Technical Report

Two Protocols with Heterogeneous Real-Time Services for High-Performance Embedded Networks

Carl Bergenhem
Magnus Jonsson

School of Information Science, Computer and Electrical Engineering
HALMSTAD UNIVERSITY
Halmstad, Sweden, 2012

Two Protocols with Heterogeneous Real-Time Services for High-Performance Embedded Networks

Carl Bergenhem
SP - Technical Research Institute of Sweden
Department of Electronics
SE-501 15 Borås, Sweden
Tel: +46-10-516 5553,
carl.bergenhem@sp.se

Magnus Jonsson
School of Information Science, Computer and Electrical Engineering
Halmstad University
Halmstad, Sweden

ABSTRACT

High-performance embedded networks are found in computer systems that perform applications such as radar signal processing and multimedia rendering. The system can be composed of multiple computer nodes that are interconnected with the network. Properties of the network such as latency and speed affect the performance of the entire system. A node´s access to the network is controlled by a medium access protocol. This protocol decides e.g. real-time properties and services that the network will offer its users, i.e. the nodes. Two such network protocols with heterogeneous real-time services are presented. The protocols offer different communication services and services for parallel and distributed real-time processing. The latter services include barrier synchronisation, global reduction and short message service. A network topology of a unidirectional pipelined optical fibre-ribbon ring is assumed for both presented protocols. In such a network several simultaneous transmissions in non-overlapping segments are possible. Both protocols are aimed for applications that require a high-performance embedded network such as radar signal processing and multimedia. In these applications the system can be organised as multiple interconnected computation nodes that co-operate in parallel to achieve higher performance. The computing performance of the whole system is greatly affected by the choice of network. Computing nodes in a system for radar signal processing should be tightly coupled, i.e., communications cost, such as latency, between nodes should be small. This is possible if a suitable network with an efficient protocol is used. The target applications have heterogeneous real-time requirements for communication in that different classes of data-traffic exist. The traffic can be classified according to its requirements. The proposed protocols partition data-traffic into three classes with distinctly different qualities. These classes are: traffic with hard real-time demands, such as mission critical commands; traffic with soft real-time demands, such as application data (a deadline miss here only leads to decreased performance); and traffic with no real-time constraints at all. The protocols are analysed and performance is tested through simulation with different data-traffic patterns.

Keywords: Optical, Ring, Pipeline, Distributed, Parallel-Processing, Real-time, SAN (System Area Network), Heterogeneous, Service, Medium Access Protocol

Contents

ACKNOWLEDGEMENTS

This research work is part of M-NET, a project financed by SSF (Swedish Foundation for Strategic Research) through ARTES (A Real-Time network of graduate Education in Sweden). Further financial support was provided by the Electronics department at SP - The Technical Research Institute of Sweden.

1. INTRODUCTION

Radar signal processing is an application that places high demands on the computing device both concerning real-time properties and computational capacity. These demands can be realised with distributed processing. The aim of distributed processing is to provide more computing power than can be achieved with a single node. The system that performs the distributed processing consists of computing nodes that are interconnected with a network. The network is hence an integral part of the system. As the complexity of the applications increases, so do the performance requirements on the system, and the requirements of the network itself.

The network itself consists of several parts: the physical architecture, the protocols that control access and the protocols that implement services to users of the network. The choice of network architecture affects performance of the system to execute a particular application. In addition to network performance in terms of speed, also the architecture of the network affects performance of the system. The architecture is the pattern in which nodes are interconnected. In this report the focus is on a specific class of network architecture – the pipelined ring network. Such a network is composed of unidirectional point-to-point links between each node to form a ring. In this context pipelining refers to the capability of the network to support several non-overlapping transmissions of messages concurrently. The aggregated throughput can thus be higher than one message per slot.

In this report two protocols are presented: The Two Cycle Medium Access protocol (TCMA) and the Control Channel based Ring network protocol with Earliest Deadline First scheduling (CCR-EDF). Both provide a medium access control protocol and support heterogeneous real-time services. They are suitable for the class of high-performance pipelined ring networks that is assumed in this report. Heterogeneous real-time services imply that the services provided to the users of the network cater for different requirements from each user; that is, one size does not fit all. The network treats packets according to its type. The data-traffic in the interconnection network of a distributed system is a mixture of data-traffic from different classes, such as data, control and logging traffic. Some data-traffic may be classified as hard real-time, while other is insensitive to delay. There may also be other constraints on the data-traffic such as guaranteed throughput. Often, guaranteeing real-time services is much more important in these systems than performance, such as low average latency.

The user of the services from the network is an application such as radar signal processing. Messages that are sent between computation nodes may also have real-time requirements. In distributed parallel processing, a large part of the overhead in computation comes from communication. This can be reduced if the network protocol offers the user services aimed at specific types of communication used in distributed parallel processing. The protocols offer two groups of services: One group of services for sending messages and another group of services specifically for computation with parallel and distributed processing.

Both protocols are independent of each other and offer a range of services; the difference being how the services are realised and hence they have different properties. The first protocol provides communication guarantees for real-time virtual channels and guarantee seeking messages through reservation of slots. The protocol can schedule, i.e. give guarantee for, more than one messages per communication slot because it can take advantage of the capability of the network architecture to support aggregated transmissions of messages in non-overlapping segments. However, reserved but unused communication slots cannot be reused by other traffic. This implies that reservation of slots leads to lower utilisation of network capacity. The first protocol also suffers from a pessimistic worst case in its scheduling framework due to a round-robin scheme of clock generation. The second protocol provides communication guarantees for logical real-time connections through earliest deadline first scheduling. The protocol can schedule, i.e. give guarantee for, up to one messages per communication slot, but cannot guarantee to support aggregated transmissions of messages in non-overlapping segments. Unused slots can be dynamically reused by other traffic which leads to better utilisation. The clock generation scheme is improved and utilises the network capacity more efficiently than the first protocol. Both protocols assume the same underlying network architecture and they are based on earlier research presented in (Jonsson 1998). The two protocols are evaluated by analysis and simulation.

When measuring the total benefit of a network, several design parameters must be taken into account. These include throughput of the network, communication latency, how well the application maps to the topology of the network, the price / performance ratio, the range of user services etc. These design parameters are evaluated for the two proposed protocols.

The rest of the report is organised as follows. Section 2 describes the networks architecture that is assumed for the protocols and simulations in the rest of the report. Section 3 describes the characteristics of the target application area. Section 5 discusses related work from three different viewpoints: application, architecture and user service. Section 6 gives an overview of the proposed protocols and result in the report. The two network protocols, TCMA and CCR-EDF, and a simulation study are presented in Part A and B of the report respectively. The overall conclusions of the report are discussed in Section 25 and, finally, the possible directions for future work are discussed in Section 26.

2. Physical Architecture of the Network

A network is considered to consist of both a particular architecture and at least a medium access control (MAC) protocol. The architecture of a network is the physical structures, such as electrical or optical elements, that interconnect nodes. In addition to the actual performance of each link also the pattern of how nodes are interconnected is of importance. The MAC protocol of the network controls the way in which nodes get access to send message to another node or group of nodes. The network may also include other layers of the OSI communications model. In this model the physical parts of the network is layer 1 and the medium access control protocol is part of layer 2. Other functions that a network performs are to offer different services for sending data-traffic. There can be several different services offering different levels of quality of service regarding timeliness of data-traffic delivery.

The proposed protocols, TCMA and CCR-EDF, assume an architecture or topology of the underlying physical network which is a unidirectional pipelined optical fibre-ribbon ring. This network architecture was proposed in (Jonsson 1998). The optical fibre link between two nodes is regarded as a point to point link with data being sent in one direction, see Figure 1. Each link contains several separate optical fibres – hence called an optical fibre-ribbon link. Each node has two ports, which connects to an incoming and an outgoing link, respectively. On the incoming link data from upstream nodes is received. On the outgoing link data is sent to downstream nodes. Pipelining implies that several independent transmissions can take place simultaneously in different segments of the ring. This is possible as the network is organised as independent point to point links. The success of pipelining depends on the data-traffic pattern and on coordination of a medium access control protocol.

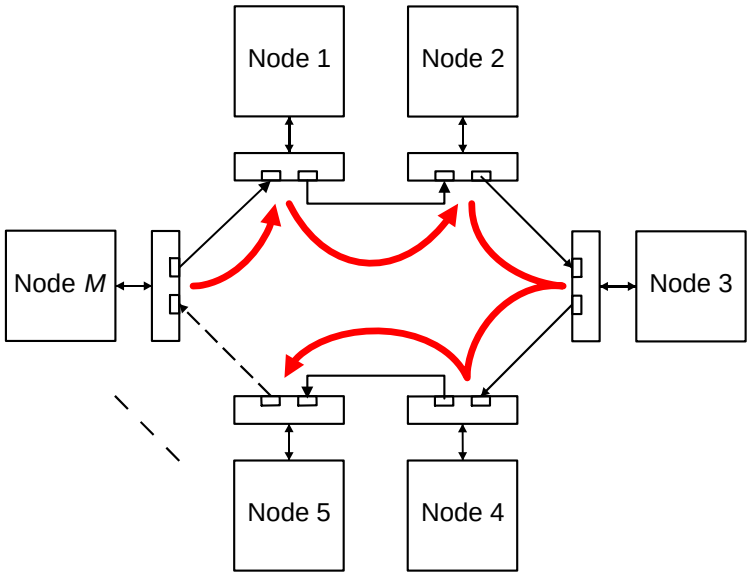

Figure 1: A simple example of pipelining in a network with *M* nodes. The figure also shows a pipelined unidirectional optical ring network.

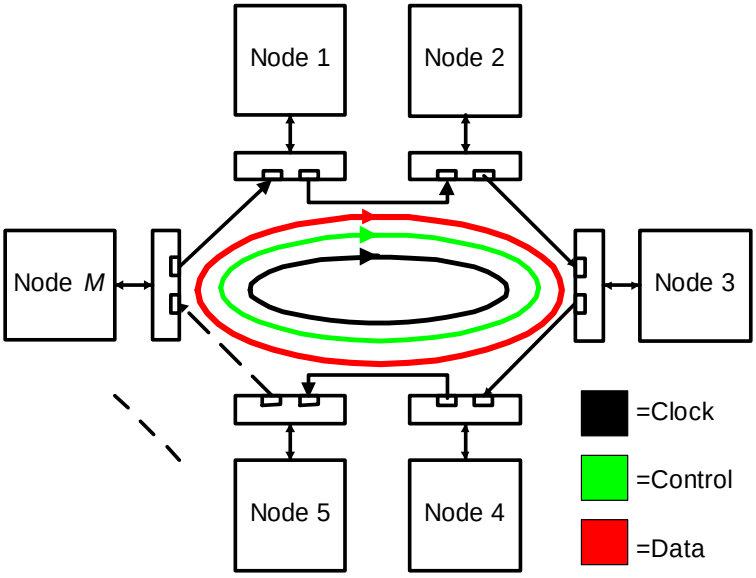

Figure 2: A Control Channel based Fiber Ribbon Pipeline Ring network.

Communication to the next neighbour makes efficient use of the pipelining feature of the network architecture. This is called spatial reuse of bandwidth. Simple pipelining in a network is depicted in Figure 1. In the figure, two single hop (unicast) transmissions take place from Node M to Node 1 and from Node 1 to Node 2. A multiple hop transmission also takes place from Node 2 to Node 5. The Intermediate nodes (node 3 and 4) can potentially also receive the transmission; depending on the capability of the protocol. If so, the latter transmission is a multicast (several receiver nodes). All of the transmissions described take place during the same time-period. The aggregated throughput during a time-period can therefore be much higher than the throughput of a single link and depends on the data-traffic pattern.

The decision to study a pipelined ring network is due to the STAP (Space Time Adaptive Processing) radar signal processing (RSP) algorithm being efficiently mapped on this architecture (Jonsson, Svensson et al. 1997) (Taveniku, Ahlander et al. 1998). This is due to the algorithm having a notion of distinct processing steps. Each processing step is performed by a single node and that the majority of the communication between the processing steps is to the next neighbour node. A more complex network, e.g. with full interconnection between nodes, could also be used successfully with the RSP algorithm but at a much higher cost, because more connections are needed to form the topology. Fibre-ribbon optical point to point links for short to medium distance have a good price / performance ratio. Examples of research on fibre-ribbon links are (Lemoff, Ali et al. 2005) (Schow, Doany et al. 2011) (Trezza, Hamster et al. 2003). Fibre-ribbon links are also available commercially from e.g. Zarlink (Zarlink 2009).

The network with the proposed protocol is suitable for LANs and SANs (system area networks) where the number of nodes and network length is relatively small, e.g. one hundred meters or less. This is important since the propagation delay adversely affects the medium access protocol. Examples of applications are as an interconnection network in embedded systems and cluster computing.

An optical interconnect with bi-directional links and with ten fibres per direction (such as Motorola OPTOBUS™) (Lebby, Gaw et al. 1996) is used. The links are arranged in a unidirectional ring architecture where only $\lceil N / 2 \rceil$ bi-directional links are needed to close a ring of N nodes. Fibre-ribbon links offering an aggregated bit rate of several Gbits/s reached the market in the mid nineties (Bursky 1994). The increasingly good price/performance ratio for fibre-ribbon links indicates a great success potential for the proposed type of networks.

The physical ring network is divided into three rings or channels (see Figure 2). For each fibre ribbon link, eight fibres carry data, one fibre is used to clock the data, byte by byte, and one is used for the control channel. Access is divided into slots like in an ordinary TDMA (Time Division Multiple Access) network. The control channel ring is dedicated for bit-serial transmission of control packets. These are used for the arbitration of data transmission in each slot. The clock signal on the dedicated clock fibre also clocks each bit in the control packets. Separate and dedicated clock- and control fibres

simplify the transceiver hardware implementation in that no clock recovery circuitry is needed (Bergenhem 2000). The control channel is also used for the implementation of low-level support for barrier-synchronisation, global reduction, and reliable transmission.

The ring can dynamically (for each slot) be partitioned into segments to obtain a pipeline optical ring network (Wong and Yum 1994). Several transmissions can be performed simultaneously through spatial bandwidth reuse, thus achieving an aggregated throughput higher than the single-link bit rate. Even simultaneous multicast transmissions are possible as long as multicast segments do not overlap. Although simultaneous transmissions are possible in the network because of spatial reuse, each node can only transmit one packet at a time.

3. CHARACTERISTICS OF THE NETWORK

The network protocols presented in this report are suitable for range of applications that have common requirements. The most important network related properties are listed below and commented:

- High throughput/performance. It is difficult or even meaningless to exactly quantify the actual throughput of "high performance". However, the text below will try to clarify what is meant and the region of performance that is implied. As a comparison the current record transmission over one optic fibre resulted in 25.6 Tbit/s throughput (Gnauck, Charlet et al.). A common commercial-off-the-shelf (COTS) network that is aimed for server environments of small to medium size is 10 Gigabit Ethernet (Cunningham, Div et al. 2001). The data rate from the radar antenna in the RSP case studied (Bergenhem, Jonsson et al. 2002) is approximately 6 Gbit/s. A high throughput COTS fibre ribbon optical interconnect that could be used as links in the pipelined ring network is the Zarlink ZL60101(Zarlink 2009). It features 12 optical transmitters (at 2.72 Gbit/s per channel) with an aggregated rate of 32.6 Gbit/s over a maximum distance of 300m (using multi-mode fiber). Note that the Zarlink parallel optical interconnect has wider link parallelism than assumed for the pipelined ring network in this report. This implies that data can, for example, be transported in a more parallel way and offer other user service features. Finally, the actual capacity or bit rate of the optical links is increased by continuous development of materials and manufacturing process. An example, although not COTS, is reported on in (Lemoff, Ali et al. 2005).
- Short to medium distances. These are defined as the distances covered by SANs and small LANs. The upper limit is dependent on the type of interconnect used. A longer distance implies longer latency because of propagation delays. This affects the usefulness of the network in applications where latency is important, even though the protocols can be designed to better deal with the propagation delay. See also Section 5.1 for a survey of SANs. The networks discussed in this report can even be used in future optical backplanes (Wu 2012).
- Embedded system. The network in an embedded system is not normally directly connected to an outside network. If there is a connection it is via a gateway that provides e.g. security and translation to another type of network, such as optical fibre to copper-based. Another reason is performance, i.e., the network outside the embedded system is generally slower than the SAN inside (Wolf 2002).
- Heterogeneous data-traffic. Different types of data-traffic in the network can be identified, application and control data-traffic. The proportions may be partly known (as in the applications studied in this report). The mixture of data-traffic is "heterogeneous" in that it is of different types that must be treated accordingly. An example of different treatment of data-traffic is giving real-time application data high priority over less important data logging traffic. The network must be deterministic, meaning that behaviour must be independent of data-traffic, and that it must be able to give guarantees of timeliness if required. Distinction of data-traffic is an important function since it affects the performance and capabilities of the whole system. Some systems cannot provide service unless its data-traffic is given adequate guarantees such as for timeliness (Stankovic 1988).

Radar signal processing is an application that has these characteristics and is discussed in the next section.

4. THE RADAR SIGNAL PROCESSING APPLICATION

Radar is widely used to aide air and sea based transport. Different types of radar perform different kinds of tasks. Modern radar systems are often based on the use of a phased array antenna. This antenna is composed of multiple fixed antenna elements and digital beam forming instead of a physically being moved. Airborne radar, implying that the computing equipment is onboard, is considered here. The fact that the application is airborne implies limitation such as physical size and power consumption. An introduction to airborne radar is given in (Stimson 1998). Radar signal processing (RSP) requires much computing power because of the complex algorithms used. The more advanced the algorithms are the more useful the result, but also requires larger processing capacity. There are various algorithms with different features for airborne RSP. An example is space-time adaptive processing (Klemm 1999). The algorithm itself is not important for this work but the requirements of the computations are important. Such a requirement is for example the approximate sizes of data being transferred during computation. The requirements lead to the conclusion that the embedded system performing RSP needs both real-time and supercomputer capabilities.

