

논문 2008-45SD-3-11

멀티모드 이동 통신 모뎀을 위한 전력 효율적 다기능 캐리스킵 가산기

(A Power Efficient Versatile Carry Skip Adder Architecture for the
Multimode Mobile Modem)

한 태 희*

(Tae Hee Han)

요 약

다양한 무선 통신 표준 규격을 수용하는 멀티모드 단말기 모뎀은 가변적인 워드 길이와 광범위한 데이터율을 처리할 수 있는 다기능 산술 연산 회로를 필요로 한다. 일반적으로 이런 목표를 위한 하드웨어는 요구되는 최고 성능을 달성하도록 설계되어지므로 종종 전력 소모 측면에서 낭비적인 요소가 있으며 특히 낮은 데이터율에서 심화되는 경향이 있다. 따라서 동작 속도와 전력에 적합한 산술 연산 회로는 무선 통신 응용 분야에서 매우 필수적인 요소이다. 본 논문에서는 계층적인 캐리스킵 로직을 기본 구성 블록으로 하는 전력 효율적 다기능 가산기 구조를 제안하고 다양한 동작 모드에서의 면적, 성능, 전력 효율을 보임으로써 효과를 입증하였다.

Abstract

The multi-mode terminal modem which is capable of accommodating a variety of wireless communication standards needs versatile arithmetic units for processing a variety of word lengths and wide range of data rates. Since the target hardware is usually designed to meet the required highest performance, it is often wasteful in power consumption especially when low rate data processing cases. Thus, a speed and power adaptability of the arithmetic unit is a desirable feature for the wireless applications. In this paper, we propose a power efficient versatile adder architecture with carry skip logic as a basic building block constructed in hierarchical manner. The validity of the architecture is shown with respect to size, performance, and power efficiency in diverse operating modes.

Keywords: power efficient, multimode modem, carry skip adder, hierarchical structure

I. 서 론

무선 이동 통신용 모뎀에서 음성 통화 시 필요한 데이터 처리율은 9.6~14.4 kbps이고 심볼 워드 길이는 4 비트 정도이나, 광대역 고속 데이터 처리를 목표로 하는 CDMA2000, W-CDMA/HSPA, IEEE 802.16e 및 802.11a/b/g/n 규격들은 수십 Mbps의 데이터 처리율을 요구하는 동시에 심볼 워드 길이도 6비트에서 10비트 이상 넓은 영역에 걸쳐 있다. 또한, 음성 신호의 경우

프레임의 길이가 10ms~20ms 범위이며 모뎀 하드웨어 동작 속도가 10MHz 내외에 비해, 광대역 고속 데이터의 프레임 또는 패킷은 수 ms 또는 그 이하의 매우 짧은 길이와 버스티한 트래픽 특성을 가지고 있고, 수십 MHz에서 수백 MHz의 하드웨어 동작 속도를 필요로 한다^[1]. 소프트웨어 기반의 프로그래머블 프로세서를 이용할 경우 클럭 주파수는 더욱 증가하고 결국 전력 소모 제한에 당면한다.

최근 무선 이동 통신 단말 모뎀의 추세는 다양한 통신 표준을 하나의 기저대역 프로세서로 처리하는 것을 목표로 하므로, 가변 데이터율과 워드 길이를 갖는 디지털 신호 처리 하드웨어를 성능, 면적, 전력소모 측면

* 평생회원, 성균관대학교 정보통신공학부
(School of Information and Communication
Engineering, Sungkyunkwan University)
접수일자: 2007년7월20일, 수정완료일: 2008년2월13일

에서 최적화 설계하는 것이 필수적이다. 이러한 구현 목표를 달성하기 위해서는 제한된 하드웨어 자원을 최대한 공유하고 필요 동작 시에만 활성화시키는 유연성 있는 제어 방식이 요구된다. 특히, 데이터 변복조 기능을 소프트웨어적으로 처리하면서 오류 발생시 손쉽게 수정할 수 있는 것을 목표로 하는 SDR(Software Defined Radio)기술에서는 초저전력 고성능 프로그래머블 DSP 프로세서를 이용하고자 하는 경향이 두드러지고 있다. 이 경우 앞서 언급한 특수 목적의 가변적 데이터 신호 처리가 일반화된다. 그 동안 발표된 SDR을 위한 DSP 프로세서의 경우 32비트 SIMD(Single Instruction and Multiple Data stream) 방식의 하드웨어 연산장치를 핵심 자원으로 하며 다중 코어로 병렬 처리하는 추세이다. 이러한 모델의 디지털 신호 처리 요구 사항을 만족시키기 위한 하드웨어 연산기의 조건은 다음과 같다.

- 가변 워드 길이(비트 폭)를 갖는 데이터 연산 처리 가능.
- 데이터율과 워드 길이에 따라 성능도 가변적이고 확장성이 있어야 함. (즉 비트 폭이 넓은 고속 데이터 처리 시에는 동작 속도를 증가시킬 수 있는 구조)
- 최대 데이터 비트 폭보다 작은 여러 개의 데이터를 병렬 처리 가능하도록 하고 각각의 병렬 처리 데이터에서 균일한 속도로 동작 가능.
- 전력 소모를 최소화할 수 있는 구조

각각의 조건들은 독립적인 측면도 있으나 면적 및 전력 소모 최적화를 위한 상관관계가 존재한다. 32비트 혹은 그 이상의 워드 길이를 처리할 수 있는 하드웨어 블록이 상기 조건을 만족시키기 위해선 여러 개의 작은 데이터 워드 단위로도 동작할 수 있어야 한다. 즉 32비트 연산기라면 단일 32비트 워드 연산, 2개의 16비트 연산, 4개의 8비트 연산, 8개의 4비트 연산 등이 가능해야 확장성과 활용도 측면에서 효과적이다.

