

Reprint

Analysis and Implementation of the Gateway between the ETHERNET and a High-speed Multiservice LAN

J. Koutsonikos, T. Antonakopoulos, V. Pallios and V. Makios

The 8th IASTED International Symposium on Applied
Informatics

INNSBRUCK, AUSTRIA, FEBRUARY 1990

Copyright Notice: This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author's copyright. In most cases, these works may not be reposted or mass reproduced without the explicit permission of the copyright holder.

"Analysis and Implementation of the Gateway between the ETHERNET and a High-Speed Multiservice LAN"

**J. Koutsonikos, T. Antonakopoulos, V. Pallios and V. Makios
Laboratory of Electromagnetics, University of Patras, Greece**

Abstract: The wide diffusion of communication services demands the generation of Integrated Services Local Networks to provide these services in a cost effective manner. These networks have also to provide powerful interconnection with traditional LANs like the ETHERNET. This paper addresses the ETHERNET interworking solution adopted for a Local Integrated Optical Network (LION). The hardware architecture of the Gateway is described as well as the used communication software. Finally, the impact of the Gateway to the LION node performance is also presented.

I. Introduction

Integrated Services Local Networks (ISLNs) connect usually a few hundred nodes, supporting thousands of individual users and integrate different kinds of traffic, encompassing data, voice and images. The interconnection needs between these systems become essential and the requirement for economic and flexible access to external homogeneous networks, through Bridges (BRG) and heterogeneous networks, through Gateways (GTW), becomes critical as far as the network performance is concerned.

LION [1] is intended to integrate the above mentioned services covering areas of diameter in the range of a few hundred meters up to ten kilometers. Thus, a two-level architecture has been adopted and a new high-performance medium access protocol has been developed, based on the so-called hybrid-switching technique, which provides both circuit-switching and packet-switching capabilities.

The stream traffic is supported through a transparent "bearer" service at the MAC sublayer of the International Standards Organization (ISO) model for Open Systems Interconnection (OSI) and the I.450, I.451 ISDN protocols. The packet traffic is supported by an OSI protocol profile, as following: the 2a OSI sublayer (MAC) is provided through a specially developed Access Protocol and hardware based on the Hybrid Switching concept [2], [3], due to the integration of stream and packet traffic and the expected

workload; the 2b OSI sublayer follows the LLC1 protocol; the Network Layer is based on the inactive Internet Protocol (IP) for the LION users and the active IP for the external users; finally, the Transport Layer follows the Transport Class 4 Protocol (TR4). The impact of the internetworking problems is mirrored on the protocol profile choice since the network is considered as a "distributed end system" resulting to a connectionless approach up to layer 3 (network layer). The implemented ETHERNET to LION Gateway (ELGTW) provides access to the network layer and supports the required address transformation and relay functions.

The ELGTW hardware structure is described in section II, where the need for the development of a special interface board is highlighted. In section III, the used communication software structure is explained, emphasizing the way the information exchange is performed under Real-Time constraints. The impact of the Gateway to the LION node performance and vice versa, based on simulation methods, is described in the last section.

II. The Gateway hardware structure

The internetworking problem requires a careful choice between a protocol profile, matching that of the external network and a protocol profile, conceived to support efficiently the internal communications. The critical parameters are specified by traffic requirements, especially in view of the huge workload of an ISLN. The LION node architecture is shown in Fig.1, including the ETHERNET Gateway, based on the network node multiprocessing architecture. This structure offers several benefits in terms of modularity, flexibility and capability to adapt to the specific characteristics of the interconnected networks. The ETHERNET-LION Gateway takes advantage of the adopted communication protocol profile, based on ISO 8473/AD1 Internet Protocol. End-systems exchange data units in a connectionless mode. The crossed subnetworks are only requested to

