

論文2000-37SD-1-10

Switched Capacitance 감소를 통한 저전력 16비트 ALU 설계

(A Design of Low Power 16-bit ALU by Switched Capacitance Reduction)

柳 凡 善 * , 李 重 錫 * , 李 奇 榮 * , 趙 泰 元 *

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

요 약

본 논문에서는 새로운 16비트 저전력 ALU(Arithmetic Logic Unit) 구조 및 회로를 제안하여 트랜지스터 레벨로 설계, 제작 및 테스트하였다. 설계한 ALU는 16개의 명령어를 수행하며 2단계 파이프라인 구조를 가진다. 제안한 ALU는 switched capacitance를 줄이기 위해 논리연산시에는 덧셈기가 스위칭하지 않도록 하였으며, P(propagation)블록의 출력을 듀얼버스(dual bus)구조로 하였다. 또한 이와 같은 ALU구조를 위한 새로운 효율적인 P 및 G(generation)블록을 제안하였다. 그 외에 저전력 실현을 위해서 ELM덧셈기, 이중모서리 천이 플립플롭(double-edge triggered flip-flop) 및 조합형 논리형태(combination of logic style)를 사용하여 ALU를 구현하였다. 모의실험결과, 제안한 구조는 기존의 구조^[1,2]에 비교하여 수행되는 산술연산의 사용횟수에 대하여 논리연산의 사용횟수가 증가할수록 전력감축의 효과가 증가하였다. 수행되는 산술연산 대 논리연산의 전형적인 비율을 7:3이라고 가정할 때, 제안한 구조는 기존 구조에 비해서 12.7%의 전력감축을 보였다. 설계한 ALU는 0.6 μ m 단일폴리, 삼중금속 CMOS 공정으로 제작하였다. 칩 테스트 결과 최대동작 주파수는 53MHz로 동작하였고 전력소모는 전원전압 3.3V, 동작 주파수 50MHz에서 33mW를 소모하였다.

Abstract

In this paper, a new low power 16-bit ALU has been designed, fabricated and tested at the transistor level. The designed ALU performs 16 instructions and has a two-stage pipelined architecture. For the reduction of switched capacitance, the ELM adder of the proposed ALU is inactive while the logical operation is performed and P(propagation) block has a dual bus architecture. A new efficient P and G(generation) blocks are also proposed for the above ALU architecture. ELM adder, double-edge triggered register and the combination of logic style are used for low power consumption as well. As a result of simulations, the proposed architecture shows better power efficient than conventional architecture^[1,2] as the number of logic operation to be performed is increased over that of arithmetic operation. It shows 12.7% power reduction assuming that the typical ratio of arithmetic to logic operation to be performed is 7 to 3, compared to conventional architecture. The proposed ALU was fabricated with 0.6 μ m single-poly triple-metal CMOS process. As a result of chip test, the maximum operating frequency is 53MHz and power consumption is 33mW at 50MHz, 3.3V.

* 正會員, 忠北大學校 電氣電子工學部
(School of Electronic and Electrical Engineering
Chungbuk National Univ.)

※ 본 연구는 산업자원부와 과학기술부 및 정보통신부
에서 시행하는 주문형 반도체 개발 사업의 지원을
받아 수행되었습니다.

接受日字:1998年12月19日, 수정완료일:1999年11月11日

I. 서론

최근 이동통신의 급격한 발전과 반도체 칩의 집적도 증가에 따른 칩의 비용(cost) 및 신뢰성(reliability) 측면에서 저전력 회로 설계의 필요성이 크게 대두되고 있다.

범용 마이크로프로세서와 DSP(digital signal processor)의 공통적인 블록 중의 하나가 ALU블록이다. 지금까지 발표된 ALU는 효율적인 ALU의 구현방법이나^[1, 2] 칩의 동작속도를 높이기 위한 구조들이었으며^[3], 전력감축에 대한 측면은 크게 고려되지 않았다. 문헌^[1]은 트랜지스터레벨에서 트랜지스터 수를 최소화하고 설계규칙성을 뛰어난 P 및 G블록을 포함한 효율적인 ALU 구현방법이 소개되었으나 고속동작 및 전력소모에 대한 언급이 없다. 문헌^[3]은 RISC 코어에 사용된 ALU로서 게이트 레벨로 설계되어 트랜지스터 레벨에 비해 하드웨어 오버헤드가 큰 것이 단점이다. 즉, 트랜지스터 레벨의 설계는 덧셈기 입력으로 들어가는 P 및 G블록의 프로그래밍이 가능하나, 게이트 레벨에서는 프로그램이 불가능하여 P 및 G 블록 앞에서 입력패턴 생성기가 그 역할을 하고 있다. 이 경우 트랜지스터 레벨의 설계와 비교하면 트랜지스터 레벨에서는 P 및 G블록을 프로그램 하는데 사용되는 제어신호를 발생시키는 디코더 블록만 필요한데 비하여, 게이트 레벨에서는 각 비트별로 입력패턴 생성기가 각각 존재해야 하기 때문이다. 문헌^[4]는 새로운 ALU구조에 대한 내용이 아닌 회로레벨에서 DPL(double pass-transistor logic)의 소개에 초점을 두고 있으며, ALU에서 가장 중요한 블록중의 하나인 덧셈기는 carry lookahead adder와 carry select adder를 혼합한 덧셈기를 사용하였다. 주지하는 바와같이 DPL은 패스 트랜지스터 논리형태중의 하나로서 상보입력이 필요하고 정적 CMOS에 비하여 interconnection수가 많기 때문에 정적 CMOS에 비해 전력소모가 큰 단점이 있다^[5].

