디지털산업정보학회 논문지 제4권 제1호 - 2008년 3월

이동형 시스템에서 프로세서의 전력 소모 최소화를 위한 주파수 선택 알고리즘

김 재 진* · 강 진 구** · 허 화 라*** · 윤 충 모****

A Frequency Selection Algorithm for Power Consumption Minimization of Processor in Mobile System

Kim, Jae Jin · Kang, Jin Gu · Hur, Hwa Ra · Yun, Choong Mo

<Abstract>

This paper presents a frequency selection algorithm for minimization power consumption of processor in Mobile System.

The proposed algorithm has processor designed low power processor using clock gating method. Clock gating method has improved the power dissipation by control main clock through the bus which is embedded clock block applying the method of clock gating. Proposed method has compared power consumption considered the dynamic power for processor, selected frequency has considered energy gain and energy consumption for designed processor. Or reduced power consumption with decreased processor speed using slack time.

This technique has improved the life time of the mobile systems by clock gating method, considered energy and using slack time. As an results, the proposed algorithm reduce average power saving up to 4% comparing to not apply processor in mobile system.

Key Words: Process, Mobile System, Low Power, Frequency Selection Algorithm

I. 서론

이동형 컴퓨팅(mobile computing)과 무선 통신 등의 산업 분야를 비롯한 군수용 산업 분야까지 다양한 분야에서 사용되고 있다. 특히, 이동형 시스템(mobile system)은 점점 수요 및 분야가 증가되고 있어 관심이 날로 증대되고 있다. 또한 이러한 분야의 증가뿐만 아니라 기존의 제품들도 다양한 기능의 첨가를 통한 경쟁력 증가를 추구하고 있다. 이러한 요구를 만족시키기 위한 방법으로 고성능의 프로세서(processor)를 이용한 이동형 시스템의 개발이 가속화 되고 있으나 이러한 시스템들은 고성능을 발휘하기 위해 전력 소모도 증가되고 있다. 이러한 문제를 해결 할 수 있는 방안으로 이동형 컴퓨팅을

* 극동정보대학 컴퓨터정보과 교수
** 극동정보대학 애니메이션과 교수
*** 송호대학 컴퓨터정보과 교수
**** 서일대학 정보기술학부 교수

위한 저전력 프로세서가 개발되었으며 이로 인하여 적은 소모 전력으로 고성능을 발휘할 수 있게 되었다. 그러나 저전력 프로세서를 이용하여 프로세서의 전력 소모는 줄어든 반면에 시스템을 차지하고 있는 디바이스(device)들의 전력 소모는 증가되고 있다. 이동형 시스템의 전력 소모를 줄이기 위한 방안으로 프로세서의 경우는 클럭 게이팅(clock gating)을 이용하여 전력 소모를 줄이는 방법들이 제안되었다[1,2,3,4,5]. 또한, 저전력의 프로세서를 설계하기 위한 다른 방법으로 스케줄링(scheduling) 방법을 이용하여 소모 전력을 낮추는 방법이 있으며, 실시간 시스템을 기반으로 외부/내부 슬랙(external/internal slack)을 활용하는 DVS(Dynamic Voltage Scaling)가 제안되었다[6]. 그러나 이러한 방법들은 모두 프로세서의 소모 전력을 감소시켜 저전력의 프로세서를 설계할 수 있는 방법들이지만, 프로세서의 주파수를 고려하여 최적의 주파수를 결정하고 이를 통하여 소모 전력을 최적화하는 방법이 제안되지 않은 단점이 있다.

따라서 본 논문에서는 이와 같은 단점을 보완하여 이동형 시스템에 사용되는 프로세서에 대해 소모 전력을 줄일 수 있는 방법으로 프로세서에 대해 클럭 게이팅 방법과 에너지, 슬랙을 고려하여 프로세서의 전력소모가 최적화 될 수 있는 주파수를 결정하는 방법을 제안하였다.