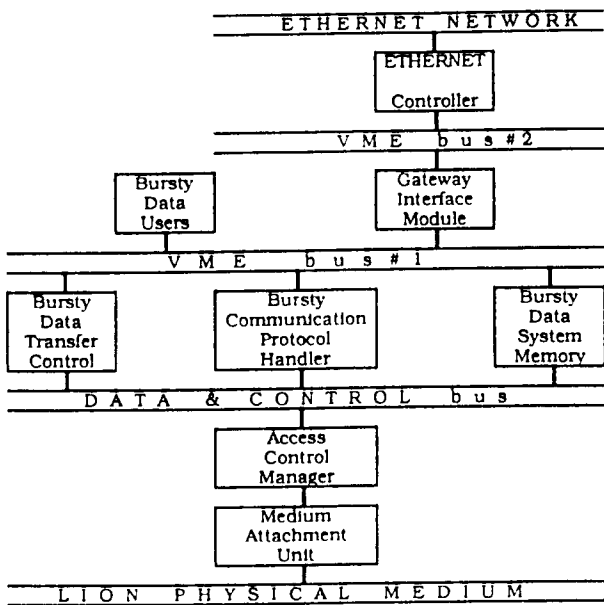


Figure 1. The LION node Architecture

provide a data pipeline, on which data are routed independently. The convergence between ISO IP and the underlying LLC1 is a simple one-to-one primitive mapping. In this interconnection scheme, the Gateway is not burdened, but it has only to perform encapsulation-decapsulation functions, buffering and CPU allocation.

The hardware implementation of the Gateway is shown in Fig.2. It includes an ETHERNET Controller and an Interface Module. The ETHERNET Controller, which is based on the MVME330 board, includes a 68000 CPU at 10 MHz, dynamic RAM of 512 Kbytes, ROM of 64 Kbytes, where the Operating System Kernel and the communication protocol software are loaded, and the LANCE chip, which is the hardware implementation of the CSMA/CD access protocol.

The special Interface Module (IM) has been developed to accommodate:

- i) Additional ROM requirements for the Gateway software,
- ii) The needed control mechanisms (special interrupt handling for the Gateway and the network node communications),
- iii) Dual port RAM for the implementation of a mailbox intercommunication system between the Gateway and the network node, and
- iv) FIFO memories (receive and transmit), which provide fully independent operation of the Gateway and the network node.

The communication between the Gateway and the LION node comes through the use of the Data Transfer Control Block (DTCB) mechanism. The Controller is polling the flag of the receive DTCB, which indicates, when set, that the LION node has a packet to send to the Gateway, then it sends an interrupt to the node and, when the transfer is completed, it copies the packet data in the local memory. When the Gateway has a packet to send to the node, the Controller copies the packet data to the transmit FIFO of the Interface Module, informs the DTCB with the packet length and the packet address, sets the flag of the transmit DTCB and sends an interrupt to inform the node to start the DMA transfer of the packet.

The transmission of a packet from the ETHERNET Controller to the LION node, using the FIFO memories of the Interface Module, is shown in Fig.3. The Controller stores the packet in the transmit FIFO and generates an interrupt to the node using the interbus connection mechanism. The node recognizes the interrupt and receives the packet. The signal FIFO-E is the empty flag of the transmit FIFO.

The communication between the CPU and the LANCE chip is attained through receive and transmit descriptor rings. When a packet is ready to be transmitted to the ETHERNET network, the CPU moves the packet to a free transmit buffer, informs the respective transmit message descriptor with the length of the packet and

