

論文 98-35C-7-5

# 웨이브 파이프라인 클럭 제어에 의한 1V-2.7ns 32비트 자체동기방식 병렬처리 덧셈기의 설계

## (1V-2.7ns 32b Self-Timed Parallel Carry Look-Ahead Adder with Wave Pipelined Clock Control)

林 訂 殖 \*, 曹 帝 秧 \*, 孫 日 憲 \*

(Jeong-Sik Lim, Je-Young Cho, and Ilhun Son)

### 요 약

본 논문에서는 1V 전원전압에 2.7nsec의 지연시간과 8mW의 전력소모를 갖는 32비트 병렬 덧셈기의 설계를 보여주고 있다. 회로설계는 단일문턱전압을 갖는 0.5 $\mu$ m 채널길이의 저전력 CMOS소자기술을 이용하였으며 논리구조로는 NORA논리 게이트를 채택하여 자체동기방식으로 동작하도록 하였다. 스테틱 CMOS 논리구조의 병렬처리 덧셈기와 차동전달게이트 논리구조(DPL)의 덧셈기에 비하여 전력소모에서는 큰 차이가 없으면서도 30%이상 빠른 계산속도를 가질 수 있었다. 이는 회로설계에서 웨이브 파이프라인 기법을 씀으로써 NORA논리의 가장 큰 결점인 예비충전(precharge)기간중의 불필요한 단락전류를 제거함으로써 가능하였다. 또한 NORA논리의 낮은 잡음마진도 스테틱 CMOS 논리구조와 같은 물리적설계를 함으로써 개선할 수 있었다.

### Abstract

A 32-b self-timed parallel carry look-ahead adder (PCLA) designed for 0.5 $\mu$ m single threshold low power CMOS technology is demonstrated to operate with 2.7nsec delay at 8mW under 1V power supply. Compared to static PCLA and DPL adder, the self-timed PCLA designed with NORA logic provides the best performance at the power consumption comparable to other adder structures. The wave pipelined clock control plays a crucial role in achieving the low power, high performance of this adder by eliminating the unnecessary power consumption due to the short-circuit current during the precharge phase. The noise margin has been improved by adopting the physical design of static CMOS logic structure with controlled transistor sizes.

### I. 서 론

시스템의 이동성이나 집적화를 향상시키기 위하여

\* 正會員, 檀國大學校 電子工學科

(Dept. of Electronics Engineering, Dankook University)

※ 본 연구는 '96년도 교육부 반도체 분야 학술 연구 조성비 (ISRC 96-E-2008)에 의하여 연구되었음  
接受日: 1998년 3월 23일, 수정완료일: 1998년 6월 29일

저전력 CMOS VLSI의 설계는 매우 중요한 문제로 부각되었다. 더불어 대부분 시스템의 성능은 산술회로 특히 덧셈기의 속도에 크게 영향을 받으므로 저전력, 고성능 덧셈기의 구현이 필수적이다. 저전력 VLSI의 설계에서 성능을 높이는 방법으로는 병렬처리와 파이프라인 구조를 적절히 배합함으로써 구조설계 단계에서 시스템의 전력소비 대비 성능을 높이는 것이 유효한 방법이다<sup>[1]</sup>. 따라서 본 논문에서는 병렬처리 덧셈기의 알고리즘과 구조를 검토하고 병렬처리 알고리즘

의 선택이 설계의 복잡성과 성능 및 셀의 면적등에 미치는 영향을 고려하였다. 저전력 VLSI 기술에서 또다른 중요한 기술적요소로는 공정과 소자기술로서 시스템의 성능 외에도 회로의 기능률을 고려하여 기존의 CMOS 기술에서 큰 변화 없이 저전력 시스템에 적합하고 실용적인 0.5 $\mu$ m CMOS 기술을 채택하여<sup>[2]</sup> 1V의 전원 전압에 200MHz로 동작시 8mW의 전력 소모에 2.7ns 지연시간을 가지는 32비트 자체동기방식 병렬처리 덧셈기의 설계를 제시하였다. 이 덧셈기의 회로설계에서 가장 큰 특징으로는 1V의 저전압에서도 높은 성능의 구현을 위하여 이진 트리(Binary tree)구조의 병렬처리 덧셈기<sup>[3]</sup>를 회로 구현함에 있어서 NORA 논리 구조<sup>[4]</sup>를 채택하였다는 것이다. 동급의 덧셈기로는 가장 빠른 계산 속도를 실현하였으며 DPL 구조 덧셈기<sup>[5]</sup> 등과 비교하여 시스템 성능지수인 PDP (Power-delay product) 및 절전효과 면에서 전혀 손색이 없음을 입증하였다. 이처럼 NORA 논리 회로가 저전압에서 효율적으로 동작할 수 있는 것은 본 논문에서 제안하는 웨이브 파이프라인 기법을 이용한 자체 동기방식 클럭 제어에 크게 기인한다.

웨이브 파이프라인 기법은 디지털 동기 시스템에서 성능을 높이기 위하여 일반적으로 사용하는 파이프라인 기법을 하드웨어 수준에서 보다 세분화한 것으로서 시스템 클럭에 의하여 데이터의 입출력이 제어되는 논리블럭 내의 신호 흐름을 파이프라인 기법으로 제어하는 회로기법으로서 일반 논리회로<sup>[6]</sup>로부터 데이터 경로가 복잡한 메모리<sup>[7,8]</sup>의 설계에까지 응용되고 있다. 본 논문에서 특이한 것은 이러한 웨이브 파이프라인 기법을 신호경로가 아닌 클럭 구동회로에 적용함으로써 NORA 논리구조로 구성된 병렬 덧셈기의 각 논리단계가 불필요한 전력소모와 내부 클럭간의 신호 대기시간이 없이 동작할 수 있도록 하였다는 것이다.

