Halmstad University Post-Print

A Fibre-Optic AWG-Based Real-Time Network and Its Applicability to High-Performance Embedded Computing

Annette Böhm, Magnus Jonsson, Kristina Kunert and Hoai Hoang

N.B.: When citing this work, cite the original article.

©2005 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Böhm A, Jonsson M, Kunert K, Hoang H. A Fibre-Optic AWG-Based Real-Time Network and Its Applicability to High-Performance Embedded Computing. In: 19th International Parallel and Distributed Processing Symposium: proceedings. IEEE Computer Society; 2005. p. 133b-133b.

DOI: http://dx.doi.org/10.1109/IPDPS.2005.22
Copyright: IEEE

Post-Print available at: Halmstad University DiVA
http://urn.kb.se/resolve?urn=urn:nbn:se:hh:diva-383

A Fibre-Optic AWG-based Real-Time Network and its Applicability to High-Performance Embedded Computing

Annette Böhm, Magnus Jonsson, Kristina Kunert, and Hoai Hoang
School of Information Science, Computer and Electrical Engineering, Halmstad University
Box 823, S-301 18 Halmstad, Sweden.
Magnus.Jonsson@ide.hh.se, http://www.hh.se/ide

Abstract

In this paper, an architecture and a Medium Access Control (MAC) protocol for a multi-wavelength optical communication network, applicable in short range communication systems like System Area Networks (SANs), are proposed. The main focus lies on guaranteed support for hard and soft real-time traffic. The network is based upon a single-hop star topology with an Arrayed Waveguide Grating (AWG) at its center. Traffic scheduling is centralized in one node (residing together with the AWG in a hub), which communicates through a physical control channel. The AWG's property of spatial wavelength reuse and the combination of fixed-tuned and tunable transceivers in the nodes enable simultaneous control and data transmission. A case study with defined real-time communication requirements in the field of Radar Signal Processing (RSP) was carried out and indicates that the proposed system is very suitable for this kind of application.

1. Introduction

As new applications are developed, the requirements on communication networks continue to increase. Real-time services play a vital role in high-performance networks used in a wide range of different applications, as e.g. cluster computing, radar signal processing or streaming video.

In this paper, we present a single-hop Wavelength Division Multiplexing (WDM) star network based on an Arrayed Waveguide Grating (AWG), and a Medium Access Control (MAC) protocol, which makes use of the special properties of the architecture and the AWG component. The MAC protocol's main function is to handle three different traffic classes: hard real-time, soft real-time and non-real-time traffic. The network is developed for short range communication systems like System Area Networks (SANs) and embedded systems. Possible application areas are cluster computing, distributed large routers and distributed video and imaging applications. In a case study, we took a closer look at the applicability of our system in the field of radar signal processing (RSP).

Due to their advantageous properties, optical interconnects and devices are commonly used in modern communication systems. High bandwidth, low loss and cost, a small diameter and immunity from Electro-Magnetic Interference (EMI) are some of the reasons that give optical architectures an advantage over those merely built upon electronics. Extensive research has been done in the field of single-hop WDM star networks, where broadcast-and-select networks based on the Passive Star Coupler (PSC) were the center of attention [1]. Opposed to the PSC, the AWG is a wavelength-sensitive device that makes it possible to reach selected parts of the network and increase its concurrency and throughput. Nevertheless, research on AWG-based network architectures is limited. A comparison of the PSC and the AWG is presented in [2], while the network proposed in [3] combines the two components in a hybrid solution. AWG-based single-hop networks are presented in [4] and [5], where the latter one focuses on the development of a MAC protocol. A common feature of the related works mentioned, is the lack of real-time support. In our proposed system, each node has its individual control channel. We make use of the AWG's property of spatial wavelength reuse to maximize the concurrency of both control and data traffic, and time-slotted access to enable collision-free transmissions. The high level of concurrency can be compared to control channel-based WDM star networks, where the transmission of control packets is serialized instead [6] [7]. Together with the centralized scheduling algorithm, this architecture enables support of real-time traffic with the possibility of both delay and throughput guarantee.

