An Adaptive Supply-Voltage Scheme for Low Power Self-Timed CMOS Digital Design

W. Kuang and J.S. Yuan
University of Central Florida, Orlando, Florida, USA

Abstract

This paper combines an adaptive supply-voltage scheme with self-timed CMOS digital design, to achieve low power performance. The supply-voltage automatically tracks the input data rate of the data path so that the supply-voltage can be kept as small as possible while maintaining the speed requirement. This adaptive supply-voltage scheme employs the handshake signals directly to detect the speed of data path without using FIFO buffer. This leads to a very simple logic control whose power loss is negligible. Cadence SPICE simulation shows the effectiveness of this scheme for low power applications based on 0.18μm CMOS process.

1. Introduction

Lowering supply voltage is an aggressive approach to low power design. The dominant power dissipation [1] in digital CMOS circuits is the dynamic power dissipation

$$P_{dynamic} = \alpha \cdot f_{clk} \cdot C_L \cdot V_{DD}{}^2 \qquad (1)$$

where f_{clk} is the clock frequency, C_L is the total node capacitance in the circuit, α is the average fraction of the total node capacitance being switched, and finally V_{DD} is the supply voltage. Although reducing supply voltage V_{DD} leads to an increase in circuit delays [2], shown in Fig.1, the increased delays are allowed as long as the circuit still meets the speed requirements. Significant power and energy savings are possible if the supply voltage is to scale down to the smallest possible while maintaining the specific speed requirements.

A variable supply-voltage scheme has been developed [3] for synchronous system. Special attention should be paid on the increased delays due to the complicated clock distribution. This scheme includes a large logic control circuit, which consumes the area of a chip and significant power. So it is not efficient when applied to smaller circuits.

It is inherently convenient to apply a variable supply-voltage scheme to self-timed circuits. The maximum processing speed of the self-timed circuit is physically determined by the circuit hardware, not depending on any external synchronous clock. Traditionally, V_{DD} is constant, and the margin between the data rate of input from a synchronous system and the circuit maximum processing speed is needed to guarantee functionality. Therefore, there is a significant unused speed potential in the self-timed circuit. The idea of variable supply-voltage is to convert this speed potential into a corresponding power saving, by reducing the power supply until the delay of the data path just fits the available time slot. Especially, the handshake signals in speed-independent circuits can be used to detect if the circuit is so fast or so slow that the supply voltage should be reduced or increased for power saving and a specific speed requirement.

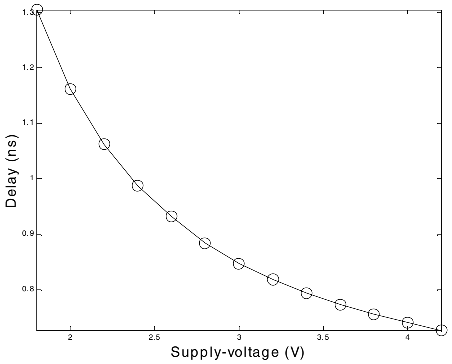

Fig.1 Delay vs. V_{DD} for a dual-rail full adder

A technique that combines self-timed circuitry with adaptive scaling of supply voltage was proposed by L.S.Nielsen et al [4]. However, this technique needs FIFO buffers, and the minimal sizes of the buffers depend on the data rate. The power dissipated by the buffers degrades the advantage of adaptive scaling.

This paper describes a novel adaptive supply-voltage scheme applied to speed-independent circuit. The speed-independent circuit uses dual-rail code for data path and a four-phase handshake protocol for communication control. The handshake signals can be directly used to detect circuit speed relative to the input data rate, and thus the logic control circuit for supply voltage is very

Proceedings of the 16th International Conference on VLSI Design (VLSI'03)
1063-9667/03 $17.00 © 2003 **IEEE**

simple and effective. Unlike the scheme in [4], the FIFO buffers can be eliminated.

