

A System-on-a-chip for MPEG-4 Multimedia Stream Processing and Communication

E. Juárez, M. Mattavelli, D. Mlynek

Integrated Systems Laboratory (LSI), Swiss Federal Institute of Technology

CH-1015, Switzerland

Phone: +41 21 6936973; Fax: +41 21 6934663

E-mail: Eduardo.Juarez@epfl.ch

ABSTRACT

Keywords: *VLSI, systems-on-a-chip, DMIF, MPEG4, multimedia applications.*

A VLSI architecture for multimedia stream processing and communication is presented. This system-on-a-chip is able to communicate multimedia applications -like interactive multimedia or mobile multimedia- over generic networks. Main features of the architecture are scalability in the number of multimedia streams managed, bandwidth sharing, capacity to control the offered service quality and possibility to implement mobile applications. MPEG4 multimedia stream requirements are included in the architecture. Four main units are distinguished in the design: cell communication, QoS control, protocol processing and DMA (Direct Memory Access). Preliminary results of an implementation of the cell communication unit as an ATM-cell-based multiplexing one show the suitability of the architecture for STS-12/STM-4/OC-12 throughputs (622.08 Mb/s).

1. PROBLEM STATEMENT

Available high-speed network throughputs, in the order of Gb/s, allow multimedia applications to interconnect using their infrastructures. In order to give service to these applications a system offering multimedia stream processing and transport capabilities is needed. Interactive multimedia and mobile multimedia applications (either computer supported cooperative work, CSCW, or emergency-response) are examples that will use such a system.

One potential problem of multimedia stream communication over high-speed networks is information loss [1]. When several sources transmit at their peak rates simultaneously, some buffers available in network switches may overflow and the subsequent drop of information, e.g. cells, packets, leads to severe degradation in service quality due to synchronization loss in channel coding mechanisms.

Another point to face is the information exchange between the application running on a host processor and the system. Data is considered to be touched any time is read from or

written to main memory. Any system architecture should try to minimize data touches because of the large negative impact they can have on performance.

2. PROBLEM REQUIREMENTS

2.1 General requirements

A system offering this service of multimedia stream processing and communication over a generic network should meet the following requirements if it wants to cover applications such as those mentioned above:

- The system should easily scale the number of streams it manages and the bandwidth associated to each of them to accommodate future service demand increases. Scalability of multimedia content and management is critical for associating new information products with various services and applications.
- The system should fairly share the available bandwidth between all different users. This feature will enable, for instance, either to increase the number of streams to be multiplexed when the available bandwidth is fixed or to reduce the necessary bandwidth to multiplex a fixed number of them. If users with heterogeneous traffic patterns want to be simultaneously served different guarantee bandwidths should be reserved for different streams.
- The system should be able to give service to mobile/portable users connected by either wireless or infrared links.
- The system should be able to control the quality of service (QoS) offered. If no control is applied in order to keep it constant, quality degradation will depend sharply on network congestion conditions [2]. Hence, to prevent this degradation some kind of multimedia stream processing will be needed.

QoS control, fair bandwidth sharing and mobility/portability are system requirements not included in other works [3].

2.2 MPEG4 requirements

We have considered MPEG4 multimedia streams as inputs to the system. Figure 1 below positions system functionality with respect to DMIF (Delivery Multimedia Integration Framework) [4] and MPEG4-Systems [5] standards. The requirements that should be met are the following:

