Two Efficient Pipelined Designs for MIMO Asynchronous Control

YING-HAW SHU, SHING TENQCHEN, MING-CHANG SUN, WU-SHIUNG FENG
Department of Electrical Engineering
National Taiwan University
5F, No. 69-10, SEC. 2, Chung Cheng E. RD. Tamshui, Taipei
TAIWAN

Abstract: - Two-phase micro-pipeline asynchronous modules show faster performance than common four-phase control systems, but the conventional systems with multi-port modules normally suffer from longer signal paths on stacked C-elements. NOR-based control schemes provide an alternative solution to problems such as propagation delay. This paper presents two modified designs from the common two-phase and alternative NOR based four-phase pipeline system. The HSPICE performs the evaluation based on TSMC 0.25um fast-mode CMOS model, and HSPICE simulation results show the two-phase pipelined system is still a reliable solution with a limited number of inputs even when the theoretically lower control overhead is disregarded. A power reduction of over 27% and a propagation improvement of more than 11% were achieved by replacing some decision circuits with modified C-elements.

Key-Words: - Interlocked Pipelined CMOS, two-phase pipeline, Symmetric C-element, Strobe, MIMO, Asynchronous

1 Introduction

The pipeline concept is widely employed in current asynchronous designs to get greater throughputs. By dividing tasks into several function-units, pipelined systems allow all the separate function-stages to work in parallel, therefore achieving higher performance [1-4]. Two pipelined control protocols are commonly used in asynchronous systems, the two-phase and the four-phase. In the event-driven or asynchronous protocol, the two-phase system uses both rising and falling transitions on the control bus to signify "data ready" and "data captured", as shown in Fig. 1(a) [5-7]. Under this control protocol, the control circuit does not wait for signal transition on the control bus. Thus, the two-phase protocol may be more efficient than the four-phase protocol, but needs a more complex double-edge triggered flip-flop. By contrast, the four-phase protocol treats only rising transitions on the control bus as "data ready" or "data captured". The falling transition triggered by the computing module strobe circuit on the control bus is used to indicate the pre-charging or "ready" condition of the computing module, as shown in Fig 1(b) [5]. Because the "request" and "acknowledge" signals can be directly used to control input latching, the four-phase protocol is easier to implement in a GALS (global asynchronous and local synchronous) system or a modular synchronous system. The Muller C-element is commonly used to achieve logical behavior among successive signals regardless of pipeline protocol. The C-element transits only when all inputs reach a matched logic level, hence functions as a great control mechanism to prevent free-run problems. But the C-element also has inherently larger propagation delays when used in stacked C-elements for MIMO (multi-input or multi-output) modules, especially in low voltage systems. Thus, reducing the cost of signal transitions on stacked C-elements is the key to getting better performance in such MIMO asynchronous implementation.

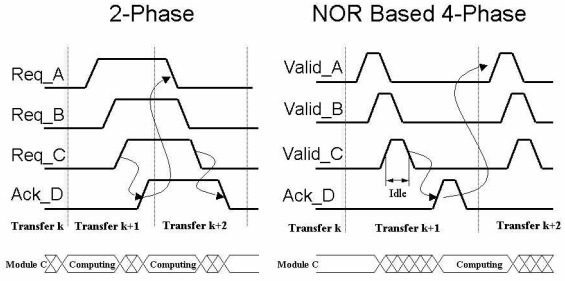


Fig. 1. Pipelined protocol for 3 inputs: (a) Two-phase pipelined protocol, and (b) NOR-based four-phase pipelined protocol.

Paper [5] presented another choice called the interlocked pipelined CMOS (IPCMOS) and shown in Fig. 2(b) and Fig. 6(a). The IPCMOS combines the function of static NOR gate and input switches, and frees asynchronous systems from degradation by stacked C-elements as the number of input signals increases. By contrast, the cost of such a static NOR-based strobe circuit is more power dissipation by self-locked inverter pairs of strobe and reset switches. The inverter pairs in strobe switches are used as previous-

status keepers, and are also the keys that reset the strobe circuits for the next control cycle. Three stacked inverters and feedback paths in the strobe circuit are used to generate wide enough clock pulses for transparent latching, and may also reset the local clock after the last input signal arrives. Furthermore, it is assumed for proper functioning of such pseudo-NOR gates used in these strobe circuits that since cascaded P-MOSFETs are not allowable for low voltage systems, the arriving valid signal appearing only on the P-MOSFET must be the last one.