The algorithms in RSP comprise mainly of linear operations such as matrix-by-vector multiplication, matrix inversion, digital signal processing, such as FIR-filtering and DFT, etc. In general, the functions will work on relatively short vectors and small matrices, but at a fast pace and with large sets of vectors and matrices. Even if the incoming data can be viewed as large 3-dimensional data cubes, the data can be divided into smaller blocks so that processing is possible on separate processing elements in a parallel computer.

After calculation of one algorithmic step in a radar signal processing chain the data is transferred to the node(s) responsible for the next algorithmic step. Alternatively that dataset on which processing is taking place in a node needs to be rearranged. This rearrangement of data, called a corner-turn, involves many-to-many communication between involved nodes. In addition to the transfer of radar data, there are control data that control the processing and auxiliary data which includes logging and monitoring of the system.

The RSP computing device is a typical embedded system (Wolf 2002). The pilot is only interested in being able to use the computer for the task at hand, namely processing the radar signal and displaying the results. Thus the computer is specialised and dedicated to one specific task. The maximum distance between two modules is typically below 2 m (intra-rack communication), but it might be valuable if it is possible to include a near-antenna system a little bit longer away.

RSP is a real-time application. This is because the value of the results, i.e. service to the human operator, depends on them being produced in a timely fashion as well as on the correctness of the result (Stankovic 1988). To solve RSP in real-time a distributed parallel computer system can be used. In such a system many nodes co-operate since processing in a single node is not sufficient. Such a distributed system is proposed in (Taveniku, Ahlander et al. 1998). The combined processing power is great enough for the task. However, new problems arise, e.g. because of communication latency, and a novel approach is needed to the design of such a system. Considerations on physical size and power efficiency while still meeting the goal of processing power are important. The design of the computing nodes is not further discussed in this report, but focus is rather on the network that interconnects nodes.

The performance of a system such as a parallel computer or distributed real-time system, is highly dependent on the performance of their interconnection network. This is especially true in a data intensive application such as radar signal processing. Processing in this application system and also the dataset is distributed throughout several nodes. The network architecture is assumed to be a pipelined unidirectional ring network. A study by (Jonsson, Svensson et al. 1997) has shown that the radar processing algorithms map suitably on a pipelined ring topology.

In addition to data communication, services that support processing such as process synchronization are valuable to radar systems since the manufacturers are striving for engineer-efficiency (Åhlander and Taveniku 2002). Motivations for engineering-efficiency in radar systems are: (i) long development time for a low number of sold systems, (ii) updates of the systems or product families are made several times during their life-time, and (iii) it is often difficult to employ enough good engineers.

5. RELATED WORK

Networks may be classified according to how nodes communicate among themselves. The medium used for communication may or may not be shared. In a shared medium network several nodes share access. This is also known as a multiple access protocol. A point to point protocol, implies that there can only be two nodes, one sender and one receiver, in the network. A point to point protocol is generally simpler than a multiple access protocol because there are fewer concerns to solve. The protocol that resolves medium access is called a medium access control (MAC) protocol. This protocol forms a sublayer of the Data Link Layer (layer 2) as specified in the seven-layer OSI communications model. The other sublayer of the data link layer is called Logical Link Control (LLC) and concerns acknowledgement, flow-control and error notification.

Communication may take place over a network that is switched or routed. Communication in the former is known as forwarding and only the MAC-protocol (layer 2) is involved. Communication takes place within one network segment also known as a subnet. Each individual segment executes its own instant of MAC-protocol. In the latter case network layer (layer 3) protocols are involved to enable communication between different network segments. Here nodes communicate via one or more intermediate devices, called routers, which execute a network layer protocol. The protocols that are presented in this report are in function mainly MAC-protocols, i.e. execute in one network segment only to resolve access. Generally, in real-time communication, the protocol should also possess characteristics such as determinism, fairness between nodes and guaranteed delivery.

Figure 3 shows a classification of communication protocols which is based on work by (Rom and Sidi 1990). The protocols of interest in the classification are all decentralised in that no single node coordinates the network. This further implies that all nodes are executing the same protocol. The highest level of the classification tree distinguishes between point to point and multiple access protocols. A multiple access protocol can be used in a network with multiple nodes, (one up to many). However, in a network with only two nodes the multiple access protocol is, strictly speaking, unnecessary. A point to point protocol cannot successfully be used in a network with multiple nodes. The point to point category is not of further interest since it cannot coordinate more than two nodes.

Multiple access protocols are then classified as being either conflict free or contention based protocols. Contention implies that two or more nodes may simultaneously require access to transmit to the network. This situation must be resolved by the MAC-protocol. Alternatively some allocation of network resource has been done before transmission occurs, i.e. no more than one node will access the network at a time; hence the network is conflict free.

In a contention free scheme successful transmission is guaranteed provided that a correct (and successful) allocation has been made. The allocation of network access in a contention free protocol can be done with either a static or dynamic scheme. Static allocation can be done according to some physical property of the channel such as time, frequency or a combination. Allocation according to time is known as Time Division Multiple Access (TDMA) and implies that nodes are allocated access according to the passage of time. Allocation according to frequency implies that nodes are allocated access according to a particular frequency in the channel, e.g. radio frequency in a wireless medium.

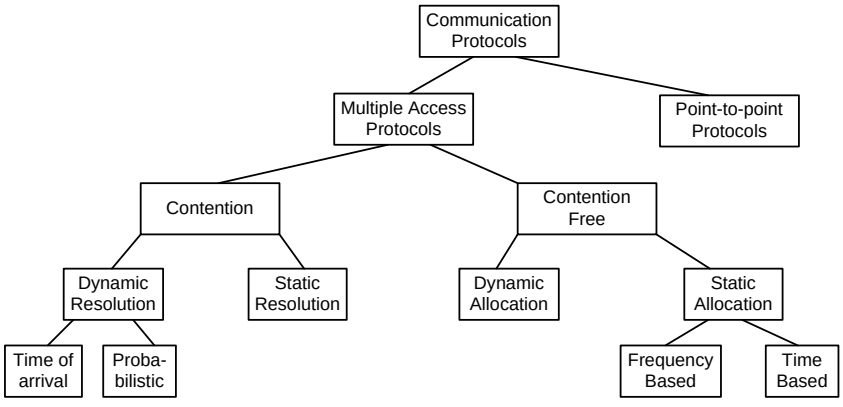

Figure 3: Classification of communication protocols

In contrast to static allocation schemes, the dynamic allocation scheme implies that network access is divided among nodes according to the current needs of the nodes. Two examples of dynamic allocation of network access are via reservation or token passing. Reservation implies that a node that has a message to send will announce this request to other nodes. In the token passing scheme, a logical token is passed among nodes. The token must be held by a node to enable access to the network. An example of this is the Token Ring network (IEEE 1985).

When contention occurs in a contention based protocol resolution can be either static or dynamic to decide which node gets to send. Static resolution can be according to the ID of the node or of the message. An example of the latter is CAN (CAN 1991). An identical conflict, e.g. between the same messages, will always be resolved identically. This resolution scheme for CAN can give timing guarantees for messages. Dynamic resolution implies that resolution depends on the state of the conflicting messages, such as age of each message. Dynamic resolution can also be probabilistic; this implying indeterminism. An example of this is the Ethernet MAC protocol (CSMA/CD 1985).

In contrast to a conflict free scheme, transmission in a contention based scheme is not guaranteed to be immediately successful because no pre-allocation of the network can be made. Success or failure depends on other data-traffic in the network, and timing can not be known in advance. Normally a contention based protocol still aims to be fair (allow all nodes equal access) and allow all messages to be sent eventually.

Due to the lack of determinism in some contention based protocols, they are therefore not well suited for real-time communication. It is possible that some real-time capabilities may be offered if other, higher level protocols are added. An example of a contention based protocol that is not directly suitable for real-time data-traffic is the Ethernet MAC protocol. However, it has been shown in (Fan and Jonsson 2005) (Hoang, Jonsson et al. 2002) that Ethernet may be used for real-time communication if the architecture is constrained to be point to point and switch based. An example of a contention-based protocol that supports prioritisation via message ID is CAN (CAN 1991). It can be augmented to support true real-time communication (Davis, Burns et al. 2007).

According to the classification system discussed above, the pipelined ring network (Jonsson 1998) is a contention free network with dynamic allocation. The network resource is done via reservation in a separate control channel.

There are many networks that are both commercially available and that are reported in literature, but most of them do not offer real-time services. Networks that do have some real-time support do not offer the range of services that are required in heterogeneous environments. Two networks, CAN and Ethernet, have been briefly mentioned above. RACEway (ANSI 1999) (Kuszmaul 1995) from Mercury Computer Systems supports priority stamped messages but as with CAN the priority does not necessarily relate to deadlines of messages. The TD-WDMA (Time division Wavelength Division Multiple Access) network (Jonsson, Borjesson et al. 1997) and the CC-FPR (Control Channel based Fiber-ribbon Pipeline Ring) network (Jonsson 1998) offer deadline based prioritisation of messages locally in each node but no support for global optimisation of messages in all nodes.

Other surveys have been done on real-time features in networks such as that by Zhang (Zhang 1995). However, the focus in that survey is on packet-switched wide area networks (WANs). The authors study the function of the switches in the network and the handling of each separate logical connection from sender to receiver. Another survey of real-time communication is given in (Malcolm and Wei 1995). However, the article does not discuss the concept of multiple services. An overview of quality of service capabilities in three interconnect networks, Infiniband, Advanced Switching Interconnect (ASI) and backplane Ethernet (Reinemo, Skeie et al. 2006).

Our research focus is on methods to offer services for heterogeneous real-time requirements on a system-wide integration/optimisation basis. To ensure that an embedded system has a correct function, the real-time requirements must be considered for the system as a whole, i.e. among all nodes. It is insufficient to only meet the requirements in each node separately.

The following sections give an overview of related protocols and networks and categorise each of them into three main areas of focus. Some networks are mentioned in more than one section because they fall into more than one category.

- The first focus is on networks that are used in the same area of application, which is high-speed communication in high-performance embedded systems. These are known as system area networks (SAN).

- The second focus is on networks that have an architecture similar to the pipelined ring network. This can be optical networks, pipelined networks, ring networks, networks with spatial reuse etc.
- The third focus is on user services that are similar in some sense to those offered in the proposed protocols. Protocols that offer heterogeneous real-time services are especially relevant.

5.1. High-performance System Area Networks

The architecture of embedded systems used for high-performance computing applications can be organised such that internal communication is required. This internal communication may range from a simple passive backplane to a multi-service data network. The latter type of network is commonly referred to as a system area network (SAN). Other requirements on the SAN are e.g. real-time support with deterministic transmission delay and scalability of the number of nodes. A SAN is a relatively new class of networks (Mehra 2001). They are designed for interconnection inside a single cabinet and up to interconnection of multiple cabinets in a single room, i.e. the length is ten to one hundred meters (Hennessy, Patterson et al. 2003). The connected nodes should be regarded as being part of the same system. High performance relates to high network capacity and low latency. An example of a SAN is TNET (Horst 1995).

The basic part of the RACEway network architecture (ANSI 1999) (Kuszmaul 1995) is a six-port switch chip that may be statically connected to other chips to form different topologies. Most common is the fat tree topology, although mesh or clos networks are possible. A mesh network is a direct network topology where each node is directly connected to other nodes. A clos network is a type of multi-stage network topology with three intermediate switch stages.

A pre-emptable circuit-switched path is established between the source and destination. The message header contains information about the path to the destination. Messages have four levels of priority where higher priority messages pre-empt lower priority messages. This procedure is called “killing blocking messbages” and occurs when the path of a lower priority message conflicts with the path of a higher priority message. In this case the path of the lower priority message is pre-empted and the higher priority message is sent. The path of the lower priority message is later re-established automatically as part of the functionality in the chip. The RACEway network has a known worst-case latency for a high priority message to get through the longest path in the tree and time needed to kill any blocking lower priority messages. The RACEway network is used in current commercial radar signal processing systems (Einstein 1997).

The PONI network is aimed for use in “Campus” environments (Sano and Levi 1998) and (Raghavan, Kim et al. 1999). It is aimed at solving I/O bottle necks for interconnects on length scales of 1 to a few 100 m, i.e. the domain of medium scale interconnects (SANs up to small LANs). The architecture is unidirectional slotted ring using parallel fibre-optical links. The fibre-ribbon links have ten fibres per direction. Eight of these are used as a parallel data path and the other two for clock and frame control channel. The medium access protocol implements a slotted ring protocol. Average ring latency (data-traffic destined half-way around the ring), without other data-traffic in the ring, is 0.8 – 1.6 μs.

Myrinet (Boden, Cohen et al. 1995) is a switched network built of high performance communication links with a capacity of 1.28 Gbit/s full duplex (in both directions). Myrinet uses wormhole switching and source routing. In wormhole switching the network packets are broken into small pieces called flits. The first flit, holds information about the destination address of the packet and sets up the routing behaviour for all subsequent flits associated with the packet. The head flit is followed by zero or more body flits, containing the actual pay load of data. The final flit, called the tail flit, closes the connection between the two nodes. An advantage of wormhole switching is that the entire packet does not need to be stored at intermediate stages before being forwarded. More information on wormhole switching can be found in (Duato, Yalamanchili et al. 2002). In Source routing, the packet header holds information that partially or completely specified the routing for the packet, i.e. every switch that the packet will pass from sender to destination. Switches, used between Myrinet nodes, are “perfect”, meaning that packets do not conflict unless they are directed to the same outgoing port. Otherwise, switches have no further capabilities other than source routing and wormhole switching. No broadcast capability is available. The worst-case switching time is 500 ns. An arbitrary topology is possible provided that the interface cards and switches are suitably configured.

Infiniband (IB) (Pfister 2001) is a relatively new high-speed serial point to point linked, switch and router based high-speed interconnect. It aims at a wide range of interconnection levels, from SAN to replacing today's common, often bus-based, internal I/O bus, e.g. the PCI-bus. Different systems can also be interconnected with IB. A node in an IB network can be a processor, memory, storage device etc. Self-contained systems may be partitioned into sub-networks. IB supports any topology, defined through configuration in the switches and routers. Routing is based on forwarding tables and uses virtual cut-through switching. In virtual cut-through switching forwarding of the packet to the appropriate outgoing switch port commences as soon as the destination header can be read. The technique decreases latency but also decreases reliability since a corrupt packet (e.g. failed CRC) will still be forwarded but not be detected until the entire packet has been received by the switch (Kermani and Kleinrock 1979). The large address structure of IB (similar to IPv6) enables each connected node to be addressed individually. Infiniband can be enabled with QoS support through differentiating data-traffic into different classes (similar to IPv6) (Pelissier 2000). Switches and routers can then treat the packets accordingly. However, the QoS concept in Infiniband is based on prioritisation of data-traffic in switches and routers. It is therefore uncertain whether the network can offer hard real-time guarantees without further development or analysis.

One Gbit/s Ethernet (Gigabit Ethernet) and ten Gbit/s Ethernet are standardised and accepted in industry (Saunders 1998). Both can be used as SAN (Vaughan-Nichols 2002). Gigabit Ethernet, at least, is already a relatively cheap technology. The packet format of the Ethernet standards mentioned are backward compatible with older (lower bit rates) versions of Ethernet. Both network standards can take the form of a function as a switched, point to point linked network. With prioritisation of data-traffic and intelligent scheduling in the switches, some real-time capability is possible (Hoang, Jonsson et al. 2002). The latter research is however for 100 Mbit/s Ethernet (Fast Ethernet). There also exists an Ethernet standard denoted 802.3ap for use specifically as a backplane (Healey 2007).

HIPPI is a high-speed parallel interface running at 800 MByte/s (Tolmie, Boorman et al. 1999). It uses point to point links, connected to switches to achieve communication. The network uses circuit switching. Because of arbitrary blocking times in the switches (partly due to the circuit switching scheme), HIPPI cannot directly support real-time communication. However, additional upper-layer protocols can provide limited success in real-time networking (Bettati and Nica 1995).

5.2. Networks with related architectures

The focus in this subsection is on networks that have a similar or related architecture. This includes

- Pure ring networks with various medium access methods,
- Pipelined networks,
- Networks with spatial reuse.

Ring networks in which nodes are interconnected node-to-node can support spatial reuse. In this case concurrent transmissions of messages are possible in different segments of the ring. This capability is available in a number of single and double ring protocols. A brief survey of these networks is given in (Wong and Yum 1994). Ring networks are divided into four main groups:

- Token based networks: FDDI (Ross 2002), DQBD (Mukherjee and Bisdikian 1992) and Token ring (IEEE 1985) are networks based on using a single token. The entire ring is used for each transmission because the message is circulated the entire ring before being removed by the sender. Consequently only one packet can be transmitted at a time and spatial reuse is therefore not supported.
- Slotted rings: the destination node removes the message from the ring so that down stream nodes can immediately use the unused segments. Examples are the Cambridge Ring (Hopper and Needham 1988) and the Orwell Ring (King and Gallagher 1990).
- Buffer insertion rings: the ring interface contains a variable length shift register that can buffer incoming messages during the transmission of locally generated messages. Buffer insertion rings offer spatial reuse of network capacity by allowing concurrent transmissions in non-overlapping segments of the ring. Examples of networks (architecture and protocol) are DLCN (Liu and Reames 1977) (Distributed Loop Computer Network), DDLCN (Liu and Wolf 1978) (Double Distributed-Loop Computer Network) and SILK (System for Integrated Local Communication) (Huber, Steinlin et al. 1983). The latter features a braided ring network which is used to increase

availability of the network. A medium access control layer protocol for a buffer insertion ring is SRP (Spatial Reuse Protocol) (Tsiang and Suwala 2000).

