Implementation of OAM functions in a 622Mbps ATM Layer ASIC

Chan Kim*, Sang-Ho Lee, Jae-Geun Kim

Switching and Transmission Labs. ETRI,
161 Kajeong-dong Yuseong-goo Taejon, 305-350 Korea, email : ckim@etri.re.kr

Abstract

This paper describes the design and implementation of OAM function in a 622Mbps ATM layer ASIC called ASAH-L4. ASAH-L4 is a 622Mbps ATM layer processing ASIC which performs header translation, routing, real time OAM, QOS buffering and ABR processing for both ingress and egress traffics. The implemented OAM functions contain fault management and performance management for all connections. The hierarchical OAM principle is fully supported via the table lookup method and table contents that are associated by the OAM implementation.

1. Overview of ASAH-L4 ASIC

ASAH-L4 is a 622Mbps ATM layer processing ASIC which performs header translation, routing, real time OAM, QOS buffering and ABR processing for both ingress and egress traffics. It has UTOPIA 2 interface for maximum 31 physical ports and receive ready type switch interface. All the connection information and cell data are stored in local SRAM physically separated for receive VC table, transmit VC table and the receive and transmit cell buffer memory through three independent SRAM interfaces. Each processing block of the ASIC accesses the local SRAM through memory arbiter and interfaces. This paper describes the design and implementation of the OAM functions of ASAH-L4 ASIC.

2. OAM Functions in ASAH-L4 ASIC

2.1 Hierarchical OAM Principles

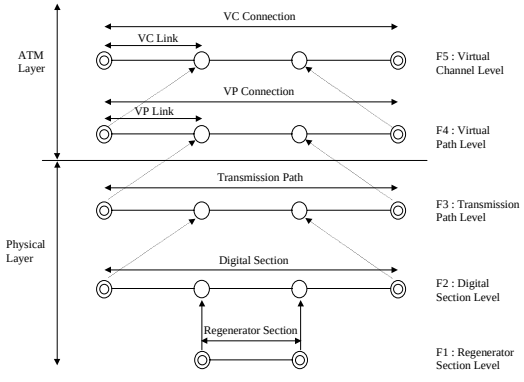

Fig. 1. Hierarchical OAM Architecture

Fig. 1 shows OAM principles of ITU-T I.610 Recommendation. ASAH-L4 fully supports this hierarchical OAM in its table setup capability and other OAM implementation. It also enables the user to arbitrarily set a point as a connection end point or segment end point as appropriate.

2.2 Connection Setup

ASAH-L4 supports both VC and VP switching. In VC switching, the header translation and routing is performed based both on VPI and VCI while in VP switching, only the VPI has meaning and is translated and used for routing. Fig. 2 shows how VPI/VCI are used in VC and VP switching.

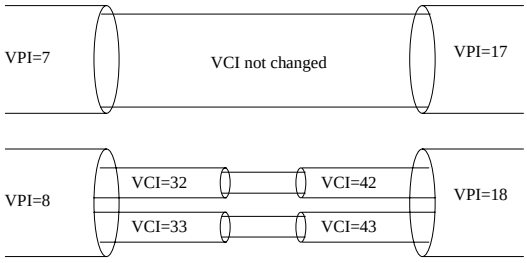

Table
7 / x => 17 / x
8 / 32 => 18 / 42
8 / 33 => 18 / 43

Fig. 2. Connection Diagram for VP and VC Connection

There is an important notion in that at a VC connection point, the VPC the VC connection belongs to should be terminated at both ends of the system. This means that we should be able to manage the VPC table and VCC table for a single VPI value (Consider the case when a VP-AIS is received, VP-RDI should be sent back and many VC-AIS should be forwarded). This system-wide function of fig.2 is implemented using several ASAH-L4 chips for switch ports as shown in Fig. 3. As can be seen, the VPI and VCI value is given meaning only associated with the physical link. Also we see in the figure that to have 8/32 VC connection, we should also have the 8/x VP connection setup which are terminated.