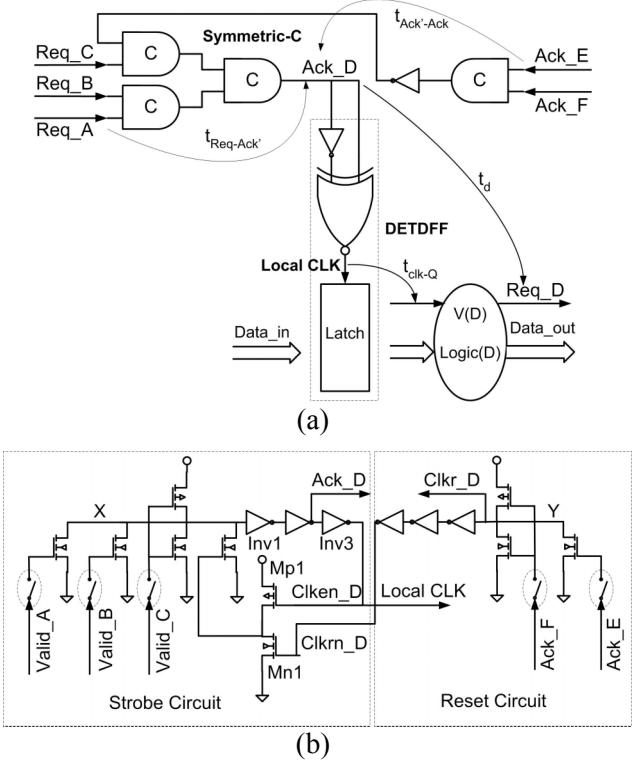


Fig. 2. Pipelined systems for multi-port asynchronous modules: (a) two-phase pipelined system, and (b) NOR-based four-phase pipelined system—interlocked pipelined CMOS [5].

Since the generated clock pulses also perform the logical function of resetting switches used in the IPCMOS control system, C-elements are intrinsically better for constructing such reciprocal control logic. In this paper, we also illustrate a new IPCMOS control system in which simple C-elements take the place of strobe/reset switches. The HSPICE simulation results show such C-element-based IPCMOS, free of self-locked inverter pairs, not only have better faster responses, but also have better power saving.

2 Modified Symmetric C-element used in Two-Phase Pipeline System

The Muller C-element was first introduced in 1959 [8], and is in general use in asynchronous control protocol schemes. Most developed C-elements have a feedback circuit called a keeper to lock the output stage until all inputs reach the new match, and resetting this keeper circuit makes all conventional C-elements, except the symmetric C-element [4, and 9], slow and wasteful. The symmetric C-element shown in Fig. 3(a) uses two sets of symmetrically arranged switches on the input connection that work as a pair of split inverters when a matching signal arrives. However the keeper circuit, Mp6 and Mn6, shown in Fig. 3(a) is controlled by the output state and provides an alternative signal path between the switches when an unmatched signal arrives. Since the symmetric C-element keeper does not resist changes in output state, its transmission delay is nearly equal to the delay of two stacked inverters. Generally, only 2~3 input-ports can be integrated into one stage of its switch circuit when low-voltage operation is considered. And that implies the performance degradation induced by C-element becomes worse when multi-port asynchronous modules with stacked C-elements like those shown in Fig. 2(a) are used. Since the output inverter used in the symmetric C-element only provides two basic functions, which are output buffer for the next inputs and feedback state for the internal keepers. As consider using of the stacked C-elements shown in Fig. 4(a), it is not necessary to insert output buffers between switch circuits. Given the sufficient driving capacity of the switches in the symmetric C-element, we propose skipping the internal stacked C-element inverter-buffers in favor of the odd/even C-element scheme shown in Fig. 3(b). The only differences are that the keeper circuit gets the feedback state from the last output buffer, and signal flow passes through fewer inverters. Therefore, the proposed two-phase pipelined control circuit for 3 input-ports and 2 output-ports is shown in Fig. 4(b).