- Segmented rings: the ring is logically partitioned into segments, where each can support a message transmission. Examples are the Jafari loop (Jafari, Lewis et al. 1980), the Leventis Double Loop (Leventis, Papadopoulos et al. 1982), the circuit-switched play-through rings (Silio Jr 1986), the T-S ring (Pacifici and Pattavina 1986), the Concurrent Transmission Ring (Xu and Herzog 1988). Finally, the pipelined ring protocol proposed in (Wong and Yum 1994) is also an example of a segmented ring.

The ring networks that belong to the latter three classes all feature spatial reuse, thus having a maximum throughput greater than one. The network protocols proposed by us in this report belong to the segmented ring class of networks. Some of the mentioned networks are described further below.

The Distributed Queue Dual Bus (IEEE 806.2 DQDB) is a metropolitan area network (MAN), i.e. aimed to span over an area the size of a city. The basic architecture is two parallel, unidirectional buses, where each node is connected to both buses. Each bus is a multiple access broadcast bus similar in principle to a CSMA/CD (Carrier Sense Multiple Access / Collision Detection) bus. However, a slotted access method is used to overcome the access control limitations of the medium access protocol. DQDB has a network architecture related to the pipelined ring; its physical layer is a dual slotted bus. Concurrent transmission in multiple unused bus segments is possible in the slotted bus, i.e. spatial reuse. This capability can be used to overcome limitations caused by the large propagation delays in MANs.

The 802.5 token-ring network is a simple local area network. The protocol uses eight priority levels in a circulating token to decide which node has the highest priority message. When a node wants to send a packet, it “grabs” the token, marks it as busy, and appends the packet to it. The destination node registers that the frame contains a message that is destined to it and copies the frame into its own buffer. The busy token (with the packet still appended) continues around the network until the sender node removes it and issues a “normal” token instead. Because of this, only one transmission is possible at a time. The most common bit rates are 4 Mbit/s and 16 Mbit/s. At higher bit rates, the protocol becomes increasingly inefficient because of constant ring propagation delay.

FDDI, short for Fibre Distributed Data Interface, is a local area network that uses optical fibre links. It is modelled after the token ring network 802.5 (IEEE 1985), although the MAC layer more closely resembles 802.4 (token bus) (Alijani and Morrison 1990). The MAC protocol uses timing information of the passing token to share access among nodes. FDDI defines two types of data, synchronous and asynchronous. The synchronous data type can be used for data-traffic with real-time requirements since the network can guarantee this data-traffic (Zhao, Kumar et al. 1994). The MAC protocol used in FDDI implements the timed-token protocol (Malcolm and Wei 1994) and includes standards for physical layer and methods for achieving, e.g., fault tolerance. The timed token protocol is analysed in (Krishna and Shin 1997). The most common FDDI bit rate is 100 Mbit/s.

The Resilient Packet Ring network (RPR IEEE 802.17) (Davik, Yilmaz et al. 2004) is aimed for Local, Metropolitan and Wide Area Networks. Its data rate is scalable to many gigabits per second. The basic network architecture is ring based. Links between nodes are bidirectional point-to-point and the network can therefore be reconfigured in the event of node or link failure and continue to provide service. Several physical layers are defined such as Ethernet and SDH.

A pipelined ring protocol is proposed in (Wong and Yum 1994). The protocol allows the destination node to remove the message body from the ring and to issue a new token for the succeeding nodes to establish another transmission in the remaining ring segments. Spatial reuse of unused network links is therefore possible. The research presented in (Wong and Yum 1994) has greatly influenced the design of the protocols presented by us in this report.

5.3. User services

The main goal of a network is to support communication between a sender and one or more receiver nodes. To this end one or more user services can be offered for different types of data-traffic with different requirements. The user, i.e. programmer of the system, will utilise the different communication services from the network depending on what information is to be sent. Requirements mainly concern latency and throughput. Three examples of data-traffic classes are non real-time data-

traffic, best effort data-traffic and real-time data-traffic. A similar partition of data-traffic types is proposed in (Arvind, Ramamritham et al. 1991). In the rest of this subsection the user services from various networks are investigated. A network is studied mainly without additional software schemes etc. that enhance its capabilities.

A commonly used network is Ethernet (Tanenbaum 2002). Ethernet treats all messages equally and. gives no guarantees of timely delivery. The medium access method used in Ethernet, CSMA/CD, introduces non-determinism to communication especially under load. The problem is evident in shared medium Ethernet, i.e. a logical bus topology, but can be partially avoided by using switched Ethernet. Here there is only one node per network segment and a dedicated centralised switch; hence star topology. Non-determinism can still occur when two messages contend at a single switch port.

The CAN protocol (CAN 1991) is based on Carrier Sense Multiple Access / Bitwise Arbitration (CSMA/BA) for medium access control. The collision avoidance mechanism is a basic binary count down method (Tanenbaum 2002) of the ID of contending messages. This provides rudimentary support for priority. Several schemes to provide real-time scheduling have been proposed for CAN. In the report by Tindell et al. (Tindell, Hansson et al. 1994) the rate-monotonic scheduling algorithm is adapted to the CAN protocol. This research is later refined in (Davis, Burns et al. 2007). In (Zuberi and Shin 2000) the earliest deadline first and deadline monotonic scheduling algorithms are adapted to CAN. This work has influenced the scheduling frame work developed by us. CAN with real-time scheduling is used in safety-critical real-time systems. Observe that CAN is aimed for a different network environment: as an industrial field-bus or vehicle application. In these environments the focus is on e.g. immunity to electromagnetic disturbance and low production cast rather than throughput. The maximum bit rate for CAN is 1 Mbit/s but lower bitrates are more common. CAN (together with related research on real-time scheduling) is included in the related work for two reasons: first, because of the related deadline scheduling methods that can be put on top of it and, second because of the novel methods for encoding priority in the CAN frame.

Protocols based on Time Division Multiple Access (TDMA) can give guarantees for throughput and timeliness for real-time data-traffic. However, with the TDMA scheme unused time slots allocated to a particular node cannot automatically be reused by data-traffic in other nodes. This leads to wasted capacity and inflexibility. The TDMA scheme is used in the Time Triggered Protocol (Kopetz and Bauer 2003).

In Flexray the communication cycle is divided into two main parts: A prescheduled static TDMA segment for time-triggered messages and a dynamic segment for event-triggered messages. In the time-triggered segment communication is statically allocated and communication recurs in cycles with an identical communication pattern. In the event-triggered segment messages are arbitrated based on their message IDs. A higher message ID has priority over a lower ID. The communication system designer can analyse the event-triggered messages and assign IDs such that dynamic messages can be given certain guarantees of timeliness. Dynamic messages can also be assigned a low ID such that the transmission behaviour is best effort. Timing analysis for the Flexray dynamic segment is investigated in (Pop, Pop et al. 2006). Flexray hence offers three user services: static time-triggered messages, guaranteed event-triggered messages and best effort event-triggered messages.

The IEEE 802.5 (IEEE 1985) token ring has eight priority levels that may, e.g., be used for mapping deadlines. This only implies allocating priority between the levels, however and is not sufficient for the applications discussed in this report. An evaluation of token ring (and other protocols) with real-time data-traffic is given in (Alijani and Morrison 1990).

A network protocol with multiple services is proposed in (Arvind, Ramamritham et al. 1991). The services offered are: a connection-less service, a connection-oriented service and a real-time virtual channel service. These are listed in order of increasing quality of service, e.g. timeliness. This classification has influenced the protocols proposed by us.

FDDI (Ross 2002) has two services: synchronous and asynchronous. The usual mode of operation is that the synchronous service is used for high priority data-traffic (real-time data-traffic) and the asynchronous for low priority data-traffic. Greatly simplified, the two services function as follows. The timing of the synchronous service is set up in accordance with the requirements of the nodes in the network. This guarantees that the timing requirements will be met. There is, however, some slack in the system, and asynchronous data-traffic can use capacity that is not used by synchronous data-traffic.

The basic architecture of the Distributed Queue Dual Bus (DQDB) is a pair of slotted unidirectional buses with data-traffic flowing in opposite directions. Stations (i.e. nodes) are connected to both buses. If a station wishes to transmit to a down-stream station on one bus, a request is sent to up-stream nodes via the opposite bus. In this way requests from stations are queued up in stations in a virtual global FIFO queue. There is hence no central FIFO queue. Without additional protocols, DQDB provides a pre-arbitrated service for isochronous (real-time) data-traffic and a queue arbitrated service for non-isochronous (best effort) data-traffic. These services are similar to the services offered in FDDI. DQDB is less greedy than other 802 LAN protocols under normal load situations. However, studies have shown that it suffers from unfairness at high load and is therefore unsuitable for real-time communication (Tran-Gia and Stock 1990). It has also been shown that the a station which is further away from the head of the bus has an increased average waiting time, i.e. less opportunity to transmit. Unfairness in a protocol can prevent some nodes from using the network, i.e. cause starvation, and is an unwanted property. Research has been done to evaluate and suggest improvements in the fairness of DQDB under heavy load situations (Mukherjee and Banerjee 1993). Solutions for real-time data-traffic in DQDB are proposed in (Sha, Sathaye et al. 1992). The slotted bus structure has the ability to maintain almost constant (and very low) access delay when the network is loaded up to approximately 90%, which is significantly better than FDDI (Halsall 1995).

The medium access protocol for the Resilient Packet Ring network is designed to ensure fairness among nodes (Gjessing and Maus 2002). The protocol offers a priority scheme with three basic classes of data-traffic: 1) low latency low jitter, 2) predictable latency and 3) jitter, and best effort.

6. OVERVIEW OF RESULTS PRESENTED IN THE REPORT

Development of the protocols for the optical pipelined ring network, has taken place in three main steps. In this report only the two protocols relating to the two latter steps are described in depth. The articles that roughly relate to these steps are: (in the order of their publication)

- A basic protocol for a CC-FPR network (Jonsson, Bergenhem et al. 1999),
- The TCMA protocol (Bergenhem, Jonsson et al. 2001),
- The CCR-EDF protocol (Bergenhem and Jonsson 2002).

The first protocol, CC-FPR, is mentioned here only as background. It forms the basis for the two later protocols that are presented here. The focus of the CC-FPR protocol is on services for parallel processing and distributed real-time systems. The article presents the basic medium access protocol, packet format and assumed network architecture. The protocol includes user services for application traffic and services for parallel and distributed computing. Three basic services for application traffic are supported by the protocol: Real-time virtual channels, guarantee seeking messages and non-realtime messages. Scheduling of messages is done on individual basis at each nodes, i.e. there is no co-ordination between nodes. Because of the lack of co-ordination the user services real-time virtual channels and guarantee seeking messages use slot reservation to realise guarantees. The non-realtime service (in the report called best effort service) utilises left over capacity and gives no guarantees.

The Two Cycle Medium Access (TCMA) protocol has the advantage of better co-ordination of communication requirements between nodes. TCMA is described in depth later. Each data packet in the system contends for access with all other packets in the network, not only packets that are locally queued. Three user services for data-traffic are available: guarantee seeking messages, best effort messages, and non real-time messages. The first service is realised with slot reservation and is therefore completely deterministic. In the second service each individual packet is given a priority (deadline is mapped to priority). The third service neither gives any guarantees nor takes into account any timing constraints on the data-traffic. Instead capacity that is “left over” by the higher priority services is used. The TCMA protocol features deterministic services for parallel and distributed computing. Clock generation in the network is a task that is shared equally among the nodes according to round robin. This clocking scheme is however unsuitable and causes the protocol to suffer from an overly pessimistic worst-case deterministic performance. Average performance for best effort data-traffic is still good due to the pipelining capabilities of the network.

The protocol called Control Channel based fibre-ribbon pipeline Ring with support for Earliest Deadline First scheduling (CCR-EDF) represents an improvement over TCMA. An alternative

clocking scheme is proposed that improves the worst-case performance and removes serious impediments to the scheduling of network data-traffic. It is described in depth later. Clock generation in CCR-EDF is directly related to which node currently has the highest priority message. A dynamic scheduling framework for network messages is proposed. CCR-EDF supports global earliest deadline first (EDF) scheduling of messages. Scheduling up to 100 % capacity (one packet per slot) can be guaranteed. Due to pipelining throughput higher than one packet per slot is possible depending on the data-traffic pattern. Two data-traffic services based on (dynamic) EDF scheduling are available. The logical real-time channel service guarantees that data-traffic will be handled in a timely manner and the best effort service makes a “best effort” at timely transmission. Both of the services use the earliest deadline first policy. Slot reservation is not required to be able to give guarantees. The third data-traffic service is for non real-time data-traffic with no requirement for timeliness. This service has the lowest priority and uses the capacity that is left over from the other higher priority services.

The parameters from the radar signal processing case presented in (Bergenhem, Jonsson et al. 2002) are put into a simulator and simulated with an implementation of the CCR-EDF protocol. These simulations give a picture of how effective the protocol (and network) is with realistic data-traffic. As expected, the protocol correctly differentiates and allocates priority to the different data-traffic classes depending on the service with which the data-traffic is sent. On the basis of the simulation results it is concluded that the logical real-time channel service can guarantee data-traffic and that the best effort data-traffic service does make a best effort to deliver data-traffic. As expected, the best effort service fails to deliver when the network is saturated. Measurements of data-traffic levels and delays etc. in the network are made during simulation. A discussion of data-traffic and throughput in different situations is also presented.

PART A:

THE TWO CYCLE MEDIUM ACCESS PROTOCOL

7. INTRODUCTION TO TCMA

This part describes a novel medium access protocol called Two Cycle Medium Access (TCMA) protocol. It uses the deadline information of individual packets, queued for sending in each node, to make decisions, in a master node, about who gets to send. The new protocol may be used with the control channel based fibre ribbon pipeline ring (CC-FPR) network architecture. Correct function of the protocol is shown through simulation.

TCMA provides the user with a service for sending best effort messages, which are globally deadline scheduled. The global deadline scheduling is a mechanism that is built into the medium access protocol. No further software in upper layers is required for this service. Upper layer protocols can be added to a network, such as Ethernet, to achieve better real-time characteristics, but it is difficult to achieve fine deadline granularity.

Real-time services in the form of best effort messages, as mentioned above, guarantee seeking messages, and real-time virtual channels (RTVC) are supported for single destination, multicast and broadcast transmission by the network. There is also a service for non real-time messages. The network also provides services for parallel and distributed computer systems such as short messages, barrier synchronisation, and global reduction. Support for reliable transmission service (flow control and packet acknowledgement) is also provided as an intrinsic part of the network (Bergenhem and Olsson 1999).

A disadvantage with the CC-FPR protocol is that a node only considers the time constraints of packets that are queued in it, and not in downstream nodes. As an example a node may decide that it will send and book two downstream links, i.e. crossing the path of the neighbouring downstream node. This can be done regardless of what the downstream node may have to send. This implies that messages with tight deadlines may miss their deadlines. This problem is avoided with the TCMA protocol.

8. TWO-CYCLE MEDIUM ACCESS PROTOCOL

The proposed protocol has two basic phases for controlling medium access: the collection phase and distribution phase, see Figure 4. The protocol is therefore referred to as the two-cycle medium access protocol (TCMA). TCMA shares access between nodes with time division multiplexing. The basic time unit is called a slot. The minimum slot size is analysed in Section 10. Slots are organised into cycles with a predefined number of slots. The number of slots is chosen so that each node is master at least once per cycle (exactly once is assumed for simplicity).

In TCMA the role of being network master is cycled around the ring. In this respect all nodes are identical. The role as master is passed on to the next down stream node at the end of the slot. The master node is responsible for generating a clock signal which propagates through the ring to the other nodes. Every node detects when the clock signal is interrupted at the end of the slot and increments a counter which determines the succeeding master.

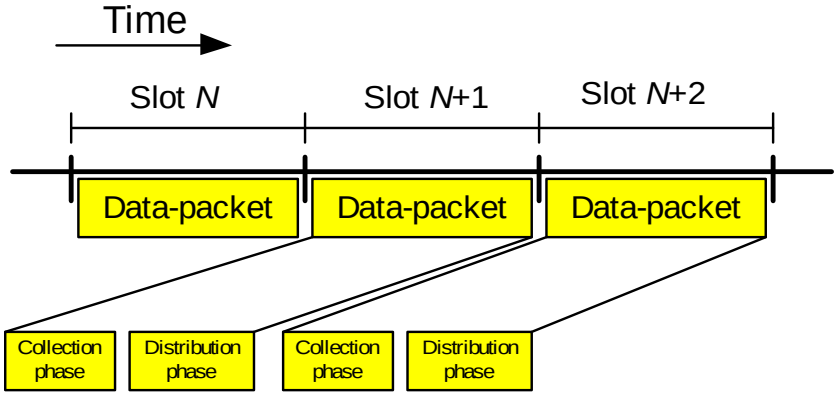

Figure 4: The two phases, collection and distribution, of the TCMA protocol. The network arbitration, for data in slot *N+1,* is performed in the previous slot, slot *N.* The lengths of the phases and placement in time in the figure are not to scale.

