Report Number: WUCS-81-01

1981-04-01

VLSI Based Interconnection Networks

Authors: Mark A. Franklin and Donald F. Wann

Interest in tightly coupled multiprocessor computer systems has grown as the possibilities for high performance with such systems have been recognized. Central to their design is the structure of the network over which the processors communicate. Unless properly designed, such networks can be both a cost and performance bottleneck. This paper focuses on the design of VLSI communications networks, this is, on communications network which can be placed on a single VLSI chip. Traditional SSI-based boost and complexity measures for such networks have principally involved switch aggregate counts. In a VLSI domain, however, more appropriate measures involve chip area, and space-time product. The effects of network topology and VLSI layout on these measures are reviewed with regard to two network types. Another important question related to the VLSI communication network problem related to the chip pin constraints. This problem is discussed and some effects and options presented by bit slice network designs are described.
... **Read complete abstract on page 2.**

VLSI Based Interconnection Networks

Complete Abstract:

Interest in tightly coupled multiprocessor computer systems has grown as the possibilities for high performance with such systems have been recognized. Central to their design is the structure of the network over which the processors communicate. Unless properly designed, such networks can be both a cost and performance bottleneck. This paper focuses on the design of VLSI communications networks, this is, on communications network which can be placed on a single VLSI chip. Traditional SSI-based boost and complexity measures for such networks have principally involved switch aggregate counts. In a VLSI domain, however, more appropriate measures involve chip area, and space-time product. The effects of network topology and VLSI layout on these measures are reviewed with regard to two network types. Another important question related to the VLSI communication network problem related to the chip pin constraints. This problem is discussed and some effects and options presented by bit slice network designs are described.

VLSI Based Interconnection Networks

Mark A. Franklin and Donald F. Wann

WUCS-81-01

April 1981

Department of Computer Science
Washington University
Campus Box 1045
One Brookings Drive
Saint Louis, MO 63130-4899

Presented at the International Symposium on Circuits and Systems, Chicago, Illinois, April 27-29, 1981.

This work was supported in part by NSF Grant MCS-78-20731 and ONR Contract N00014-80-C-0761 and NIH Grant RR00396.

VLSI BASED INTERCONNECTION NETWORKS*

Mark A. Franklin and Donald F. Wann

Washington University
St. Louis, Missouri

ABSTRACT

Interest in tightly coupled multiprocessor computer systems has grown as the possibilities for high performance with such systems have been recognized. Central to their design is the structure of the network over which the processors communicate. Unless properly designed, such networks can become both a cost and performance bottleneck. This paper focuses on the design of VLSI communications networks, that is, on communications networks which can be placed on a single VLSI chip. Traditional SSI -based cost and complexity measures for such networks have principally involved switch aggregate counts. In a VLSI domain, however, more appropriate measures involve chip area, and space-time product. The effects of network topology and VLSI layout on these measures are reviewed with regard to two network types. Another important question related to the VLSI communication network problem relates to chip pin constraints. This problem is discussed and some effects and options presented by bit slice network designs are described.

INTRODUCTION

In recent years there has been increasing interest in tightly coupled, physically local multiprocessor computer systems (1,2). This has been due both to the enhanced performance possibilities for such systems, (e.g., increased computational power resulting from parallel processing and higher reliability resulting from component redundancy) and the steady decrease in hardware costs associated with these systems. A central issue in the design of such systems concerns performance degradation due to costs associated with interprocessor communication. One aspect of this problem relates to the question of user problem decomposition and scheduling (3,4). Another relates to the structure and design of the network over which the multiple processors communicate. As the number of processors increases the characteristics of both the decomposition and scheduling algorithms, and the communications network, become critical in establishing acceptable overall system performance cost. This paper is concerned with certain communications network design questions which arise in context of multiprocessor systems designed in a VLSI environment.

Various studies aimed at characterizing and quantifying the performance of SSI based networks have already been pursued (5,6,7). Typically the principal figures of merit used in these studies have been related to the number of switches required by the communications network and the bandwidth of the path between processors. For a given network architecture, determination of the number of switches is straightforward, while estimation of the bandwidth has, in most cases, been derived from an analysis of the average number of switches through which a signal must pass and the blocking characteristics of the network.