본 논문의 2장에서는 병렬 덧셈기의 구조를 3장에서는 저전력, 고성능 회로설계의 기본 개념과 회로기법을 설명하였으며 4장에서는 웨이브 파이프라인 기법에 의한 클럭 제어와 이의 결과를 논의하였다. 마지막으로 5장에서는 다른 구조의 덧셈기와의 비교를 포함한 시뮬레이션에 의한 결과 검토로서 끝맺음하였다.

## II. 32비트 이진트리 덧셈기의 구조

오늘날 DSP나 마이크로프로세서와 같은 VLSI에서

설계되는 거의 모든 덧셈기는 그 성능 향상을 위해 CLA(Carry Look-Ahead) 방식으로 설계되어진다. 이 방식에 의한 덧셈기를 구현하는 방법중 하나인 이진트리(Binary tree) 구조 덧셈기는 그 성능의 우수함과 VLSI 실현 가능성이 일찍 증명되었으나, 그 복잡성으로 인하여 불과 최근에 채택되어졌다<sup>[9]</sup>.

n-비트 병렬 덧셈기의 구조는 비록 반복적이지만 셀의 수가  $n \log_2 n$ 에 비례하므로 단순히 n에 비례하는 캐리선택방식의 덧셈기(Carry Select Adder)보다 셀의 수가 월등히 많고 내부에서의 신호연결이 복잡해 지므로 이러한 설계의 복잡성과 덧셈속도간에 적절한 타협을 하는 것이 매우 중요하다. 따라서 이 장에서는 병렬덧셈기에서의 캐리계산을 위한 알고리즘을 통하여 그 구조가 어떻게 정해 지는지를 검토한다.

덧셈기에서 j번째 비트 캐리는 그 전의 i번째 비트 캐리와 다음과 같은 논리 순환식(recursive equation)에 의해 표현된다.

$$c^j = g^j + b^j c^i, \quad n \geq j \geq i \geq 0. \quad (1)$$

여기에서  $g^j$ 와  $b^j$ 는 각각 j번째 비트에서 캐리를 생성(generate)하거나 i번째 비트의 캐리값을 그대로 전달(propagate)하는 작용을 표현하며 두 입력 데이터 A, B에 의하여는

$$b_i^{j+1} = A_i + B_i \text{ 또는 } A_i \oplus B_i, \quad g_i^{j+1} = A_i \cdot B_i \quad (2)$$

로 주어지며 식(1)의 순환식으로부터 p와 g의 일반적인 관계는

$$(b_i^k, g_i^k) \cdot (b_i^j, g_i^j) \rightarrow (b_i^k, g_i^k) = (b_i^k \cdot b_i^j, g_i^k + b_i^j \cdot g_i^j) \quad (3)$$

와 같이 표현된다. 위의 식에서 한가지 주목할 것은 왼쪽항의 논리연산이 CMOS논리 회로에서 AOI와 AND 또는 OAI와 OR를 번갈아 씌우어서 구현될 수 있다는 것이다. 식(1)로부터  $C^{32}$ 와  $C^0$ 의 관계는

$$C^{32} = (bg)_{31}^{32} \cdot (bg)_{31}^{30} \cdot \dots \cdot (bg)_0^1 \cdot C^0 \quad (4)$$

로 주어진다. 병렬덧셈기에서는 위의 모든 연산이 순차적으로 일어나도록 기다리지 않고 [10]에서 제안된 병렬 알고리즘에 의해 한꺼번에 병렬처리하고 있다. 즉,

$$\begin{aligned}
 0th : C^{32} &= (pg)_{31}^{32} \cdot (pg)_{31}^{30} \cdot \dots \cdot (pg)_0^1 \cdot C^0 \\
 1st : C^{32} &= (pg)_{30}^{32} \cdot (pg)_{28}^{30} \cdot \dots \cdot (pg)_0^2 \cdot C^0 \\
 1st : C^{31} &= (pg)_{29}^{31} \cdot (pg)_{27}^{29} \cdot \dots \cdot (pg)_1^3 \cdot C^1 \\
 2nd : C^{32} &= (pg)_{28}^{32} \cdot (pg)_{24}^{28} \cdot \dots \cdot (pg)_0^4 \cdot C^0 \\
 2nd : C^{31} &= (pg)_{27}^{31} \cdot (pg)_{23}^{27} \cdot \dots \cdot (pg)_1^5 \cdot C^1 \\
 2nd : C^{30} &= (pg)_{26}^{30} \cdot (pg)_{22}^{26} \cdot \dots \cdot (pg)_2^6 \cdot C^2 \\
 2nd : C^{29} &= (pg)_{25}^{29} \cdot (pg)_{21}^{25} \cdot \dots \cdot (pg)_3^7 \cdot C^3 \\
 &\vdots
 \end{aligned}
 \tag{5}$$

위의 식은 왼쪽에 병렬처리 단계를 표시하고 각 단계에서 행해지는 논리연산 내용을 나타낸다. 0번째 단계에서는  $p_i^{i+1}, g_i^{i+1}, i=0, 31$  이 식(2)에 의해 계산되고 1번째 단계에서는  $(pg)_i^{i+2}, i=0, 30$  들이 식(3)에 의해 계산된다. 2번째 단계에서는 이것들로부터  $(pg)_i^{i+3}$  이 계산되고 이런 식으로 5번째 단계에서는  $C^{32}$  와 그 이하 모든 캐리들의 계산이 끝나게 된다. 식(5)에서 주목할 것은 각 단계에서의 (pg)를 구하기 위한 논리연산의 수는 32개로 일정하므로 각 단마다 AOI 와 AND 또는 OAI 와 OR로 구성된 (pg)셀은 32개씩 있게 되며 각 단계마다의 지연시간 역시 (pg)셀의 지연시간에 의해 결정된다.

