A Class of Fiber-Ribbon Pipeline Ring Networks for Parallel and Distributed Computing Systems

Magnus Jonsson and Carl Bergenhem

School of Information Science, Computer and Electrical Engineering, Halmstad University, Halmstad, Sweden, Box 823, S-301 18, Sweden.

Abstract

In this paper, three ring networks made up of fiber-ribbon point-to-point links are presented. The first network is a control-channel based network in which one fiber in each link joins with others to form a control-channel ring. This ring improves performance of the network by sending medium access control information immediately before the data transmissions. High throughputs can be achieved in the network due to pipelining, i.e., several packets can be traveling through the network simultaneously but in different segments of the ring. Also, real-time demands can be met using slot reserving. The network, called CC-FPR (Control-Channel based Fiber-ribbon Pipeline Ring), can be built today using off-the-shelf fiber-optic components. The increasingly good price/performance ratio for fiber-ribbon links indicates a high potential for the success of the proposed kind of networks. The second network is similar to the first one except that it divides the network into two sub-networks, one for packet switched traffic and one for circuit-switched traffic. When the main data flow in the network does not change rapidly, this is a good choice of a simple but powerful network. In the third network, the medium access protocol of the first network is exchanged with a one with global deadline scheduling to support best-effort real-time traffic.

Keywords: fiber-optic communication, ring network, fiber-ribbon, parallel processing, distributed systems, real-time communication.

1 Introduction

The authors and colleagues have presented, in earlier papers, computing systems with computational modules that function as stand alone SIMD (Single Instruction stream Multiple Data streams) computers, in which all processing elements work together closely. We use these computational modules as building blocks when building larger, highly parallel computer systems. At the global level, these systems are multiple SIMD computers with one instruction stream per computational module. The systems have been developed as part of a joint project between Halmstad University, Ericsson Microwave Systems AB, and Chalmers University of Technology.

Application examples are future radar signal processing systems, distributed multimedia systems, satellite imaging and other image processing methods. A typical example is the radar signal processing system described in [1] [2]. Often, these systems are classified as real-time computer systems. In a real-time computer system, correct function depends both on the time at which a result is produced and on its accuracy [3]. In distributed real-time systems, the interconnection network is a very important part of the computer system. Often, guaranteeing real-time services is much more important in these systems than performance, e.g., average latency.

Since each module itself can have a sustained data output rate of several Gb/s, a powerful interconnection network is needed. Systems that require a few tens of nodes can be realized by using optical fiber-ribbon links. Fiber-ribbon links offering an aggregated bandwidth of several Gb/s have already reached the market [4]. The price performance ratio is very promising.

In this paper, we describe three ring networks suitable for different situations. The proposed networks are pipeline ring networks based on optical fiber-ribbon point-to-point links. In a pipeline ring network, several packets can be traveling through the network simultaneously because of spatial reuse, thus achieving an aggregated throughput higher than the capacity of a single link. Motorola OPTOBUS™ bi-directional links [5] with ten fibers per direction are used but the links are arranged in a unidirectional ring architecture (Figure 1) where only \( \lceil M / 2 \rceil \) bi-directional links are needed to close a ring of \( M \) nodes.
The first network is called CC-FPR (Control-Channel based Fiber-ribbon Pipeline Ring). The physical ring network is divided into two rings: a data ring and a control ring. In each fiber-ribbon link, eight fibers carry data and one fiber is used to clock the data, byte for byte. Together, these fibers form a data channel that carries data packets. The access is divided into slots as in an ordinary TDMA (Time Division Multiple Access) network. The tenth fiber is dedicated to bit-serial transmission of control-packets that are used for the arbitration of data transmission in each slot. The clock signal, on the dedicated clock fiber, that is used to clock data also clocks each bit in the control-packets.