저전력 시스템 구현을 위해서는 각 기능블록의 전력소모가 최소화되어야 한다. 임의의 시스템에서 전력감축의 효과는 상위레벨로 갈수록 크며, 상위레벨에서의 전력감축 방법중의 하나는 switched capacitance를 줄이는 것이다^[6, 7]. 여기서 switched capacitance값은 임의의 노드의 실제 캐패시턴스값에 그 노드의 스위칭 액티비티(switching activity)를 곱한 값이며, 임의의 노드

의 스위칭 액티비티는 그 노드가 클럭주기당 천이하는 평균 횟수를 말한다^[7].

일반적으로 ALU의 연산은 연산코드에 따라 덧셈연산을 필요로 하는 산술연산과 덧셈연산이 불필요한 논리연산으로 구분할 수 있다. 본 논문에서는 구조레벨에서 저전력 실현을 위하여 연산코드로부터 산술연산 감지기(AD: Arithmetic Detection) 신호를 발생시켜, 이 신호를 이용하여 논리연산에서는 덧셈기를 동작시키지 않으며, 또한 기존의 구조와는 달리 P블록의 출력을 듀얼버스 구조로 만들어 데이터 버스에서의 switched capacitance를 최소화시킨 구조를 제안한다. 그 외에 ALU에서 속도 및 전력소모 면에서 가장 우수하다고 보고된 ELM 덧셈기를 사용하였다^[8, 9]. 회로레벨에서 전력소모를 줄이기 위해서 조합형 논리형태(combination of logic style)를 사용하였다^[10].

II. 제안한 ALU

1. 명령어 및 입출력 핀

본 논문에서 제안 및 설계한 ALU는 고정소수점 16비트 정수연산이며 총 16개의 명령어를 수행하며 4개의 플레그 레지스터를 갖는다. ALU의 입·출력 핀은 표 1과 같고, 수행하는 명령어 집합은 ALU 구조에 대한 설명 부분의 표 2에 나타나 있다. 이 명령어 집합은 아날로그 디바이스사의 DSP 제품중의 하나인 ADSP 2101의 ALU 명령어와 동일하다.

표 1. 입/출력 핀
Table 1. Input/output pin descriptions.

| Pin name | in/out | function |
|----------|--------|----------------|
| ψ | input | clock signal |
| X | input | source 1 input |
| Y | input | source 2 input |
| C_m | input | carry input |
| OP | input | OP code |
| S | output | result |
| AS | output | sign flag |
| AC | output | carry flag |
| AV | output | overflow flag |
| AZ | output | zero flag |

2. 구조

그림 1은 본 논문에서 제안 및 설계한 저전력 ALU 구조이다. 제안한 구조는 첫 번째 단계에서는 연산코드 (op code)를 해독하고 두 번째 단계에서는 연산을 실행하는 2단계 파이프라인 구조를 가진다. 전체회로의 동작 설명은 다음과 같다. 연산코드는 디코더 블록을 통과하여 현재 명령에 대한 P 및 G블록의 제어신호를 발생시킨다. P 및 G블록에 입력된 두 개의 16비트 데이터가 디코더 블록에서 발생된 제어신호에 따라 적당한 값을 계산하여 ELM 덧셈기 또는 최종단의 멀티플렉서로 전달된다. 산술연산의 경우에는 P 및 G블록의 출력값이 ELM 덧셈기와 멀티플렉서를 통해 최종 출력에 나타나고, 논리연산의 경우에는 P블록의 출력이 덧셈기를 거치지 않고 곧바로 멀티플렉서를 통해 최종 출력에 나타난다. 각 연산코드에 따른 P 및 G블록의 출력값 및 ELM 덧셈기의 캐리 입력값을 표 2에 나타내었다. 산술연산 감지기 신호는 현재의 명령이 산술연산인 경우에 '1'이 되며 이 신호가 P 및 G 블록으로 입력되어 A, B, C, D 버스를 제어한다. 이 신호가 '1'이면 B, C, D가 활성화되며 '0'이면 A버스로 활성화된다. 또한 이 신호가 논리연산 결과와 산술연산 결과를 선택하는 최종 멀티플렉서의 선택신호로 사용된다. 플레그 레지스터로는 AS, AV, AC 및 AZ가 있으며 각각은 AS는 sign flag, AV는 overflow flag, AC는 carry flag, AZ는 zero flag를 나타낸다.