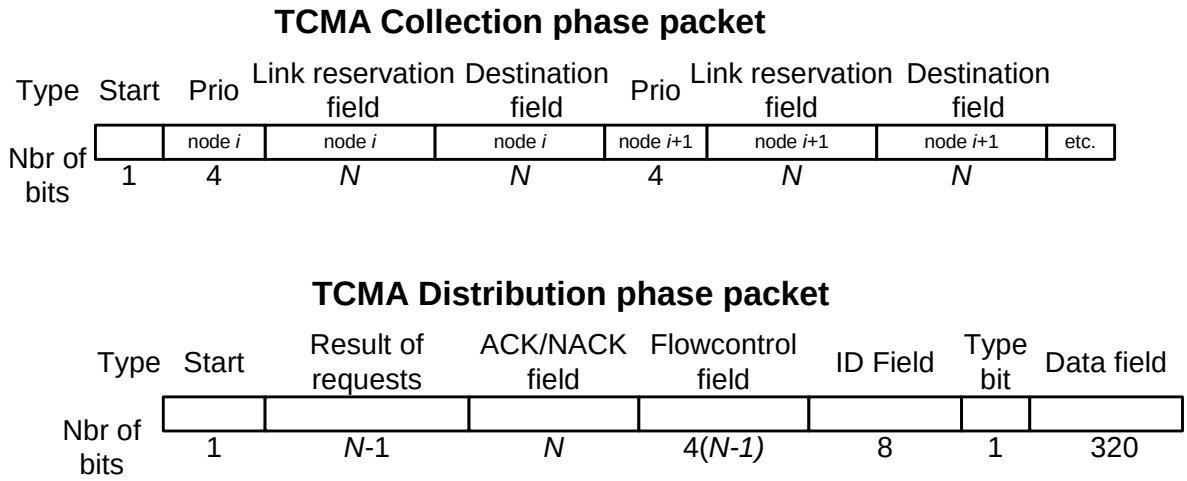

Figure 5: The contents of the TCMA control packets: Collection and distribution phase.

There are two types of TCMA control packets, which are used in each of the two phases during a slot: The collection phase packet and the distribution phase packet, see Figure 5.

A collection phase packet contains a start bit and a total of $N-1$ requests that are added one by one by each node. The master receives its own request internally. Each request consists of three fields. The "prio"-field contains the priority level of the request which is further described below. Nodes use the link reservation and destination fields to indicate destination node(s) and which links that must be traversed to reach the destination node. For the link reservation field, each bit corresponds to one link and indicates whether the link is reserved (1) or not (0). The destination field has one bit for each node in a corresponding way. A node can write several destination nodes into the destination field, hence multicast or broadcast is possible.

In the distribution phase packet, the "result of requests"-field contains the outcome of each node's request. This is the only field, in this phase, which contains network arbitration information. The other fields are used for services such as reliable transmission ("ACK/NACK"- and "flow control" fields) and global reduction (the "Extra information"-field). These are described later.

During the collection phase, the current master node initiates by generating an empty packet, with a start bit only. The packet is transmitted on the control channel. Each node appends its own request to this packet as it passes and then passes the packet on to the next node. The master node receives the complete request packet and processes it to determine which requests are possible to fulfil, see Figure 5.

The time until deadline (referred to as laxity) of a packet is mapped into a four bit number in the priority field of the request from a node. A shorter laxity of the packet implies a higher priority of the request. The result of the mapping is written to the priority field. One priority level is reserved (15 in the proposed implementation of the protocol) and used by a node to indicate that it does not have a request. If so, the node signals this to the master by using the reserved priority level and writes zeros in the other fields of the request packet. Request priority is a central mechanism of the TCMA protocol. A larger priority field, in each request (see Figure 5), is possible and would provide higher resolution of priority and possibly also positively affect performance. Two mappings between deadline and priority, logarithmic and linear have been simulated, see Figure 6. Results show a negligible difference in performance of throughput, packet-loss, and latency. Further evaluation of how the performance is affected by different mappings and priority resolution is not included in this report. Logarithmic mapping is used in the simulations in Section 5. This mapping gives higher resolution of laxity, the closer to its deadline a packet gets. For the linear transformation, deadlines longer than 14 slots are all mapped to priority level 14.

Packets queued locally in nodes are sorted by laxity and distance. Each node selects its most urgent packet as the request. In the case that there are several packets that are equally urgent, the packet that is destined furthest and possible to transmit in the next slot is selected. Nodes may not request transmission of a packet that will pass the master since the clock signal is interrupted there and data cannot pass. A node will only make a request that may be possible to fulfil regarding RTVCs (see sections 9.3) in the own or other nodes that would use links in the path of the packet that the node

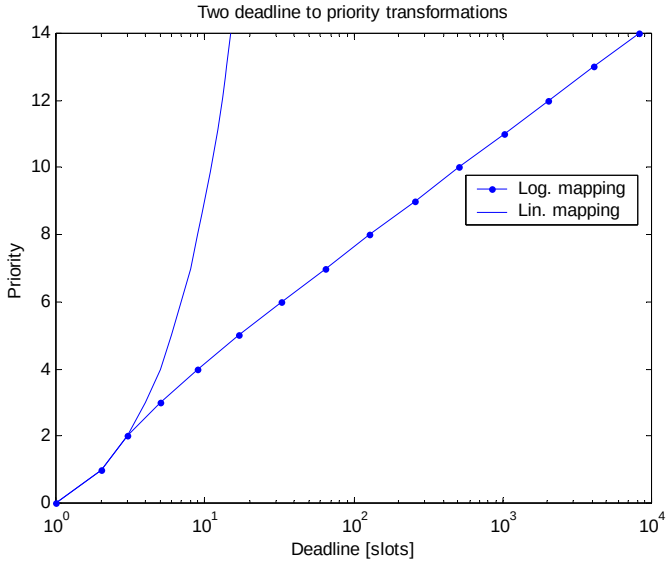

Figure 6: Linear and logarithmic mapping of deadline-to-priority.

would want to send. This implies that a request will only be rejected if requests from other nodes are more urgent. Slots belonging to RTVCs do not need to be "requested" since all nodes know which slots are already reserved.

When the completed collection phase packet arrives back at the master, the requests are processed. There can only be *N* requests in the master, as each node gets to send one request per slot. The list of requests is sorted in the same way as the local queues. The master traverses the list, starting with the request with highest priority (closest to deadline) and then tries to fulfil as many of the *N* requests as possible. In case of priority ties, the request with the largest distance to its destination is chosen. If there still is a tie, then requests from upstream nodes (closer to the master) have priority over other requests.

When the master has scheduled the requests, it distributes the result to all nodes in the distribution phase. In this phase, the master node (only) has the possibility to use the other fields in the distribution phase packet, such as sending acknowledges for packets sent during the previous slots. For further explanation of this, see Section 9.4. When all nodes have received the results of the request, each node is ready for the beginning of the next slot where data may be transmitted. A request was granted if the corresponding bit in the "request result field" of the distribution phase packet contains a "1".

9. USER SERVICES

The user services described below are: best effort messages (Section 9.1), non real-time messages (Section 9.2), real-time virtual channels (Section 9.3), guarantee seeking messages (Section 9.4), reliable transmission (Section 9.4), barrier synchronisation (Section 9.4) and global reduction (Section 9.7).

9.1. Best effort messages

The TCMA protocol supports best effort messages (Arvind, Ramamritham et al. 1991). The best effort message service implies that messages at nodes are accepted for transmission but are not given any guarantees. However, the network will try to meet all deadlines. Messages are queued in the node according to deadline and transmitted according to deadline order in a global queue generated by the TCMA protocol. Transmission of the message may however fail e.g. due to congestion in the network. When the deadline of a message expires the user may optionally be notified. The user can choose to have the message removed from the queue or to keep the message queued for sending despite its expired deadline. The choice depends on the application that uses the messages. Communication with best effort message is evaluated by simulation in Section 11.

9.2. Non real-time messages

The non real-time message service is suitable for users that do not require real-time constraints at all. Such a user could be a bulk file transfer. The messages do not contain any timing requirements and are queued according to destination and transmitted in first come first served order. Non real-time messages are similar to best effort messages but have lower priority and are only sent when there is no best effort message to be sent from the node.

9.3. Real-time virtual channels

Logical connections with guaranteed bit rate and bounded latency can be realised in the network by using slot reserving. Such connections are referred to as RTVCs (Arvind, Ramamritham et al. 1991), (Ferrari and Verma 2002). Either the whole ring is reserved for a specific node in a slot, or one or more segments of the ring are dedicated to specific node(s). A single slot can be allocated for several RTVCs simultaneously since there can be separate transmissions in different segments of the ring. Slots are organized into cycles with a predefined number of slots. Nodes keep track of the current slot index in the cycles. A slot that has been reserved for an RTVC, guarantees that transmission is possible every cycle, thus guaranteed bit rate. Several slots may be reserved for an RTVC in order to increase the guaranteed bit rate. Initially each node has J non-reserved slots where it is master, giving a cycle length of $N \cdot J$ slots, where N is the number of nodes. In order to always have bandwidth for best effort data-traffic controlled by the TCMA protocol, only $J - 1$ slots are reserveable.

When a node wants to reserve a slot for an RTVC, it searches for slots where the required links are free, so allocation of a new segment can be done. First, the node's own slots are searched. If not enough slots could be allocated for the reservation, the search is continued in other nodes. In this case, the node broadcasts a packet containing a request to all other nodes to allocate the desired segment in their slots. The packet contains information about the links required and the amount of slots needed. Each node then checks if any of its own slots have the required free links. All nodes sends a packet back to the requesting node to notify which slots, if any, that have been allocated. When the requesting node has received the answers, it decides if it is satisfied with the number of allocated slots. If not, it sends a release packet. Otherwise, it can start using the reserved slots immediately. If so, the node notifies other nodes by broadcasts the details of the RTVC. It should also send a release packet if more slots than needed were allocated. A node wishing to set up an RTVC can thus "borrow" slots from other nodes. Internally in the nodes, each RTVC has its own queue for packets. All nodes have information about which slots are reserved, for RTVCs between pairs of nodes.

9.4. Guarantee seeking messages

Guarantee seeking messages are sent within an RTVC that is owned by a node. Such messages normally have hard timing constraints. If the communication system cannot guarantee the timing constraints of a guarantee seeking message, the message will not be sent. The owner of the message will be notified of the outcome. In the proposed network, a guarantee is only given if enough deterministic bandwidth (slots) through RTVCs is owned by the node. The guarantee seeking message must be schedulable before its deadline. Normally an RTVC is first set up for the required segments of the ring (possibly the whole ring). The reservations for guarantee seeking message that are made are tracked by the node.

9.5. Low-level support for reliable transmission

The proposed network has low level support for reliable transmission (Bergenhem and Olsson 1999). Network control information, acknowledge/negative acknowledge and flow control, is sent in the control channel instead of in ordinary data packets. This results in less or no overhead in the data channel, i.e., better bandwidth utilisation. The field in the control packet named ACK/NACK contains bits that correspond to the N packets that may have been received by the current master during the previous N slots. The ACK/NACK information is also sent when a node is master. If a packet was correctly received (correct checksum) a "1" is written in the position of the ACK/NACK field that corresponds to the slot that the packet was received. If a faulty packet or no packet was received, a "0"

is written. All nodes must keep track of their transmissions and can therefore resolve the meaning of the bits in the ACK/NACK field. In this way the nodes can be notified if their packet was correctly received or has to be retransmitted. The latency for a node to send and receive ACK/NACK is bounded and deterministic which is desirable.

The $4(N-1)$ bits in the flow control field relate to independent logical connections. Put simply, each node can have up to four logical connections in average, with low-level support for flow control, depending on the implementation. The master sets the bit corresponding to a logical connection to "1" if it is to be halted temporarily; else the bit is set to "0". This service may be combined as required with any of message sending services.

9.6. Barrier synchronisation

Barrier Synchronisation (BS) is an operation to control the flow of processes in a distributed processing system. A logical point in the control flow of an algorithm is defined, at which all processes in a processing group must arrive at before any of the processes in the group are allowed to proceed further. When, during execution, a BS point is encountered in the application program, the node broadcasts the encountered BS_id in the control packet when the node is master. In this way all nodes are notified that the node has reached the BS point. Nodes belonging to a different BS group can ignore the broadcast, but nodes belonging to the same group, i.e., has the same id, will make a note in an internal table. The control packet contains a field in which BS_id can be sent (see Figure 5). The id field contains 8 bits, which permits ids ranging from 1 to 255. When the field is zero no BS command is sent.

When a node participating in the BS group has received the correct BS_id from all the participants it knows that all the other nodes are at the same executing point and may proceed. The worst case latency, for a node that reaches the BS point until it can broadcast this to the other nodes, is one cycle. Clearly the implications of sending BS information in the control channel is both bounded latency and better bandwidth utilisation of the data channel. The whole BS mechanism is handled by the communication interface, transparent to the calling user processes.

In the description above, static allocation of barrier synchronization BS_ids is assumed. The programmer (or the compiler) allocates the required parameters for BS and GR off-line, before runtime. With minor adjustments, dynamic allocation is also possible but is not investigated further in this report.

9.7. Global reduction

Global Reduction (GR) is similar to barrier synchronisation where data is collected from distributed processes when they signal their arrival at the synchronisation point. A global operation, e.g., sum or product, is performed on the collected data so that only a single value is returned. At the end of the GR all participating nodes have access to the same data. As in the case of BS we assume that the programmer (or the compiler) statically allocates the necessary parameters, off-line, before runtime.

The GR command requires the following parameters: operation, length, data type, GR_id, and the ids of the participating nodes. The last two parameters are similar to the parameters for the BS command. The operation parameter tells the nodes what operation (sum, product, max, min, etc.) should be performed on the received data. The data type parameter indicates how the data field in the control packet (see Figure 5) is to be interpreted and the length parameter tells the length of the data field. In the studied case, the data field is 320 bits long and may facilitate the transfer of, e.g., up to five double precision floating point numbers (IEEE-754). Having multiple data items in the data field gives the opportunity to have, e.g., vector operations. Other data types may also be distributed. Except for the additional fields and the global function, the nodes treat GR commands in the same way as BS commands. Further on the same reasoning of performance advantages also holds for GR.

The type bit tells whether the control packet contains a BS or a GR command, and hence whether data is contained in the data field or not (see Figure 5). The data field is currently only used for data reduction, but may be used for, e.g., sending short messages.

10. IMPLEMENTATION ASPECTS

T_{tcma} is the time required to complete network arbitration according the TCMA protocol. This also sets the minimum possible slot length for the network. T_{tcma}, scales for increasing network length and number of nodes as follows:

$$T_{tcma} = T_{collection} + T_p + T_{selection} + T_{distribution} \quad (1)$$

The master requires a finite time for processing the collected requests to select which requests may be sent. Part of this processing is sorting the incoming requests. This can be done as they arrive by checking them bit by bit and thus the sorting time is incorporated in the time for the collection phase, $T_{collection}$. When requests have been sorted, the list of requests has to be traversed to select which packet that may be sent. The time for this is denoted as $T_{selection}$ and is assumed to be $N \cdot 30$ ns. Propagation delay, T_p, is part of required arbitration time only once since the master depends on feedback, i.e., the requests from the other nodes only in the collection phase. L is the total length of the ring, while V is the propagation speed through the optical fibre and is assumed to be $2 \cdot 10^8$ m/s. The delay through each node (approximately 1 bit time per node) is neglected in the propagation delay. The total propagation delay around the ring is:

$$T_p = \frac{L}{V} \quad (2)$$

From Figure 5 one can see the size of the fields in the current implementation of the control packets which lead to:

$$T_{collection} = \frac{1 + (2N + 4)(N - 1)}{C} \quad (3)$$

and

$$T_{distribution} = \frac{6N + 325}{C} \quad (4)$$

where N is the number of nodes and C is the bit rate of the links. For the Motorola Optobus, C =800 Mb/s. The minimum slot length (T_{tcma}) is plotted in Figure 7. The minimum slot length increases with increased network length (increasing propagation delay) and number of nodes (each node contributes with delay). This is expected since the network requires feedback from all nodes in its medium access protocol.

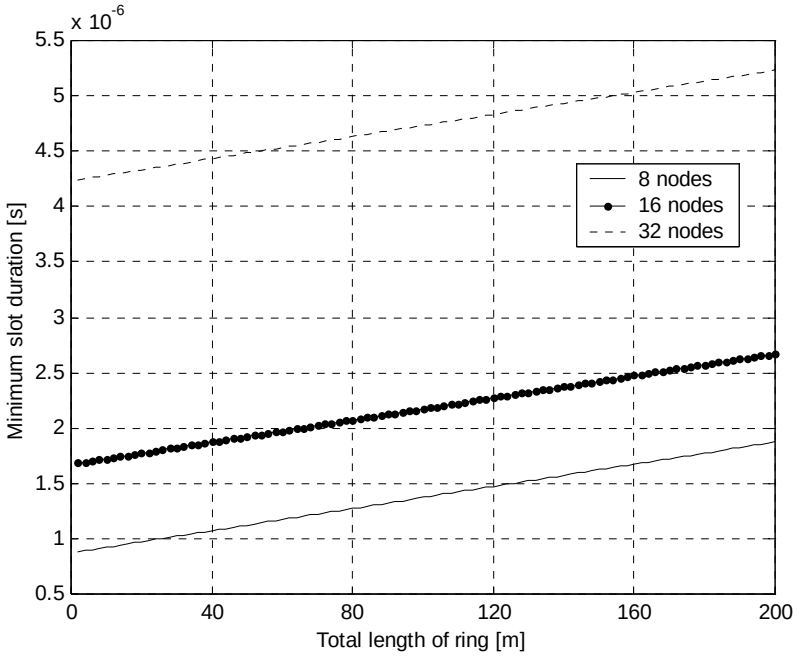

Figure 7: The minimum slot length is shows for three different numbers of nodes.