본 논문에서는 이러한 가변 길이 워드 데이터의 병렬 처리가 가능할 뿐 아니라, 하드웨어를 계층적으로 활성화하여 성능을 조절할 수 있는 전력 효율적 가산기 구조를 제안하였다. CMOS 디지털 회로에서 성능(동작 속도)은 공급 전압에 비례하고, 회로의 에너지/전력 소모는 공급 전압의 제곱에 비례하므로 성능 조절을 통해 필요한 결과를 마감 시간에 맞추어 완료하는 'Just-in-Time Computing' 원리를 적용하면 에너지/전력 효율화

를 극대화할 수 있다. 이러한 저전력 효율화 기법은 1990년대 초 중반 Brodersen과 Chandrakasan에 의해 체계적으로 정리 발표된 바 있다^[2~4]. 이를 계기로 소위 '멀티 코어 아키텍처', 즉 하드웨어 병렬화를 통한 전력 효율적 성능 개선이 오늘날 마이크로프로세서, DSP 등에서 주류 기술로 자리 잡고 있다.

본 논문의 구성은 다음과 같다. II장에서는 제안하는 연산기의 기본 구성 블록인 캐리스킵가산기에 대해 설명 분석하고, III장에서는 가변적 동작 성능과 워드 길이에 적합한 전력 효율적 가산기 구조를 제시한다. IV장에서는 제안된 구조의 면적, 성능, 전력 효율에 대해 분석하여 효과를 입증하며 V장에서 결론을 맺도록 한다.

II. 기본 구성 블록과 분석

덧셈은 모뎀을 포함한 대부분의 디지털 신호 처리 영역에서 가장 많이 사용되는 연산이다. 사칙 연산과 절대값 계산, 증가/감소 등은 모두 가산기를 기반으로 구현되므로 이에 대한 최적화는 매우 중요하다. 가산기 구현 방식으로는 리플 캐리(Ripple Carry), 맨체스터 캐리 체인(Manchester Carry Chain), 캐리 스킵(Carry Skip), 캐리 예견(Carry Look-ahead), 캐리 선택(Carry Select), 그리고 다양한 병렬 캐리 프리픽스(carry prefix) 등이 존재한다^[3~5].

리플캐리 가산기(Ripple Carry Adder: 이하 RCA)는 가장 단순한 구조인 반면 속도 역시 가장 낮다. n -비트 RCA는 $O(n)$ 면적과 $O(n)$ 지연 시간을 보인다, 발전된 형태인 캐리 예견 가산기(Carry Look-ahead Adder: 이하 CLA), 캐리 선택 가산기(Carry Select Adder: 이하 CSLA)와 병렬 캐리 프리픽스 가산기(Parallel Carry Prefix Adder: 이하 PCPA)는 $O(n \log(n))$ 면적과 $O(\log(n))$ 지연 시간을 갖는다. 이중 CLA와 CLA로부터 파생된 PCPA는 캐리 예견 로직을 갖고 있어 가장 빠른 성능을 보이지만, 로직 증가와 더불어 불규칙한 회로 구조로 인해 반도체 레이아웃 면적상의 손실을 동반한다. PCPA는 캐리 예견 회로의 규칙성을 좀 더 체계화하였지만 물리 구현 수준의 회로 배치와 배선 연결에 있어 설계자의 개입을 필요로 한다. 이런 이유로 인해 $O(n)$ 면적과 $O(n^{1/m})$: m 은 계층 수 지연시간을 갖는 캐리스킵가산기(Carry Skip Adder: 이하 CSkA)가 단순하고 규칙적인 레이아웃 구조 등의 추가적인 장점으로 인해 최근 많이 사용되고 있다^[6~10]. 또한 VHDL이

나 Verilog 같은 하드웨어 설계 언어를 통한 구현 및 로직 합성에도 용이할뿐더러 비교적 간단한 구성과 연결 배선의 규칙성으로 인해 캐리 예견 가산기나 캐리선택가산기보다 면적 및 전력지연시간 곱에서도 우수한 것으로 나타나있으며^[2-6], 규칙적인 회로 분할과 계층화를 통한 복수개의 워드 병렬 처리에도 적합한 특성을 가지고 있다.

1. 캐리스킵가산기의 구조 및 동작

두 개의 n -비트 입력 벡터 X, Y 를 갖는 가산기의 부울 대수적 표현은 널리 알려진 대로 아래 식(1)~(4)과 같다.

$$X = x_{n-1}x_{n-2} \cdots x_1x_0, Y = y_{n-1}y_{n-2} \cdots y_1y_0 \quad (1)$$

$$g_i = x_i y_i, p_i = x_i \oplus y_i \quad (2)$$

$$s_i = x_i \oplus y_i \oplus c_i \quad (3)$$

$$c_{i+1} = x_i y_i + y_i c_i + x_i c_i \quad (4)$$

캐리 비트의 재귀적 성격에 의해 비트 수가 늘어날수록 논리식이 복잡해지고 계산에 시간이 많이 소요된다. 캐리 예견 회로는 생성항 g 와 캐리 전파항 p 를 이용해 구성한다.