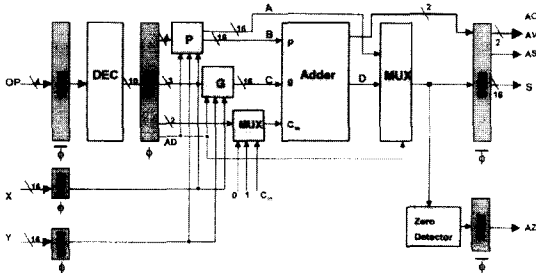


그림 1. 제안한 ALU 구조
Fig. 1. The Proposed ALU Architecture.

본 논문에서 제안한 ALU가 기존에 발표되었던 ALU^[11]와 차별성을 기술하면 다음과 같다.

첫째, 논리연산수행시 덧셈기가 스위칭하지 않도록 하여 덧셈기에서의 동적 전력소모를 제거하였다.

둘째, 제안한 구조는 P블록의 출력이 듀얼버스 구조

를 갖기 때문에 ALU 연산시에 필요한 최소의 캐패시턴스만 스위칭 시켜 switched capacitance값을 최소화시킨 구조이다. 기존 ALU의 P블록은 하나의 출력버스가 덧셈기 및 멀티플렉서에 공동으로 연결되어 산술연산시에는 멀티플렉서의 입력캐패시턴스가, 논리연산시에는 덧셈기의 입력캐패시턴스가 불필요하게 스위칭하는 단점이 있다. 제안한 구조는 P블록의 출력을 A, B 등 2개의 듀얼버스로 내보내어 산술연산시에는 B버스로, 논리연산시에는 A버스로 활성화시켜 switched capacitance를 최소화하였다.

그 외에 사용한 덧셈기는 속도, 전력소모, 면적 및 설계규칙성을 고려하여 가장 성능이 우수한 ELM 덧셈기를 사용하였고^[9], 논리형태는 정적 CMOS와 저전력 XOR 게이트를 구성된 조합형 논리형태를 사용하였다^[10]. 또한, 레지스터는 단일 모서리 천이 레지스터 (single-edge triggered register) 대신 클럭네트의 스위칭 액티비티를 감소시켜 전력소모를 줄인 이중 모서리 천이 레지스터를 사용하였다^[12-13].

표 2. ALU 명령어 및 명령어에 따른 P, G 블록의 출력값

Table 2. ALU instructions and output values of P, G blocks according to the instructions.

| Instruction | Arithmetic/Logic | MSB of X | Output of P block | Output of G block | Carry input of Adder |
|------------------------|------------------|----------|-------------------|-------------------|----------------------|
| X+Y | Arithmetic | x | X⊕Y | XY | 0 |
| X+Y+C _{in} | | x | X⊕Y | XY | C _{in} |
| Y+1 | | x | Y | 0 | 1 |
| X-Y | | x | X⊖Y | XY' | 1 |
| Y-X | | x | X⊖Y | X'Y | 1 |
| X-Y+C _{in} -1 | | x | X⊖Y | XY' | C _{in} |
| Y-X+C _{in} -1 | | x | X⊖Y | X'Y | C _{in} |
| Y-1 | | x | Y' | Y | 0 |
| -Y | | x | Y' | 0 | 1 |
| Not X | | Logic | x | X' | x |
| Not Y | x | | Y' | x | x |
| Y | x | | Y | x | x |
| X AND Y | x | | X∧Y | x | x |
| X OR Y | x | | X∨Y | x | x |
| X X OR Y | x | | X⊗Y | x | x |
| [X] | Logic | 0 | X | x | x |
| | Arithmetic | 1 | X' | 0 | 1 |

$$X⊗Y=X'Y'+XY'$$

$$X⊖Y=X'Y'+XY$$

\wedge : logical AND

\vee : logical OR

x : don't care

3. 덧셈기

최근의 연구에 의하면 여러 종류의 병렬 덧셈기 중에서 속도, 전력소모, 면적 및 설계규칙성을 기준으로 하여 가장 최적의 구조는 ELM 덧셈기라고 보고되고 있다^[9]. ELM 덧셈 알고리즘은 다단계 논리 합성 툴(multi-synthesis tool)에 의해 개발된 것으로 셀 간의 연결 횟수를 최소화한 알고리즘이다^[8-9].