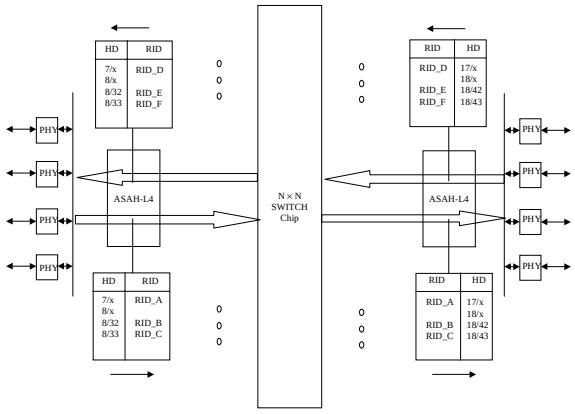

Fig 3. Look-up Table Setup in an ATM System using ASAH-L4 chips

The table lookup mechanism was devised to make it possible to setup a VP connection table with a VPI and many VC connection using the same VPI at the same time while meeting the lookup speed and memory usage requirements. If a received ATM cell is detected as a predefined VP layer cell(like VCI 3,4 and 6), the VP connection table is directly looked up via the dedicated region of the base table allocated for this cases, and if it is not, the normal base table region is read to determine if it belongs to VP connection or VC connection. If it belongs to VP connection, the associated address is that of the VP connection table and if it is not, the address is start address of the hash table for the case. Then the hash table is read with the hash index made from the VCI value. The hash table directs the hash chain of the connections having the same port id and VPI values and VCI hashing index. Fig. 4 shows this lookup method of ASAH-L4.

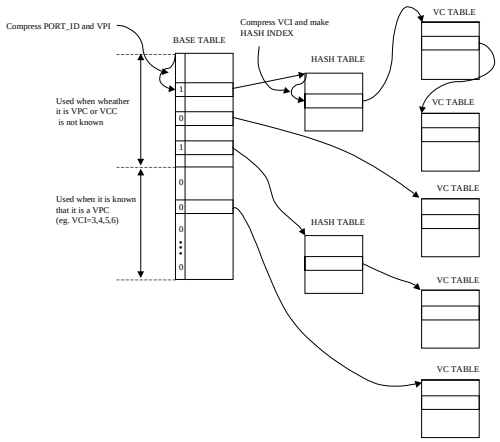

Fig. 4. Table Look-up Mechanism in ASAH-L4 receiver

2.3. Fault Management

All the alarm state changes are detected and reported to the CPU interface through the status queue and interrupt process for all connections. The OAM fault management is supported for VP or VC layer and for segment end and connection end points. So the OAM cell's layer and point definition should match to those of the connections looked up for the OAM cell to be processed.

The OAM fault events processed for each connection at ASAH-L4's receiver side is as follows.

- AIS detection : when an AIS cell was received when not in AIS state.
- RDI detection : when a RDI cell was received when not in RDI state
- CC alarm detection : when CC cell or user cell was not received for 3 seconds when CC enabled
- AIS release : when AIS cell was not received for 3 seconds or a user cell was received when in AIS state
- RDI release : when RDI cell was not received for 3 seconds when in RDI state
- CC alarm release : when CC cell or user cell was received when CC enabled
- Undefined OAM cell : when undefined OAM cell was received

Besides, any AIS or RDI cell can optionally be extracted after normal alarm processing is done to CPU for further S/W processing without begin discarded.

Related to Fault OAM, ASAH-L4 also performs

- OAM loopback cell processing(insert, loopback, drop, pass)
- Activation / Deactivation OAM cell sending / extracting
- System Management OAM cell's sending / extracting

These are also governed by the layer and point definition of the cell and connection.

