

New Queuing Strategy for Large Scale ATM Switches

Mohsen Guizani
Computer Science Department
University of West Florida
mguizani@cs.uwf.edu

Ala I. Al-Fuqaha
Research and Development
Princeton Optical Networks, Inc.
gmpls@yahoo.com

Abstract

In this work, we study the different buffering techniques used in the literature to solve the contention problem in ATM switching architectures. The objective of our study is to determine the buffer requirements needed to achieve a given Quality of Service (e.g., a given cell loss probability). Based on this study, we propose a Combined Central and Output Queuing (CCOQ) technique to be used in designing large-scale ATM switches. Also, we propose a general design technique for an NxN large-scale ATM switch with a suitable CCOQ buffer size to reduce both the cell loss probability and the complexity of the memory modules. The switch has to be designed such that it can be implemented using the smallest number of VLSI chips possible. It should be also reliable for commercial use. The switch should support multicast and priority control functions.

1. Combined Central and Output Queuing (CCOQ)

For large-scale ATM switches (in terms of input and output ports), the amount of memory that needs to be installed in the switch is large, so reducing the size of the memory is of great importance.

In many cases, the amount of memory that needs to be installed in a large-scale ATM switch is large and it can not be integrated on a single chip either because of its size or because of the lower operating speed that it will have. In such cases and based on our simulation studies, we propose to implement the large-scale ATM switch out of smaller switches where a small amount of memory needs to be installed in each of these smaller switches. This technique will slightly increase the total amount of memory that will be used to implement the large-scale ATM switch. But, a number of other advantages of this approach can be listed as:

- The amount of memory that needs to be integrated on a single switch is small.
- The operating speed of these smaller memories installed in the small switches that make up the large-

scale ATM switch is more than that of one large memory.

- Using this approach, the design of the switch will be more modular. The number of output ports of the switch can be increased at any time by adding smaller switches inside the large-scale ATM switch.

2. Performance of CCOQ

Figure 1 shows different buffering configurations that can be used within a 16x16 ATM switch. These configurations include output buffering, shared buffering, and two other Combined Central and Output Queuing (CCOQ) configurations in which all the input ports and part of the output ports have access to the same memory module. Table 1 shows the number of memories, size of memory integrated on a single chip, and the total memory needed in all of these configurations. The table shows that in case of shared buffering, the amount of memory that needs to be integrated on a single chip is 600*424 bits. In case of output buffering, the table shows that the amount of memory that needs to be integrated on a single chip is 150*424 bits but 16 of these memories are needed.

Studying Table 1, we conclude that the two CCOQ configurations (with $M=2$ and $M=4$) are better than the output buffering configuration in terms of the total memory that needs to be installed in the switch. The table also shows that the two CCOQ configurations are better than the shared buffering configuration in terms of the amount of memory that needs to be integrated on a single chip.

Figure 2 shows that when the number of memories used in the switch (16x16 in this example) increases from one to two and then to four, the amount of memory that needs to be integrated on a single chip decreases dramatically. When the number of memories used in the switch is more than four, we only see a slight decrease in the buffer size that needs to be integrated on a single chip. Thus, using four memory modules for a 16x16 switch to achieve 10^{-5} cell loss probability is a good selection in this example, assuming that the most critical design condition is the amount of memory to be integrated on a single chip and the speed of this memory.

3. Proposed Architecture

This section presents a new fault-tolerant multicast 4X4 ATM switch. The switch is designed based on queuing strategies, multicasting considerations, and error detection for fault-tolerance implementation. It is based on a novel idea on realizing multi-casting. A small address memory will be used to hold a reference for the cell needs to be switched to several outputs.

Figure 3 depicts the proposed 4x4 ATM switching architecture. Each of the two switching modules (SM) in the switch receives all the four inputs. The memory controller module (MCM) of the SM then determines where to store the incoming cells if they are intended to one of the two output ports controlled by that SM. Otherwise, it discards the cell.

In the following sections, we describe the Memory module (MM) and the Memory Controller Module (MCM) that make up each SM.

3.1. Memory Module (MM)

The MM of each SM shown in Figure 3 should be accessed 6 times (4 times to write and 2 times to read) instead of 8 times in case of shared buffering architecture. If the memory unit available can not be accessed 6 times per cell slot, the MM can be designed using more than one memory unit.