Use of these types of figure of merit make sense in an environment where the cost of switching elements is substantially greater than the cost of wires and connection paths. The situation changes however, with the introduction of VLSI technology. This fabrication methodology has the potential for economically placing large switching networks or subnetworks on a single chip. Cost here becomes related to chip area. Unfortunately, a new challenge appears: the implementation of the connection paths may use substantial amounts of the chip area, thus limiting the area available to the switch elements themselves. This has the effect of reducing the size of a switching network that can be fabricated on a chip of a given size. The time delay associated with the connection paths also contributes to the overall delay, thus directly effecting bandwidth. Area, topology and layout, mainly ignored in traditional communication network analysis, become important interrelated factors in VLSI network design (8).

The advent of VLSI has thus significantly changed the design space with which engineers must contend. More meaningful figures of merit based on parameters of time (on/off chip time) and chip area now seem more appropriate in many situations. In the next section two interconnection networks Crossbar (9) and Banyan (10) are compared in terms of their space-time products when implemented on a single VLSI chip.

While single chip design questions are important, when large networks requiring multiple chips are to be designed with single chip subnetworks as the building components, interchip communications delays can dominate overall delay times. Furthermore there is often a close relationship between intra and interchip communications network design. It may be advantageous, for instance, for the communications network on the chip to have a topology

This work was supported in part by NSF Grant MCS-78-20731, ONR Contract N00014-80-C-0761 and NIH Grant RR00396.

different from the larger interchip network. Furthermore, this interacts directly with questions relating to chip pin limitations, network conrol, and communications path width. It may turn out that bit serial communications paths are best because they preserve pins and permit large networks to be placed on a single chip. Reducing the delays associated with interchip communications may more than offset the extra time necessary for serial data transmission. The question of pin limitations is explored later in this paper and some preliminary results reviewed.

SPACE-TIME NETWORK COMPARISONS

Banyan and Crossbar Networks

This section reviews certain research results comparing banyan and crossbar interconnection networks (11). The crossbar network considered is of the form shown in Figure 1. While there are many ways of designing a crossbar (e.g., demultiplexer/multiplexer designs, switched bus designs), the approach examined has a number of generally desirable capabilities. These include distributed (local) network path control, asynchronous and pipelined data transfer, and a high degree of modularity (9). Furthermore its naturally regular and planar layout appears to make it well suited to VLSI implementation. The banyan network considered is of the form shown in Figure 2. It too can be designed for distributed network path control, asynchronous data transfer and pipelining. On the other hand its modularity properties are not quite as straightforward as the crossbar and its topology, while regular in a certain sense, is inherently nonplanar. Note that both networks have a full interconnection capability in that any single input/output connection can be made by placing the appropriate switches in the proper positions.

While the properties mentioned above are important, most work to date has compared the networks principally on the basis of switch complexity (i.e. number of switches required for implementation), and network bandwidth. For square N input, N output networks switch complexity for the crossbar is $A'_{CB} = O(N^2)$ while for the banyan it is $A'_{BA} = O(N \log N)$.

Network bandwidth is associated with three items. First, pipeline characteristics of the network and message length distributions must be determined. For analysis simplicity, a nonpipelined design and circuit switched mode of operation are assumed. Message length considerations therefore do not enter into these bandwidth comparisions. The second item to consider relates to the average number of switches through which a message must pass assuming uniform addressing of input and output ports. For the crossbar this is $D'_{CB} = O(N)$ while for the banyan it is $O(\log N)$. The final item to consider relates to the networks blocking charateristics and the protocol to be used when a message is blocked. The crossbar is a strict sense nonblocking network. That is, as long as each input port addresses a unique output port, no message is blocked in the network due to path contention (and no rearrangement of paths is necessary). The banyan network on the other hand is a blocking network, and under certain situations message blocking can occur. For N less than about 2000, the probability of this blocking can be approximated by $P_N \approx 1 - b/N^a$ with $a = .19$ and $b = 1.05$. Assuming a saturated system, synchronized messages of equal length, and a message retry protocol where blocked messages reenter the system again with the next message batch, the average delay through a banyan network can be derived as $D'_{BA} = O(\log N)/(1-P_N)$.

