

Link Division Multiplexing (LDM) for Network-on-Chip Links

Arkadiy Morgenshtein, Avinoam Kolodny, Ran Ginosar

Electrical Engineering Department, Technion – Israel Institute of Technology, Haifa, 32000, Israel

[arkadiy@tx.technion.ac.il]

Abstract - Large Systems-on-Chip (SoC) can employ packet-switched Networks on-Chip with Quality-of-Service (QNoC) architecture. Communication in QNoC links typically involves Time Division Multiplexing (TDM). In this paper we propose the Link Division Multiplexing (LDM) technique based on optimal division of link wires among the data blocks of various applications and QoS levels that are transmitted through the same physical link. LDM allows simultaneous data transport in various QoS levels with full utilization of the link resources and elimination of timing dependency between QoS levels. Simulations comparing LDM with TDM were performed for routers with four clients with various probability profiles. The rate of flits transportation is increased by up to 50% in LDM.

I. INTRODUCTION

Networks-on-Chip (NoC) were proposed as interconnection approach in large Systems-on-Chip (SoC) [1][2][3]. NoC is based on packet switching and allows design modularity and high level of abstraction in architectural modeling of the systems. NoCs are also shown to be very attractive solutions for assuring Quality of Service (QoS) on chip communication [4][5]. A generic QoS based NoC architecture termed QNoC was proposed in [4] providing efficient QoS communication between SoC modules.

The communication in QNoC links is typically performed using time-sharing techniques. Time division Multiplexing (TDM) is used for modulation of the data by dedicating specific time slots for transmission of data with certain QoS level. This technique allows maintaining of priority rules and fulfillment of communication requirement of data at all QoS levels. However, this technique does not fully utilize the bandwidth of the link, while dedicating all link resources to a single data source at each time, regardless of its throughput requirements and size.

In this paper we propose to modulate the data by dedicating specific wires of a link for certain QoS levels, defining the Link Division Multiplexing (LDM). LDM modulation is based on optimal division of link wires among the data blocks of various applications and QoS levels that are transmitted through the same physical link. LDM allows simultaneous data transport in various QoS levels with full utilization of the link

resources and elimination of timing dependency of lower QoS levels on higher levels. Application of LDM results in significant increase in the data transport rate and can contribute to reduction of design area and power consumption in QNoC.

The paper is composed of the following sections. The Link Division Multiplexing concept is described in Section II. Section III presents the architecture of LDM. LDM application to variety of communication scenarios are shown in Section IV and discussed in Section V. The work is summarized in Section VI.

II. LDM CONCEPT

The concept of LDM can be described using gradual evolution scheme shown in Fig. 1. Typical communication model of existing on-chip links is based on parallel link with m wires in which the data flow is modulated by Time Division Multiplexing (TDM). In this manner, at each given time slot all the wires of the link are dedicated to transmission of data from a single source. The sources of transmission change at each time slot as shown in Fig. 1a. Certain priority rules can be applied (like in QoS) by defining the order and the duration of the period allocated to each source.

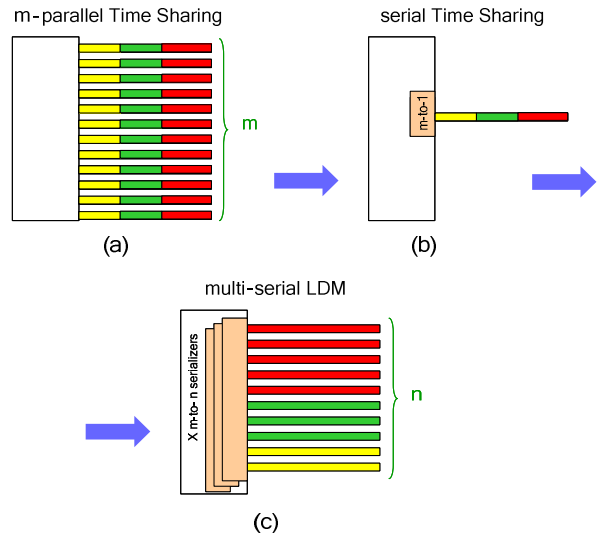


Fig. 1. Evolution from parallel link with TDM (a), through serial link with TDM (b), towards multi-serial link with LDM (c).