그림 1에 보인 바와 같이 캐리스킵 가산기의 기본 아이디어는 각 비트 별로 병렬 생성 가능한 캐리 전파항들의 논리곱이 참이면, 최초 입력 캐리 비트(c_{in})가 직접 최종 출력 캐리로 전달되는 것이다. 이때 블록 내부 캐리(그림1에서 c_3)는 최초 입력 캐리와 독립적으로 생성되므로 리플캐리 가산기와 구분되는 속도 향상의 기회가 있다. 4비트 단위의 캐리스킵 가산기를 16비트로 확장하여 이때 최대 지연 시간 경로를 보인 것이 그림 2이다. 최대 지연 시간은 3개 부분으로 구분할 수 있다:

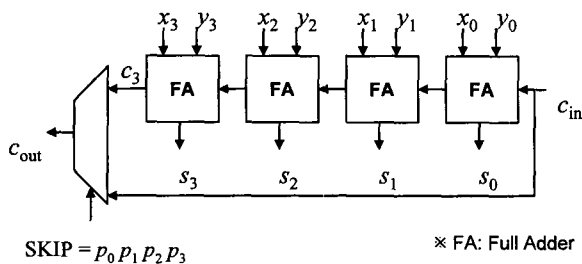


그림 1. 4비트 캐리스킵가산기 구조
Fig. 1. 4-bit carry skip adder.

1) 캐리 생성 시간, 2) 캐리 스킵 시간, 3) 캐리 흡수 시간^[8]. 이때 최대 지연 시간은 그림 2에 굵은 점선으로 표시된 경로를 통과하는 경우로 N 을 워드의 비트 길이, B 를 블록 그룹 크기(여기서는 4)라 했을 때 다음과 같이 주어진다.

$$T_{ADD} \cong B \cdot ((N/B)-2) \cdot T_{skip} + (B-1) \cdot T_{carry} + T_{sum} \quad (5)$$

이때 T_{carry} 는 단위 비트에서 캐리 지연 시간, T_{skip} 은 스킵회로 지연시간, T_{sum} 은 마지막 비트에서의 합 계산 시간이다. 가장 긴 캐리 입력 신호를 통한 전달 시간 동안 각각의 단위 블록들은 병렬적으로 캐리 전파항과 내부 캐리를 연산할 수 있기 때문에 지연 시간이 흡수되는 것으로 이해할 수 있다^[5].

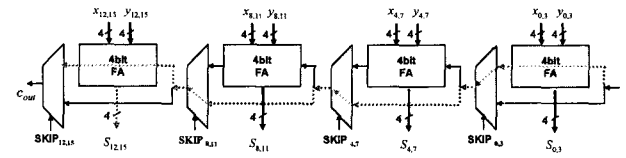


그림 2. 16비트 캐리스킵가산기의 최대 지연시간 경로
Fig. 2. The longest delay path of 16-bit carry skip adder.

2. 성능 개선 및 전력 효율 향상 기법

2.1에서 소개한 캐리스킵 가산기의 경우 성능 향상, 즉 동작 속도 개선을 위해 계층적 구조와 가변 블록 크기 기법 등이 제시되어 있다^[10-11]. 캐리스킵가산기는 전가산기가 블록으로 그룹핑되어 구성되므로 각 블록에서의 전가산기 개수가 전체 성능에 영향을 주며, 하나 또는 그 이상의 계층을 통해 설계하기 용이한 장점이 있다. 가변블록크기 캐리스킵가산기의 경우 그림 3에 보인 바와 같이 최초 캐리 전달 시간을 고려해 지연 시간을 최적화한 구조로 캐리 전달항 p 의 생성 시간과 비트 별 캐리 지연 시간을 각각 ΔT 로 가정하였을 때, 16 비트 연산에 있어 대략 RCA에 비해 53%의 속도 개선(RCA: $17\Delta T$, 그림3: $8\Delta T$)을 기대할 수 있다. 추가적인 성능 향상을 위해 계층적인 캐리스킵회로를 구현할 경

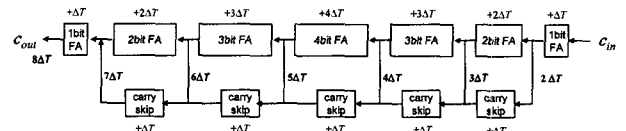


그림 3. 16비트 가변 블록 크기 캐리스킵 가산기
Fig. 3. 16-bit variable size carry skip adder.

표 1. 캐리스킵 가산기의 상대적 지연 시간
Table 1. Relative delay time of the carry skip adder.