Ⅱ. 관련 연구

프로세서에 대한 저전력 구성 방법에서 고려하여할 부분은 CMOS에서의 저전력 방법과 RTL 에서의 저전력 방법으로 나누어 볼 수 있으며, 저전력 프로세서를 구성하여 전체 시스템에 적용된 후 전체 시스템의 소모 전력을 감소시키기 위한 방법으로 디바이스에 대해 고려한 DPM 방법에 대한 이론을 필요로 한다.

2-1. CMOS에서의 저전력 방법

CMOS회로에서 전력 소모의 주된 원인은 크게 스위칭전력, 단락전류, 누설전류이며, 이들중 가장 큰 원인은 스위칭전력이다. CMOS 게이트에서 소모되는 평균 전력은 식 (1) 과 같이 표현된다.

$$P = P_{switching} + P_{short\text{-}circuit} + P_{leakage} = \frac{1}{2} \cdot \frac{C \cdot V_{dd} \cdot N}{T} + I_{sc} \cdot V_{dd} + I_{leakage} \cdot V_{dd} \quad (1)$$

여기서 P는 소모되는 전체 전력을 나타내며 V_{dd}는 공급전압, T는 클록 주기이다. $P_{switching}$은 CMOS 회로 입력의 스위칭에 따라 충·방전될 때 발생되는 스위칭 또는 동적(dynamic) 전력소모를 나타낸다. C는 로드 캐패시턴스(load capacitance)를 나타내며, N은 각 클록 주기마다 게이트 출력에서의 스위칭 동작 수를 나타낸다. $P_{short-circuit}$는 NMOS와 PMOS가 동시에 동작할 경우 공급전원에서 접지로 직접 흐르는 단락전류(short-circuits)에 의해 소모되는 것을 나타낸다. $P_{leakage}$는 벌크(bulk)지역에서 역 방향 다이오드의 누설 전류에 의하여 발생하는 것으로 IC 공정 기술에 의해 결정 된다. 이러한 소모 전력 요인 중에서도 VLSI 회로에서 전체 전력 소모량의 대부분(90%)을 차지하는 것은 스위칭 전력이다[7]. 스위칭 전력을 최소화하는 저전력 방법은 공급 전압 Vdd을 줄이거나 캐패시턴스 또는 스위칭 동작수(N/T)를 줄이는 것이다[8,9,10]. 소모되는 전력에서 공급전압을 줄이는 것은 매우 효과적이긴 하나 공급 전압이 감소함에 따라 회로의 지연시간이 늘어나게 되는 단점이 있다. 따라서 이러한 단점을 해결하기 위한 방법으로 병렬화와 파이프라이닝 등을 사용하여 회로의 성능을 증가시킨 다음 공급 전압을 낮추어 처리량(throughput)을 유지하는 방법을 사용한다. 또 다른 방법은 소비 전력이 주파수와 직접 비례하는 특성을 이용하여 주파수를 낮추는 것이다. 하지만 이러한 방법은 고정

된 타이밍 제한 조건을 가지는 실시간 응용 분야에는 주파수를 감소시킬 수 없으므로 적용이 어려운 문제가 있다. 단 외부 데이터 율(external data rate)이 고정되면, 하드웨어가 시간을 최대로 활용할 수 있도록 내부 클록을 선택할 수 있다.

2-2. RTL에서의 저전력 방법

디지털 시스템의 순차회로에서 가장 큰 전력 낭비는 클록에 의한 전력 낭비이다. 클록은 스위칭 과정을 통해 많은 불필요한 게이트 활동을 초래하며 높은 부하를 갖는 경향이 있다. 그러므로 클록을 구분하고 제어하기 위한 클록 버퍼를 지닌 클록 네트워크를 필요로 한다. 현재 연구된 결과 디지털 구조에서 클록 신호가 시스템 전력의 15% - 45% 전력을 사용하는 것으로 알려져 있으며 클럭을 제어하여 전력 낭비를 줄일 수 있는 방법이 제안되었다[11,12]. 일반적으로 클록에 의한 전력 손실을 줄이는 방법은 다음과 같다[13].