11. SIMULATION ANALYSIS

The simulation analysis of the TCMA protocol is done with a discrete time computer simulation. Networks of 8, 16, 32 and 64 nodes were simulated. Each best effort packet is given a relative deadline at generation which is decremented each time slot. The packet is queued until it is either sent successfully or, when deadline reaches zero, is deemed lost and removed from the queue. On request from the application a packet could be kept in the queue even after the deadline is passed but this is not simulated.

Some further assumptions for the simulations:

- Messages are one packet long and take one time slot to send. The term packet and message is therefore used synonymously.
- Uniform data-traffic is assumed, i.e., all nodes have equal probability of message generation and uniformly distributed destination addresses. This implies that, on average for an ideal ring network, it is theoretically possible to transmit two packets each slot, since the packet on average is destined "half way" around the ring, i.e. $N / 2$ hops. However, this disregards protocol effects, which lowers the average utilisation, which we will see later. An example of protocol effect is that a node may not send past the master since the clock is interrupted there. The pipelined ring topology of the network suggests that it is very effectively utilised when data-traffic is mostly destined one hop to the nearest neighbour such as in some types of radar signal processing (Jonsson, Svensson et al. 1997) (Taveniku, Ahlander et al. 1998). If this special mode of data-traffic were simulated, which it is not, the effect would simply be higher throughput because of aggregation, i.e., more packets would be sent during one slot.
- Messages were generated according to a Poisson process and all messages were of single destination type.
- The deadline of all best effort packets is set at generation to 800 slot times ahead, which would equate to a deadline of 4 ms with a slot time of 5 μs (4kBytes per packet if Motorola Optobus is used).
- Physical effects, such as the propagation delay through the optical fibre, and the time required to

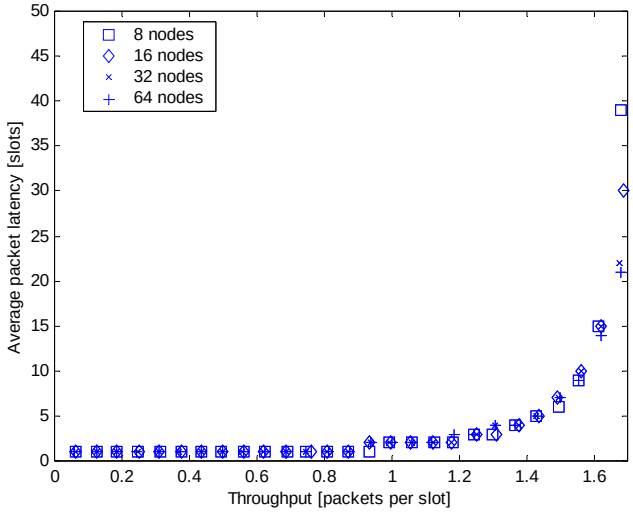

Figure 8: Best effort packet latency vs. throughput for a varying number of nodes.

detect the end of a slot, were assumed to be less than one slot time and are therefore neglected.

- Message latency is defined as the time elapsed from the moment a message is generated until the entire message is received in the receiver.
- Infinite size of message queues is assumed.
- The simulator is run for a total of 100 000 slot times and starts to log statistics at 20 000 slot times.

- The “total packet generation intensity” is the generation intensity for the network regarded as a whole, not of the individual nodes. That is, the sum of all individual packet generation intensities at nodes is the total packet generation intensity.

Figure 8 shows the best effort packet latency for varying levels of network throughput. At a useful level of packet latency, the network has an average throughput of approximately 1.6 packets per slot. This is 60 % better than the theoretical limit, of one packet per slot, for networks without spatial reuse. Similar results for varying number of nodes are obtained.

Figure 9 shows the packet throughput against packet generation intensity. Throughput has a linear relation to packet intensity up to the point of saturation, which can be seen in the figure as the point on the plot where throughput starts to decrease with increasing packet intensity. As packet intensity passes the point of saturation it becomes increasingly difficult for the packet transmission scheduler to

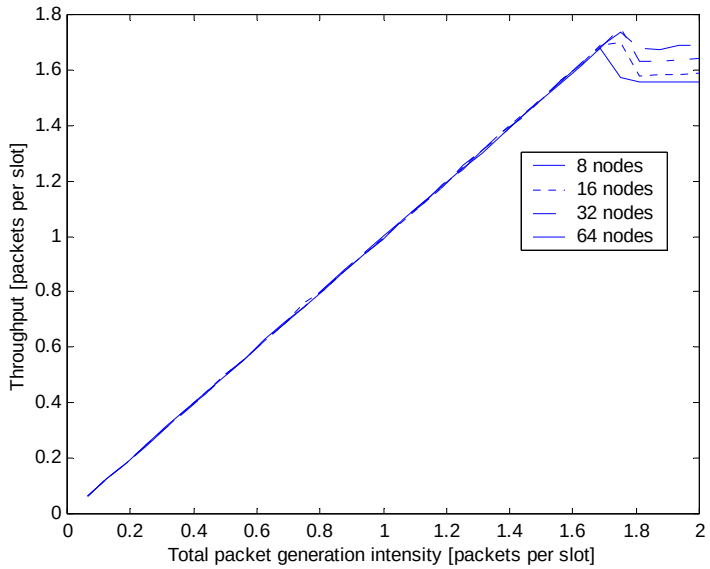

Figure 9: Total throughput of best effort packets vs. packet generation intensity.

effectively utilise the bit rate of the network. This is because it is always increasingly difficult to schedule packets the further their destinations are (Wong and Yum 1994). When the network is saturated each node’s queues will tend to always contain far destined packets with short deadlines. These cannot always be scheduled together with the shorter destined packets because of conflicting links. When two packets have the same priority level the packets with the further destination has priority. Thus slot utilisation will decrease which is expressed in the decline of the throughput after the “summit” in the figure. The simulation shows that the utilisation of slots does not perform as well as may be theoretically expected (two packets transmitted per slot, because of pipelining). This is attributed to the policy of using deadline to selecting packets for transmission. The policy is not as bandwidth conserving as, e.g., the pure FWS (Furthest Within Segment) policy (Wong and Yum 1994) which can achieve higher average throughput. This is the cost of having good support for real-time data-traffic and fairness as will be seen shortly.

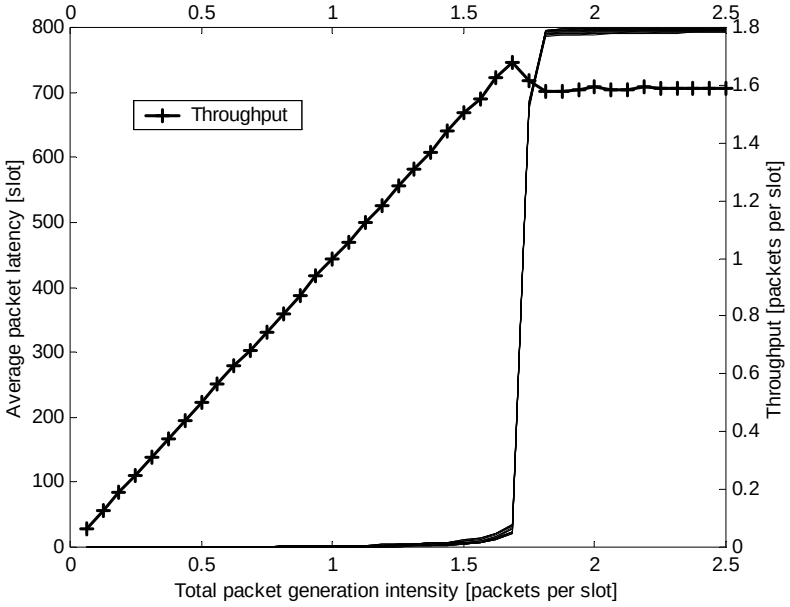

Figure 10: Packet throughput vs. packet generation intensity for 16 nodes. The plain plots represent latency in slots vs. packet generation intensity.

Figure 10 shows total packet throughput (crossed plot) and average packet latency (plain plots) for each destination distance plotted against packet generation intensity. The simulation is for 16 nodes. Concluded from this simulation is that TCMA treats packets fairly regardless of the distance to the destination even when the network approaches and is saturated. Observe that there are 15 (for $N-1$ different distances to destination) plots for latency but that these overlap and appear as one plot.

12. SUMMARY OF TCMA

TCMA is a medium access protocol with global deadline scheduling for an optical ring network. The user services include best effort messages, non real-time messages, guarantee seeking messages, real-time virtual channels (RTVCs), group communication such as barrier synchronisation and global reduction, and services for reliable transmission. RTVCs and guarantee seeking messages are realised through slot reservation. Simulation results of the medium access protocol with best effort data-traffic have been presented. The protocol is shown to be fair even when the network is saturated and it supports spatial reuse to achieve throughputs higher than one. The network is suitable for applications such as in embedded systems, e.g., for use as interconnection network in a radar signal processing system, or as a high performance network for use in a LAN environment. It can be built today using fibre-optic off-the-shelf components.

PART B:

THE CONTROL CHANNEL BASED RING NETWORK WITH EARLIEST DEADLINE FIRST SCHEDULING PROTOCOL

13. INTRODUCTION TO CCR-EDF

In this part, a novel fibre-ribbon ring network and its medium access protocol is presented. The network is called CCR-EDF (Control Channel based Ring network with EDF scheduling). The CCR-EDF network consists of a network-architecture as described earlier and a medium access protocol. Together they form a network that offers services to the user such as scheduling of hard real-time data-traffic.

The CCR-EDF network is similar to the TCMA network (Bergenhem, Jonsson et al. 2001) that was described in the preceding part of this report and has similar target applications such as radar signal processing. The architecture of the new protocol is that same as before with minor changes. The TCMA network has a pessimistic worst-case schedulability bound which makes it unsuitable for hard real-time data-traffic, because of very low guaranteed utilisation. However, the TCMA network is suitable for best effort data-traffic because of the priority mechanism in the medium access protocol. The CCR-EDF medium access protocol uses the deadline information of individual packets, queued for sending in each node, to make decisions about who gets to send. The decision is made in a master node; a role that is circulated.

Other networks with EDF-scheduling include (Shin 1991) and (Kandlur, Shin et al. 1994). Advantages of the class of fibre ribbon ring networks (Jonsson and Bergenhem 2001) (including the network presented here) over these other networks include the use of high-bandwidth fiber-ribbon links and the close relation between a dedicated control channel and a data channel without disturbing the flow of data-packets. This implies that transmission of control and data are overlapped in time. With less header-overhead in the data-packets the length of communication slots can be shortened to reduce latency without sacrificing bandwidth utilization. Also, the separate clock and control fibres simplify the transceiver hardware implementation.

The medium access protocol provides the user with a service for sending periodic messages in logical real-time connections that have been checked for feasibility with an admission control mechanism. The messages in the logical real-time connections are then scheduled with earliest deadline first. Logical real-time connections may be added and removed from the system during runtime. The global deadline scheduling is a distributed scheduling mechanism that is built into the medium access protocol. No further software in upper layers is required for this service. Other networks may have upper layer protocols added to them to give them better characteristics for real-time data-traffic, but it is difficult to achieve fine deadline granularity by using upper layer protocols.

The CCR-EDF provides similar communication services and services for parallel and distributed processing as the TCMA network (Bergenhem, Jonsson et al. 2001). There are slight differences in how the communication services are realised. The CCR-EDF protocol is analysed through simulation with a radar signal processing application.

14. THE CCR-EDF NETWORK ARCHITECTURE

As mentioned earlier the networks architecture presented in this report has a distributed clocking strategy. During each slot, one node has the task of generating the clock signal for the entire network. This node is called the master node. The clock signal from the master node is propagated in the clock fibre around the network over $N - 1$ hops i.e. almost back to the source node that generated the clock signal. The end of a slot is detected implicitly by detecting that the clock signal has ended. This signals that the role of master is to be handed over to another node in the network. In the implementation of distributed clock strategy found in (Jonsson 1998) and in (Bergenhem, Jonsson et al. 2001), hand over is always to the next downstream node. The advantage of this is simplicity; the clock hand over time between slots is constant. Therefore there will be the same time-gap between all slots, as long as the link length between each pair of neighbours is roughly the same. However, the simple clock hand over strategy has drawbacks concerning scheduling anomalies. It is shown in (Bergenhem and Jonsson 2002) that analysis of the network is complicated by the clocking strategy. The worst-case performance is pessimistic to such a degree that worst-case analysis is of little use. This deficiency is

attributed to the unsuitable clocking strategy which causes priority inversion. Highest priority messages may be pre-empted in some situations due to clock interruption. If the clocking node is in the path of the highest priority message then it cannot be sent during that slot. This situation occurs because clock hand over is done in a round robin fashion that doesn't take into account the arrival of a highest priority message in a node. The clocking strategy proposed in this report, does not suffer from the same pessimism.

For the CCR-EDF network an alternate clocking scheme is proposed. For the following discussion, the term master node is exchangeable with "the node with clocking responsibility etc." That is, the master node also clocks the network. The clocking strategy functions as follows. During a slot, the node that has the highest priority message, according to the arbitration process has the responsibility to clock the network. In the following slot, the clocking responsibility is handed over to the node that has the highest priority message in that slot. This may be another node or the same node as in the previous slot. Thus clock hand over is always done in accordance with the result of the medium access arbitration process, described further in next section.

The result of the arbitration process is knowledge of messages at the head of the local queues in all nodes, and therefore also knowledge about which node has the highest priority message in the system. The current master distributes this information to all nodes. A distribution packet is sent so that the end of the packet corresponds with the end of the slot. This implies that when the master stops the clock at the end of the slot, all nodes have the information that they need to perform clock hand over that takes place in the gap between slots. The node that has highest priority in the coming slot detects when the clock signal is stopped and assumes the master role. The highest priority node knows that it will be master because of the information received in the distributions phase packet.

The node that is master will also contain the highest priority message. Since this node has responsibility for generating the clock this message will always be possible to send. This is because the node will at most send $N - 1$ hops (where N is the number of nodes) and will never have to transmit past a master, i.e. cross the clock break at the master node.

A drawback with this method is that the clock hand over time, the gap between slots, is not constant. The size of the gap between slots depends on the distance to the next master, which will vary between 1 and $N - 1$. See also Section 17, on timing analysis.

In the CCR-EDF network, access to the network is divided into slots as described previously. There is no concept of cycle since a cycle cannot be defined. In other cases a cycle would be e.g. when all node have been master once. However, in CCR-EDF the master role is not shared equally among nodes but is given to the node with the highest priority message. The minimum slot length is discussed Section 17.

Table 1: The allocation of priority levels to user services. A higher priority within the traffic class implies shorter laxity and a more urgent message

0	Nothing to send
1	Non Real-Time
2-16	Best Effort (soft real-time)
17-31	Logical real-time connection

15. THE CCR-EDF MEDIUM ACCESS PROTOCOL

The medium access protocol has two main tasks. The first is to decide and signal which packet(s) is to be sent during a slot. The second task is that the network must know exactly which node has the highest priority message in each slot. This is to be able to perform clock hand over to the correct node. Therefore, this information is included as an index in the distribution phase packet.

The protocol is time division multiplexed into slots to share access between nodes as before (see section 2). Medium access control is organised into two phases: the collection phase and distribution phase. Both these two phases occur during one slot. Arbitration occurs in the time slot prior to the actual transmission slot.

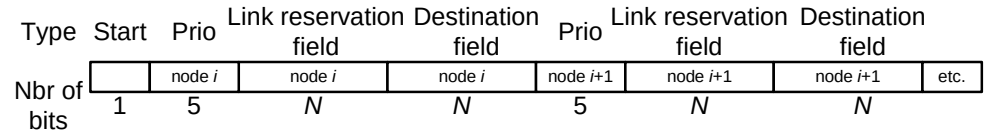

Figure 11: Contents of the CCR-EDF Collection phase packet. The figure shows that one request per node make up the complete packet.

In the collection phase, the current master initiates by generating an empty packet, with a start bit only, and transmits it on the control channel. Each node appends its own request to this packet as it passes. The master receives the complete request packet (see Figure 11) and sorts the requests according to priority (urgency). This is similar to the TCMA distribution phase packet. In the event priority ties the index (known by the master) of the node resolves the tie. The request from each node contains three fields for: priority, link reservation and destination.

The network can handle three classes of data-traffic: logical real-time connection, best effort, and non-real-time. Which class of data-traffic that a certain message belongs to, is signalled to the master with the priority field in the request, see Figure 11. Table 1 shows the allocation of the priority field to each user services in the network. The time until deadline (referred to as laxity) of a message is mapped, with a certain function, to be expressed within the limitation of the priority field. This applies to both logical real-time connection and best effort data-traffic. A shorter laxity of the packet implies a higher priority of the request. The result of the mapping is written to the priority field. One priority level is reserved (0 in the proposed implementation of the protocol) and used by a node to indicate that it does not have a request. If so, the node signals this to the master by using the reserved priority level and also writes zeros in the other fields of the request packet. Observe that messages that are part of logical real-time connections always have higher priority than any other service. However, a possible situation, considering spatial reuse, is that a best effort message uses the spatially reused capacity and may be transmitted simultaneously as a logical real-time connection message. The best effort message does not affect the logical real-time connection message. Observed locally in a node, best effort messages will only be requested to be sent if there is no logical real-time connection message queued. The same applies to non real-time message. They are only sent if there are no best effort and no logical real-time connection messages. Request priority is a central mechanism of the CCR-EDF protocol. Deadlines are mapped with a logarithmic mapping function to priority. Note that best effort messages also have deadline constraints and are hence considered to be soft real-time applications in contrast to logical real-time connections which have higher priority and hence for hard real-time applications.

