

論文99-36C-5-4

글리치 감소를 통한 저전력 16비트 ELM 덧셈기 구현

(An Implementation of Low Power 16bit ELM Adder by Glitch Reduction)

柳凡善*, 李奇榮*, 趙泰元*

(Beom Seon Ryu, Kie Young Lee, and Tae Won Cho)

요 약

저전력을 실현하기 위하여 구조, 논리 및 트랜지스터레벨에서 16비트 덧셈기를 설계하였다. 기존의 ELM 덧셈기는 입력 비트 패턴에 의해 계산되는 블록캐리발생신호(block carry generation signal) 때문에 특정 입력 비트 패턴이 인가되었을 때에는 G셀에서 글리치(glitch)가 발생하는 단점이 있다. 따라서 구조레벨에서는 특정 입력 비트 패턴에 대해서 글리치를 피하기 위해 자동적으로 각각의 블록캐리발생신호를 마지막 레벨의 G셀에 전달하는 저전력 덧셈기 구조를 제안하였다. 또한, 논리레벨에서는 정적 CMOS(static CMOS) 논리형태와 저전력 XOR게이트로 구성된 저전력 소모에 적합한 조합형 논리형태(combination of logic style)를 사용하였다. 게다가 저전력을 위하여 트랜지스터레벨에서는 각 비트 전파의 논리깊이(logic depth)에 따라서 가변 크기 셀들(variable-sized cells)을 사용하였다. 0.6 μ m 단일폴리 삼중금속 LG CMOS 표준 공정변수를 가지고 16비트 덧셈기를 HSPICE로 모의 실험한 결과, 고정 크기 셀(fixed-sized cell)과 정적 CMOS 논리형태만으로 구성된 기존의 ELM 덧셈기에 비해 본 논문에서 제안된 덧셈기가 전력소모면에서는 23.6%, power-delay-product면에서는 22.6%의 향상을 보였다.

Abstract

We have designed a 16bit adder which reduces the power consumption at each level of architecture, logic and transistor. The conventional ELM adder has a major disadvantage which makes glitch in the G cell when the particular input bit patterns are applied, because of the block carry generation signal computed by the input bit pattern. Thus, we propose a low power adder architecture which can automatically transfer each block carry generation signal to the G cell of the last level to avoid glitches for particular input bit patterns at the architecture level. We also use a combination of logic styles which is suitable for low power consumption with static CMOS and low power XOR gate at the logic level. Furthermore, The variable-sized cells are used for reduction of power consumption according to the logic depth of the bit propagation at the transistor level. As a result of HSPICE simulation with 0.6 μ m single-poly triple-metal LG CMOS standard process parameter, the proposed adder is superior to the conventional ELM architecture with fixed-sized cell and fully static CMOS by 23.6% in power consumption, 22.6% in power-delay-product, respectively.

* 正會員, 忠北大學校 電氣電子工學部
(School of Electronic and Electrical Engineering,
Chungpook University)

※ 본 연구는 산업자원부와 과학기술부 및 정보통신부

에서 시행하는 주문형 반도체 개발 사업의 지원을
받아 수행되었습니다.

接受日字: 1998年12月3日, 수정완료일: 1999年4月22日

I. 서 론

이전의 회로설계에서 대부분의 연구는 VLSI의 면적과 성능을 최적화하는 방법에 대하여 집중적으로 연구되어왔다. 최근들어, 디지털 시스템의 성능과 복잡도의 증가에 따른 과도한 열 방출과, PDA와 개인 통신 장비와 같은 휴대용 장비의 수요 급증으로 말미암아 저전력 설계가 회로설계에서 중요한 목표중의 하나가 되었다.

덧셈기는 디지털 회로 시스템에서 가장 많이 사용되는 연산기 중의 하나이며, 주어진 시스템의 전력소모를 줄이기 위해서는 덧셈기 등과 같은 매크로 블록(macro block)의 전력소모를 줄이는 것이 필수적이다. 지금까지 수많은 병렬 덧셈기들이 연구되어 왔으며^[1-4], 최근에는 carry look-ahead adder의 변형이고 캐리의 계산을 parallel prefix computation 방법으로 구현하는 tree-based look-ahead adder들이 보고되고 있다^[1-3]. 이들은 이진트리(binary tree)로 구성되며, 블록 단위의 블록캐리발생신호 및 블록캐리전파신호(block carry propagation signal)를 계산하여 병렬적으로 합을 계산함으로써, 설계규칙성이 뛰어난 덧셈기들이다. 그 중에서도 ELM 덧셈기는 속도, 전력소모, 면적 등을 평가기준으로 하여 모든 비트 수에 걸쳐서 최적이라고 보고되고 있다^[4].

ELM 덧셈 알고리즘은 주어진 블록에 대해 모든비트에 대한 블록캐리발생신호를 계산하는 B & K 알고리즘(Brent & Kung algorithm)^[1]과는 달리, 하위 레벨의 블록캐리발생신호가 이진트리를 통과하면서 부분합(partial sum)에 반영됨으로써 주어진 블록에 대해 오직 한 비트의 블록캐리발생신호를 계산한다. 이로 인해 ELM 덧셈기는 셀 간의 연결선의 수가 Brent & Kung 덧셈기에 비해 적고 따라서 면적이 줄어들고 동작속도도 증가한다^[2,4].