ELM 덧셈기는 E, P, G, S 등 4개의 기본 셀이 존재하는데 각 셀이 수행하는 연산은 각각 $E = ps \oplus G$, $P = PQ$, $G = G+PH$, $S = ps \oplus PG$ 이다. E 및 S셀은 부분합을 계산하고 G셀은 블록캐리발생신호(block carry generation signal)를, P셀은 블록캐리전파신호(block carry propagation signal)를 계산한다. 그림 2의 (a)는 ELM 덧셈기의 4개의 기본 셀을 나타내고 (b)는 4비트 ELM 덧셈기의 구조이다. 그림 2에서 보면 출력인 S_3, S_2, S_1, S_0 와 0비트에서 3비트까지의 블록 전파를 나타내는 블록캐리전파신호 P_i , 그리고 4비트 블록캐리발생신호인 G_3 가 있다. P_i 는 0비트에서 3비트까지의 각 비트를 XOR연산 후에 AND연산을 한 것으로 다음 식(1)로 쓸 수 있으며, 이는 우측 노드의 블록캐리발생신호인 G_i 를 현재 비트의 합에 전달할지를 결정한다. 즉, P_i 가 '1'이면 현재 비트에 우측 노드의 G_i 가 전달되고, P_i 가 '0'이면 G_i 가 전달되지 않는다. $G(j, i)$ 는 식 (2)에서 보는 바와 같이 i 번째와 j 번째 사이의 블록캐리발생신호를 나타낸다. 식 (2)에서 임의의 블록은 상위비트 블록과 하위비트의 블록으로 구성되는데, 임의의 블록의 블록캐리발생신호($G(j, i)$)는 상위비트의 블록캐리발생신호(g_j)가 '1'이거나 하위비트의 블록캐리발생신호($G(j-1, i)$)가 '1'이고 하위비트의 블록캐리발생신호가 '1'이면, '1'이 된다. 4비트 ELM 덧셈기는 두 개의 4비트 입력을 받아들이며, 4비트의 부분 합, 4비트의 P_i , 그리고 1비트의 G_3 로 구성된다.

$$P_i = (A_i \oplus B_i) (A_{i-1} \oplus B_{i-1}) (A_{i-2} \oplus B_{i-2}) \dots (A_{i-3} \oplus B_{i-3}) \quad (i = 0, 3) \tag{1}$$

$$G(j, i) = g_j + b_j G_{j-1, i} \tag{2}$$

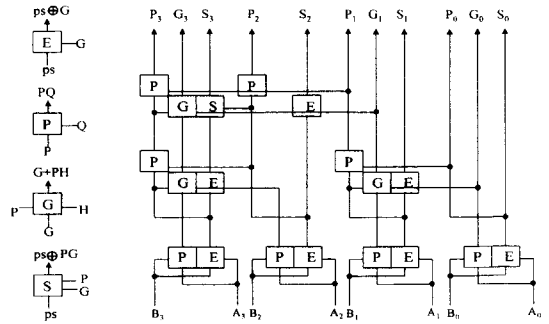


그림 2. 4비트 ELM 덧셈기
Fig. 2. The 4-bit ELM adder.

4. P 및 G 블록

그림 3은 기존의 ALU구조에 사용되었던 P 및 G 블록을 트랜지스터 레벨로 나타낸 회로도이다^[1,5].

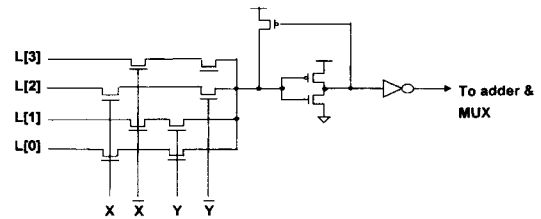


그림 3. 기존의 P 및 G 블록
Fig. 3. Conventional P and G block.

회로도 동작은 현재의 X, Y 입력에 대해 $L[0], L[1], L[2], L[3]$ 등 4개의 제어신호에 따라 16개의 연산을 수행한다. 정제환 PMOS는 high level을 복원하기 위한 weak 트랜지스터이다. 이 회로는 동일한 회로를 사용하여 4개의 제어신호로써 출력 값을 프로그램 할 수 있기 때문에, 게이트 레벨의 설계^[3]에 비해서 하드웨어가 작고 규칙적인 설계가 가능하다. 그런데 기존의 G블록은 논리연산시에도 출력노드가 스위칭하여 불필요한 전력을 낭비하는 단점이 있다. 따라서 본 논문에서는 논리연산시에는 출력노드가 스위칭하지 않는 G블록을 제안하였다. 제안한 G블록의 트랜지스터레벨의 회로는 그림 4와 같다. 그림 4에서 AD신호는 산술연산 감지기 신호를 나타낸다. 이 신호가 '1'이면 산술연산을 의미하며, 출력값이 입력값에 의해 계산된다. 반면에 이 신호가 '0'이면 논리연산을 의미하며, G블록 회로가 disable되며 출력노드가 스위칭하지 않는다.