2.4 Performance Management

2.4.1 PM Destination Function

When a user cell is received for a PM enabled connection, the TRCC(total received cell count) is incremented for every received user cell and the value is updated in the VC table. And when a Forward Monitoring PM cell is received, the TUC values for CLP0+1 and CLP0 are extracted and saved in the VC table and the TRCC value for CLP0+1 and CLP0 is also saved in the VC table. At the same time, the difference of the newly received TUC value and the last TUC value is calculated and we call this value B0+1 and B0 respectively for CLP. This B0+1 and B0 values mean the number of transmitted user cells between the two Forward Monitoring PM cells seen from the sending end system. Also, every time a Forward Monitoring PM cell is received, the difference of the TRCC value at the moment and the TRCC value save when last Forward Monitoring PM cell was received is calculated and we call this A0+1 and A0 respectively for CLP. This A0+1 and A0 mean the number of actually

received user cells between the two Forward Monitoring PM cells seen from the receiving system. Then, also during the PM cell reception, the difference of A0+1 and B0+1, and the difference of A0 and B0 are calculated to derive lost or mis-inserted cell counts.
For BIP violation calculation, the BIP value is calculated for all user cell's payloads and updated in the VC table. And when an FMPM cell is received, the BEDC value is extracted and compared to the actually calculated BIP value for the received user cell during the PM block. If any difference exists, the number of different bits is accumulated in the BIP violation count in the VC table. The BIP accumulation restarts every time a FMPM cell is received and the BIP violation calculation is disabled when there is a PM SN error or lost or mis-inserted cell count detected.
Every time a Forward Monitoring PM cell is received, the PM cell is sent back as a backward reporting PM cell. In the Backward Reporting PM cell, the TUC values are carried intact and the TRCC values stored in the VC table is copied to the tail part of the payload for CLP0+1 and CLP0. Again, the BIP violation count is copied only when there was no SN error or cell count error. When not copied, the BIPV is marked as 0.
2.4.2 PM Source Function
When PM is enabled for a connection in the transmitter, Forward Monitoring PM cell is inserted in the user cell stream every PM cell block. The TUC(Total User Cell Count) value which were incremented for user cells and kept in the VC table up to that point is then carried to the Forward Monitoring PM cell. Also, the BEDC value being updated for the cell payloads of the cell block is carried out on the PM cell and the BEDC is reset and calculated for the next PM block.
The PM cells sent in the transmitter would be delivered to the receiver as Backward Reporting PM cell. Every time a Backward Reporting PM cell is received, as in the Forward Monitoring PM cell reception case, the TUC values and the TRCC values are extracted and stored in the VC table. Also, at the same time, the difference of the newly received and last received TUC and TRCC values are calculated respectively and the difference of the differences is calculated to know out the lost or mis-inserted cell counts in the far end receiver system. The BIPV values are just extracted and accumulated.

3. OAM Implementation in ASAH-L4 ASIC

3.1 Architecture of ASAH-L4 ASIC

Figure 5 shows the architecture of ASAH-L4 ASIC. The function of the whole chip will be briefly explained here. In the receive direction, UTOPIA_RX reads cells from a port and looks up the connection table's address while receiving the cell. The VTSB_RX reads the contents of the VC table necessary to perform the enabled function for the connection. Then as soon as the connection data is ready, with the cell still being written or already written out into the UTOPIA_RX 's internal FIFO, the UPC and OAM_RX begins action for the cell. After the UPC and OAM action is finished, the QBWB_RX, based on information from all the blocks, reads and routes the cell to the appropriate block or discards it. The QBWB_RX performs cell routing such as looping, CPU extraction, passing to the switch(in non QOS mode), or writing to the QOS buffer(in QOS mode) which is implemented using linked list method on the external SRAM. The QBWB_RX also does all the chores for the queue management and scheduling. In QOS mode, when a queue is selected for service, the QBWB_RX commands its service and the QBRB_RX reads the cell from the QOS buffer and sends it to the switch interface through the RX_MUX. In the mean time, header is translated in the QBWB_RX and routing tag is attached in the QBRB_RX. All the processing are pipelined to increase the throughput so that cell moving transaction is performed locally between blocks autonomously.