한편 ELM 덧셈기는 주어진 비트수에서 상위비트가 '1'인 비트패턴에 대해서는 현재 블록의 블록캐리발생신호가 다른 하위비트의 값에 관계없이 '1'이 된다. 즉, ELM 덧셈기에서 블록캐리발생신호를 계산하기 위해서 모든 레벨의 G셀을 통과하지 않고도 현재 입력되는 비트패턴에 따라서 미리 현재 블록의 블록캐리발생신호가 결정된다. 일반적인 ELM 덧셈기에는 E, P, S, 및 G셀 등 4개의 기본 셀이 존재하는데^[2], G셀과 P셀은 각각 블록캐리발생신호나 블록캐리전파신호

를 계산하며, E 및 S셀은 부분합을 계산한다. 기존의 ELM구조는 임의의 블록의 상위비트 값이 '1'인 특정 입력 비트 패턴이 인가되었을 때, 현재 블록의 블록캐리발생신호를 계산하기 위하여 불필요하게 상위레벨의 G셀을 동작시키게 된다. 이로 인하여 상위레벨의 G셀의 내부노드 및 출력이 스위칭하게 되고 따라서 부가의 전력소모가 발생한다. 본 논문에서는 이러한 기존의 ELM 덧셈기의 단점을 해결하기 위하여 현재 입력되는 비트패턴을 고려하여 글리치를 최소화시킨 저전력 덧셈기를 제안하고자 한다. 기존의 ELM덧셈기가 모든 입력 비트 패턴에 대해서 일반경로를 거쳐서 블록캐리발생신호를 계산하는데 비하여, 제안한 구조는 특정 입력 비트 패턴에 대해서는 일반경로를 거치지 않고 단축경로(short path)를 통과함으로써 상위레벨에 위치한 G셀에서의 글리치를 제거한 구조이다. 본 논문에서 사용한 특정 입력 비트 패턴이란 제안한 덧셈기구조에서 블록캐리발생신호가 기존의 구조처럼 일반경로를 통과하지 않고 단축경로를 통과하는 모든 입력 비트 패턴들을 말한다. 또한 제안한 구조가 특정 입력 비트 패턴을 고려한 구조이므로, 비트 수의 확장에 따른 특정 입력 비트 패턴의 발생확률을 구하여, 비트 확장시 제안한 구조의 효율성에 대해서도 언급한다.

한편 저전력 실현을 위해 논리레벨에서 이루어지는 일은 적절한 논리형태(logic style)의 선택이다^[5-13]. 현재까지 저전력 실현을 위한 다양한 논리형태가 제안되었으며^[5-13], 최근에는 여러 종류의 논리형태 중에서 성능이 우수한 개별 논리 게이트 등을 혼합하여 사용하는 조합형 논리형태가 저전력에 적합하다고 보고되었다^[11-12]. 따라서 본 논문의 논리레벨에서는 저전력 실현을 위해 조합형 논리형태로 덧셈기를 구현한다. 트랜지스터레벨에서 전력소모를 줄이기 위해서는 동작속도와 무관한 트랜지스터의 크기를 작게 하여 기생 캐패시턴스를 최소화해야 한다^[5-6]. ELM 덧셈기 구조는 하위비트에 위치한 셀의 논리깊이는 상위비트에 위치한 셀 보다 작아 지연시간이 상대적으로 작은 특징이 있다. 따라서 이 곳에 위치한 셀의 트랜지스터 크기를 작게 해도 전체속도에는 영향을 미치지 않는다. 전력소모를 줄이기 위해 본 논문의 트랜지스터레벨에서는 가변 크기 셀을 이용하여 16비트 덧셈기를 설계한다. 본 논문의 구성은 다음과 같다. II절에서는 특정 입력 비트 패턴을 고려하여 G셀의 글리치를 감소

시킨 저전력 구조 및 실제적으로 저전력을 위해 사용되는 논리 및 트랜지스터레벨에서의 저전력 덧셈기의 구현에 대해 설명한다. III절에서는 각각의 경우에 대한 실험결과를 보이고 이를 분석한다.

II. 제안한 덧셈기 설계

일반적인 ELM 덧셈기 구조는 이진트리로 구성되며 각각의 이진트리는 하나의 블록을 형성하고 각 블록은 세 종류의 신호를 계산하여 상위레벨로 올려준다^[2,4]. 세 종류의 신호는 각각 블록캐리발생신호(G_{i-j}), 블록 캐리전파신호(P_{i-j}) 및 각 비트의 부분합 신호(ps_i)이다^[2]. 여기서 i 및 j 는 입력신호의 비트 위치를 나타낸다. 이와 구분하여 carry look-ahead adder에서 정의된 generate signal 및 propagate signal은 비트 발생신호(g_i)와 비트전파신호(p_i)가 된다.

16비트 ELM 구조에서는 16번째 비트 위치에서 블록캐리발생신호를 계산하는 G셀들을 일반적으로 그림 1-(a)와 같은 블록 다이어그램으로 나타낼 수 있다. 그림 1-(a)의 개략도는 그림 1-(b)와 같은 논리게이트로 실현될 수 있다. 여기서 각각의 신호를 설명하면 다음과 같다.