Figure 4 shows the SM used in the 4x4 switch shown in Figure 3. This SM is designed with 3 memory units, each memory should be accessed 2 times per cell slot (one memory unit can be accessed for read and the other two units can be accessed for write).

Table 1 shows the simulation results of the buffer requirements to achieve a given cell loss probability for 16x16 switch under 0.9 loading conditions for different memory locations: $M = 16$ is output buffering and $M=1$ is shared buffer. For M between 1 and 16, we have Combined Central and Output Queuing (CCOQ) where M represents the number of memory modules shared by the outputs.

These results show that the buffer requirements for our buffering scheme ($M=4$) is more than that needed in case of shared buffering and less than that needed in case of output buffering to achieve a given cell loss probability.

Our motivation to use this buffering scheme, despite that it requires more memory, was to reduce the memory complexity by reducing the number of memory access needed. The other motivation is that, normally the memory that is needed for a large-scale switch is larger than that can be integrated on a single chip. So, by adopting our scheme, the memory is already partitioned and those partitions can be designed in such a way to be integrated on a single chip.

3.2. Memory Control Module (MCM)

Our architecture uses two types of Memory Controller Modules, mainly MCM and MCM-F. In the following, we describe in general the main functionality of each of those two modules:

3.2.1. MCM. This module compares the two bits of the error control field of the ATM cell with the two most significant bits of the destination field. If the bits do not match, then the MCM will discard the cell assuming that an error occurred in one switching element of a previous stage in the switching fabric. In case that the bits match, the MCM will write the incoming ATM cell to the location pointed to it by the write pointer. Figure 5 shows the traffic generated by a properly working switch [1,2].

The MCM keeps track of two bitmaps per output port as shown in Figure 6. When a cell arrives to the switch and it is destined to one or more output ports with a specific priority level, the MCM writes a "1" bit in the bitmap associated with the destination output port and the priority level needed.

During each cell period, the MCM checks the read pointers of bitmaps of all output ports and selects to route the cells based on First In First Out (FIFO) service scheme. The MCM also checks the bitmap of the higher priority level of each output port so that it services the higher priority cells first.

3.2.2. MCM-F. This module will be activated in case one or more MCM modules in the switch are not working. The main functionality of this module is to broadcast the incoming cells to all possible outgoing links while attaching to them the appropriate error flag. Figure 5 shows the traffic generated by a faulty switch [1,2].

The MCM-F keeps track of two bitmaps for the output port (one for each priority level). The memory of the switch is completely shared by all the input and output ports to reduce the cell loss probability over that can be obtained by using the same amount of memory that is not shared among all of the input and output ports.

For an $N \times N$ ATM switch, the amount of memory needed by the memory controller to support P -levels of priority is as follow [3]:

$Total\ Capacity = P * B * N^2 * \log_2(B * N)$, where B is the required buffer size per output port.

In our scheme, the total memory required by the memory controller is as follows:

$Total\ Capacity = P * B_s * N * (N/s)$, where B_s is the required buffer size for s ports sharing the same memory.

4. Verilog Simulation

The Verilog Hardware Description Language (HDL) was used to verify the functionality of the switch. Figure

7 shows some of the signals from the Verilog simulation program of the proposed architecture. These signals are taken from the main components of the switch, mainly the Memory units and the Memory Controller Management (MCM). These signals show the correctness of the functionality of the proposed switch.

5. Reliability Analysis of Proposed Switch

Almost all of the implementations of fault-tolerant MINs introduce redundancy in the network. Different type of redundancy can be used as explained previously. These solutions are expensive in terms of number of number of extra switches per stage, and/or the size of the switching elements. Moreover, these solutions have a high hardware complexity that need complex routing algorithms. In our design, the reliability of the switch was increased by adding an extra unit (MCM-F) within the switch.