By carrying out a case study in the area of radar signal processing, we were able to verify the feasibility of the proposed system. The analysis of the throughput, delay and deadline-miss-ratio for defined requirements showed the network's suitability for applications with heterogeneous real-time communication requirements.

Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05)
1530-2075/05 $ 20.00 **IEEE**

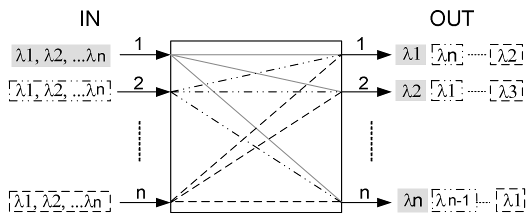

Figure 1. Wavelength routing pattern of an $N \times N$ AWG.

The rest of the paper is organized as follows: The AWG-component and the proposed network architecture are described in chapter 2, while the MAC-protocol and its scheduling algorithm are explained in chapter 3. Our solution is evaluated in a case study, which is discussed in chapter 4 and conclusions are drawn in chapter 5.

2. Network Architecture

2.1. The Arrayed Waveguide Grating

The optical component AWG, also known as Wavelength Grating Router (WGR) or Phased Array (PHASAR) [8], consists of an arrangement of optical waveguides on a substrate plate, which gives it special spatial and spectral properties for highly effective future photonic communication networks. AWGs can be used as wavelength multiplexers or demultiplexers or to implement WDM equipment such as Optical Cross-Connects (OXCs) or Optical Add-Drop Multiplexers (OADMs) [5]. In recent years, AWGs have been more frequently used as passive wavelength routing devices in optical communication networks; this application of the AWG is applied in the network proposed in this paper.

An $N \times N$ AWG has N input ports (input waveguides) and N output ports (output waveguides). Offering full connectivity, it can simultaneously accept N wavelengths at each input port and route each one of them to a specific output port without collision (Figure 1), which makes AWGs strictly nonblocking devices [8]. In order to be able to send and receive on all possible wavelengths, the nodes attached to the AWG need tunable transmitters and tunable receivers at the input and output ports, respectively (or, alternatively, arrays of fixed-tuned transmitters and receivers).

In order to make multiple communication channels possible, the AWGs use spatial wavelength reuse (Figure 1), which means that every wavelength can be used on all output ports at the same time without collision. This kind of routing makes AWGs extremely efficient and improves the throughput and latency behavior of optical networks [2].

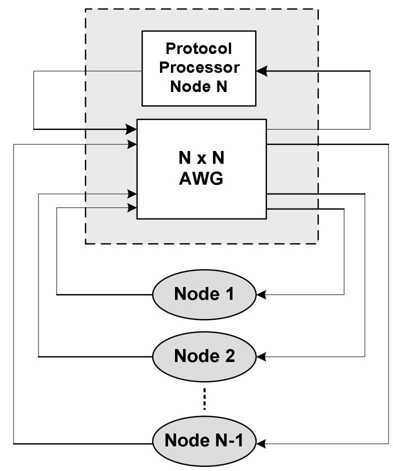

Figure 2. The network architecture.

2.2. The architecture

The proposed network architecture (Figure 2) consists of one $N \times N$ AWG, integrated in a hub with one node used for the centralized scheduling (further on denoted as "protocol processor") and $N - 1$ end nodes. The AWG component is the center of the physical WDM star network, and all nodes are connected to the AWG by two fibers, one for transmission and one for reception.

The network is based upon the principle of CC-TT-TR-FT-FR (Control Channel - Tunable Transmitter - Tunable Receiver - Fixed Transmitter - Fixed Receiver) [1]. The fixed receiver and transmitter are used to establish physical control channels between the end nodes and the protocol processor. All $N - 1$ fixed transmitters will send on different wavelengths and the fixed receivers will be tuned correspondingly, i.e. a node's fixed transmitter will send on the same wavelength as its fixed receiver receives on. On the control channels, the protocol processor receives all information from the nodes that it needs to run its scheduling algorithm, and sends out control information back to them. This means that all intelligence about real-time traffic scheduling is embedded in the protocol processor, while the end nodes and the AWG component are completely passive in this respect. Of the N possible wavelengths out from each node, one is always used to send control information as described above. In addition to the control wavelength, one of the remaining $N - 1$ wavelengths can, in each instance, be used for transmission of a data packet. The AWG routes the packet to the correct destination node based on the chosen wavelength.