- g_{15} : 16번째 비트의 비트발생신호
($g_{15} = a_{15} \wedge b_{15}$)
- g_{14} : 15번째 비트의 비트발생신호
($g_{14} = a_{14} \wedge b_{14}$)
- p_{15} : 16번째 비트의 비트전파신호
($p_{15} = a_{15} \oplus b_{15}$)
- G_{13-12} : 13번째와 14번째 비트 사이의 블록 캐리발생신호
- G_{15-0} : 1번째와 16번째 비트 사이의 블록 캐리발생신호
- P_{15-8} : 9번째와 16번째 비트 사이의 블록 캐리전파신호

여기서 a_i , b_i 는 i 번째 비트의 입력신호를 나타내고 \wedge 는 AND 연산, \vee 는 OR연산, \oplus 는 XOR연산을 나타낸다. 기존의 16비트 구조에서 a_{15} 와 b_{15} 의 신호가 '1'이면, 1번째에서 16번째 비트 사이의 블록캐리발생신호(G_{15-0})는 나머지 신호에 상관없이 '1'이 된다. 마찬가지로 a_{14} 와 b_{14} 의 신호가 모두 '1'이고 $a_{15} \oplus b_{15} = '1'$ 이면, 블록캐리발생신호(G_{15-0}) 역시 나머지 신호

에 상관없이 '1'이 된다. 위의 두 경우 중에서, 첫 번째 경우는 $g_{15} = '1'$ 인 경우이고, 두 번째는 그림 1에서 첫 번째 레벨의 G셀 출력값이 '1'인 경우이다.

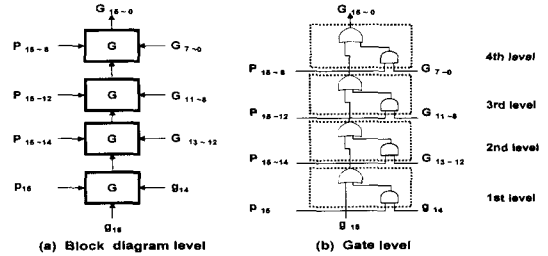


그림 1. 일반적인 16비트 ELM 덧셈기의 G셀 연결도
Fig. 1. G cell connections of a conventional 16bit ELM adder.

그림에서 보듯이 $g_{15} = '1'$ 이면 이 신호가 상위레벨의 G셀 안에 존재하는 OR게이트를 거쳐서 최종 출력(G_{15-0})에 나타난다. 마찬가지로 첫 번째 레벨의 G셀의 출력신호가 '1'이면 상위레벨의 G셀의 OR게이트를 통과하여 최종 출력으로 나타난다. 따라서 위의 두 경우는 블록캐리발생신호(G_{15-0})가 이미 g_{15} 또는 첫 번째 레벨의 G셀의 출력 값으로부터 결정되었음에도 불구하고 불필요하게 이 신호가 상위레벨의 G셀을 통과하면서 글리치를 발생시킴으로써 전력이 낭비된다.

본 연구에서는 ELM덧셈기를 구조적인 측면에서 저전력 실현을 위하여, 그림 2와 같이 특정 입력 비트 패턴이 들어오는 경우에 단축경로를 통과하여 직접 블록캐리발생신호를 계산함으로써 상위레벨에 위치한 G셀에서 발생하는 글리치를 제거시킨 구조를 제안하였다. 그림 2-(a)는 제안한 구조의 블록 다이어그램이며, 그 논리게이트는 그림 2-(b)와 같이 나타낼 수 있다. 여기서 g_{15}^{\sim} 는 active low 신호를 나타낸다.

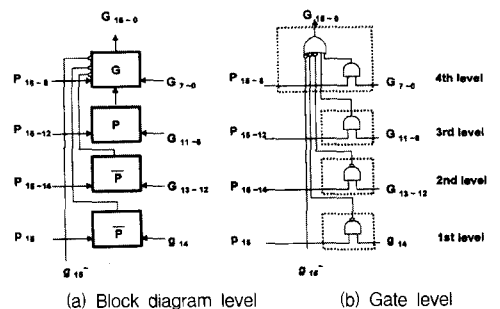


그림 2. 제안한 16비트 덧셈기의 G, P 및 P-bar 셀 연결도
Fig. 2. G, P and P-bar cell connections of a proposed 16bit adder.

그림 2에서 보면 앞의 예에서 첫 번째 경우인 16번째 비트발생신호인 g_{15} 가 액티브(active) 되면 그것이 바로 네 번째 레벨의 G셀에 연결되어 블록캐리발생신호(G_{15-0})를 '1'이 되게 한다. 마찬가지로 앞의 예의 두 번째 경우는 첫 번째 레벨의 \bar{P} 셀의 출력이 액티브 되어 바로 네 번째 레벨의 G셀에 연결되어 블록캐리발생신호(G_{15-0})를 '1'로 만든다. 제안한 덧셈기 구조는 블록캐리발생신호를 계산하는데 있어서 최상위 레벨은 G셀, 그 다음 레벨은 P셀, 그리고 나머지 하위 레벨은 \bar{P} 셀들이 위치한다. 여기서 g_{15} 및 \bar{P} 셀의 출력은 부논리(negative logic)를 나타내는데 그 이유는 만약 P셀이 정논리(positive logic)를 발생한다면, 마지막에 위치한 G셀에서 이 신호가 기존의 G셀에서 정논리 신호를 받는 PMOS가 Vdd와 직렬로 연결되어 지연시간이 상당히 증가하게 된다. 반면에 부논리를 사용하면, 이 부논리를 받는 NMOS가 접지에 직렬로 연결되기 때문에 동일한 트랜지스터 크기로 했을 경우 PMOS에 비해 이동도가 크기 때문에 상대적으로 지연시간이 줄어든다. 또한 P셀이 \bar{P} 셀로 바뀌므로 인버터가 하나 줄게 된다.