The node synchronization requirement is more relaxed than for a traditional TDMA network and the network is somewhat similar to a slotted ring network (but without the need of a central controller). This is because the access to the network circulates among the nodes according to the physical order of the nodes in the ring. In addition, the ring can dynamically (for each slot) be partitioned into segments to obtain a pipeline ring network where several transmissions can take place simultaneously. Even simultaneous multicast transmissions are possible when the multicast segments do not overlap. Also, slot reserving is used to obtain guaranteed bandwidth in real-time computer systems.

Other high-performance ring networks include the WDM passive ring [6] and the hierarchical WDM ring [7] which are more closely related to the WDM star network and star-of-stars network that we proposed in [8] and [1]. Other pipeline ring networks are described in [9-11] and more references are available in [9]. Advantages of the CC-FPR network 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. In other words, control and data are overlapped in time. With less header overhead in the data-packets the slot-length can be shortened, to reduce latency, without sacrificing too much in bandwidth utilization. Also, the separate clock and control fibers simplify the transceiver hardware implementation.

The network described by Jafari et al. also relies on a separate control channel but needs a central control node that brings additional cost in hardware and in latency when waiting for response from the central control node [10]. The CC-FPR network is insensitive to propagation delay in the sense that no feedback is needed, from other nodes or from a central controller, between control-packet and data-packet transmissions.

The physical ring of the second network is sub-divided into two networks, which carry different kinds of traffic. Nine of the fibers are used for time multiplexed circuit switched traffic; eight fibers are for data and one for clocking. The tenth fiber is dedicated to packet switched traffic using, for example, a token ring protocol. This fiber also carries control messages to reconfigure the TDMA schedule, (i.e., circuit establishment) for the other nine fibers. This network is a good choice when the main data flow in the network does not change rapidly. More information on the first two networks, including analysis etc, is found in [12] [13].

The third network is similar to the first one but the use of the control channel is divided into two phases, one phase for collection of queue status from each node and one phase to distribute information of which node or nodes that gets to send in the next slot. The protocol is therefore called TCMA (Two-Cycle Medium Access). The protocol provides global deadline scheduling of packets to support best-effort real-time traffic.

The rest of the paper is organized as follows. The CC-FPR network is presented in Section 2. In Section 3, the network supporting both packet and circuit switched traffic is described. Then, in Section 4, the TCMA network is presented. This is followed by conclusions in Section 5.

2 The CC-FPR network

First, the CC-FPR protocol is described with the assumption that slot reserving is not used. Then, slot reserving is described.

2.1 The CC-FPR protocol

Before the CC-FPR protocol arbitration mechanism is explained, a description of how data-packets travel on the ring is given. The access to the network is cyclic; each cycle consists of $M$ time-slots, where $M$ is the number of
nodes. Each node is denoted \( m_i \), \( 1 \leq i \leq M \). In each slot there is always one node responsible for initiating the traffic around the ring. This node is called the slot-initiator. Each node is slot-initiator in one slot per cycle, as shown in Figure 2. At the end of the slot, the role of being slot-initiator is asynchronously handed over to the next node downstream. This can be done implicitly simply by sensing the end of the slot, i.e., the last bit.

The CC-FPR medium access protocol is based on the use of a control-packet that, for each slot, travels almost one round (over \( M - 1 \) links) on the control-channel ring, as shown in Figure 3. The node that will be the slot-initiator in the next slot initiates the transmission of the control-packet, as shown in the figure. In the time domain, the control-packet always travels around the ring in the time-slot preceding the one for which it controls the arbitration (see Figure 4). Accordingly, the control-packet always passes each node one time-slot before the data-packet to which it is related.

The contents of the control-packet are shown in Figure 5. The control-packet consist of a start-bit followed by an \( M \) bit long link-reservation field and an \( M \) bit long destination field, where \( M \) is the number of nodes. Each bit in the link-reservation field tells whether the corresponding link is reserved for transmission in the next slot. In the same way, each bit in the destination field tells whether the corresponding node has a data-packet destined to it in the next slot. Additional information, such as node insertion, could also be included in the control-packet; for clarity, this is not shown in the figure.

