平行處理系統訊息傳遞架構及其結構之研究
An algorithm with its architecture for message-passing parallel systems

黃聯全
Liang-Chiang Huang
逢甲大學電機研究所
Institute of Electrical Engineering, Feng Chia University, Taichung, Taiwan
huanglc@com.ece.fcu.edu.tw

楊豐瑞
Fongray Frank Young
逢甲大學電機研究所
Institute of Electrical Engineering, Feng Chia University, Taichung, Taiwan
young@com.ece.fcu.edu.tw

摘要

本文介紹一種利用編碼技巧實現無虛擬通道之蟲洞尋徑網路，一種適合數學分析的模組將被討論，而此一編碼技巧的效能和結果將在本文做一分析與討論。

Abstract

An approach with virtual channel free using coding techniques in wormhole-routed network is introduced. A numerical model is applied for analysis. Performance results that confirm the advantage of coding method is presented and discussed in this paper.

Keyword: virtual channel, wormhole routing, adaptive routing, pseudonoise (PN).

I. Introduction

Efficient communication among nodes is critical to the performance of massively parallel computers (MPCs). No matter how well the computation is distributed among the processors, communication overhead can severely limit system speedup. In order to reduce network latency and minimize buffer requirement, the wormhole routing switching strategy [1] is being used in most current MPCs [2]. In wormhole routing, a packet is divided into a number of flits for transmission. The header flit of a packet governs its route, and the remaining flits follow in a pipeline fashion. A survey of wormhole routing in direct networks, can be found in ref. [3].

A routing technique is adaptive if the path choice depends on dynamic network conditions, such as the presence of faulty or congested channels. In wormhole routing, blocked messages are not buffered at intermediate nodes, but rather continue to hold previously acquired channels while waiting for others. A deadlock occurs when two or more messages are delayed forever due to an acquired-waiting cyclic dependency among their requested resources. In wormhole-routed networks, the small buffers associated with each channel are used.

To avoid deadlocks in wormhole-routed networks the virtual channels was proposed [4]. A virtual channel is a logical link between two nodes. It is formed by a flit buffer in the source node, a physical channel between them, and a flit buffer in the receiver node. Figure 1 shows the concept of four virtual channels sharing a single physical channel.

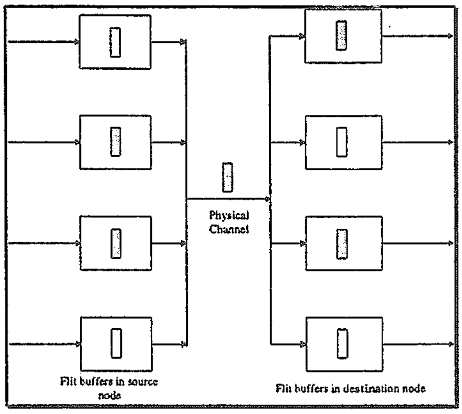

Fig.1 Four virtual channels sharing a physical channel with time multiplexing on a flits-by-flits basis.

The physical channel is time-shared by all virtual channels. Besides buffers and the involved channel, some channel states (handshaking protocol) must be identified with different virtual channels. Source buffers hold flits awaiting the use of channel, and the receiver buffers hold flits just transmitted over the channel. The channel (wires or fibers) provides the communication medium between them. To implementing the virtual channels, crossbar switch control, a multiplexer, and a demultiplexer are needed. The sharing of a physical channel by a set of virtual channels is processed by time-multiplexing on a flit-by-flit basis [4].

Since the concept of virtual channels is time-sharing, system time latency is increased, while network states are crowded. An approach with virtual channels free in wormhole-routed networks using coding techniques is introduced. The coding techniques applied in this approach is spread spectrum technique. One of many advantages using spread spectrum techniques is multiple access. Several data can be transmitted simultaneously at the same physical channel without waiting. Time latency of the system may be reduced through this approach. However, coding techniques applied for parallel systems have not been widely used yet. Algorithms for the spread spectrum model and pseudonoise sequence is discussed in section II. In section III, the spread spectrum model is presented. Section IV shows the system performance results and analysis from the model simulation. Conclusion and discussion are remarked in section V.