The protocol processor transmits and receives data via wavelength array components, while the other nodes are

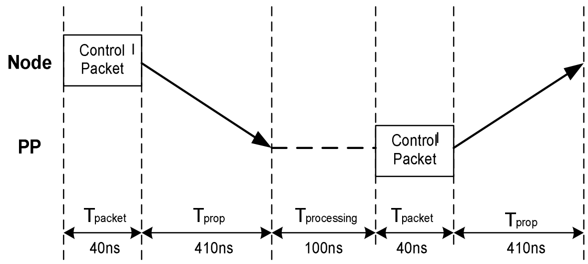

Figure 3. Control packet propagation.

equipped with one tunable transmitter and receiver and one fixed tuned transmitter and receiver. The array component on the transmitting side of the protocol processor is attached to an $(N-1) \times 1$ combiner, which demultiplexes the $N-1$ different wavelengths onto one fiber. Correspondingly, a $1 \times (N-1)$ splitter separates incoming wavelengths before they enter the $N-1$ array of receivers. The fixed tuned and tunable transmitters on the remaining $N-1$ end nodes are connected to a 2×1 combiner in each node, that demultiplexes the two wavelengths (one fixed wavelength to the protocol processor and the tunable wavelength for data transmission to the nodes) onto one fiber. The corresponding construction on the receiving side of a node is a 1×2 splitter that is connected to one fiber with incoming traffic.

3. Medium Access Control Protocol

As the nodes do not have global knowledge of which other nodes that want to send in which time slot and on which wavelength, the control processor has to employ a MAC protocol and organize the traffic in a centralized fashion. The main task of the MAC protocol is to delegate the available wavelengths between the sending nodes based on control information, and to make sure that the hard real-time messages with the shortest deadlines are sent first. As our system is developed for a SAN, the fibers interconnecting the end nodes and the protocol processor are assumed to have a maximum length of a few tens of meters. In other words, the propagation delay is assumed to be a fraction of a time slot.

3.1. The control traffic

Each node has two different message queues, one for hard real-time messages and one for soft and non-real-time traffic; non-real-time is defined as soft real-time traffic with infinite deadlines. The node sorts its two queues by deadline, and sends out a control message to the protocol processor, requesting permission to send the message with the shortest deadline (where the hard real-time queue always gets priority over the soft real-time queue). The control message contains information about source node, destination node, deadline, and traffic class for the packet in question.

The traffic is organized in time slots, where each slot has the length of the transmission time of one data packet, measured from when the first bit of the data packet leaves the source node, to when the last bit reaches the destination node. As a control packet is simple and contains little information and short distances are assumed, one slot can be used to send control traffic to the protocol processor, to run the scheduling algorithm and to send control messages back to the nodes. At the end of the control slot, the nodes tune their respective transmitters and receivers according to the control information.

For a 64×64 AWG, the length of a control packet (T_{packet}) to the PP is less than 100 bits (6 bits to identify the destination, 20 bits to state the deadline and some bits of overhead), i.e. about 40 ns, assuming a bit rate of 2.5 Gbit/s. A control message from the PP back to the node is even shorter (a flag bit to tell whether the request to send data is accepted or denied and, if the node can expect incoming traffic, 6 bits to identify the sending node). Assuming a bit rate of 2.5 Gbit/s, a 100 ns processing time ($T_{processing}$) at the PP and defining one slot (T_{slot}) as 1 μs, i.e. 2500 bits, leads to the following calculation of the propagation time, T_{prop}:

$$T_{prop} = 0.5 \cdot (T_{slot} - 2 \cdot T_{packet} - T_{processing}) = 410\text{ns} \qquad (1)$$

In other words, a slot of 1 μs leaves enough communication time to support a fiber length of 82 m between the PP and each regular node. See Figure 3 for details.

Data transmissions take place in the following time slot. Due to the fixed tuned and individual wavelength of the control channel, it is possible to handle data and control traffic in parallel. The AWG allows control traffic from all the nodes to be sent to the protocol processor simultaneously. Therefore, the medium access control does not result in more than one single slot of delay and does not have any impact on the data traffic flow. This principle is illustrated in Figure 4.