제안한 P블록의 회로는 그림 5와 같다. G블록과의 차이점은 AD 및 \overline{AD} 신호에 의해 ELM 덧셈기 및 멀티플렉서의 입력이 되며, 한 신호가 계산되는 동안에는

다른 신호는 이전상태를 유지하여 스위칭을 하지 않는다. 산술연산의 경우에는 $AD=1$ 이고, 논리연산의 경우에는 $\overline{AD}=1$ 이 된다.

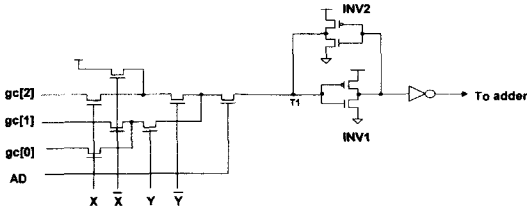


그림 4. 제안한 G 블록
Fig. 4. Proposed G block.

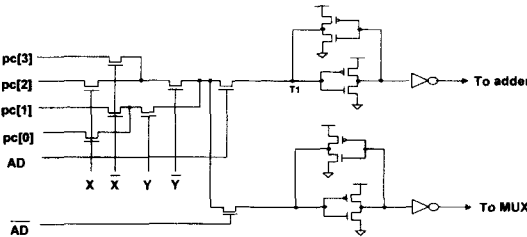


그림 5. 제안한 P 블록
Fig. 5. Proposed P block.

5. 레지스터

일반적인 단일 모서리 천이 플립플롭은 클럭의 한 주기 동안에 상승에지 또는 하강에지에서 데이터를 한번만 저장한다. 반면에 이중 모서리 천이 플립플롭은 클럭의 한 주기동안 상승에지 및 하강에지에서 모두 데이터를 저장하기 때문에, 단일 모서리 천이 플립플롭에 비교하여 동일한 산출량(throughput)을 유지하는 조건에서 클럭네트에서의 스위칭 액티비티가 반으로 줄

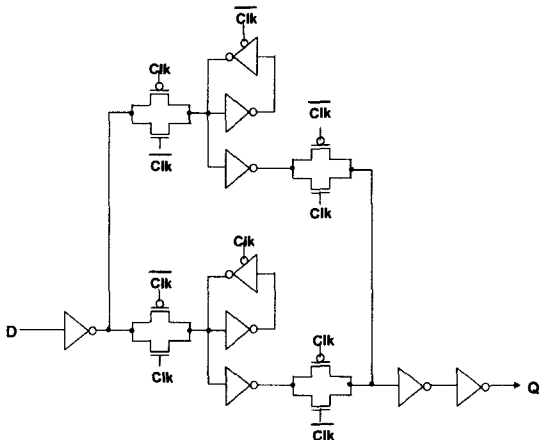


그림 6. ALU에 사용된 이중 모서리 천이 D 플립플롭
Fig. 6. Double-Edge Triggered D Flip-flop to be used in ALU.

어들어 전력소모가 45%까지 감소된다고 보고되고 있다^[13]. 따라서 ALU에 사용한 레지스터는 그림 6에서 보는 바와 같이 static style이며, 기존 회로에 비해 testability를 개선한 회로이다^[13]. 표 3은 단일 모서리 천이 플립플롭 및 이중 모서리 천이 플립플롭의 비교를 위해 16비트 ALU를 각각 단일 모서리 천이 플립플롭 및 이중 모서리 천이 플립플롭으로 구현한 다음, 실험결과를 나타낸 표이다. 표 3으로부터 이중 모서리 천이 플립플롭이 단일 모서리 천이 플립플롭에 비하여 약 14.1%의 전력감축이 있음을 알 수 있다. 이는 이중 모서리 천이 플립플롭이 단일 모서리 천이 플립플롭과 비교하여 트랜지스터 증가에 따른 전력증가보다는 부하가 큰 클럭네트의 스위칭 액티비티가 줄어들어 인하여 전력감축의 효과가 큼을 의미한다.

