A Multicasting Solution for ATM Video Applications^{*}

Jer-Min Tsai, Hsin-Hsiung Fang, Chi-Cheng Ju and Chen-Yi Lee

Dept. of Electronics Engineering, National Chiao Tung University,

1001, University Road, Hsinchu 300, Taiwan, ROC China

Tel: 886-03-5712121 ext. 54238 E-mail: tjm@royals.ee.nctu.edu.tw

Abstract: This paper presents a multicasting solution for shared-buffer ATM switch. The cell duplicating function is performed by a one-to-many modified demux circuit. A shared multicast server is used to translate input multicast cells to find destination ports and corresponding routing information. By using link-list based ring structure, the multicast server can provide high speed (5ns for search entry in 256-entry block in 0.8 μ m CMOS process) and cascable multicast translating function. In addition, the channel complexity of multipoint-to-multipoint multicast applications can be reduced to N channels per N users. The shared-buffer controller is modified by adding controlling and scheduling functions for multicasting queues to process the multicast feature.

1. Introduction

Broadband integrated services digital networks (B-ISDN) are expected to provide efficient and reliable transport for various services. The asynchronous transfer mode (ATM) technique [1] has been standardized and widely accepted as a basis for transporting and switching information in B-ISDN. However the applications that move the needs toward B-ISDN are not the traditional point-to-point telecommunication or data-communication applications, but the applications which need a lot of bandwidth and have the inherent variable-bit-rate characteristic such as video conference and video-on-demand.

There are three major video application types, namely point-to-point (P2P), point to multipoint (P2M), and multipoint-to-multipoint (M2M). Video applications such as video phone belong to the P2P type. Applications like video-on-demand belong to the P2M type. And applications like video conference belong to the M2M type. The challenges of P2M and M2M are not only how to provide high quality video signals through lower bandwidth communication channel, but also how to reduce the loading of video server and network.

In original ATM environment, an N users' video conference needs $N \times N$ communication channels to broadcast video signals to every user. The network traffic and video source loading can be reduced with the aid of efficient multicasting function. In addition, The multicasting circuit must translate M2M video cells fast enough to prevent the serious decay of video quality due to video cell delay.

Several multicasting solutions can be found in the literature [2-9]. The major architectures can be classified into the following types, namely copy network [6][7][9], recycle output cell [4][5], cell copy [8], and address copy [2][3]. The major topics which are discussed in those papers are cell duplication, cell routing, and performance analysis. Several excellent solutions on these topics can also be found in the above-mentioned papers. But the detailed descriptions on the mechanisms of multicast cell translation and M2M multicast function can not easily be found. The traditional N^2 channel complexity of M2M multicast applications will result in heavy loading on switch and

client system. In order to solve this problem, we propose a novel multicast solution based on shared-buffer ATM switch to reduce the channel complexity of M2M multicast applications to N connections for N users. In addition, the proposed solution can provide an efficient, flexible, and cascable M2M multicasting solution for video applications.

This paper is outlined as follows. In section 2, an overview of the shared-buffer ATM switch system will be described. Section 3 will present our algorithm for the support of multicasting function. In addition, a cost-effective VLSI architecture is also proposed to realize the algorithm. Discussion and comparison will be given in section 4.

2. System Concept and Overview of Shared-Buffer ATM switch system

Among the ATM systems found in the literature, shared-buffer ATM system [10] is attractive for its high memory utilization. From the viewpoint of "cost-effectiveness" of ATM system, it is therefore that the shared-buffer architecture is selected as the target architecture to realize our multicasting solution.

Fig. 1: Data flow diagram of the proposed multicasting shared-buffer ATM switch system.

Our shared-buffer ATM switch system for video applications mainly consists of 3 major parts, namely header translator (HT), shared buffer memory (SBM) and multicasting server (MCS) as shown in Fig. 1. HT is used to decode the VPI/VCI of each input ATM cell and find corresponding new VPI/VCI or multicast group ID and other routing information for each input ATM cell. From the routing information found by HT, SBM and its control circuit route input cells to individual output queue, where these cells are stored in the shared buffer which are waiting to be read out and sent to the destination output port. The multicast cells are translated by multicasting server. According to the group ID of each multicast cell, the parties of input cell can be found out by MCS to deliver this multicast cell. Data flow of the shared-buffer ATM switch system can also be found in Fig. 1.