When the completed collection phase packet arrives back at the master, the requests are processed. There can only be *N* requests in the master, as each node gets to send one request per slot. The list of requests is sorted in the same way as the local queues. The master traverses the list, starting with the request with highest priority (closest to deadline) and then tries to fulfil as many of the *N* requests as possible.

The second phase, the distribution phase, is described as follows. The master sends a distribution phase packet on the control channel, see Figure 12. This is similar to the TCMA distribution phase

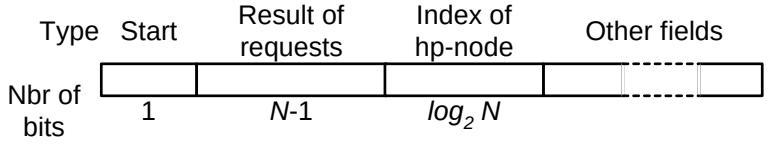

Figure 12: The CCR-EDF distribution phase packet. The field labeled "index of hp-node" contains the index of the node that has the highest priority message. Several fields at the end of the packet have been omitted because they are not relevant to the discussion.

packet. The packet is received by all nodes and contains the result of the arbitration. Each node's request is either accepted or denied. The packet also informs on which node contains the highest priority message in the slot. A request was granted if the corresponding bit in the "request result field" of the distribution phase packet contains a "1". The protocol also has a feature to permit several non-overlapping transmissions, that is grant several requests, during one slot. This property is called spatial reuse and can used during run-time depending on the traffic pattern. The distribution phase packet also may contain other information such as acknowledgement for transmission etc. These are further described in (Jonsson, Bergenhem et al. 1999) and will not be part of the discussion here.

The addition of an index that points to the node that has highest priority in the current coming slot, see Figure 11, enables all nodes to know who will have the highest priority message in the coming slot and that that node will assume the role as master and clock the network. The index field needs to be $\lceil \log_2 N \rceil$ bits wide to represent numbers up to N. When all nodes have received the distribution phase packet, and hand over has taken place, the new slot commences and data may begin to flow in the data channel.

16. USER SERVICES

The user services available in CCR-EDF for sending message are: logical real-time channels, best effort messages and non real-time messages. They are similar to the services available in TCMA with the exception that they are not based on pre-reservation of slots. In CCR-EDF the message services are based on EDF-scheduling. The services for parallel and distributed computing are the same as for the TCMA protocol.

17. TIMING PROPERTIES

The worst-case hand over time is when hand over is to the node that is $N - 1$ hops or segments away. This corresponds to hand over to the upstream neighbour node. The general equation for hand over time is:

$$t_{handover} = P \cdot L \cdot D \tag{5}$$

where P is the propagation delay of the light per meter in the fibre, L is the average length (in meters) of the segments and D is the number of segments that are traversed. In the worst-case, D is equal to $N - 1$.

An example of the clock hand over process is given in Figure 13 and Figure 14. In Figure 13, Node 1 had the highest priority during slot $i - 1$. The network arbitration process discovers that node 3 will take over the role as master. Figure 14 points out some relevant timing-points in the hand over

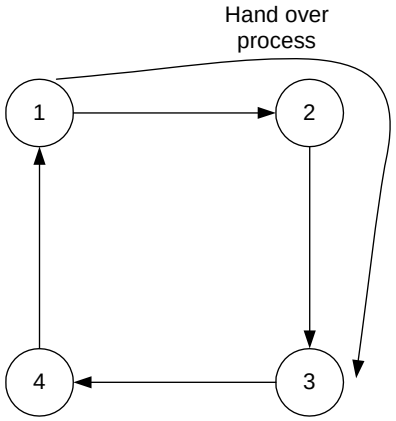

Figure 13: In the current slot, node 1 is master. The arbitration process is run during slot $i - 1$ and discovers that node 3 has highest priority and will be master in slot i.

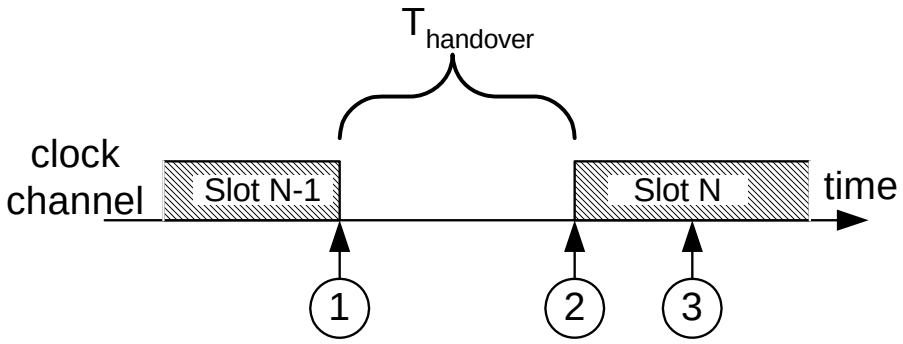

Figure 14: Shows important timing-points during the hand over procedure. The numbers are for reference. Note that the figure is not drawn to scale.

process. At point 1, the entire distribution packet has been sent. Node 1 stops generating the clock-signal after one more bit time. At point 2, Node 3 receives the distribution packet and discovers that it will be master in the next slot. It senses that the clock has stopped after one more bit time. This is the trigger for the end of the slot and the node assumes the role as master. The node then starts clocking again with only one bit time gap with no clock. At point 3 Node 4 receives the distribution packet and discovers that it will not be master. Node four receives the clock signal again after one bit time after the distribution phase packet. Node four may then transmit if it was granted permission to send.

The general equation for minimum slot length, because the collections phase must be finished before the end of the data transmission in one slot, is:

$$t_{minslot} = N \cdot t_{nodes} + t_{prop} \quad (6)$$

where N is the number of nodes in the network, t_{nodes} is the delay experienced by the control packet (during the collection phase) through each node and t_{prop} is the propagation delay through the whole ring.

18. ASSUMPTIONS FOR THE SCHEDULING FRAMEWORK

The network provides a feature for spatial reuse that is used during run-time. However, the benefit of the feature is not taken into account in the analysis which assumes that only one message may be transmitted per slot. The reason for this is that the degree of spatial reuse in a specific slot can only be known statistically. One message per slot can always be guaranteed. During run-time spatial reuse can be used and always results in positive effects.

For scheduling it is assumed that the smallest time unit is a slot. An assumption concerning messages is that the relative deadline is equal to the period of the logical real-time connection. The maximum delay that a message may encounter is

$$t_{maxdelay} = t_{deadline} + t_{latency} \quad (7)$$

where $t_{deadline}$ is the deadline of the message, and $t_{latency}$ is the worst-case latency that the message may experience. The worst-case latency is:

$$t_{latency} = 2 \cdot t_{slot} + t_{handover\_max} \quad (8)$$

The time slot delay assumes that one slot is just missed and one slot is needed for arbitration, while the hand over time assumes worst-case delay for the master to hand over to the correct node. On the user level, the deadline is $t_{maxdelay}$, i.e., the deadline of the message is used for scheduling, but the user perceives $t_{maxdelay}$. Thus the scheduling is not affected by $t_{latency}$

19. THE SCHEDULING FRAMEWORK

With the following discussion a test to find out if a new logical real-time connection can be accepted and guaranteed, will be presented. After the logical real-time connections are accepted their messages will be scheduled by earliest deadline first.

The basic EDF feasibility test is as follows:

$$\sum_{\forall i} \frac{e_i}{P_i} \leq U_{\max} \tag{9}$$

where e_i is the size (number of slots) required for a message in logical real-time connection *i* and P_i is the period of messages in logical real-time connection *i*. Their quotient is the utilisation of logical real-time connection *i*. $U_{\max}$ is the worst-case maximum utilisation, i.e., the lowest utilisation that can be gained at full load, due to always experiencing a worst-case hand over delay between the slots. Since message parameters do not affect this, it is also considered to be the worst-case throughput at full load. However, actual throughput depends on the number of messages in the network. The worst-case is:

$$U_{\max} = \frac{t_{slot}}{t_{slot} + t_{\max\, handover}} \tag{10}$$

t_{slot} is the slot size and $t_{\max\, handover}$ is the worst-case time for clock hand over. Since there is a gap in between the slots, which cannot be used effectively, and considering the restriction of no spatial reuse, the quotient $U_{\max}$ is lower than one. The total bandwidth can be seen as the length of slots together with the gap. The effective bandwidth can be seen as the duration of the slots.

Below, an online schedulability analysis of periodic messages, referred to as logical real-time connections, will be discussed. The basic function is dynamic schedulability testing or online centralised admission control. The assumption for the analysis is that logical real-time connections arrive one at a time at any time even during run time. Another assumption is that logical real-time connections are well behaved, i.e., agreed parameters are always honoured by the transmitting node. A specific node in the system is designated to solely handle new logical real-time connections added to the system and to remove them when required. Communication with this node is handled with the best effort data-traffic user service.

The set ***Ma*** contains the logical real-time connections that have been tested for feasibility and are accepted. The admission test is as follows. If the utilisation of the logical real-time connections in ***Ma*** together with the new connection is below or equal to $U_{\max}$ (see Equation 10) then the new logical real-time connection is admitted into ***Ma***. The node that owns the new logical real-time connection is notified. The logical real-time connection can now be activated and may be used for data-traffic. If the utilisation of the new connection and ***Ma*** is higher than $U_{\max}$ then the new logical real-time connection is rejected. The requesting node is notified of the failure. The above procedure is repeated for every request for a logical real-time connection that arrives at the node responsible for admissions control.

20. THE RADAR SIGNAL PROCESSING CASE USED FOR SIMULATION

The studied application is radar signal processing (RSP). In the main and supporting algorithms we can identify several different types of data-traffic. Examples of data-traffic and what they may require are:

- The radar signal data itself has soft real-time requirements, but performance must be deterministic.
- The control data-traffic has hard real-time requirements.
- Other general data-traffic such as logging and long term statistics do not have any real-time constraints at all.

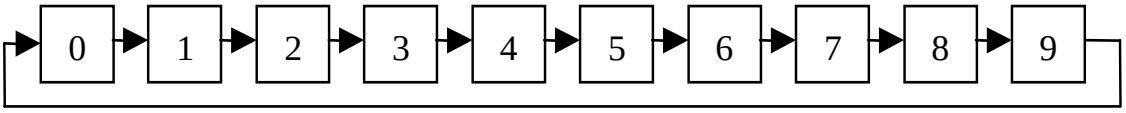

Figure 15: The straight-pipeline ”wiring” of BE-channels. Each arrow is a unicast channel.

These three data-traffic types are summarised in Table 2. System start-up, bootstrapping, firmware upgrading are tasks that don't require real-time support. However, some kind of performance analysis is needed in order to design for desired performance. Background to the assumptions in Table 2 are found in (Jonsson, Ahlander et al. 1996) and (Agelis, Jacobsson et al. 2002). The maximum latency through the pipeline is 100 ms, i.e. the delay from start of processing at the start of the pipeline until the result is available. The latency includes communications and processing. This latency is assumed to be composed of ca. 10 % communications delay (Jonsson, Ahlander et al. 1996). Both directions of the control data-traffic are periodic with a period of 1 ms. Processing is assumed to be co-ordinated in a master / slave fashion and that there is a node in the network that acts as master over the other processing nodes, i.e. slaves. It is assumed that the incoming data rate from the antenna is 6.0 Gb/s.

The arrival model for application data-traffic is related to how often a new data cube is accepted and to the pipelined fashion. However, data is sent with finer granularity (minimum ca 1500 bits per message), than a whole data cube between the nodes. We assume a target throughput that can deliver the whole data cube in time. The communication delay bound for the whole pipeline is 10 ms, with a new complete data cube arriving every 16 ms. The (soft) deadline for each packet in the data cube is 1 ms (10 ms / 10 steps). Priority based or soft deadline based service class (guaranteed service might be implemented on a higher level).

In Table 2 each of the three data-traffic types has a particular communication pattern. The control data-traffic is assumed to be “master – slave”, i.e. control is centralised at one node. Other data-traffic in the network is assumed to be broadcast. Radar data-traffic, the largest portion of the communication, is assumed to be to the next downstream neighbour node, i.e. straight pipeline. Processing is done in a pipelined fashion such that the RSP processing steps are divided among all modules equally. This is a simplification since different steps of processing will vary in computational intensity. Either nodes will have different capacity or several nodes are utilised during a processing step. Two alternate processing pipeline models are discussed in (Bergenhem, Jonsson et al. 2002). For the straight pipeline we assume a network with 10 identical modules and a required latency for the whole pipeline of 100 ms. This implies that 1 ms is allocated for communication per data cube per module, i.e. a total of 10 ms, and a total of 90 ms processing time, per data cube.

21. SIMULATOR SETUP

The details of the case study mentioned above are adapted to setup the simulation experiments. In all simulations the “wiring” of the channels for the three data-traffic types is the same, i.e. the same

Table 2: The data-traffic types in the RSP case study.

	Communication pattern	**Delay Bound**	**Data-traffic amount**	**Arrival model**
Control data-traffic	Master / Slave. Outgoing: broadcast Incoming: many-to-one	100 μs per direction (guaranteed or highest priority)	1 kByte	Periodic @ 1 ms, for both directions
Radar data-traffic	Straight pipeline	1 ms for each packet in the data cube	96 Mbit data cubes @ 62.5 Hz	A new data cube arrives periodically every 16 ms
Other data-traffic	Uniformly distributed unicast	Non real-time. No bound but a certain capacity is needed.	100 Mbit/s is assumed to be representative	Periodic with an average period of 50 ms is assumed to be representative

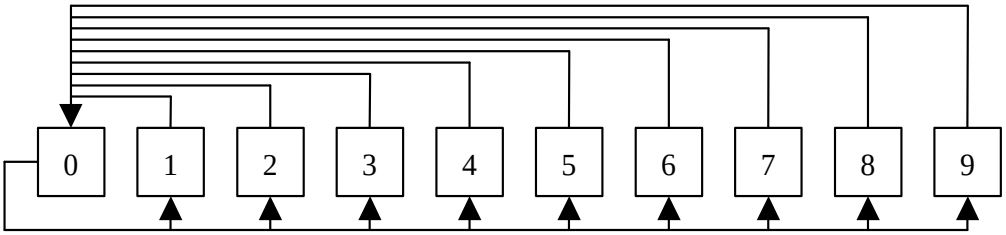

Figure 16: The ”wiring” of the LRTCs. Node 0 is the master node and the others are slaves. The arrows above the nodes indicate unicast transmissions to the master. The arrows below the nodes indicate a broadcast transmission from the master to the slaves; typically the sending of control information.

channel setup are re-used for each experiment. However, the load of a set of channels may change. Each slot (smallest simulated time unit) is equivalent to 1 μs and corresponds to 1 kByte of data.

The data-traffic in the RSP case definition consists of three main types (in decreasing order of timeliness requirements): Control data-traffic, radar data-traffic and other data-traffic. These types are conveniently serviced with the three data transports services offered by the CCR-EDF network protocol: Logical Real-time channels (LRTC), Best effort and non real-time, respectively.

The radar data-traffic has soft real-time constraints and constitutes the bulk of the data; in a pipelined fashion from node to node. Figure 15 depicts the straight-pipeline “wiring” of the Best effort (BE) channels. The data-traffic is periodic with a period of 16 ms, i.e. 16000 slots, and the default payload is 10 MByte, i.e. 10000 slots. During simulation, the payload may be varied from 1 MByte – 16 MByte. Best effort data-traffic has lower priority than LRTC, but higher priority than NRT data-traffic, see also section 15. If the BE channels are used by themselves in a system, the set of BE channels have a maximum throughput of 10 packets per slot with 16 MByte total payload. This is the theoretical limit due to spatial reuse. This limit is also later found by simulation.

The control data-traffic requires guarantees of timeliness and is therefore serviced by the LRTC service. Figure 16 depicts the wiring of LRTCs. The case study has a fixed set of LRTCs. These are organised as master / slave communication. The data is control information and therefore the amount is less than BE data-traffic. The data-traffic is periodic with a period of 100 μs i.e. 100 slots, and default payload is 1 kByte, i.e. one slot. During simulation, the payload may be varied from 1 kByte to 16 kByte. The LRTC data-traffic class is a guaranteed service with the highest priority. When used by themselves in a system, the set of LRTCs have a maximum throughput of 1.1 packets per slot per channel with 11 kByte payload per packet. This result is found in simulation 3 performed in Section 22.3 and is also further explained here.

The other data-traffic in the RSP-case is mapped to the NRT (non real-time) service. It is not organised in channels with fixed destinations. From each source the destination is uniformly distributed. Every node is a NRT data-traffic source and creates an equal load of unicast data-traffic. The deadline of NRT data-traffic is fixed to 100 slots and always has the lowest priority among other data-traffic. In the simulator, NRT data-traffic arrival is always Poisson distributed and is set at a high enough intensity such that NRT data-traffic will use all available communication capacity. When used by itself in a system, the NRT data-traffic will have a maximum throughput of 2 packets per slot. This is because a node on average will send a message to a destination that is “half-way” around the ring, which implies that two such transmissions are possible.

The case definition also states the data-traffic loads under normal operation. With this default mix of data-traffic the total throughput for all three data-traffic types is 5.97 packets per slot, see Table 3. The communications system is not saturated here and there is spare capacity. In Simulation 2, Section 22.2 it will be seen that the load of the network can be increased to achieve an even higher throughput. The default NRT throughput is given as 0.6 packets per slot. However, this throughput is entirely dependent on the available capacity and hence on the other data-traffic in the system. The maximum throughputs given in the table are dependent on the data-traffic pattern of the data-traffic type and relate to how much pipelining effect is present.