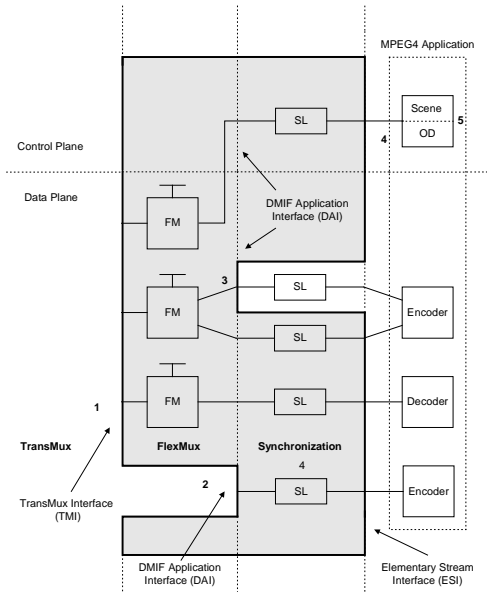


Figure 1: MPEG4-related functionality considered in the system

- At the TransMux Interface (TMI), the system should generate a FlexMux (FM) Stream, i.e. any arbitrary mixture of Simple mode FlexMux_PDUs and MuxCode mode FlexMux-PDUs. At the DMIF Application Interface (DAI), an SL_PDU (Synchronization Layer Protocol Data Unit) packetized stream should be generated. (see numbers 1 and 2 in figure 1).
- As an MPEG4 implementation does not have to include the Elementary Stream Interface (ESI), the system should be able to receive/send a sequence of packets through the DAI (DMIF Application Interface) – number 3 in figure 1.
- The system should generate SL variable syntax packets and hence, it should need to have access to the SLConfigDescriptor for each ES that determines the syntax elements. SLConfigDescriptors are part of Object Descriptors (ODs) - see number 4 in figure 1.
- The system should be responsible for splitting Access Units in appropriate SL_PDUs that do not lead to transport packets that are larger than the maximum size of the path transfer unit.

- The system should encapsulate Elementary Streams carrying control data: Object descriptors (ODs) and BIFS (Binary Format for Scene). ODs dynamically describe hierarchical relations, location and properties of Elementary Streams (see number 5 in figure 1).

3. ARCHITECTURE DESCRIPTION

Distributing the multimedia stream processing and communication functions between different sources allows meeting efficiently the requirements of mobility and streaming scalability. As can be seen in figure 2, this goal can be achieved using a basic unit (BU) with a communication processor (CP) and a stream processor (SP). The BU locally applies a communication function to each stream group that has been processed or not by the SP.

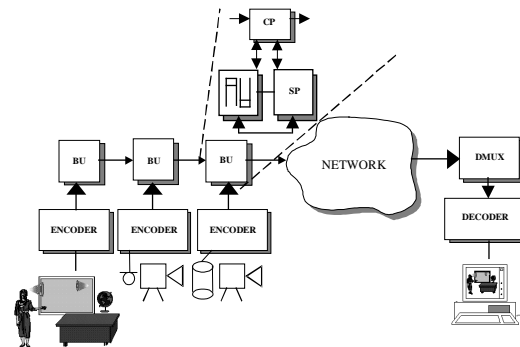


Figure 2: Decomposition of the system in an array of independent basic units and possible working scenario.

Inside the BU a mapping from network PDUs and multimedia stream DUs (data units) to generic, programmable-syntax cells is done (figure 3). A cell network, whose extension is the whole chip, interconnects different processing elements (PEs) of the BU: the communication processor, the stream processor and the storage. This concept of ChIP Area Network (CIAN) is an extension to a smaller scale of that of Desk-Area Network (DAN)[6].

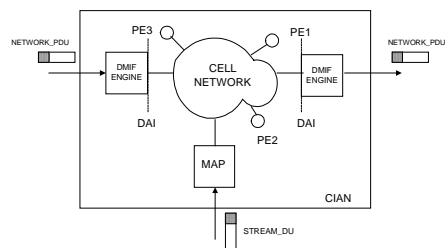


Figure 3: Mapping functionality and Chip Area Network (CIAN) inside the BU.

Figure 4 shows how the communication processor works when it is programmed as a multiplexer: cells carrying information from the multimedia source are stored in a queue until the MAC (Medium Access Control) distributed algorithm gives permission to insert cells. When an empty cell is found at the communication processor input and the MAC algorithm allows insertion, this cell disappears from the flow and a new one is inserted.