3. Multicasting Hardware Architecture

3.1 Brief Description of The Multicasting Algorithm

Two procedures are needed to distribute multicast ATM cells. First, multicast cells must be translated to find their corresponding destination port and routing information. The second procedure is to copy multicast cells to corresponding output ports.

The translation circuits of multicast can be separated into three

^{*} WORK SUPPORTED BY THE NATIONAL SCIENCE COUNCIL OF TAIWAN, ROC UNDER GRANT NSC 86-2221-E-009-020.

In order to provide a flexible and efficient M2M cell translation function, we use **ring** structure to represent each multicast group. By linking all members in the same multicast group as a ring, no matter starting from any member, all members can be found by tracing the links one by one. The **ring** structure used to represent multicast groups is shown in Fig. 2. For the given example, a multicast cell is inputted from port A and then B can be traced, followed by C and D through the links. When the trace is back to A through the pointer from D, the member look up procedure is finished. Because of this ring structure feature, the N users M2M applications only need N connections instead of N^2 connections. This feature greatly reduces the cost and traffic load on both switches and end users. Thus it can be seen that the ring structure is very suitable to implement the M2M multicast translation function.

The relation among SBM, SBM controller, and MCS is shown in Fig. 3. When the SBM controller receives multicast cell, the multicast cell will be enqueued to a certain multicast queue. If MCS is free, the controller will send the headers of cells in multicast queues to MCS. The MCS extracts the *Group IDs* and *Source Port* data of input headers to translate multicast cells and find corresponding routing information and destination ports. In addition, the proposed MCS architecture can provide a group based translation. In other words, no matter the input cell comes from which source port, the MCS will find out the rest members of the same group to form the bitmap of destination ports. This feature makes the proposed architecture very suitable to solve the M2M multicast request.

Figure 1 consists of two diagrams. Diagram (a) is labeled '(a) Original DMX' and shows a single input line labeled 'From SRF' entering a triangular block. From the right side of the triangle, multiple output lines emerge, labeled 'out0', 'out1', 'out2', and 'outN-1'. Diagram (b) is labeled '(b) Multicast DMX' and shows a single input line labeled 'From SRF' entering a larger rectangular block. Inside this block, there are multiple parallel processing paths. Each path starts with an input line that goes through an 'en0' block and then an 'en1' block. The outputs of these paths are labeled 'out0', 'out1', and 'outN-1'.

The original DMX of shared buffer ATM switch is used to distribute the output cells of SBM to each output port. In order to provide a cost-effective cell duplicated circuit, we modify the DMX architecture to a multicast DMX. Fig. 4 shows the block diagram of those DMXs. Cell duplication can be achieved by setting the control registers (en0-en31) using the translation result from MCS. The unicast ATM cells can be routed by setting only corresponding control register to 1 and the rest to 0.

3.4 Multicast Server

Multicast server is used to translate input multicast cells to find destination port bit-map and corresponding VPI/VCIs. As the description in section 3.1, we use ring structure to construct an efficient M2M cell translation mechanism. In real implementation, we use the link-list data structure to construct the ring structure. In the meantime we can map the ring structure to a table with an index field, and the traced function of the multicast ring to table search function as well.

Fig. 5: Block diagram of the multicast server dedicated to handle multicast transmission.

Block diagram of the proposed multicast server is shown in Fig. 5. The multicast table is used to store and search the routing information of multicast cells and the translated buffer is used to store the translated data (bit-map of destination port and corresponding VPI/VCIs). The actual data format of multicast table is given in Table 1. The data fields of multicast table include: group ID, source port ID, next port ID, and new VPI/VCI. Group ID (GID) represents multicast connection ID to specify a certain multicast virtual connection to which the input multicast cell belongs. Source port ID (PID) represents which input port this multicast cell arrives. Next port ID is used to link current entry to next entry to be searched in the same group.

