

이동기기에서 배터리를 고려한 저전력 알고리즘 연구

김재진*

A Study on Battery Driven Low Power Algorithm in Mobile Device

Jae-Jin Kim*

요약

본 논문에서는 이동기기에서 배터리를 고려한 저전력 알고리즘을 제안하였다. 제안한 알고리즘은 이동기기에서 배터리의 잔량에 따라 태스크를 수행할 수 있도록 소모전력을 줄이는 방법으로 주파수를 변화시킨다. 배터리의 잔량이 수행하고자하는 태스크의 소모 전력보다 작을 경우 태스크를 수행시키기 위해 주파수를 낮춘다. 주파수 조절은 태스크에 따라 전체 시스템에서 사용되는 디바이스들 중에서 가장 높은 주파수를 가진 디바이스를 우선대상으로 수행한다. 주파수의 감소는 두 번째로 큰 주파수를 가진 디바이스의 주파수로 변화하여 소모전력을 계산한다. 계산된 소모전력이 배터리의 잔량보다 클 경우 단계적으로 주파수를 조절하여 소모전력을 낮추며 배터리의 잔량으로 태스크를 수행 할 수 있을 때 까지 주파수를 조절한다. 실험은 주파수를 조절하여 소모 전력을 줄이는 방법을 이용한 [6]과 같은 환경에서 배터리의 잔량을 고려하여 태스크를 수행할 수 있도록 주파수를 조절하였다. 실험 결과 [6]에서 수행되지 않는 배터리 잔량에서도 태스크가 동작되는 결과를 나타내었다.

▶ Keyword : 이동기기, 저전력, 태스크, 주파수, 배터리

Abstract

In this paper, we proposed battery driven low power algorithm in mobile device. Algorithm the mobile devices in power of the battery for the task to perform power consumption to reduce the frequency alters. Power of the battery perform to a task power consumption of is less than the task perform to frequency the lower. Frequency control the task, depending on in the entire system devices used among the highest frequency with devices first target perform to. Frequency in the decrease the second largest frequency with of the device the frequency in changes the power consumption to calculate. The calculated consumption power the battery of level is greater than

• 제1저자 : 김재진

• 투고일 : 2010. 07. 16, 심사일 : 2010. 09. 19, 게재확정일 : 2010. 09. 28.

* 극동정보대학 컴퓨터정보과 부교수(Dept. of Computer Information, Keukdong College)

※ 극동정보대학 2010학년도 교내학술연구비의 지원받아 게재함

level the frequency by adjusting power consumption, lower power of the battery the task perform when you can to the frequency to adjust. Experiment the frequency by adjusting power consumption a method to reduce using [6] and in the same environment power of the battery consider the task to perform frequency were controlled. The results in [6] perform does not battery power on task operates that the result was.

▶ Keyword : a mobile device, low power, task, frequency, battery

1. 서론

최근 이동기기의 발전과 함께 동일한 배터리를 이용하여 사용시간을 증가시키기 위한 많은 노력과 연구가 병행되었다. 이러한 연구는 저전력 프로세서 개발과 저전력 디바이스의 사용을 통하여 저전력을 구현하는 방법이 대부분이었다. 통신 시스템에서 소모 전력을 줄이기 위한 방법으로 저전력의 프로세서를 이용하여 저전력의 회로를 구현하고 있으나 이에 반해 디바이스(device)들의 전력 소모는 증가되고 있다. 프로세서의 소모 전력을 줄이기 위한 방법으로 클럭 게이팅(clock gating)을 이용한 방법이 제안되었다[1-3]. 또한 배터리의 상태를 실시간으로 점검하여 배터리의 상태에 따라 QoS를 조절함으로써 배터리의 사용시간을 개선한 방법을 제안하였다[4]. 그러나 이와 같은 연구는 많은 연구기간과 비용이 많이 들어가는 단점을 가지고 있다.