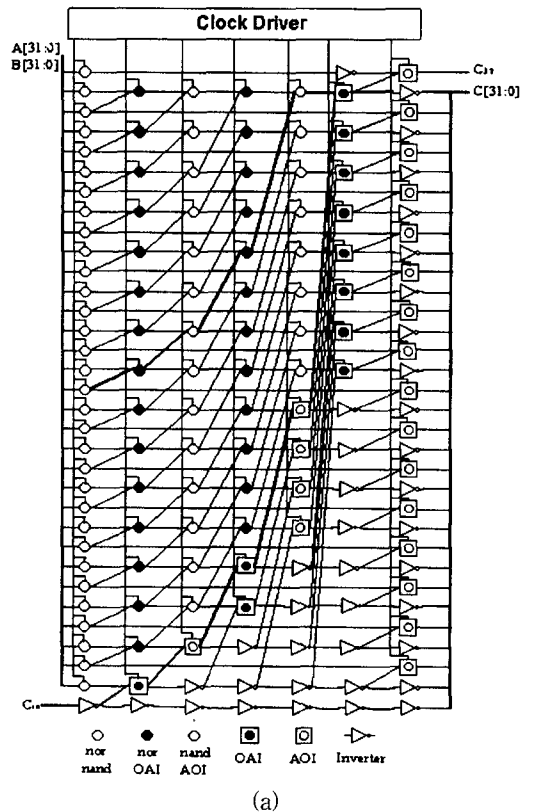
위에서 보인 병렬덧셈기의 구조에서는 병렬처리되는 넓이는 32이고 직렬처리되는 깊이는 6단계이다. 이러한 병렬처리되는 넓이와 직렬처리 깊이간에는 관련이 있으며<sup>[11]</sup> 식(5)의 병렬구조에서 (pg)셀의 수는 1번째 단계 이후에서 짝수 비트의 캐리들만 계산함으로써 거의 반으로 줄어 들수 있다. 이것은 실제로 본 연구에서 채택한 구조로서 각 단계별 논리구조는 다음과 같다.

$$\begin{aligned}
 0th : C^{32} &= (pg)_{31}^{32} \cdot (pg)_{31}^{30} \cdot \dots \cdot (pg)_0^1 \cdot C^0 \\
 1st : C^{32} &= (pg)_{30}^{32} \cdot (pg)_{28}^{30} \cdot \dots \cdot (pg)_0^2 \cdot C^0 \\
 2nd : C^{32} &= (pg)_{28}^{32} \cdot (pg)_{24}^{28} \cdot \dots \cdot (pg)_0^4 \cdot C^0 \\
 2nd : C^{32} &= (pg)_{26}^{30} \cdot (pg)_{22}^{26} \cdot \dots \cdot (pg)_2^6 \cdot C^2 \\
 &\vdots
 \end{aligned}
 \tag{6}$$

위의 논리 식(6)에서는 직렬처리의 깊이가 1단계 늘어나는 대신 1단계 부터의 (pg)셀의 수는 16으로 줄어들었다. 즉 주요신호경로에서 동작하는 게이트는 하나 증가되어 지연시간은 그만큼 길어졌지만 홀수 비트

캐리를 계산하지 않기 때문에 전체 게이트 수를 거의 절반으로 줄일 수 있어 이진트리 구조 덧셈기의 가장 큰 문제점인 설계의 복잡성을 크게 개선하였다. 또한 홀수 캐리가 계산되지 않은 것은 최종단의 합을 계산하는 회로부(sum logic)를 2비트 리플 캐리 덧셈기로 대체함으로써 해결된다. 식(6)의 구조를 가진 덧셈기의 전체 게이트수와 주요신호경로에서 동작하는 게이트수는 각각  $\frac{n}{2}(\log_2 n + 2), \log_2 n + 2$ 이다.

그림 1(a)는 식 (6)에 표현된 병렬처리 알고리즘에 따라 구성된 32비트 이진트리 구조 덧셈기에서 캐리전달 부분의 블럭도와 주요신호경로의 회로를 나타낸 것이다. 그림 1(a)에서는 0번째 단계에서 NAND와 NOR 게이트에 의해 생성된 p, g값들이 어떻게 그 이후 단계의 (pg)셀들에 전달되는 지를 보여준다. 그림 1(b)는 0번째와 1, 2번째 단계의 NORA 논리구조 (pg)셀들의 회로도를 보여주고 있다. 그림 1(a)에서 굵은 선으로 표시한 부분은 주요신호경로를 표시하고 이 주요신호경로를 최적화 하기 위하여 그 평가 척도로서 PDP를 최소화 하였으며 이를 위한 게이트 설계는 다음 장에서 설명하였다.



