

# A 64 x 10 Mb/s to 16 x 50 Mb/s Area-Efficient Time-Domain Multiplexer and Packetizer

Vaughn Betz, Mark Stoodley, Guy Lemieux, David Jones, Ralph Wittig and Paul Chow

Department of Electrical and Computer Engineering  
10 King's College Road  
University of Toronto  
Toronto, Ontario  
CANADA M5S 3G4

## Abstract

A time-domain multiplexer and packetizer chip whose architecture is applicable to both ATM packet generation and hierarchical-multiplexing switches is described. This chip adds packet headers and performs 4-to-1 packet-by-packet time-domain multiplexing on its 64 10 Mb/s inputs, resulting in 16 50 Mb/s data outputs and an aggregate output data rate of 800 Mb/s. The chip features a clock-switched architecture using minimal double-buffering which allows all data manipulation to be performed bit serially. This clock-switched architecture is both fast and area-efficient, allowing the 50 MHz target frequency to be met in a 1.2  $\mu\text{m}$  CMOS process while keeping the core area small.

## 1. Introduction

Recently there has been a great deal of interest in implementing video servers and videoconferencing over Asynchronous Transfer Mode (ATM) communication networks. The first stage in any such system is the conversion of the continuous compressed digital video bitstream into packets suitable for ATM transmission. This is accomplished by breaking the continuous bitstream into 48-byte packets, and adding a 5-byte header to each packet. Since the average data rate of MPEG-2 compressed video is approximately 4 Mb/s, several video packets must then be time-domain multiplexed together packet-by-packet in order to make efficient use of the 155 Mb/s ATM transmission rate. The time-multiplexed packets can then be sent over an ATM communication network to their destination.

In this application, entire packets are time-domain multiplexed together, so considerably more buffer space is required than in the case of byte-by-byte time-domain multiplexing. This fact, coupled with the fairly high data rates at the multiplexer output, motivates the search for an architecture that is both fast and area-efficient.

This paper describes a chip implementing such an architecture. This chip was designed both to provide proof-of-concept for the ATM application described above and to

function as part of a 64 x 64 video switch. This paper first describes this video switch, which uses an architecture that fits into the broad class of hierarchical multiplexing switches [1]. Hierarchical multiplexing switches have good scalability properties, and the use of this architecture allows us to employ an existing 16 x 16 packet switch as a switching core in a 64 x 64 switch [2,3].

The overall switch architecture requires that the 64 continuous input data streams be buffered, packetized and packet-by-packet time-domain multiplexed by a factor of 4. This paper describes a fast, area-efficient architecture for this packetizing and time-domain multiplexing function and presents experimental results from a 1.2  $\mu\text{m}$  CMOS implementation. The architecture of this chip is also directly applicable to systems in which ATM packets are assembled and time-domain multiplexed prior to transmission over an ATM network.

## 2. Video Switch Architecture

A block diagram of the video switch architecture is shown in Fig. 1. It consists of three distinct components: a packetizer and time-domain multiplexer chip operating on 64 continuous 10 Mb/s input data channels of MPEG-2 compressed video, a 16 x 16 packet switch, and a chip that strips off the packet headers and demultiplexes the signals back into 64 continuous data channels. Note that while the average data rate of MPEG-2 compressed video is approximately 4 Mb/s, the peak data rate can be considerably higher, so it was decided that the entire system would be designed for 10 Mb/s data rates.

The packetizer/multiplexer (PM) chip buffers the input data streams, adds a packet header containing a six-bit destination address and four control bits, and then time-domain multiplexes a group of four inputs packet-by-packet onto each output at 50 Mb/s. The packet header for each input channel is programmed via 14 inputs to the packetizer chip; these 14 signals consist of 6 address (channel number) lines, 7 data lines (three of the ten bits in the header are control bits hard-wired to zero or one) and one programming strobe line.

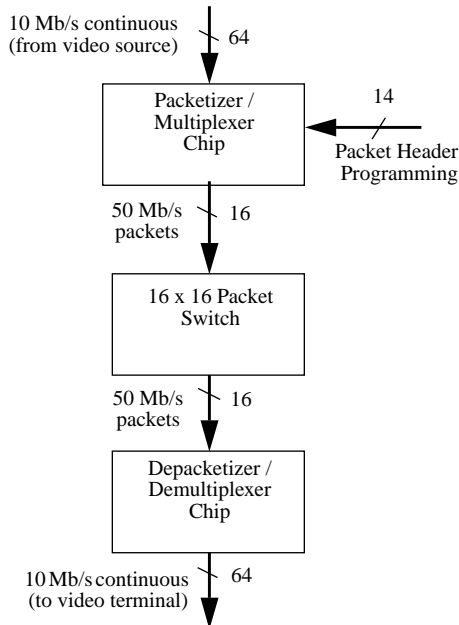


Fig. 1. Video switch architecture.