Overall cost measures for the banyan and crossbar can now be obtained as:

$$C'_{CB} = A'_{CB} \cdot D'_{CB} = O(N^3) \qquad [1]$$

$$C'_{BA} = A'_{BA} \cdot D'_{BA} = O(N^c \log^2 N) \quad (1<c<2) \qquad [2]$$

From these equations it is clear that by traditional measures, the banyan is much less costly than the crossbar. This is also true for other blocking networks such as the Omega (12) and indirect binary N-cube (13) whose switch complexities are also $O(N \log N)$.

VLSI Network Implementation Model

Consider next that the layout of a crossbar network on a single chip directly follows the topology of Figure 1. Assuming the logic associated with each crosspoint fits into a square with sides of length L, and the spacing between squares follows the Mead and Conway (8) recommendations for spacing between metal interconnect lines (i.e., 3 feature sizes, 3λ), then the area in units of λ^2 is given by:

$$A_{CB} \approx [NL + 3(N-1)]^2 = O(N^2) \qquad [3]$$

Unfortunately, for the banyan case the analysis is not as straightforward and one must refer to reference 11 for details. The thrust of the analysis, however, is as follows. First assume again that the switches in the banyan network fit into a square with sides of length L. Next assume that two layers of metal interconnect are available, one for horizontal and one for vertical lines. Various layouts may be proposed, but as long as the general form shown in Figure 2 is preserved (i.e., successive rows of switches) two things become evident. First, the horizontal distance required by the network will be $O(N)$ since this will vary directly with the number of input and output ports. Second, the vertical spacing between switch rows will increase as the network grows. This is because the number of horizontal lines which require routing between the right and left halves of the network increases as the network grows. These lines, being routed on the same plane, need more area as one moves from level (row) to level. This is illustrated in Figure 3. The result of this is that although there are $O(\log N)$ levels, the vertical distance grows as $O(N)$ and thus the area grows as $A_{BA} = O(N^2)$.

The interesting point to note here is that this is just the same as the crossbar, and is considerably different from what is predicted from switch aggregate counts. One other point to note is that these same results can be obtained by following a graph theoretic argument developed by Thompson (14).

Developing time delay models follows the same approach discussed above. As pointed out, for the

crossbar an average path contains N crosspoints. If each crosspoint is implemented in NMOS NOR gates with: a fanout f; a transit time τ; a pullup to pulldown transitor impedance ratio of 4; m levels of logic; a metalization capacitance/transistor gate capacitance ratio of α; and intercrosspoint drivers of minimum area; then the crossbar delay can be derived as:

$$D_{CB} \simeq 2.5Nmf\tau + (N-1)\tau(1+2.25\alpha) = O(N) \qquad [4]$$

For the banyan case a more complex expression can be obtained. Here certain assumptions must be made concerning driving the metal lines between levels which increase in length from level to level. Assuming the metal lines present purely capacitive loads, and are driven by a matched sequence of driver stages (8) to minimize delay, it can be shown that the delay presented by the lines is $O(\log^2 N)$. Introducing the multiplicative factor related to the blocking probability yields an overall delay of $D_{BA} = O(N^a \log^2 N)$ where $0<a<1$. For large N this is less than D_{CB}, however it is greater than that predicted from traditional analysis.

The overall space-time product measure is given below:

$$C_{CB} = A_{CB} \cdot D_{CB} = O(N^3) \qquad [5]$$

$$C_{BA} = A_{BA} \cdot D_{BA} = O(N^d \log^2 N) \qquad (2<d<3) \qquad [6]$$

These results indicate that while C_{BA} is still less than C_{CB} for large N, the costs are much closer than predicted from standard switch aggregate analysis. A more detailed analysis indicates that for reasonable values of N (i.e., values that could be currently implemented on a single chip), the two networks have roughly comparable space-time performance.