II. Algorithms

A. Spread Spectrum processing

The initial application of spread spectrum techniques was in the development of military guidance and communication systems. By the end of World War II, spectrum spreading for jamming resistance was already a familiar concept to radar engineerings [5]. Several years later, spread spectrum investigation was motivated primarily to achieve highly jam-resistant communication systems. As a result , applications on several fields are developed such as energy density reduction, high-resolution ranging, and multiple access, etc. Investigation of this topic will just focus on the part of multiple access.

In a spread spectrum system, a data signal is modulated with a binary pseudonoise (PN) signal to have a nearlyflat spectrum before transmission. Due to the added signal, the transmission bandwidth is much greater than the message bandwidth. At the receiver, the incoming signal is "despread" by correlating it with the PN signal. Figure 2. illustrates this process.

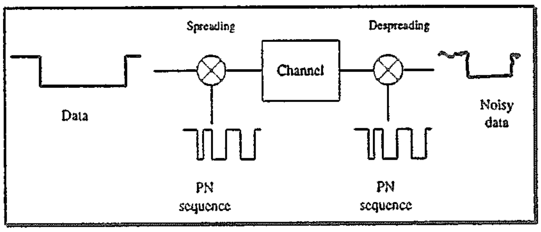

Fig. 2 Spreading/despreading processing

As noted above, the spreading of the data signal energy over a sufficiently wide bandwidth allows it to co-exist with narrowband signals with only a minimum of interference for either signal. Obviously, the low spectral density of the spread spectrum signal assures that it will cause little damage to the narrowband signal beyond that already caused by the ambient wideband noise in the channel. Although the narrowband signal has very high spectral density, this energy is concentrated near one frequency and is of very narrow bandwidth. The despreading operation of the spread spectrum receiver has the effect of spreading this narrowband energy over a wide bandwidth, and collapses the energy of the originally spread data signal down to the original data bandwidth. Therefore, after despreading, the situation is reversed between the original narrowband interferer (now wideband), and the original data signal (now narrowband). A bandpass filter can be used for the despreading process. Only the interferer power that falls in the bandwidth of the despread signal causes any interference. This will be only a fraction, 1/G, of the original narrowband interference that could have occupied that same bandwidth before despreading. Figure 3 illustrates the signal spectrum before and after despreading processing.

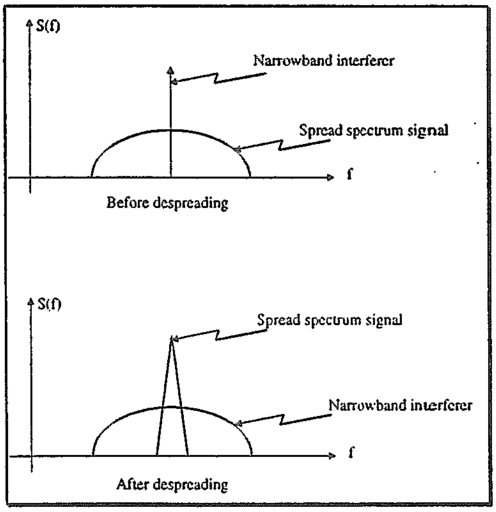

Fig. 3 Spectral effects

As mentioned above, information are spread and occupy a bandwidth much great than the minimum bandwidth necessary to send information. Since the bandwidth of the wired system was limited essentially, whether the spread spectrum signal can be transmitted in wired systems is an important issue.

Fortunately, The architecture of networks used for this approach is wormhole routing. In wormhole-routed networks, packets are further subdivided into flits. The bandwidth in wormhole-routed systems is not large. In general, a 256-nodes network required 8-bits per flits [4]. In addition, the PN code length can be adjusted to satisfy the bandwidth of wired system. Therefore, the bandwidth problem in wired system using spread spectrum techniques can be solved.