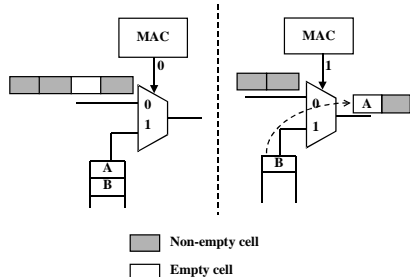


Figure 4: Communication processor working as multiplexer.

The medium access control (MAC) algorithm solves access conflicts in the distributed environment when several users try to communicate using the same resource and allows to meet the requirements of fairly bandwidth sharing and guaranteed reservation. [7]

Figure 5 shows the details of the basic unit. There are four main blocks:

- Cell communication unit, with the input/output DMIF engines, input FIFO module and the communication processor.
- QoS control unit. It manages multimedia information in order to produce a smooth quality of service degradation when network suffers from congestion.
- Protocol processing unit. It acts as a protocol stack builder and adapts information either coming from the multimedia stream source for network transmission or coming from the network to the multimedia stream receiver
- DMA (Direct Memory Access) unit. It communicates with the application running on a host processor.

4. PRELIMINARY RESULTS

The cell communication unit has been initially implemented as a multiplexing unit for ATM networks. The main functionality of this unit consists on adding an ATM-cell stream to the ATM-cell input stream. Internally, the information unit used is an ATM cell and hence, no protocol mapping is necessary. Neither, the DMIF engines have been included.

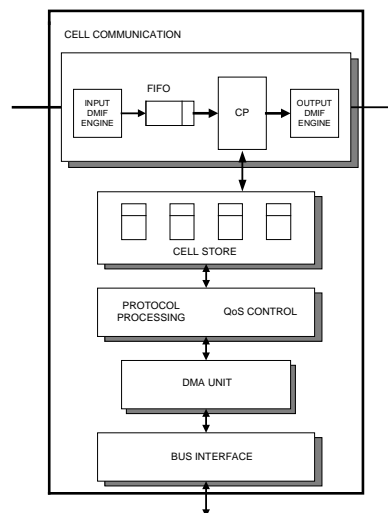


Figure 5: Basic Unit Architecture

There are four modules in this unit (Figure 6): input, input FIFO, multiplexing and output. Interconnection between them is done through a simple linear network. Main design features of the different modules are as follows:

- Input module receives a flow of UTOPIA cells (54 octets/cell) [8]. In this flow it identifies non-assigned cells [9], checks cell parity, assures cell integrity and implements UTOPIA protocol levels 1 and 2 [8,10].
- Output module generates non-assigned cells when there is no information to be sent, inserts a parity code in the UTOPIA cell UDF2 (User Defined Field 2) [8] to check transmission errors between cell communication units and implements UTOPIA protocol levels 1 and 2.
- To have an autonomous behavior in each communication unit, each of them is synchronized with independent clocks of the same nominal frequency. This means that two clock domains cohabit within one cell communication unit: the input cell and the output cell ones. These two plesiochronous clock domains generate metastable behavior at the domain interface flip-flops. Techniques to reduce the probability of having metastable behavior in these flip-flops have been implemented to achieve reliable system function.
- Input FIFO module main functionalities are isolation of the two different clock domains and reduction of the consequences of having a metastable behavior. Besides, it stores up to nine input cells when the output stream is stopped as a consequence of the UTOPIA fullness signal activation. To avoid memory overflow problems due to the slightly different throughputs of the input FIFO write and read processes, non-assigned cells, that the MAC cannot use, are inserted to limit the maximum number of consecutive assigned cells.

