특집

알고리즘 및 아키텍쳐 수준 저전력 설계 자동화

조 준 동
성균관대학교 전자공학과

I. 서 론

최근의 건전지를 사용하여 동작하는 휴대용 통신 시스템(cellular phone, pagers, wireless modem등)에서 DSP(Digital Signal Processor)의 수요가 늘어나고 있으며 성능 개선 및 전력량감축을 위한 노력이 경주되고 있다. 그러한 무선 통신 디바이스의 사용 증가는 건전지 수명의 연장을 위한 embedded DSP를 필요로 하고 있으며 보다 복잡한 speech 와 channel coding 알고리즘의 계산을 위하여 많은 계산 산출량(computational throughput)을 요구하고 있다. 기능의 복잡도 및 고주파수에 의한 데이터 흐름은 스위칭 동작의 급속한 증가의 원인을 제공하고 있다. 각각의 스위칭에 의한 전기적 에너지는 열 및 전력 소모로 변환한다. Deep-submicron 공정의 발달로 성능이 지속적으로 향상되고 있으나 성능이 2배로 증가하면 전력소모도 2배 늘어나고 있는 추세이다. 따라서 저전력 설계는 휴대용 건전지로 동작되는 개인용 통신 및 멀티미디어 시스템에 필수적으로 적용되어야 할 기술이다. 또한 저전력 및 고산출량에 대한 요구는 휴대용 장비 뿐만 아니라 고성능 멀티미디어 시스템에도 필수적인 요소가 되고 있다.

디지털 CMOS VLSI의 동적 전력 소모는 다음의 식으로 표현된다.

$$P_D = \sum T_i C_i V_{dd}^2 f_s \qquad (1)$$

여기에서 T_i는 평균 천이동작수(transition activity), C_i는 hardware unit i에 대한 스위칭 정전용량, V_{DD}는 공급 전압, 그리고 f_s는 동작 주파수를 말한다. 그러한 전력 소모에 영향을 주는 요소들을 최소화하기 위하여 전력량 감축을 위한 노력은 설계의 모든 계층에서 고려되고 있다. 논리회로에는 pass logic, multi-threshold logic, swing suppression, charge recycling circuit, 그리고 adiabatic logic circuit[18] 등이 사용되고, LSI 아키텍쳐에서는 parallel, pipeline, switching capacitance reduction, algorithm 변환, 그리고

power management[1]등의 방법 등이 적용되고 있다. 또한 compilation, scheduling, resource-allocation[7]을 포함하는 high-level synthesis도 전력소모 감축에 크게 기여하고 있다. 특히 본 논고에서는 알고리즘 및 아키텍쳐 수준의 저전력 기법을 중심으로 최근의 DSP 및 통신용 subsystem에 대한 저전력 기법을 다룬다.

본 논문의 구성은 다음과 같다. 2절에서는 정전용량 감축을 통한 저전력 설계 기법을 다루고, 3절에서는 아키텍쳐 변환에 의한 전압 scaling, 4절에서는 부호화를 통한 bus activity 감소 기법을, 5절에서는 알고리즘 변형을 통한 전력 감축 기법에 대하여 설명한다. 마지막으로 6절에서는 저전력용 마이크로프로세서를 소개하고 7절에서 결론을 맺는다.

II. 정전용량 감축을 통한 저전력 설계

식 (1)에서 보인 전력소모의 원인을 제공하는 요소 중 정전용량은 다음의 식으로 표현된다.

$$C_i=\sum N_{res}C_{res}+N_{reg}C_{reg}+N_{mem}C_{mem} \quad (2)$$

알고리즘 수준에서는 operation 수를 줄이거나 algebraic 변환(예 : kernel extraction에 의한 common subexpression 제거)과 reduction of strength(예 : 곱하기 대신 shifter+adder)를 사용하여 정전용량을 줄인다. 또한 off-chip 메모리 operation은 연산과 비교하여 10배이상의 전력을 필요로 하기 때문에 off-chip 메모리 access를 줄이는 것이 무엇보다 중요하다. 따라서 알고리즘 수행을 위하여 필요로 하는 메모리 access를 줄이기 위하여 메모리 access는 processor 근처에서 이루어지도록 한다(locality of reference);즉, 레지스터, cache, RAM의 순서로 한다. 또한 single word load 보다는 multiple word parallel load를 이용하여 가용 bandwidth를 효과적으로 사용한다. 예를 들면, signal processing의 loop nesting과 operation ordering의 영향은 다음의 예를 통하여 알 수 있다.