Each node succeeding the slot-initiator checks the control-packet as it passes to determine: (i) if it will receive a data-packet in the next slot, which is indicated by the node’s bit in the destination field, and (ii) if a data-packet will pass the node in the next slot, which is indicated by the bit in the link-reservation field corresponding to the outgoing link of the node. If no data-packet is to pass the node, i.e., the rest of the ring back to the slot-initiator is free, then the node can transmit a data-packet in the next time-slot in this part of the ring.

When a node has a packet ready for transmission, it prepares, in advance, new link-reservation and destination fields to reserve needed links and notify destination node(s). In this way the node can immediately change the control-packet when it passes, provided the bit in the link-reservation field corresponding to the outgoing link of the node, is set to zero. Since there is no data-packet that will pass the node, succeeding nodes have no use for the overwritten information in the control-packet.

Because all of the nodes succeeding the slot-initiator repeat the procedure of checking the control-packet, multiple transmissions in different segments of the ring could occur in the same slot. An example of how the control-packets travel around a five-node network is shown in Figure 6. The arbitration results in two concurrent data-packet transmissions in the next slot, one single-destination and one multicast packet, as shown in Figure 7. Node \( m_1 \) is the slot-initiator in the example;

---

**Figure 2:** The role of being slot-initiator is cyclically repeated. Each of the \( M \) nodes is the slot-initiator in one slot per cycle.

**Figure 3:** The node succeeding the slot-initiator initiates the control-packet transmission.

**Figure 4:** In each slot, a node passes/transmits one control-packet and one data-packet, where the control-packet is used for the arbitration of the next slot.

**Figure 5:** A control-packet contains a start bit, a link-reservation field, and a destination field.
therefore it initiates the control-packet transmission described in Figure 6. It reserves Link 1 and Link 2 for transmission to node \( m_3 \) and informs this node, by setting the corresponding bits in the link-reservation field and the destination field, respectively, that it will have a data-packet destined to it in the next slot. While node \( m_2 \) and node \( m_3 \) do not change the control-packet, they check it to see if there will be any data-packets destined to them in the next slot. Node \( m_4 \) reserves Link 4 and Link 5 for a multicast transmission to node \( m_5 \) and node \( m_1 \). Node \( m_5 \) then receives the control-packet and removes it from the ring.

The reason why the control-packet travels only among the first \( M - 1 \) links after the slot-initiator is that the clock signal is interrupted before the last link to avoid interfering with itself (see Figure 3). The node that initiated the transmission of the control-packet does not return the packet. Consequently, it will not be informed of whether or not there is a data-packet destined to it in the next slot. However, the node will receive either a packet destined to the node or an empty packet.

Each transmitter has \( M - 1 \) queues, one for each possible destination (the node itself excluded). When a multicast packet arrives for queuing, it is put in the queue corresponding to the multicast destination furthest away from the source node downstream. In this way multicast packets are treated in the same way as single-destination packets and multiple multicast packets can be traveling in the network at the same time whenever possible.

### 2.2 Slot reserving

Many computer systems have real-time demands for which the network must offer guaranteed bandwidth for certain communication patterns. This can be done in the network by using slot reserving. Either the whole ring is reserved for a specific node in a slot, or several segments of the ring are dedicated to some specific nodes.

![Diagram of a network with five nodes. Node 1 is slot-initiator.](image)

**Figure 6:** A control-packet travels around a network with five nodes. Node 1 is slot-initiator.

<table>
<thead>
<tr>
<th>Node</th>
<th>Outgoing control-packet</th>
<th>Transmission allocated</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 1 0 0 0 0 0 0 1 1 0 0 0</td>
<td>To Node 3</td>
</tr>
<tr>
<td>2</td>
<td>1 1 0 0 0 0 0 0 1 0 0</td>
<td>Could not allocate</td>
</tr>
<tr>
<td>3</td>
<td>1 1 0 0 0 0 0 0 1 0 0</td>
<td>Could allocate transmission to Nodes 4, 5, and 1 but had nothing to send</td>
</tr>
<tr>
<td>4</td>
<td>0 0 0 1 1 1 0 0 1 0 0 0</td>
<td>Multicast to Node 5 and 1</td>
</tr>
<tr>
<td>5</td>
<td>0 0 0 1 1 1 0 0 0 1 0 0</td>
<td>Could not allocate</td>
</tr>
</tbody>
</table>