In our proposed switch design approach, a large scale ATM switching fabric can be constructed by using multiple switching elements (e.g., the 4 x 4 switching element described above). If the number of failures is k , where $0 \leq k \leq n 2^{n-1}$, then the number of configurations in which these k failures can occur is given by:

$$\binom{nx2^{n-1}}{k}$$

The Survival function $Q(k)$ is given by:

$$Q(k) = \frac{\binom{nx2^{n-1}}{k}}{\binom{(n-1)2^{n-1}}{k}}$$

Assuming that the failure rate of the memory units, MCM modules, and MCM-F is the same (equals λ). The reliability of the proposed 4x4 switching element is given by (See Figure 8):

$$R_{Switch} = e^{-2\lambda} [1 - (1 - e^{-\lambda})(1 - e^{-2\lambda})]$$

6. Conclusion

In this work, we studied the three different buffering configurations used in the literature. This study inspired us to propose a new buffering technique that is based on Combined Central and Output Queuing (CCOQ). Using CCOQ, the total memory that needs to be installed in the switch to achieve a given cell loss probability is less than that required in the case of output buffering. Also, the total memory size will be more than that required in the case of shared queuing. But, the memory speed that is

required using CCOQ will be less than that required in the case of shared queuing. Analytical and simulation studies were presented to study the performance of the proposed buffering technique.

This work also presented a new Memory Control Module (MCM) to manage the memory modules installed in the switch. The proposed MCM has a small address memory that is used to hold the address of the cell that needs to be switched to its desired output(s). This leads to a better use of the available buffer space.

A 4x4 ATM switch was then proposed. The proposed switch uses the proposed CCOQ buffering technique. It also uses the proposed MCM to manage the memory modules installed in the switch. The proposed architecture was then simulated in Verilog HDL to show the correctness of the design. Finally, reliability analysis of the switch was performed to find the survivability of the switch.

7. Future Work

In the future, the CCOQ buffer requirements can be evaluated against the other buffering techniques under bursty and self-similar traffic conditions. Also, the throughput of the proposed CCOQ can be evaluated against the other buffering techniques.

Finally, the proposed MCM can be used to manage the memory module that is installed in shared buffering ATM switches. This will reduce the size of the address queues needed to manage the shared memory used in Shared buffering ATM switches.

8. References

- [1] M. Guizani and A. Memon, "SEROS: A Self Routing Optical ATM Switch," *International Journal of Communication Systems*, vol. 9, no. 2, March/April 1996, pp. 115-125.
- [2] J. W. Causey and H. S. Kim, "Comparison of buffer allocation schemes in ATM switches: Complete sharing, partial sharing, and dedicated allocation," *Proc. ICC'94*, vol. 2, May 1994, pp. 1164-1168.
- [3] H. Yamanaka, H. Satio, H. Kondoh, Y. Sasaki, H. Yamada, M. Tsuzuki, S. Nishio, H. Notani, A. Iwabu, M. Ishiwaki, S. Kohama, Y. Matsuda, and K. Oshima, "Scalable Shared-Buffering ATM Switch with a Versatile Searchable Queue," *IEEE J. Select. Areas in Commun.*, vol. 15, no. 5, 1997, pp. 773-784.