제안한 구조는 입력 비트 패턴을 고려하여 특정 입력 비트 패턴이 인가되었을 때, G셀에서 발생하는 글리치를 제거하여 전력소모를 줄인 구조이다. 반면에 비트 수가 증가하면 마지막 레벨의 G셀의 입력수가 증가하기 때문에 설계복잡도가 증가하고, 지연시간 측면에서 최악 조건이 들어왔을 때 마지막 레벨의 G셀의 pull down 시간이 증가되므로 지연시간이 증가하는 단점이 있다.

한편 논리레벨에서 전력소모를 줄이기 위해서는 적절한 논리형태의 선택이 필요하다. 그 동안 저전력에 적합한 여러 종류의 논리형태가 보고되었다^[5-13]. 논리형태는 크게 정적 CMOS, 동적 CMOS(dynamic CMOS), 전달 게이트(transmission gate) 및 조합형 논리형태 등으로 나눌 수 있다. 그 중에서 동적 CMOS 논리형태는 클럭에 의한 전력소모와 출력 노드의 천이 확률(transition probability)이 높아 저전력용으로는 적합하지 않다^[5]. 정적 CMOS는 DC 전류가 없고 높은 잡음여유(noise margin) 및 회로 구현이 용이한 장점이 있는 반면, 출력단 구동을 위한 인버터에 단락전류(short-circuit current)가 흐르는 단점이 있다. 전달 게이트는 트랜지스터의 소오스 및

드레인이 모두 신호 선에 연결되어 있으므로 신호가 전파하면서 단락 전류가 흐르지 않아 몇몇 논문에서는 정적 CMOS에 비해 전력소모가 적다고 주장하였다^[7-10]. 그런데 이 전달 게이트는 입출력이 분리(decoupling)되어 있지 않아 전류가 양방향으로 흐르고 출력단 구동력(driving capability)이 약하므로 완전한 논리형태를 이루기 위해서는 출력단에 인버터를 추가해야 한다. 또한 이전의 논문의 경우에 여러 종류의 논리형태의 비교기준이 되었던 것은 1비트 전가산기였는데 이 전가산기에는 XOR게이트가 두 개 포함된다. 그런데 XOR게이트는 정적 CMOS의 성능이 특히 떨어지는 게이트이므로 전가산기로 여러 가지 논리형태를 비교하는 것은 공정하지 않다는 사실이 보고되었다^[13]. 이 논문에서 여러 가지 논리형태들을 비교한 결과, 대부분의 게이트가 전달 게이트보다는 정적 CMOS가 성능이 우수함을 입증하였다^[13]. 한편 정적 CMOS로 구현한 XOR 게이트는 12개의 트랜지스터로 구성되는데^[13] Wang에 의해서 6개로 구성된 새로운 형태의 XOR/XNOR가 제안되었다^[14]. 이것은 정적 CMOS로 구현한 회로에 비해 트랜지스터 수가 1/2이므로 기생 캐패시턴스가 작아 전력소모가 적고 반전레벨(inversion level)이 작아 지연시간도 작다^[12]. 따라서 저전력 소모를 위한 논리형태는 개별 게이트 종류별로 성능이 우수한 게이트를 선택하여 사용하는 조합형 논리형태가 적합하다^[11-12]. 본 논문에서 조합형 논리형태는 정적 CMOS와 Wang의 XOR 게이트로 구성된다. 따라서 본 논문의 논리레벨에서는 정적 CMOS 논리형태와 조합형 논리형태로 각각 16비트 덧셈기를 구현하여 성능을 비교하였다.

한편 트랜지스터레벨에서는 트랜지스터 사이징(transistor sizing)이 전력소모에 큰 영향을 미친다. ELM 덧셈기의 특징 중의 하나는 레이아웃 형태가 하위비트에 있는 셀들의 논리깊이가 작아 상위 비트에 있는 셀들에 비해 지연시간이 작다는 점이다. 일반적인 동기 시스템에서 덧셈기의 지연시간은 임계경로(critical path)의 지연시간에 의존하므로 비임계경로에 위치한 셀의 트랜지스터 크기를 작게 해도 동작속도에는 영향을 주지 않는다. 여기서 셀의 위치에 관계없이 동일한 트랜지스터 크기를 갖는 셀을 고정 크기 셀이라 정의하고 비트 위치에 따라서 트랜지스터 크기를 다르게 한 셀을 가변 크기 셀이라 정의한다^[12].

가변 크기 셀의 트랜지스터 크기는 임계경로의 지연시간이 계산되면 그 시간을 넘지 않으면서 공정변화, 온도 및 전압변화에 따른 설계여유(design margin)를 고려하여 크기를 정하였다.

III. 실험 결과

회로구현에 앞서 제안한 덧셈기구조에서 각 비트의 경우에 단축경로를 통과하는 특정 입력 비트 패턴의 예와, 그 특정 입력 비트 패턴의 발생확률이 비트수의 증가에 따라서 어떻게 변화하는지를 알아보았다. 표 1은 제안한 8비트 덧셈기 경우에서, 단축경로를 통과하는 모든 특정 입력 비트 패턴 및 확률을 나타낸다. 표 1에서 보듯이 8비트 덧셈기에서 A와 B의 최상위 비트인 8번째 비트가 '1'이고 나머지 비트가 don't care 인 경우가 특정 입력 비트 패턴의 한 예이며, 그러한 비트 패턴의 발생확률은 전체 비트 패턴의 25%를 차지한다. 이와 마찬가지로 7번째 비트와 8번째 비트 패턴에 의한 특정 입력 비트 패턴의 발생확률은 12.5%이며 4, 5, 6, 7, 8번째 비트 패턴에 의한 발생확률은 1.56%이다.