The better performance achieved by the two-phase control system is due to the dual data captures per cycle in the control signal transition. The direct practical application uses a newly defined interface that makes the conventional transparent latch active on the rising and falling edges of the control signal. The advanced processor technology (APT) group has proposed a two-to-four-phase interface to make such a level sensitive latch work in a two-phase control scheme [1]. Another direct approach [6] to the two-phase control system is to design a new truly double edge-triggered D-flip-flop (DETDFF) that allows the signal latch used in the two-phase micro-pipeline scheme to be active during the rising and falling transitions on the control bus. Most DETDFFs are constructed using pairs of complementary edge-

triggered flip-flops, and such solution certainly implies double power dissipation.

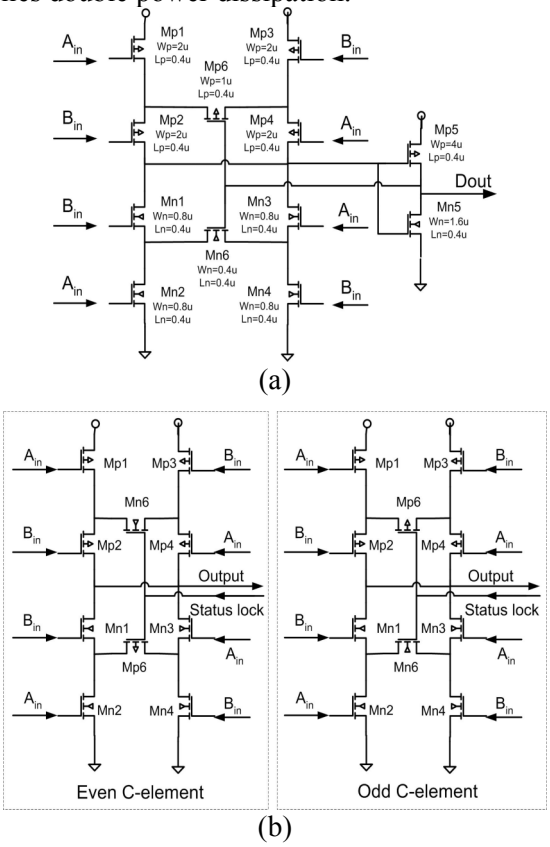


Fig. 3. C-element implementations: (a) symmetric C-element [9], and (b) odd/even C-element.

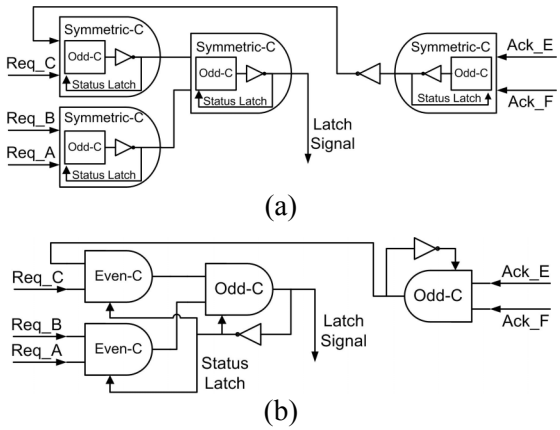


Fig. 4. Two-phase pipelined control system: (a) (a) symmetric C-element [9], and (b) odd/even C-element.

We chose a pulse generator to trigger the transparent latch twice per control signal cycle. The pulse generator contains only two common logic gates, the XNOR and the inverter, shown in Fig. 5. The inverter provides the delayed complementary control signal, and the XNOR logic provides pulse signals on the phase difference between the original signal and the shifted signal, theoretically, the rising and the falling control signal transitions.

