

# Micro-network for SoC : Implementation of a 32-port *SPIN* network

Adrijean ANDRIAHANTENAINA Alain GREINER

Pierre and Marie CURIE University

LIP6 laboratory

4 place Jussieu, 75252 Paris Cedex 05, France

{Adrijean.Andriahantenaina,Alain.Greiner}@lip6.fr

## Abstract

We present a physical implementation of a 32-ports *SPIN* micro-network. For a 0.13 micron CMOS process, the total area is  $4.6 \text{ mm}^2$ , for a cumulated bandwidth of about 100 Gbits/s. In a 6 metal process, all the routing wires can be routed on top of the switching components. The *SPIN32* macro-cell will be fabricated by ST Microelectronics, but this macrocell uses symbolic layout, and can be manufactured with any CMOS process including 6 metal layers.

## 1. Introduction

The integration density doubles every two years and permits today the integration of tens of millions of transistors on a single chip. The challenge in the next few years is to implement, on a single chip, heterogeneous systems containing twenty or thirty heterogeneous processors, coprocessors or other IP cores.

The bottleneck at the architectural level resides in the management of the communications between the cores.

To deal with it, we developed the micro-network *SPIN* (*Scalable Programmable Integrated Network*), which is able to provide to system designers a bandwidth which increases linearly with the number of embedded cores. The main goal of the *SPIN* project is to replace a shared bus by a multi-stage network based on packet switching: a well known interconnect strategy used in parallel computing and telecommunications [6].

The routing strategy is distributed, adaptive and of worm-hole type [3]. The *SPIN* point-to-point, full duplex physical links are 36 bits wide in each direction and use a credit-based flow control [2]. A *SPIN* network consists of three VLSI macrocells. One router and two wrappers respecting the VCI standard (Virtual Component Interfacing) [1].

The *RSPIN* router routes the packets to their final destinations

while the two wrappers (i.e. *VCI/SPIN* and *SPIN/VCI*) are in charge of interfacing the *SPIN* network with the subscribers (i.e. processors, coprocessors, memories ...). The wrappers provide the subscribers with an Advanced VCI interface so that all the subscribers see a simple shared address space and don't have to manage the *SPIN* protocol. In this paper, we focus on the physical implementation of a 32-port *SPIN* network containing 16 *RSPIN* routers.

## 2 The *SPIN* network topology

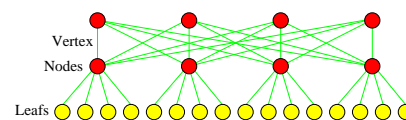
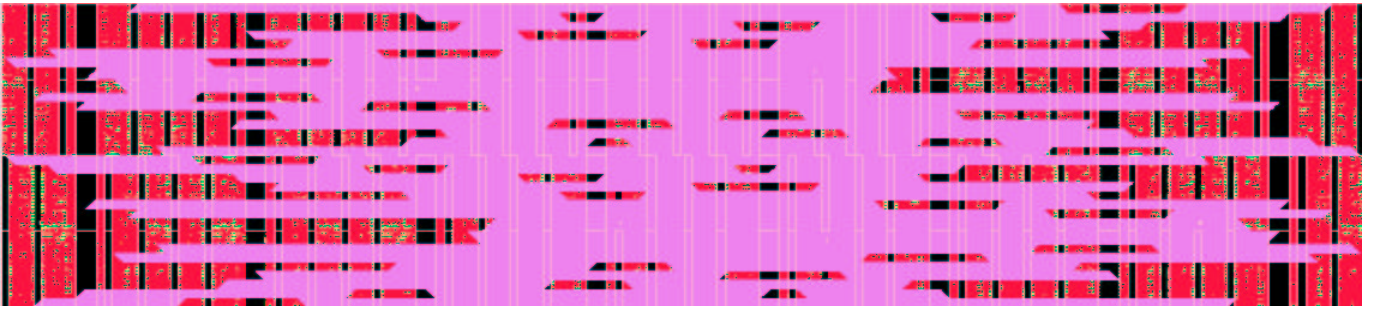


Figure 1. Fat tree topology

A *SPIN* network has a fat tree topology [7] [5], where every node has four sons and the father is replicated four times at any level of the fat tree (figure 1). This topology is intrinsically redundant, since the four fathers offer four equivalent paths in order to route a message between two sons of the same father. In this topology, the shortest path between two subscribers is the one that goes through by the nearest common ancestor.