When slot reservation is allowed, the cycle is prolonged to contain \( Q = M + R \) slots, where \( R \) is the number of slots used for reservation. The value of \( R \) is chosen when the system is designed and remains unchanged during operation of the network, provided the system function does not change radically. For example, there could be a mode change in a radar system, such as switching from the task of scanning the whole working range to that of tracking a certain object. For fairness, the \( M \) ordinary slots are not allowed to be reserved.

When a node is going to reserve a slot, it searches for slots where the required links are free, so allocation of a new segment can be made. First, the node’s own slots (i.e., where the node itself is the slot-initiator) are searched. When too few slots (actually only a segment in each slot) for the reservation can be allocated, the search is extended to other slots. In that case, the node broadcasts a data-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 number of slots needed. Each node then checks its own slots for the required free links. All of the nodes send a packet back to the requesting node to notify which slots, if any, have been allocated. When the requesting node has received the answers, it decides if the number of allocated slots is sufficient. If not, it sends a release packet. Otherwise, it can start using the reserved slots immediately. However, if more slots than needed were allocated, a release packet is sent out.

The advantage of this slot reservation method over circuit-switching is that when a node does not need its reserved slot, the slot can be used by other nodes in the segment. For example, suppose that node \( m_1 \) has a segment reserved containing the four links between itself and node \( m_5 \). If node \( m_1 \) does not need the slot in a cycle, the other nodes in the segment are informed of that when the control-packet passes in the slot before. Node \( m_2 \) will
have the first chance to take over, followed by the node $m_1$ and $m_e$. Multiple nodes can even reuse the same slot when the communication demands of the other nodes in the segment allow for that.

3 The packet and circuit switched ring network

Compared to the CC-FPR network, the network for both packet and circuit switched traffic is slightly simpler at the expense of somewhat reduced support for dynamic traffic patterns. However, in many systems only a fraction of the traffic is irregular. Below, circuit switched and packet switched traffic, and circuit establishment is described.

3.1 Circuit switched traffic

For circuit switched traffic, the first nine fibers in each link form a high-speed channel. All of the high-speed channels, together, form a high-speed ring network for circuit switched traffic. The access is divided into slots as in an ordinary TDMA network. However, in each slot the network can be divided into segments as in the CC-FPR network. Also, for each slot there is always a slot-initiator node. The same kind of asynchronous slot-synchronization method is also used.

The access is cyclic and each cycle consists of $K$ slots. In a typical case, $K$ is a multiple of $M$, where $M$ is the number of nodes, and each node is the slot-initiator in $K/M$ slots. In each segment and slot, one, and only one, node can be the owner of the links, and hence has the right to use the segment links for transmission.

3.2 Packet switched traffic

The tenth fibers from each of the links are combined to form a ring network totally dedicated for packet switched traffic. An ordinary ring protocol can be used. However, there are two requirements: (i) it must be possible to halt the protocol when special packets for circuit establishment are to be transmitted (see Section 3.3), and (ii) the latency must be upper bounded to assure transmission of the packets for circuit establishment. When using, e.g., a token ring protocol on the packet network, this network will support low latency communication for sporadic packets at moderate traffic rates. At the same time, it is assured that the circuit switched traffic (often real-time traffic) is not disturbed by packet switched traffic.