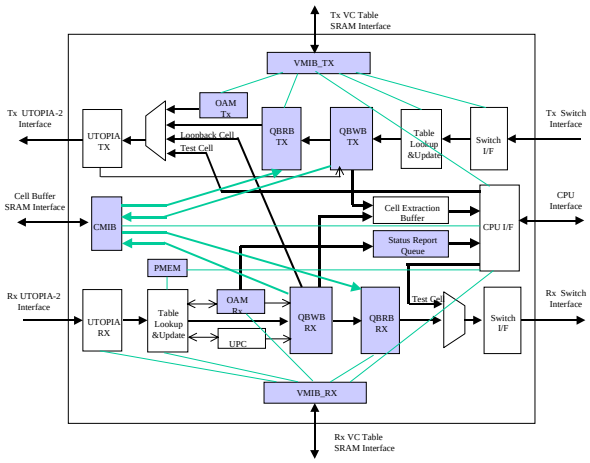

Fig 5. Architecture of ASAH-L4

In the transmit direction, the actions are similar to the receiver. The SWIB_RX receives cells and searches for the VC table address. The VTSB_RX loads and updates the connection data. When the connection data is ready, the cell is routed to appropriate block by the QBWB_TX as in the receiver. In QOS mode, the cells are stored in the external SRAM and read and sent by the QBRB_TX after scheduling. The UTOPIA_TX sends the cell to the destination port with multicast capability. In the next sections, the OAM implementation in the OAMB_RX, OAMB_TX and QBRB_TX will be explained.

3.2 OAMB_RX

OAMB_RX is largely divided into two, the FM block (in this section, the OAM and RX is omitted from the block name because it's obvious) which performs fault

management OAM and the PM block which performs the performance management OAM. The FM block is again divided into the CELL_FM block that runs every time a cell is received and the SEC_FM block that runs every second to scan all the VC table memory for OAM status monitoring. The CELL_FM and the PM blocks start by a signal coming from the VTSB_RX every time a set of data for a received cell in the UTOPIA_RX FIFO is ready regardless of whether the cell is OAM cell or not. For pipelining operation mentioned earlier, the CELL_FM and PM blocks latch all the data from the VC table and the data extracted from the cell by this start signal to process the OAM action related to the received cell. Because of this latching, the UTOPIA_RX and VTSB_RX can proceed to the beginning part of the next cell processing.

The FM part detects the alarm status conditions and reports the change through status queue and interrupts to the CPU block. The CELL_FM detects generation and release(by user cell) of AIS, generation of RDI, release of CC alarm and other OAM cell errors. The SEC_FM detects the release of AIS, release of RDI, and assertion of CC alarm by checking if the continuity counter in the VC table has reached zero. The continuity counter is set to 3 every time a cell implying the continuity of a state is received (for example a AIS or RDI cell in those states and user cell or CC cell for a CC enabled connection) and decremented every time the SEC_FM visits the table.

When there's a status change, it is updated in the table and reported to the CPU through status queue and interrupt.

In the CELL_FM, when an OAM loopback cell is received, it orders the looping or extraction of the cell judging from the loopback location id, the state of loopback mode in the receiver to the QBWB_RX. If the cell should be looped back, the decrement of the loopback indication and CRC10 attachment is done at the QBWB_RX during actual loop back passing. Fig. 6 shows the block diagram of the OAMB_RX.