Each “simulation” consists of several, usually 16, data points. Each data point is an execution of the simulator with fixed parameters. To attain the curve, a parameter is varied and several runs later, the result is the data points that make up the curves in the presented figures.

Periodic data-traffic channels in the simulator are treated as follows. All data-traffic that a channel will send during one period will be generated and queued for sending at the start of the period. There is no "intelligent" functionality that smoothes the incoming data-traffic over the whole period. This lack of smoothing affects the maximum latency that a packet endures during a period. If the incoming data-traffic was smoothed, then the maximum would be closer to the average latency.

Table 3: Results from simulation experiments.

Parameter	Packets per slot
Total throughput with the default load of the RSP case . The system is not saturated here.	5.97
Default LRTC throughput	0.10
Default BE throughput	5.26
Default NRT throughput	0.6
Maximum throughput of LRTC traffic only	1.11
Maximum throughput of BE traffic only	10
Maximum throughput of NRT traffic only	2

22. SIMULATIONS

In this section three main simulations are presented. The first simulation concerns the latency and packet loss of BE data-traffic under increasing load of BE data-traffic. The second and third simulations have the mixture of data-traffic described in the case definition. In the second and third, the load of the BE data-traffic and LRTC data-traffic, respectively, is varied while the other is fixed. The effect on the mixture of data-traffic is studied. Important results from the simulations are summarised in Table 3. Observe that NRT data-traffic in the RSP case always strives to use all left over available capacity. Therefore it is meaningless to cite the throughput of this class. Further details of the simulations can be found in (Bergenhem and Jonsson 2003).

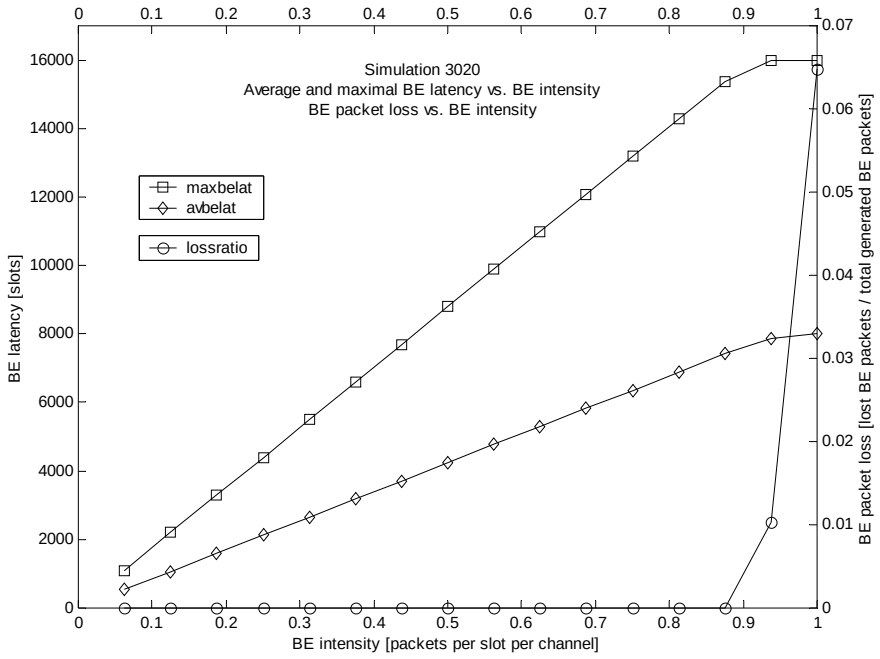

Figure 17: The figure shows the BE-traffic latency and BE-packet loss vs. the intensity of the BE-traffic.

22.1. Simulation 1

The “straight pipeline” of the case study definitions was implemented in the simulator and tested with different loads of BE data-traffic. The network setup for the simulation is as follows:

- All LRTCs at constant load: 1 data packet with a period of 100 slots period (the same as the RSP case definition).
- The load of each BE channel is varied in the interval 1 MByte – 16 MByte (6 % - 100 %).
- No Non real-time data-traffic in the system.

The result of the simulation is shown in Figure 17 and shows three plots: Maximum (maxbelat) and average BE latency (avbelat) vs. BE intensity and BE packet loss (lossratio) vs. BE data-traffic intensity. See the end of this section for a deeper discussion on data-traffic intensity. The maximum BE latency is the maximum latency that any BE packet was subject to during the simulation (for the respective intensity of BE data-traffic). As the BE data-traffic intensity increases, so does the maximum latency. The trend holds until the network cannot accept more BE data-traffic. At maximal BE intensity, 1 packet per slot per channel, the theoretical throughput (with no other data-traffic in the system) is 1 packet per slot per channel (total of 10 packets per slot for all BE channels). However, in the simulation, higher priority data-traffic also contends for capacity (LRTC data-traffic), and the throughput of BE data-traffic is therefore lower than the theoretical value. This can be observed since packets are lost at high intensity levels of BE data-traffic. At this point the trend of the latency curve changes and BE packets begin to be lost. The packet latency curve levels out (does not continue to increase) at a value of 16000 slots. This is because the latency cannot be higher than the deadline of the packets, i.e. 16000 slots. A BE-packet is considered lost if it remains queued for longer than its deadline.

Regarding BE intensity, observe that the x-axis in Figure 17 indicates the ratio per BE channel, not total BE intensity. The total BE intensity would at its peak approach 10, i.e. the number of nodes in the system. Observe also that the average latency of the BE data-traffic is always roughly half that of the maximum throughput.

22.2. Simulation 2

In this simulation, all three types of data-traffic are generated during the simulation. How the data-

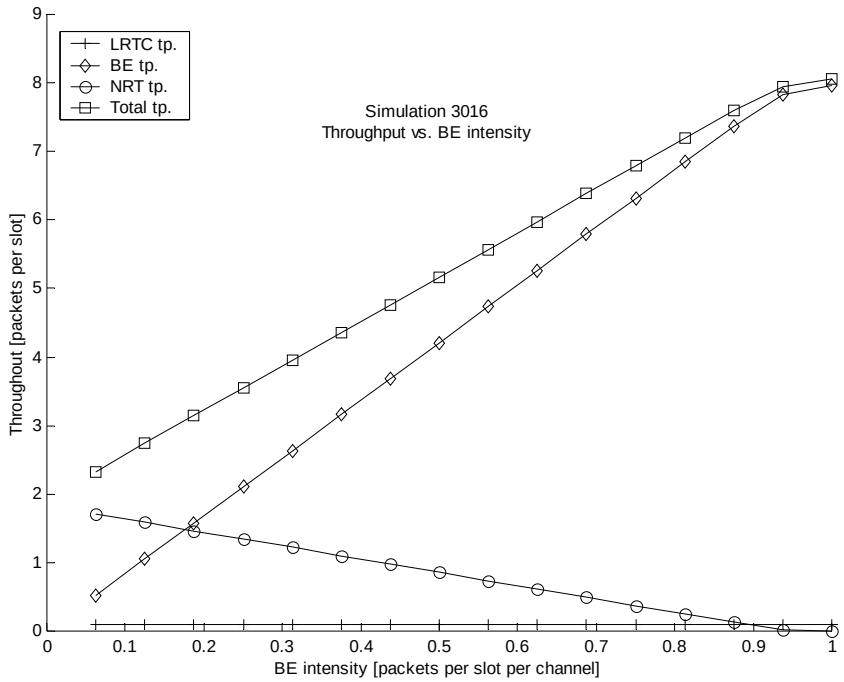

Figure 18: The throughput of the different traffic classes change as the BE intensity is increased. LRTC traffic load is constant.

traffic types affect each other is studied. In short the BE data-traffic is constant and the LRTC data-traffic is varied. The network set-up for the simulation is as follows:

- The LRTC set and behaviour of the set is the same as in simulation 1, i.e. constant.
- The set of best effort channels and behaviour of these are the same as in simulation 1, i.e. varied in the interval 1 MByte – 16 MByte (6 % - 100 %).
- The intensity of NRT data is enough to saturate the network. This means that although the other data-traffic classes have priority, NRT data-traffic will always be sent as soon as there is an opportunity. The intensity of the NRT data-traffic is constant throughout the simulation.

Figure 18 shows the result of the simulation. The concept of "throughput ceiling" is dependent on the data-traffic pattern, (see Section 23). In the figure, it can be seen that the highest priority data-traffic (LRTC) is not affected by the increasing level of BE data-traffic. Also note that the throughput of the NRT data-traffic decreases as the intensity of BE increases. In other words, prioritisation of different data-traffic classes in the simulator works as specified in the protocol.

As can be seen in the figure, the maximum total throughput (before any BE data-traffic is dropped) is approximately 7.7 packets per slot. The total throughput is the combined throughput of all data-traffic types.

22.3. Simulation 3

In this simulation, all three types of data-traffic are generated during the simulation. How the data-traffic types effect each other is studied. In short the BE data-traffic is constant and the LRTC data-traffic is varied. The network setup for the simulation is as follows:

- The load of the BE set is constant at 4000 packets (4 MByte) per period of 16000 slots (16 ms), (less load than the default RSP case).
- The load of LRTC data-traffic is varied between 1-16 kBytes per channel with a period of 100 slots.
- The intensity of NRT data is enough to saturate the network. This means that although the other data-traffic classes have priority, NRT data-traffic will always be sent as soon as there is an opportunity. The intensity of the NRT data-traffic is constant throughout the simulation.

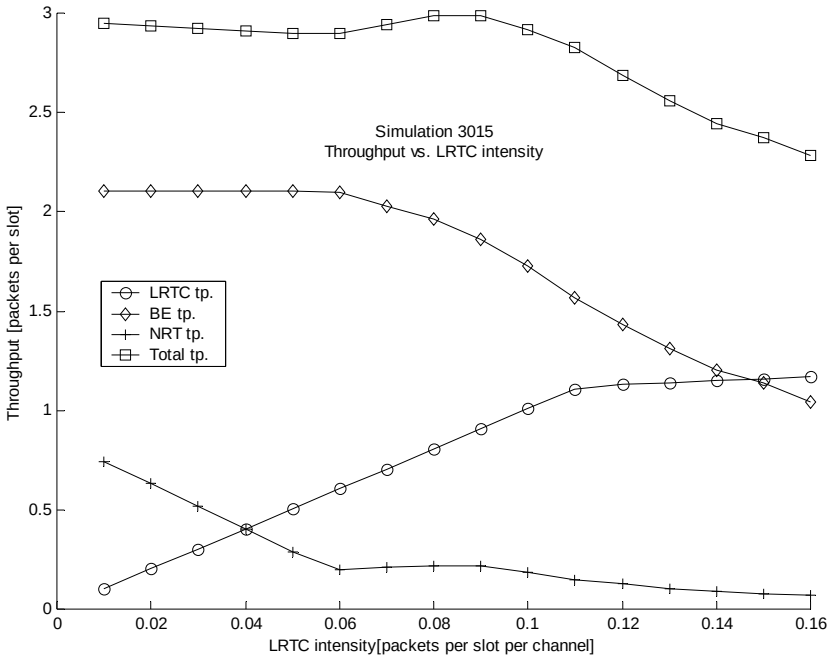

Figure 19: Throughput of the different classes of traffic when BE and NRT traffic stays constant as LRTC traffic varies.

Figure 19 shows the result of the simulation. Observe that the concept of “throughput ceiling” is dependent on the data-traffic pattern, (see Section 23).

The maximum total throughput, before any LRTC data-traffic is dropped, can be seen in the figure. The throughput is then approximately 2.8 packets per slot. Observe that in a real implementation of the network there would be admission control of LRTC. Thus it would be impossible to reach a level where LRTC data-traffic is lost. The total throughput is the combined throughput of all data-traffic types. The maximum throughput of LRTC data-traffic (before packets begin to be dropped) is approximately 1.1 packets per slot per channel. This can be found in Figure 19. When LRTC data-traffic begins to be dropped, the payload is 11 kByte per packet, i.e. the LRTC intensity is 0.11 packets per slot per channel (11 packets / 100 slots per channel). Here the LRTC throughput is 1.1 packets per slot

As the LRTC intensity increases, the total throughput drops. This can be seen in the figure. The effect is because there is decreasingly less data-traffic in the system that takes advantage of spatial reuse (pipelining), i.e. BE data-traffic. This phenomenon is further discussed in Section 23. As expected, throughput for the two lower priority data-traffic classes decrease as LRTC intensity is increased.

In the simulations two and three we refer to “channel load” and “intensity”. These refer to the same thing and are defined as the quotient of the “weight” of the channel (how many data packets will be transmitted during a period) over the period of the channel. For example, the number of data packets to be transmitted during one period is *y*. The period of the channel is *x*. The intensity of the channel is *y/x*. An intensity of 0.5 implies that each channel in the system (of the same type) is sending data packets during half of its period. This may be written: the channels are at 50% load. Observe that during simulation, the period of channels are not varied, only the payload of each message (consisting of one or several packets).

23. DISCUSSION ON THROUGHPUT CEILING

In simulation two and three a concept of total system throughput is used. This is a measure of the number of data packets that are sent during each slot. Each data packet takes one slot to transmit. When comparing the throughput of the different data-traffic classes, it must be taken into account that the classes have different data-traffic patterns and therefore have different throughput ceilings. This “100%”-level varies depending on the data-traffic pattern. In a system, the throughput ceiling is constant and does not change unless the data-traffic pattern does so.

The following example illustrates this. The worst case is when all communication is destined one node upstream. Here the throughput can be at most one. The best case is when all communication is destined one node down stream. Here the throughput can be at most *N*, where *N* is the number of nodes in the network. For data-traffic that is well-specified in channels, it is easy to find a value for throughput. However, if data-traffic is Poisson distributed then the throughput can only be known statistically. Four different scenarios of data communications patterns are shown as explained below. Observe that they are discussed in increasing order of throughput.

- Scenario one occurs when all data-traffic is destined “furthest around the ring” i.e. to the node’s upstream neighbour. This communication pattern does not suit the network topology under discussion (unidirectional pipelined ring). The level of throughput achieved is equivalent to that of a shared media network (e.g. a standard shared ring or bus), i.e. one data packet per slot.
- Scenario two occurs when the destination of all data-traffic is evenly distributed, i.e. on average the packets will travel half way around the ring. In this case the average throughput is two packets per slot. This is possible because of the pipelining feature of the network, where several transmissions may take place in non-overlapping segments.
- Scenario three is the radar case currently being discussed. Here, a large part of the communication is pure pipelined (the BE data-traffic), which is advantageous for total throughput. Therefore the total throughput will be larger than two. Observe that the total throughput depends on the data-traffic pattern. We have seen in the simulations that the total throughput with the radar case is 5.97 packets per slot, see Table 3.

- Scenario four occurs when all data-traffic is to the next downstream neighbour. Here the pipelining feature of the network is optimally utilised. Throughput will be *N* packets per slot, where *N* is the number of nodes in the network.

24. SUMMARY OF CCR-EDF

The CCR-EDF network protocol is suitable for hard real-time data-traffic. A pipelined optical ring network forms the network architecture. It has a distributed clocking strategy that makes it suitable for global deadline scheduling. The node that has the highest priority message in each slot is also handed the role as master, which includes responsibility for clocking. Because of this, the highest priority message from any node, in the system, can always be sent to any destination. This forms the basis for the scheduling framework. Clock hand over is done in accordance with the result from the medium access protocol. The medium access protocol has two basic functions, arbitration of access to the network and deciding which node to take over the clock role. Each node in the network sends a request for transmission of its locally highest priority message to the current master. The result is a list of requests for transmission of messages, one from each node in the network.

The user services include best effort messages, non real-time messages, logical real-time connections, group communication such as barrier synchronisation and global reduction, and services for reliable transmission. Logical real-time connections are realised by admission control and earliest deadline first scheduling of messages.

The function of the CCR-EDF protocol has been verified by simulation. Also, the concept of having different data-traffic classes to differentiate between data-traffic has been tested in simulation, and shown to work. Results from the simulations of CCR-EDF together with the radar signal processing case study show that the network is an effective choice.

25. OVERALL CONCLUSIONS

Two novel protocols for heterogeneous real-time services are presented in this report. The protocols support heterogeneous data-traffic by offering different communication services to address different requirements of the data-traffic. A pipelined optical ring network forms the network architecture. It has a distributed clocking strategy that makes it suitable for global deadline scheduling. Two protocols for the same basic network architecture are presented, analysed and tested by simulation: The TCMA and CCR-EDF protocols.

The advantage of the TCMA protocol is that the deadline requirements for packets from all nodes are taken into account and, for one packet from each node, thus considered at a global level. Since the global queue that is used for deciding which packets will be sent, consists of one request per node (in the current implementation), there will be situations where lower priority packets are sent even though there are higher priority packets queued at another node. This situation is referred to as a priority inversion and is due to the fact that the global queue of requests does not have the “complete picture” of the individual queues in the nodes. Since the priority for a message is dynamically increased as the laxity decreases, the TCMA protocol implements an approximation of the optimal “earliest deadline first” algorithm. The limitation is, as stated above, that only one message from each node is considered in each slot. However, for each node it is always the most urgent message that is considered, but the round-robin strategy for clock hand over in the TCMA-protocol leads to a pessimistic worst case. The TCMA protocol supports slot reservation for guaranteed real-time virtual channels.

In the CCR-EDF protocol the node that has the highest priority message in each slot is also handed the role as master, which includes responsibility for clocking. Because of this, the highest priority message from any node, in the system, can always be sent to any destination. This forms the basis for the scheduling framework. Clock hand over is done in accordance with the result from the medium access protocol. The medium access protocol has two basic functions, arbitration of access to the network and deciding which node to take over the clock role. Each node in the network sends a request for transmission of its locally highest priority message to the current master. The result is a list of requests for transmission of messages, one from each node in the network.