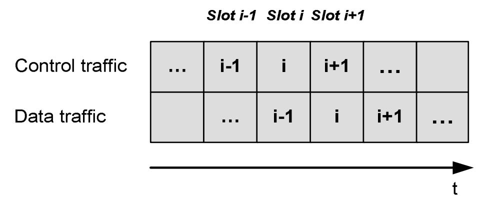

Figure 4. Control packet propagation.

Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05)
1530-2075/05 $ 20.00 **IEEE**

As each node has its own control channel, clock synchronization on the packet level is not required. Control messages from the protocol processor to the end nodes are sent out simultaneously and as interconnection distances in our network are assumed to be short, the propagation delay due to dispersion is negligible. Hence, all end nodes are synchronized on incoming control packets and further synchronization is not needed. (A discussion of timing and dispersion in WDM star networks can be found in [9].)

3.2. The scheduling algorithm

After having sorted their two queues according to the EDF (Earliest Deadline First) algorithm, each node has to give priority to the first packet in one of the queues. As long as there are packets queued in the hard real-time queue, the soft real-time traffic has to wait. This guarantees that the hard real-time traffic does not get disturbed or compromised in any situation.

The protocol processor is responsible for accepting or denying a node's request to send. To avoid receiver collisions, there must be no more than one transmission to a single destination in any given time slot and an approach to selecting the right set of transmissions is proposed below. When the protocol processor has received all the control messages, it sorts them by deadline, checks them one by one and determines whether the requested transmission can be accepted or not. As soon as a transmission to a certain destination node is accepted, any other request for communication with this destination in the next slot is denied. When the algorithm has found the right set of packets, the protocol processor sends out an individual message to each node (but all messages simultaneously) to let it know whether it is allowed to send and if it will receive traffic (and from whom) in the next time-slot.

3.3. Feasibility analysis

In order to determine the performance characteristics for hard real-time traffic, a deterministic analysis of throughput guarantee, delay and latency is given below. Even when all the nodes want to send data to the same destination node in any given time slot, there is always at least one packet (the one with the shortest deadline) that will be guaranteed access to the medium, i.e. a capacity of 1 can be guaranteed for hard real-time traffic.

One can look upon the channel as a periodic task, while the network would constitute a CPU or processing system (from a scheduling point of view). The capacity, C_i, would be the worst-case-execution-time (WCET) for the task (the logical channel with index i, where $d_i \leq t$. Before we describe the feasibility test, we make the following definitions:

Utilization: According to basic EDF theory [10] the ***utilization*** of periodic real-time traffic is defined as

$$U = \Sigma C_i / P_i \ . \tag{2}$$

Hyperperiod: The ***Hyperperiod*** for a set of periodic tasks is defined as the length of time from when all tasks' periods start at the same time, until they start at the same time again.

BusyPeriod: A ***BusyPeriod*** is any interval of time in which a link is not idle.

Workload function**:* ***h(t) is the sum of all the capacities of the tasks with absolute deadline less than or equal to t, where t is the number of timeslots elapsed from the start of the Hyperperiod. It is calculated as follows:

$$h(t) = \Sigma [1 + ((t - d_i) / P_i)] C_i \ . \tag{3}$$

A ***feasible system state*** is a system state with all logical channels in the system being feasible.

Following the discussion from above, and the new definitions, testing if a new logical channel can be added is therefore equivalent to testing if the new state is still feasible, given that the new logical channel has been added. The feasibility test is done in two steps, each step being a test of its own.

First Constraint: The utilization of the link has to be less than or equal to one (100%), following the discussion above about guaranteed access to the medium.

Second Constraint: For all values of t, the workload function $h(t)$ has to be less or equal to t.

