

Selectively Clocked Skewed Logic (SCSL): A Robust Low-Power Logic Style for High-Performance Applications

Naran Sirisantana, Aiqun Cao, Shawn Davidson, Cheng-Kok Koh, and Kaushik Roy

School of Electrical and Computer Engineering,
Purdue University, West Lafayette, IN.

{sirisant, caoa, sdavidso, chengkok, kaushik}@ecn.purdue.edu

ABSTRACT

In very high performance designs, dynamic circuits, such as Domino Logic, are used because of their high speed. Skewed logic circuits can be used to achieve designs having performance comparable to that of Domino but with better scalability. Moreover, a selective clocking scheme may be applied to enhance the power savings for skewed logic circuits. This paper proposes Selectively Clocked Skewed Logic (SCSL), a new circuit style based on skewed logic aiming for low clock power consumption. The results on ISCAS benchmark circuits implemented with this circuit design style show that the total power consumption can be reduced by (52.05)% when compared to that of Domino circuit with comparable performance.

Categories and Subject Descriptors

1.3 [Logic and Microarchitecture Design]: Logic and RTL design.

1. INTRODUCTION

With the demand for high performance systems, dynamic circuits, such as Domino logic, have been used in critical paths of the design to achieve the desired performance. With the continued scaling of supply voltages, threshold voltages, and transistor sizes, it becomes more difficult to scale Domino circuits because of the dependence of noise margin on the transistor threshold voltage [5]. Another problem with Domino is the clock load: the clock has to be connected to every gate in the circuit and this consumes a considerably large amount of power. As every gate is being precharged at the same time, this also impacts the peak current, peak power, and power supply noise of the circuit.

The abovementioned problems can be mitigated by using skewed logic [3][5]. Since a skewed logic gate has the same circuit topology as a standard static CMOS gate, the noise immunity is better than that of the corresponding Domino gate.

By operating skewed logic circuits in two phases — evaluation and precharge — as is done in Domino circuits, we can achieve performance comparable to that of dynamic circuits. We use the fast transition in the evaluation phase and the slow transition in

the precharge phase. Due to the nature of skewed logic circuits, we can selectively connect the logic gate to the clock, resulting in a lower clock load and lower clock power consumption when compared to Domino circuits.

The rest of the paper is organized as follows. The operation of skewed logic gate is described in section 2. Synthesis for dual-phase circuitry is presented in section 3. Section 4 shows a selective clocking scheme for SCSL circuits. The remainder of the paper discusses the results from simulations based on the algorithm and compares them with that of Domino circuits.

2. SKEWED LOGIC

A skewed logic gate has the same circuit topology as a classical static CMOS gate but the size of either the PUN or the PDN is increased for fast low-to-high or high-to-low transitions, respectively. For example, in a 2-input NAND gate, the size of the NMOS transistors in the PDN is increased for fast high-to-low transition. Similarly, the width of the PMOS transistors in the PUN of a 2-input NOR gate is increased for fast low-to-high transition. Skewing is defined as changing the ratio between the PMOS and NMOS devices of the gate, $R_p = W_p/W_n = 1/R_m$, from its original value. Figure 1 illustrates the structures of both the 2-input NAND and NOR skewed logic gates.

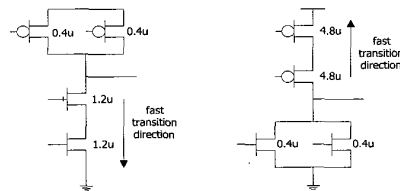


Figure 1. 2-input NAND and NOR skewed logic gates

The method for changing the skew value of a gate in [3] is applied. With this method, we can change the skew value of the gate *without* changing the overall transistor width and gate capacitance.

In order to achieve performance comparable to Domino circuits, the skewed logic circuit operates in two phases: precharge and evaluation. During precharge phase, all gates are reset to their initial state through their respective slow transitions. While in the evaluation phase, the circuit performs its associated function. To ensure the highest performance, only fast transitions are allowed during the evaluation phase. This can be done by arranging the skew directions of the gates in a chain such that the gate skewed for fast high-to-low transition is followed by gates skewed for fast low-to-high transition, and vice versa, as illustrated in Figure 2.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ISLPED '01, August 6-7, 2001, Huntington Beach, California, USA.
Copyright 2001 ACM 1-58113-371-5/01/0008...\$5.00.