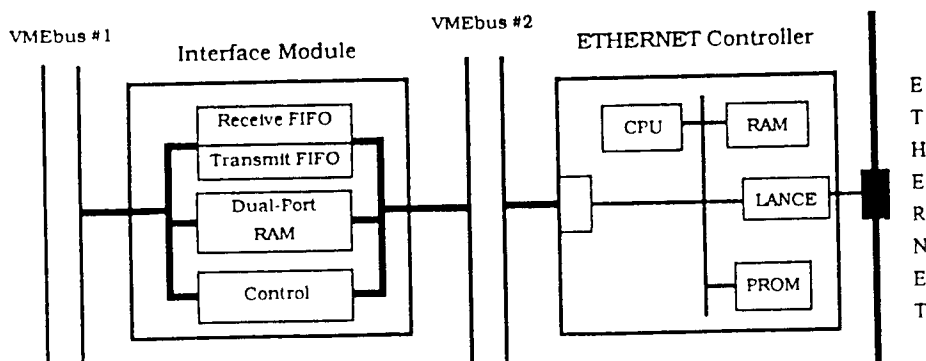


Figure 2. The LION-ETHERNET Gateway Architecture

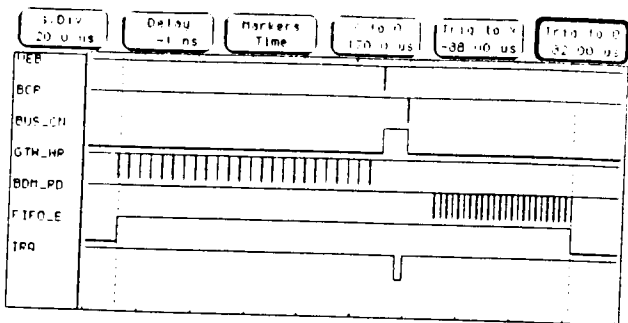


Figure 3. GTW to LION node transmission

informs the LANCE for the existence of a ready for transmission packet. The transmission of a packet from the local memory to the ETHERNET network is shown in Fig. 4. The CPU informs the LANCE with the parameters of the transmission and the LANCE gains the control of the internal bus of the MVME330 board. Then it transmits the packet in burrs of 16 bytes (16-bit transfer mode), using the internal DMA capability.

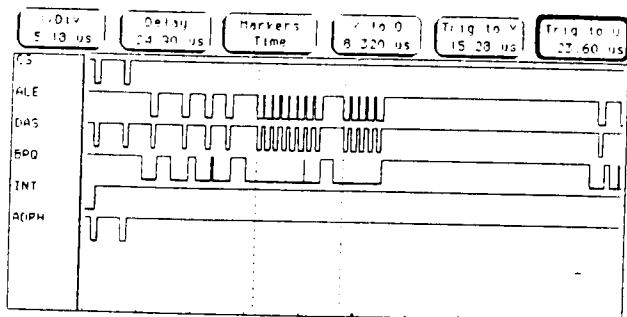


Figure 4. GTW to ETHERNET network transmission

III. The Communication software

The GTW software consists of four parts, namely:

1. The Communication Protocols Software (CPS),
2. A Real-Time Multitasking Operating System (RMOS),
3. A specially developed Low-Level Software (LLS), and
4. A specially developed software, providing the interface between RMOS and CPS, called Environment (ENV).

The ETHERNET-LION interconnection takes advantage of the adopted communication protocol profile, based on ISO 8474/AD1 Internet Protocol, carried out as is shown in Fig. 5.

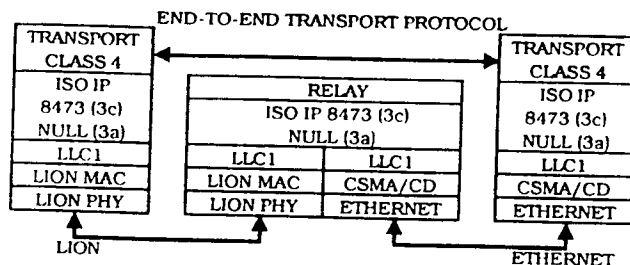


Figure 5. LION-ETHERNET Internetworking

The ELGTW follows the same protocol profile as the LION network node up to the network layer, namely CPS, which is a portable communication protocol software developed by ALCATEL / TITN, France [4], based on the OSI model. The RMOS is a modified version of RMS68K Kernel of VERSADOS, Motorola [5], suitable for the specific ELGTW requirements.

LLS is a software module, specially developed for the ELGTW, with the following responsibilities:

- i) Initialization of the ELGTW hardware and software modules,
- ii) Control of the data packets between the ELGTW and the network node, and
- iii) Management of the implemented dual port and FIFO buffers of the IM.

ENV is also a specially developed software, which includes after adaptation the AXI, GTIM and GBUF software modules, developed by ALCATEL/TITN. The GBUF module is responsible for the CPS memory resources management under the RMOS procedures and functionalities. GTIM is responsible for the timing resources of the system and the RMOS task execution scheduler. AXI has the responsibility to initialize the system tasks with the specified parameters and to provide the required system resources and resource management to the rest of the system tasks. The RMOS communication to the system tasks is carried out through the Environment module in the form of primitives exchanged between each task and RMOS. The responsibilities of the ENV are:

- i) Interface for subsystem access by user entities and entity access by the subsystems.
- ii) Isolation of the Operating System and hardware environment, providing the required format transformation, drivers, utilities and monitoring interfaces.
- iii) Management and allocation of the system resources, especially buffers, queues and timers.
- iv) Intercommunication between the various system tasks, and
- v) User interface in the form of C language functions for transmitting and receiving the primitives of the various subsystems and the Real-time Multitasking

Operating System.