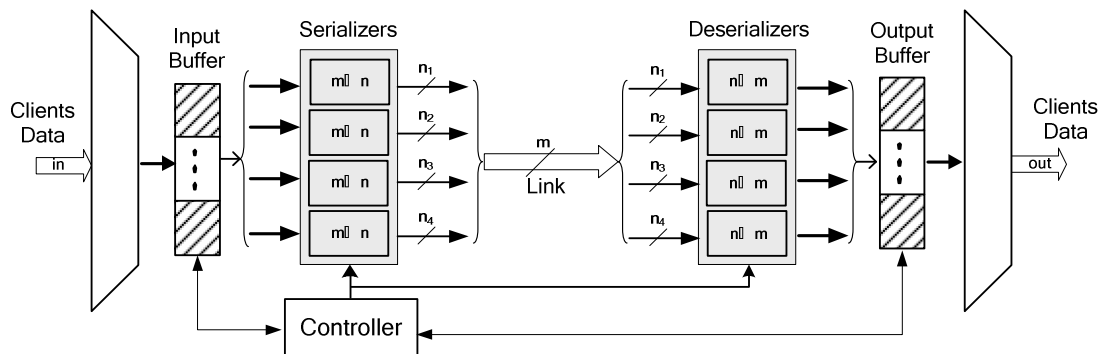


Fig. 2. Block diagram of ABC transmitter.

As was presented in [6], serialization can be performed in order to obtain more efficient link in terms of power and area. In case of serial link TDM modulation can be performed in the same way as in the parallel (Fig. 1b). However, the concept of serialization can be extended from a single-wire serial link to more general case of link with tunable number of wires. This can be performed by m -to- n serializers which are capable of transforming the m -bit parallel data to blocks of length from l (serial) to n bit (named multi-serial).

Combination of several m -to- n serializers into a common link interface allows implementation of Link Division Multiplexing (LDM) as shown in Fig. 1c. In this example the n -bit link is created by selective connection of outputs of several m -to- n serializers. The output width of each serializer can be controlled in a way that the total number of the wires will be m . Each serializer is dedicated to transmission of data in certain QoS level and the number of wires of each level is allocated according to the predefined priority. This technique allows simultaneous transport of data in various QoS levels.

Note that LDM allows full utilization of the link resources while in parallel TDM the link is not necessarily fully utilized due to different size formats of various QoS levels. LDM also solves the timing dependency of lower QoS levels on higher levels. Now the transmission is performed simultaneously while maintaining the demands of relative throughput and latency of various QoS levels.

III. LDM ARCHITECTURE

The architecture of LDM is presented in Fig. 2. The structure comprises the link, the input and output interfaces and the LDM controller.

The data from clients of the router reaches the input interface and is stored in the input buffer. Each data packet contains tags with its destination address and the QoS level. The packets with different QoS levels differ by their lengths, priority, latency and throughput

requirements. In this work we distinguish between four service levels as proposed in [6]:

1. *Signaling* covers urgent messages and very short packets that are given the highest priority in the network to assure shortest latency. This service level represents interrupts and control signals and alleviates the need for dedicated wires.
2. *Real-Time* service level guarantees bandwidth and latency to real-time applications, such as streamed audio and video processing.
3. *Read/Write (RD/WR)* service level provides bus semantics and is designed to support short memory and register accesses.
4. *Block-Transfer* service level is used for the transfer of long messages and blocks of data, such as cache refill and DMA transfers.

A priority ranking is established among these service levels, where Signaling is given the highest priority and Block-Transfer the lowest. Additional service levels may be defined if desired.

The QoS tags of the packets are accessed by the LDM controller while the data is in the buffer. The controller allocates the suitable number of wires in the link in order to fulfill the throughput requirements of the transmitted packets according to the QoS levels. This is performed by setting the proper serialization rate of each of the serializers. Note, that the total number of wires used for data transport is always equal to m in order to maintain the maximal utilization of the link resources.

After each wire of the link is designated to a certain packet, the information about the wires allocation is transmitted to the receiver. The controller adjusts the settings of the deserializer and the output buffer so that the data could be correctly reconstructed and stored.