본 논문에서는 이러한 단점을 보완하여 배터리의 잔량을 고려하여 태스크를 동작시킬 수 있는 알고리즘을 제안하였다. 제안하고자 하는 알고리즘은 배터리의 잔량에 따라 수행할 수 있는 태스크의 수가 한정적이라는 제한 조건에 따라 배터리의 잔량이 수행해야할 태스크의 소모전력보다 작을 경우 태스크의 소모 전력을 배터리의 잔량범위 안에서 이루어 질 수 있도록 태스크의 주파수를 조절하였다.

주파수 조절은 태스크에서 수행해야할 디바이스들 중에서 가장 높은 주파수를 대상으로 아래 단계의 주파수 값을 가진 디바이스의 주파수 값으로 단계별로 조절하여 주파수를 조절하여 배터리의 잔량에 따라 태스크를 수행 할 수 있도록 한다.

II. 관련연구

소모전력을 감소시켜 저전력의 회로를 구현하기위한 알고리즘들이 많이 제안되었다. 이러한 알고리즘들 중에서 최근에 제안되었으며 태스크와 주파수를 고려한 알고리즘으로 [6]의 알고리즘이 있다[6]. 알고리즘은 태스크에 따라 저전력 회로

를 구현하기 위해 각 태스크에 따라 동작되는 회로의 특성을 분석한다. 회로는 크게 프로세서와 주변 입출력 장치를 포함한 디바이스로 분리하여 볼 수 있다. 따라서 태스크에 따라 3 가지 상태로 구분되어 볼 수 있다. 첫 번째는 프로세서만을 이용하는 경우이며, 두 번째는 디바이스만을 이용하는 경우, 세 번째는 프로세서와 디바이스 모두를 사용하는 경우이다

2.1 프로세서에만 적용

태스크의 종류에 따라 프로세서만 동작되는 태스크는 프로세서에만 저전력 알고리즘을 적용하면 된다. 이러한 적용을 위한 조건은 다음의 <조건 1>과 <조건2>에 나타내었다.

<조건 1> 태스크의 마감 시간 제약 조건(deadline constraint)보다 주어진 시간 제약 조건(time constraint)이 클 경우

$$t_{dead} < t_{time}$$

t_{dead} :태스크의 마감 시간 제약 조건 (deadline constraint)

t_{time} :시간 제약 조건(time constraint)

<조건 2> 디바이스의 sleep 상태에서 동작 상태로 전환하는 시간과 동작 상태에서 sleep 상태로 전환하는 시간의 합이 디바이스의 idle 시간보다 클 경우

$$t_{idle,i} < t_{wu,i} + t_{sd,i}$$

$t_{idle,i}$: 장치 k_i 의 idle 시간

이와 같은 조건의 경우 프로세서에만 저전력을 적용하며 DVS(Dynamic Voltage Scaling) 알고리즘을 적용하였다.

2.2 디바이스에만 적용

디바이스에만 저전력 알고리즘을 적용하는 경우는 <조건 3>과 <조건 4>를 만족하는 경우에 한하여 적용한다.

<조건 3> 태스크의 마감 시간 제약 조건(deadline constraint)보다 주어진 시간 제약 조건(time constraint)이 같을 경우

$$t_{dead} = t_{time}$$

<조건 4> 입출력 장치가 sleep 상태에 있기 위해서는 sleep과 동작 상태로의 서로 전환하는 시간의 합보다 클 경우에만 가능하다.

$$t_{idle,i} > t_{wu,i} + t_{sd,i}$$

 $t_{idle,i}$: 장치 k_i 의 idle 시간

이와 같은 조건을 만족하는 디바이스에만 DPM(Dynamic Power Management) 알고리즘을 적용하였다.

2.3 프로세서와 디바이스 모두 적용

태스크의 동작에 따라 프로세서와 디바이스를 모두 사용하는 경우는 아래 <조건 5>를 만족하는 경우이다.

<조건 5> <조건 1>과 <조건 4>를 모두 만족하는 태스크 일 경우

이러한 경우에는 F_DVS(Frequency Dynamic Voltage Scaling) 알고리즘과 Comb_DPM(Combination Dynamic Power Management) 알고리즘을 모두 적용하여 최소의 소모 전력을 갖도록 구성하였다.