워드길이 (비트 폭)	RCA지연시간	1계층 고정블록 크기 CSkA 지연시간	1계층가변블록 크기 CSkA 지연시간 [각 블록별 비트 수 분할]
8	9ΔT	7ΔT = 2+2+3 (2비트 블록4개)	6ΔT [2 3 3 1]
16	17ΔT	9ΔT = 1+4+4 (3비트 5개, 1비트 1개)	8ΔT [1 2-3 4 3 2-1]
32	33ΔT	15ΔT = 4+6+5 (4비트 블록8개)	12ΔT [1 2 2-3 4 5 5 4 3 2 1]
64	65ΔT	20ΔT = 4+9+7 (6비트 블록10개, 1비트 블록 4개)	16ΔT [1 2 3 4 5 6 7 6 5 4 3 2 1]

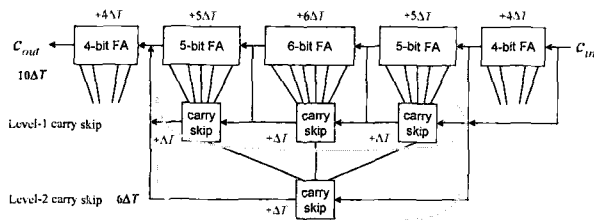


그림 4. 24비트 2-계층 가변 블록 크기 캐리스킵 가산기
Fig. 4. 24-bit two-level variable size carry skip adder.

우 그림 4에 보인 바와 같이 추가적인 이득이 존재한다. m 개의 캐리스킵로직 계층을 갖는 캐리스킵가산기를 구현할 경우 지연시간은 근사적으로 $O(n^{1/m})$: m 은 계층 수)의 관계가 있다^[10].

표 1은 8비트에서 64비트에 걸친 워드 길이에 따라 1비트 전가산기에서의 캐리 지연 시간을 ΔT 로 가정했을 때 근사화된 지연 시간 모델이다. 실제 VLSI 구현 시에는 여러 가지 다른 변수가 존재하나 이런 단위 지연 시간 모델링을 통해 대략적인 경향은 추정할 수 있다.

그림 3과 4에 보인 바와 같이 캐리스킵가산기는 블록 별로 모듈화된 구조와 계층화된 캐리스킵로직을 통해 확장에 유리한 규칙성과 대칭성을 가지고 있으며, 3장에서 제안할 계층적 활성화 로직을 통해 전력 효율을 제고할 수 있다. 동적 전력 소모 P 는 동작 활성화율 α , 캐퍼시턴스 C , 동작 전압 V 의 제곱 및 동작 주파수 f 에 비례하므로 ($P \propto \alpha CV^2f$) 최적화를 위해서는 각각의 요소를 최소화해야 한다. 동작 활성화율을 줄이기 위해서는 필요한 연산을 필요한 시간 동안만 동작하도록 제어하고, 커패시턴스를 줄이기 위해선 하드웨어의 면적과 인터커넥션의 길이를 최소화해야 한다.

전력 소모에 있어 전압은 제공에 비례하는 항이므로

가장 큰 영향력을 가진 요소이다. 동적인 전력 감소 기법으로는 매우 보편적인 클럭 게이팅과, 고도로 복잡한 동적 전압 스케일링, 다중 전압/전력 영역 분할 등 다양한 방법이 제시되어있다. 대기 시간 및 초미세 반도체 공정에서 특히 중요한 누설 전류 제어와 관련된 저전력 기법들도 있으나, 여기서는 동작 상태에서 회로의 스위칭과 관련된 동적 전력 감소에 집중하고자 한다.

동적 전력 제어는 고효율의 DC-DC 변환기와 성능 요구에 따른 전압 제어를 위한 복잡한 하드웨어 및 소프트웨어를 수반하기 때문에 범용 마이크로프로세서 등에 제한적으로 사용되고 있다. 이보다 완화된 방식으로 고정된 몇 가지 이산적인 공급 전압원을 이용할 수도 있다. 적응적 다중 전압 및 전력 영역으로 VLSI 회로를 분할하여 요구 성능에 따라 다른 전압을 공급하거나 아예 동작하지 않는 시간 구간 동안 전원이나 클럭을 차단하는 방법이다. 이때 회로 분할의 기준은 특정 시간의 동작 여부와 회로 활성화율, 요구 성능에 따른다. 그러나 이런 방식들도 여전히 회로 설계와 동작 검증이 매우 복잡한 편에 속하므로, 하드웨어 자체만으로 스위칭 동작을 최소화하고 활성화를 제어할 수 있는 방법이 저비용 구현 측면에서 더 유리하다.

III. 제안된 가산기 하드웨어 구조

캐리예전 가산기는 비트 수 증가에 따른 지연 시간 증가가 로그 함수적 특성을 지니고 있어 최적의 성능을 달성할 수 있다. 그러나 각각의 비트에서 캐리 생성량과 캐리 전달량을 필요로 하고 반복적 계산을 통해 출력 캐리량을 만들어내기 때문에 게이트의 입력 단자 수 (Fan-In)와 구현 로직이 급격하게 증가한다. 결과적으로 가장 많은 하드웨어 자원을 소비하며 데이터 경로 구성이 불규칙하여 연결 배선도 상당히 증가한다. 통상적으로 4비트 단위의 블록으로 분할 설계하며 블록단위의 캐리예전로직을 계층적으로 추가하는 구조를 취한다. 이때 캐리예전로직은 상호 의존적이므로 독립적으로 적용하기 어렵다. 이에 비해, 캐리 스킵 가산기는 성능 면에선 캐리예전가산기에 미치지 못하나, 캐리 전파항만을 필요로 하여 복잡도가 높지 않으며, 성능 향상을 위해 가변 크기 블록이나, 다단계 스킵로직을 독립적으로 적용할 수 있어 계층적 구조 설계도 용이하므로 요구 성능 대비 구현 비용 최적화에 보다 근접할 수 있다.