There are several alternatives that can be used for control implementation in LDM system:

1. The control data can be added to the transmitted packet, so that each wire will carry the information about the packet to which it is designated. This may reduce the hardware and

wiring overhead needed for control, but will cause a penalty in terms of efficiency rate of the transported data.

2. The wires allocation can be performed similarly in the transmitter and the receiver without communication between the components, by predefining the allocation patterns according to operation mode or application type in the related QNoC node. This technique, however, reduces the flexibility of the system and does not allow maximal utilization of the link for communication scenarios that are different from the predefined ones.
3. The controller can communicate to both transmitter and receiver by implementing few additional wires dedicated to transportation of control data. In this manner the efficiency rate of the data is maintained together with the flexibility of operation.

In this work we adopt the third option of LDM controller implementation.

The differences between the architecture of regular TDM router and the LDM router in QNoC can be seen in details in Fig. 3. In the TDM (Fig. 3a), the data arriving to the router from each client is classified according to the QoS level and packets of a certain QoS level are stored in dedicated buffers together with other packets of the same level. In this way, the packets of different levels are stored and treated separately by the router. The transmission of the packets is controlled by the scheduler that sets the indexes of the MUX so that the predefined time-sharing protocol will be maintained. All the wires of the link are allocated to the transmission of the packet regardless to its QoS level. The QoS levels can set the priority and the time slot duration for packet transmission, but not the wires allocation.

In the LDM router (Fig. 3b), the number of buffers can be reduced due to the fact that the packets of different QoS levels can be transported simultaneously and there is no need for separate storing. After the data is received and stored in the buffer, it is accessed and identified by the controller. The controller performs the allocation of wires in the link, by setting the proper serialization rate so that the communication requirements of the packets will be fulfilled. Same hierarchy of priority and throughput requirements is maintained here for QoS levels, but it is implemented by simultaneous wires allocation instead of time sharing.

Conditions can be defined for effective wires allocation in order to maintain the maximal utilization of the link by means of simple LDM control:

- a. The sum of allocated wires n_i is equal to the total number of wires in the link m :

$$m = \sum_{i=1}^k n_i \quad (1)$$

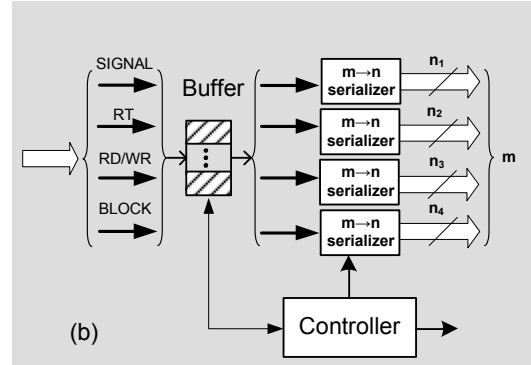
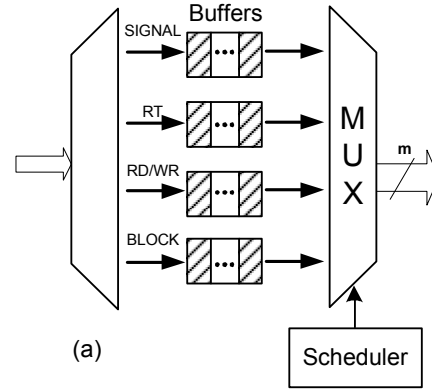


Fig. 3. Architecture of QNoC router with TDM (a) and LDM (b) link interface. Fewer buffers are needed in LDM.

- b. The wires allocation is proportional to the throughput requirements defined by the QoS level:

$$\frac{n_i}{n_j} = \frac{Thr_i}{Thr_j} \quad (2)$$

In order to simplify the control design, the wires allocation was limited to even integer values. In this work we implement the system with 32 bit link and four clients. Thus, there are only two possible patterns of wires allocation - for equal requirements for all clients the allocation is {8,8,8,8}; for different requirements the allocation is {16,8,4,4}. The number of possible patterns and the allocation can be changed according to the number of wires in the link and according to the LDM control implementation.