The developed drivers have on one hand the role of data format transformation, between the various parts of ENV software, which provides more efficient operation. On the other hand, they provide the required communication primitive and data format transformation with the external world (the LION node), during initialization for identity, quality of service and various procedural functions, required by the ELGTW and the LION global software architecture.

The ELGTW software has the advantage of a modular and largely independent from the Operating System implementation, since the porting requirements are relied only in implementing a new Environment module and drivers and the rest of ENV, CPS and LLS software can be directly reused.

IV. Performance Analysis

The Gateway performance analysis is based on simulation methods, concerning the node architecture shown in Fig.1. The VMEbus access from the Gateway, the users and the packet processing modules is priority-driven, where the packet processing modules have the higher priority and the packet users with the Gateway have the lower priority.

In Fig.6 the throughput-delay characteristics of the Gateway are shown in the cases that the packet users have a low or a high packet arrival rate. It is obvious that, if the Gateway has strict delay constraints, the packet users must be lightly loaded, or extracted from this node.

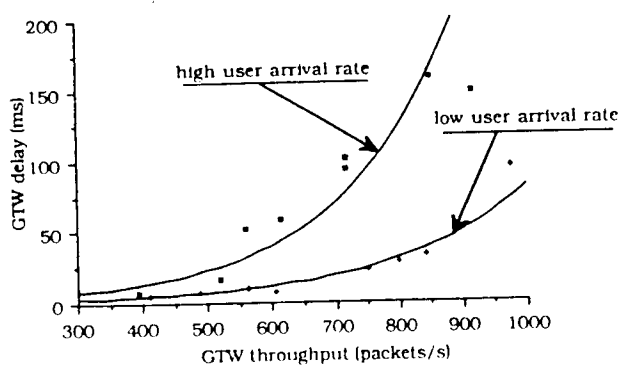


Figure 6. Performance of the Gateway

In Fig.7, which is presented in logarithmic scale, the throughput-delay characteristics of the packet users are presented in three cases:

- the Gateway is not connected to the VMEbus,
- the Gateway has a low packet arrival rate, and
- the Gateway has a high packet arrival rate.

The impact of the Gateway to the packet users is

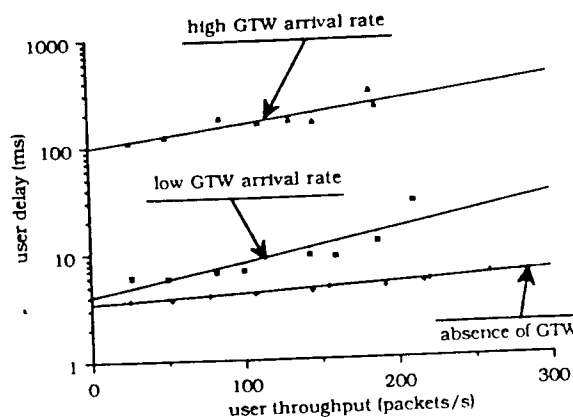


Figure 7. Performance of the packet users

clearly very high, especially in the case that the Gateway has a heavy load.

V. Conclusions

The implementation of an ETHERNET to High-Speed LAN Gateway is presented. The hardware architecture, in conjunction with a Real-Time Operating System and a specially developed Scheduler, gives the required system performance for protocol conversion and fast system response.

References

- [1] A.Luvison, G.Roullet, F.Toft: "The LION project: A status report", in Proc. 4th Annu. ESPRIT Conf., Brussels, Belgium, Sept. 1987, pp. 1477-1489.
- [2] V.Pallios, T.Antonakopoulos, V.Makios: "Architecture and Implementation of the Access Mechanism for a Bus-Structured Multiservice LAN", the 8th International Conference on Distributed Computing Systems, IEEE, San Jose California, June 1988.
- [3] V.Pallios, T.Antonakopoulos, V.Makios: "Access Control Module for Local Integrated Optical Network", Electronics Letters, vol. 25, no. 3, pp.183-185, 1989.
- [4] TITN: "General Specifications IP and TR4 Products", ESPRIT project 169, Internal Report, March 1988.
- [5] Motorola: "VERSADOS Overview", M68KOVER/D7, 1986.