B. Pseudonoise Sequence

a. PN Code Generator

To generate a PN code, the PN code generator is used and shown in Figure 4. It is made up of a four-stage register for storage and shifting, a module-2 adder (equivalent to the logic exclusive-or operation), and a feedback path from the adder to the input of the register.

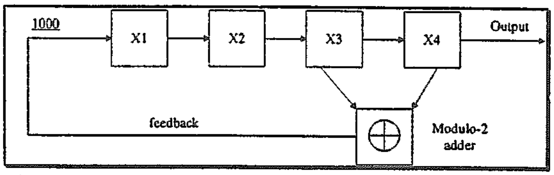

Fig. 4 PN code generator

Assume that stage X1 is initially filled with a one and the remaining stages are filled with zeros, this means, the initial state of the register is 1000. According to Figure 4, the succession of register states is as the following:

1000 0100 0010 1001 1100 0110 1011 0101 1010 1101 1110 1111 0111 0011 0001 1000

Since the last state, 1000, corresponds to the initial state, the register repeats the foregoing sequence after 15 clock pulses. The output sequence is obtained by noting the contents of stage X_4 at each clock pulse. The output sequence is observed to be

0001 0011 0101 111

, where the leftmost bit is the earliest bit.

Let us test the above output sequence for the randomness properties outlined in the preceding segment.

i).Balance property:

There are seven zeros and eight ones in this sequence. The numbers of zeros and ones of a PN code are different only by one. It satisfies the balance property shown above.

ii).Run property:

There are four "zero" runs (or "ones" runs):runs=4. Half of runs have the length 1; i.e., two single "zeros (or ones)." One-fourth of runs have the length 2; i.e., one "2 consecutive zeros (or ones)." One-eighth of runs are of length 3; i.e., one "3 consecutive zeros (or ones)." In the above example, 1/8 of runs cannot be counted for too short a code. The correlation property is treated in next segment.

The shift register generator produces sequences that depend on the number of stages, the feedback tap connections, and initial conditions. The output sequences can be classified as either maximal length or nonmaximal length. Maximal length sequences have the property that for an n-stage linear feedback shift register the sequence repetition period in clock pulses p is

$$p=2^n-1 \qquad (1)$$

Therefore, it can be observed that the sequence generated by the shift register generator of Figure 4 is an example of a maximal length sequence. If the sequence length is less than (2^n-1), the sequence is classified as a nonmaximal length sequence. Table 1 [11] show the proper feedback connections for several values of n.

n	Sequence Length	Sequence (Initial State: All Ones)	Feedback Digit
2	3	110	$x_1 \oplus x_2$
3	7	11100 10	$x_2 \oplus x_3$
4	15	11110 00100 11010	$x_3 \oplus x_4$
5	32	11111 00110 10010 00010 10111 01100 0	$x_2 \oplus x_5$

Table 1. Feedback connections for generation PN codes.

b. PN Autocorrelation Function

The autocorrelation function $R_x(\tau)$ of a periodic waveform x(t), with period T_o, is given in Equation (2) and is shown below in normalized form.

$$R_x(\tau)=\frac{1}{T_0}\int_{-T_0/2}^{T_0/2}x(t)x(t+\tau)dt$$

$$for -\infty<\tau<\infty \qquad (2)$$

$$R_x(\tau)=\frac{1}{K}\frac{1}{T_0}\int_{-T_0/2}^{T_0/2}x(t)x(t+\tau)dt$$

$$for -\infty<\tau<\infty \qquad (3)$$

where

$$K=\frac{1}{T_0}\int_{-T_0/2}^{T_0/2}x^2(t)dt \qquad (4)$$

When x(t) is a periodic pulse waveform representing a PN code, we refer to each fundamental pulse as a PN code symbol or a chip. For such a PN waveform of unit chip duration and period p chips, the normalized autocorrelation function may be expressed as

$$R_x(\tau)=\frac{1}{p}(N_s-N_d) \qquad (5)$$

where D denote the "difference," and S denote the "same" by compareing two PN sequence. N_d denote the number of D, and N_s denote the number of S.