III. 배터리를 고려한 저전력 알고리즘

태스크의 종류에 따라 배터리의 잔량보다 사용할 수 있는 디바이스의 수가 적을 경우에는 무관하나, 사용하여야 할 디바이스의 수가 많을 경우에는 태스크를 수행하기 어려운 점이 많다. 이러한 문제를 해결하기 위해 태스크의 소모 전력을 배터리의 잔량범위 안에서 이루어 질 수 있도록 태스크의 주파수를 조절한다. 주파수는 전력소모와 비례하므로 주파수를 조절하여 전력소모를 조절할 수 있으며 관계식은 (1)과 같다 [6].

$$P_{processor} = C_{eff} \cdot V_{dd}^2 \cdot f + P_{intrinsic} + P_{leakage} \quad \dots(1)$$

$P_{processor}$: 프로세서의 소모전력
 V_{dd} : 입력 전압
 f : 입력 주파수

C_{eff} : 스위칭 동작(switching activity)에 의한 유효 캐패시턴스(effective capacitance)

$P_{intrinsic}$: 프로세서의 고유 전력(intrinsic power)

$P_{leakage}$: 누설전력(leakage power)

주파수를 조절 하는 대상은 다음과 같다.

$$\begin{cases} \text{주파수 조절 없이 실행} & BP_{rem} \geq P_{task}(i) \\ \text{주파수조절} & BP_{rem} < P_{task}(i) \end{cases} \dots(2)$$

BP_{rem} : 배터리 잔량

$P_{task}(i)$: i번째 태스크의 전력소모량

수식 (2)에서 i번째 태스크의 전력소모량은 i번째 태스크에서 수행되어야 할 디바이스들의 전력 소모량으로 계산되어지며 수식 (3)에 나타내었다.

$$P_{task}(i) = LD_{task(i)}(j) \times DP[LD_{task(i)}(j)] \dots (3)$$

$LD_{task(i)}(j)$: i번째 태스크에서 수행되어야 할 디바이스 목록

$$LD_{task(i)}(j) = LD_{task_{com}}(j) + LD_{task_{seq}}(j)$$

$LD_{task_{com}}(j)$: task(i)에서 사용하는 조합논리회로 목록

$LD_{task_{seq}}(j)$: task(i)에서 사용하는 순차회로 목록

$DP[LD_{task(i)}(j)]$: i번째 태스크에서 수행되어야 할 디바이스들의 전력소모량

$$DP[LD_{task(i)}(j)] = DP[LD_{task_{com}}(j)] + DP[LD_{task_{seq}}(j)]$$

$DP[LD_{task_{com}}(j)]$: task(i)에서 사용하는 조합논리회로 $LD_{task_{com}}(j)$ 의 전력소모량

$DP[LD_{task_{seq}}(j)]$: task(i)에서 사용하는 순차회로 $LD_{task_{seq}}(j)$ 의 전력소모량

수식 (2)에서 주파수 조절 대상 태스크의 경우 주파수의 조절을 배터리 잔량으로 태스크를 수행 할 수 있는 범위의 최대 주파수로 조절하여야 한다. 주파수 조절은 i번째 태스크에서 수행해야할 디바이스들 중에서 가장 높은 주파수를 대상으로 두 번째 주파수 값을 가진 디바이스의 주파수 값으로 조절한다. 이러한 조절은 주파수 조절 대상 디바이스의 선택을 가장 큰 값을 가진 디바이스부터 작은 값을 가진 디바이스의 순서로 자연스럽게 결정될 수 있는 장점을 가지고 있다. 디바이

스가 가지고 있는 주파수 값을 $f_{LD_{task(i)}}$ 라 정의 하며 $f_{LD_{task(i),min(j)}} \leq f_{LD_{task(i)}} \leq f_{LD_{task(i),max(j)}}$ 이다. 이러한 조건은 수식 (4)에 나타내었다.

$$\alpha = \frac{f_{LD_{task(i),max(j-1)}}}{f_{LD_{task(i),max(j)}}} \dots (4)$$

$f_{LD_{task(i),max(j)}}$: i 번째 태스크에서 수행해야할 디바이스들 중에서 가장 높은 주파수
 $f_{LD_{task(i),max(j-1)}}$: i 번째 태스크에서 수행해야할 디바이스들 중에서 두 번째로 높은 주파수

위와 같은 수식을 이용하여 제한한 배터리를 고려한 저전력 알고리즘은 그림 1에 나타내었다.

