Double-Data-Rate, Wave-Pipelined Interconnect for Asynchronous NoCs

DWP, a new interconnect structure for asynchronous networks on chip in multiprocessing SoCs, yields higher throughput, consumes less power, suffers less from crosstalk noise, and requires less area than traditional interconnect structures. Its advantages stem from techniques including wave pipelining, double-data-rate transmission, interleaved lines, misaligned repeaters, and clock gating.

Jiang Xu
Hong Kong University of Science and Technology

Wayne Wolf
Georgia Institute of Technology

Wei Zhang
Princeton University

To effectively reduce cost, improve reliability, meet performance requirements, and produce versatile products, designers of multiprocessor systems on chip (MPSoCs) not only require efficient functional units, they must also emphasize cooperation among these units. The on-chip communication subsystem, which determines the effectiveness of this cooperation, has gradually evolved from buses and ad hoc interconnects into sophisticated networks on chip (NoCs).[1] On-chip interconnect structures, the fundamental elements of NoC design, greatly affect design choices at the architectural level. But as the smaller feature sizes of each new generation of process technology speed the migration toward MPSoCs, they also make on-chip communications more difficult.

Figure 1, based on data from the International Technology Roadmap for Semiconductors (http://www.itrs.net), compares the delays of transistors and global interconnect at each process generation. While the transistor delay decreases exponentially in each generation, the interconnect delay increases exponentially. Although designers can insert more and larger repeaters to reduce the delay (at the cost of greater power consumption), the basic trend can't be reversed. As feature sizes shrink, on-chip communication will need not one but multiple clock cycles to send a bit from a source to its destination.[2] Moreover, reduced feature sizes make crosstalk noise significant enough to threaten signal integrity; lower power supply voltage exacerbates this situation.[3] Another complication arises from the fact that NoCs will have to communicate asynchronously; globally asynchronous, locally synchronous (GALS) clock schemes offer an effective way to save power on distributing and synchronizing high-frequency clock signals throughout chips, and hence NoCs have to operate asynchronously.[4] Some low-power techniques, such as dynamic voltage-frequency scaling, also require asynchronous communication among different

Published by the IEEE Computer Society

clock domains. All these issues challenge the traditional interconnect structure and call for innovation.

In this article, we introduce DWP, a new on-chip interconnect structure that uses double-data-rate transmission and wave pipelining to increase throughput, and interleaved lines and misaligned repeaters to reduce crosstalk noise. (See the "Previous Work in Wave Pipelining" sidebar for an introduction to this technique.) DWP can operate synchronously as well as asynchronously, so it's suitable for MPSoC designs using GALS clock schemes. With these capabilities, DWP provides a more flexible design space for NoCs than the traditional interconnect structure, and it requires only half the data lines. After introducing DWP and its features, we analyze its performance in a case study, applying it to the asynchronous NoC for an H.264 HDTV decoder MPSoC.

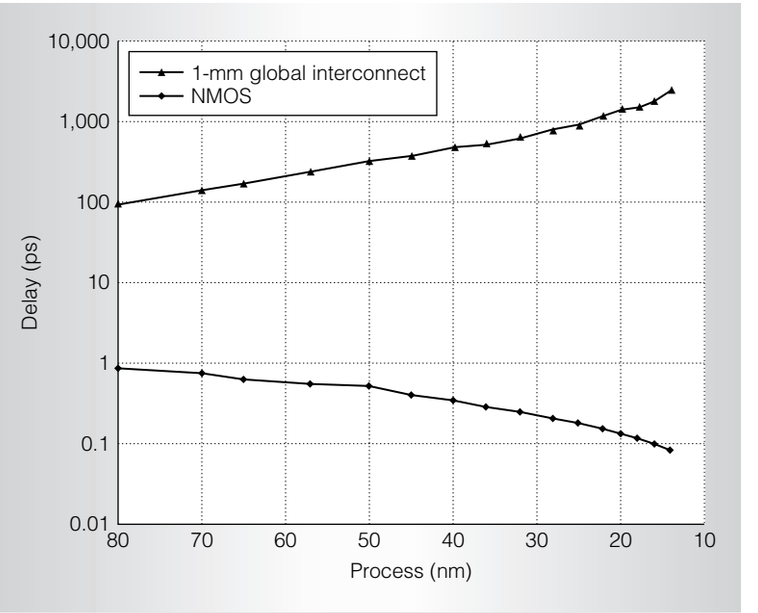

Figure 1. As feature sizes shrink, transistor delays decrease exponentially while global interconnect delays increase exponentially.

DWP interconnect structure

In general, a point-to-point interconnect connects two functional units on a chip. The functional unit that sends data is called the *source,* and the unit receiving the data is called the *destination.* An interconnect consists of transmitters, receivers, and lines, which are composed of metal wires and repeaters. DWP consists of transmitters, data and clock/control lines, receivers, and clock buffers (see Figure 2). Data and clock/control signals are fed into the transmitters, and then transmitted over the data and clock/control