Example 1:

Before:	After:
FOR I : =1 TO N DO	FOR I: =1 TO N DO
B[i] : =f(A[i]);	B[i]: =f(A[i]);
FOR I : 1 TO N DO	C[i]: =g(B[i]);
C[i] : =g(B[i]);	

Before의 경우를 보면, B array는 너무 커서 레지스터에 저장하기 어렵기 때문에 메모리 transfer가 필요하게 된다. 반면에 After 경우는 중간 값 $B[i]$가 레지스터에 보관될 수 있기 때문에 2N 메모리 transfer를 레지스터 access로 치환할 수 있게 된다.

소자의 스위칭 동작 최소화를 통한 디지탈 회로 저전력 아키텍쳐 수준, 논리합성[8]기법은[7]에 소개되었다. 곱셈기 또는 덧셈기를 time-sharing 방식으로 사용할 때, 연속된 두 입력값에 대한 상이한 bit 수를 Hamming Distance라고 한다. 평균 Hamming distance는 다음의 식으로 주어진다.

$$AHD(x)=\lim_{n\to\infty}\frac{1}{n}\cdot\sum_{n=1}^{\infty}H(x_i,x_{i-1}) \quad (3)$$

여기서 $H(x_i, x_{i-1})$는 두변수 x_i, x_{i-1}사이의 Hamming distance이고 x_i는 제어구간 i에서의 피연산자의 값이다. 전력 소모를 줄이기 위해서는 평균 Hamming distance를 줄여야 하며 그 값은 scheduling 및 resource sharing에 따라 변하게 된다. 그러므로 (3)식을 최소화하는 scheduling 및 resource-sharing 알고리즘이 제안되었다[12].

Clock network에서 소모되는 switching 정전용량은 전체의 15-30%를 차지한다. 그러므로 clock network에 의한 전력소모를 최소로 하기 위하여 local clock을 이용한 Gated Clock은 동기회로에서 상당한 전력 감축의 효과를 보인다. 그러나 testability를 감소시킨다는 단점이 있다.[3]은

gated clock의 finite state machine에서 single-stuck-at testability를 보장하는 방법을 제안하였다. 첫째 방법은 obsevability를 증가시키고 fully-testable gated clock을 얻기 위해서 redundancy-removal 기법과 연결되어 사용될 수 있다. 두 번째 방법은 redundancy removal이 가능하지 않은 observability와 controllability를 동시에 고려하여 large FSM에 적용하는 방법으로 fully-testable gated-clock FSMs를 보장한다. Deep submicron 설계시 Clock network의 설계는 고성능 시스템 설계시 매우 중요하며 clock skew 및 지연시간이 적어지도록 설계하여야 한다[5]. DEC의 Alpha 또는 Motolora의 PowerPC 설계시 칩레벨에서는 tapering된 H-tree 구조를 사용하고 distributed buffer 및 clock regenerator를 이용하여 clock을 분배하며 block내에는 Mesh형태를 갖는 복잡한 구조의 clock network topology가 사용되고 있다.

III. 아키텍쳐 변환에 의한 전압 Scaling

아키텍쳐 변환에 의한 전압 scaling 방식은 전력 감축에 큰 효과를 나타낸다. 전압 scaling을 위하여 알고리즘 변환 기법[30], pipelining, parallel processing, retiming[20], folding, unfolding, look-ahead, 그리고 교환, 분배 법칙에 의한 algebraic 변환 기법 등을 사용한다. CMOS gate의 지연시간 T_a은 다음과 같이 주어진다.

$$T_d=\frac{C_L V_{dd}}{I}=\frac{C_L V_{dd}}{k(W/L)(V_{dd}-V_T)^2} \quad (4)$$

여기에서 C_L은 gate 정전용량, I는 출력전류, V_T는 threshold voltage, k는 공정의존 변수, W와 L은 트랜지스터의 폭과 길이를 말한다. 즉 V_{dd}가 V_{th}에 접근함에 따라 T_a가 증가하는 것을 알 수 있다. 고성능 시스템의 설계시 성능 및 산출량을 보존하면서 전력량을 줄이는 것이 중요하므로 전압 축소에 따른 지연은 적절한 아키텍쳐 변형을 통하여 보상되어야만 한다. 그 방법에는 크게 parallel processing과 pipelining의 두 가지 방법이 있다. Parallel processing은 hardware unit을 복제(replication)하여 수행하므로 단일 프로세서를 이용할 때와 같은 산출량을 얻기 위해서는 각각의 프로세서의 필요한 주파수를 1/2로 감소할 수 있게 된다. 따라서 주파수 감소에 따른 지연에 의하여 필요한 전압은 2.9V로 감소하게 된다. Pipelining은 critical path를 latch로 분리하여 분리된 각각의 block을 동시에 수행하는 기법으로 critical path delay가 1/2로 줄어들면 산출량은 대략 두 배로 증가하게 된다. 그때 critical path delay의 감소로 인하여 전압은 식(4)에 의하여 2.9V로 낮춰지게 된다. 간단한 adder-comparator data path에 대하여 hardware 복제에 의한 병렬구조, pipelining, 그리고 그 두 가지를 조합한 pipelined 병렬구조를 적용한 결과는 표 1에 나타나 있다.