With this arrangement, the precharging of gates in the chain can also be propagated to the subsequent gates as long as it does not exceed the precharge phase of the clock period. As a result, not every gate in the circuit has to be connected to the clock and power consumption can be reduced. Figures 3a) and b) show the structures of a clocked NAND gate skewed for fast high-to-low transition for fully clocked skewed logic circuits (FCSL) and SCSL, respectively. A footer transistor is not required in FCSL circuits since the inputs of the gate are guaranteed to be precharged to zero. However, it is required for SCSL to prevent short circuit current.

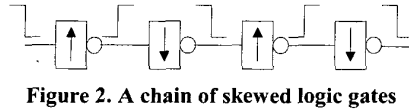
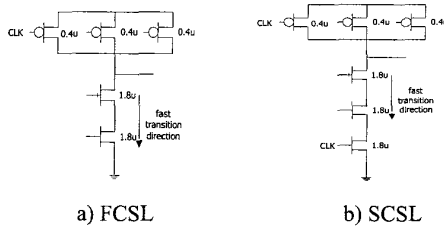


Figure 2. A chain of skewed logic gates



a) FCSL b) SCSL
Figure 3. 2-input NAND skewed logic gate with clock

Pipelining with skewed logic circuits can be implemented by following the same technique as in pipelining with Domino circuits as shown in Figure 4. In the first half of the cycle, when the clock is high, logic block A evaluates the circuit's function while latch A holds the input data for it. Logic block B precharges while latch B is transparent. In the second half of the cycle, the operations are reversed with logic block A precharging and logic block B evaluating.

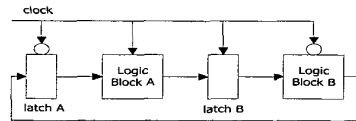


Figure 4. Basic pipeline structure

As mentioned earlier, the evaluation time of each logic block must not exceed the evaluation phase of the clock. This is also true for the precharge phase of the design. Figure 5 shows a logic block using skewed logic circuits with arrows indicating the direction of the fast transition. In this example, we assume that the fast transition (evaluation) delay time for each gate is t_{ev} and the slow transition (precharge) delay time is $3t_{ev}$. We also assume that the delay for precharging the gate connected to clock is $t_{cl} < 3t_{ev}$. The signal waveforms at each point of the circuit can be illustrated as in Figure 6. We can see that gates 1, 4, and 7 start precharging as soon as the falling edge of the clock signal. The precharging is then propagated to gates 2,5,8 and 3,6,9, respectively, provided the total precharge delay time of the gates between each clock position does not exceed the precharge phase of the clock cycle. With this concept in mind, we propose a selective clocking scheme to place the clock properly such that the power consumption is reduced.

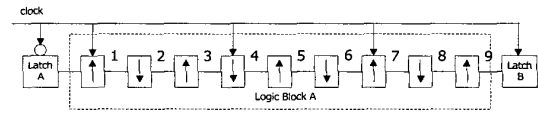


Figure 5. Logic block using skewed logic circuits

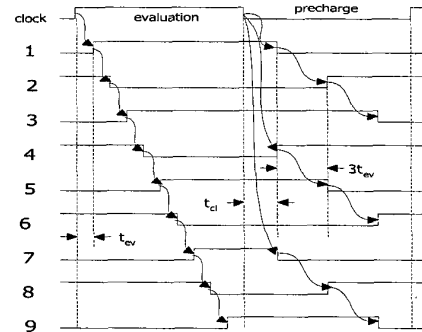


Figure 6. Signal waveforms at each point of the circuit in Figure 4

3. SYNTHESIS FOR DUAL-PHASE CIRCUITS

Although skewed logic circuits are essentially static circuit, the synthesis of skewed logic circuits requires special consideration. As mentioned in section 2, each gate has to be assigned a skew direction, either fast high-to-low or fast low-to-high transition, in order to meet the dual-phase gate cascading requirements. That poses a problem when we assign the skew directions to gates in reconvergent paths.