PIN LIMITATIONS

One of the key constraints on placing very large networks on a single chip is the limited number of pins supported by standard integrated circuit carriers. Consider for instance the interconnection network depicted in Figure 4 which establishes a B'-bit path between device x_i and device y_j where $1<i,j<N'$. Our interest is in developing networks that are general in that we place minor, if any, conditions on the specific numerical values for N' and B'. If the network of Figure 4 were to be implemented on a single VLSI chip then the number of required pin connections (ignoring power, ground, and general control such as reset) is given by 2N'B'. Suppose, for example, that we have a square interconnection network with N' = 12 and B' = 16. Then the number of required pins is 384, much larger than common commercially available integrated circuit carriers. The total number of pins is limited mainly by the increase in the physical length of the package; the pins are typically placed on 100 mil centers if the package is to be inserted in pads on printed circuit boards. (We ignore here certain more advanced schemes such as the array configuration used by IBM). For this pin placement, a 64 pin dual-in-line package is 3.2 inches in length. This becomes physically awkward to manipulate and also becomes more susceptable to breaking forces. The 384 pin example, for instance would require a 19.2 inch dual-in-line package!

There are a number of potential solutions to this pin limitation problem. We review here two of the more obvious ones with details being presented in reference 15. The first approach is to implement a large network requiring many pins as a collection of smaller networks where each of the smaller networks can be contained on a single chip in which the pin constraints of the chip are met. An N'*N' network would therefore be decomposed into a set of subnetworks (each subnetwork of size N*N) which would themselves be interconnected in some fashion.

The second approach is to slice the network so that one creates a set of network planes, each plane handling one or more bits (e.g., B bits) of the B'-bit wide datapath. This is commonly done in memory designs. Note that a potential problem arises here due to the difficulty in synchronizing the multiple planes. Although details of this issue will not be discussed here, there are ways of dealing with this problem.

The question to be considered is what represents the "best" combination of datapath slice B and chip network size N given: an overall network size N'; a data path width B'; an intrachip network type T; an interchip network type T'; a maximum allowable number of pins N_p; and a required number of pins for power, ground and control N_k.

A Chip Count Model

While the "best" B and N selection refers to both the chip count and bandwidth of the overall N'*N' network, due to space limitations only the chip count analysis is reviewed here. With regard to network types T and T', the overall chip count is a function only of T', the interchip network type. Although there are numerous ways of connecting subnetworks together to achieve an overall network, two basic network types are considered.

The first is the common crossbar network. An 8*8*1 crossbar network for example can be configured using 4*4*1 chip components. The number of N*N*B chips required to implement an N'*N'*B' network is given by:

$$Ncb = \left\lceil \frac{B'}{B} \right\rceil \left\lceil \frac{N'}{N} \right\rceil^2 \qquad [7]$$

The second type of network considered is the banyan. The number of N*N*B chips needed to implement an N'*N'*B' network is given by:

$$Nba = \left\lceil \frac{B'}{B} \right\rceil \left\lceil \frac{N'}{N} \right\rceil \left\lceil \log_N N' \right\rceil \qquad [8]$$

The first term in this expression is the number of bit slices or network planes; the second term is the number of chips at each level (row) and the third term is the number of levels necessary to achieve a full interconnection.

Pin constraints can be introduced by noting that:

$$N_P \geq K_1 BN + N_k \qquad [9]$$

where $K_1 = 4$ for a fully modular crossbar network, and $K_1 = 2$ for a banyan network. Consider next two cases. For case I the number of power, ground and control pins are small compared to the data pins and thus, using all available pins, $N = N_P/K_1B$. This is typical of clocked systems where a small number of clock lines are needed to synchronize all the data lines. For case II N_k is not negligible

and the number of control lines is proportional to the number of ports, N; thus, $N = N_p/(K_1B+Q)$. This would be an appropriate model if the network chips communicated with each other in an asynchronous manner and a request/acknowledge control line pair (Q=2) were associated with each port.