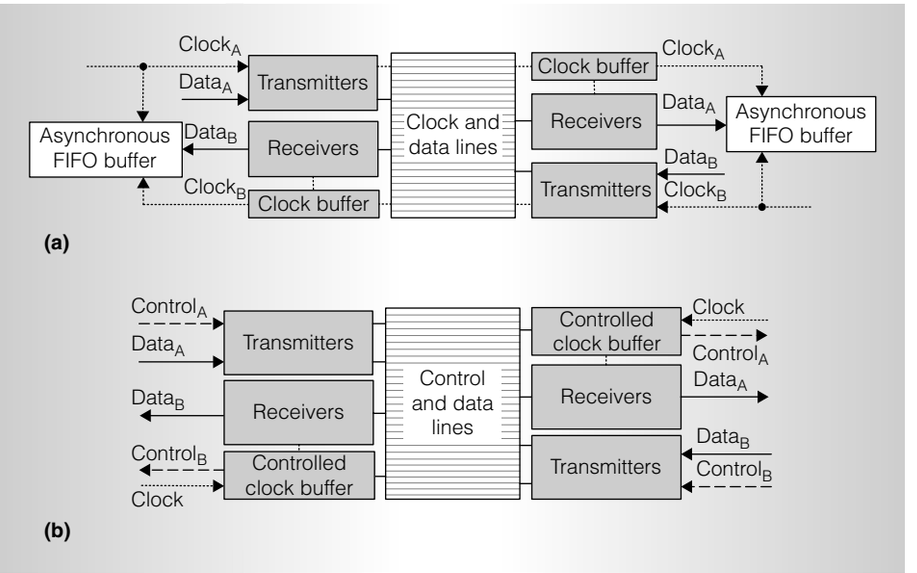

Figure 2. The double-data-rate, wave-pipelined interconnect (DWP): asynchronous structure (a) and synchronous structure (b).

Previous Work in Wave Pipelining

Global interconnect delay increases exponentially with every new technology generation, and this trend is made worse by increasing working frequency. Under long delays, pipelining can effectively increase throughput. Wave-pipelining theory was developed by Cotton, who called it maximum-rate pipelining.[1] At the root of the theory is the fact that transistors and wires store signals for a short period of time when they transform and transmit them. Except at the inputs and outputs, wave pipelining doesn't require latches to separate pipeline stages, and signals are pipelined directly in circuits.[2] Usually wave pipelining is difficult to design and implement, because multiple paths from one input to one output of a circuit often have different delays, and all the delays of these paths must be balanced. However, wave pipelining has the advantage of not incurring the delay, area, and power overheads of latches.

In a wave-pipelined interconnect, we define the number of pipeline stages as the number of bits that exist simultaneously on the interconnect. We calculate this value as $N_p = D \times f$, where N_p is the number of pipeline stages, D is the delay from an input to an output, and f is the operating frequency. As we show in this article, a fractional number of pipeline stages is possible in a wave-pipelined system.

For global interconnects, wave pipelining is more suitable than pipelining using latches for several reasons. First, using latches to pipeline global interconnect is costly. In fact, latches on global interconnects are either much larger than those in pipelined logic circuits or require large chain buffers, because they must drive large loads caused by long wires in the subsequent pipeline stage. These large latches or buffers require significant power and area, and they add an extra penalty to the global interconnects' already long delay. Because wave pipelining doesn't need these latches, it avoids the related power, area, and delay overheads. Second, because data signals are self-timed between the inputs and outputs of a wave-pipelined interconnect, wave pipelining eases the design of global interconnects crossing multiple clock domains. Third, interconnect circuits are relatively simple. One-bit slices are identical to each other and have very low delay path diversity. This greatly simplifies the job of balancing delay.

Several groups have studied wave-pipelined interconnects. We previously proposed using wave pipelining for the global interconnects in MPSoCs and published preliminary designs in 0.25-μm technology.[3] A comparison with traditional interconnects shows a significant throughput improvement and moderate power savings. Hashimoto, Tsuchiya, and Onodera show significant advantages for wave-pipelined global interconnects over traditional and differential interconnects in terms of throughput, latency, area, and power in 45-nm technology.[4] Deodhar and Davis proposed a method to simultaneously optimize voltage, repeater insertion, and wire sizes for wave-pipelined interconnects.[5] They also compared the optimized wave-pipelined interconnects with interconnects using low-voltage differential singling, and showed that the wave-pipelined interconnect reduced area by 70 percent and power by 10 percent for the same throughput. Joshi, Lopez, and Davis proposed circuits for wave-pipelined multiplexed routing, which uses wave-pipelined interconnects.[6] They also showed that the wave-pipelined interconnect can tolerate a wide range of process variations and operating conditions. Zhang, Hu, and Chen proposed a wave-pipelined circuit based on the phase-locked loop (PLL).[7] This design achieves greater throughput and uses less power and area than a traditional pipeline. They found that the break-even point is 6.5-mm in 0.18-μm technology, where wave-pipelined interconnects begin to perform better than traditional interconnects.