1. 전체 구조 및 동작 모드

무선 이동 통신용 모뎀에서 많이 사용되는 상관기 기반의 레이크 수신기나 등화기, 비터비/터보 복호기, FIR 필터, 푸리에 변환기 등은 일반적으로 4에서 16비트 사이의 워드 길이를 가지며, 병렬 처리가 필요하다. SDR을 지향하는 DSP 프로세서의 경우 최대 32비트 연산을 지원하는 경우가 대부분이다. 부동소수점 연산을 지원하는 경우 64비트 확장성의 필요가 있으나 일반적으로 통신 신호 처리 연산은 고정소수점으로 처리되기 때문에 여기서는 최대 32비트 워드 길이를 지원하도록 하였다^[12]. 또한, 동일한 32비트 하드웨어 자원을 이용해 2개의 16비트 연산, 4개의 8비트 연산, 8개의 4비트 연산 등이 동시에 병렬 처리 가능하도록 설계하였다. 터보 복호기나 상관기, 필터 블록 등에 최적화된 워드 길이는 4의 배수가 아닌 경우도 존재하나, 모든 워드 길이를 지원하는 것은 추가 하드웨어로 인해 면적과 속도 효율성에서 떨어진다. 프로그래머블 DSP 연산기에서 4의 배수 단위 워드 길이로 동작하는 점도 고려하였다.

그림 5에 전체 32비트 구조를 계층적으로 도시하였다. 4비트 동작 시에는 1단 캐리스킵로직만 존재하고, 32비트 시 최대 4단계까지 캐리스킵을 사용할 수 있다.

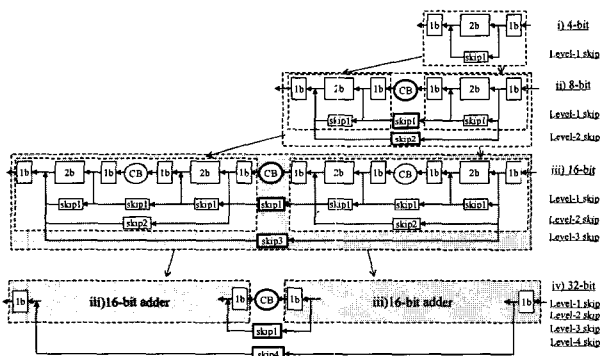


그림 5. 제안된 32비트 저전력 가변 길이 워드 및 데이터율 연산을 위한 계층적 가산기 전체 구조

Fig. 5. Overall structure of the proposed low power 32-bit hierarchical carry skip adder for variable length words and data rates.

표 2. 제안된 캐리스킵가산기 동작 모드

Table 2. Operating modes of the proposed carry skip adder.

워드 길이	병렬처리 워드 개수	가능한 동작 모드
4비트	8개	2가지 (RCA, 1-계층 CSkA)
8비트	4개	3가지 (RCA, 1- / 2- 계층 CSkA)
16비트	2개	4가지 (RCA, 1- / 2- / 3- 계층 CSkA)
32비트	1개	5가지 (RCA, 1- / 2- / 3- / 4- 계층 CSkA)

가변 워드 크기 별 동작 모드를 표 2에 보였다.

각 워드길이와 동작 성능에 따라 1단계(skip1)에서부터 4단계(skip4)까지의 계층적 캐리스킵로직이 제어 신호에 따라 활성화된다. 한편, 4비트, 8비트, 16비트 단위의 독립적인 가산기 역할을 위해서는 각 4비트 단위에서 상위 비트로 캐리 전파를 제어하는 로직이 필요하다. 그림 5의 CB(Carry Blocking) 로직으로 구현되었다.

2. 논리 회로 구성과 전력 효율적 동작 원리

제안된 32비트 가산기 구조의 최소 동작 단위인 4비트 블록을 그림 6에 좀 더 자세히 도시하였다. 2.1절의 식 (5)에서 보인 바와 같이 4비트 단위에서 LSB에서의 초기 캐리 전파 시간과 MSB에서의 캐리 흡수 시간을 고려했을 때 동작속도와 게이트 면적에서 최적 효율의 비트 분할은 1비트-2비트-1비트로 도출된다. 또한, 그림 1의 캐리스킵로직에서 최종 출력 캐리를 선택하는 2-input MUX 게이트 대신 논리적으로 동등한 2개의 AND 게이트 $G1$, $G2$ 와 1개의 2-input OR $G3$ 로 대체하여 구성하였다.

