The RDT network Router Chip

Hiroaki Nishi†[1] , Hideharu Amano†, Katsunobu Nishimura†, Ken-ichiro Anjo†, Tomohiro Kudoh‡
†Keio University, ‡Tokyo Engineering University

1 Design Outline

The RDT network Router chip is a versatile router for the massively parallel computer prototype JUMP-1, which is currently under development by collaboration between 7 Japanese universities[1].

The major goal of this project is to establish techniques for building an efficient distributed shared memory on a massively parallel processor. For this purpose, the reduced hierarchical bit-map directory (RHBD) schemes [2] are used for efficient cache management of the distributed shared memory.

In order to implement (RHBD) schemes efficiently, we proposed a novel interconnection network RDT (Recursive Diagonal Torus)[3], and developed a sophisticated router chip for the RDT which equips a hierarchical multicast mechanism without deadlock and acknowledge combining mechanism.

By using the 0.5μBiCMOS SOG technology, it can transfer all packets synchronized with a unique CPU clock(60MHz). Long coaxial cables(4m at maximum) are directly driven with the ECL interface of this chip. Using the dual port RAM, packet buffers allow to push and pull a flit of the packet simultaneously. The mixed design approach with schematic and VHDL permits the development of the complicated chip with 90,522 gates in a year.

2 JUMP-1 and the RDT

JUMP-1[1] consists of clusters connected with an interconnection network RDT[3]. Each cluster is a bus-connected multiprocessor coarse-grained processors (CPU:SUN SuperSparc+), 2 fine-grained processors (Memory Based Processor or MBP) each of which is directly connected to a main memory and the RDT router chip. The MBP, the heart of JUMP-1, is a custom designed fine-grained processor which manages the distributed shared memory, synchronization, and packet handling. The first prototype of JUMP-1 provides 256 clusters, thus, 1024 processors.

2.1 Interconnection network RDT

The RDT is a network consists of recursively formed two-dimensional square diagonal tori. Assume that four links are added between a node (x, y) and nodes $(x \pm 2, y \pm 2)$. Then, the additional links form a new torus-like network. The direction of the new torus-like network is at an angle of 45 degrees to the original torus. Here, we call the torus-like network the rank-1 torus. On the rank-1 torus, we can make another torus-like network (rank-2 torus) by providing four links in the same manner.

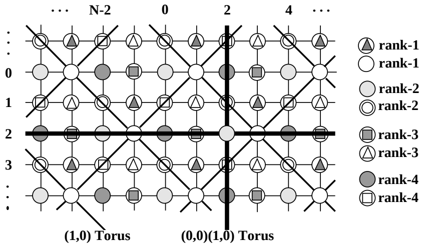

Figure 1: Torus assignment used in the JUMP-1

Recursive Diagonal Torus RDT(2,4,1) can be defined as a class of networks in which each node has links to form base (rank-0) torus and only 1 upper torus(rank1, rank2, rank3 , rank4). The network used in JUMP-1 is shown in Figure 1.

In this network, every node has eight links, four for the base (rank-0) torus and four for rank (1-4) torus (Most of links for upper rank tori are omitted in Figure 1). Each node has other three ranks in its neighbor nodes. This property reduces the diameter and average distance between nodes.

In JUMP-1, to decrease the coherence maintenance messages, using the hierarchical bit-map directory scheme used in COMA (Cache Only Memory Architecture), a message is transferred for different destinations simultaneously (i.e. multicast) using a tree structured multicasting paths (multicasting tree).

For maintaining cache consistency, acknowledge packets are usually required. These packets are transferred from the destination nodes to the source node, and informs to finish of the invalidation (or data update). Unlike the other directory methods, in the hierarchical directory method, acknowledge packets can collect and combine packets in each hierarchy, and it reduces the network traffic.

Since JUMP-1 is a massively parallel processor, large directory entry is not feasible. In order to cope with this problem, the Reduced Hierarchical Bit-map Directory scheme (RHBD) was proposed[2].

For the efficient implementation of this method, the router chip with high speed multicast and acknowledge packet combining mechanism are required.