In Section 2, the self-timed architecture is reviewed, and the analysis of timing is presented. In Section 3, an adaptive supply-voltage scheme is proposed. Cadence simulation results are given, and performance analysis is presented in Section 4. Section 5 is dedicated to conclusion.

2. Self-timed architecture and timing

In the data path of the self-timed circuit, each bit of the data is encoded by dual rails, shown in Table 1. The state DATA 0 (D0=1, D1=0) corresponds to a Boolean logic 0. The DATA 1 (D0=0, D1=1) corresponds to a Boolean logic 1. Spacer (D0=0, D1=0) corresponds to the empty set meaning that value of the bit is not yet available. The state (D0=1, D1=1) is forbidden.

Table 1 Dual-rail encoding scheme

Bit value	Rail logic value	
	D0	D1
DATA 1	0	1
DATA 0	1	0
Spacer	0	0
Invalid	1	1

In order to achieve speed-independence, the data path must work under Seitz's weak condition [5], shown in Fig. 2. The orderings labeled in Fig. 2 are explained as follows: (1) Some input becomes DATA before some output becomes DATA. (2) All inputs become DATA before all outputs become DATA. (3) All outputs become DATA before some input becomes Spacer. (4) Some input becomes Spacer before some output becomes Spacer. (5) All inputs become Spacer before all outputs become Spacer. (6) All outputs become Spacer before some inputs becomes DATA. The orderings (1), (2), (4), (5) are guaranteed by the proper design of the data path while ordering (3) and (6) are realized by the register control.

Some techniques, such as differential cascode voltage switch logic [6] and NULL convention logic [7], can be exploited to design the dual-rail data path. Martin's delay-insensitive full adder [8] can be used in the data path. Registers need to be dedicatedly designed for orderings (3) and (6) in Fig. 2. The structure of a 2-bit register is composed of C-elements and OR gates, as shown in Fig. 3. If the request signal from the post-stage is high to request data, then data are allowed to pass through the register, and when each bit is datum, the acknowledge signal will become low to request spacer from the pre-stage, which means the computation is finished and the circuit needs to be reset. Similarly, if the request signal from the post-stage is low to request spacer, then spacer is allowed to pass through the register, and when all of bits are spacers, the acknowledge signal will become high to request another data from the pre-stage, which means the reset is finished and the circuit can start another computation.