Table 1: Multicast table for a group of members.

	Key field		Link field	Routing Information
Member	Group ID (GID)	Source port (PID)	Next port	Information
A	N	3	1	New VPI/VCI
B	N	1	4	
C	N	4	2	
D	N	2	3	

Both Group and Source port ID are used as search key to locate the data entry of member specified. As the entry being searched is located, the data in the Next port field will replace current Source port ID which is used as a part of searching key for next table search. The Next port and new VPI/VCI in the information field are the output of table look up operation. The data in Next port field points to the output port which the multicast cell will be routed. And the data in

new VPI/VCI represents the corresponding new VPI/VCI for that output port.

In order to minimize the translation delay and to enhance flexibility, the whole multicast table is divided into several cascable table blocks. Each table block is designed by using Content Addressable Memory (CAM). The table block is composed of many table entries as shown in Fig. 6. The table search is performed by all blocks of whole table simultaneously. The table search of these blocks is synchronized by cascading control signals. The table search and table setup inside a table block is controlled by a local block controller. Block diagram of this table block is shown in Fig. 7.

Fig. 6: Block diagram of the table entry which is based on CAM.

Fig. 7: Block diagram of table block.

The usage of table block is monitored by an entry counter. This counter has two functions, the first is to provide the data pattern for CAM table initialization before any table operation. The second is to monitor the usage condition of table block, where the counter value indicates the number of entries been used. When this counter reaches the table size, the full flag of this block will be enabled to start the operation of next table block. Because this multicast table is designed using CAM to match the input ID pattern, all the table entries must be initialized before it is operational. During the initialization procedure, the Group ID field of table entries are written with counter values, from 0 to N-1, where N denotes the number of entries in a single table block. The entry located at the first physical location is written with 0, and the last one is written with N-1. The value of counter points to the location that could be used to add new entry. When a new entry is added to the table, all blocks are searched with the value pattern of its own counter and active bit is equal to 0. Then, the new entry is inserted into the matched location of the block which is enabled by cascading function. To delete an entry, all blocks are searched with the entry pattern whose active bit equals to 1. The content of the matched location is replaced by (counter -1) and then content of the counter is decremented by 1.

During the operation process, many repeated add and delete entry commands are requested. Then the counter patterns in the unused locations are generally distributed. These table entries can be managed efficiently with the aid of this counter. Moreover with the aid of the counter, cascading multiple table blocks can easily be achieved. This is illustrated Fig. 8. The insertion function of a certain table block is enabled when **full_in** signal has been set. When this

block is full and the **full_in** signal is set, the controller will set the **full_out** signal to enable the insertion function of next table block. The insertion function of a certain table block can only be activated when its previous table blocks are in full status.

Fig. 8: Block diagram of the cascading table blocks.

By using link-list structure, each entry of multicast table can be used by any multicast group. Thus the external fragment of storage can be eliminated. Coupled with the centralized translation method, the hardware utilization can be greatly reduced.

4. Discussion and Comparison

From above discussion, we can see the proposed multicast architecture provides a cost-effective and cascable multicast solution with M2M ability for shared-buffer ATM switch. The channel complexity of M2M is now reduced to N connections per N users through the aid of MCS.

By coupling the cell duplicating circuit into the DMX of shared-buffer ATM switch, the hardware cost can greatly be reduced and the performance of whole system remains unchanged. Table 2 shows the comparison among different cell duplicating schemes. From this comparison we can see our approach provides a more cost-effective cell duplicating scheme compared to available solutions under the same cell loss rate.

Table 2: The comparison among cell duplicating schemes.