Retiming[20]은 data flow 그래프상의 latch의 위치를 변경함으로써 clock period 및 latch의 개수를 최소화하는 것이다. 저전력을 위한 retiming 기법은 모든 조합회로 및 latch의 출력 단자에서 $\sum T_i C_i$가 최소가 되도록 latch의 위치를 변경하는 것이다. 그렇게하기 위해서는 $T_i C_i$(T_i는 glitch를 포함하며 C_i는 노드 i의 loading 정전용량과 모든

〈표 1〉 아키텍쳐 변형에 대한 비교

Architecture	V_{dd}	C_{eff}	Area	Power	Frequency	Throughput
Reference	5.0V	1	1	1	1	1
Paralle	2.9V	2.15	3.4	0.36	1/2	1
Pipelined	2.9V	1.15	1.3	0.39	1	1
Pipelined Parallel	2.0V	2.50	4	0.2	1/4	1

fanout 노드의 입력 정전용량을 포함한다)가 큰 출력 노드에 latch를 사용하는 것이 효과적이다. 그것은 glitch를 제거하고 fanout node의 capacitive loading을 masking해주는 역할을 하여 전체 전력량을 감축시킨다. Glitch(spurious switching)는 시그널이 같은 시간에 도달하지 않기 때문에 발생하는 불필요한 swithing을 말하며 회로 전체 switching의 20%-70%에 해당한다. Logic path가 같도록(즉 differential path delay가 없도록) 설계하면 glitch를 제거해 줄 뿐만 아니라, 불필요한 전류 낭비를 제거할 수 있다. 그 것은 짧은 지연시간을 갖는 logic path는 필요 이상의 전류가 공급되어 전력을 낭비하기 때문이다. Logic path사이의 지연시간이 같도록 하여 불필요한 전류의 낭비를 제거하는 방법은 트랜지스터 sizing을 통하여 해결될 수 있다. 트랜지스터 크기는 switching 전류와 capacitive loads에 비례하므로 빠른 path상의 트랜지스터를 크기가 작은 것으로 치환하면 구동 전류 및 capacitive load가 줄어들어 그 path의 지연시간이 늘어나게 된다.

Unfolding은 software pipelining[33]이라고도 하며 data flow program에 내재되어 있는 concurrency를 이용 병렬처리를 하는 방법을 말한다. Folding 변환은 bit-parallel 아키텍쳐를 digit-serial 또는 bit-serial로, 또는 digit-parallel을 bit-serial로 변환한다.

Look-Ahead 변환 기법은 sequential recursive 알고리즘을 concurrent 알고리즘으로 변형하여 pipelining 또는 parallel processing을 가능하도록 하는 방법이다. 이 방법은 recursive digital filter, adaptive lattice digital filters, two-dimensional recursive digital filters, Viterbi decoders, 그리고 Huffman decoder에 성공적으로 적용되어 왔다. Filter의 difference equation은 $x(n)=ax(n-1)+u(n)$으로 표현하며 pipelining을 이용하여 이 시스템의 speed를 두 배로 증가시키기 위해서는 $x(n)=a[ax(n-2)+u(n-1)]+u(n)$로 표현된 시스템을 사용한다.

IV. 부호화를 통한 Bus Activity 감축

부호화를 이용한 I/O 시그널의 스위칭 동작회수를 줄이는 것은 전력소모 감축에 큰 효과가 있다. 따라서 data 및 address bus의 worst case와 평균 case transition을 최소화하는 부호화 알고리즘의 개발이 필요하다. Data bus 부호화의 한 예로 bus-inverting scheme이 있다. 그 알고리즘은 두 개의 연속된 word pattern의 Hamming distance를 비교하며 bus에 부가적인 line 하나를 추가적으로 사용한다.