The second constraint, in the form given above, does not lend itself out particularly well to computation. It is shown in [11] how to reduce the time and memory complexity of the second constraint check. If $h(t) \leq t$ in the first busy period of the Hyperperiod in the supposed schedule to come, then $h(t) \leq t$ for all t. The following upperbound of the interval to be checked is therefore an improvement of the algorithm above:

$$t, \text{ such that } 1 \leq t \leq BusyPeriod \tag{4}$$

where *BusyPeriod* is the first BusyPeriod in the schedule at the start of the Hyperperiod. Furthermore, one needs not check every integer from the first timeslot, but only the integers t where

$$t \in \bigcup_{i=1} \{mP_i + d_i : m = 0,1,2...\} \tag{5}$$

4. Case Study

To be able to verify the feasibility of the proposed network and protocol, we evaluated the system according to specific application requirements. A simulation based on a radar signal processing (RSP) case was carried out and delay, throughput and deadline-miss-ratio for the network were analysed. RSP is an application area requiring system area networks with support for heterogeneous real-time services. With its three traffic classes, the proposed AWG-based network suits this purpose very well.

4.1. Case Definition

In [12], Bergenhem et al. describe a full case definition for a RSP case scenario. Our simulations are inspired by the "straight pipeline" case, one of three possible alternatives explained in [12]. Due to its clear traffic pattern, this case simplifies analysis and understanding of the results, while still providing three different traffic classes.

An architecture with a 16 x 16 AWG at its centre, consisting of one PP and fifteen communicating nodes for the RSP system, was chosen for the simulation. One node serves as a master node, while the remaining fourteen slave nodes are used for the pipelined data flow (Figure 5). We further assume periodic traffic and a bit rate of 2.5 Gbit/s and a time slot of 1 µs (one slot corresponds to one data packet of 2500 bits). The traffic in the RSP case definition consists of three main types: control traffic, which flows in two directions between the master node and each node in the straight pipeline; data traffic from node to node in the pipeline; and other traffic, such as logging or long term statistics from the pipeline nodes to the master node. These three traffic types and their deadline requirements are conveniently represented by the three traffic classes in our protocol: hard real-time (HRT), soft real-time (SRT) and non real-time (NRT) traffic. The delay bound for the three traffic classes is equal to their individual period time.

4.2. Simulation results

In this chapter, the results of the main simulations are presented. We chose to analyse throughput, average delay and deadline-miss-ratio of the different traffic classes, by using fixed, predefined parameters or by varying the amount of SRT or HRT traffic.

Table 1 shows the results of running the simulation for a fixed set of parameters, which resemble possible RSP application data. The HRT traffic of the master-slave communication is periodic with a period of 100 µs, i.e.100 slots and a default payload of 2500 bits, i.e. 1 slot, while the SRT traffic has a period of 5 ms, i.e. 5000 slots and a payload of 4000 slots. NRT traffic has a period of 5000 slots and a payload of 950 slots, a value chosen to saturate the network. The traffic class parameters are listed in Table 2. All traffic that a channel will send during one period is generated and queued for sending at the start of the period. There is no smoothing of the incoming traffic over the whole period. In real RSP applications, the radar data transmissions are divided into smaller sets instead. As seen in Table2, no deadlines of the real-time traffic are missed.

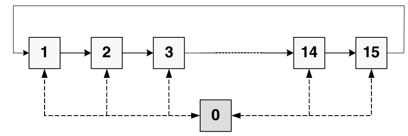

Figure 5. Master node and straight pipeline of slave nodes according to RSP case definition

Table 1. Throughput, average delay and deadline-miss-ratio at fixed traffic loads

	Throughput [packets/slot]	**Delay** [slots/packet]	**Deadline-miss-ratio**
Total *	12.71	-	-
HRT *	0.28	4.00	0.00
SRT *	11.20	2057.00	0.00
NRT *	1.23	6225.80	-
Max HRT**	0.28	4.64	0.00
Max SRT***	11.20	2000.5	0.00

*) Using the traffic load stated above
) Hard real-time traffic only *) Soft real-time traffic only

Table 2. Traffic type parameter

Traff. Type	**Traff. Class**	**Comm. Pattern**	**Period** [slots]	**Payl.** [pack.]	**Delay Bound**
Contr.	HRT	Master/ Slave	100	1	100
Data	SRT	Straight Pipel.	5000	4000	5000
Other	NRT	Many-to-one	5000	950	None

Simulating HRT traffic only, leads to a throughput of 0.28 HRT packets per slot, i.e. 100% as can be concluded from the 0.00 deadline-miss-ratio. Solely SRT traffic leads to a 100% throughput of 11.20 packets per slot. Using all three traffic classes with the parameters stated above, leads to a total throughput of 12.71 packets per slot, which comprises of 0.28 HRT, 11.2 SRT and 1.23 NRT packets per slot. This indicates that, in spite of the three different traffic types in the system, a throughput of 100% for the HRT and SRT traffic can be reached. (As shown earlier in chapter 3.3., a throughput guarantee of one packet per slot can be given for HRT traffic, and therefore its deadline-miss-ratio at the simulated traffic loads will always be 0.)