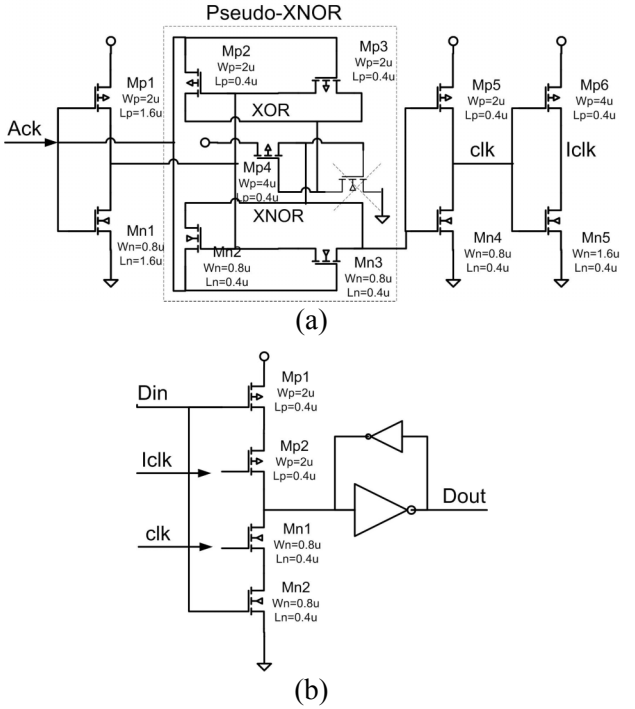


Fig. 5. Double edge-triggered flip-flops constructed with XNOR and transparent latch: (a) clock-pulse generator constructed with XNOR, and (b) transparent latch.

The transmission paths or so-called "capture-pass logics" perform the logical XNOR used in our design [12]. We add an input inverter with a longer channel to allow sufficient delay time to generate wide enough pulse during the control signal transitions. Actually, how small the input inverter width-to-length ratio is determines whether our pseudo-DETDFF functions normally and how low the supply voltage can be. Simulation results clearly show significant deterioration in this pseudo-DEFDFF as supply voltage drops down to 1.8 volts. This is because of the threshold voltage in the dedicated CMOS process, and there being not enough voltage drop on the transmission path. However, we shall emphasize that this slower inverter does not affect the entire response, but adjusts the pulse width.

The two-phase pipeline protocol scheme used was modified from Sutherland's micro-pipeline, which is discussed in [6, 10], and shown in Fig. 2(a). Obviously, there may be a race problem or synchronization failure in such a simple two-phase control system. If the delay time from the computing cell strobe circuit, V(D) in Fig. 2(a), is not sufficient to allow the DETDFF to capture correct data before enabling the next stage, the right stage may latch an uncertain logic state, even the previous logic state.

Thus, the delay buffer timing constraint must fit the equations given in [6]. Its simplified form is written as

$$t_d \geq t_{clk \to Q} + t_{\log ic} + t_{Ack'-Ack} - t_{\mathrm{Re}q \to Ack'} \quad (1),$$

where t_{logic} is the logic cell computing time.

3 Modified NOR-based Four-Phase Pipeline Protocol

3.1 NOR mechanism used in the four-phase protocol for multi-port modules

As discussed in the preceding section, using the conventional Muller C-element in control circuits with multiple inputs or multiple outputs will degrade overall system performance. Paper [5] proposed using a static NOR gate with state-self-locked switches (interlocked pipelined CMOS) to approximate the four-phase protocol function, as shown in Fig 1(b) and Fig. 2(b). The rising transition of a valid signal enables the strobe switch to the ready state, and the falling transition pulls the strobe switch output low. If no rising transition appears on a valid input, the strobe switch output state will be locked low until a rising acknowledgement signal transition (Ack_D) appears. The feedback path from the local clock output (Clken_D) to Mp1 is used to reset this strobe circuit for the next control cycle. Of course the three inverters (Inv1~3) used for the output buffer must be fast for better response performance. Thus, the feedback response from Clken_D, though Mp1, to node "X" must be slower to allow a wide enough clock pulse and reset the strobe switch. Since only the signal level performs the various operating functions in such a protocol, and the valid signal pulse width must be treated as system idle time, this IPCMOS is still a sort of four-phase protocol. And the unavoidable overhead cost is the width of the last valid signal, which is generally propagated from the local clock cycle of the previous stage through the logic cell strobe path. Due to the low-voltage operation, only one P-MOS is used in such a static NOR gate, and that causes the disappointing assumption of Valid_C being the last arriving signal. Furthermore, there are two self-locked inverter pairs used in each strobe switch, and that also implies more power dissipation.

The reset circuit used in [6] is similar to the strobe circuit, but simpler. Because such a simple input switch used, the reset switch output cannot be estimated as Ack_E and Clkrn_D are initialized to low and high, respectively. And that means the logic state of this reset output may be neither logical "1" nor logical "0" after the initial setting, unless the only P-MOSFET used in the circuit dominates the "Y" node logic state.