즉, 캐리스킵로직과 결부된 2비트 전가산기의 출력 캐리로 $c_3 = g_2 + p_2c_2 = g_2 + g_1p_2 + g_0p_1p_2 + c_0p_0p_1p_2$ 로 표현되고 그림 6의 논리 회로 구성에서 $c_{tmp0} = g_2 + g_1p_2 + g_0p_1p_2$, $c_{tmp1} = c_0p_0p_1p_2$ 로 구분되는데, c_{tmp0} 는 최초 입력 캐리 c_0 와 무관하므로 MUX 게이트와 논리적으로 동등한 것이다. 이때 AND게이트 구성으로 인해 active 신호에 의한 $G1$ 과 $G2$ 의 출력 노드 스위칭 제어도 가능하다. 예를 들어, 4비트 워드 길이에서 캐리스킵로직을 거치지 않고 리플캐리가산기처럼 동작시키고자 할 때 AND 게이트 $G1$ 에 입력되는 active 신호를 0으로 미리 고정하면 $G1$ 과 $G2$ 의 출력 노드의 스위칭은 다른 입력값에 무관하게 차단된다. 이때 $G3$ 의 한쪽 입력도 미리 고정되므로 게이트 지연 시간도 2개 입력이 동시에 스위칭 할 경우보다 줄어든다. 확장하여

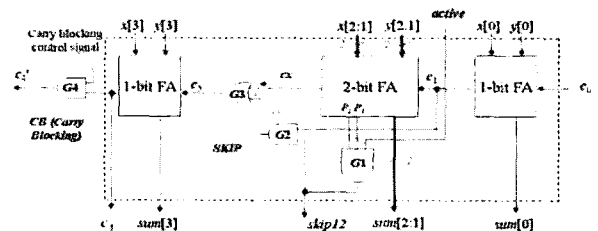


그림 6. 제안된 4비트 단위 캐리 스킵 가산기 구성

Fig. 6. Organization of the proposed 4-bit unit carry skip adder.

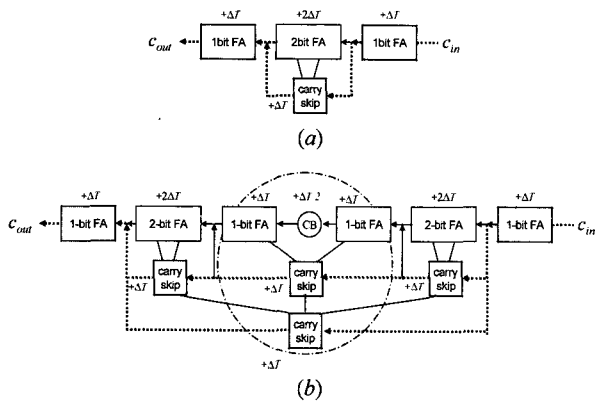


그림 7. (a) 4비트 동작 모드시 제안된 가산기 최대 지연 시간 경로 (b) 8비트 동작 모드시 1단 및 2단 캐리 스킵시 최대 지연시간 경로

Fig. 7. (a) The longest path delay of the proposed adder in 4-bit word addition mode. (b) The longest path delay of the proposed 8-bit word addition mode with respective one-stage and two-stage carry skip.

8비트나 16비트, 32비트 연산의 경우도 계층적으로 서로 다른 *active* 신호(*activeL1*, *activeL2*, *activeL3*, *activeLA*)를 통해 캐리 스킵 로직을 활성화함으로써 필요한 성능에 적합하게 제어할 수 있다.

4비트, 8비트, 16비트 모드로 동작하는 경우 상위 비트로 출력 캐리가 전파되는 것을 막기 위해 그림 6에 도식한 바와 같이 캐리 블로킹 제어 신호를 이용한 캐리 차단 로직을 추가하였다. 이 신호는 워드 길이에 따라 부호화된 제어 신호로 전달된다. 단순히 RCA로 동작할 경우에는 그림 6에서처럼 마지막 출력단에서 CB 게이트를 거치기 전에 c_4 로 출력하므로 추가적인 지연 시간 (CB 로직에서 2-input AND 게이트만큼의 지연 시간)은 없다. 8비트 이상으로 동작할 시에도 그림 5와 7의 (b)에서 보인 바와 같이 CB 로직은 캐리 스킵 회로에 의해 최대 지연 시간 경로에서 벗어나므로 전체 동작 속도에 미치는 영향은 최소화되어 있다.

저속 데이터 처리 시에는 RCA로 동작시키거나 캐리 스킵로직을 부분적으로만 활성화할 수 있다. 이 경우 처리 심볼의 워드 길이가 4비트 내외로 짧아 32비트 데이터 경로를 이용한 복수 워드 병렬 처리가 가능하므로 가산기의 동작 주파수는 추가적으로 더 감소시킬 수 있다.

IV. 구현 결과 비교

제안된 가산기와 비교에 사용할 가산기를 Verilog 하

표 3. 여러 워드 길이에서 제안된 구조(Proposed)와 리플캐리가산기(RCA), 캐리에건가산기(CLA) 게이트 면적 비교 (단위: μm^2)

Table 3. Comparison of area for each type of adder in multiple word lengths (unit: μm^2)

	4비트	8비트	16비트	32비트	비고
RCA	306.0	612.0	1,224.1	2,448.2	각각 4/8/16/32비트 단일모드로만 동작가능
CLA	538.8	1,150.9	2,335.1	4,743.4	
Proposed	362.5	811.6	1,696.4	3,579.2	하위 단위 워드 병렬 처리 가능

표 4. 여러 워드 길이에서 제안된 구조(Proposed)와 리플캐리가산기(RCA), 캐리에건가산기(CLA) 최대 지연 시간 비교 (단위: nano sec)

Table 4. Comparison of area for each type of adder in multiple word lengths (unit: ns)