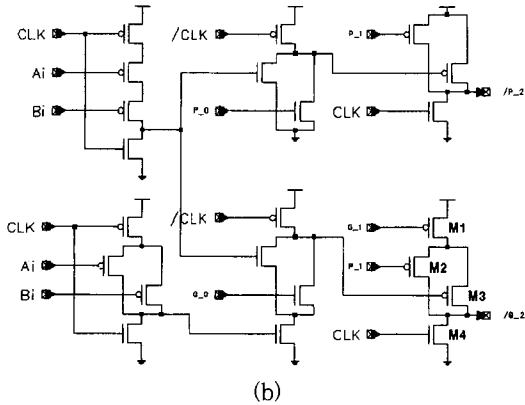


그림 1. (a)병렬 캐리 전달회로의 구성도와 (b) NORA 논리구조의 (pg)셀 회로도  
 Fig. 1. (a) Structural view of PCLA schematic (b)Schematic of NORA logic structured (pg) cells

III. PCLA의 최적 구현을 위한 회로 기법

일반적으로 저전압 설계에서는 낮은 잡음 내성과 높은 누설 전류로 인한 저주파수 동작시의 문제 등으로 다이내믹 논리 구조를 기피하지만 성능 면에서 NORA와 스테틱논리구조 간에는 2배 가까운 큰 차이를 보인다<sup>[4]</sup>

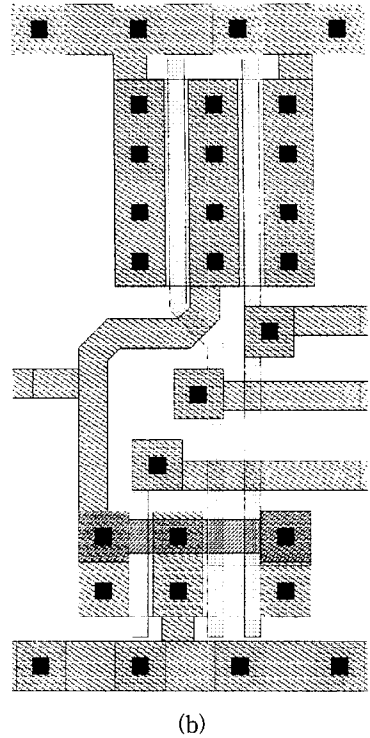
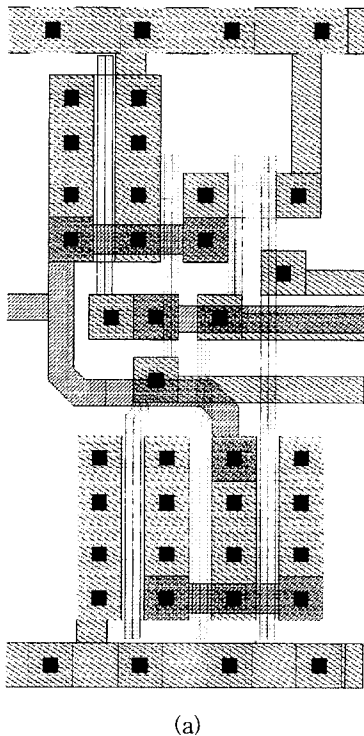


그림 2. 기본 논리셀의 물리적 설계 (a)OAI 게이트 (b)NAND 게이트  
 Fig. 2. Physical layout of basic logic gates (a) OAI and (b)NAND

본 연구에서는 이러한 문제를 해결하기 위하여 실제 회로의 물리적 설계에서는 그림 2에 보인 바와 같이 NORA와 스테틱 논리 구조를 결합한 회로 구성을 하였으나 NORA 논리구조를 구성하는 소자들을 제외하고는 그 크기를 물리설계 룰이 허용하는 최소 크기로 함으로써 회로동작과 성능 면에서는 NORA 논리 구조에 의한 설계와 거의 같도록 하면서도 NMOS와 PMOS NORA논리 게이트의 논리문턱전압이 각각 0.3V와 0.46V가 되도록 함으로써 충분한 잡음내성을 갖추도록 하였다. 따라서 설계최적화를 위한 검토에서는 스테틱 논리구조에 따라 추가된 소자들은 무시하도록 한다.

그림 1(a)에서는 이진 트리 구조의 덧셈기에서 병렬 캐리 전달회로의 구성도를 보여주고 있다. 이 회로는 첫째 단의 (pg) 발생회로가 NAND와 NOR 게이트들로 구성되었으나 둘째단 이후의 캐리전달 회로 부분은 OAI와 AOI가 번갈아 가며 연결되어 있다<sup>[4]</sup>. 중요한 것은 이 모든 게이트들이 그림 1(b)에 보인 것처럼



기본적으로 다이내믹 회로의 NORA 논리구조로서 구성되어 있으며 저전력 고성능 덧셈기 설계를 위하여 0.5 $\mu$ m 저전력 CMOS 기술<sup>[2]</sup>을 채택하였다는 것이다.

이 저전력 CMOS 기술의 소자특성은 PMOS와 NMOS가 각각 350mV, 200mV의 낮은 문턱 전압을 가지며 1V의 전원 전압에 비하여 그 상대적인 차이가 큼으로 전류 구동 능력에서도 3.5배의 큰 차이를 보인다. 일반적인 게이트 최적화 방법에서는 출력의 상승/하향 시간을 같도록 맞추지만<sup>[12]</sup> 이처럼 PMOS와 NMOS 간의 차이가 클 경우 특히 PN 소자가 병렬처리 단계마다 교대로 동작하고 출력부하비율(fan-out)이 일정한 NORA 논리회로에서는 각 단계마다의 게이트를 이룬 트랜지스터의 크기비율(tapering factor)보다는 PMOS와 NMOS 간의 크기의 비율을 최적화하는 것이 중요하다. 그림3에 보인 NORA 논리 구조에서 PDP는 트랜지스터의 크기와 드레인 포화전류  $I_{D, sat}$ 에 대하여

$$PDP \propto \frac{[(W_p + W_n)C_j + \beta W_p C_g]^2}{W_n \cdot I_{n, sat}} + \frac{[(W_p + W_n)C_j + \beta W_n C_g]^2}{W_p \cdot I_{p, sat}}$$

$$= \frac{W_n[(1 + \alpha)C_j + 1.5\alpha C_g]^2}{I_{n, sat}} + \frac{W_n[(1 + \alpha)C_j + \beta C_g]^2}{\alpha \cdot I_{p, sat}} \quad (7)$$