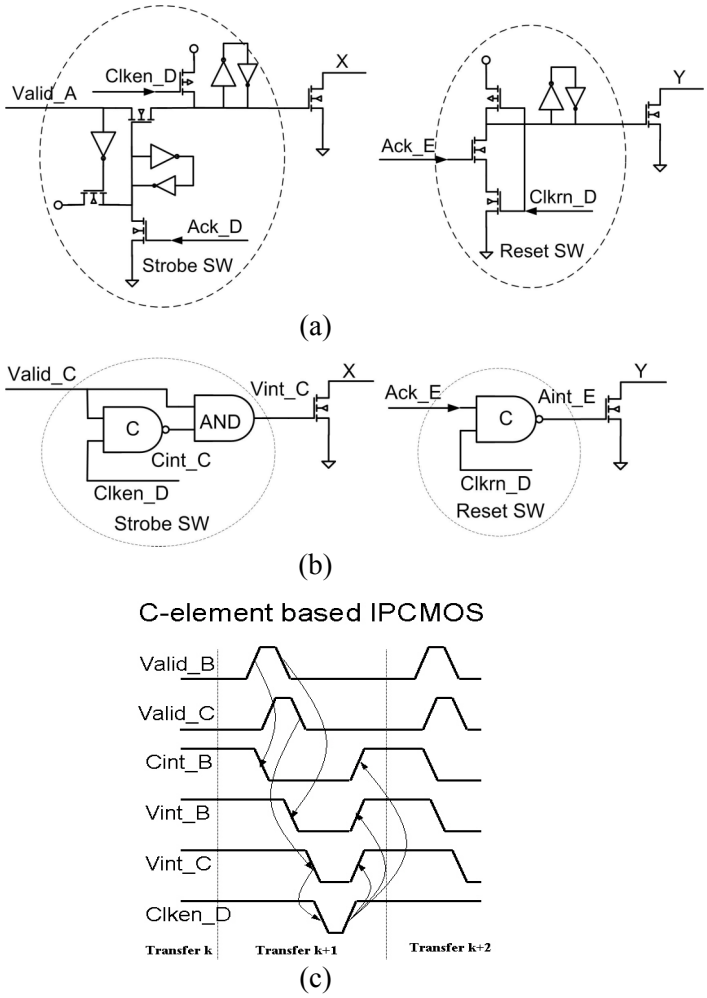


Fig. 6. Input switches used in IPCMOS protocol: (a) strobe and reset switches used in IPCMOS [5], (b) C-element-based strobe and reset switches, and (c) the wave-flow of (b).

3.2 Use of Symmetric C-element to Implement Input Switches

Since the strobe and reset switch output signals are also used to reset those switches, it is theoretically possible to replace the switch circuits with conventional C-elements, as shown in Fig. 6(b). The reasons for introducing a C-element solution into the IPCMOS control protocol are that the C-element mechanism is as robust as a strobe switch for successive inputs, and a symmetric C-element is more power-saving due to its freedom from the interlock drawback. Furthermore, there is no possibility of synchronization failure as with the original reset switch. As shown in Fig. 6(c), the output of the inverse-C-element (Cint_B) is pulled to logic low as

the high level of a valid signal arrives. Since there is a little propagation delay between the valid transition and the inverse-C-element transition, the output of the OR gate (Vint_B) maintains the logic high until the valid signal transits to logic low. Since all OR gate outputs go logic low, the strobe circuit output (Clken_D) will be triggered to low. After passing through the strobe circuit Mp1, the strobe output is pulled to logic high by its previous logic low. In the meantime, the local clock logic-low signal (Clken_D) resets the outputs of all inverse-C-elements and OR gates to logic high to wait for next signal cycle.

4 Simulation Results

We performed HSPICE simulation on the TSMC 0.25um fast-mode model CMOS. All the N-MOSFETs used in the simulation were of the same transistor size, except for weak inverter lengths and output buffer widths. We adjusted the PMOS width-to-length ratio to be two-and-one-half times than that of the NMOS, based on simulation results for simple inverter rising and falling times. To simplify the simulation, we used a series of delay buffers to mimic the internal signal path from the local clock to the next-stage acknowledge, as shown in Fig. 7.