IV. RESULTS

The architecture and the communication environment of LDM in QNoC link were implemented and emulated using Matlab. The implementation included a 32-bit link between two QNoC routers with four clients each. Four QoS levels were used in the data generation. For each QoS level, the following parameters were defined:

- Size of the packet in every QoS level defined in flits with basic length of 32 bit.

distribution type	scenario	Client	QoS probability					Flits transmitted	
			$P_{no-data}$	$P_{signaling}$	$P_{real-time}$	$P_{read/write}$	$P_{block-trans}$	LDM	TDM
homogeneous	A	all	0.30	0.25	0.05	0.25	0.15	99806	99876
	B	all	0.995	0.001	0.001	0.001	0.002	89983	89963
heterogeneous	C	C ₁	0.1	0.1	0.2	0.2	0.4	55890	39792
		C ₂	0.994	0	0.001	0.005	0		
		C ₃	0.994	0	0.001	0.005	0		
		C ₄	0.994	0	0.001	0.005	0		
	D	C ₁	0.993	0.001	0.001	0.004	0.001	89983	87572
		C ₂	0.99	0.006	0.002	0.001	0.001		
		C ₃	0.3	0.3	0.05	0.05	0.3		
		C ₄	0.99	0.001	0.004	0.004	0.001		

Table 1. Results of LDM effectiveness evaluation.

- *Probability* of appearance of the packet in certain QoS level. The probability is relative and given for each QoS level in every clock cycle, together with the probability of no data appearance.
- *Delay* was defined for each QoS level in order to express the processing time needed by each of the packets before transmission in the actual system. The times are different for various QoS levels, and bigger packets are naturally having the longer delay.

The probabilities of data appearance were used in order to build a profile of each client communicating with the router. Each client was described by a set of five probabilities – one for no data and four for the different QoS levels.

At the first stage, the effectiveness of LDM technique was evaluated as function of several distribution scenarios. Four simulations were performed, each containing data generation and transportation during 100,000 clock cycles. The simulation scenarios were divided into two types: homogeneous – where all the clients have the same QoS probability profile, and heterogeneous – where the probability profiles of the clients are different. The effectiveness of LDM was measured by the number of flits that were transported in the link during the simulation. The simulation setups and the results comparing LDM with TDM are presented in Table 1.

As can be seen, in case of the homogeneous scenarios, the LDM shows similar results to TDM. This result is expectable since in case of homogeneous scenario, none of the clients have outstanding communication requirements that could significantly influence the wires allocation. The effectiveness of LDM is manifested in the heterogeneous cases. In scenario D client C₃ has relatively high probabilities of data appearance and the application of LDM results in 3% increase in number of transported flits. Higher improvement is achieved in scenario C, where in addition to a dominant client C₁, the probabilities of signaling and block transfer in other

clients are equal to zero. In this case, the increase in number of transported flits by LDM is more than 40%.

At the second stage of simulations, the effectiveness of LDM was evaluated as function of the delay of the packets before transmission. In this case, the simulations of the heterogeneous scenario C were repeated for values of delay varying between 10% and 200% of the default delay used in the first stage. The results of the simulations are shown in Fig. 4.

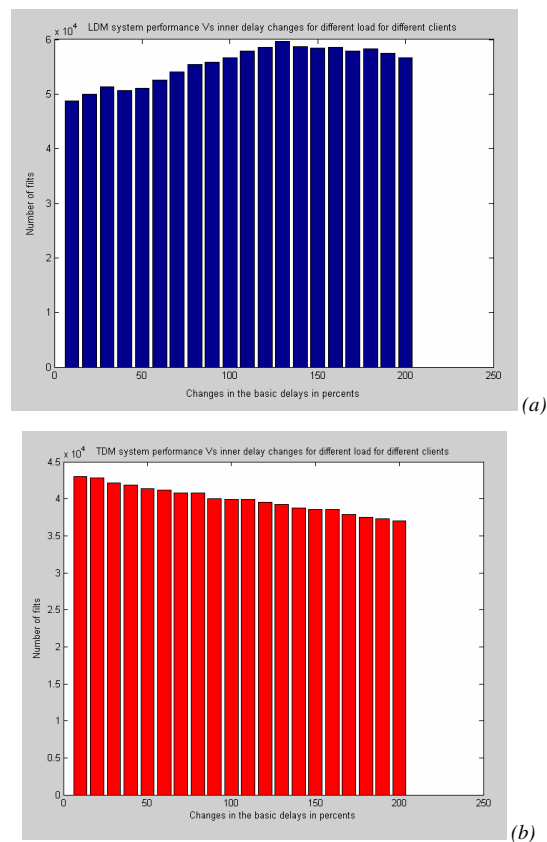


Fig. 4. Performance comparison of LDM (a) and TDM (b) for various values of packet delay.