References

1. L. Cotton, ''Maximum Rate Pipelined Systems,'' *Proc. AFIPS Spring Joint Computer Conf.* (AFIPS 69 Spring), ACM Press, 1969, pp. 581-586.
2. W.P. Burleson et al., ''Wave-Pipelining: A Tutorial and Research Survey,'' *IEEE Trans. Very Large Scale Integration Systems,* vol. 6, no. 3, Sept. 1998, pp. 464-474.
3. J. Xu and W. Wolf, ''Wave Pipelining for Application-Specific Networks-on-Chip,'' *Proc. Int'l Conf. Compilers, Architecture, and Synthesis for Embedded Systems* (CASES 02) ACM Press, 2002, pp. 198-201.
4. M. Hashimoto, A. Tsuchiya, and H. Onodera, ''On-Chip Global Signaling by Wave Pipelining,'' *Proc. IEEE 13th Topical Meeting on Electrical Performance of Electronic Packaging* (EPEP 04), IEEE Press, 2004, pp. 311-314.
5. V.V. Deodhar and J.A. Davis, ''Optimal Voltage Scaling, Repeater Insertion, and Wire Sizing for Wave-Pipelined Global Interconnects,'' *IEEE Trans. Circuits and Systems I,* vol. 55, no. 4, May 2008, pp. 1023-1030.
6. A.J. Joshi, G.G. Lopez, and J.A. Davis, ''Design and Optimization of On-Chip Interconnects Using Wave-Pipelined Multiplexed Routing,'' *IEEE Trans. Very Large Scale Integration Systems,* vol. 15, no. 9, Sept. 2007, pp. 990-1002.
7. L. Zhang, Y. Hu, and C.-P. Chen, ''Wave-Pipelined On-Chip Global Interconnect,'' *Proc. Asia and South Pacific Design Automation Conf.* (ASP-DAC 05), vol. 1, IEEE CS Press, 2005, pp. 127-132.

lines respectively. The transmitters convert data from single data rate (SDR) to double data rate (DDR), which is twice the clock frequency. The receivers catch the data signals and convert them back to SDR. By using DDR, DWP reduces the number of data lines that traditional interconnects require by 50 percent.

With slightly different clock buffers, DWP can be used for both asynchronous

and synchronous communication. For asynchronous communication (Figure 2a), the source clock signal is transmitted along with data signals over the clock line, and amplified by the clock buffer before driving the receivers. Data must be synchronized to the destination clock, which can be done by asynchronous FIFO buffers.[5] For synchronous communication (Figure 2b), a control signal that indicates data availability is transmitted along with data signals. The receivers are driven by the destination clock signal, which is gated by the source control signal through the controlled clock buffer.

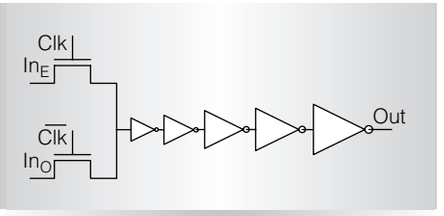

Figure 3. Double-data-rate (DDR) transmitter converts single-data-rate (SDR) data signals into DDR data signals.

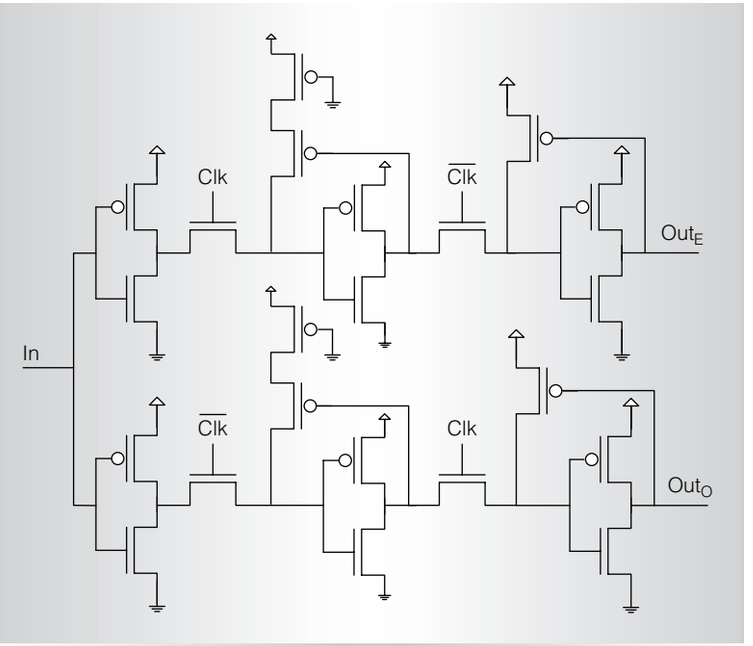

Figure 4. The semistatic double-edge-triggered flip-flop (SDETFF) is designed to receive high-speed DDR data signals.

Transmitters, receivers, and clock buffers

The transmitters convert SDR data signals into DDR data signals by transmitting even-numbered bits on the clock's falling edges and odd-numbered bits on its rising edges (see Figure 3). NMOS transmission gates are used to select corresponding bits based on the clock signal. The transmitters use a cascaded driver circuit based on a chain of inverters to boost the input data signals. The inverters are sized to achieve the shortest delay, and the transmitter inputs are specified to be driven by the minimum-sized inverter. This specification also applies to DWP's clock and control inputs.

Similar to the transmitters, the clock buffer uses a chain of inverters to amplify the clock signals to drive the receivers. The clock buffer for the asynchronous structure amplifies the clock signal from the source and drives the receivers. For the synchronous structure, the destination clock signal is gated by the control signal from the source through an NMOS transmission gate. Clock gating saves power by reducing the clock activities in the clock buffer. Because of a large fan-out, the critical paths of the wave-pipelined interconnect structures are associated with the control signals for the synchronous structure and the clock signals for the asynchronous structure.