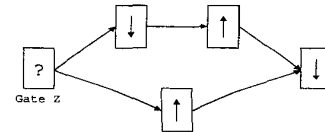


Figure 7. Logic reconvergence in skewed logic circuit

As Figure 7 illustrates, the skew direction of Gate Z, which drives two gates that have opposite skew directions, cannot be determined. This is similar to the reconvergent path problems encountered in Domino circuits.

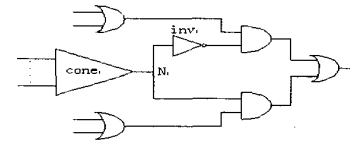


Figure 8. Reconvergent path in Domino circuit synthesis

Domino circuits contain only non-inverting gates that can make only a '0' to '1' transition during the evaluation phase. Thus, the synthesis of Domino circuit involves pushing inverters from the primary outputs to the primary inputs to guarantee this requirement. With reconvergent path, inverter may be trapped, as shown in Figure 8. Logic duplication [4] is required to remove the trapped inverters. As in Figure 8, $cone_i$ must be duplicated so that both positive and negative phases are implemented and the trapped inverter inv_i can be pushed back until reaching the

primary inputs. This logic duplication will at most double the original logic [9].

Obviously, logic duplication can also be applied to solve the reconvergence problem in skewed logic circuits. Instead, we use a circuit technique using pass-transistor logic to solve this problem. The technique has minimal impact on area. For example, we replace the NAND gate in Figure 9(a) with pass-transistor logic in Figure 9(b). One inverter is added after signal B to implement the function of a NAND gate by using pass-transistors. Two additional transistors are incorporated and connected to the clock signal. This enables the pass transistor to be incorporated into a design that utilizes precharge and evaluation phases.

In comparison with logic duplication, the pass-transistor logic technique can solve the reconvergence problem in place, as opposed to propagating the change back to the primary inputs of the circuit. Our experiments show that only 10% extra logic is required for the ISCAS benchmark circuits for reconvergent paths. This is far superior compared to logic duplication, which requires logic duplication for 80% of the gates. We can apply this technique for synthesizing circuits, such as SCSL or *np*-CMOS.

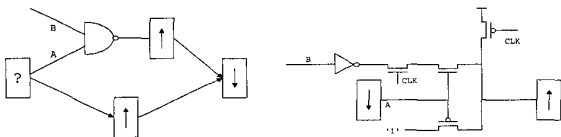


Figure 9(a). Reconvergence Figure 9(b). Pass transistor

Unfortunately, this circuit technique cannot be applied to Domino circuits. This is due to the fact that if pass-transistor logic were used, the output of this pass-transistor gate could not be guaranteed to make only a '0' to '1' transition during the evaluation phase.

4. A SELECTIVE CLOCKING SCHEME

The concept introduced in section 2 is used to derive a selective clocking scheme for SCSL circuits.

Assume that a set of gates with different skew values is given. A gate with a larger skew value has smaller evaluation delay time (t_{ev}), whereas a gate with a smaller skew value has smaller precharge delay time (t_{pc}). In order to achieve the highest performance of the circuit, gates with largest skew value are used on the critical evaluation paths.

For a 50% duty-cycle clock period, the required evaluation period is equal to the critical evaluation delay, which is also equal to the critical delay of the precharge period. In order to have the least number of gates connected to the clock, we try to maximize the number of gates in each precharge period without exceeding the critical precharge delay. With smaller skew values, a gate has a smaller precharge delay time. Therefore, more gates (without clock) can be placed in the precharge chain, which results in a smaller clock load.

One problem in assigning a smaller skew value to a logic gate is that the evaluation delay time of the gate increases. Hence, it is necessary to check that the new evaluation delay time does not exceed the critical evaluation delay time; otherwise, the critical evaluation delay may change. The algorithm for determining the placement of the clock gates can be divided into 2 steps: Initialization and Skew value and clock assignment.