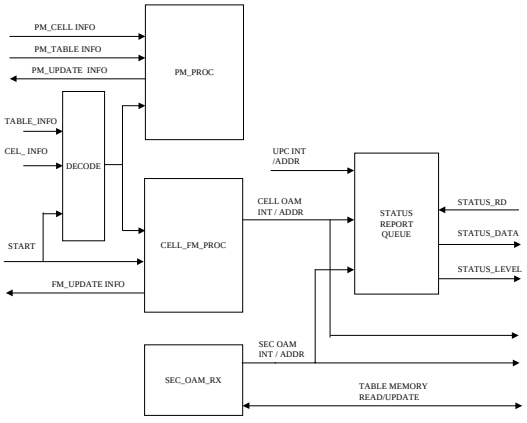

Fig. 6. Block Diagram of OAMB_RX

In the PM block, if PM is enabled for a connection, the TRCC and the BIP value is updated for every user cell received. Every time a FMON PM cell is received, the extracted TUC values from the FMON PM cell and counted TRCC values at the moment are saved in the VC table and the difference of the cell count calculated from delivered TUC value and actual TRCC is calculated to derive the number of lost or mis-inserted cells. The calculation is similar for the BREP PM cell reception, except that the TUC and TRCC values extracted from the BREP PM cell is used in the cell count error. For BIP error calculation, when the FMON PM cell is received, the BEDC on the PM cell is compared to the BIP value accumulated for the received user cells during the PM block to count the errors. For BREP PM cell, the BIPV value is just extracted and accumulated in the table. For all the cases, both for FMON and BREP PM cells, when the cell count error and BIP violation error gets over the threshold, the PM cell block is marked as severely errored block and the SECB count is incremented for forward and backward traffic in the VC table. Unlike other connection data, the PM data in the receiver is located in the internal DPRAM and pointed to by a pointer in the VC table for the sake of throughput(parallel loading). Table 1 shows the PM table format of the receiver VC table. Due to the DPRAM size, the number of concurrently PM enabled connections is limited to 32 in the receiver.

The FMON PM cell is, when enabled (default), looped back as a BREP PM cell. The OAMB_RX passes the related information needed to form the BREP PM cell to QBWB_RX with related routing information so that the cell is looped back.

Table 1. Receive PM Table

FMCSN(8)	BMCSN(8)	BIP16(16)	
TRCC 0+1		TRCC 0	
F_TRCC 0 + 1		F_TRCC 0	
F_TUC 0 + 1		F_TUC 0	
B_TRCC 0 + 1		B_TRCC 0	
B_TUC 0 + 1		B_TUC 0	
F_SECB(8)	F_LOST(8)	F_MISINS(8)	F_BIPV(8)
B_SECB(8)	B_LOST(8)	B_MISINS(8)	B_BIPV(8)

Sending BRPM cells and OAM loopback cells or extracting OAM loopback cells are actually done in QBWB_RX where header translation is occurring while cell is passed to destined location. The QBWB_RX analyzes all the data coming from the table, cell paylaoad, and OAM, UPC action and determines where to send the cell and how to change the cell payload if necessary. all All the data and information needed to change the payload data for BRPM or OAM loopback cells or even RM cells are provided by OAM_RX, VTSB_RX and UTOPIA_RX(extracted data or BIP calculation). Every time the payload is altered, CRC10 is newly calculated while transfer. Fig. 7 shows the block diagram of

QBWB_RX where each transfer block has internal FIFO to enhance the throughput capability using pipelining concept.

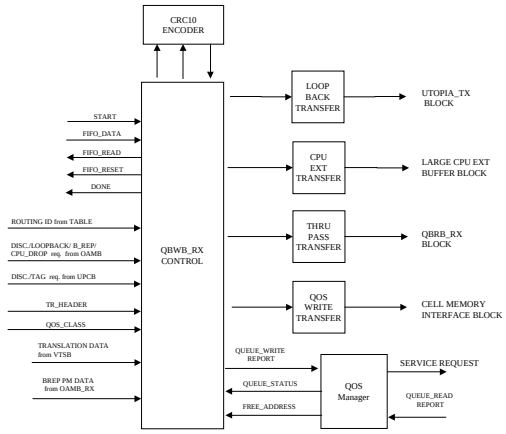

Fig. 7 block diagram of QBWB_RX

3.3 OAMB_TX and QBRB_TX

In Fig. 5, the transmit OAM implementation is divided into OAMB_TX and QBRB_TX. The OAMB_TX only performs the periodic transmission of alarm or CC cells. Every second, it scans all the transmit VC table and sends AIS, RDI, or CC cell as indicated in the table. The header and the PTI values are set by the definition of the connection – whether it is VC or VP connection and whether the OAM cell sent should be segment OAM cell or end-to-end OAM cell.