(1) 마스터 클록의 부하 및 클록 네트워크에 요구하는 버퍼 수 감소.
(2) 슬레이브 클록을 수신하는 플립-플롭이 아이들(idle) 상태에서 트리거하지 않는다.
(3) 슬레이브 클록에 의해 트리거(trigger)되는 플립-플롭이 트리거 되지 않을 때의 사이클 조건을 무의조건(Don't care) 처리하여 단순화 시킨다.

2-2-1. 클록 게이팅 방법

동기 시스템에서 플립-플롭은 클록 신호의 특정 천이 동작에 의해서 트리거 된다. 또한 마스터 클록과 또 다른 클록인 슬레이브 클록도 플립-플롭의 트리거를 위해서 마스터 클록과 같은 특정 천이 동작을 제공해야 한다.

순차회로는 n개의 플립-플롭이 있고, 그들의 출력과 클록 입력을 Q_i 와 clk_i로 표현한다. 여기서 i=0, 1, ..., n-1 이다. 동기식 순차회로에서 플립-플롭들의 클럭인 clk_i는 마스터 클록 clk에 의해서 트리거 된다. 그런데 몇몇 아이들 사이클에서 마스터 클록으로부터의 영향을 받지 않기 위해서 슬레이브 클록을 사용하여야만 한다. 따라서 이 슬레이브 클록도 순차회로의 동기를 유지하기 위해서 마스터 클록에 동기되어야 한다.

2-2-2. 파생 클록 방법

파생 클록(derived clock) 방법은 클럭 게이팅 방법들 중에 한 가지 방법으로 위에 기술된 클록에 의한 전력 손실방법 중 (3)의 부분에 해당된다. 이러한 방법은 슬레이브 클록에 의해 트리거 되는 플립-플롭에 대해 슬레이브 클록에 의해 트리거 되지 않을 때의 사이클 조건을 무의조건(Don't care)으로 처리하여 동작 조건을 단순화 시킴으로서 전체 소모 전력을 감소시키는 방법이다.

2-3. 슬랙을 이용한 방법

슬랙 시간(slack time)은 시스템의 시작 시간(start time)과 데드라인(deadline)을 근거로 하여 태스크(task)의 수행이 데드라인보다 일찍 완료되어 프로세서가 아이들 상태(idle state)로 있는 시간을 말한다. 이러한 슬랙은 외부 슬랙(external slack)과 내부 슬랙(internal slack)으로 구분된다. 외부 슬랙이란 특정 태스크의 WCET(Worst Case Execution Time)를 기준으로 데드라인과 비교하여 계산된 아이들 시간(idle time)이다. 이러한 외부 슬랙은 태스크 수행 전에 값을 미리 알 수 있어 개시 전압(starting voltage)을 조절 할 수 있다. 내부 슬랙은 WCET와 실제 실행 시간(actual execution time)과의 차이로 코드상의 분기(branch)들로 인하여 전체 실행 시간이 다양하게 나타나며, 이로 인해 내부 슬랙이 결정 되어진다. 내부 슬랙은 태스크가 실행되지 전에는 계산이 되지 않으므로 개시 전압에 영향을 줄 수 없다. 따라서 모든 태스크의 속도는 외부 슬랙에 의존하여 결정되어 진다. 내부 슬랙을 이용하는 방법으로 컴파일 시간(compile time)에 실행해야 할 태스크들의 WCET를 기반으로 수

행할 때 내부적으로 발생되는 슬랙을 이용하는 방법이 제안되어 있다[14].