1) 만일 현재 pattern과 다음 것의 Hamming distance가 n/2보다 적거나 같으면 그 pattern은 그대로 전송된다.
2) 만일 현재 pattern과 다음 것의 Hamming distance가 n/2보다 크면 그 pattern은 inversion 되어 전송된다. 이 방법을 이용하면 clock cycle당 transition의 수가 n/2를 초과 할 수 없게 된다. 이때, 평균 transition의 수는 binomial distribution에 의해 원래 값의 25%이하가 된다.

Address bus 부호화는 인접된 code가 한 bit씩 변하는 Gray code를 이용하는데 그 이유는 마이크로프로세서에 의해서 발생된 address는 memory 공간에 연속적으로 보관되기 때문이다. Gray code를 사용하여 37%의 전력감축 효과를 보일 수 있다. 단점으로는 Gray 부호화기가 필요하다는 것이다. Bus 정전용량이 0.5 pF 이상인 경우에 off-chip binary-to-Gray converter를 이용하면 binary code 보다 더욱 power-efficient한 결과를 얻는다.

DSP 아키텍쳐는 1개의 하드웨어 곱셈기와 두 개의 분리된 메모리를 사용하여 상수와 입력 값을 두 개의 독립적인 bus를 이용하여 동시에 access하기 위하여 Harvard 아키텍쳐를 사용하는 것이 전력 소모를 줄이는 데 효과적이다. 특히 상수 메모리 데이터 bus와 곱셈기에서의 스위칭 동작을 줄이는 것이 중요하다. 또한 시그널의 자체의 정전용량 이외에도 시스널과 시그널 사이에서 발생하

는 inter-signal 정전용량은 인접된 bit가 역상일 때 발생하게 된다. 예를 들면 (0101)에서 (1010)로 천이할 때 필요한 전류의 양은 (0000)에서 (1111)로 천이할 때의 전류량의 25%를 추가적으로 필요로 한다.[27] 따라서 temporal과 spatial한 2-Dimensional encoding 기법이 필요하게 된다.

데이터 전송시 데이터 입력 stream을 재정렬하여 스위칭 동작 회수(전송되는 bit의 천이수)를 줄이는 알고리즘이 [6]에 의하여 제안되었으며 2-Dimensional TSP(Traveling Salesman Problem)으로 문제를 정형화하여 접근하였다.

V. 알고리즘 변형을 통한 전력 감축

DSP의 VLSI 구현 방식은 짧은 설계 기간동안 power-area-speed 최적의 VLSI 시스템을 추구한다. Implementation styles로는 bit-serial, bit-parallel 그리고 digit-serial architectures등이 있다. DSP 시스템에서 Multimedia data의 특성은 DBT(Dual Bit Type)[22]로 표현된다. 즉, lower order bit는 data가 서로 독립적으로 상호 연관성(correlation)을 적게 갖고 있는 반면 high order bit는 상호연관성이 높다(positive correlation이라고 함). 연속된 두개의 data sample의 binary 표현은 많은 bit가 공통인 것을 알 수 있다. 그러나 delta modulation 같은 coding 방법을 사용할 때는 data 값이 큰 양수에서 큰 음수로 변하므로 상호연관성이 적다 (negative correlation이라고 함). Bit toggle frequency를 plot하면 LSB 영역인 경우는 maximum frequency의 반(0.5)이 되고 (연속된 데이터 sample이 totally uncorrelated된 경우를 말하며, uniform white noise라고 함) MSB (sign bit)의 경우는 negative correlation의 경우는 maximum frequency(1)가 되고 positive correlation의 경우는 0가 된다. Sign bit와 white noise 구간사이는 grey 영역이라고 하며 frequency가 0.5에서 +- 0.5 사이가 된다. Image data의 LSB는 다른 bit보다 더욱 많은 스위칭 동작을 하게 된다. 그 모델은 switch level simulator와 비교하여 15%의 차이를 보인다.

Viterbi decoder(화성 탐사선인 Pathfinder가 보내는 데이터와 그림은 Viterbi Decoder를 이용하여 decoding 되었다), motion compensation, 2D-filtering, 그리고 data transmission systems에서 저전력 설계 기법이 절실히 필요로 하고 있다. 실시간 video 또는 image data compression 방법중에 DCT (Discrete Cosine Transformation)는 압축된 pixel data를 전송하는 대신 motion vector와 prediction error가 부호화되어 전송된다. 이 방법은 temporal/spatial redundancy를 줄이고 macro-block을 표현하는 bit의 수가 줄어들게 된다. 신호처리 모듈중 motion estimation 모듈은 계산시간이 전체 시스템의 50%를 차지하고 그만큼의 전력을 소모하게 된다.