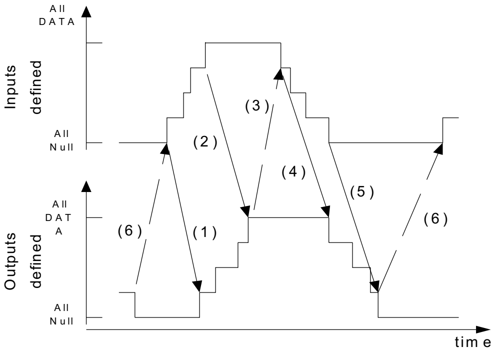

Fig.2 Seitz's weak condition

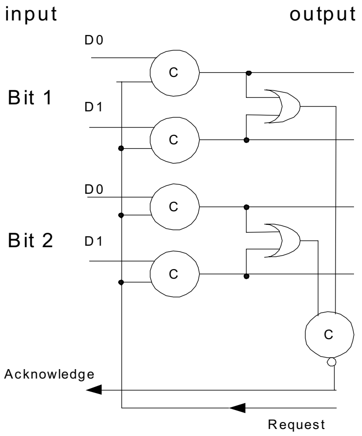

Fig.3 A 2-bit register

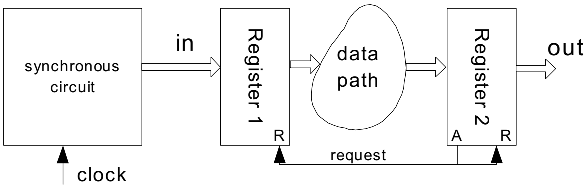

Fig. 4 (a) A self-timed circuit receives data from a synchronous circuit

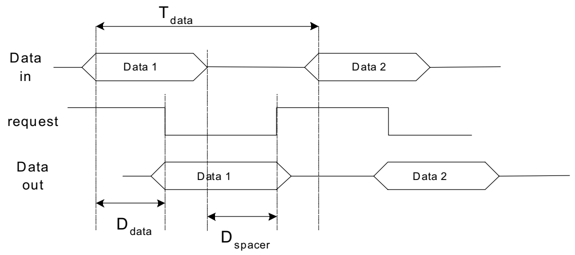

Fig.4 (b) Timing constraint of self-timed circuit

In many applications, the input data of the self-timed come from a synchronous system, such an A/D converter, and the data rate is constant and independent of the delay of the self-timed circuit, as shown in Fig. 4 (a). However, the allowed maximum input data rate is limited by the speed of the self-timed circuit. The timing constraint is illustrated in Fig. 4 (b), where T_{data} is the input data cycle, D_{data} is the propagation delay of data from register 1 to register 2, which includes the delays of two registers and the data path, similarly D_{spacer} is the propagation delay of spacer from register 1 to register 2. The sum of D_{data} and D_{spacer} must be no more than T_{data}. Otherwise, the self-timed circuit will miss some input data. Therefore, the allowed maximum input data rate is given by

$$f_{\max} = \frac{1}{D_{data} + D_{spacer}} \tag{2}$$

Usually a speed margin is needed to guarantee that the self-timed circuit works correctly.

3. Adaptive supply-voltage scheme

Under the assumption that the data path works faster at a fixed V_{DD} than the speed required by the input data rate, a feedback circuit can be designed to provide the data path with a lower supply voltage, as long as the delay of the data path meets the timing constraint in Fig. 4(b). The feedback circuit is implemented as shown in Fig. 5. It consists of a completion detector, a D-flip-flop and a DC-DC buck converter [9]. The completion detector can be constructed by C-elements. The output of the detector is high when a complete set of spacers arrives. The output is low when a complete set of data arrives. Otherwise, the output doesn't change. The high level output of the D-flip-flop implies that the data path is waiting for data input and that the data path works faster than required, and therefore that the supply voltage is allowed to decrease, and vice versa.

For the purpose of simulation, the data path is implemented by a long chain of inverters, which has a significant delay while the circuit is relatively simple for reasonable simulation time. The typical waveforms of the input and output of the buck converter are shown in Fig.6. Initially, the maximum voltage V_{DD} is applied to the self-timed data path by set M_1 on and M_2 off. Data and spacers are input to register 1 alternatively at a constant rate below the maximum allowed data rate. After one data cycle, the output of the D-flip-flop becomes high until A (in Fig. 6) because the request signal arrives at register 1 earlier than corresponding data does, which means that the data path operates too fast, and that the supply-voltage needs to decrease for saving power. However, V_x becomes low after A, which means that the data path operates too slowly, and that the supply-voltage needs to increase for speed. Actually, even when V_x is low, V_{out} still decreases from A to B due to the continuity of current in inductor, instead of increasing. To guarantee a safe speed, two methods can be used to compensate for the amount of voltage drop from A to B. One is to design some buffers before register 1 to store some data temporally in case the data path is too slow from A to B. Another one is to put a delay element on the clock input of the D-flip-flop to make the falling edge of V_x occur earlier a little bit, thus to increase V_{out} by a proper amount.