Ⅲ. 최소 전력 소모를 위한 주파수 선택

이동형 시스템에 사용되는 프로세서의 전력 소모를 최소화하기 위한 주파를 선택하기 위한 방법으로 소비 전력이 주파수와 직접 비례하는 특성을 이용하여 주파수를 낮출 수 있는 클럭 게이팅을 이용하여 프로세서를 설계한다. 다음으로는 에너지의 이득과 손실을 고려하여 태스크의 실행 시간에 만족하는 범위내에서 프로세서의 속도를 결정하고, 프로세서만을 사용하는 태스크의 경우에는 슬랙 시간을 이용하여 프로세서의 속도를 낮추어 저전력을 구현한다. 이러한 방법들은 다음과 같다.

3-1. 클럭 게이팅 방법

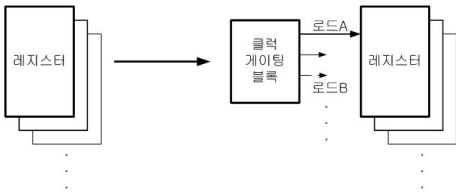
클럭 게이팅을 적용하기 위해서는 우선 클럭 게이팅 블록을 형성하고 이를 레지스터에 적용하여 저전력의 프로세서를 수현할 수 있다.

3-1-1. 클록 게이팅 블록

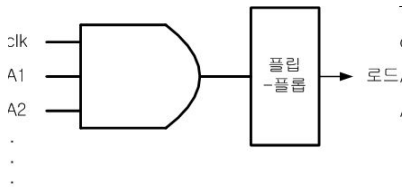
적용한 클록 게이팅 방법은 구조적 측면에서 프로세서의 RTL을 설계할 때 클록 게이팅을 통해 슬레이브 클록을 구현함으로서 전전력의 프로세서를 구현하였다.

<그림 1>(a)에서는 클록 게이팅 블록(Clock gating block)에 의해 각각의 레지스터(Register)를 위한 슬레이브 클록인 로드(load)신호들이 생성되는 것을 나타내었다. <그림 1>(a)에서 클록 게이팅 블록이 각각의 레지스터에 고유의 로드 신호를 생성할 수 있는 것은 <그림 1>(b)에서와 같이 외부 입력신호인 A1, A2 등의 외부 입력 신호들이 레지스터마다 서로 다른 고유의 신호를 가지고 있기 때문이다. 이들 외부 입력 신호의 값에 의해서 플립-플롭(Flip-Flop)은 로드신호를 생성하고, 레지스터의 슬레이브 클록으로 사용된다.

<그림 1>에서와 같이 레지스터의 클록에 의한 동작을 하기위한 블록으로 클록 게이팅 블록을 이용하고, 이를 통해 생성된 로드 신호에 의해 데이터를 저장한다.



(a) 클럭 게이팅 블록을 이용한 로드 신호 생성



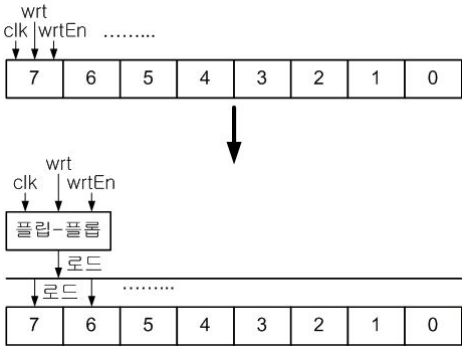
(b) 클럭 게이팅 블록의 구조

〈그림 1〉 클럭 게이팅 블록을 이용한 기본 다이아그램

3-1-2. 레지스터 적용

<그림 2>는 기존의 레지스터 입력과 클록 게이팅에 의해 생성된 로드 신호인 슬레이브 클록쉴호인 된 레지스터의 입력 구조를 나타낸다.