4.1 Initialization

The first step of the algorithm is to initialize all the gates in a circuit to the largest skew value of the set. During the initialization step, the circuit is leveled and the static timing analysis is done by forward tracing the circuit level by level. The propagation delay, arrival and departure times and slack of each gate are then calculated. The critical delay and critical paths are determined by tracing back from the circuit's primary outputs.

4.2 Skew value and clock assignment

The next step of the algorithm is to determine an appropriate skew and to decide whether the gate should be connected to the clock under the given delay constraint found in the Initialization step. The circuit is backtraced level by level in order to determine the optimal skew values for each gate.

The skew is determined according to the available slack. The slack of a gate ($T_s(x)$) is defined as the amount of time by which the gate can be slowed down without affecting the circuit performance. There are two kinds of slack for SCSL circuits: evaluation slack for evaluation period, and precharge slack for precharge period. We wish to have as many gates with the smallest skew possible in a precharge chain. Therefore, the smallest skew is assigned to the gate initially and then the static timing components of the gate are recalculated. If it does not satisfy the evaluation slack, the process is repeated with the next larger skew until we find the smallest skew that satisfies the evaluation slack of the circuit.

The next step involves the determination of whether the current gate should be connected to the clock. After adding the gate to the precharge chain, the precharge slack is evaluated to determine if there is sufficient slack to accommodate the largest precharge delay time with clock among all fan-in gates. If the slack is sufficient, then the gate does not have to be connected to the clock. If the gate has to be connected to the clock, all its fan-in gates will see the new precharge slack of the whole precharge period. The new static timing components with a new condition of the gate are assigned to the gate. Since the slack of the gates on critical paths is 0, the skew value of these transistors must not be changed to assure the highest performance of the circuit.

5. RESULTS

We synthesize the ISCAS benchmark circuits with SCSL under the Berkeley SIS environment. The library of gates is limited to contain only INVERTER, 2 to 4 -input NAND and 2 to 4 -input NOR gates with skew values of 3 to 5 for simplicity. For each SCSL circuit, we compare its total power consumption with that of Domino circuits of 2-input AND and up to 6-input OR gates running at the same clock frequency. The power dissipation savings are obtained from PowerMill's GAP simulations with searching level 5, with a supply voltage of 3.3V. A 0.35 μ m CMOS technology is used and the unskewed effective channel widths for PMOS and NMOS transistors are 6.3 μ m and 1.8 μ m, respectively, for all experiments.

Table 1 summarizes the results of applying the proposed clocking scheme to ISCAS benchmark circuits and comparing the results with Domino circuit implementations operating at the same clock frequency. The percentage of gates connected to clock for SCSL circuits is approximately 26.6% of the total number of gates. This results in a significant savings in clock power. Also, synthesizing

circuits with pass-transistors favors SCSL circuits over Domino circuits with gate duplication by approximately a factor of two in terms of number of gates. This gives us advantages in both circuit power and clock power consumption. The total power of the circuits implemented with SCSL can be improved by approximately (52.05)% compared to Domino circuits. Figure 10 shows the total power consumption comparison between SCSL and Domino for the ISCAS circuits.

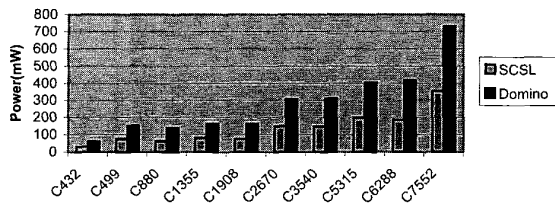


Figure 10. Total power comparison between SCSL circuits and Domino circuits (PowerMill results)

6. CONCLUSIONS

A new circuit style based on skewed logic, Selectively Clocked Skewed Logic (SCSL), is presented in this paper. SCSL circuits can be used in high performance applications instead of Domino circuits for better scalability and power consumption while maintaining high performance. Results from simulations indicate that SCSL circuits, coupled with a pass-transistor technique, reduce the total power consumption of the ISCAS benchmark circuits by (52.05)% when compared to that of Domino circuits of comparable performance, respectively. Optimizing power with high noise immunity while maintaining high performance makes SCSL very promising for low power high performance circuit design.