3.3 Circuit establishment

When a node is to establish a new circuit, it searches for slots where the required links are free, so allocation of a new segment can be made. First, the node’s own slots (i.e., where the node itself is the slot-initiator) are searched. When too few slots (actually only a segment in each slot) for the circuit can be allocated, the search is continued in other slots. In that case, a special request packet is transmitted on the packet network to ask other nodes to allocate the desired segment in their slots. This packet is immediately followed by a collect packet to collect information on the success of the slot segment allocations.

The request packet, which is broadcasted to all other nodes, contains information about the links required and the number of slots needed. Each node then checks if any of its own slots have the required free links. If so, it prepares to modify the collect packet when it arrives (before forwarding it), to notify the requesting node of which slots have been allocated. However, if any of the previous nodes have already allocated slot segments and modified the collect packet, the number of slots needed would be decreased by the corresponding number of allocated slots. The number of slots still needed is indicated by a dedicated field in the collect packet. In this way, allocation of more slots than needed is avoided. However, several nodes can each allocate some of the slots needed and information about all of these allocations is added to the same collect packet.

When the requesting node receives the collect packet after one round, it decides if the number of allocated slots is sufficient. If not, it sends a release packet. Otherwise, it can start using the established circuit immediately.

4 The TCMA network

The two phases of medium access for the TCMA protocol are collection and distribution phases (see Figure 8). The network arbitration information, for data in slot $N + 1$, is sent in the previous slot, slot $N$. The number of slots is chosen so that each node is slot-initiator at least once per cycle (here assumed to be once per cycle).

There are two types of TCMA control packets, which are used in each of the two phases (see Figure 9). A complete collection phase packet will contain a start bit and a total of $N − 1$ requests that are added one by one by each node. The slot-initiator receives it’s 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 must be traversed to reach the destination node. For the link reservation field, each bit corresponds
to one link and tells whether the link is reserved (1) or not (0). The destination field has one bit for each node in a corresponding way. Since a node may write several destination nodes into the destination field, 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). Due to limited space, these are not described here but more information on this topic, for the CC-FPR network, is found in [13].

In the collection phase, the current slot-initiator 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 and then passes the packet on to the next node. The slot-initiator receives the complete request packet and determines, according to earliest deadline first, which packet(s) that are allowed to be transmitted in the next slot.

The time until deadline (laxity) of a packet is mapped, with a certain function, to be expressed within the four-bit limitation of the current version of TCMA’s priority field. 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.

Two mappings between deadline and priority, logarithmic (higher resolution of laxity, the closer to its deadline a packet gets) and linear, have been simulated. Results show a negligible difference in performance of throughput, packet-loss, and latency.

Packets queued locally in nodes are sorted by laxity and distance and 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 will not request transmission of a packet that will pass the slot-initiator since the clock signal is interrupted there and data cannot pass.

There can only be $N$ requests in the slot-initiator, 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 slot-initiator 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 slot-initiator) have priority over other requests.

When the slot-initiator has scheduled the requests, it distributes the result to all nodes in the distribution phase. 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 “result request field” of the distribution phase packet contains a “1”.

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 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 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.

Slot-reserving can be implemented as in the CC-FPR network. Due to limited space, a performance analysis of the network is to be published elsewhere.

5 Conclusions

Three ring networks based on fiber-ribbon links have been presented. Very high throughputs can be achieved in the networks, especially in systems for which a pipelined
data flow between the nodes exists. The CC-FPR network relies on the use of a dedicated control sub-network that is easy to implement using a dedicated fiber on each fiber-ribbon link. Guaranteed bandwidth is supported in the CC-FPR network by slot reserving, a method that allows for slot reuse when the guaranteed bandwidth is temporarily not in use. In a typical system, slot reserving can be used for time-critical data flows, guaranteeing that they are not disturbed by, e.g., control information. In the second network, circuit-switching is offered to support guaranteed bandwidth, while a packet-switched sub-network at the same time supports flexible traffic. The third network, TCMA, has support for best-effort real-time traffic through the use of global deadline scheduling.

6 References