레지스터의 클럭은 크기에 무관하게 존재한다. 또한



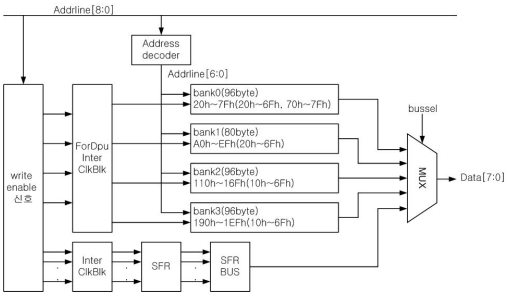
〈그림 2〉 클럭 게이팅을 이용한 레지스터

레지스터를 구성하는 플립-플롭은 클럭 뿐만아니라 외부 입력 신호들도 존재한다. 이는 비트 단위당 클록과 외부 입력 신호가 존재하는 것을 의미한다. 따라서 8비트 레지스터는 8개의 클록과 n(외부입력신호수) × 8개의 외부 입력 신호선이 존재한다. <그림 2>는 (8×Clock + n ×8) 개의 레지스터 입력 신호수를 클록 게이팅 블록을 통해 생성된 슬레이브 클록으로 레지스터 입력의 수를 단일화하고 입력 신호수를 줄였다. <그림 2>에서 레지스터는 write enable 인 wrtEn 신호와 write 인 wrt 신호를 입력으로 가지고 있고 이를 클록에 동기시켜 슬레이브 클록인 로드신호를 생성한다.

본 논문에서 적용하고자 하는 저전력 프로세서 데이터 패스 설계 기법은 레지스터 입력신호인 클록과 외부 입력 신호간의 논리식을 통해 공통항을 생성하고 클록을 효율적으로 통제하기 위한 방법으로 클록 게이팅을 적용하는 것이다. 이러한 적용 방법에서 가장 중요한 것은 클럭 게이팅 블록의 역할이다. 기존의 레지스터는 데이터 쓰기(write) 동작이외에도 클록이 공급되도록 되어 있으나 이러한 것은 클럭을 소모하게 되며 이에 따라 전력 소모가 발생되어 진다. 따라서 이러한 클럭을 줄이기 위해서는 특정 조건외의 클록에 대해서는 로드 신호를 '0' 으로 만들고 트리거가 발생하지 않도록 한다. 이러한 방법은 클록이 차지하는 전력 낭비를 근원적으로 줄이게 된다.

3-1-3. 데이터 패스 응용

<그림 3>은 메모리 블록을 나타내었다. <그림 3>의 "InterClkBlk" 블록은 SFR(Special Function Regiter)을 위한 클록 게이팅 블록이고, "ForDpuInterClkBlk"는 GPR(Global Purpose Register)을 위한 클록 게이팅 블록이다. 각각의 클록 게이팅 블록은 이미 <그림 1>과 <그림 2>를 통해 나타내었듯 각각의 레지스터에 고유의 슬레이브 클록을 생성하고 이를 통해 클록을 제어함으로써 전력손실을 줄일 수 있다.



〈그림 3〉 프로세서 데이터 패스 블록

3-2. 프로세서 속도 결정 방법

프로세서의 속도를 결정하는 것은 프로세서의 속도 변화에 따른 에너지 이득과 손실을 구하여야 한다. 에너지 이득과 손실은 각각 식 (2)와 식 (3)에 나타내었다[15].

$$E_{gain} = P_{proc}^{now} t - P_{proc}^{fix} t' = \frac{(f_{now})^2 - (f_{fix})^2}{(f_{now})^2} P_{proc}^{now} t \qquad (2)$$

$$E_{loss} = P_{proc}^{fix} t' - P_{proc}^{now} t = \frac{(f_{fix})^2 - (f_{now})^2}{(f_{now})^2} P_{proc}^{now} t \qquad (3)$$

E_{gain}은 프로세서의 에너지 이득을 나타내며, P_{proc}^{now}는 현재 시점에서의 단위시간당 프로세서의 전력소모량을 나타낸다. P_{proc}^{fix}는 속도를 조절했을 때의 단위시간당 프로세서의 전력소모량을 나타내며, f_{now}는 현재시점 프로세서의 주파수를 나타내고 f_{fix}는 프로세서의 속도를 조절했을 때의 주파수를 나타낸다. E_{loss}는 프로세서의 에너지 손실을 나타낸다. 식 (2)와 식 (3)을 이용하여 에너지 이득과 손실의 차이($\triangle E$)를 구한다. $\triangle E$를 구하는 식은 식 (4)에 나타내었다.