Each of the above expressions for N can now be substituted back into equations 7 and 8, and a value of B, and thus N, yielding the minimum number of chips obtained. To get some feeling for this one can assume that large values of B' and N' are present and that the number of chips can be approximated by expressions 7 and 8 with the ceiling functions removed. From these continuous functions it is clear that for case I the number of chips is minimized with B = 1. This is reassuring since it corresponds to experience with memory chip design where the slice width is generally taken as one bit. A discrete optimization search procedure verifies that this is true with the ceiling functions in place for crossbar networks, and for banyan networks above a certain size (N'>256). For case II, the B = 1 result generally holds for the crossbar case. For the banyan case, however, the best value of B varies considerably depending on the particular N_p, N' and B' values being considered. For instance with B'=16, N'=512 and N_p=60, a B=2(N=10) results in Nba = 1248 while a B = 1(N=15) results in Nba=1680. Typically large differences in chip counts occur when a nonoptimum value of B is used in this situation. Two other points should be noted. First, the control pin overhead in case II (Q=2) is substantial. For instance with B'=16, N'=256 and N_p=90; Nba=192 with Q=0 and Nba=348 with Q=2. Second, from a chip count point of view, there is a heavy penalty associated with using a crossbar interchip network due to the $O(N^2)$ versus $O(N \log N)$ network growth in number of chips.

The above sort of chip minimization analysis suggests that in designing an interconnection network chip set, chip control procedures which are proportional to the number of I/O ports be avoided (i.e., no request/acknowledge pairs on a per port basis), and a banyan like interchip network be used. Under these conditions, a path slice of B=1 seems appropriate. Not considered here is the question of how this path width and interchip network selection effect overall bandwidth. Initial results (15) indicate that the selections above are also appropriate for bandwidth optimization.

Other questions remain to be answered. One relates to whether having separate network chips are appropriate given their pin requirements. Perhaps structures which include both networks and processors are more reasonable. A variety of questions relating to centralized versus decentralized network control, the tradeoffs associated with circuit switched, packet and pipelined network designs, and the various options associated with synchronous versus asynchronous/delay insensitive design remain to be explored.

References

1. Enslow, P.H.,Jr., "Multiprocessor organization - a survey," ACM Comp. Sur. 9,1 (March 1977).
2. Kuck, D.J.,"A Survey of Parallel Machine Organization and Programming," ACM Comp. Sur. 9,1 (March 1977).
3. Chu, W.W. et.al.,"Task Allocation in Distrubuted Data Processing", Computer 13,1 (Nov.1980).
4. Cornett, D.H. and Franklin, M.A.,"Scheduling Independent Tasks with Communications",Proc.17th Allerton Conf. on Comm., Cont. and Comp. (Oct. 1979).
5. Siegel, H.J.; McMillen, R.J., and Mueller, P.T., Jr.,"A Survey of interconnection methods for reconfigurable parallel processing systems," Proc. 1979 Nat. Comp. Conf. (June 1979).
6. Anderson, G.A., and Jensen, E.D.,"Computer interconnection structures: taxonomy, characteristics, and examples" ACM Comp. Sur. 7,4 (Dec. 1975).
7. Thurber, K.J.,"Interconnection networks - a survey and assessment", Nat. Comp. Conf. (May 1974).
8. Mead, C. and Conway, L., Introduction to VLSI Systems, Addison-Wesley Pub. Co. Reading,MA (1980).
9. Franklin, M.A.; Kahn, S.A., and Stucki, M.J., "Design Issues in the Development of a Modular Multiprocessor Communications Network", Proc. Ann. Symp. on Comp. Arch. (April 1979).
10. Goke, L.R. and Lipoviski, G.J., "Banyan Networks for Partitioning Multiprocessor Systems", Proc. Ann. Symp. on Comp. Arch. (1973).
11. Franklin, M.A., "VLSI Performance Comparison of Banyan and Crossbar Communications Networks", IEEE Trans. on Comp. C-30,4 (April 1981).
12. Lawrie, D., "Access and alignment of data in an array processor", IEEE Trans on Comp. C-24,12 (Dec. 1975).
13. Pease, M.C.,"The indirect binary n-cube microprocessor array," IEEE Trans. Comp. C-26, 5 (May '75)
14. Thompson, C.D.,"Area-Time Complexity for VLSI", Proc. 11th Ann. ACM Symp. on the Theory of Comp. (April 1979).
15. Franklin, M.A. and Wann, D.F.,"Pin Limitations in VLSI Interconnection Networks", Int. Rept. #CCSD-81-01, Washington Univ., St. Louis, MO Ctr. for Comp. Sys. Design (Feb. 1981).