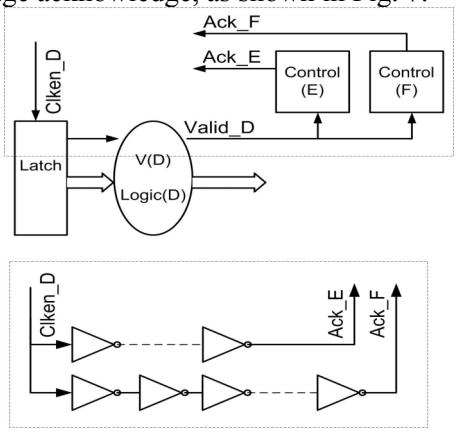


Figure 7. Simulated internal control-signal path replaced by series inverters.

Fig. 8 shows simulation results for propagation delay from last valid signal to active clock, plotted against number of inputs. Since the strobe circuit transits after the falling transition of the last valid signal, the propagation delay here is the time delay from the falling transition of the last valid signal to the falling transition of the local clock. The two-phase pipeline is assessed from the request transition, either rising or falling, to the clock pulse rising transition. It is clear that the best solution is the C-element-based IPCMOS, not only was it the fastest, it also had the least incremental delay for added inputs. The odd/even C-element solution was able to maintain a smaller delay than that of the IPCMOS up to eight inputs. We must emphasize that like IPCMOS idle time, the theoretical overhead cost of waiting for control bus signal transitions could be zero if the computing time for the last module, t_{logic} in equation 1, could be precisely estimated, although it is almost impossible to estimate computing times under varying operating conditions.

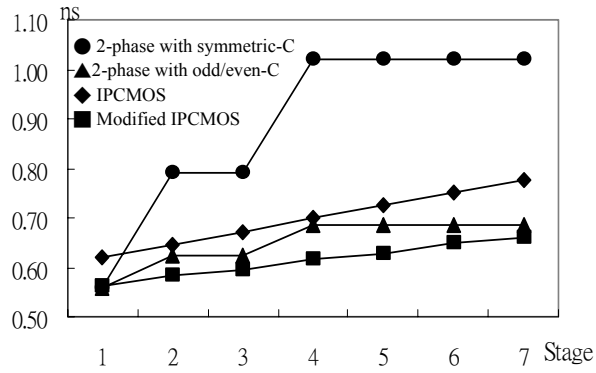


Fig. 8. Propagation delay from the arrival of the last valid-signal to the clock enable.

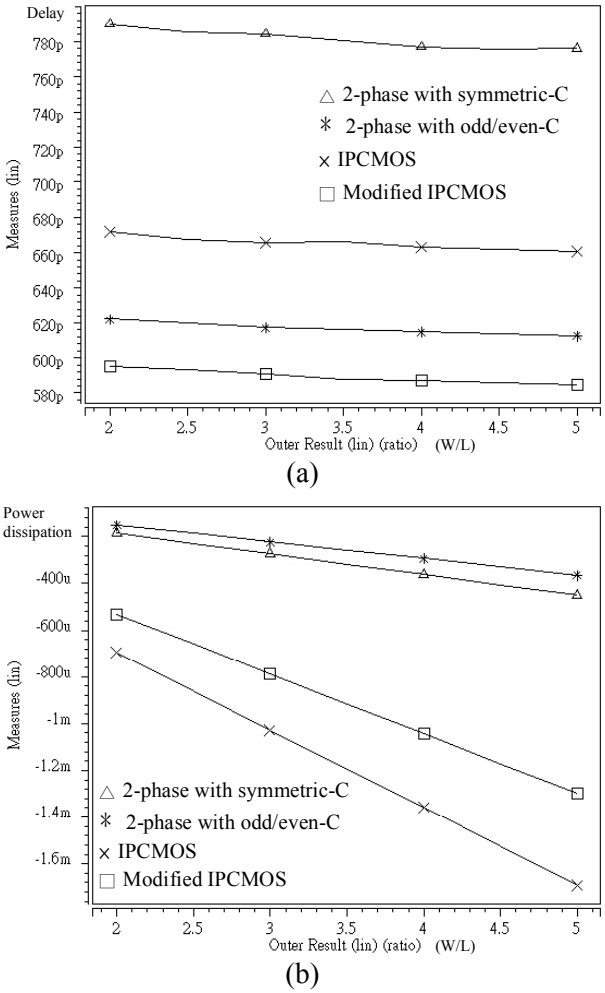
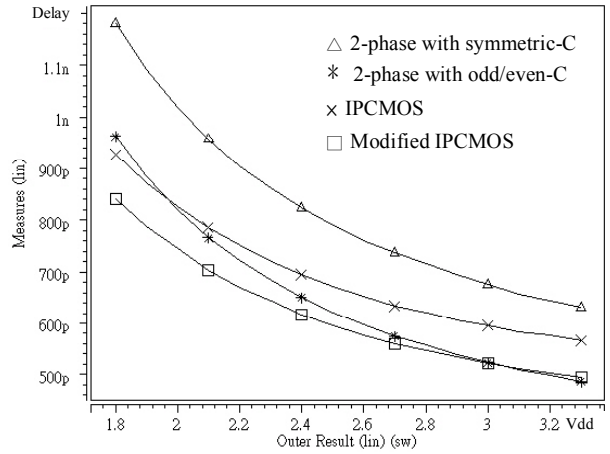
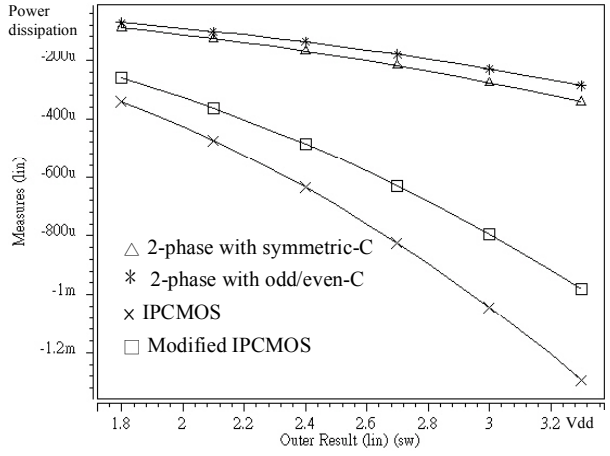