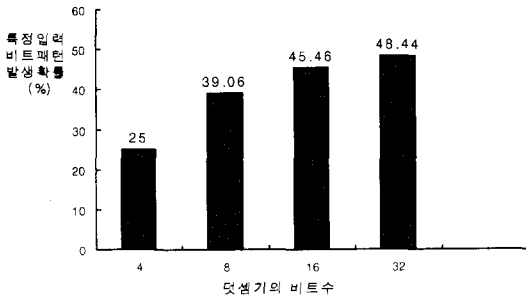


그림 3. 비트 확장에 따른 특정 입력 비트 패턴이 발생할 확률

Fig. 3. Probability of particular input bit patterns according to bit extension.

그림 3은 비트 확장에 따라서 특정 입력 비트 패턴이 발생할 확률을 나타낸다. 비트 확장에 따라서 특정 입력 비트 패턴의 발생확률은 점차로 증가되다가 최종적으로는 50%에 포화된다. 특정 입력 비트 패턴의 발생확률이 증가하면 G셀의 출력단에 글리치가 감소되므로 전력소모가 줄어든다. 비트 수가 증가할수록 모든 입력 비트 패턴에 대해서 특정 입력 비트 패턴이 발생할 확률이 증가하는 이유는, 비트 수가 증가할 수

록 여러 비트 위치에서 현재 블록의 블록캐리발생신호를 결정할 수 있기 때문이다. 비트 수가 확장되면 특정 입력 비트 패턴의 발생 확률이 50%에 포화되므로 비트 확장의 경우에도 전력절감 효과가 있음을 예상할 수 있다.

표 1. 제안한 8비트 덧셈기에서 단축 경로를 통과하는 특정 입력 비트 패턴들

Table 1. Particular input bit patterns to pass short paths in the proposed 8bit adder.

패턴 구분 \ 입력 비트		A	B	경우의 수	확률(%)
단축 경로 선택	8번째 비트패턴에 의해 G 신호가 결정되는 경우	1xxxxxxx	1xxxxxxx	128	250
	7,8번째 비트패턴에 의해 G 신호가 결정되는 경우	01xxxxxx 11xxxxxx	11xxxxxx 01xxxxxx	64 ² 64 ²	125
단축 경로 선택	4, 5, 6, 7, 8번째 비트 패턴에 의해 G 신호가 결정되는 경우	00001xxx	11111xxx	8 ²	1.56
		00011xxx	11101xxx	8 ²	
		00101xxx	11011xxx	8 ²	
		00111xxx	11001xxx	8 ²	
		01001xxx	10111xxx	8 ²	
		01011xxx	10101xxx	8 ²	
		01101xxx	10011xxx	8 ²	
		01111xxx	10001xxx	8 ²	
		10001xxx	01111xxx	8 ²	
		10011xxx	01101xxx	8 ²	
		10101xxx	01011xxx	8 ²	
		10111xxx	01001xxx	8 ²	
		11001xxx	00111xxx	8 ²	
		11011xxx	00101xxx	8 ²	
11101xxx	00011xxx	8 ²			
11111xxx	00001xxx	8 ²			
일반 경로 선택			39936	60.94	

G : 블록캐리발생신호
x : don't care

그림 4는 기존의 8비트 ELM 덧셈기 구조를 나타내며, 그림 5는 제안한 8비트 덧셈기를 나타낸다. 실제 모의실험에 사용한 덧셈기의 비트수는 16비트이며,

그림 6은 제안한 16비트 덧셈기에서 4번째 레벨의 G 셀을 CMOS회로로 나타낸 것이다. 각 트랜지스터 크기의 단위는 λ 이며, $\lambda=0.3\mu\text{m}$ 이다. 기존의 구조는 블록캐리발생신호를 계산하는 G셀이 레벨에 관계없이 동일한데 비해서, 제안한 구조에서의 G셀은 레벨에 따라 다른 특징이 있다.

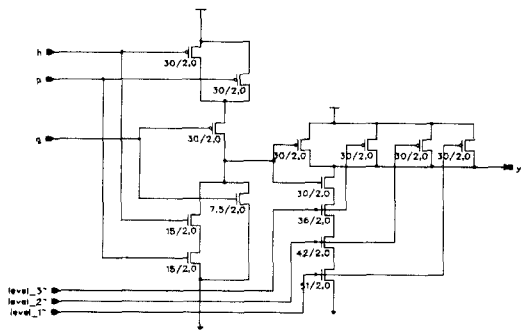


그림 6. 제안한 16비트 덧셈기의 네 번째 레벨의 G셀
Fig. 6. G cell at the fourth level of a proposed 16bit adder.

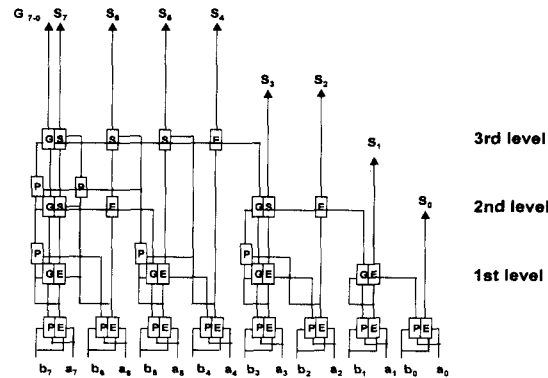


그림 4. 기존의 8비트 ELM 덧셈기 구조
Fig. 4. Conventional 8bit ELM adder architecture.