표 3. 16비트 ALU구현을 통한 단일 모서리 천이 플립플롭과 이중 모서리 천이 플립플롭의 전력소모 비교

Table 3. Power consumption comparison of SET F/F with DET F/F by implementing 16bit ALU.

| | SET F/F으로 구현한 ALU | DET F/F으로 구현한 ALU | 전력감축율 (%) |
|------------------|-------------------|-------------------|-----------|
| 트랜지스터 수 | 3,820 | 4,318 | |
| F/F의 전력소모(mW) | 34.71 | 29.84 | 14.1 |
| 전체 ALU의 전력소모(mW) | 50.03 | 43.15 | 13.7 |

* 모의실험 조건 : 100MHz, 3.3V, 25°C, Typical process parameter

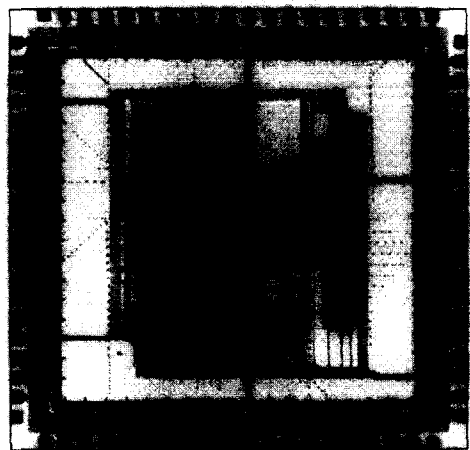


그림 7. 제안한 16비트 ALU의 레이아웃
Fig. 7. Physical Layout of the proposed 16-bit ALU.

그림 7은 설계한 ALU의 레이아웃을 나타낸 그림이다. 코어 부분의 좌측이 입력 레지스터와 제안한 P 및 G블록이며, 우측은 ELM 덧셈기, 그리고 우측 가장자리는 멀티플렉서 및 출력 레지스터이다. 또한 코어 부분의 하단부는 연산코드를 해독하기 위한 디코더 블록이다. 클럭신호와 같은 스위칭 빈도수가 높고 부하 캐패시턴스가 큰 노드의 배선은 단위 길이당 캐패시턴스 값이 가장 작은 메탈 3 레이어를 사용하였다^[14].

III. 모의실험결과

본 절에서는 제안한 구조가 기존의 구조[1,2]에 비해 전력감축 효과를 알아보기 위한 모의실험을 다음과 같이 수행하였다. SPICE 모의실험을 위한 각각의 회로설계는 제안한 ALU구조의 전력감축 효과를 공정하게 나타내기 위하여 F/F 및 디코더 블록을 제외한 순수한 ALU의 하드웨어인 P블록, G블록, 덧셈기 및 멀티플렉서를 그리고 표 4와 같이 기존의 구조와 제안한 구조의 차이점을 두어 회로설계를 하였다.

표 4는 기존의 구조와 제안한 구조의 특징 비교 및 기대효과를 나타낸 표이다. 제안한 구조의 첫 번째 특징은 논리연산시에 덧셈기의 스위칭 횟수를 줄인 것으로서, 산술연산의 명령어와 논리연산의 명령어의 사용 빈도수에 따라서 전력감축의 효과가 달라진다. 두 번째 특징은 P블록의 출력이 듀얼버스 구조를 갖는다. 기존의 구조에서는 P블록의 출력이 덧셈기 및 멀티플렉서에 공통으로 연결되어 있기 때문에 산술연산시에는 멀티플렉서의 입력 캐패시턴스가 불필요하게 스위칭하며, 논리연산시에는 덧셈기의 P입력에 해당하는 입력 캐패시턴스가 불필요하게 스위칭 한다. 제안한 구조는 P블록의 출력을 덧셈기 및 멀티플렉서 입력으로 분리함으로써 동작에 필요한 캐패시턴스만을 스위칭 한다. 실험 조건은 전원전압 3.3V, 온도 25°C, 표준공정변수, 입력 신호의 상승/하강시간은 1ns, 입력신호의 주파수는 100MHz에서 HSPICE를 이용하여 수행하였다. 제안한 ALU는 산술 및 논리연산의 사용 빈도수에 의해 전력감축의 효과가 달라지므로 실험을 산술 및 논리연산의 사용 빈도수를 각각 9:1, 7:3, 5:5, 3:7 및 1:9 로 나누어 각각 수행하였다.

그림 8은 표 4를 근거로 하여 수행되는 산술연산 대 논리연산의 사용 빈도수에 따라 기존의 ALU^[1, 2]와 제안한 ALU의 전력소모비교를 나타낸 것이다. 지연시간

은 기존구조가 4.023ns가 걸린 반면에, 제안한 구조는 4.313ns로 증가하였는데 그 이유는 P, G블록의 AD신호를 계산하는 시간 및 내부노드의 캐패시턴스값이 증가하였기 때문이다. 반면에 전력소모는 산술 및 논리연산의 사용 빈도수에 따라서 전력감축 효과가 달라짐을 확인하였다. 전력감축율은 산술연산 : 논리연산의 비가 9:1일 때 3.7%, 7:3일 때 12.7%, 5:5일 때 17.1%, 3:7일 때 33.1%, 1:9일 때 38.9%로 나타나 논리연산의 사용 빈도수가 증가할수록 제안한 구조의 전력감축 효과가 커짐을 확인하였다. DSP에 사용되는 ALU의 산술명령 및 논리명령의 전형적인 사용 빈도수를 7:3이라고 가정할 때, 제안한 ALU가 기존 구조에 비해서 12.7%의 전력감축이 있음을 알 수 있다.