의 관계를 가지며 식에서  $\alpha = W_p / W_n$ ,  $\beta$ 는 출력 부하비율로서 NORA 논리회로에서는 그림 1(a)에서 알 수 있는 바와 같이 3의 값을 가진다.  $C_j$ 와  $C_g$ 는 각각 접합캐패시턴스와 게이트캐패시턴스를 나타낸다. 그림 4는 식(7)을 몇 가지 서로 다른  $C_j$ 와  $C_g$ 의 비율에 대하여 계산한 결과이며 이를 최소화하는  $\alpha$  값은 주어진 소자 특성에서 대략 1.2 ~ 1.3 이 된다. 이 결과는 일반적인 설계방식에서 출력의 상승/하향 시간을 같도록 맞추기 위하여  $W_p$ 와  $W_n$ 의 크기 비율을 이 경우 3.5로 하는 것과 비교하면 그림 4에서 알 수 있듯이 PDP에서 상당한 차이를 보이게 된다. 실제 SPICE 시뮬레이션 결과는 지연시간에서 더욱 차이가 나며 PDP가 최적화되는 점은 식(7)에 의한 계산과 일치함을 알 수 있다. 즉 NORA 논리 구조와 같은 다이내믹 회로에서는 이  $\alpha$  값이 PDP 뿐만 아니라 게이트 지연 시간을 최소화함으로, 본 연구에서 추구하는 저전력 고성능 회로 설계에 있어 매우 중요하다. 실제로  $\alpha$ 를 1.3로 하였을 경우와 일반적인 게이트 최적화 방법대로  $\alpha$ 를 3으로 하였을 경우에는 속도면에서

각각 2.5ns와 2.9ns으로 큰 차이를 보인다.

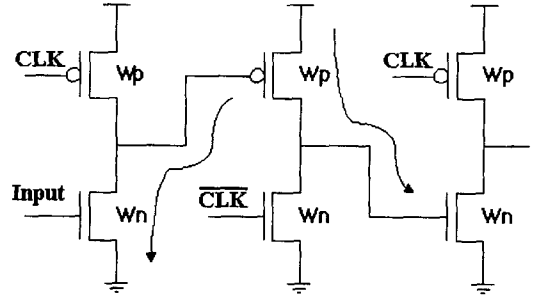


그림 3. NORA 논리게이트의 스위칭 동작  
Fig. 3. Switching operation of NORA logic gates.

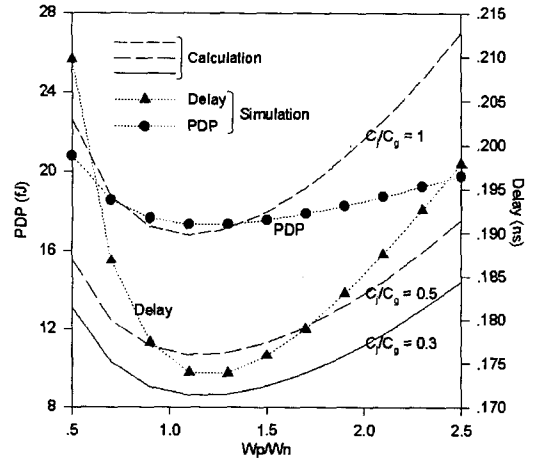


그림 4. 트랜지스터 크기 비율에 따른 PDP 및 지연시간의 변화  
Fig. 4. PDP and gate delay as functions of transistor size ratio.

#### IV. 웨이브 파이프라인기법에 의한 자체동기방식 회로 구성

그림 5에 표시된 블록도에서 병렬처리 덧셈기는 입력 레지스터와 출력 래치, 그리고 캐리 전달부와 합을 계산하는 회로부 및 이들 각 회로부에 서로 다른 타이밍의 자체 발생된 클럭을 공급하는 내부 클럭구동부로 이루어져 있다. 여기서 전체 덧셈기의 성능을 좌우하는 캐리전달부는 이미 설명된 바와 같이 다이내믹 NORA 논리구조로 구성되어 있으며 그 (pg)셀의 회로도도 그림 1(b)에 보인 바와 같다. NORA 논리구조는 특성상 신호 경주(racing)의 문제없이 다이내믹 회로를 직접 연결함으로써 매우 빠른 속도를 실현할 수 있으나, 도미노 회로나 스테틱 회로 설계에 비하여

예비충전 기간 중에 앞단의 신호가 미처 예비충전되어 있지 않음으로 인하여 발생하는 단락 전류가 지나치게 크다. 또다른 문제는 이러한 NORA 논리 구조로서 저전력 고성능 시스템을 구현하기 위하여 자체동기방식 회로로 구성하려 할 경우, 신호 입력으로부터 출력까지의 지연 시간을 정확히 예측하여 신호 경주에 의한 오동작 없이 지연 시간을 최소화할 수 있는 자체동기방식 클럭 발생회로가 필요하다는 것이다. 이를 위하여 본 논문에서는 그림 6에 보인 바와 같은

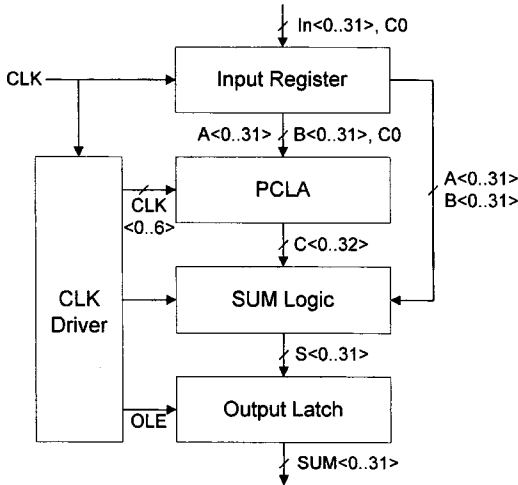


그림 5. 병렬처리 덧셈기의 시스템 구성도  
Fig. 5. Block diagram of PCLA.