The receivers catch data signals and convert DDR data signals into SDR data signals. We designed a semistatic, double-edge-triggered flip-flop (SDETFF) for the receivers (see Figure 4). To match the path delay of the clock and control signals, the receiver uses inverters to delay and rectify the data signals. SDETFFs in the receivers then catch the data signals. The SDETFF, which can effectively reduce the clock fan-out and operates at a high frequency, has three stages separated by NMOS transmission gates. In the first stage, two inverters isolate the two sides. Because of imperfect clock signals and timing, the two sides of a DETFF will interfere with one another's input signal; the isolation inverters eliminate this interference. In the second stage, two PMOS transistors give a weak feedback to maintain latched data and improve the operation speed. In the third stage, one PMOS transistor gives stronger feedback. Instead of CMOS transmission gates, NMOS transmission gates are used to

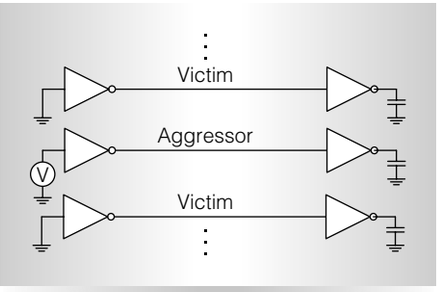

Figure 5. This circuit model is used to analyze crosstalk noise among adjacent metal wires in an interconnect.

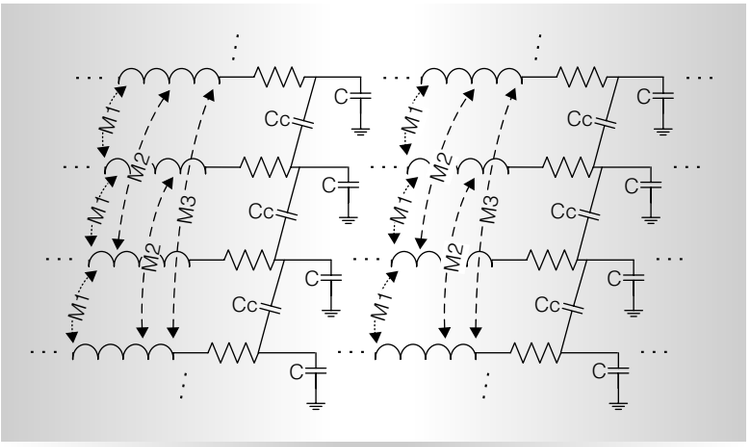

Figure 6. The wire model with mutual inductances and coupling capacitances is used to capture crosstalk noise in simulations.

reduce the input capacitance and clock fanout. We set the width of the NMOS transmission gates to 130 nm to reduce the channel resistance.

Data, clock, and control lines

DWP's lines, composed of metal wires and repeaters, transmit data as well as clock and control signals. In synchronous circuits, a dedicated clock network distributes clock signals. In this case, it doesn't need to transmit source clock signal along with data, and the destination clock can be used. The control signal, which indicates data availability, is transmitted along with the data through the control line. In GALS clock schemes, clock information must be transmitted with the data whenever an interconnect connects two clock domains. One method of transmitting clock information with data is to regenerate the clock signal from data signals using phase-locked loops (PLLs). This method saves clock lines, but requires additional high-frequency PLLs and even special signal schemes, which make the design more complex and introduce extra delay.

We choose to transmit an explicit clock signal using the clock line. There are no separate control lines in asynchronous DWP. Instead, control signals, which mark the beginning and the end of a data transmission, are embedded into the clock signal by clock gating. The source clock is only transmitted when data is present. Each clock edge corresponds to a set of data on the data lines. Clock gating saves power by reducing the switching activities on the clock lines. Because we use DDR transmission, the clock frequency is only half of the data rate, which further reduces power consumption.

Crosstalk noise. The simple and uniform interconnect structure greatly simplifies the job of delay balancing, but an interconnect's parallel structure is prone to crosstalk noise among adjacent lines. Crosstalk, which takes effect through coupling capacitance and mutual inductance among parallel lines, distorts signals and causes large delay variations among 1-bit slices. We analyzed the crosstalk among adjacent metal wires using the circuit shown in Figure 5. One of the parallel lines, the *aggressor,* is active*;* the other lines are called *victims.*

Figure 6 shows a lumped wire model that considers mutual inductance up to the third nearby wire. Figure 7 shows the simulation results. Through coupling capacitances, increasing the voltage on the aggressor drives the voltage on the victim to increase, and decreasing the voltage induces a voltage drop on the victim. Through mutual inductances, an increasing current on the aggressor causes a voltage drop on the victim, and a decreasing current leads to a voltage increase on the victim.

In terms of delay, coupling capacitances cause signals in nearby lines to accelerate each other if they both change from high to low or low to high, and to hold each other back if they change in opposite directions. Mutual inductances have a similar effect but over a longer effective range than

coupling capacitances. Crosstalk can not only increase delays to the worst case, but also decrease delays to the best case.

