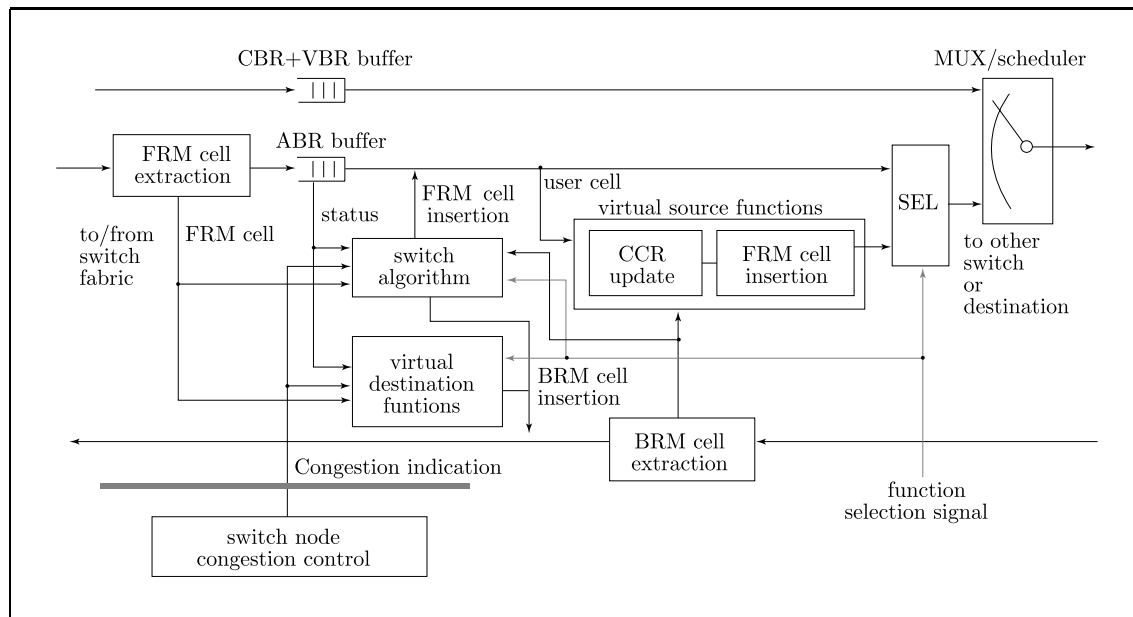


Fig. 9. Block diagram of the UNI/NNI interface for the ABR service.

The ABR buffer usually has some thresholds for easy congestion control. If the switch node does not operate as the segment end point, only the switch algorithm works. In case of the EFCI switch algorithm, the EFCI bit of the forward RM (FRM) or backward RM (BRM) cell is set or reset using the state of the ABR buffer or the congestion indication information offered by the switch node congestion control block. In case of the EPRCA, the MACR (Mean Allowed Cell Rate) value is recalculated and, if necessary, ECR (Explicit Cell Rate) in the FRM cell is updated and also the ECR value in the BRM cell is updated. The MUX/scheduler located just before the output link schedules the departing time of cells giving higher priority to the CBR or VBR cells and guaranteeing only the MCR (Minimum

Cell Rate) for the ABR cells. If the switch node operates as the segment end point, the switch algorithm is disabled. Instead, VS and VD functions work. The VD changes the FRM cell to the BRM cell and updates the EFCI bit using the ABR buffer state or the congestion information from the switch node congestion control block. The VS updates the CCR (Current Cell Rate) using the CI and NI (No-Increase) information and inserts the FRM cells. This stream is also scheduled with the CBR or VBR streams by the MUX/scheduler.

V. CONCLUSIONS

In this paper, we summarized requirements for the ATM interface block and presented the

UNI/NNI architecture to meet these requirements. We compared and evaluated some alternatives for the multiplexing structure. We presented our approaches in implementation of the UPC function and the shaper. By processing decision parts in the VSA in parallel, it takes only 850 ns to police an incoming cell when we use a 23 Mb/s master clock. In case of our shaper, more than 99 % of departing cells conform to the negotiated peak interval and there is no cell which departs the shaper faster than the negotiated peak cell rate. Therefore, we can establish a clear bandwidth allocation strategy and thus expect more efficient bandwidth utilization. We also proposed an interface architecture for accommodation of the ABR transfer capability. The proposed structure that performs virtual source and virtual destination functions as well as a switch algorithm can efficiently accommodate the ABR service.