웨이브 파이프라인 기법을 이용한 자체동기방식 클럭 구동회로를 제안한다. 그림 6의 자체동기방식 클럭 구동회로는 기본적으로 인버터들로 구성된 클럭 구동회로와 NAND 게이트로 된 펄스 발생회로의 입력에 D-래치를 연결함으로써 그림 7(a)의 회로시뮬레이션 결과에서 보인 것처럼 Adder Enable(AE) 신호에(일반적인 시스템에서는 시스템 클럭이 AE에 공급된다) 가해지는 타이밍의 제약을 해결하였다. 즉 AE신호는 입력 레지스터와 클럭구동부의 D-래치에 동시에 공급되며 그림 7(b)에서 볼 수 있듯이 클럭 구동의 0번째 클럭은 입력 레지스터의 지연시간 만큼 지연동기되어 발생된다. 또한 각 병렬처리 단계에 공급되는 클럭들의 지연시간을 캐리전달부의 NORA 게이트들이 가지는 지연시간과 일치되도록 함으로써 캐리 전달부의 주요 신호경로에서 불필요한 지연이 생기지 않도록 하였으며 그림 7(c)에서 볼 수 있듯이 덧셈기 지연 시간과 일치하는 출력 Latch Enable(OLE) 신호를 발생시키

도록 하여 다이내믹 회로가 가지는 입출력 데이터와의 타이밍 제약(setup/hold time)에 따른 문제를 해결하였다. 그림 7(b)의 Clk1은 병렬 덧셈기의 첫번째단인 (pg)발생회로에 공급되며 Clk2에서 6까지는 그 다음의 다섯 논리단계에 각각 공급된다.

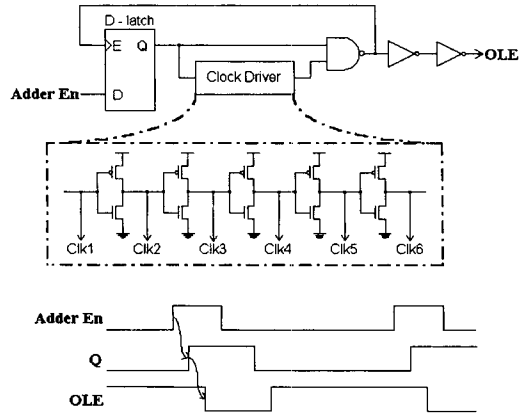
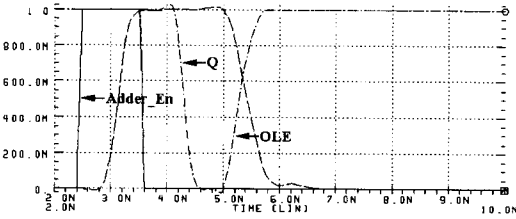


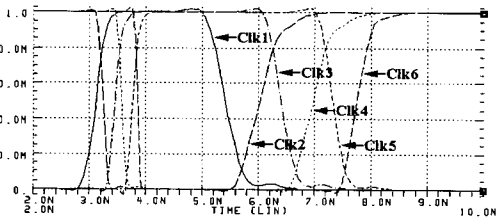
그림 6. 웨이브 파이프라인 기법에 의한 자체동기 방식 클럭 구동회로  
Fig. 6. Schematic of self-timed clock driver with wave-pipelined clock control.

클럭 구동 회로는 기본적으로 덧셈기의 NORA 게이트들의 지연시간과 일치하는 인버터들로 구성되었으며 논리평가기간(evaluation)과 예비충전 기간 중에 서로 다른 PN 소자들이 교대로 동작하는 이치를 이용하여 그림 7(b)에 보인 것처럼 두 기간 중에 상이한 클럭 전파의 지연 시간을 가지도록 설계하였다. 이로써 논리평가기간 중에는 NORA 논리회로의 지연 시간과 일치하는 특성을 가지도록 하고, 예비충전 기간 중에는 긴 지연 시간을 갖게 하여 각 병렬처리단계들이 차례로 예비충전되도록 함으로써 그림 8에 보인 바와 같이 예비충전 기간 중의 단락 전류를 제거하였다. 그림 1(b)에서 M1은 PMOS 논리소자이고 M4는 NMOS 예비충전소자이다. 따라서 이 게이트는 클럭이 올라갈 때 예비충전이 일어나며 그림 8(a)에서 클럭이 올라갈 때 보이는 I(M1)은 M1의 입력 데이터가 클럭과 상충되므로서 흐르는 불필요한 단락전류이다. 그림 8(b)에서는 웨이브 파이프라인 기법으로 M4에 공급되는 클럭(Clock2)의 타이밍이 M1의 입력이 예비충전되는 것과 맞게 함으로써 그러한 단락전류를 제거하였다. 그림 9에서는 전력소모 성분들을 예비충전 기간 과 논리평가기간으로 나누어 캐리전달부와 합을

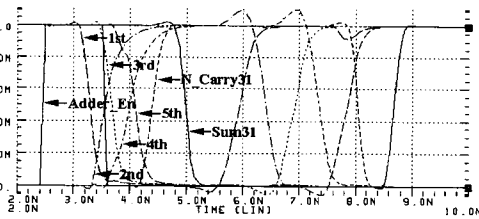
계산하는 회로부, 클럭구동부의 각각에 대하여 나타내었다. 한 가지 타이밍의 클럭만을 쓸 경우 약 20% 정도의 추가 전력을 예비충전 기간 중의 단락 전류로 인하여 소모함을 알 수 있다.