표 4. 기존의 구조와 제안한 구조의 특징비교
Table 4. Architecture comparison between conventional and proposed ALU.

| 항 목 | 기존의 구조 | 제안한 구조 | 기대효과 |
|-----------------|--------|----------|----------------|
| 논리연산시에 덧셈기의 스위칭 | 스위칭함 | 스위칭하지 않음 | 불필요한 스위칭 감소 |
| P블록의 출력 | 1개 | 2개 | 필요한 캐패시턴스만 스위칭 |

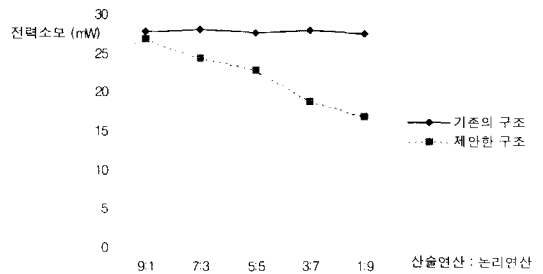


그림 8. 산술연산 대 논리연산의 비에 따른 기존 구조와 제안한 구조의 전력소모 비교

Fig. 8. Power consumption comparison between conventional and proposed architecture according to the ratio of arithmetic to logic operation.

IV. 측정결과

설계한 16비트 ALU는 0.6μm 단일폴리, 삼중금속 CMOS 공정으로 제작되었다. 그림 9는 shmoo 테스트 결과를 나타낸 표이다. 최대동작 주파수가 50MHz로 낮은

이유는 설계시에는 probe capacitance를 10pF으로 가정하고 설계하였는데 측정장비의 probe capacitance의 값은 72pF으로, 설계시에 테스트 환경을 충분히 고려하지 못했기 때문이다. 전력소모 측정은 50MHz의 클럭과 테스트 벡터를 인가한 상태에서 코어부분의 전원전압과 접지 사이에 흐르는 전류를 측정하여 평균 전력소모값이 33mW임을 확인하였다. 이 측정치는 레이아웃으로부터 기생 캐패시턴스를 추출하여 실험한 결과와 비교하면, 실험결과는 50MHz에서 35.56mW가 소모되었는데 이것은 측정치보다 약 7.2% 증가한 수치이다.

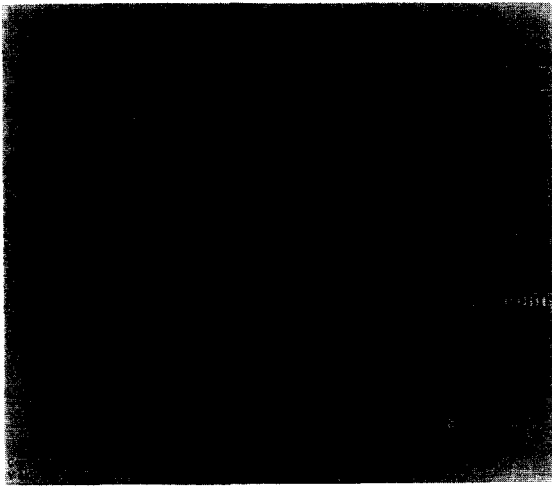


그림 9. Shmoo 테스트 결과
Fig. 9. Shmoo test result.

표 5. ALU 테스트 칩의 특성 및 참고문헌과의 성능비교

Table 5. Characteristics of the ALU test chip & performance comparison with reference.

| | This work | Reference [4] |
|-------------------------------|----------------------------|------------------------------|
| Process Technology | 0.6μm CMOS SPTM | 0.25μm CMOS SPTM |
| Procession(bit) | 16 | 32 |
| Architecture | ELM Addition | Carry Select Addition |
| Logic Style | Combination of Logic Style | Double Pass transistor Logic |
| Maximum Frequency(MHz) | 53 | 50 |
| Power Dissipation(mW) | 33@50MHz, 3.3V | 8@50MHz, 2.5V |
| Supply Voltage(V) | 3.3 | 2.5 |
| Transistor Counts | 4318 | |
| Active area(mm ²) | 5.28 | 0.60 |

표 5는 설계한 ALU의 테스트 결과 및 참고문헌^[4]와의 성능비교를 나타낸 표이다. 기존의 발표된 ALU로서는 제작 및 측정을 마친 참고문헌은 문헌^[4]이 유일하다. 이 논문은 ALU의 정확한 구조 설명이 없고, 고속동작을 목표로 하였기 때문에 전력소모의 비교대상으로는 적절하지 않아 직접적인 전력비교가 어려웠다.