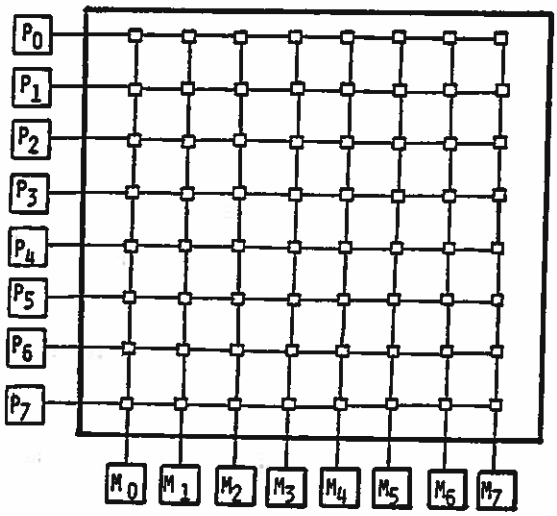

FIGURE 1 : 8 * 8 CROSSBAR SYSTEM

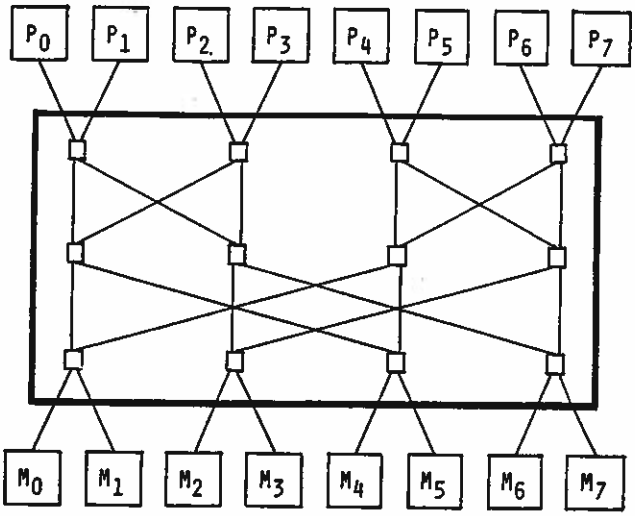

FIGURE 2 : 8 * 8 BANYAN SYSTEM

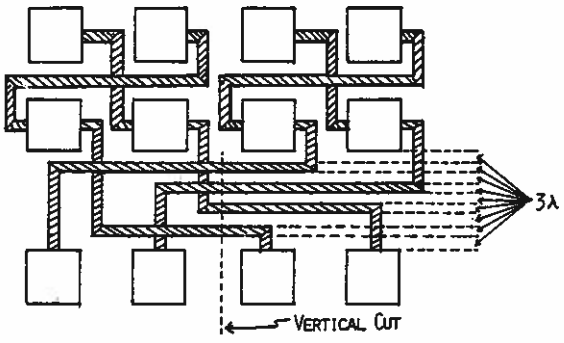

FIGURE 3: VERTICAL LAYOUT FOR BANYAN (B=1)

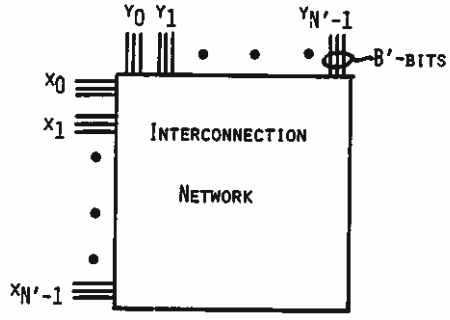

FIGURE 4: AN N'*N' NETWORK (B' WIDE DATAPATHS)