The normalized autocorrelation function for a maximal length sequence, $R_x(\tau)$, is shown plotted in Figure 5. It is clear that for $\tau = 0$, that is, when x(t) and its replica are perfectly matched, $R(\tau) = 1$. However, for any cyclic shift between x(t) and $x(t + \tau)$ with $(1 \leq \tau < p)$, the autocorrelation function is equal to -1/p (for large p, the sequences are virtually decorrelated for a shift of a single chip).

It is now easy to test the output PN sequence of the shift register in Figure 4 for the third randomness property-correlation . Below is shown the output sequence; also shown is the same sequence with a single end-around shift:

```
0 0 0 1 0 0 1 1 0 1 0 1 1 1 1
1 0 0 0 1 0 0 1 1 0 1 0 1 1 1
DSSD DSDS DDDD SSS
```

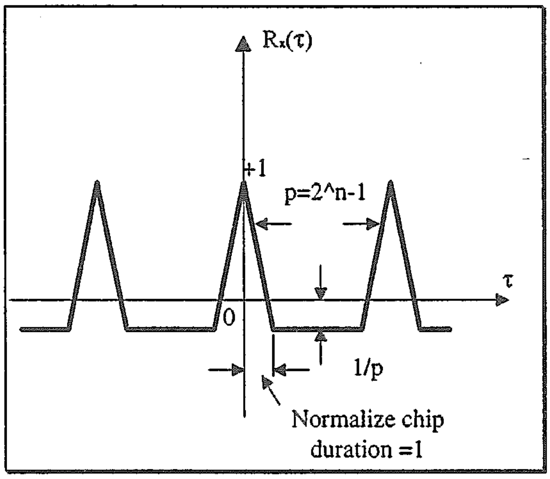

Fig. 4 PN autocorrelation function

Following Equation (5), the value of the autocorrelation function for this single one-chip shift is seen to be

$$R(\tau=1)=\frac{1}{15}(7-8)=-\frac{1}{15}$$

Any cyclic shift yielding a mismatch from perfect synchronization results in the same autocorrelation value, -1/p. Hence the sequence meets the third randomness property.

III. Simulation Model

A. Bit Rate of Using Virtual Channel and Spread Spectrum Basis

Figure 6. show three PN code sets are transmitted into a physical channel simultaneously. Since the concept of virtual channel is time-sharing, it will be seen as time division multiple access (TDMA)[10,12]. Assume that the capacity of the communication resource (the bandwidth of the wire) is R bits/s. For using virtual channel system (TDMA), M virtual channels are used (the frame is divided into M orthogonal time slots). Hence each of the M sources bursts its transmission at R bits/s.

Let the information generated by each of the sources be organized into f bits per packet, and f-bit packets are transmitted in T seconds over each of the M disjoint channels.

In the case of virtual channels, the f bits are transmitted in T/M seconds from each source. Therefore, the bit rate, W_{vc}, required is

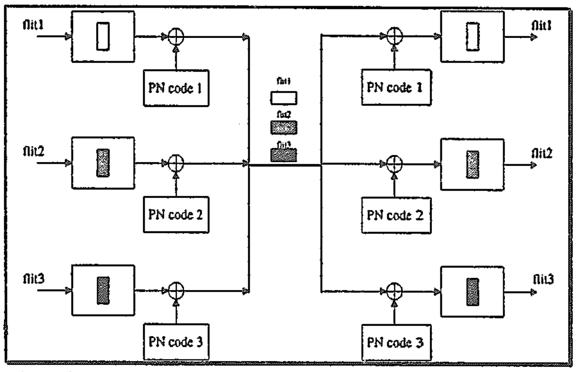

Fig. 6 Three PN code sets are transmitted into a physical channel simultaneously.

$$W_{VC}=\frac{f}{T/M}\ (=\mathrm{R})\quad \text{bits/s} \qquad (6)$$

In the case of spread spectrum(SS) basis, assume that N lengths of PN code is used. Therefore, the bit rate, W_{ss}, required is

$$W_{ss}=NR \quad \text{bits/s} \qquad (7)$$

b. Message Delay In Using Virtual Channel and Spread Spectrum basis

As before, we assume that in the case of using virtual channels, M virtual channel are used, and in the case of spread spectrum basis, N lengths of PN code is used. For analysis of message delay, the simplest case is that of deterministic data sources. It is assumed that the communication resources is 100% utilized, so that all time slots are filled with data packets. For simplicity, it is also assumed that there are no overhead costs such as guard times.