The best-case delays are as important as the worst-case delays when we're determining the lower bound of a wave-pipelined system's clock period. As a demonstration of this principle, consider the following analysis of a wave-pipelined interconnect with clocked latches at the input and output. The clock has a period of T_c, and there is a phase difference α between the clocks used by the input and output latches. The single edge skew of the clock is $\pm\Delta$. The output latch has a setup time T_s and a hold time T_h. The worst-case and best-case delays are D_{max} and D_{min}, respectively. Similar to Burleson et al.,[6] we can show the effect of the interconnect delay variation $V = D_{max} - D_{min}$ in the following inequality:

$$T_c > V + 2\Delta + T_s + T_h.$$

However, in real signal schemes, when a signal edge reaches the worst case, the very next signal edge can't reach the best case, and vice versa. In this case, $V < D_{max} - D_{min}$. Our simulation shows that V is close to $(D_{max} - D_{min})/2$. Therefore, we get

$$T_c > (D_{max} - D_{min})/2 + 2\Delta + T_s + T_h. \quad (1)$$

This inequality shows that the maximum frequency of a wave-pipelined system, $1/T_c$, is bounded by the delay variation between the worst-case and the best-case delays. Hence, we must reduce the delay variation to improve a wave-pipelined system's performance.

Reducing crosstalk. In DWP, we combined two methods to reduce the delay variation. The first method uses interleaved lines, and the second uses misaligned repeaters. The two methods reduce delay variation from 292 ps to 41 ps on a 10-mm DWP, a reduction of 86 percent. In the first method, we interleave the data and clock/control lines of one direction with those of the other direction, as Figure 8 shows. Interleaving the lines increases the distance among the lines of each direction. When the two directions aren't active at the same time, the lines of the idle direction serve as grounded shields for those of the active direction. When both directions are active at the same time, misaligned repeaters reduce crosstalk among nearby lines.

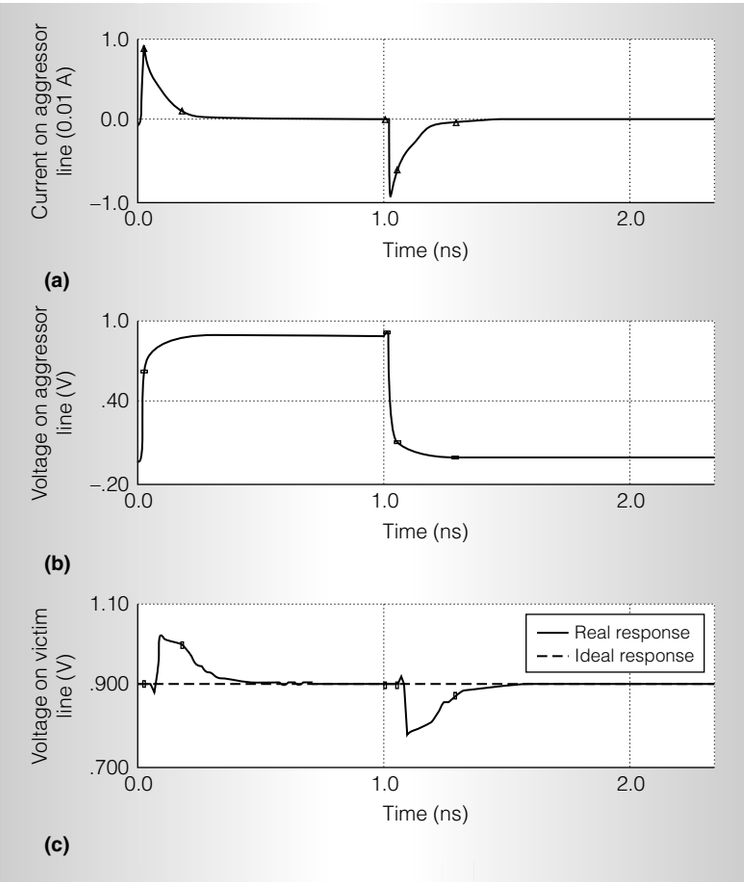

Figure 7. Crosstalk noise simulation results: current on aggressor line (a), voltage on aggressor line (b), and voltage on victim line (c).

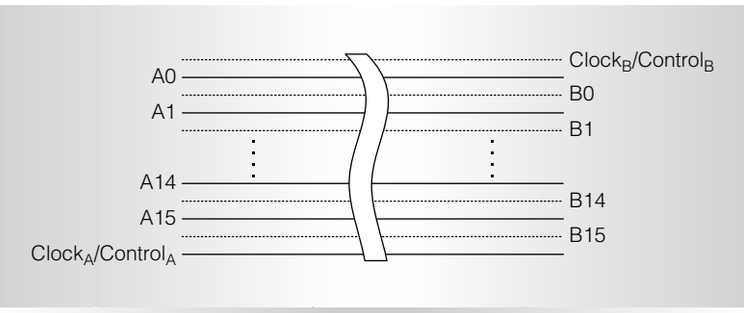

Figure 8. First crosstalk reduction method: Interleaved lines allow the lines of the inactive direction to shield those of the active direction.

Table 1. DWP vs. traditional interconnect.

Interconnect type	Frequency (GHz)	Throughput (Gbps)	Delay variation (ps)	Silicon area (mm^2)	Metal area (mm^2)	Power consumption (pJ/bit)	Leakage power (μW)
32-bit traditional	1.1	35	292	0.00133	0.9	2	352
32-bit DWP	5	160	41	0.0008	0.47	1.5/1.3	196

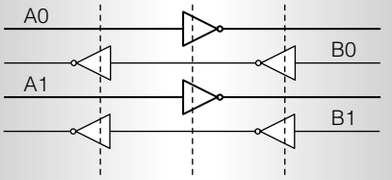

Figure 9. Second crosstalk reduction method: Misaligned repeaters reduce the crosstalk among nearby lines by generating reversed crosstalk noise on the same section of wire.