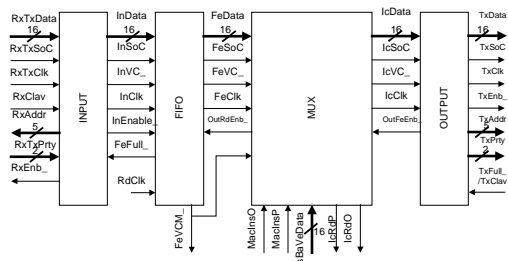


Figure 6: Cell communication unit for ATM networks implemented as a multiplexing unit

- Multiplexing module changes empty cells by assigned ones and generates the new stream.

Table 1 summarizes area results for the Cell Multiplexing Unit (DLM CMOS 0.7 μm ATMEL-ES2 technology):

Area (mm^2)	Inp	FIFO	Mux	Output	TOTAL
Combinatorial	.13	.11	.11	.14	.49
Sequential	.14	.19	.19	.13	.65
Estimated interconnect.	.61	.63	.63	.55	2.42
Memory	.00	2.38	.00	.00	2.38
TOTAL	.88	3.31	.93	.82	5.94

Table 1: Area results for the different modules of the cell communication unit (ATM case)

The throughput achieved of 0.9 Gbit/s (16 bit @ 57Mhz) allows the unit to be used with SDH/SONET standards, STS-12/STM-4/OC-12 (622.08 Mb/s). Although in [11] a better throughput was achieved, 2.5 Gbit/s, no QoS control block has been provided. We expect to have similar throughput results using similar technology (0.5 μm).

5. ACKNOWLEDGEMENTS

The authors thank R. Herranz and E.A. Gilarranz for design contributions, S. Alexandres (Pontificia-Comillas University of Madrid), F. Moreno (Technical University of Madrid) and J. Meneses (Technical University of Madrid) for support and encouragement and all people mentioned for fruitful discussion.

Part of this research was done while the first author was with the Electronic Engineering Department at the Technical University of Madrid.

6. REFERENCES

[1] Verscheure, O., X. Garcia, G. Karlsson, J-P. Hubaux. " User Oriented QoS in Packet Video Delivery". *IEEE Network*, November/December 1998, pp. 12-21.

[2] Luo, W., and M. El Zarki. "Transmitting Scalable MPEG-2 Video over ATM-based Networks". *Technical Report*, Video

Processing and Telecommunications Lab., Dept. of Electrical Engineering, University of Pennsylvania, 1996.

[3] Zubin, D. D., J. R. Cox Jr., G. M. Parulkar. "Design of the APIC: A High Performance ATM Host-Network Interface Chip". *Proceedings of IEEE INFOCOM 1995*, pp. 179-187.

[4] ISO/IEC 14496-6 V2 PDAM1. "Delivery Multimedia Integration Framework, DMIF" Jul. 1999.

[5] ISO/IEC 14496-1 V2 FPDAM1. "MPEG-4 Systems" Jul. 1999.

[6] Barham, P., M. Hayter, D. McAuley, and I. Pratt. "Devices on the Desk Area Network". *IEEE Journal on Selected Areas in Communications*, Vol. 13, No. 4, May 1995, pp. 722-732.

[7] Varma, A., D. Stiliadis. "Hardware Implementation of Fair Queuing Algorithms for Asynchronous Transfer Mode Networks", *IEEE Communications Magazine*, Dec. 1997, pp. 54-68.

[8] ATM Forum Technical Committee AF-PHY-0017. "UTOPIA (Universal Test & Operations PHY Interface for ATM) Specification, Level 1, Version 2.01" March 1994.

[9] ATM Forum Technical Committee AF-UNI-0010.002 "ATM User-Network Interface Specification Version 3.1", 1994.

[10] ATM Forum Technical Committee AF-PHY-0039. "UTOPIA (Universal Test & Operations PHY Interface for ATM) Level 2, Version 1.0" Jun. 1995.

[11] Riesco J., J.C. Díaz, L.A. Merayo, J.L. Conesa, C. Santos and E. Juárez. "On the Way to the 2.5 Gbit/s ATM Network: ATM Multiplexer/Demultiplexer ASIC". *The European Design and Test Conference. ED&TC 97*, Paris, March 1997.