(a)



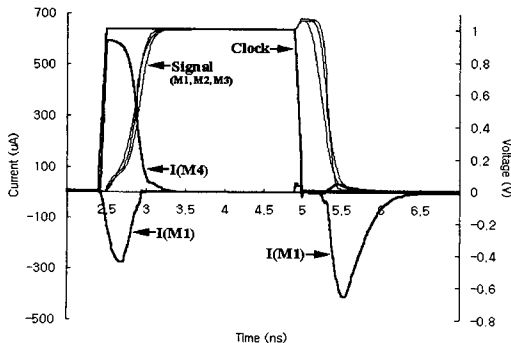
(b)



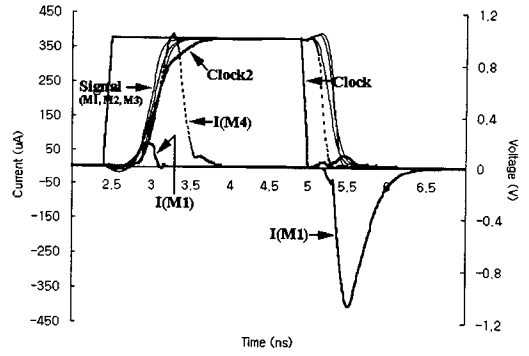
(c)

그림 7. 회로 시뮬레이션에 의한 주요 신호 전압파형 (a) 입출력 제어클럭 (b) 병렬처리 각 단계에 공급되는 클럭 (c) 주요 신호 경로에서의 캐리 전달

Fig. 7. SPICE simulation results. (a) I/O control signals, (b) clock signals for each parallel carry look-ahead logic stages and (c) carry propagation through the critical path



(a)



(b)

그림 8. 그림 1(b)의 M1 과 M4에 흐르는 전류소모 (a)고정클럭과 (b) 웨이브 파이프라인 클럭에 의한 동작시

Fig. 8. Switching current consumptions in M1 and M4 of Fig.1(b) for (a) fixed timing clock and (b) wave-pipelined clock control

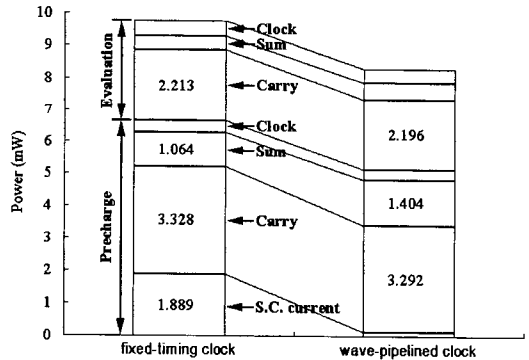


그림 9. 고정클럭과 웨이브 파이프라인 클럭에 의한 덧셈기의 전력소모 비교

Fig. 9. Comparison of PCLA power consumptions between fixed-timing clock and wave-pipelined clock control.

### V. 시뮬레이션 결과 및 검토

그림 10에서는 스택 CMOS와 NORA 논리회로로 설계한 병렬처리 덧셈기 및 DPL 덧셈기의 성능을 PDP 값으로서 여러 가지 다른 NMOS 문턱전압에 대하여 보여주었다. PDP면에서는 DPL 덧셈기에 비하여 병렬처리 덧셈기가 10 ~ 20% 우수하다. 스택 논리구조와 다이내믹 NORA 구조 사이에는 PDP의 차이는 별로 없지만 지연시간에서 3.6ns와 2.7ns로 크게 차이가 나고 특히 저전력 설계에서 중요한 대기 상태에서의 전력소모가 다이내믹 NORA 구조의 경우

2배 가량 적다. 이는 다이내믹 NORA 논리구조에서 누설전류는 주로 직렬연결된 논리소자들을 통해 흐른다는 점과 다이내믹 구조에서 작아진 부하크기 만큼 줄어든 트랜지스터 크기에 기인한다. [2]에서와 같이 덧셈기의 설계에서도 동작시의 소모 전력에 비하여 대기상태 전력소모가 충분히 적기(0.1% 이하) 위하여는 NMOS소자의 문턱전압은 200mV가 적합함을 알 수있다.

표 1. 200mV 문턱전압에서 3가지 유형 덧셈기의 성능 비교

Table 1. Performance comparison of 3 different adders with V<sub>TN</sub> of 200mV.

	트랜지스터 수	속도 [ns]	소비 전력 [mW]	대기상태 전력소모[ $\mu$ W]
DPL	2610	4.8	2.8	1.4
Static	1620	3.6	3.2	1.8
Dynamic	1876	2.5	4.6	0.92

로 구성으로 인하여 대기상태에서 클럭구동으로 인한 전력소모가 없다는 장점을 가진다. 트랜지스터의 수는 DPL의 경우 캐리 선택방식과 차동회로구성으로 인하여 직렬 처리 방식임에도 불구하고 2610개로 가장 많으며 같은 병렬처리 구조에서도 NORA 논리구조의 경우 pseudo 다이내믹 회로구조로 인하여 클럭 스위칭 및 클럭 구동회로를 위한 트랜지스터 수만큼이 많다. 그림 11은 IDEC MPC에서 0.8 $\mu$ m, 단일 폴리, 2 메탈 공정으로 물리적 설계한 것으로서 전체 셀면적은 2880 x 490  $\mu$ m<sup>2</sup>이다.