3 The RDT router chip

3.1 Structure of the router chip

The structure of the RDT router chip [4] which supports the RHBD scheme is shown in Figure 2. The core of the chip is a 10 × 11 crossbar which exchanges packets from/to ten 18-bits-width links., that is, four for the rank-0 torus, four for the upper rank torus, and two for the MBPs. In JUMP-1, two RDT router chips are used in the bit-sliced mode to form 36 bits width for each link.

All packets are transferred between router chips synchronized with a unique 60MHz clock. In order to maximize the utilization of a link, packets are bi-directionally transferred. Maximum packets length is 16flits (36 bits-width 16flits-length) so as to carry a line of the cache. 3-flits header which carries the bit-map of the RHBD is attached to every packet, but the length of the body is variable. Unlike common router chips, efficient deadlock free asynchronous wormhole routing, acknowledge packet combining, shootdown/setup, and error/handling mechanism are available.

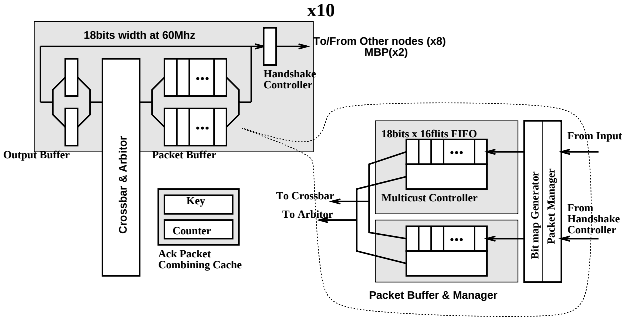

Figure 2: The structure of the RDT router

3.2 Hierarchical multicast

3.2.1 The method of Multicasting

In the RDT router chip, the asynchronous wormhole routing is adopted to cope with the frequent multicasting. And it has a packet buffer which the whole packet can be stored.

To utilize all chances of multicast, each buffer provides the multicast bit map indicating required destination of packets. The packet is multicast whenever the empty opposite buffers are found, and then the corresponding bits of the multicast bit map are reset.

[1] Hiroaki Nishi, Dept. of Computer Science, Keio University 3-14-1 Hiyoshi, Kouhoku-ku, Yokohama City 224 JAPAN
Phone: +81-45-560-1063 Fax: +81-45-1064, E-mail:west@aa.cs.keio.ac.jp

ASP-DAC '97
0-89791-851-7$5.00 © 1997 IEEE

3.2.2 Deadlock avoidance

In the RDT, the multicast is performed without deadlocks using a simple modification of the e-cube routing.

For this modified e-cube routing, two virtual channels are required for South direction of every torus and two for East/West direction of base torus. To cope with it, the RDT router chip provides two virtual channels to each link. And a detour on base torus to utilize neighbor node which has another upper rank causes the deadlock. To avoid this, east and west links also has special channel.

These channels are automatically changed in the deadlock free mode. In the user selection mode, the user can use them freely. In this router, the FIFO assumption is ensured since the route of the multicast is fixed.

3.3 Acknowledge packet combining

In the router, a combining buffer is provided, and acknowledge packets are automatically combined if the key of each acknowledge packet matches to the key of the buffered multicasting packet.

If another multicast is performed when the acknowledge packet combining buffer is full, the combining is done in the MBP. The bit to disable this combining mechanism is provided in the header flit.

3.4 Packet shooting down and setting up

In case of job switching or debugging, it is sometimes required that all packets in the network are flushed out. After doing another job or finishing the debug, these packets must be returned. These mechanisms are called the packet shooting down and setting up.

When the request for the shooting down by the MBP is issued, the router changes its mode into the packet shooting down mode. The request is also issued with transmission errors inside the router. A simple barrier synchronization line which connects all RDT router chips of every node in cascade can also carry the request. In this mode, all packets in packet buffers are forced to send to the MBP. The FIFO assumption is ensured even if shooting down and setting up are performed.

3.5 Error handling mechanism

Reliability is important for such a complex router for massively parallel machines. The parity bit is attached to the header flits to handle the errors. Another parity checker is also provided for the internal status of all buffers. Since two chips are used in the bit-sliced manner in JUMP-1, the inconsistency of the internal status parity of two chips also indicates a error. If any error is detected, the router changes its mode into the shooting down automatically.

Each buffer also provides a timer for flushing a packet which stays in the buffer too long. The firing time can be selected in the range from $100\mu sec$ to $100msec$.