In the second method, we place a repeater of one direction between two repeaters of the other direction, as shown in Figure 9. Kahng et al. use a similar technique among lines of the same direction.[7] The reversed voltage and current changes on the two sides of a repeater will induce crosstalk noise in reverse phases, which will cancel each other on the same section of wire. The clock line is always outermost because it has the most switching activity and is noisy. The clock signals use the same lines as data signals, which makes it easier to match delays between data and clock.

Useful properties

DWP has two useful properties. First, the working frequency of asynchronous DWP can be changed dynamically. Because the clock signal is always synchronized with the data signals along the clock and signal lines, when the inputs dynamically change the data rate and the corresponding clock signal, the receivers can still catch the right data signals. This feature helps to implement dynamic frequency scaling and reduce the scaling overhead. Second, when DWP becomes longer or shorter, the working frequency doesn't change; only the number of pipeline stages changes. This implies that throughput will remain the same when the distance between the transmitter and receiver changes. This property will ease architectural design of NoCs, because regardless of the final floorplan, DWP can guarantee the required throughput.

Comparison and analysis

We simulated and analyzed a 10-mm DWP and compared it with the traditional interconnect structure. On the data and clock/control lines, we inserted repeaters every 1 mm. Our technology parameters come from the global interconnect feature sizes and transistor models for 65-nm technology from the Predictive Technology Model (http://www.eas.asu.edu/~ptm/). The operating voltage is 0.9 V. We chose a wire width of 0.45 μm, spacing of 0.9 μm, and thickness of 1.2 μm. We used the RLC model (resistor, inductor, capacitor) for every 10-μm wire section as in Figure 6. Our analysis considered mutual inductances in three nearest lines. For simulation, we used Cadence Spectre. The input signals are 200 ps wide, and the rising and falling edges are both 20 ps.

Our DWP works at 5 GHz, and the data rate is 10 Gbps on each data line. For a 32-bit DWP interconnect, the throughput is 160 Gbps for each direction, which is 4.5 times higher than the traditional interconnect structure (see Table 1). A traditional interconnect structure with the same feature sizes and length can work at 1.1 GHz and has a throughput of 35 Gbps. This high working frequency lets NoCs using DWP match the frequency of the IP cores in an MPSoC. This match can reduce the queue size at the NoC interface and hence the network queuing delay.

The 10-mm interconnect using a traditional structure has a delay variation of 292 ps. Using interleaved lines and misaligned

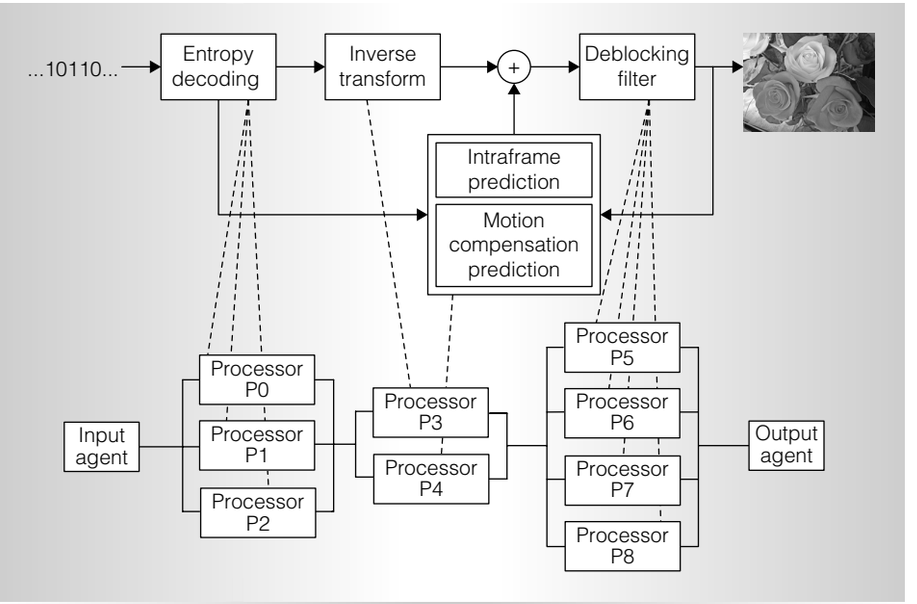

Figure 10. H.264 decoder architecture and mapping. The H.264 decoder is partitioned and mapped to an MPSoC architecture.

repeaters, DWP reduces the delay variation to 41 ps, an 86 percent reduction. As Inequality 1 shows, a small delay variation is critical to achieving a high working frequency. The 10-mm DWP has 2.3 pipeline stages; the fractional number is due to the different phases of the inputs and the receiver clock. This phase difference changes with the interconnect's length. Because the clock and control signals are transmitted along with the data signals using the same lines, they're naturally synchronized with each other.

The 32-bit DWP uses 0.0008 mm^2 silicon area and 0.47 mm^2 metal area, which are 40 and 48 percent less than the traditional interconnect. On average, asynchronous DWP consumes 1.5 pJ to transmit a 1-bit datum over 10 mm. Synchronous DWP doesn't transmit clock signals and consumes 1.3 pJ to transmit a 1-bit datum on average. The traditional interconnect structure needs 2 pJ to transmit a 1-bit datum. In comparison, synchronous DWP saves 35 percent in power. The leakage power of the 10-mm DWP is 196 μW, which is 44 percent less than the traditional interconnect. When transmitting data, the clock consumes 12 percent of the total power in asynchronous DWP. Thanks to our signaling scheme, when no data is sent, we can gate the clock signals and save considerably on power.