	4비트	8비트	16비트	32비트
RCA	1.46	2.71	5.21	10.21
CLA	1.14	1.79	2.23	2.93
Proposed (Level-0)	1.56	2.81	5.63	11.46
Proposed (Level-1)	1.25	2.18	3.18	5.48
Proposed (Level-2)	-	1.97	2.87	4.76
Proposed (Level-3)	-	-	2.46	3.89
Proposed (Level-4)	-	-	-	3.24

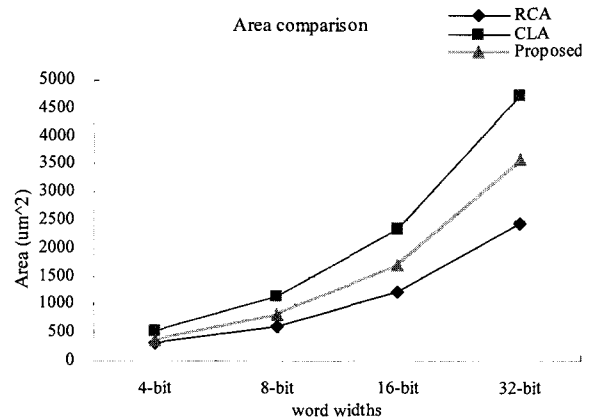


그림 8. 표 3의 면적 비교 결과 그래프
Fig. 8. Graphical representation of Table 3.

드웨어 설계 언어를 이용해 구조적으로 기술하고 TSMC 0.18 μm 표준 셀 라이브러리로 로직 합성한 결과를 표 3, 4와 그림 8에 보였다^[13]. Worst case에 대해 타이밍 분석을 수행하였으며, 로직 합성 시 논리회로 기본 구조가 변경되는 것을 방지하기 위해 컴파일 최적화 조건은 모든 가산기 로직 합성 시 생략하였다. 비교에 사용된 가산기는 최소 면적 구현이 가능한 리플캐리가산기(RCA)와 최대 동작 속도가 가능한 캐리에건가산기(CLA)이다.

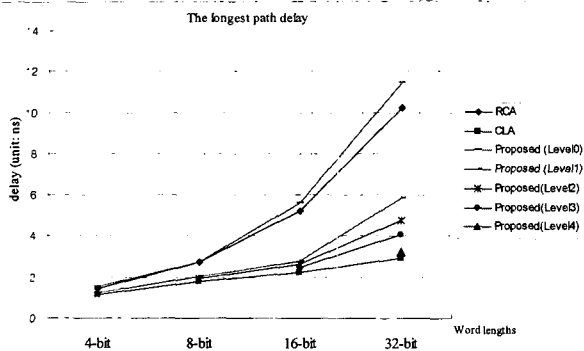


그림 9. 표 4의 최대 지연 시간 비교 결과 그래프
Fig. 9. Graphical representation of Table 4.

제안된 구조에서는 32비트 경우 8개의 4비트, 4개의 8비트, 2개의 16비트, 1개의 32비트 연산이 각각 가능하나, RCA와 CLA는 32비트 단일 워드 연산만이 가능하다. 16비트와 8비트에서도 마찬가지로 제안된 가산기는 하위 단위의 워드 길이를 처리할 수 있는 반면 RCA와 CLA는 각각 16비트, 8비트 처리만 가능하며, 4비트에서는 세 가지 연산기 모두 4비트 처리 기능만 존재한다. 표 3과 그림 8에서 보듯 제안된 구조는 면적 측면에서 RCA와 CLA의 중간 정도에 있다. 면적에 미치는 큰 영향은 없으나, 앞서 언급했다시피 제안된 가산기는 8비트에서 4비트 워드 2개 병렬 처리, 16비트에서 4비트 워드 4개와 8비트 워드 2개 병렬 처리 등 작은 단위의 워드 길이 처리가 가능하므로 2~3%의 추가적인 면적 오버헤드 요소를 포함하고 있다는 점을 감안해야 한다.

제안된 가산기는 그림 5와 표 2에 제시된 바와 같이 워드 길이와 요구 성능에 따라 동작 모드가 다양한 구조이다. 즉 요구되는 데이터 처리 속도에 따라 캐리스킵 로직을 단계별로 활성화시켜 성능을 가변할 수 있고 복수개의 짧은 워드를 병렬 처리할 수 있다. 캐리스킵 회로를 전혀 활성화하지 않았을 때가 표 4와 그림 8에서 0-계층(Level-0)에 해당한다. 이 경우 리플캐리가산기와 같은 모드로 동작하나 짧은 워드 단위의 병렬 처리시 캐리 전파를 차단하는 CB로직(그림 5, 6 참조)에 의해 약간의 성능 저하가 있다. 32비트의 경우 최대 4-계층(Level-4)의 캐리 스킵 로직이 활성화 가능하며, 이때 제안된 가산기의 최대 동작 속도(캐리 스킵을 4계층 모두 활성화 했을 때)는 CLA의 90% 수준이나, RCA에 비해서는 3배 이상 고속 동작시킬 수 있다(표 4, 그림 9 참조).