Further study is required to evaluate the performance of the architecture and to find optimal parameter values for the ABR service. When we consider the new implementation issues such as the ABR transfer capability, cost-effective implementation of the UNI/NNI interface block becomes the key to realization of the economical ATM switching system. Moreover, the VLSI design and processing technology is critical to implementation of the reliable switching system. Further study is also required in the areas such as the development of the ABR switch algorithm and the realization of the protection switching mechanism.

REFERENCES

- [1] ITU-T draft recommendation I.371, "Traffic control and congestion control in B-ISDN," 1994.
- [2] ATM Forum, "ATM User-Network Interface Specification," Version 3.1, July 1994.
- [3] F. Guillemin and A. Dupuis, "A basic requirements for the policing function in ATM networks," *Computer Networks and ISDN Systems*, vol. 24, pp. 311-320, 1992.
- [4] ITU-T SG13, TD73(P), "Report of working party 2/13," July 1995.
- [5] ITU-T SG13, TD PL/11-97, "Draft new ITU-T recommendation Q.2210," April 1995.
- [6] ITU-T recommendation G.782, "Types and general characteristics of SDH multiplexing equipment," 1992.
- [7] P. Boyer, M. Sevel, and F. Guillemin, "The spacer-controller: an efficient UPC/NPC for ATM networks," *Proceedings of ISS '92*, vol. 2, pp. 316-320, 1992.
- [8] H. S. Park and J. K. Kim, "Policing and spacing of multiplexed ATM traffic," *IEICE Trans. Commun.*, vol. E77-B, no. 9, pp. 1122-1132, Sept. 1994.
- [9] R. Jain, "Congestion control and traffic management in ATM networks: recent advances and a survey," to appear in *Computer Networks and ISDN Systems*.
- [10] K. M. Park, S. Y. Kang, and C. M. Han, "Performance evaluation of a cell reassembly mechanism with individual buffering in an ATM switching system," *ETRI Journal*, vol. 17, no. 1, pp. 23-36, Apr. 1995.

Hong-Shik Park received the B.S. degree from Seoul National University, Seoul, Korea in 1977, the M.S. and Ph.D. degree from KAIST, Taejeon, Korea in 1986, 1995, respectively, all in electrical engineering. He joined ETRI in 1977, where he was involved in developing the TDX-1, 1A, and TDX-10 switching systems. Since 1990, he has participated in the ATM switching system development project and he is now a Head of the ATM Access Section. His research interests are ATM traffic control, ATM protocols, and performance evaluation of ATM systems. Dr. Park is a member of the KITE, KICS of Korea and the IEICE of Japan.

Yool Kwon graduated from Sogang University with the B.S. degree in electronic engineering in 1978 and received the M.S. and Ph.D. degrees from Pusan National University in 1985, 1993, respectively. He had worked with Agency for Defence Development as a researcher and then with Dong Eui Technical College as an instructor. He joined ETRI in 1993 and since then participated in developing an ATM switching system. He is interested in hardware design of communication system and traffic control using neural networks.

Young-Sup Kim received the B.S. and M.S. degree from Hong-Ik University, Seoul, Korea in 1984, 1988, respectively, all in electronics engineering. He joined ETRI in 1987, where he was involved in developing the TDX-10 switching system. Since 1992, he has been engaged in development of an ATM switching system and he is now a senior member of engineering staff of ATM Access Section. His

research interests are ATM protocols, and performance evaluation of ATM systems.

Seok-Youl Kang received the B.S. degree in electronics engineering from Pusan National University and the M.S. degree in computer science from KAIST. He had extensive experience in computer design and EPABX at Samsung Electronics Co. from 1976 to 1979. He joined ETRI in 1980 for the development of a large scale ESS called TDX-1 and TDX-10 as a Head of Software Development Section. Now he is engaged in developing an ATM switching system as a Director of ATM Technology Department. He is currently interested in operating system, software engineering, ATM switching technology, and optical switching.