A 16 x 16 packet switch [2,3] takes these 50 Mb/s data packets as input and uses the high-order four bits of their destination address to determine to which of its 16 outputs each packet is routed. Finally, the depacketizer and demultiplexer chip uses the low-order two address bits of each packet to determine which of four adjacent outputs should receive the packet. It then strips off the packet header, and buffers the packets so that the 64 data outputs are again 10 Mb/s continuous bit streams.

This system provides full connectivity; each input channel can be switched to any output channel. This paper will focus on the architecture of the packetizer/ multiplexer (PM) chip; the design of the key blocks of the depacketizer/demultiplexer chip is similar and employs the same clock-switched buffer technique.

### 3. PM Chip Architecture

The PM chip consists of central control logic and 16 identical datapath blocks, each of which packetizes four inputs and time-domain multiplexes them onto a single output. As the PM chip is assembling and multiplexing full packets, it requires considerably more buffer space than byte-by-byte time-domain multiplexers. Hence it is important that we have an area-efficient structure to buffer and multiplex the input signals.

Fig. 2 shows one possible architecture for these four-channel datapath blocks. For each channel, the appropriate header is transmitted, an input buffer is parallel loaded into the output buffer, and then shifted out serially at 50 MHz.

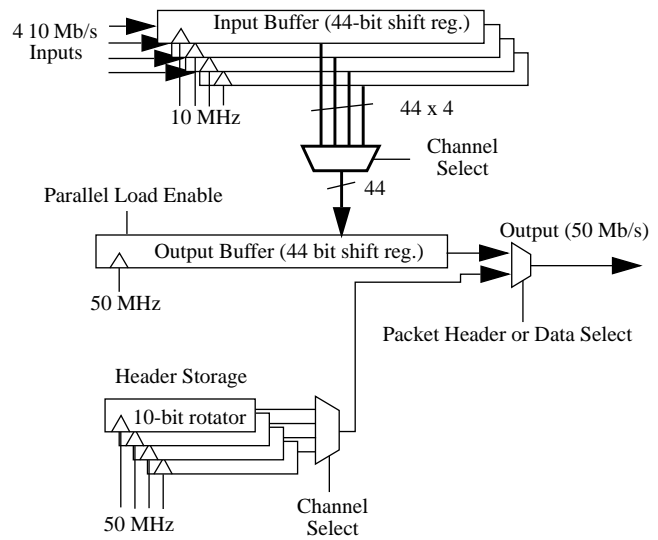


Fig. 2. Straightforward architecture of 4-to-1 time-domain multiplexer and packetizer.

This sequence is then repeated for the next channel in the group of four. The input buffers are 44 bits long because this is the minimum data packet length that permits four 10 Mb/s data channels, with the 10-bit packet header and one spacer bit required by the packet switch added, to fit within the 50 Mb/s data rate expected on the packet switch inputs. While the architecture of Fig. 2 is simple and fast, it is not very area-efficient, since it requires 44 4:1 multiplexers, an extra 44-bit shift register and considerable wiring to implement the parallel load. Dual-ported SRAMs could be used instead of shift registers, providing a slower, but possibly more area-efficient implementation. Our design takes a different approach; we switch the clock frequency on some buffers in order to remove the area penalty associated with the parallel load, while maintaining the intrinsic speed of a shift-register based design.

#### 3.1. Clock-Switched Input/Output Buffer

Fig. 3a shows this clock-switched input/output buffer for one channel, while Fig. 3b shows how four of these buffers and the storage for four packet headers are connected to form a complete 4-to-1 time-domain multiplexer and packetizer. By altering the clock rates on the input/output buffers in Fig. 3a, we can perform all the time-domain multiplexing in a bit serial fashion and significantly reduce the circuitry required. The architecture of Fig. 3b uses the same number of storage bits as the simpler architecture of Fig. 2, while reducing the number of multiplexers required from 45 4:1 and one 2:1 to only two 4:1 and five 2:1 multiplexers. During the 900 ns taken to output the 44 data bits plus the one spacer bit required by the packet switch, nine new input data bits will arrive from