Figures 6 to 11 show throughput, average delay and deadline-miss-ratio when changing the traffic load in the system. The curves are attained by running the simulator for 20000 time slots with a new set of parameters for each data point in the curve. The statistical results are computed from slot 5000 and forward. Periodic traffic channels in the simulator are treated as follows.

Figures 6 to 8 indicate the system's behaviour when varying the SRT traffic from 500 to 5000 packets per channel with a period of 5000 slots, while the HRT traffic is kept at a constant level of 1 data packet and a period of 100 slots. A constant amount of non real-time traffic (950 packets with a period of 5000 slots) is introduced to make use of the bandwidth that is not needed by the two real-time traffic classes.

The throughput of the three traffic classes, together with the total throughput, is shown in Figure 6. As the intensity of the SRT traffic increases, the throughput of this traffic class reaches 13.62 packets per slot at a traffic intensity of 5000 packets per period. The throughput of the HRT traffic remains constant at 0.28 packets per slot. Due to its high priority, it is not affected by the increased SRT traffic intensity or the NRT traffic in the system. The NRT traffic, on the other hand is starved as the network starts to get saturated at an SRT traffic intensity of about 3000 packets per 5000 slots. Figure 7 shows the average delay of the three traffic classes. As the traffic load in the system increases, the SRT traffic experiences a slight increase in delay, while the HRT traffic remains at a delay of 4.64 slots per packet throughout the simulation. Because of its low priority, the NRT traffic has to wait for the real-time traffic classes to be sent, which increases its delay considerably. The break in the NRT graph's gradient can be explained by irregular behaviour after the saturation of the network. Figure 8 verifies that the HRT traffic meets all its deadline requirements. In spite of the increasing SRT traffic load, the deadline-miss-ratio of the HRT traffic remains 0.00, while the deadline-miss-ratio of the SRT traffic increases slightly as its intensity gets close to 5000 packets per period.

In the second simulation setup, the HRT traffic is varied between 1 and 16 data packets per channel with a period of 100 slots. This time, the SRT traffic is constant at 4000 packets per period of 5000 slots and the NRT traffic load is the same as in the previous simulation.

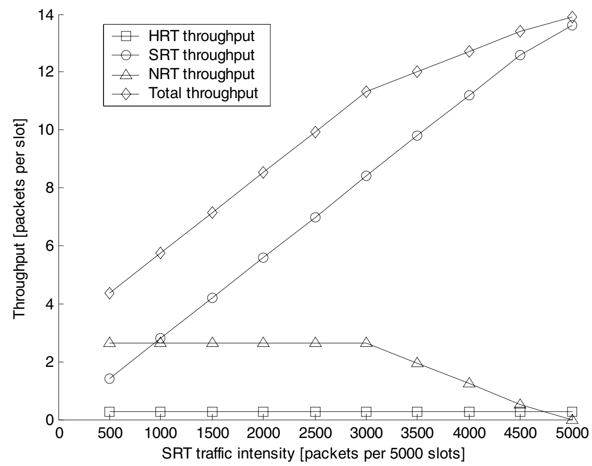

Figure 6. Throughput of the different traffic classes and total throughput against SRT traffic intensity.