Battery Driven Low Power Algorithm

($LD_{task(i)}(j)$, $DP[LD_{task(i)}(j)]$,
 $f_{LD_{task(i)}}$)

if($BP_{rem} < P_{task}(i)$)

for($f_{LD_{task(i),max(j)}}$; $f_{LD_{task(i),min(j)}}$; $j--$)

/* $f_{LD_{task(i),min(j)}}$: i 번째 태스크에서 수행해야할 디바이스들 중에서 가장 낮은 주파수 */

$\alpha = \frac{f_{LD_{task(i),max(j-1)}}}{f_{LD_{task(i),max(j)}}$;

$ADP[LD_{task(i)}(j)] = f_{LD_{task(i)}}(j) \times \alpha$;

/* $ADP[LD_{task(i)}(j)]$: 주파수 변경에 따른 변화된 디바이스의 전력 소모량 */

$P_{task}(i) = LD_{task(i)}(j) \times ADP[LD_{task(i)}(j)]$;

if($BP_{rem} \geq P_{task}(i)$)

process;

else

return;

end

그림 1. 배터리를 고려한 저전력 알고리즘
 Fig 1. Low power algorithm driven the battery

그림 1의 배터리를 고려한 저전력 알고리즘은 수행할 태스크의 동작과 디바이스, 주파수, 신호들의 정보를 입력받아 배터리의 잔량과 수행해야할 태스크의 소모 전력을 비교하여 태스크의 소모전력이 클 경우에 저전력을 수행하여 태스크가 수행될 수 있도록 구성되었다. 동작 순서는 태스크의 소모 전력

이 배터리의 잔량보다 클 경우 태스크를 배터리의 잔량 범위에서 동작이 가능하도록 주파수를 변경시켜 소모 전력을 낮추는 것이다. 이러한 주파수의 변경은 태스크를 수행할 때 사용되는 디바이스들 중에서 가장 높은 주파수를 가진 디바이스와 두 번째 크기의 주파수를 가진 디바이스를 선택하여 모든 디바이스를 두 번째 크기의 주파수로 변경하여 소모전력을 계산하고 배터리의 잔량으로 태스크를 수행할 수 있을 때까지 수행된다.

그림 1의 저전력 알고리즘의 적용 예로 그림 2의 데이터패스를 선정하여 알고리즘을 적용하였다. 그림 2의 데이터패스는 3개의 레지스터와 1개의 멀티플렉서, 1개의 가산기로 구성된 데이터패스이다. 이러한 데이터패스의 소모 전력은 식 (5)와 같다.

$$P_{DF} = 3 \cdot P_{reg} + P_{MUX} + P_{adder} \dots (5)$$

P_{DF} : 데이터 패스의 소모전력
 P_{reg} : 레지스터의 소모전력
 P_{MUX} : MUX의 소모전력
 P_{adder} : 가산기의 소모전력

태스크에 따라 소모 전력의 차이가 발생되면 1회 가산의 태스크는 식 (5)와 같은 값이 되며, $n \times m$ 의 연산 태스크의 경우 소모 전력의 값은 식 (6)과 같다.

$$P_{task} = (3+m) \cdot P_{reg} + m \cdot P_{MUX} + m \cdot P_{adder} \dots (6)$$

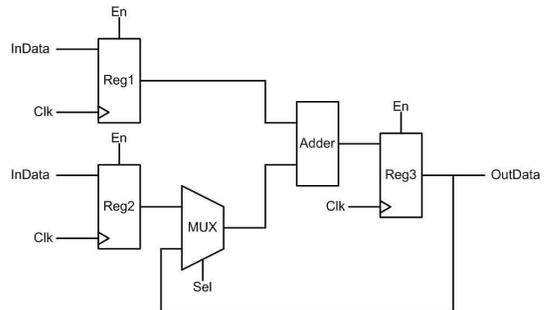


그림 2. 데이터패스 예
 Fig 2. Example of datapath

식 (6)에서 클럭이 인가되는 레지스터의 경우는 $(3+m)$ 배의 소모 전력을 가진다. 따라서 레지스터의 주파수를 1/2로 감소시키면 식 (6)의 소모 전력은 식 (7)과 같이 감소하게