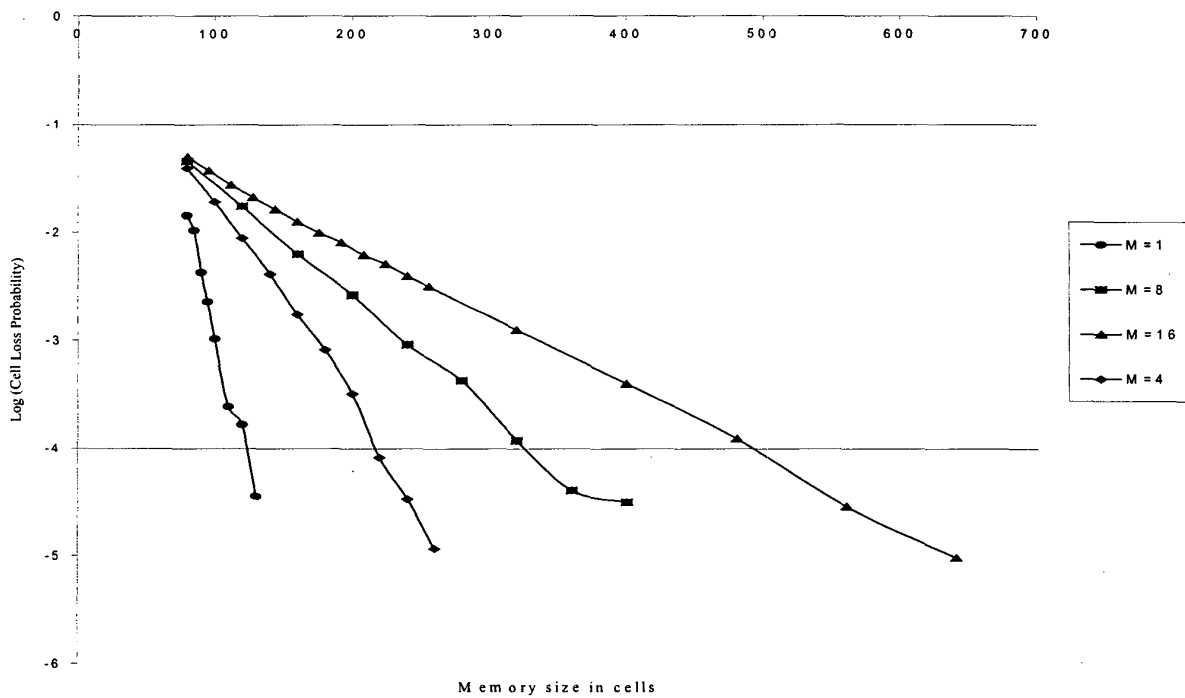


Figure 1. Log (Cell Loss Probability) Vs. Memory Size

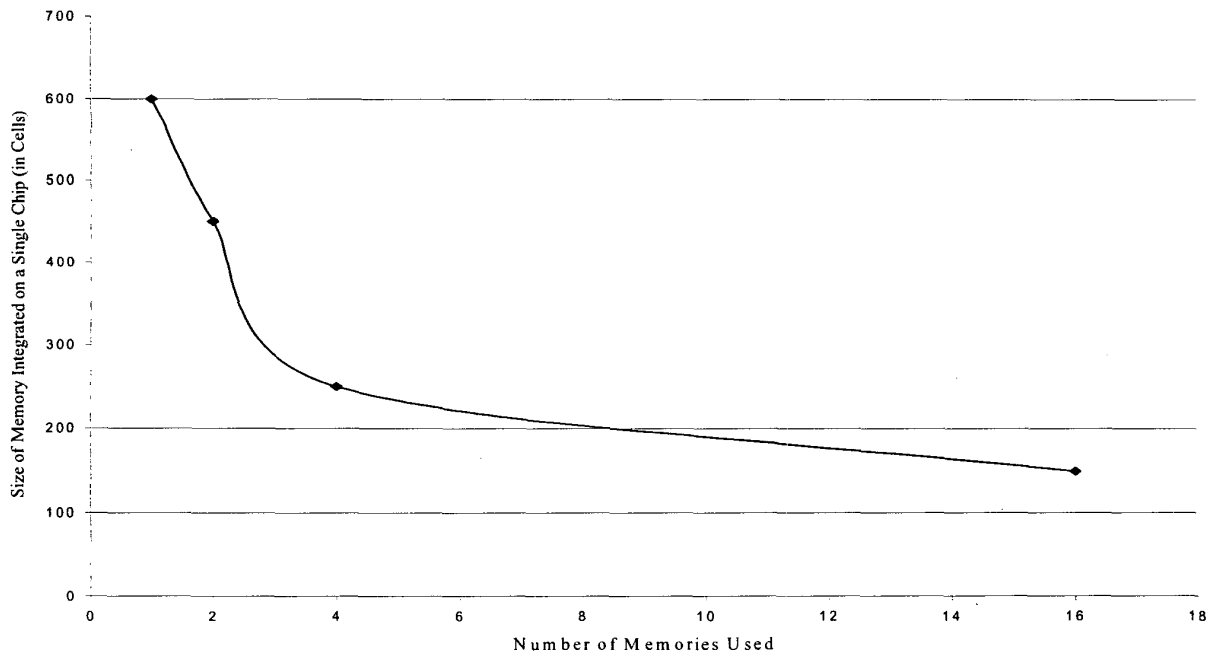


Figure 2. Size of memory integrated on a single chip (in cells) Vs. Number of memories used to implement a 16x16 ATM switch and achieve 10^{-5} cell loss probability