The message delay, D, can be defined as

$$D=w+\tau \qquad (8)$$

where w is the average flit waiting time, and τ is the flit transmission time. In the case of virtual channels, each flit is sent in slots of T/M seconds. The virtual channel flit transmission time, τ_{vc}, with Equation (6).

$$\tau_{VC}=\frac{T}{M}=\frac{f}{R} \qquad (9)$$

$$w_{VC}=\frac{T}{2}\left(1-\frac{1}{M}\right) \qquad (10)$$

For the virtual channel(TDMA) system [10]. So, the message delay of virtual channel system is

$$D_{VC}=w_{VC}+\tau_{VC}=\frac{T}{2}\left(1-\frac{1}{M}\right)+\frac{f}{R} \qquad (11)$$

For the spread spectrum (SS) basis, the average flit waiting time,

$$w_{SS}=0 \qquad (12)$$

and the SS flit transmission time, τ_{ss}, with the used of Equation (7).

$$\tau_{SS} = \frac{N+f}{NR} \qquad (13)$$

Thus, the message delay of SS, is that

$$D_{SS} = w_{SS} + \tau_{SS} = \frac{N+f}{NR} \qquad (14)$$

For wormhole-routed networks, assume that there are K nodes in the networks. The total message delay can be written as $D \times K$. For the used of virtual channels, the total message delay is

$$D_{VC,T} = \left(\frac{T}{2}\left(1-\frac{1}{M}\right)+\frac{f}{R}\right)\times K \qquad (15)$$

And for the used of SS basis, the total message delay can be written as follow:

$$D_{SS,T} = \left(\frac{N+f}{NR}\right)\times K \qquad (16)$$

IV. Results

Based on the above model, Equation (15)-Equation(16), system performance results are simulated and shown below in tables. Simulation parameters are defined as the following: the flit, f=8 bits, the virtual channels, M=2,3,4, the lengths of PN code, N=15 (here, three PN code sets are used), and the amount of nodes, K=4,16,32,64,128. Finally, the transmission rate R, for Table 2, Table 3, and Table 4 are 10 Kbits/s, 15 Kbits/s, and 20 Kbits/s, respectively. For time slots, T=1.6 ms, 2.4 ms, 3.2, ms for 2, 3, and 4 virtual channels, respectively in Table 2. Time slots for Table 3 and Table 4 can be calculated based on the transmission rate mentioned previously.

	$D_{VC,T}(s)$			$D_{SS,T}(s)$
K	2	3	4	
4	9.6e-3	0.0104	0.0117	6.13e-4
16	0.0384	0.0416	0.0469	2.45e-3
32	0.0768	0.0832	0.0939	4.91e-3
64	0.1536	0.1664	0.1877	9.81e-3
128	0.3072	0.3328	0.3755	0.0196

Table 2 The case of transmission rate R=10 Kbits/s

In wormhole-routed networks, when amount of virtual channels increases, system time latency increases also. This situation can be observed in Table 2- Table 4. From Table 2, Table 3, and Table 4, results can obviously observed that the total message delay of SS basis system is more less than that in virtual channels. It is because there needs no precise time coordination among various simultaneous transmitters, and may transmit several data simultaneously using different PN codes.

	$D_{VC,T}(s)$			$D_{SS,T}(s)$
K	2	3	4	
4	6.4e-3	6.93e-3	7.81e-3	4.09e-4
16	0.0257	0.0277	0.0313	1.64e-3
32	0.0513	0.0555	0.0625	3.27e-3
64	0.1026	0.1109	0.1250	6.54e-3
128	0.2052	0.2219	0.2500	0.0131

Table 3 The case of transmission rate R=15 Kbits/s.