7. REFERENCES

- [1] J. D. Meindl, *Low power Microelectronics: Retrospect and Prospect*, Proceedings of the IEEE, v 83, n 4, pp. 619, 1995.
- [2] A. P. Chandrakasan, S. Sheng and R. W. Brodersen, *Low-Power CMOS Digital Design*, IEEE Journal of Solid-State Circuits, v 27, n 4, pp. 473, 1992
- [3] D. Somasekhar, *Power and dynamic noise considerations in high performance CMOS VLSI design*, Ph.D. Thesis, Purdue University, August 1999.
- [4] M. Zhao and S. S. Sapatnekar, *Technology Mapping for Domino Logic*, Proceedings of the IEEE International Conference on Computer-Aided Design, pp. 248--251, 1998
- [5] A. Solomatnikov, D. Somasekhar, K. Roy, C. K. Koh, *Skewed CMOS: Noise-Immune High-Performance Low-Power Static Circuit Family*, International Conference on Computer Design 2000, 2000.
- [6] J. M. Rabaey, *Digital Integrated Circuits: A Design Perspective*, Prentice-Hall, Inc., 1996.
- [7] S. M. Sze, *Physics of Semiconductor Devices*, John Wiley & Sons, Inc., 1981.
- [8] E. M. Sentovich, et al., *SIS: A System Sequential Circuit Synthesis*, Electronics Research Laboratory Memorandum No. UCB/ERL M92141, University of California, Berkeley, 1992.
- [9] S. M. Reddy, *Complete test sets for logic functions*, IEEE Transaction on Computers, C-22(11): 1016-1020, November 1973.
- [10] Y. Taur and T.H. Ning, *Fundamentals of Modern VLSI Devices*, Cambridge University Press, 1998.

Table 1. Total power comparison between SCSL circuits and Domino circuits

Circuit	Type	# gate	# transistor	Total width (μm)	# clocked gate	Circuit power (mW)	% reduction	Clock power (mW)	% reduction	Total power (mW)	% reduction
C432	SCSL	305	1 037	4 189	136	21.73	55.0	9.55	15.1	31.28	47.5
	Domino	510	3 096	9 245	510	48.28		11.25		59.53	
C499	SCSL	657	2 212	8 959	329	49.04	57.1	26.24	27.6	75.28	50.0
	Domino	1 097	6 846	20 221	1 097	114.21		36.23		150.44	
C880	SCSL	433	1 700	6 885	195	42.02	61.9	19.07	29.5	61.09	55.5
	Domino	695	4 183	12 533	695	110.24		27.06		137.30	
C1355	SCSL	650	2 455	9 940	290	54.25	56.6	24.78	30.3	79.03	50.8
	Domino	1 126	6 960	20 635	1 126	125.01		35.54		160.55	
C1908	SCSL	873	3 099	12 549	377	51.96	59.8	23.34	27.9	75.30	53.4
	Domino	1 416	8 934	26 276	1 416	129.16		32.37		161.53	
C2670	SCSL	1 145	4 532	18 354	527	101.36	58.1	45.57	28.2	146.93	51.9
	Domino	1 990	12 231	36 344	1 990	241.72		63.46		305.18	
C3540	SCSL	1 670	5 764	23 344	638	106.88	57.5	39.67	33.7	146.55	52.9
	Domino	2 787	17 903	52 293	2 787	251.36		59.86		311.22	
C5315	SCSL	2 519	9 948	40 290	1033	141.54	56.7	55.84	28.3	197.38	51.3
	Domino	4 114	25 926	76 287	4 114	327.21		77.88		405.09	
C6288	SCSL	3 146	10 099	40 860	1624	149.75	58.4	34.71	39.5	184.46	55.8
	Domino	5 159	132 963	386 603	5 159	360.19		57.36		417.55	
C7552	SCSL	3 858	13 498	54 670	1454	259.14		94.78		353.92	
	Domino	6 425	41 435	120 838	6 425	590.50	56.1	138.24	31.4	728.74	51.4