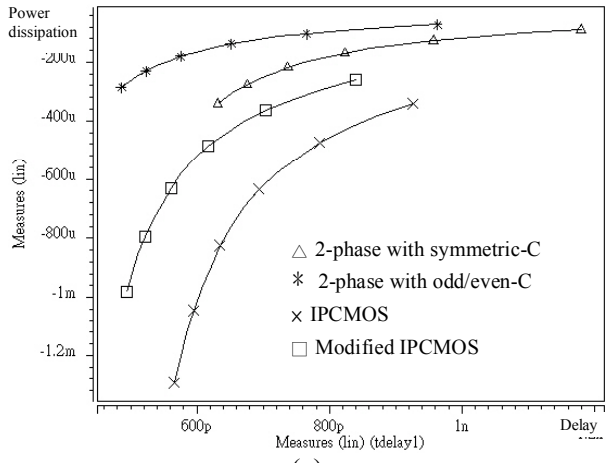
Figure 9. Simulation results for various W/L ratios: (a) propagation delay, and (b) power dissipation.



(a)



(b)



(c)

Figure 10. Simulation results for various supply voltages: (a) propagation delay, (b) power dissipation, and (c) the product of propagation delay and power dissipation.

Fig. 9 shows power dissipation and propagation delay for four types of pipelined systems as the ratio of width-to-length was increased. The conventional two-phase pipelined system had the better performance in power reduction, 73%, but had 18% delay expansion compared with the conventional IPCMOS. Furthermore, the two-phase pipelined system modified with odd/even-C-element had the best performance in power saving, and it was 78% reduction compared to the conventional IPCMOS. It seems that the two-phase pipelined protocol had lower power dissipation at the expense of slower response. But as mentioned above, this is based on disregarding IPCMOS idle time. Of course, we must note that the IPCMOS modified with a symmetric C-element had better speed and power dissipation performance, 11% reduction in delay and 23% reduction in power, compared to the conventional IPCMOS.

Power dissipation and propagation delay for four types of pipelined systems as supply voltage was varied is shown Fig. 10. As discussed in Section II, the performance deterioration of the pseudo-DETDFF as supply voltage was reduced to 1.8V was due to the capture-pass scheme used in the XNOR gate. Fig. 10(c) specifically depicts the trend of the product of propagation delay and power dissipation, and clearly shows that the odd/even-based two-phase pipelined system still had the best performance when there were only three inputs. Even the conventional C-element had better performance than the conventional and the modified IPCMOS under those conditions. And there was no limitation on the last arriving port with the two-phase system. The modified IPCMOS had a 23% reduction in power and an 11% reduction in delay compared to the conventional IPCMOS at all tested operating voltages.