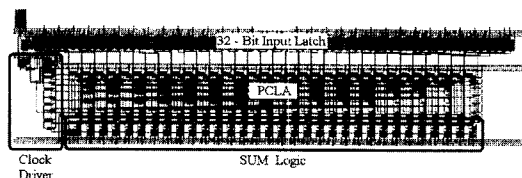


그림 11. 병렬처리 덧셈기의 물리적 설계  
Fig. 11. Physical layout of PCLA.

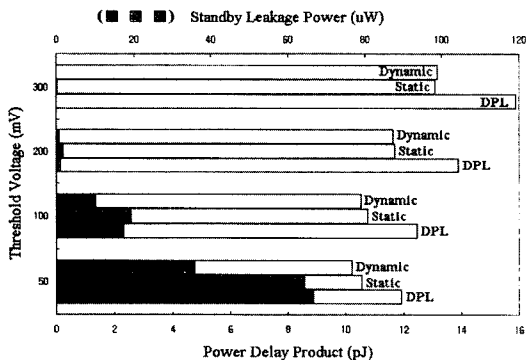


그림 10. NMOS트랜지스터의 문턱전압 변화에 따른 서로 다른 덧셈기들의 성능 비교

Fig. 10. Comparison of PDP and stand-by power for 3 different adder designs with various NMOS threshold voltages.

표 1에는 NMOS소자의 문턱전압 200mV로 설계된 세가지 덧셈기의 성능을 정리하였다. 2.5ns의 덧셈기만의 지연시간에 자체동기방식 클럭에 의한 타이밍 마진 포함하여 전체 지연시간 2.7ns의 다이내믹 병렬처리 덧셈기는 4.8ns의 지연 시간을 보이는 DPL 덧셈기와는 전혀 다른 응용분야를 가질 것이다. 다이내믹 덧셈기는 클럭이 "0"로 머물러 있는 대기상태(AE = 0)에서 DPL 덧셈기의 1.4 $\mu$ A에 비하여 0.9 $\mu$ A의 매우 낮은 대기 상태 전류를 보이며 또한 자체동기방식 회

참 고 문 헌

- [1] C. Piguet, "Ultra Low-Power Digital Design," in *Low-Power/Low-Voltage IC Design*, by C. Piguet, Jun., 1995, Lausanne, Switzerland.
- [2] K.Chung, et al., "Low Power-High Performance Embedded SRAM Circuit Techniques With Enhanced Array Ground Potential", in this issue.
- [3] R.P.Brent and H.T.Kung, "A Regular Layout for Parallel Adders", *IEEE Trans. on Computers*, vol. 31, pp. 260-264, 1982.
- [4] N.F.Goncalves and H.J.De Man, "NORA: A Racefree Dynamic CMOS Technique for Pipelined Logic Structures", *IEEE J. Solid-State Circuits*, vol. 18, pp. 261-266, 1983.
- [5] A. Charnas, et. al., "A 64b Microprocessor with Multimedia Support," *ISSCC Digest of Technical Papers*, pp. 178-179, 1995.
- [6] D. C. Wong, et al., "A Bipolar Population Counter Using Wave Pipelining to Achieve 2.5x Normal Clock Frequency", *IEEE J. Solid-State Circuits*, vol. 27, pp.



- 745-753, 1992.
- [7] H.-J. Yoo, et. al., "A 150MHz 8-banks 256M Synchronous DRAM with Wave Pipelining Methods," ISSCC Digest of Technical Papers, pp. 250-251, 1995.
- [8] K. Ishibashi, et. al., "A A 300MHz 4Mb Wave-Pipeline CMOS SRAM Using a Multi-Phase PLL," ISSCC Digest of Technical Papers, pp. 308-309, 1995.
- [9] P.M.Kogge and H.S.Stone, "Parallel Algorithm for the Efficient Solution of a General Class of Recurrence Equations", IEEE Trans. on Computers, vol. C-22, no. 8, pp. 786-793, Aug. 1973.
- [10] J.Y.Cho, et al., "Low voltage high performance binary tree adder design", Proc. of KITE Summer Conf., vol. 19, pp. 341-344, 1996.
- [11] J.S.Choi and K.R.Lee, "Design of CMOS Tapered Buffer for Minimum Power-Delay Product", IEEE J. Solid-State Circuits, vol. 29, pp. 1142-1145, 1994.
- [12] M.Suzuki, et al., "A 1.5ns 32b CMOS ALU in Double Pass-Transistor Logic", ISSCC Tech. Dig., pp. 90-91, 1993.

---

 저 자 소 개
 

---



林 訂 殖(正會員)

1974年 10月 21日生 1996년 2월 단국대학교 전기공학과(학사). 1996년 8월 ~ 현재 단국대학교 대학원 전자공학과 석사 과정 재학중. 주관심분야는 저전력 VLSI 설계 및 전력 전자 회로



曹 帝 奭(正會員)

1972年 11月 7日生 1995년 2월 단국대학교 전자공학과(학사). 1997년 2월 단국대학교 대학원 전자공학 석사. 1997년 3월 ~ 현재 해태전자연구소 연구원. 주관심분야는 VLSI 설계, DSP 등

孫 日 憲(正會員) 第 35卷 C編 第 2號 參照