The user services include best effort messages, non real-time messages, logical real-time connections, group communication such as barrier synchronisation and global reduction, and services for reliable transmission. Logical real-time connections are realised by admission control and earliest deadline first scheduling of messages. Both TCMA and CCR-EDF based networks are suitable for applications with demands for real-time performance, such as for use as interconnection network in a radar signal processing system, or as a high performance network for use in a LAN environment. The network can be built today using fibre-optic off-the-shelf components.

Support in CCR-EDF for different data-traffic classes to differentiate between data-traffic has been tested in simulation, and shown to work. Results from the simulations of a radar signal processing case study show that the CCR-EDF network is an effective choice.

26. FUTURE WORK

Potential directions for future work include:

- The proposed protocols have disregarded the occurrence of faults. Such faults could disrupt operation of the protocol and cause system failure. A simple example is a physical break in the ring due to failure of a single node. Future work should investigate and add robustness to the protocols.
- Extend the research to also support related network topographies such as a dual ring. An interesting property of a dual ring is fault-tolerance.
- Extend the research to cover the possibility of having more parallelism in the optical links. Apart from achieving higher data rates, more advanced services would be possible.
- Include support in the protocol for multiple interconnected rings.
- Investigate start-up of the ring. Problems here include how to establish communication without any node initially being master.

27. REFERENCES

Agelis, S., S. Jacobsson, et al. (2002). Modular interconnection system for optical PCB and backplane communication. Proc. Workshop on Massively Parallel Processing (WMPP'2002) in conjunction with International Parallel and Distributed Processing Symposium (IPDPS'02), Fort Lauderdale, FL, USA.

Alijani, G. S. and R. L. Morrison (1990). An evaluation of IEEE 802 protocols and FDDI in real-time distributed systems, Minneapolis, MN, USA, Publ by IEEE, Piscataway, NJ, USA.

ANSI (1999). ANSI/VITA 5.1-1999 American National Standard for RACEway Interlink, VITA, . Scottsdale, Ariz.

Arvind, K., K. Ramamritham, et al. (1991). "A local area network architecture for communication in distributed real-time systems." Real-Time Systems 3(2): 115-147.

Bergenhem, C. (2000). A demonstrator for a CC-FPR network. Halmstad, Sweden, School of Information Science, Computer and Electrical Engineering (IDE), Halmstad University.

Bergenhem, C. and M. Jonsson (2002). Analysis problems in a spatial reuse ring network with a simple clocking strategy, IDE0253 Halmstad, Sweden, School of Information Science, Computer and Electrical Engineering (IDE), Halmstad University.

Bergenhem, C. and M. Jonsson (2002). Fibre-ribbon ring network with inherent support for earliest deadline first message scheduling. Workshop on parallel and Distributed Real-Time Systems (WPDRTS´02) in conjunction with Parallel and Distributed Processing Symposium., Proceedings International, (IPDPS´02), Fort Launderdale, FL, USA.

Bergenhem, C. and M. Jonsson (2003). The CCR-EDF Optical Pipelined Ring Network - Heterogeneous Real-time in Radar Signal Processing. Parallel and Distributed Computing Networks (PDCN) in conjunction with the 21st IASTED International Multi-Conference on Applied Informatics (AI 2003), Innsbruck, Austria, IASTED.

Bergenhem, C., M. Jonsson, et al. (2002). Heterogeneous Real-Time Services in High-Performance System Area Networks – Application demands and Case Study Definitions. Technical Report 263. Halmstad, Sweden, Halmstad University.

Bergenhem, C., M. Jonsson, et al. (2001). Fibre-ribbon pipeline ring network with distributed global deadline scheduling and deterministic user services, Proc. Workshop on Optical Networks (WON'01) in conjunction with 2001 International Conference on Parallel Processing (ICPP'01), Valencia, Spain, Sept. 3-7, 2001, pp. 311-318, IEEE Computer Society.

Bergenhem, C. and J. Olsson (1999). Protocol suite and demonstrator for a high performance real-time network. Centre for Computer Architecture (CCA). Halmstad, Sweden, Halmstad University. MSc.

Bettati, R. and A. Nica (1995). Real-time networking over HIPPI, Santa Barbara, CA, USA, IEEE Computer Society Press.

Boden, N. J., D. Cohen, et al. (1995). "Myrinet: a gigabit-per-second local area network." IEEE Micro 15(1): 29-36.

Bursky, D. (1994). "Parallel optical links move data at 3 Gbits/s." Electronic Design 42(24): 79-80.

CAN (1991). CAN Specification Version 2.0, Robert Bosch GmbH.

CSMA/CD (1985). Carrier sense Multiple Access with Collision Detection. IEEE Standard 802.3. New York, IEEE.

Cunningham, D. G., N. S. Div, et al. (2001). The status of the 10-Gigabit Ethernet standard. Optical Communication, 2001. ECOC'01. 27th European Conference on.

Davik, F., M. Yilmaz, et al. (2004). "IEEE 802.17 resilient packet ring tutorial." Communications Magazine, IEEE 42(3): 112-118.

Davis, R., A. Burns, et al. (2007). "Controller Area Network (CAN) schedulability analysis: Refuted, revisited and revised." Real-Time Systems 35(3): 239-272.

Duato, J., S. Yalamanchili, et al. (2002). Interconnection Networks, Morgan Kaufmann.

Einstein, T. H. (1997). Mercury Computer Systems' modular heterogeneous RACE(R) multicomputer, Geneva, Switzerland, IEEE Computer Society Press.

Fan, X. and M. Jonsson. Guaranteed real-time services over standard switched Ethernet, Proc. of the 30th Annual IEEE Conference on Local Computer Networks (LCN'2005), Sydney, Australia, Nov. 15-17, 2005.

Ferrari, D. and D. Verma (2002). "A scheme for real-time channel establishment in wide-area networks." Selected Areas in Communications, IEEE Journal on 8(3): 368-379.

Gjessing, S. and A. Maus (2002). A fairness algorithm for high-speed networks based on a resilient packet ring architecture. IEEE International Conference on Systems, Man and Cybernetics, Hammamet, Tunisia.

Gnauck, A. H., G. Charlet, et al. "25.6-Tb/s C+ L-band transmission of polarization-multiplexed RZ-DQPSK signals." Journal of lightwave technology 26(1).

Halsall, F. (1995). Data communications, computer networks and open systems. Essex, UK, Addison-Wesley Longman Ltd.

Healey, A. (2007). Challenges and Solutions for Standards-Based Serial 10 Gb/s Backplane Ethernet.

Hennessy, J. L., D. A. Patterson, et al. (2003). Computer Architecture: A Quantitative Approach. San Francisco, CA, USA, Morgan Kaufmann.

Hoang, H., M. Jonsson, et al. (2002). Switched real-time ethernet with earliest deadline first scheduling protocols and traffic handling. Proceeding of the Workshop on Parallel and Distributed Real-Time Systems (WPDRTS'2002) in conjunction with International Parallel and Distributed Processing Symposium (IPDPS'02), Fort Lauderdale, FL, USA.

Hopper, A. and R. M. Needham (1988). "The Cambridge fast ring networking system." Computers, IEEE Transactions on 37(10): 1214-1223.

Horst, R. W. (1995). "TNet: a reliable system area network." IEEE Micro 15(1): 37-45.

Huber, D., W. Steinlin, et al. (1983). "SILK: An Implementation of a Buffer Insertion Ring." Selected Areas in Communications, IEEE Journal on 1(5): 766-774.

IEEE (1985). IEEE Standard for Local Area Networks: Token Ring Access Method and Physical Layer Specifications, Standard, 802.5. ANSI/IEEE. New York.

Jafari, H., T. G. Lewis, et al. (1980). "Simulation of a Class of Ring-Structured Networks." IEEE Transactions on Computers 29(5): 385-392.

Jonsson, M. (1998). "Two fiber-ribbon ring networks for parallel and distributed computing systems." Optical Engineering 37(12): 3196-3204.

Jonsson, M., A. Ahlander, et al. (1996). Time-deterministic WDM star network for massively parallel computing in radar systems, Maui, HI, USA, IEEE Computer Society Press.

Jonsson, M. and C. Bergenhem (2001). A Class of Fiber-Ribbon Pipeline Ring Networks for Parallel and Distributed Computing Systems. Proc. of CSREA International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA-2001), Las Vegas, NV, USA.

Jonsson, M., C. Bergenhem, et al. (1999). Fiber-ribbon ring network with services for parallel processing and distributed real-time systems. Proceedings of the ISCA 12th International Conference. Parallel and Distributed Systems, Fort Lauderdale, FL, USA, ISCA.

Jonsson, M., K. Borjesson, et al. (1997). Dynamic time-deterministic traffic in a fiber-optic WDM starnetwork. Proceedings of the 9th Euromicro Workshop on Real Time Systems, Toledo, Spain.

Jonsson, M., B. Svensson, et al. (1997). Fiber-ribbon pipeline ring network for high-performance distributed computing systems. Third International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN '97), Taipei, Taiwan, IEEE Computer Society.

Kandlur, D. D., K. G. Shin, et al. (1994). "Real-time communication in multihop networks." IEEE Transactions on Parallel and Distributed Systems 5(10): 1044-1056.

Kermani, P. and L. Kleinrock (1979). "Virtual cut-through: A new computer communication switching technique." Computer networks 3(4): 267-286.

King, T. J. and I. Gallagher (1990). ORWELL-a multiservice network protocol, Munich, West Germany, IGI Europe.

Klemm, R. (1999). "Introduction to space-time adaptive processing." Electronics & Communication Engineering Journal 11(1): 5-12.

Kopetz, H. and G. Bauer (2003). "The time-triggered architecture." Proceedings of the IEEE 91(1): 112-126.

Krishna, C. M. and K. G. Shin (1997). Real-time systems, McGraw-Hill New York.

Kuszmaul, B. C. (1995). RACE network architecture. Proceedings of the International Parallel Processing Symposium, IPPS, Santa Barbara, CA, USA, IEEE, Los Alamitos, CA, USA.

Lebby, M., C. Gaw, et al. (1996). Use of VCSEL arrays for parallel optical interconnects.

Lemoff, B. E., M. E. Ali, et al. (2005). "Demonstration of a compact low-power 250-Gb/s parallel-WDM optical interconnect." IEEE Photonics Technology Letters 17(1): 220-222.

Leventis, S., G. Papadopoulos, et al. (1982). A new experimental computer network and its simulated performance. Proceedings of INFOCOM ´82, Las Vegas, NV, IEEE Computer Society Press.

Liu, M. T. and C. C. Reames (1977). "Message communication protocol and operating system design for the distributed loop computer network (DLCN)." Proceedings 4th Annual Symp. Computer Architecture 5(7): pp. 193-200.

Liu, M. T. and J. J. Wolf (1978). A distributed double-loop computer network (DDLCN). Proceedings of 7th Texas Conference on Computer Systems, Texas, USA.

Malcolm, N. and Z. Wei (1994). "The timed-token protocol for real-time communications." Computer 27(1): 35-41.

Malcolm, N. and Z. Wei (1995). "Hard real-time communication in multiple-access networks." Real-Time Systems 8(1): 35-77.

Mehra, P. (2001). Trends in system area networking. Proceedings IEEE International Symposium on Network Computing and Applications. NCA 2001, Cambridge, MA, USA, IEEE Computer Society.

Mukherjee, B. and S. Banerjee (1993). "Alternative Strategies for Improving the Fairness in and an Analytical Model of the DQDB Network." IEEE Transactions on Computers 42(2): 151-167.

Mukherjee, B. and C. Bisdikian (1992). "A journey through the DQDB network literature." Performance Evaluation 16(1-3): 129-158.

Pacifici, G. and A. Pattavina (1986). T-S protocol: An access protocol for ring local area networks. Proceedings of IEEE GLOBECOM´86, USA.

Pelissier, J. (2000). Providing quality of service over Infiniband architecture fabrics. Proceedings of the 8th Symposium on Hot Interconnects, Stanford, California, .

Pfister, G. F. (2001). Aspects of the InfiniBandTM architecture. Proceedings 2001 IEEE International Conference on Cluster Computing, Newport Beach, CA, USA, IEEE Computer Society.

Pop, T., P. Pop, et al. (2006). Timing analysis of the FlexRay communication protocol. 18th Euromicro Conference on Real-Time Systems (ECRTS 06), Dresden, Germany, Institute of Electrical and Electronics Engineers Inc., Piscataway, NJ 08855-1331, United States.

Raghavan, B., Y. G. Kim, et al. (1999). "A gigabyte-per-second parallel fiber optic network interface for multimedia applications." IEEE Network 13(1): 20-28.

Reinemo, S. A., T. Skeie, et al. (2006). "An overview of QoS capabilities in infiniband, advanced switching interconnect, and ethernet." IEEE Communications Magazine 44(7): 32-38.

Rom, R. and M. Sidi (1990). Multiple access protocols: performance and analysis. New York, NY, USA, Springer-Verlag New York, Inc. .

Ross, F. (2002). "An overview of FDDI: The fiber distributed data interface." Selected Areas in Communications, IEEE Journal on 7(7): 1043-1051.

Sano, B. J. and A. F. J. Levi (1998). Multimedia Technology for Applications. Networks for the professional campus environment. Piscata-way, New Jersey, IEEE Press: 413-427.

Saunders, S. (1998). Data Communications Gigabit Ethernet Handbook. New York, NY, USA, McGraw-Hill, Inc. .

Schow, C. L., F. E. Doany, et al. (2011). "A 24-channel, 300 Gb/s, 8.2 pJ/bit, full-duplex fiber-coupled optical transceiver module based on a single "Holey" CMOS IC." Journal of Lightwave Technology 29(4): 542-553.

Sha, L., S. S. Sathaye, et al. (1992). Scheduling real-time communication on dual-link networks. Real-Time Systems Symposium, Phoenix, AZ, USA, IEEE Computer Society Press.

Shin, K. G. (1991). "Real-time communications in a computer-controlled workcell." IEEE Transactions on Robotics and Automation 7(1): 105-113.

Silio Jr, C. B. (1986). PERFORMANCE APPROXIMATIONS FOR MULTIMESSAGE CIRCUIT-SWITCHED RINGS'. Proceedings: Computers and Communications Integration Design, Analysis, Management (INFOCOMM '86), Washington D.C., USA, Institute of Electrical and Electronics Engineers.

Stankovic, J. A. (1988). "Misconceptions about real-time computing: A serious problem for next-generation systems." Computer 21(10): 10-19.

Stimson, G. W. (1998). Introduction to airborne radar, SciTech Pub Mendham, NJ.

Tanenbaum, A. S. (2002). Computer Networks. Upper Saddle River, N.J. USA, Prentice Hall PTR.

Taveniku, M., A. Ahlander, et al. (1998). The VEGA moderately parallel MIMD, moderately parallel SIMD, architecture for high performance array signal processing. Proceedings of the 12th International Parallel Processing Symposium & 9th Symposium on Parallel and Distributed Processing (IPPS/SPDP '98), Orlando, FL, USA, IEEE Computer Society.

Tindell, K. W., H. Hansson, et al. (1994). Analysing real-time communications: controller area network (CAN). Proceedings of the Real-Time Systems Symposium, San Juan, Puerto Rico, IEEE Computer Society Press.

Tolmie, D., T. M. Boorman, et al. (1999). From HiPPI-800 to HiPPI-6400: A changing of the guard and gateway to the future. Proceedings. 6th International Conference on Parallel Interconnects (PI'99) (Formerly Known as MPPOI), Anchorage, AK, USA, IEEE Computer Society.

Tran-Gia, P. and T. Stock (1990). "Approximate performance analysis of the DQDB access protocol." Computer Networks and ISDN Systems 20(1): 231-240.

Trezza, J., H. Hamster, et al. (2003). “Parallel optical interconnects for enterprise class server clusters: needs and technology solutions.” IEEE Communications Magazine 41(2): S36-S42.

Tsiang, D. and G. Suwala (2000). "RFC2892: The Cisco SRP MAC Layer Protocol." Internet RFCs.

Vaughan-Nichols, S. J. (2002). Will 10-Gigabit Ethernet Have a Bright Future? IEEE Computer. vol. 35 22 –24.

Wolf, W. (2002). "What is embedded computing?" Computer 35(1): 136-137.

Wong, P. C. and T.-S. P. Yum (1994). "Design and analysis of a pipeline ring protocol." IEEE Transactions on Communications 42(2-4): 1153-1161.

Wu, E. (2012). “A framework for scaling future backplanes.” IEEE Communications Magazine 50(11): 188-194.

Xu, M. and J. H. Herzog (1988). Concurrent token ring protocol. Proceedings. Seventh Annual Joint Conference of the IEEE Computer and Communcations Societies, INFOCOM '88. , New Orleans, LA, USA, IEEE.

Zarlink. (2009). "Zarlink.com ZL60101." Retrieved 10th July, 2009, from www.zarlink.com/zarlink/hs/82_ZL60101.htm.

Zhang, H. (1995). "Service disciplines for guaranteed performance service inpacket-switching networks." Proceedings of the IEEE 83(10): 1374-1396.

Zhao, W., A. Kumar, et al. (1994). Real-time communication in FDDI-based reconfigurable networks, Seattle, WA, USA, IEEE Computer Society Press.

Zuberi, K. M. and K. G. Shin (2000). "Design and implementation of efficient message scheduling for controller area network." IEEE Transactions on Computers 49(2): 182-188.

Åhlander, A. and M. Taveniku (2002). Engineer efficient parallel systems - a preliminary requirements specification, Technical Report 1/0363-FCP 104 825 Uen. Mölndal, Sweden, Uen Ericsson Microwave Systems.