The insertion of the Forward Monitoring PM cell is performed by the QBRB_TX that reads cells from the QOS buffer and sends them to UTOPIA_TX through the MUX_TX. This is because the content of the FMON PM cell is closely related to the actual transmission of user cells enclosed by adjacent FMON PM cells. When PM is enabled for a connection, the QBRB_TX updates the TUC0+1, TUC0 and BEDC every time a user cell is transmitted and if the FMON PM cell should be inserted, the values at the moment are carried by the FMON PM cell with the CRC10 insertion. The PM block size is programmable and the PM insertion condition is decoded using the lower portions of the TUC counts.

Fig. 8 shows the block diagram of QBRB_TX where FMPM cell is generated in Cell Transfer block when it is time to do so. Periodic OAM cell transmission occurs in OAMB_TX(not shown in the figure)

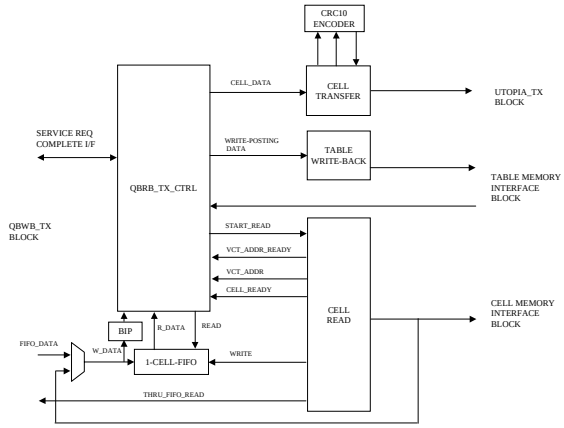

Fig. 8 Block diagram of QBRB_TX

4. CONCLUSION

In the work described in this paper, the full OAM capability was implemented in an ATM layer ASIC which incorporates UPC and QOS buffering as well as other basic routing and translation function. This ASAH-L4 ASIC provides this OAM capability by well-defined actions among blocks and the data organization on the external memory and internal dual port SRAM. The performance requirement is met and it is easy to use in a system. The chip, which is about to come out from fabrication at this time of writing, is believed to be a single chip solution for any ATM system with its full and perfect OAM capability and other functions.

This chip is one of the ATM chipset ASICs developed in ETRI.

References

1. ITU-T Recommendation I.610 "B-ISDN Operation and Maintenance Principles and Functions," Nov.95
2. ITU-T Rec. I.361. "B-ISDN ATM layer Specification"
3. Bellcore GR-1248-CORE "Generic Requirements for Operations of ATM Network Elements" Aug.94
4. PMC-Sierra, RCMP 800 Chip Data book.
5. Lucent Technology, ALM, ABM Chip Data book.
6. Chan Kim et al, "An Implementation of a 622Mbps ATM Physical Layer ASIC," ICT97, Melbourne, Apr. 1997.
7. Chan Kim et al, "Design and Implementation of a High-Speed ATM Host Interface Controller," ICOIN-12, Tokyo, Jan. 98
8. Chan Kim et al, "Design and Implementation of an ATM Segmentation Engine with PCI Interface," ISCAS'98, Monterey, May. 1998.
9. Chan Kim et al, "Implementation of a QOS buffering in ASAH-L4 ASIC with Weighted Round-Robin and Maximum Delay Threshold," ICACT'99, Muju, Feb. 1999.