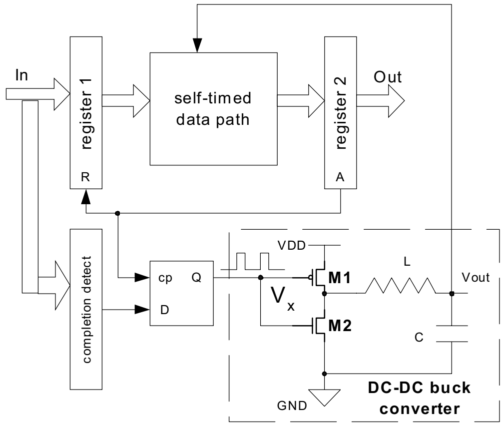

Fig. 5 The proposed adaptive supply-voltage scheme

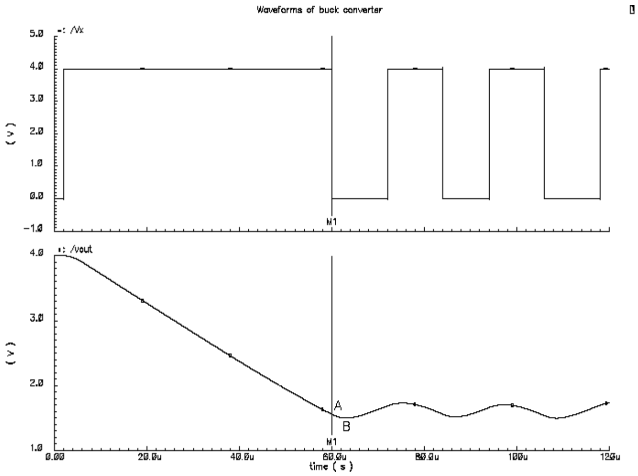

Fig.6 Waveforms V_x and V_{out} from Cadence simulation

When the circuit is stable, the waveform of V_x should be a pulse signal with a duty cycle D associated with an input data rate. The output of the buck converter is a rough DC voltage with a small ripple, and the DC component is given by

$$V_{out} = (1 - D) \cdot V_{DD} \tag{3}$$

The frequency of V_x , the sensitivity of data path delay to supply voltage V_{out} , and the efficiency of the converter are inextricably linked. A high sensitivity of data path delay to supply voltage leads to a high frequency of V_x. From Fig. 1, it is noticed that the delay sensitivity typically increases with the decrease of supply voltage. Therefore, the frequency of V_x mainly depends on the input data rate and the delay characterization of data path. To achieve a reasonable ripple on the output voltage, the values of L and C are chosen so that the LC frequency constant $f_0 = \frac{1}{2\pi\sqrt{LC}}$ is much smaller than the frequency of V_x. On the other hand, the increase of L and/or C will result in a longer time for the circuit to track the input data rate. Thus, choosing the values of L and C requires making tradeoff between ripple and transient performance.

4. Simulation results and discussion

In this section, some simulation results are presented to show the effectiveness of the proposed scheme, and possible improvements are discussed.

When the feedback loop is in steady state, the DC-DC converter will supply a DC voltage associated with the input data rate. As an example, the data path is designed as a chain of 26 inverters, and the adaptive supply-voltage scheme is simulated based on 0.18 μm CMOS technology. The dependences of supply voltage V_{out} and power saving on data rate are plotted in Fig. 7. When the maximal supply voltage V_{DD} = 4V is applied to the data path, the maximal allowed data rate is approximately 100 MHz. The supply voltage will adaptively decrease whenever the input data rate goes down.

The efficiency of the buck converter will affect the power performance of the whole circuit while the power dissipated by the logic control circuit in the feedback loop is negligible. The output power and efficiency of a basic converter are plotted in Fig. 8 when the equivalent load is 10 Ω.

To apply this adaptive supply-voltage scheme to high-speed low power system more effectively, the efficiency of the buck converter needs to be improved. An effective method of improving efficiency is zero voltage switching (ZVS) proposed in [9]. During switch transitions, the parasitic capacitance at drain terminals of transistors is shorted to either ground or the voltage V_{DD} through one of the power transistors. This causes the dissipation of energy $(C_x \Delta V^2)/2$, where C_x is the parasitic capacitance, ΔV is the voltage across the capacitor when it is shorted. ZVS is the technique of timing the switch transitions so that ΔV is zero. It is reported that the loss in the buck converter can be kept below 8% at 1 MHz switching frequency by using ZVS [9].