Asynchronous NoC case study

We applied DWP to an asynchronous NoC for an H.264 HDTV decoder MPSoC, and compared the performance, power consumption, and area with the same NoC using traditional interconnects. H.264 is an ITU-T recommendation for video that has improved coding efficiency and is more network friendly than previous video standards. It has been adopted by new storage standards, such as Blu-ray and HD DVD, and is a promising candidate for HDTV broadcast. However, H.264 is more complex than previous video standards, such as MPEG2. An H.264 HDTV decoder implementation requires low cost, low power, and high performance. MPSoC design for H.264 systems at high resolution, such as 720-pixel vertical resolution, progressive scan (720p), can benefit from NoCs. ITU-T specifies an H.264 decoder model as in Figure 10. Our case study is based on the H.264 reference model JM (http://iphome.hhi.de/suehring/tml).

Using a hardware/software codesign methodology, we designed an MPSoC and the corresponding NoC for the H.264

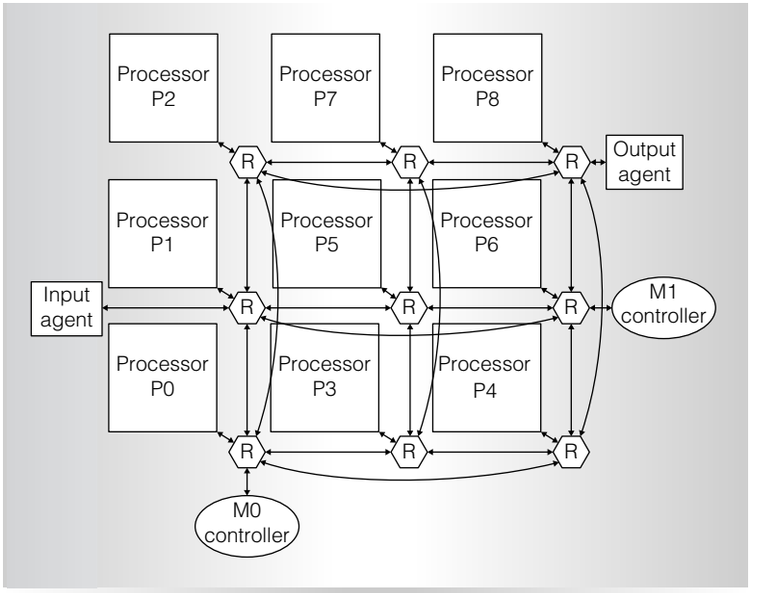

Figure 11. Network on chip for the H.264 HDTV decoder MPSoC. A 3 × 3 torus NoC connects the processors and peripherals.

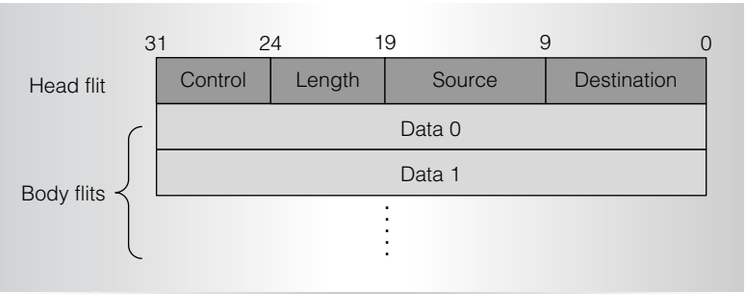

Figure 12. Packet format. A packet includes a head flit and several body flits.

HDTV decoder. The MPSoC targets 65-nm technology with a voltage supply of 0.9 V. We designed a computational architecture comprising nine embedded processors plus input and output agents. We partitioned and mapped the reference decoder model onto the computational architecture on the basis of the performance requirement of each processing stage of H.264. The input and output agents help organize the input video stream and decoded video frames. Three processors—P0, P1, and P2—implement the entropy decoding stage. Processor P3 implements the inverse transform stage. Processor P4 implements the intraframe prediction and motion compensation prediction stage. Processors P5, P6, P7, and P8 implement the deblocking filter stage.

Our network uses 2D torus structure, as shown in Figure 11. Because there are nine processors, we used a 3×3 torus. The memory controllers and input and output agents are located at the outskirts of the NoC. Memory controller M0 serves the input agent and processors P0, P1, and P2. Memory controller M1 serves the other processors and output agent. We optimized the positions of the processors, the input and output agents, and the memory controllers for system performance. For the MPSoC using the 2D torus NoC, the floorplan is similar to that shown in Figure 11. A 2-mm × 2-mm tile holds a processor and a router and forms a clock domain. We used an input-buffered pipelined crossbar router in the NoC. A processor is connected to a nearby router by a 0.1-mm 32-bit bidirectional local interconnect. Because the processor and router are close, we used the traditional interconnect structure for the local interconnects. The routers connect to each other by 2-mm and 4-mm 32-bit bidirectional global interconnects. For the global interconnects, we compared DWP with the traditional interconnect structure.

The NoC uses dimension-ordered routing. A packet has a 32-bit head flit, followed by up to 31 body flits (see Figure 12). The *destination* field holds the destination address; the *source* field holds the source address; the *length* field holds the size of the following body flits; the *control* field holds control information. Buffer metering is used for deadlock avoidance. When there isn't blocking, a router can process a packet header in one clock cycle, and the packet can be sent to a router port in the next cycle. If multiple packets compete for the same port, round-robin arbitration determines the winner.