V. 결론

본 논문에서는 16비트 ALU의 전력소모를 줄이기 위해 구조적인 측면에서 switched capacitance를 최소화시킨 ALU구조를 제안하였고, 그 외에 덧셈기, 레지스터 및 논리형태측면에서 저전력 실현을 위한 설계를 하였다. Switched capacitance를 줄이기 위해 첫째로는 논리연산시에 덧셈기가 스위칭하지 않도록 하였고 둘째로는 P블록의 출력을 듀얼버스로 하여 P블록의 출력을 덧셈기로 입력되는 부분과 멀티플렉서로 입력되는 부분으로 나눔으로써 연산 시에 필요한 캐패시턴스만 스위칭 시킴으로써 전력소모를 최소화하였다. 이를 위하여 연산코드로부터 산술연산 감지기 신호를 발생시켜, 이 신호로 P, G 및 덧셈기 블록을 제어하였다. 그 외에 저전력 소모에 적합한 ELM 덧셈기 및 이중 모서리 천이 플립플롭을 사용하였고, 정적 CMOS와 저전력 XOR게이트를 혼합한 조합형 논리형태를 사용하였다. 모의실험결과, 제안한 구조는 기존의 구조^[1,2]에 비하여 논리연산의 사용빈도수가 증가할수록 전력감축의 효과가 크게 나타났으며, 산술연산 대 논리연산의 사용빈도수의 비가 7:3일 때 12.7%의 전력감축을 보였다.

제안한 ALU는 0.6μm 단일폴리, 삼중금속 CMOS 공정으로 설계 및 제작하였으며 테스트 결과, 동작속도는 I/O설계시에 테스트환경을 충분히 고려하지 못하여 전원전압 3.3V에서 최대 53MHz로 동작하였고, 전력소모는 동작 주파수 50MHz에서 33mW를 소모하였다. 그런데 레이아웃 면적이 크에 따라 이로 인한 기생 캐패시턴스의 증가로 지연시간 및 전력소모의 증가를 초래하였으며 차후에는 레이아웃 면적을 최소화하여 성능향상을 할 예정이다.

참고 문헌

[1] Carver Mead, Lynn Conway, "Introduction to VLSI system," Addition-wesley publishing

- compony, pp 150-154, 1980.
- [2] Abdellatif Bellaouar, Mohamed I. Elmasry, "Low Power Digital VLSI Design," Kluwer academic publishers, 1995.
- [3] 박승호, 최병윤, 이문기, "PDA를 위한 32비트 RISC 코어의 설계," 한국통신학회 논문지, 제22권, 제10호, pp 2136-2149, 1997
- [4] Makoto Suzuki, et al., "A 1.5ns 32-b CMOS ALU in Double Pass-Transistor Logic," IEEE J. Solid-State Circuits, vol. 28, no. 11, Nov. 1993.
- [5] 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.
- [6] Anantha P. Chandrakasam, Robert W. Brodersen, "Low Power Digital CMOS Design," Kluwer academic publishers, 1995.
- [7] Jan M. Rabaey, Massoud Pedraw, "Low Power Design Methodologies," Kluwer academic publishers, pp 37-62, 1996.
- [8] 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, Sep. 1992.
- [9] 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, Oct. 1996.
- [10] 김문수, 유범선, 강성현, 이중석, 조태원, "하이브리드 로직 스타일을 이용한 저전력 ELM 덧셈기 설계," 전자공학회 논문지, 제35권, C편, 제6호, pp. 389-396, 1998
- [11] 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. 29, no. 7, July, 1994.
- [12] M. Afghahi, J. Yuan, "Double edge-triggered D-flip-flops for High Speed CMOS Circuits," IEEE J. of Solid-State Circuits, Vol. 26, No. 8, pp 1168-1170, 1991.
- [13] R. Llopis and M. Sachdev, "Low Power Testable Dual Edge Triggered Flip-Flop," Digest of Technical Papers, International Symposium on Low Power Electronics and Design, pp. 341-345, 1996.

 저 자 소 개

柳 凡 善(正會員) 第 36卷 第 5號 參照
 현재 충북대학교 대학원 전자공학과 박사과정

李 重 錫(正會員) 第 35卷 第 6號 參照
 현재 삼성전자 반도체 사업부 공정관리실 근무

李 奇 榮(正會員) 第 36卷 第 5號 參照
 현재 충북대학교 공과대학 전기전자공학부 교수

趙 泰 元(正會員) 第 36卷 第 5號 參照
 현재 충북대학교 공과대학 전기전자공학부 교수