The number of flits transported by LDM is higher than in TDM for all delay values. Note the differences in the behavior of the techniques as function of delay increase. The TDM link shows a constant reduction of the transported flits as the delay raises. This is explained by the fact that the delays in packet transmission are reducing the effective time slot dedicated for each packet. The behavior of LDM link is different and a maximum value can be observed for certain delay. This can be explained by the fact that for low delays the operation is suboptimal and there is a queue of data in the buffer, while for higher delays the number of the transported flits reduces similarly to TDM. The increase of number of transported flits in LDM as compared to TDM is up to 50%.

V. DISCUSSION

The LDM technique proposed in this work allows a significant increase in link utilization. The results showing up to 50% increase in number of transmitted flits as compared to TDM can be also interpreted in terms of power efficiency. Better utilization of the link bandwidth allows faster transportation of the data. In NoC communication the data can have a “burst” appearance, and the faster transportation can allow switching the link to a “slip” mode for longer period of time.

In this paper the LDM was presented in 32-bit and with wires allocation limited to two alternatives. The application of the technique to links with more wires should result in further improvement in LDM. Advance in the architecture and the control of LDM should also contribute to increased efficiency of the technique.

The LDM method can be further investigated in order to determine the potential contribution of the technique at various levels of QNoC design. The different control options presented here can be evaluated in terms of timing, power and area consumption. Architectural aspects like reduced number of buffers required in LDM router should be also considered during NoC design.

VI. SUMMARY

Link Division Multiplexing (LDM) technique was proposed in this paper. LDM is applied to packet-switched Networks on-Chip with Quality-of-Service (QNoC) architecture. The LDM technique is based on optimal division of link wires among the data blocks of various applications and QoS levels that are transmitted through the same physical link. LDM allows simultaneous data transport in various QoS levels with full utilization of the link resources and elimination of timing dependency of lower QoS levels on higher levels. Simulations comparing LDM with Time Division Multiplexing (TDM) technique were performed for routers with four clients with various

probability profiles. The simulations show increase in number of transported flits of up to 50% in LDM.

REFERENCES

- [1] W.J. Dally, B. Towles, “Route packets, not wires: on-chip interconnection networks”, DAC, pp. 684-689, 2001.
- [2] John Dielissen, Andrei Radulescu, Kees Goossens, and Edwin Rijpkema, "Concepts and Implementation of the Philips Network-on-Chip". IP-Based SOC Design, Grenoble, November, 2003.
- [3] Luca Benini, Giovanni De Micheli, “Networks on Chips: A New SoC Paradigm”, IEEE Computer, no. 35, vol. 1, pp. 70-78, 2002.
- [4] E. Bolotin, I. Cidon, R. Ginosar and A. Kolodny, “QNoC: QoS architecture and design process for Network on Chip”, Special issue on Networks on Chip, The Journal of Systems Architecture, 50(2-3), pp. 105-128, December 2003
- [5] K. Goossens, J. van Meerbergen, A. Peeters, and P. Wielage, "Networks on Silicon: Combining Best-Effort And Guaranteed Services", DATE 2002, Design automation and test conference, March, 2002.
- [6] A. Morgenshtein, I. Cidon, A. Kolodny, R. Ginosar, “Comparative Analysis of Serial and Parallel Links in Networks-on-Chip”, SoC, Finland, pp. 185-188, 2004.
- [7] E. Bolotin, A. Morgenshtein, I. Cidon, R. Ginosar and A. Kolodny, "Automatic Hardware-Efficient SoC Integration by QoS Network on Chip", ICECS, 2004.