된다.

$$P_{task} = \frac{(3+m)}{2} \cdot P_{reg} + m \cdot P_{MUX} + m \cdot P_{adder}$$

(7)

따라서 배터리의 잔량을 이용하여 태스크를 수행할 수 있도록 주파수를 조절하여 저전력의 회로를 구현한다.

IV. 실험결과

실험은 주파수를 조절하여 소모전력을 줄이는 방법을 이용한 [6]과 비교하였다. 같은 환경에서 배터리의 잔량과 주파수에 따른 소모 전력을 비교하였다. 실험에서 사용하는 프로세서는 AMD's mobile K6을 모델링하여 사용하였다[8]. 그 외의 연관 장치로는 하드디스크와 DSP, Flash로 구성하였다[9-11]. 실험 대상의 태스크에 대한 특성은 표 1에 나타내었으며 각각의 태스크가 수행될 때 사용되는 디바이스를 표시하였다. 표 2는 표 1의 각 디바이스에 대한 상태별 소모 전력을 나타내었다.

표 1. 태스크의 특성
Table. 1 Task character

태스크		T1	T2	T3	T4	T5
사용 디바이스	[9]	○	-	-	-	○
	[10]	-	-	○	○	○
	[11]	-	-	-	○	○

표 2. 연관장치의 소모 전력 특성
Table. 2 Device power consumption character

디바이스	[9]	[10]	[11]
동작시 소모전력(W)	0.95	0.63	0.13
Sleep시 소모전력(W)	0.13	0.2	0.001
Sleep 상태로 전환시 소모전력(W)	0.54	0.4	0.05
Sleep 상태에서 다른 상태로 전환시 소모전력(W)	1.61	0.4	0.05

T1의 경우 수식 (3)을 이용하여 프로세서의 주파수에 따른 소모 전력을 계산한 결과는 다음과 같다.

- 192MHz 경우

$$(0.13[W] \times 0.8[sec]) + 1.61[W] + (0.95[W] \times 2.6[sec]) + 0.54[W] + (0.13[W] \times 2[sec]) + (0.156[W] \times 4.8[sec]) = 5.7328 \approx 5.7[W]$$

- 588MHz 경우

$$4.984 + (2.352[W] \times 1.6[sec]) = 8.7472 \approx 8.75[W]$$

소모 전력이 가장 큰 T5의 경우 프로세서의 주파수에 따른 소모 전력은 다음과 같다.

- 192MHz 경우

$$(0.13[W] \times 0.4[sec]) + 1.61[W] + (0.95[W] \times 5.6[sec]) + 0.54[W] + (0.13[W] \times 6[sec]) + (0.2[W] \times 1.1[sec]) + 0.4[W] + (0.63[W] \times 2.8[sec]) + 0.4[W] + (0.2[W] \times 8.1[sec]) + (0.001[W] \times 0.8[sec]) + 0.05[W] + (0.13[W] \times 8.1[sec]) + 0.05[W] + (0.001[W] \times 3.1[sec]) + (0.156[W] \times 8.3[sec]) = 15.1577 \approx 15.2[W]$$

- 588MHz 경우

$$13.8629 + (2.352[W] \times 2.8[sec]) = 20.4485 \approx 20.45[W]$$

프로세서의 주파수 범위는 192-588MHz이며 주파수별 소모 전력은 표 3에 나타내었다.

표 3. 프로세서의 소모전력
Table. 3 Power consumption of processor

주파수(MHz)	소모전력(mW)
192	155.52
250	281.43
300	432
350	627.52
400	873.79
450	1176.6
500	1541.77
550	1975.06
588	2352

태스크에 따른 소모 전력 결과는 표 4에 나타내었다. 이동기기에서 사용되는 배터리는 2[W]에서 80[W]에 이른다. 그 외의 것도 있으나 본 논문에서는 80[W]의 배터리를

대상으로 실험을 수행하였으며 동작은 T1에서 T5까지 모든 태스크를 1회씩 수행하는 것으로 실험을 수행하였으며 실험 결과는 표 5에 나타내었다.