3.6 Implementation

$0.5\mu m$ Hitachi BiCMOS SOG which provides 125K gates in maximum is utilized. Lines are directly driven with the ECL interface of this chip. Using the dual port RAM, packet buffers allow to push and pull a flit of the packet simultaneously.

The structure of the packet buffer is shown in Figure 2. For supporting complicated packet control, following two parts in a channel are implemented and checked separately:

- Bit-map generator:

  The pattern of the multicast/broadcast is decided with the bit map carried in the header of the packet, the location of the link, mode of the packet and state of the packets. The bit-map generator generates the bit map for multicasting from the information. This circuit is complicated but not difficult to be checked since it is a pure combinatorial circuit.

- Multicast/handshake control:

  A packet is immediately multicast to the destinations when the receiver is ready, and during the last multicasting, the next packet can be inserted to the buffer. It is managed with a sequential logic.

In order to reduce the time of the arbitration, the arbitration of the crossbar and bidirectional line are performed at a time. The priority of the crossbar is decided in the round robin manner to avoid the starvation. And the arbitration and the packet sending are overlapped. The multicast bit maps of header is shifted so that the flit which will be used first is available at the next router.

Figure 3 is the photo of sample chips. This package provides 299 pins including 260 signal. To cope with 19W power consumption, a large heat sink is attached. Large power is consumed in ECL I/O buffer and Bi-CMOS cell.

The required number of gates are shown in the Table 1. Random logics require 50,000 gates in total while areas corresponding to about 4,000 gates are required for the dual-port RAM. The crossbar body and arbiter, which are simple but high performance is required, are designed in the schematic while the complicated controllers are described in the VHDL.

Figure.4 show the layout of the RDT router chip. Black blocks in the image are dual port RAM for packet buffers which provides virtual channel.

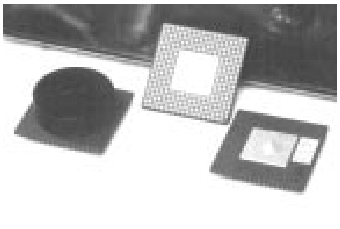

Figure 3: Sample chips

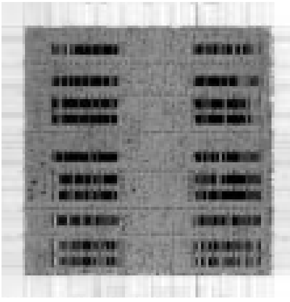

Figure 4: Chip Layout

Block name	Gates	Number of blocks	Total gates	Description
Crossbar	2,927	1	2,927	Schematic
Arbiter	2,736	1	2,736	Logic level
Multicast controller	1,558	10	15,580	VHDL
Bit map generator	2,288	10	22,880	VHDL
Acknowledge combining	2,009	1	2,009	VHDL
RAM for buffer	2,021	20	40,420	RAM
Total			90,522	

Total pins	299(Signal 260)
Power consumption	19.4W
Optimized number of gates	80,307
Rate of gate utilization	63%

Table 1: The number of gates of the RDT router

Acknowledgment

The authors use Mentor Graphics Japan design tool and the placement, and routing of the SOG is done by Hitachi-Microcomputer Co. Ltd. and fabricated by Hitachi Co. Ltd.. We express our sincere gratitude to all staffs for supporting the design.

References

[1] K. Hiraki, et al. Overview of the JUMP-1, an MPP prototype for general-purpose parallel computations. In *Proc. of the International Symposium on Parallel Architectures, Algorithms, and Networks (ISPAN'94)*, 1994.

[2] T. Kudoh, et al. Hierarchical bit-map directory schemes on the RDT interconnection network for a massively parallel processor JUMP-1. In *Proc. of the 1995 ICPP*, 1995.

[3] Y. Yang, et al. Recursive diagonal torus: An interconnection network for massively parallel computers. In *Proc. of 1993 IEEE Symposium on Parallel and Distributed Processing*, 1993.

[4] H. Nishi, et al. The Jump-1 Router Chip: A Versatile Router for Supporting a Distributed Shared Memory, In *Proc. of International Phoenix Conference on Computer and Communication (IPCCC'96)*, 1996.