The fat tree topology has the following advantages: its diameter (maximum number of links between two subscribed) remains reasonable ( $2 * \log_4 n$ , where  $n$  is the number of layer of network), the topology is scalable and uses a small number of routers for a given number of

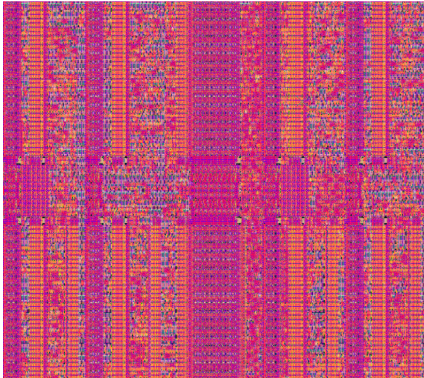


**Figure 2. The 32-port *SPIN* network macrocell**

subscribers. It has a natural hierarchical structure which can be useful in the embedded systems [4].

### 3 Implementation

The *RSPIN* router macrocell has been designed with optimized, semi-custom layout, but we used the *ALLIANCE* symbolic layout approach in order to gain process portability. We achieved the *place and route* of the *RSPIN* router macrocell in two steps. The highly regular data path topology was described explicitly with *GENLIB*, the procedural description language of *ALLIANCE* suite. The data path is organized as 8 *FIFOs* and involves a ten by ten crossbar, built with multiplexors and tristate gates [2]. The control logic contains the finite state machine in charge of the adaptive routing algorithm. For the control logic, the *place and route* process was then performed automatically by *SILICON ENSEMBLE*, using the portable cells library of *ALLIANCE* suite.



**Figure 3. *RSPIN* router layout**

The *RSPIN* router height is defined by the *FIFOs*; its width is determined by the input/output wires. The *RSPIN* macrocell has a  $0.24 \text{ mm}^2$  area, with *ST Microelectronics*  $0.13 \text{ micron}$  process (figure 3).

We also designed a 32-port *SPIN* network (*SPIN32*) [2]. It contains 16 *RSPIN* routers, which are placed in two rows of eight *RSPIN* routers each. The total area is  $4.6 \text{ mm}^2$ , where about 30% is occupied by the *FIFOs*. All the wires connecting the *RSPIN* routers are routed over the routers in layer 4,5 and 6 (figure 2).

### 4 Conclusion

We demonstrated with this work that it is possible to implement a 32-port *SPIN* network, in less than  $5 \text{ mm}^2$  area, with a  $0.13 \text{ micron}$  process, for a cumulated bandwidth of about 100 Gbits/s, as the *SPIN* network bandwidth allows every subscriber to expect about 3 Gbit/s in each direction at 200 MHz clock rate. This *SPIN32* macrocell will be fabricated by *ST Microelectronics*, but the *symbolic layout* of *ALLIANCE* suite allows us to manufacture it with any process including 6 metal levels.

### References

- [1] V. S. I. Alliance. *Virtual Component Interface Standard version 2, OCB 2 2.0*. <http://www.vsi.org> (document access may be limited to members only), April 2001.
- [2] A. Andriahantenaina. *SPIN: technical report*. Pierre and Marie CURIE University, Paris, France, 2002.
- [3] W. Dally and C. Seitz. Deadlock-free message routing in multiprocessor interconnection networks. *IEEE Transactions on Computers*, vol. C-36, no. 5, pages 547–553, May 1987.
- [4] P. Guerrier. *Un réseau d'interconnexion pour systèmes intégrés*. Pierre and Marie CURIE University, Paris, France, 2000.
- [5] P. Guerrier and A. Greiner. A generic architecture for on-chip packet-switched interconnections. *Proceedings of the Design Automation and Test in Europe Conference 2000 (DATE'2000)*, Paris, France, pages 250–256, Mars 2000.
- [6] J. Hennessy and D. Patterson. *Computer Architecture, A Quantitative Approach - 2nd Edition*. Morgan Kaufmann Publishers, San Francisco, CA, USA, 1996.
- [7] C. Leiserson. Fat-trees: Universal networks for hardware-efficient supercomputing. *IEEE Transactions on Computers*, vol. C-34, no. 10, pages 892–901, October 1985.