	Copy network	Recycle cell	Cell copy	Address copy	Our approach
Extra hardware cost	A copy network (High)	several registers (Low)	no	extra address memory (High)	A one-to-many switch (Low)
Cell delay	Low	High	Low	Low	Low
Increase Cell loss rate	No	No	Yes	No	No
Flexibility and expandability	Depend on network type	Low	High	Depend on the controller	High

Cell translation is the most important part to provide powerful multicast function such as M2M multicast. But only a few papers [2][3] discuss about that. By using link-list based ring structure, our approach provides a high speed, cascable, and M2M multicast cell translation function. The translation delay can be reduced by exploiting parallel searching of CAM-based multicast table entries. Based on HSPICE simulation, we find the searching delay is about 5ns when table block size is equal to 256 in 0.8μm CMOS process. Indeed, it is expected that the searching speed can be enhanced by using smaller table block size. Because the searching speed of MCS is

fast enough, the complexity of multicast cell scheduling circuit in SBM controller can be reduced and hence the translation delay variance will not become a problem.

5. Conclusion

A Multicasting solution based on shared-buffer ATM switch for video applications has been proposed in this paper. The MCS, DMX embedded cell duplicating circuit, and modification on SBM controller are taken into account simultaneously to add the M2M multicast feature for the target shared-buffer ATM switch. The capability of this high speed and powerful MCS not only reduces the complexity of SBM controller but also provides an efficient M2M cell translation function. In addition, the channel complexity of our solution can be reduced to N connections, instead of N² connections, per N users. Finally the multicast DMX provides an efficient cell duplicating solution and the multi QoS multicast SBM controller provides more suitable multicast features for various video applications.

From the above discussion, we can see the proposed multicast solution can greatly reduce the connection complexity for M2M. It will not only reduce the hardware cost of switch itself, but also improve the capacity requirement of whole switch system and clients.

Acknowledgement: The authors would like to thank their colleagues within the SI2 group of NCTU for many fruitful discussions, especially Mr. K.W. Liu, Mr. C.F. Lee, and Mr. Y.S. Lee. The MPC support from NSC/CIC is also acknowledged.

Reference:

- [1] CCITT SG XV III Draft Recommendation I.150 : "B-ISDN ATM functional characteristics", Geneva, May 1990.
- [2] Tsern-Huei Lee, Shun-Jee Liu, "Multicasting in a shared buffer memory switch", IEEE TENCON'93/Beijing, vol.1, p.p. 209-212.
- [3] H. Saito, H. Yamanaka, H. Yamada, M. Tuzuki, H.Kondoh Y. Matsuda, K. Oshima, "Multicasting Function and its Implementation in a Shared Multibuffer ATM Switch", IEEE INFOCOM'94, vol.1, p315-315.
- [4] Xiaoqiang Chen and Vijay Kumar, "Multicast Routing in Self-routing Multistage Networks", IEEE INFOCOM'94, vol. 1, pp.306-314.
- [5] R. Cusani, F. Sestini, " A Recursive Multistage Structure for Multicast ATM Switching", IEEE INFOCOM, Vol. 3, April 1991, p.p. 1289-1295.
- [6] Jonathan Chao and Byeong-Seog Choe, "Design and Analysis of a Large-Scale Multicast Output Buffered ATM Switch", IEEE/ACM Trans. Networking, vol. 3, NO. 2, April 1995, pp. 126-138.
- [7] Chin-Tau Lea, "A Multicast Broadband Packet Switch", IEEE Transactions on communications, vol. 41, NO. 4, April 1993, pp. 621-630
- [8] Xing Chen, Jeremiah F. Hayes, Mustafa K. Mehmet-Ali, "Performance Comparison of Two Input Access Methods for a Multicast Switch", IEEE Trans. on Communications, Vol. 42, No.5, May 1994, p.p. 2174-2178.
- [9] T. T. Lee, R. Boorstyn, E. Arthurs, "The Architecture of a Multicast Broadband Packet Switch", IEEE INFOCOM, March 1988, p.p. 1-8.
- [10] Takahiko Kozaki, "32×32 Shared Buffer Type ATM Switch VLSI's for B-ISDN's", IEEE Journal on selected areas in communications VOL. 9, NO. 8, OCT. 1991, pp. 1239-1247.