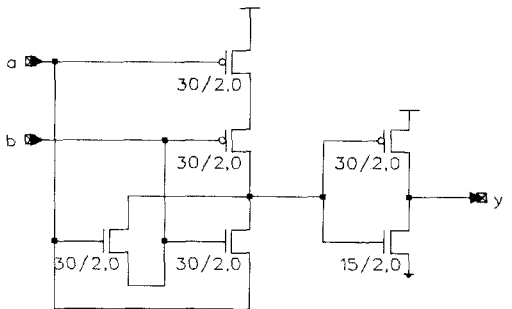


그림 7. 고정크기 셀로 구현한 E 셀
Fig. 7. E cell with fixed-sized cell.

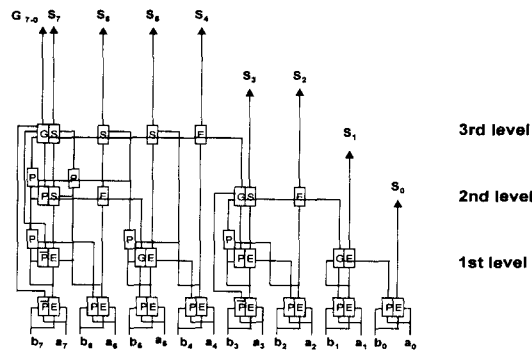


그림 5. 제안한 8비트 ELM 덧셈기 구조
Fig. 5. Proposed 8bit ELM adder architecture.

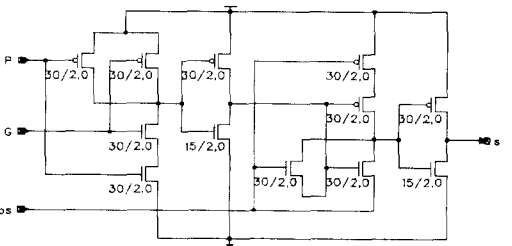


그림 8. 고정크기 셀로 구현한 S 셀
Fig. 8. S cell with fixed-sized cell.

그림 6에서 level₁~, level₂~ 및 level₃~는 active low 신호로써 level₁~은 첫 번째 레벨에서, level₂~는 두 번째 레벨에서, level₃~는 세 번째 레벨에서 직접 받는 신호이다. 그림 6과 같은 방법으로 하위레벨의 G셀들을 CMOS회로로 구현할 수 있다. 상위레벨에 위치한 G셀 일수록 입력수가 증가하는 이유는, 레벨이 증가할수록 하위레벨의 P셀 및 \bar{P} 셀의 수가 증가하기 때문이다.

그림 7은 Wang의 XOR게이트를 나타낸 것으로 본 논문에서는 Wang의 XOR게이트와 정적 CMOS로 구성된 조합형 논리형태로 16비트 덧셈기를 구현하였다.

그림 7과 그림 8은 고정 크기 셀로 구현한 E 및 S 셀을 나타내며 그림 9와 그림 10은 가변크기 셀로 구현한 E 및 S셀을 나타낸 것이다. 16비트 ELM 덧셈기에서 가변 크기 셀의 경우에 크기를 작게 한 셀의 종류는 E 및 S셀이며 개수는 총 18개가 해당된다. 가변 크기 셀의 트랜지스터 크기는 공정변화, 온도 및 전원전압 변화 등 설계여유를 고려하여 고정 크기 셀에 비해 크기를 1/2로 하였다.

0.6 μ m 단일폴리, 삼중금속 LG CMOS 공정변수를 가지고 16비트 덧셈기를 HSPICE를 이용하여 모의 실험을 하였다. 동작전압은 3.3V, 부하는 50fF, 온도는 25 $^{\circ}$ C, 입력신호의 상승/하강시간은 1ns, 입력신호의 주파수는 100MHz, 표준공정의 모델변수를 사용하였다. 모델 파라메타의 문턱전압은 바디 효과(body effect)가 없는 경우에 NMOS는 0.8V, PMOS는 -0.859V이다. 지연시간 계산을 위한 입력 조건은 캐리 입력이 모든 레벨의 G셀을 통과하여 마지막 캐리 출력으로 나타나는 경우이다. 16비트 덧셈기의 실험 결과는 표 2에 나타나 있다. 제안한 구조가 기존의 구조에 비해서 지연시간이 증가한 이유는 각 레벨의 G 셀이 출력이 '0'이 되는 경우에, 기존 구조의 G셀은 하나의 NMOS를 방전하는 것에 비해 제안한 구조의 G셀은 여러 개의 NMOS를 방전하기 때문이다. 트랜지스터레벨에서 가변 크기 셀을 사용한 경우에 지연시간이 줄어든 이유는, 가변 크기 셀의 일부가 임계경로에 연결되어 트랜지스터 크기가 감소되면서 임계경로의 부하 캐패시턴스의 값이 감소되었기 때문이다.

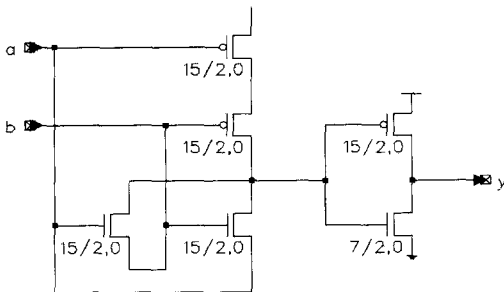


그림 9. 가변크기 셀로 구현한 E 셀
Fig. 9. E cell with variable-sized cell.