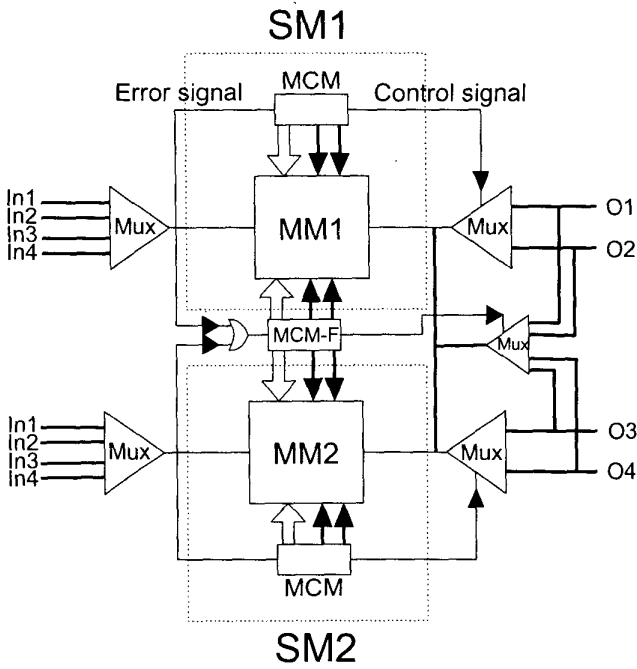


Figure 3. Architecture of 4x4 CCOQ switch

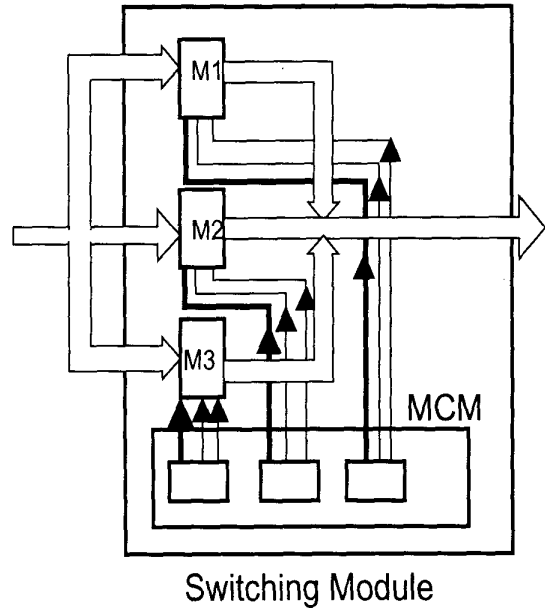


Figure 4. The switching Module

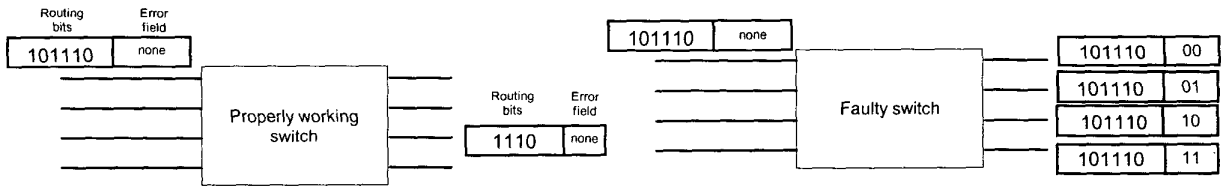


Figure 5. Traffic generated by faulty and properly working switches

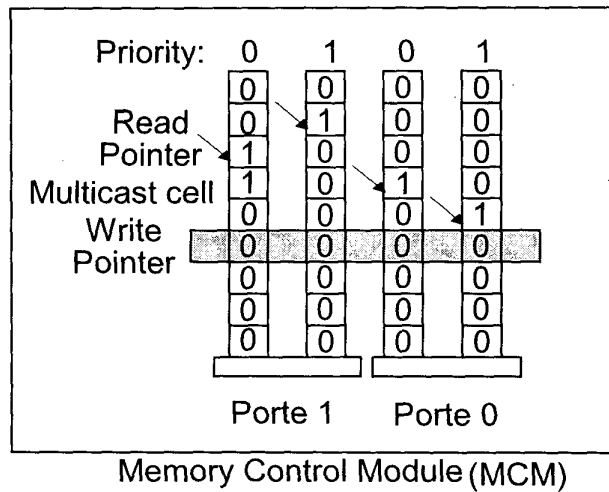


Figure 6. The Memory Control Module description

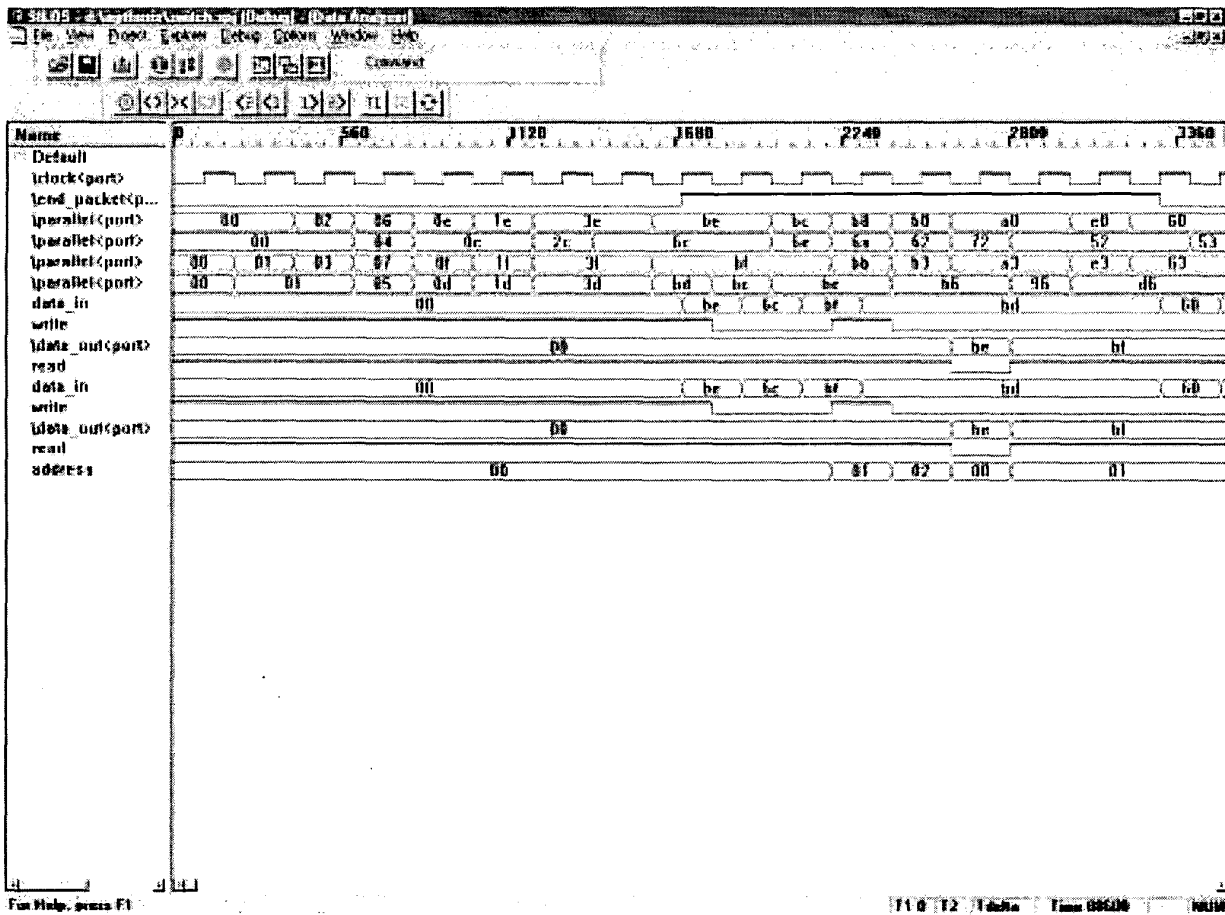


Figure 7. Verilog simulation results

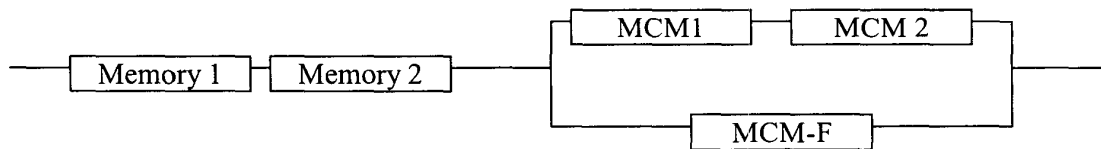


Figure 8. Reliability model of proposed switch

Table 1. Comparison of buffering techniques for 16x16 ATM switch

Number of Memories Used	Size of Memory Integrated on a Single Chip	Total Memory Size Needed
1 (Shared Buffering)	600	600
2 (CCOQ M=2)	450	900
4 (CCOQ M=4)	250	1000
16 (Output Buffering)	150	2400