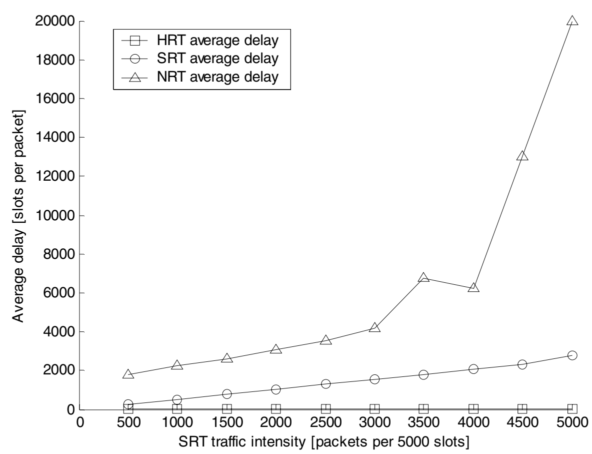

Figure 7. Average delay of the different traffic classes against SRT traffic intensity.

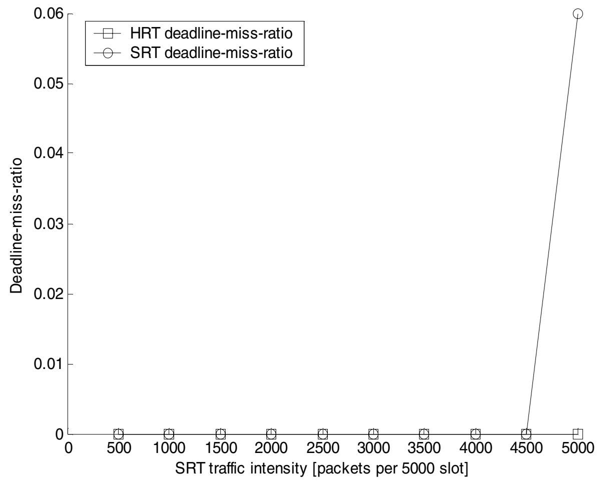

Figure 8. Deadline-miss-ratio of HRT and SRT traffic against SRT traffic intensity.

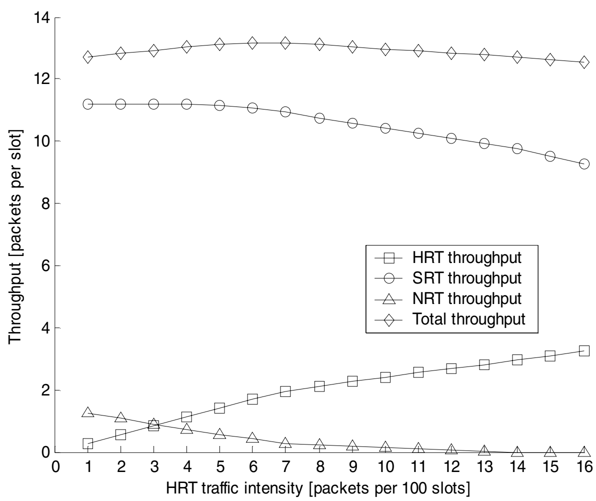

Figure 9. Throughput of the different traffic classes and total throughput against HRT traffic intensity.

Figure 9 shows how varying the HRT traffic intensity influences the three traffic classes in terms of throughput. An increasing HRT traffic load leads to steady increase in HRT throughput, while the SRT and the NRT traffic experience a slight decrease in throughput due to their lower priority. The total throughput reaches its maximum value at about 13.2 packets per slot at a HRT traffic intensity of 7 packets per 100 slots. The behaviour of the delay in Figure 10 can be explained accordingly.

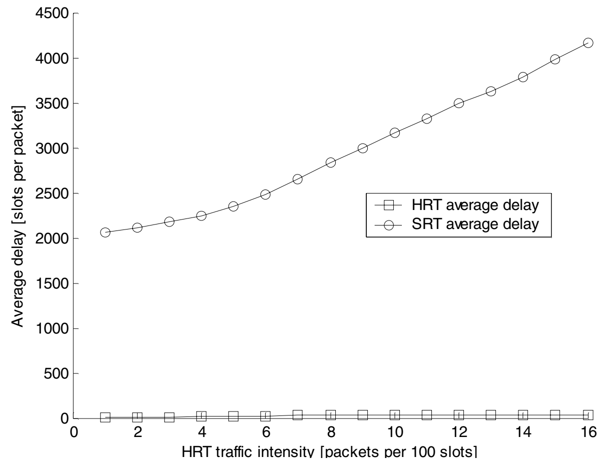

Figure 10. Average delay of the different traffic classes against HRT traffic intensity.