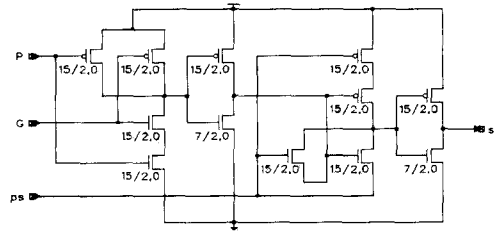


그림 10. 가변크기 셀로 구현한 S 셀
Fig. 10. S cell with variable-sized cell.

전력소모의 계산방법은 HSPICE 기능중에 measure문장을 사용하여 RMS(root mean square) 전력 값을 취하였다. 본 논문에서 제안한 구조는 비트 패턴에 따라서 전력소모 절감 효과가 나타나므로 다양한 입력 벡터를 주어야 한다. 모의 실험은 25개의 입력 벡터를 무작위로(randomly) 준 다음, 그것들의 평균을 취하여 전력 소모를 계산하였다. 표 2에서 보듯이 전체 전력소모 절감 비율에서 구조 측면만을 비교했을 때 기존의 구조에 비해서 제안한 구조가 9.8%, 논리형태만을 비교했을 때 정적 CMOS에 비해 조합형 논리형태가 5.3%, 비임계 경로의 트랜지스터 다운사이징(down sizing)이 10.5%의 전력소모 절감을 보였다. 전체적으로 기존의 구조와 정적 CMOS 및 고정 크기 셀로 구현한 16비트 덧셈기에 비해 제안한 구조와 조합형 논리 형태 및 가변 크기 셀로 구현한 16비트 덧셈기가 지연시간이 1.3% 증가하였으나 전력소모면에서는 23.6% 절감을 가져왔다. 표 3은 참고문헌^[4, 12]들과의 모의 실험 결과를 비교한 표이다. 전력소모는 전원전압과 동작 주파수이외에도 사용한 모

표 2. 16비트 덧셈기의 실험 결과
Table 2. Simulation result of 16bit adder.

Architecture level	Conventional	Proposed	
Logic level	static CMOS	static CMOS	combination of logic style
Transistor level	fixed-sized cell	fixed-sized cell	fixed-sized cell variable-sized cell
Propagation delay(ns)	2.97	3.21	3.25 3.01
Power consumption(mW)	17.34 @100MHz, 3.3V	15.64 @100MHz, 3.3V	14.81 @100MHz, 3.3V 13.25 @100MHz, 3.3V
PDP(pJ)	51.50	50.20	48.13 39.88
No. of transistors	1048	1028	734 734
Power reduction(%)	-	9.8	14.5 23.6

델 파라메타, 입력 벡터의 형태 및 모의실험 조건에 따라 크게 달라지기 때문에 참고문헌들과의 직접적인 비교가 어려웠다. 참고 문헌^[12]와의 전력소모 비교는 표 2를 참고로 하여 트랜지스터레벨과 논리레벨에서의 접근방법이 동일하고 덧셈기의 구조가 다르기 때문에, 동일한 실험조건 가정하에서는 약 10%의 전력절감이 있음을 알 수 있다.

IV. 결론

저전력 실현을 위하여 16비트 덧셈기를 구조, 논리 및 트랜지스터레벨에서 설계하였다. 기존의 ELM 덧셈기 구조는 모든 입력 비트 패턴에 대해서 동일한 신호 전파 경로를 갖기 때문에 특정 입력 비트 패턴이 인가되었을 때, G셀에서 글리치의 발생으로 인한 부가의 전력소모를 나타내는 단점이 있다. 본 논문에서는 이러한 특정 입력 비트 패턴들은 단축경로를 통과함으로써 G셀에 의한 글리치 발생을 제거하고 기존의 구조보다 빨리 블록캐리발생신호를 계산하는 저전력 구조를 제안하였다. 제안한 구조에서 특정 입력 비트 패턴이 발생할 확률은 4비트 경우에 25%, 8비트 경우에 39.06%, 16비트 경우에 45.46%로 나타나며 계속적으로 비트 수가 증가하면 최종적으로는 50%에 포화된다. 비트 수가 확장되면 특정 입력 비트 패턴의 발생 확률이 50%에 포화되므로 비트 확장의 경우에도 전력 절감 효과가 있음을 예상할 수 있다. 반면에 제안한 구조는 기존 구조에 비해 비트 수가 증가하면서 마지막 레벨의 G셀의 입력 수가 증가하므로, 설계규칙성이 떨어지고 지연시간이 증가하는 단점이 있다. 또한 저

전력 소모를 위해 논리레벨에서는 정적 CMOS와 저전력 XOR 게이트로 구성된 조합형 논리형태를 사용하였고, 트랜지스터레벨에서는 논리깊이가 작은 하위비트에 대해서는 상위비트에 위치한 셀보다 상대적으로 트랜지스터 크기를 작게 한 가변 크기 셀로 덧셈기를 구현하였다.

0.6 μ m 단일폴리 삼중금속 LG CMOS 표준 공정변수를 가지고 16비트 덧셈기를 모의 실험한 결과, 고정 크기 셀과 정적 CMOS 논리형태 및 기존의 ELM 구조로 구현한 덧셈기에 비해, 본 논문에서 제안한 덧셈기가 지연시간은 1.3% 증가하였으나 전력소모 면에서는 구조레벨에서 9.8%, 논리레벨에서 5.3%, 트랜지스터레벨에서 10.5%가 각각 감소되어 전체적으로 23.6%의 전력소모가 감소되었고 power-delay-product면에서는 22.6%가 감소되었다. 제안한 덧셈기는 저전력 마이크로 프로세서나 DSP(Digital Signal Processor)의 덧셈기 블록에 응용이 가능하리라 생각된다.