To achieve the required performance, each processor should work at a minimum frequency of 3 GHz. To reduce the clock tree's power consumption, we used a GALS clock scheme, which requires asynchronous communication among processors. Each tile, which includes a processor and router, forms a clock domain. For both the 2-mm and 4-mm 32-bit interconnects, asynchronous

DWP can work at 3 GHz, which is the same as the processor's working frequency. Although 2-mm 32-bit traditional interconnects can work at 3 GHz, 4-mm 32-bit traditional interconnects are partitioned into two sections of 2 mm each to reach 3 GHz. To use the traditional interconnect structure in the GALS clock scheme, we inserted asynchronous FIFO buffers at the clock domain boundaries.

We compared the NoC using DWP to the NoC using the traditional interconnect structure in terms of performance, power consumption, silicon area, and metal area. Following the method we presented in an earlier publication,[8] we based the comparisons on Spice simulations of network components in Cadence Spectre, and cycle-accurate simulations for the whole NoC in the OPNET network simulator. As the results in Table 2 show, the MPSoC using the traditional interconnect can only process 22 frames per second, whereas DWP can process 27 frames per second—23 percent higher. While using DWP, the NoC for the H.264 decoder MPSoC consumes 0.79 mJ per frame—22 percent lower than the NoC using the traditional interconnect. DWP also helps to save 9 percent in silicon area and 43 percent in metal area for the NoC compared with the traditional interconnect.

Table 2. Traditional interconnect versus DWP: NoC performance, power, and area comparisons.

Interconnect type	Performance (frames per second)	Power consumption (mJ per frame)	Silicon area (mm^2)	Metal area (mm^2)
Traditional	22	1.02	0.036	5.1
DWP	27	0.79	0.033	2.9

An elegant yet efficient on-chip interconnect structure, DWP combines the advantages of wave pipelining, DDR transmission, interleaved lines, misaligned repeaters, and clock gating to achieve higher throughput, lower power consumption, lower area, and lower crosstalk noise than the traditional interconnect structure—using only half the data lines. DWP provides a larger design space for asynchronous NoCs. MICRO

Acknowledgments

This work is supported by the US National Science Foundation and the Research Grand Council of Hong Kong.

References

1. B. Towles and W.J. Dally, ''Route Packets, Not Wires: On-Chip Interconnect Networks,'' *Proc. 38th Design Automation Conf.* (DAC 01), IEEE CS Press, 2001, pp. 684-689.
2. J. Cong, ''An Interconnect-centric Design Flow for Nanometer Technologies,'' *Proc. IEEE,* vol. 89, no. 4, Apr. 2001, pp. 505-528.
3. M. Kuhlmann and S.S. Sapatnekar, ''Exact and Efficient Crosstalk Estimation,'' *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems,* vol. 20, no. 7, Jul. 2001, pp. 858-866.
4. T. Meincke et al., ''Globally Asynchronous Locally Synchronous Architecture for Large High-Performance ASICs,'' *Proc. IEEE Int'l Symp. Circuits and Systems* (ISCAS 99), vol. 2, IEEE Press, 1999, pp. 512-515.
5. T. Chelcea and S.M. Nowick, ''Robust Interfaces for Mixed-Timing Systems,'' *IEEE Trans. Very Large Scale Integration Systems,* vol. 12, no. 8, Aug. 2004, pp. 857-873.
6. W.P. Burleson et al., ''Wave-Pipelining: A Tutorial and Research Survey,'' *IEEE Trans. VLSI Systems,* vol. 6, no. 3, Sept. 1998, pp. 464-474.
7. A.B. Kahng et al., ''Interconnect Tuning Strategies for High-Performance ICs,'' *Proc. Design, Automation, and Test in Europe* (DATE 98), IEEE CS Press, 1998, pp. 471-478.
8. J. Xu et al., ''A Design Methodology for Application-Specific Networks-on-Chip,'' *ACM Trans. Embedded Computing Systems,* vol. 5, no. 2, May 2006, pp. 263-280.

Jiang Xu is an assistant professor in the Department of Electronic and Computer Engineering at Hong Kong University of Science and Technology. His research areas include systems on chip, networks on chip, multiprocessor systems, low-power VLSI design, nanoelectronic design, hardware/software codesign, and embedded systems. He has a PhD in electrical engineering from Princeton University.

Wayne Wolf is the Farmer Distinguished Chair and Georgia Research Alliance Eminent Scholar at the Georgia Institute of Technology. His research interests included embedded computing, embedded video and computer vision, and VLSI systems. Wolf has a PhD in electrical engineering from Stanford University. He is a Fellow of the IEEE and the ACM.

Wei Zhang is a PhD candidate in the Electrical Engineering Department at Princeton University. Her research interests include embedded systems, nanotechnology-based VLSI design, low-power reconfigurable systems, nanoeletronic design automation, and bio-inspired electronic systems. She has an MS in electrical engineering from Harbin Institute of Technology, China.

Direct questions and comments about this article to Jiang Xu, Dept. of Electronic and Computer Engineering, Hong Kong Univ. of Science and Technology, Clear Water Bay, NT, Hong Kong; jiang.xu@ust.hk