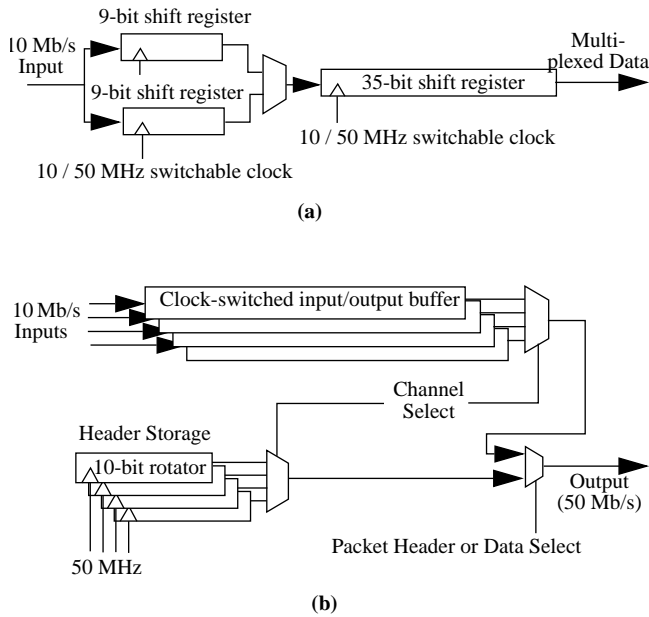


Fig. 3. (a) Clock-switched input/output buffer; (b) 4-to-1 time-domain multiplexer and packetizer.

this channel's 10 Mb/s input. Hence, we must double buffer the first nine bits of the channel's shift register. As Fig. 4 shows, the control logic cycles the input/output buffer through four distinct states by changing the multiplexer inputs and buffer clocks. In Receive A, the upper nine-bit shift register and the 35-bit shift register are chained together and clocked at 10 MHz to buffer input data. Once the buffer is full, state Transmit A is entered; the data in the upper nine-bit register and the 35-bit register are clocked out at 50 MHz, while the lower nine-bit register stores the input 10 Mb/s data. Once all

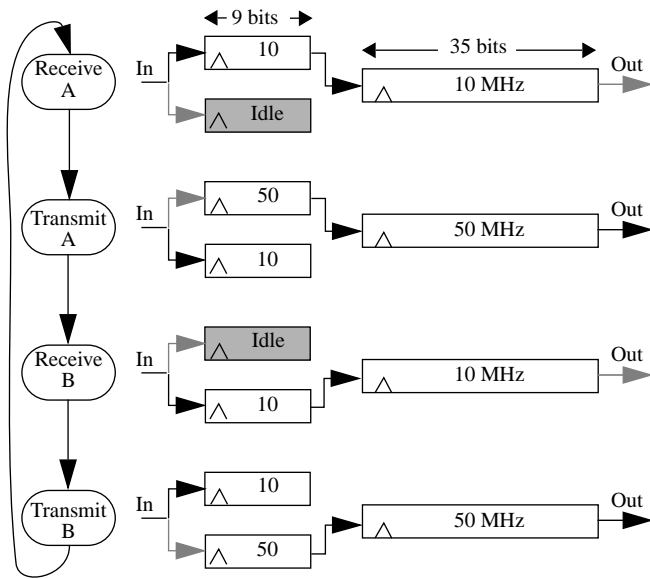


Fig. 4. Clock-switched time-domain multiplexer operation.

the upper stored data is output, state Receive B is entered, allowing the now full lower nine-bit register to be chained to the 35-bit register. Finally, state Transmit B is the mirror image of Transmit A, with the upper nine-bit register storing input data as the data in the lower nine-bit register and the 35-bit register are clocked out at 50 MHz.

### 3.2. Clock gating issues

Some care must be taken when changing the clock frequency on shift registers. The first potential problem is that the logic used to generate the various switched clocks can cause skew between these switched clocks. We use two-phase non-overlapping clocks with a small amount of dead time (5 to 10% of the cycle time) between the phases so that some skew between the clocks arriving at the different buffers in Fig. 4 is tolerable. As well, the multiplexers which switch the clocks all use an identical layout and are placed close together on the die in order to minimize process variations that could cause skew between the switched clocks.

Another problem that must be guarded against are "glitches" on the clock lines caused by the switched clock frequency being changed in the middle of a clock pulse. As Fig. 5 shows, the control signals that switch clocks are always latched on the 50 MHz clock phase opposite to the one being gated. This latching means that we always change the frequency of phase 1 before changing the frequency of phase 2, and that the signal gating a clock phase is always stable when the 50 MHz version of that clock phase is high. To ensure that the signal gating a clock phase is also stable when the 10 MHz version of that clock phase is high, all state machines are designed so that a switched clock's frequency is changed only when the 10 MHz clock of that phase is low. As Fig. 6 shows, this combination of properly latching the clock select signal and only switching a clock phase when the corresponding 10 MHz clock phase is low avoids clock feedthrough, where both clock phases are high simultaneously. When reset, the PM chip waits for the next low-to-high transition of the 10 MHz phase 1 clock before starting operation; hence the starting position of all state