참고 문헌

- [1] R. P. Brent and H. T. Kung, "A regular layout for parallel adder," IEEE Transactions on Computers, vol. C-31, pp. 260-264, March, 1982.
- [2] T. P. Kelliher, R. M. Owens, M. J. Irwin, and T. T. Hwang, "ELM-A Fast Addition Algorithm Discovered By a Program" IEEE Transactions on Computers, vol. 41, no. 9, September,

표 3. 참고 문헌들과의 모의실험 결과 비교표
Table 3. Comparison of simulation result with references.

	Reference [4]	Reference [12]	Proposed
Bit	16	16	16
Adder architecture	conventional ELM	conventional ELM	Proposed adder
Model parameter	1.2 μ m MOSIS	0.8 μ m LG	0.6 μ m LG
Supply voltage(V)	5.0	5.0	3.3
Temperature(°C)	27	70	25
No. of transistors	892	언급 없음	734
Propagation delay(ns)	8.0	2.9	3.01

- 1992.
- [3] B. W. Y. Wei and C. D. Thompson, "Area-time optimal adder design", IEEE Transactions on Computers, vol. 39, pp. 666-675, May, 1990.
- [4] C. Nagendra, M. J. Irwin, and R. M. Owens, "Area-Time-Power Tradeoffs in Parallel Adders" IEEE Transactions on Circuits and Systems, vol. 43, no. 10, October, 1996.
- [5] Abdellatif Bellaouar, Mohamed I. Elmasry, "Low Power Digital VLSI Design", Kluwer academic publishers, pp. 176-187, 1995.
- [6] Anantha P. Chandrakasam, Robert W. Brodersen, "Low Power Digital CMOS Design", Kluwer academic publishers, pp. 55-102, 1995.
- [7] K. Yano, et al, "A 3.8ns CMOS 16x16 Multiplier Using Complementary Pass-Transistor Logic", IEEE J. of Solid-state Circuits, vol. SC-25, no. 2, pp. 388-394, April, 1990.
- [8] M. Suzuki, et al., "A 1.5ns 32-b CMOS ALU in Double Pass-Transistor Logic", IEEE J. of Solid-state Circuits, vol. SC-28, no. 11, pp. 1145-1151, November, 1993.
- [9] A. Parameswar, H. Hara, and T. Sakuri, "A swing restored pass-transistor logic-based multiply and accumulate circuit for multimedia application," IEEE J. of Solid-state Circuits, vol. 31, pp. 805-809, June, 1996.
- [10] M. Song, et al., "Design methodology for high speed and low power digital circuits with economized pass-transistor logic (EEPL)", in proc. 22nd European Solid-State Circuits Conf., pp. 120-123, September, 1996.
- [11] 김문수, 유범선, 강성현, 이중석, 조태원, "하이브리드 로직 스타일을 이용한 저전력 ELM 덧셈기 설계", 전자공학회 논문지, 제35권 C편, 제6호, pp. 389-396, 1998
- [12] 유범선, 조태원 "가변크기 셀을 이용한 저전력 고속 16비트 ELM 가산기의 설계", 전자공학회 논문지, 제35권 C편, 제8호, pp. 34-41, 1998
- [13] Reto Zimmermann and Wolfgang Fichtner "Low power logic style : CMOS versus pass transistor Logic", IEEE J. of Solid-state Circuit, vol. 32, no. 7, July, 1997.
- [14] J. M. Wang, S. C. Fang, and W. S. Feng, "New Efficient Designs for XOR and XNOR Functions on the Transistor level" IEEE J. of Solid-state Circuits, vol. s29, no. 7, July, 1994.

저 자 소 개



柳 凡 善(正會員)

1991년 2월 충북대학교 전자공학과 졸업(공학사). 1997년 2월 충북대학교 대학원 전자공학과 졸업(공학석사). 1997년 3월 ~ 현재 충북대학교 대학원 전자공학과 박사과정. 주관심분야는 저전력 회로설계, 컴퓨터 구조, 집적회로 설계임

터 구조, 집적회로 설계임



趙 泰 元(正會員)

1973년 2월 서울대학교 전자공학과 졸업(공학사). 1986년 5월 미국 루이빌대 전자공학과 졸업(공학석사). 1992년 5월 미국 켄터키 주립대 전자공학과 졸업(공학박사). 1973년 8월 ~ 1983년 10월 금성전선(주). 1992년 9월 ~ 현재 충북대학교 전기전자공학부 부교수. 주관심분야는 집적회로 설계, 컴퓨터 구조, 저전력 회로 설계, DSP core 설계임

1992년 9월 ~ 현재 충북대학교 전기전자공학부 부교수. 주관심분야는 집적회로 설계, 컴퓨터 구조, 저전력 회로 설계, DSP core 설계임



李 奇 榮(正會員)

1981년 2월 서울대학교 전자공학과 졸업(공학사). 1983년 8월 서울대학교 대학원 전자공학과 졸업(공학석사). 1993년 8월 서울대학교 대학원 전자공학과 졸업(공학박사). 1983년 8월 ~ 1984년 11월 LG 반도체(주). 1987년 2월 ~ 현재 충북대학교 전기전자공학부 교수. 주관심분야는 반도체소자, 잡음해석, 집적회로 설계임

1987년 2월 ~ 현재 충북대학교 전기전자공학부 교수. 주관심분야는 반도체소자, 잡음해석, 집적회로 설계임