5 Conclusion

The concept of pipelined asynchronous modules reveals a new possibility of having greater throughput than traditional implementations. Generally, the two-phase pipelined control system achieved better performance by removing the unnecessary overhead of waiting for signal transitions on the control bus, but the stacked C-element used in multi-port asynchronous modules made system performance deteriorated due to the propagation delay on such stacked C-element. The IPCMOS provides a static NOR-gate solution for multi-port asynchronous modules, but has the drawback of assessing the last arriving control signal. In this paper, we presented two modified pipelined systems with better speed and power consumption performance. Simulations were performed using the TSMC 0.25um fast-mode CMOS model, and the results show the conventional IPCMOS consumed at least 27% more power than the other solutions, although the IPCMOS may be a good solution for multi-port modules. The modified IPCMOS with symmetric C-element-based switches

had the best speed response and better power dissipation. On the other hand, the conventional two-phase pipelined system achieved reliable performance when there were only three inputs connected. Although simulation results show there was an 18% delay expansion compared to the conventional IPCMOS solution; the IPCMOS idle time was not assessed for this result. The modified two-phase pipelined system with odd/even C-element had smallest power dissipation compared to the other solutions. Obviously, those two modified systems are better than conventional systems. And the C-element may still be a good choice for asynchronous control.

References:

[1] Paul Day and J. Viv Woods, "Investigation into Micropipeline latch design styles," *IEEE Transactions on VLSI Systems*, V3 (2): 264-272, June 1995.

[2] Allen E. Sjogren and Chris J. Myers, "Interfacing synchronous and asynchronous modules within a high-speed pipeline," *IEEE Transactions on VLSI Systems*, Vol. 8, No. 5, pp.573-583, October 2000.

[3] J. C. Ebergen, J. Segers and I. Benko, "Parallel progress and asynchronous circuit design," *in Asynchronous Digital Circuit Design*, New York: Springer-Verlag, 1995, pp. 51-103.

[4] Marc Renaudin, Bachar El Hassan, and Alain Guyot, "A new Asynchronous Pipeline Scheme: Application to the Design of Self-Timed Ring Divider," *IEEE Journal of Solid-State Circuits*, Vol. 31, No. 7, pp.1001~1013, July 1996.

[5] Stanley E. Schuster and Peter W. Cook, "Low-power synchronous-to-asynchronous-to-synchronous interlocked pipelined CMOS circuits operating at 3.3-4.5 GHz," *IEEE Journal of solid-state circuits*, vol. 38, No. 4, pp.622-630, April 2003.

[6] K. Y. Yun, P. A. Beerel and J. Arceo, "High-performance two-phase micropipeline building blocks: double edge-triggered latches and burst-mode select and toggle circuits," *IEE Proceeding, Circuits, Devices and Systems*, 143(5): 282-288, October 1996.

[7] J. Carlsson, W. Li, T. Njolstad, K. Palmkvist, L. Wanhammar, and S. Zhuang, "A Modular Asynchronous Wrapper", *National Conf. Radio Science (RVK)*, Stockholm, Sweden, Jane, 10-13, 2002.

[8] D. E. Muller and W. S. Bartky, "A theory of asynchronous circuits", in *proceedings of an International Symposium on the Theory of Switching,* Part I, pages 204--243, 1959.

[9] K. v. Berkel, "Beware the isochronic fork," *Integration, The VLSI J.* vol. 13, pp. 103-128, June 1992.

[10] Ivan E. Sutherland, "Micropipelines," *Communications of ACM*, 32(6): 720-738, June 1989.

[11] Maitham Shames, Jo C. Ebergen, and Mohamed I. Elmasry, "Modeling and Comparing CMOS Implementations of the C-Element", *IEEE Trans. on VLSI System*, Vol. 6, No. 4, pp. 563-567, Dec. 1998.

[12] J. M. Wang, S. C. Fang and W. S. Feng, "New efficient designs for XOR and XNOR functions on the transistor level," *IEEE Journal of Solid-State Circuits*, vol. 29, pp. 780-786, July 1994