표 4. 태스크에 따른 소모 전력
Table. 4 Power consumption for task

태스크	소모전력[W]	
	최소	최대
T1	5.7	8.75
T2	2.02	9.1
T3	10.1	14.8
T4	8.9	12.4
T5	15.2	20.45
합계	41.92	65.5

표 5. 실험 결과
Table. 5 Experiment result

배터리 잔량	주파수(MHz)	소모전력(W)
100%	588	65.5
90%	588	65.5
80%	563	64
70%	429	55.96
60%	296	47.94

V. 결론

본 논문에서는 이동기에서 배터리의 잔량을 고려한 태스크를 수행할 수 있는 저전력 알고리즘을 제안하였다.

제안한 알고리즘은 이동기에서 배터리의 잔량에 따라 수행할 수 있는 태스크의 수가 한정적이라는 제한 조건에 따라 배터리의 잔량이 수행해야할 태스크의 소모 전력보다 적을 경우 태스크의 소모 전력을 배터리의 잔량범위 안에서 이루어 질 수 있도록 태스크의 주파수를 조절하여 저전력을 구현하였다.

주파수 조절은 태스크에서 수행해야할 디바이스들 중에서 가장 높은 주파수를 대상으로 아래 단계의 주파수 값을 가진 디바이스의 주파수 값으로 단계별로 조절한다.

실험은 주파수를 조절하여 소모 전력을 줄이는 방법을 이용한 [6]과 같은 환경에서 배터리의 잔량을 고려하여 태스크를 수행할 수 있도록 주파수를 조절하였다.

실험 결과 [6]에서 수행되지 않는 배터리 잔량에서도 태스크가 동작되는 결과를 나타내었다.

후회 과제로는 본 논문에서 제안한 알고리즘의 수행 시간을

개선하고 최대한 빠른 시간에 태스크가 수행될 수 있도록 주파수를 세분화 하여 조절할 수 있는 방법을 연구하는 것이다.

참고문헌

- [1] Qing Wu, Massoud Pedram, Xunwei Wu, "Clock-Gating and Its Application to Low Power Design of Sequential Circuits," IEEE Custom Interated Circuits Conference, 1997.
- [2] D. Garrett, M. Stan, and A. Dean, "Challenges in clock gating for a low-power ASIC methodology," in Proc. ISLPED, San Diego, CA, pp. 176-181, August, 1999
- [3] 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, pp. 29-42, January 2005
- [4] L. Benini, G. Castelli, A. Macii, R. Scarsi, "Battery-Driven Dynamic Power Management of Portable System", ISSS(International Symposium on System Synthesis), pp. 25-33, 2000.
- [5] D. Linden, T. Reddy, *Handbook of Batteries*, McGraw-Hill, 2001
- [6] Jae-Jin Kim, "A Study of Low Power Algorithm for a Task under the Time Constraint," Journal of Korean Institute of Information Technology, Vol. 7, No. 6, pp. 27-34, 2009.
- [7] Kwan-Houng Lee, Jin-Gu Kang, Jae-Jin Kim, "A Study on A Frequency Selection Algorithm for minimization Power Consumption of Processor in Mobile Communication System," Proceeding of The Korea Society of Computer Information, pp.25-32, 2008
- [8] "Mobile AMD'6 Processor Power Supply Application Note," www.amd.com
- [9] "MHL2300AT Hard Disk Drive Product Manual," www.fujitsu.com
- [10] "TMS320C6411 Power Consumption Summary," www.ti.com
- [11] "SST Multi-purpose Flash SST39LF020", www.sst.com

저 자 소개



김 재 진

2003년 2월: 청주대학교 전자공학과
공학박사

2001년~현재 :극동정보대학
컴퓨터정보과 부교수

관심분야 :저전력 알고리즘, 인공지능,
CAD 알고리즘, 매카트로닉스

e-mail : dkinjj@kdc.ac.kr