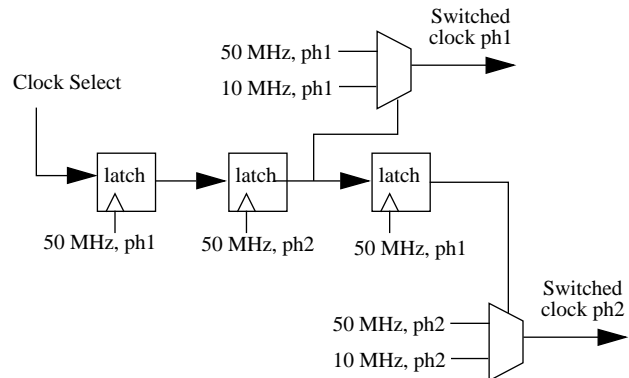


Fig. 5. Logic used to switch clock frequencies.

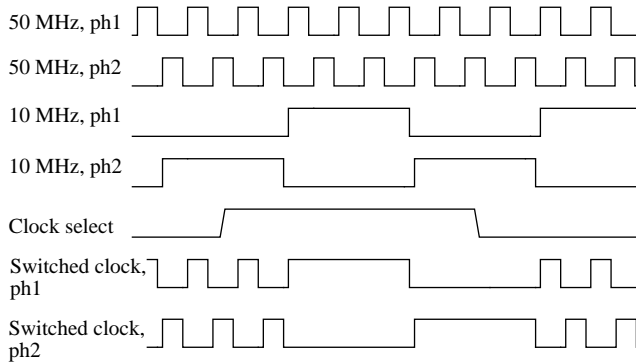


Fig. 6. Clock switching, avoiding clock feedthrough.

machines relative to the 10 MHz clock is known.

#### 4. Results

The PM chip has been implemented in a 1.2  $\mu\text{m}$ , two-level metal CMOS process. Table 1 summarizes its key characteristics and Fig. 7 is a micrograph of the chip. While tester and test fixture limitations have prevented testing at speeds higher than 20 MHz, detailed SPICE simulations show that it will run comfortably at 50 MHz. This prototype version uses ten-transistor static registers for the buffers to simplify testing. Changing the design to use six-transistor dynamic registers is straightforward, and would lead to a considerably smaller and faster implementation.

Table 1: Prototype PM chip characteristics.

Technology	1.2 $\mu\text{m}$ CMOS, 2 level metal
Power Supply	5 V
Core Size	3.44 x 3.52 mm
Die Size	5.37 x 5.37 mm
Transistors	55 884
Pads	96 (Only 32 inputs bonded out on this version. Full bonding requires 128 pads.)
Operating Frequency	50 MHz simulated, verified up to 20 MHz at room temperature.

#### 5. Conclusion

A switch architecture capable of switching 64 channels of MPEG-2 video via the hierarchical multiplexing concept was presented. This switch requires fast and area-efficient buffering, packetizing and packet-by-packet time-domain multiplexing of its inputs. The switched-clock input/output buffer presented here performs the time-domain multiplexing operation in a completely bit-serial fashion by switching the clock frequencies on its shift registers. This design is area-

efficient, and simulations show that it easily meets the 50 MHz target frequency.

The PM (packetizer/multiplexer) chip can be readily modified to create and time-domain multiplex complete ATM packets for transmission over ATM networks. The modifications required involve increasing the amount of storage in the input/output buffers to allow the generation of 384 bit long ATM data packets, increasing the header size to 40 bits, and time-domain multiplexing a larger number of packets together to efficiently utilize the 155 Mb/s ATM data rate.

#### Acknowledgments

The authors would like to thank the Canadian Microelectronics Corporation for their fabrication and infrastructure support, Jaro Pristupa for our local CAD support, and NSERC for its financial support. The authors are also indebted to Dr. Leon-Garcia of the University of Toronto for his helpful suggestions and discussions.

#### References

- [1] K. Y. Eng and M. J. Karol, "High-Performance Techniques for Gigabit ATM Switching and Networking," IEEE Int. Conf. on Communications, pp. 1673 - 1678, 1993.
- [2] P. Chow, D. Karchmer, P. Chow, R. White, T. Ngai, P. Hodgins, D. Yeh, J. Ranaweera, I. Widjaja, and A. Leon-Garcia, "A 50,000 Transistor Packet-Switching Chip for the Starburst ATM Switch," IEEE Custom Integrated Circuits Conf., pp. 435-438, 1995.
- [3] I. Widjaja and A. Leon-Garcia, U.S. Patent Number: 5274642, Issued: Dec. 28, 1993.

Fig. 7. Chip micrograph