$$\triangle E = E_{gain} - E_{loss} = 2\frac{(f_{now})^2 - (f_{fix})^2}{(f_{now})^2} P_{proc}^{now} t \qquad (4)$$

식 (4)에서 P_{proc}^{now}와 f_{now}, t는 상수 값으로 알 수 있는 값이다. 따라서 시스템이 지원하는 f_{fix}값들을 입력하여

$\triangle E$의 값이 최대가 되는 f_{fix}를 구하여야 한다. f_{fix}의 값은 태스크의 시간 제약 조건을 만족하는 범위에서 최대의 값을 갖는 주파수이다. 이를 수식으로 표현하면 식 (5)와 같다.

$$\max|f_{fix}| \le TC_{task} \quad (5)$$

식 (5)에서 TC_{task}는 태스크의 시간제약 조건(time constraint)을 나타낸다.

위와 같은 방법은 이동형 시스템에서 프로세서와 디바이스를 같이 사용할 경우에 효율적으로 전력 소모를 줄일 수 있는 주파수를 결정하는 방법이며, 디바이스를 사용하지 않는 태스크의 경우에는 슬랙 시간을 이용하여 프로세서의 전력 소모를 줄일 수 있다.

Ⅳ. 실험 결과

본 논문에서 제안한 주파수 선택 알고리즘의 효율성을 입증하기 위한 방법으로 클럭 게이팅을 적용한 프로세서에 f_{fix} 값의 변화에 따른 소모 전력과 면적을 비교하여 최소 전력 소모를 갖는 주파수와 최적의 주파수를 선택하였다. 전력 비교 값은 Altera의 Quartus Ⅱ를 이용하였으며 회로구현 소자는 Stratix-EP1S10F484C5로 하여 얻은 결과이다. 소모 전력에 대한 결과 값은 <표 1>에 나타내었으며, 면적에 대한 결과 값은 <표 2>에 나타내었다. <표 2>의 면적은 사용된 LE의 수/전체 LE의 수로 표시하였다.

〈표 1〉 소모 전력 비교

주파수 (f_{fix}) ㎒	소모전력 ㎽		평균과 비교 ㎽	
	알고리즘 적용 전	알고리즘 적용 후	알고리즘 적용 전	알고리즘 적용 후
29.2	204.56	196.95	-0.67	-0.09
29.4	204.44	197.02	-0.79	-0.02
29.6	204.49	196.87	-0.74	-0.17
29.8	204.61	196.63	-0.62	-0.41
30.0	205.03	196.66	-0.20	-0.38
30.2	205.44	196.62	+0.21	-0.42
30.4	205.98	197.46	+0.75	+0.42
30.6	207.31	198.11	+2.08	+1.07
평균	205.23	197.04		

〈표 2〉 면적 비교

주파수 (f_{fix}) ㎒	면적		평균과 비교	
	알고리즘 적용 전	알고리즘 적용 후	알고리즘 적용 전	알고리즘 적용 후
29.2	6,285/10,570 (59.46%)	5,946/10,570 (56.25%)	-0.17%	+0.56%
29.4	6,302/10,570 (59.62%)	5,876/10,570 (55.59%)	-0.01%	-0.10%
29.6	6,252/10,570 (59.15%)	5,911/10,570 (55.92%)	-0.48%	+0.23%
29.8	6,255/10,570 (59.18%)	5,864/10,570 (55.48%)	-0.45%	-0.21%
30.0	6,311/10,570 (59.71%)	5,878/10,570 (55.61%)	+0.08%	-0.08%
30.2	6,389/10,570 (60.44%)	5,789/10,570 (54.77%)	+0.81%	-0.92%
30.4	6,287/10,570 (59.48%)	5,942/10,570 (56.22%)	-0.15%	+0.53%
30.6	6,345/10,570 (60.03%)	5,887/10,570 (55.69%)	+0.40%	-
평균	6,303/10,570 (59.63%)	5,887/10,570 (55.69%)		