With an increased HRT traffic intensity, the traffic class with the highest priority, HRT traffic, is affected the least, while the remaining two traffic types experience an increase in delay. The average delay for the NRT traffic is not plotted in the figure, as the network already is saturated with NRT traffic, which would lead to irregularities in the curve. The deadline-miss-ratio of the HRT and the SRT traffic are not affected by the increase in traffic intensity and delay and remain at 0.00 throughout the simulation (not shown in a figure). This shows that the HRT and SRT capacity of the network is not challenged by the imposed traffic intensities.

5. Conclusion

In this paper, an AWG-based, single-hop star architecture, together with a MAC protocol that handles hard, soft, and non-real-time traffic, have been proposed. In the network, one node acts as a protocol processor and schedules all the traffic over the network. Using the AWG component enables all the nodes to send control information simultaneously to the processor without any collision, while, at the same time, data traffic can be sent between the nodes. This level of concurrency gives short delays and makes efficient use of the bandwidth and a throughput guarantee for hard real-time traffic is given and proven with EDF-analysis.

A case study in the field of radar signal processing indicated that the proposed system is very suitable for applications with defined heterogeneous real-time communication requirements.

6. Acknowledgement

We are grateful to Anders Åhlander at Halmstad University for sharing his insight on radar signal processing applications.

7. References

[1] B. Mukherjee, "WDM-based local lightwave networks, part I: single-hop systems", *IEEE Network*, vol. 6, no. 3, pp. 12-27, 1992.

[2] M. Maier and A. Wolisz, "Demonstrating the potential of arrayed waveguide grating based single-hop WDM networks", *Optical Networks Magazine*, vol. 2, no. 5, pp. 75-85, 2001.

[3] C. Fan, M. Maier, and M. Reisslein, "The AWG||PSC network: A performance enhanced single-hop WDM network with heterogeneous protection", In *Proceedings of INFOCOM 2003*, vol. 3, pp. 2279-2289, San Francisco, CA, USA, 2003.

[4] D. Banerjee, J. Frank, and B. Mukherjee, "Passive optical network architecture based on waveguide grating routers", *IEEE Journal of Selected Areas in Communication*, vol. 16, no. 7, pp. 1040-1050, Sept. 1998.

[5] M. Maier, M. Reisslein, and A. Wolisz, "A hybrid MAC protocol for a metro WDM network using multiple free spectral ranges of an arrayed waveguide grating", *Computer Networks*, vol. 41, no. 4, pp. 407-433, 2003.

[6] J. Diao and P.L. Chu, "Packet rescheduling in WDM star networks with real-time service differentiation", *Journal of Lightwave Technology*, vol. 19, no. 12 , pp. 1818-1828, Dec. 2001.

[7] J.S. Choi, "Performance analysis of a bandwidth guaranteed medium access control protocol for WDM local networks", *Photonic Network Communications*, vol. 6, no. 1, pp. 69-81, July 2003.

[8] M.K. Smit and C. van Dam, "PHASAR-based WDM devices: Principles, design and applications", *IEEE Journal of Selected Topics on Quantum Electronics,* vol. 2, no. 2, pp. 236-250, June 1996.

[9] G. Semaan and P. Humblet, "Timing and dispersion in WDM optical star networks", In *Proceedings of INFOCOM'93*, pp. 573-577, San Francisco, CA, USA, 1993.

[10] C.L. Liu and J.W. Layland, "Scheduling algorithms for multiprogramming in hard real-time traffic environments", *Journal of the Association for Computing Machinery*, vol. 20, no. 1, Jan.1973.

[11] J.A. Stankovic, M. Spuri, K. Ramamritham, G.C. Buttazzo, *Deadline Scheduling for Rreal-Time Systems - EDF and Related Algorithms*. Kluwer Academic Publishers, 1998.

[12] C. Bergenhem, M .Jonsson, B. Gördén and A. Ålander, "Heterogeneous real-time services in high-performance system area networks – application demands and case study definitions*", Technical Report 263, Halmstad University*, Halmstad Sweden, Oct. 1, 2002.

IEEE COMPUTER SOCIETY