성능과 전력 소모에 대한 보다 객관적인 평가를 위해 전력-지연시간 곱(Power-Delay Product)을 지표로 이

표 5. 여러 워드 길이에서 제안된 구조(Proposed)와 리플캐리가산기(RCA), 캐리예건가산기(CLA) 면적 최대지연시간 곱 비교 (단위: $\mu\text{m}^2 \cdot \text{ns}$)

Table 5. Comparison of area/delay product for each type of adder in multiple word lengths (unit: $\mu\text{m}^2 \cdot \text{ns}$)

	4비트	8비트	16비트	32비트
RCA	446.8	1,658.8	6,377.6	24,996.1
CLA	614.3	2,060.1	5,207.3	13,898.2
Proposed	452.9	1,618.6	4,427.0	11,909.3

용할 수 있다. 전력, 지연 시간 모두 적을수록 유리하므로 최소의 전력-지연시간 곱을 갖는 가산기의 효율이 우수하다. 가산기의 경우 가능한 모든 입력 패턴에 대한 전력 예측 모의실험을 해야 하는 것이 보다 정확하나, 전력 모델링된 라이브러리가 부족과 임의의 입력 패턴에 대한 충분한 시간의 모의실험이 제한적이기 때문에, 여기서는 캐피시턴스 C 와 유사 비례 관계에 있는 게이트 면적으로 전력 부분을 대치하여 계산하였다. 동작 전압은 동일하고 가산기 회로의 모든 노드들은 임의의 입력 패턴에 대해 비슷한 확률로 활성화되므로 근사적인 결과를 얻을 수 있다.

표 5의 결과에서 나타나 있듯이 제안된 가산기는 8비트, 16비트, 32비트에서 가장 우수한 결과를 보이고 있고 4비트에서도 CLA에 비해 우위에 있다. 따라서 전력 효율 면에서도 상대적인 이점이 있다고 할 수 있다.

V. 결 론

본 논문에서는 4비트에서 32비트까지의 가변 워드 길이 및 데이터율을 수용하여 기능성을 확장하고 요구 성능에 따라 하드웨어 동작 모드와 속도 조절이 가능하여 전력 효율적인 계층적 캐리스킵가산기 구조를 제안하였다. 면적, 전력 소모 효율, 동작 모드 별 성능 결과를 기존 가산기들과 비교함으로써 우수성을 입증하였다. 이 구조는 다양한 이동 통신 표준을 수용하는 모뎀 프로세서에 적합하며, 제안된 구조를 확장하여 좀 더 다양한 동작 모드 구현도 가능하여 미디어 신호 처리 등 다른 응용 분야에 활용할 수 있다.

참 고 문 헌

- [1] On-line: <http://www.sdrforum.org>
- [2] A. Chandrakasan, S. Sheng, and R. Brodersen, "Low-power CMOS digital design," IEEE J.

- Solid-State Circuits, vol. 27, no. 4, pp. 473 - 484, Apr. 1992.
- [3] A. Chandrakasan and R. Brodersen, "Minimizing power consumption in digital CMOS circuits," Proceedings of the IEEE, vol. 83, no. 4, pp. 498 - 523, Apr. 1995.
- [4] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, Digital Integrated Circuits, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, Dec. 2002
- [5] C. Nagendra, M. J. Irwin, and R. M. Owens, "Area-time-power tradeoffs in parallel adders," IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 53, no. 10, pp 689-702, October 1996.
- [6] F. Kashifi, and N. Masoumi, "Optimization of Speed and Power in a 16-bit Carry Skip Adder in 70nm Technology," Proceedings of the 49th IEEE International Midwest Symposium on Circuits and Systems, pp. 28-31, Aug. 2006.
- [7] Y. S. Lin and D. Radhakrishnan, "Delay Efficient 32-bit Carry-Skip Adder," Proceedings of the 13th IEEE ICECS, pp. 506-509, Dec. 2006.
- [8] V. Kantabutra, "Designing optimum one-level carry-skip adders," IEEE Trans. on Computers, vol. 42, no. 6, pp. 759 - 764, June 1993.
- [9] M. Alioto and G. Palumbo, "A simple strategy for optimized design of one-level carry-skip adders," IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications, vol. 50, no. 1, pp. 141-148, January 2003.
- [10] S. Turrini, "Optimum group distribution in carry-skip adders," in Proceedings of the 9th IEEE Symposium on Computer Arithmetic, pp. 96 - 103, Sep. 1989.
- [11] P. Chan, M. Schlag, C. Thornborson, and V. Oklobdzija, "Delay optimization of carry-skip adders and block carry-lookahead adders using multidimensional dynamic programming," IEEE Trans. on Computers, vol. 41, no. 8, pp. 920-930, Aug. 1992.
- [12] G. Panneerselvam, and B. Nowrouzian, "Multiply-add fused RISC architectures for DSP applications," Proceedings of the IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, pp 108 - 111, May 1993.
- [13] TSMC 0.18um Process 1.8V SAGE-X Standard Cell Databook, Rel. 4.1, Artisan Component Inc., Sep. 2003

 저 자 소 개



한 태 희(평생회원)

1992년 KAIST 전기 및
전자공학과 학사 졸업.

1994년 KAIST 전기 및
전자공학과 석사 졸업.

1999년 KAIST 전기 및
전자공학과 박사 졸업.

1999년 3월 ~ 2006년 8월 삼성전자 통신연구소
책임 연구원.

2006년 9월 ~ 2008년 2월 한국산업기술대학교
전자공학과 교수.

2008년 3월 ~ 현재 성균관대학교 정보통신공학부
반도체시스템공학 전공 교수.

<주관심분야 : IT SoC 설계, VLSI 설계 기술 및
방법론, 단말 시스템>