실험 결과 소모 전력의 경우 알고리즘 적용 전에는 주파수가 29.4㎒일 때 204.44㎽로 평균값에 비해 0.79㎽ 작은 값을 나타내어 최소의 전력 소모를 갖는 주파수로 선정되었다. 이에 반해 본 논문에서 제안한 알고리즘을 적용한 후의 결과 주파수가 30.2㎒일 경우 196.62㎽로 평균값에 비해 0.42㎽ 작은 최소의 소모 전력을 갖는 것으로 나타났다. <표 1>의 결과 주파의 경우 평균값에서는 알고리즘을 적용 한 후의 평균 소모 전력이 적용 전에 비하여 205.23㎽에서 197.04㎽로 4% 감소된 결과를 나타내

었다. 주파수의 경우도 29.4㎒에서 30.2㎒로 향상된 결과를 나타내었다.

면적에서는 알고리즘 적용전의 경우 29.6㎒일 때 59.15%로 평균값에 비해 0.45% 감소된 최소의 면적을 갖는 주파수로 나타났으며, 알고리즘 적용 후에는 30.2㎒일 경우에 54.77%로 평균값에 비해 0.92% 최소의 면적을 갖는 것으로 나타났다. 면적의 경우 평균 3.94% 감소된 결과를 나타내었다. 이러한 실험 결과를 토대로 선정한 결과 30.2㎒의 경우에 소모 전력과 면적에서 최적의 결과를 나타내는 것으로 나타나 최적의 주파수로 선정되었다.

Ⅴ. 결 론

본 논문에서는 이동형 시스템의 프로세서에 대한 최소 전력 소모를 위한 주파수 선택 알고리즘을 제안하였다.

제안한 방법은 클럭 게이팅 방법을 이용하여 저전력 프로세서를 설계한다. 클럭 게이팅 방법은 내장된 클럭 블록을 이용하여 주 클럭을 제어함으로서 전력 낭비를 개선시킨다. 설계 방법은 프로세서에 대해 동적 전력을 고려하여 소모 전력을 비교하고, 설계된 프로세서에 대해 에너지 이득과 소모를 고려하여 주파수를 결정한다. 또한, 슬랙 시간을 이용하여 프로세서의 속도를 낮추어 소모 전력을 감소시킨다.

이러한 기술은 클럭 게이팅 방법과 에너지, 슬랙 시간을 이용하여 이동형 시스템의 사용 시간이 개선하였다. 실험 결과 제안한 알고리즘은 알고리즘을 적용하지 않은 이동형 시스템의 프로세서에 비해 평균 전력이 4% 감소되었다. 또한 주파수의 경우 29.4㎒에서 30.2㎒로 향상되었으며, 면적은 평균 3.94% 감소된 결과를 나타내었다.

따라서 본 논문에서 제안한 방법의 효율성이 입증되었으며 시스템이 클수록 전력 감소량이 커질 것으로 기대되며, 유비쿼터스나 휴대용 기기의 소형화와 휴대성에 기여할 것으로 본다.

참고문헌

[1] A. Chandrakasan, R. Brodersen, "Low power digital CMOS design," Kluwer Academic Publishers, 1995.

[2] Qing Wu, Massoud Pedram, Xunwei Wu, "Clock-Gating and Its Application to Low Power Design of Sequential Circuits," IEEE Custom Interated Circuits Conference, 1997, pp.479-482.

[3] D. Garrett, M. Stan, and A. Dean, "Challenges in clock gating for a low-power ASIC methodology," in Proc. ISLPED, San Diego, CA, August, 1999, pp. 176-181.

[4] T.Mudge, "power: a first-class architectural design constraint," IEEE COMPUT., vol. 34, no. 4, April, 2001, pp.52-58.