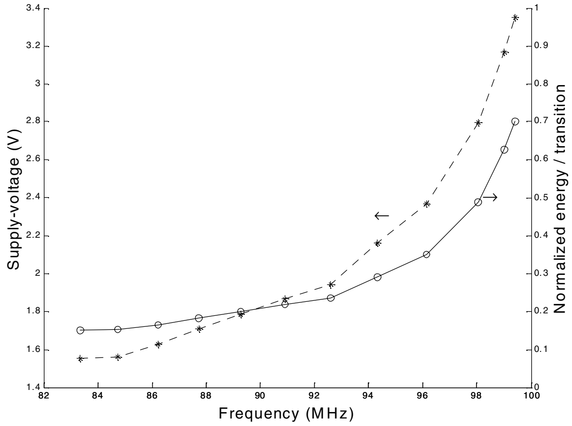

Fig.7 Supply voltage and energy vs. data rate

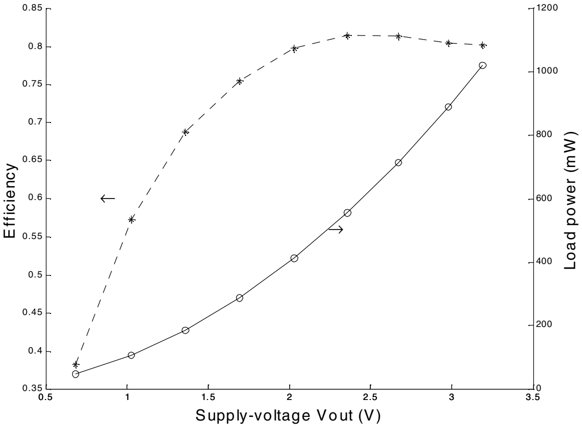

Fig.8 Power efficiency of buck converter

5. Conclusion

Low power operation can be achieved for data path by combining self-timed design with an adaptive supply-voltage scheme. The scheme proposed consists of a simple logic circuit and a buck converter. The handshake signals in self-timed pipeline are employed to track the input data rate automatically, and thus to keep the supply-voltage of the data path as small as possible. The cost of the overhead circuitry for logic control is negligible.

6. References

[1] N.H.Weste, K.Eshraghian, Principles of CMOS VLSI Design: a System Perspective,Addison-Wesley Publishing Company, 1993.

[2] A.P.Chandrakasan, et al,"Low-Power CMOS Digital Design,"IEEE *J. Solid-State Circuits*, vol.27, no.4, pp. 473-484, April 1992.

[3] T.Kuroda, et al, "Variable Supply-Voltage Scheme for Low-Power High-Speed CMOS Digital Design," IEEE *J. Solid-State Circuits*, vol.33, no.3, pp. 454-462, March 1998.

[4] L.S.Nielsen, et al, "Low-Power Operation Using Self-timed Circuits and Adaptive Scaling of the Supply Voltage," IEEE Trans. on *VLSI systems*, vol.2, No.4, pp. 391-397, Dec. 1994.

[5] C.Mead, L.Conway, *Introduction to VLSI systems*, Addison-Wesley Publishing Company, 1980.

[6] P.Ng, P.T.Balsara, and D.Steiss, "Performance of CMOS Differential Circuits," IEEE *J.Solid-State Circuits*, vol.31, no.6, pp. 841-846, June 1996.

[7] K.M.Fant and S.A.Brandt, "NULL Convention Logic: a complete and consistent logic for asynchronous digit circuit synthesis," *international conference on Application Specific Systems, Architecture, and Processors*, pp. 261-273, 1996.

[8] A.J.Martin, "Asynchronous datapaths and the design of an asynchronous adder," *Formal Methods in System Design*, vol.1, no.1, pp. 119-137, July 1992.

[9] A.Stratakos, et al, “A low-voltage CMOS DC-DC converter for a portable battery-operated system,” IEEE *Power Electronics Specialists Conference,* pp. 619- 626, 1994.