	$D_{VC,T}(s)$			$D_{SS,T}(s)$
K	2	3	4	
4	4.8e-3	5.2e-3	5.87e-3	3.07e-4
16	0.0192	0.0208	0.0237	1.23e-3
32	0.0384	0.0416	0.0469	2.45e-3
64	0.0768	0.0832	0.0939	4.91e-3
128	0.1536	0.1664	0.1877	4.81e-3

Table 4 The case of transmission rate R=20 Kbits/s.

The bandwidth limited in wired system using the spread spectrum techniques is considered and discussion in Table 5. The wired system bit rate is assumed to be limited in R=900 Kbits/s. Other parameters are set as the same as the above. The use of SS bit rate must be adjusted in R=900/15 Kbits/s=60 Kbits/s to satisfy the system bandwidth. Therefore, the message delay can be observed in Table 5.

	$D_{VC,T}(ms)$			$D_{SS,T}(ms)$
K	2	3	4	
4	0.1067	0.1156	0.1304	0.1022
16	0.4267	0.4623	0.5215	0.4089
32	0.8534	0.9245	1.0431	0.8178
64	1.7068	1.8490	2.0861	1.6356
128	3.4136	3.6981	4.1722	3.2711

Table 5 The case of bit rate limited in R=900 Kbits/s

In Table 5, the SS message delay is less than that in virtual channel, even though the system bandwidth is limited. The major reason is it is not necessary for channel waiting. The deadlock situation is also avoidance, since several data are transmitted simultaneously using different PN code sets. To achieve more efficient system, the large length of PN code is needed. However, the longer the PN code length used, the larger the bandwidth needed. The situation must be considered carefully.

V. Conclusion

A coding approach with virtual channel free in the wormhole-routed networks is introduced. Comparing with the use of virtual channels, is that there need no precise time coordination among the various simultaneous transmitters. The transmission data between the use of different PN codes is not affected each other. From this approach, system time latency is reduced. The deadlock situation is also prevented. While the large length of PN code is used, the better efficient can be achieved. However, the wired bandwidth must be considered. Experimental results and comparison are the future work on this topic.

Reference:

[1] W. J. Dally and C. L. Seitz, "The torus routing chip," Journal of Distributed Computing, vol. 1, no. 3, pp. 187-196, 1986.

[2] Xiaola Lin, Philip K. McKinley, and Lionel M. Ni, "The message flow model for routing in wormhole-routed networks," accepted for publication on IEEE TDPS.

[3] L. M. Ni and P. K. McKinley, "A survey of wormhole routing techniques in direct networks," IEEE Computer, vol. 26, pp. 62-76, Feb. 1993.

[4] K. Hwang, "Advance Computer Architecture: Parallelism, Scalability, Programmability," McGraw-Hill, Inc., 1993.

[5] Scholtz, R. A. , "The origins of spread spectrum communications," IEEE Trans. Commun., vol. COM30, no. 5, pp. 822-854, May 1982.

[6] R. J. McEliece, Finite field for computer Scientists and engineers, Kluwer Academic, 1987.

[7] R. Lupas and S. Verdu, "Linear multiuser detectors for synchronous code division multiple access channel," IEEE Trans. On Info. Theory, vol. IT-35, pp. 123-136, Jan. 1989.

[8] R. Lupas and S. Verdu, "Near-far resistance of multiuser detectors in asynchronous channels," IEEE Trans. on Commun., vol. COM-38, pp. 496-508, April 1990.

[9] S. Verdu, "Minimum probability of error for asynchronous Gaussian multiple-access channels," IEEE Trans. on Info. Theory, vol. IT-32, pp. 85-96, Jan. 1986.

[10] Bernard Sklar, "Digital Communications: fundamentals and applications," Prentice-Hall, Inc., 1988.

[11] Ziemer Tranter, "Principles of Communications: system, modulation, noise. Fourth Edition," Houghton Mifflin Company, 1995.

[12] Kaveh Pahlavan and Allen H. Levesque, "Wireless information networks," John Wiley & Sons, Inc., 1995.