[5] Pietro Babighian, Enrico Macii, "A Scalable Algorithm for RTL Insertion of Gated Clocks Based on ODCs Computation," IEEE transactions on Computer-Aided Design of Integrated Circuits And Systems, vol. 24, no. 1, Jaunuary 2005, pp.29-42.

[6] Padamnabhan Pillai, Kang G. Shin, "Real time Dynamic voltage scaling for low power embedded operating system," In Proceeding of the 18th ACM symposium on Operating System Principles (SOSP-01), 2001, pp. 89-102.

[7] A. P. Chandrakasan, S. Sheng, and R.W.Brodersen, "Low-Power CMOS digital design," IEEE J. of Solid-State Circuits, 1992, pp. 473-484.

[8] A. Chandrakasan, R. Brodersen, "Low power digital CMOS design," Kluwer Academic Publishers, 1995.

[9] 조준동, 임세진, "소자의 스위칭 동작 최소화를 통한

디지털 회로 저전력 상위 레벨 최적화에 대한 연구," 서울대 반도체 공동연구소, 10월, 1997년.
[10] 조준동, "알고리즘 및 아키텍처 수준 저전력 설계자동화," 전자공학회, CAD 기술 특집, 12월, 1997.
[11] M. Pedram, "Power minimization in IC Design: Principles and applications," ACM Transactions on Design Automaton, vol. 1, no, 1, Jan. 1996, pp.3-56.
[12] G. Friedman, "Clock distribution design in VLSI circuits: an overview," in Proc. IEEE ISCAS, San Jose, May 1994, pp.1475-1478.
[13] Qing Wu, Massoud Pedram, Xunwei Wu, "Clock-Gating and Its Application to Low Power Design of Sequential Circuits," IEEE Custom Interated Circuits Conference, 1997, pp.479-482.
[14] Johan Pouwelse, Koen Langendoen, Henk Sips, "Energy priority scheduling for variable voltage processors," In Proceeding of Low Power Electronics and Design International Symposium, 2001, pp. 8-33.
[15] 하란, "이동형 시스템에서의 저전력 기법", 홍익대학교 과학기술연구논문집 Vol 15, 2004, pp. 177-185.

■ 저자소개 ■

김재진
Kim, Jae Jin

2001년~현재
극동정보대학 컴퓨터정보과 교수
2003년 2월 청주대학교 전자공학과(공학박사)
1995년 8월 청주대학교 전자공학과(공학석사)
1993년 2월 청주대학교 전자공학과(공학사)

관심분야 : CAD 알고리즘, 저전력 알고리즘
E-mail : dkimjj@kdc.ac.kr

강진구
Kang, Jin Gu

2007년 9월~현재
극동정보대학 애니메이션과 교수
2001년~2007년 8월
극동정보대학 방송영상미디어과 교수
2001년 2월 부산대학교 전자공학과 공학박사
1998년 2월 부산대학교 전자공학과(공학석사)
1995년 2월 금오공과대학교 전자공학과 제어공학전공(공학사)

관심분야 : 제어, 알고리즘
E-mail : jgukang@kdc.ac.kr

허화라
Hur, Hwa Ra

2000년~현재
송호대학 컴퓨터정보과 교수
2001년 8월 부산대학교 전자공학과(공학박사)
1992년 2월 동아대학교 컴퓨터공학과 (공학석사)
1988년 2월 부경대학교 전기공학과(공학사)

관심분야 : 알고리즘, 컴퓨터구조
E-mail : hrhur@songho.ac.kr

윤충모
Yun, Choong Mo

2000년~현재
서일대학 정보기술학부 교수
2000년 2월 청주대학교 전자공학과(공학박사)
1990년 2월 단국대학교 전자과(공학석사)
1985년 2월 서울산업대학 전자공학과(공학사)
1981년 3월~1992년 8월
서울시청 전산실(EDPS) 근무

관심분야 : 알고리즘, 컴퓨터구조
E-mail : 5420chong@seoil.ac.kr

논문접수일 : 2008년 2월 28일
게재확정일 : 2008년 3월 10일