High prediction 정확도를 얻기 위해서 motion estimation에서 full search 방법과 같은 계산시간을 많이 요구하는 방법을 사용하여야 하나 전력소모가 늘어나게 된다. 따라서 Motion estimation에서 성능(PSNR)을 약간 감소하는 것을 허용하는 대신 pixel 값의 LSB truncation을 사용하여 전력량을 줄이는 방법이 제안되었다[15]. 실험 결과 truncation bit의 LSB truncation 을 사용하면 H/W 크기가 적어지고 전력량이 감소하게 된다. 그때의 truncation bit의 수는 4개가 적절하였다. LSB truncation은 또한 switching activity를 줄인다. 그러한 fixed bit truncation 방법을 사용한 결과 70%의 전력 감축 효과를 보인다[15].

기존의 pulse code modulation을 이용하여 상호연관성이 많은 데이터를 부호화 하면 redundancy가 많이 포함되게 된다. Redundancy를 제거하면 전력량이 감소하게 된다. Adaptive delta modulation을 이용하여 redundancy를 제거하는 방법이 [24]에 의하여 제안되었다.

DSP 시스템에서 자주 사용되는 N-tab FIR (Finite Impulse Response) filter는 다음의 convolution을 수행한다.

$$Y_j=\sum_{0\le k\le N-1} C_k X_{j-k} \quad (5)$$

여기서 C_k는 filter 상수, X_j와 Y_j는 j번째 입력 및 출력을 말한다.

Filter 출력은 각각의 product term을 곱하고 그 곱해진 product term을 더해서 구해지는데 그러한 방법을 DF (Direct Form)이라고 한다. DF의 속도 및 전력량을 감축하는 알고리즘인 DCM (Differences Methods)[31]은 두 개의 연속된 상수의 차이를 이용하여 부분 합이 저장되고 다시 재사용되어 더 많은 저장매체를 사용하여 (저장된 값을 가져오는데 걸리는 시간이 더 걸림) 전체 convolution을 계산하는데 필요로 하는 시간을 줄이는 것이다.

k 번째 상수는 다음의 식으로 표현된다.

$$C_k=C_{k-1}+\delta^1_{k-1/k} \quad (6)$$

여기서 $\delta^1_{k-1/k}$은 1st-order difference라고 한다. 식 (6)를 이용하여 (5)식을 구하면

$$Y_{j+1}=Y_j+C_0X_{j+1}+\{PS\}-C_{N-1}X_{j-N+1},$$
$$\{PS\}=\sum_{k=1}^{N-1}\delta^1_{k-1/k}X_{j-k+1} \quad (7)$$

으로 표현될 수 있다. 즉 1st-order DCM을 이용하여 출력을 구하는 방법은 먼저 계산되어 저장된 출력 값을 가져오고 {PS}와 다른 두 개의 product term을 곱셈을 이용 구한다. 이때의 계산 이득은 차이값(델타의 값)이 상수의 값보다 적을 때 (즉 적은 word-width) 얻어 진다. 상기 식(6)은 m-th order difference를 표현하는 식으로 일반화 될 수 있으며 그 차이 값은 order가 커질수록 점점 적어지는 효과를 이용하는 것이다.

DSP의 Multi-rate 설계 기법을 이용할 경우 입력 데이터 rate를 1/M배로 줄여서 전압을 줄이는 효과를 이용한 방법이 [35]에 의해서 제안되었다. Polyphase implementation 및 pseudo circulant matrix의 diagonalization기법을 이용하여 Down Sampling rate를 2로 할 경우의 전력소모는 다음의 식으로 주어지게 된다.

$$(\frac{3N/2}{N}C_{eff})(\frac{3.1V}{5V})^2(\frac{1}{2}f)\approx 0.29P_0 \quad (8)$$

여기서 P_0는 1-rate DSP 아키텍쳐를 사용한 경우이다. Multirate 아키텍쳐는 50%의 하드웨어를 추가적으로 사용하여야 하나 1-rate 아키텍쳐의 29%의 전력을 소모한다.

Adaptive filtering은 filter의 입력 시그널이 $X(n)=\{x(n),\ x(n-1),\cdots,x(n-N+1)\}^T$ 이고, filter 상수는 $W(n)=\{w_1(n),w_2(n),\cdots,w_N(n)\}^T$과 같을 때 mean squared error(MSE)를 최소화하는 filter 상수 vector를 구하는 것이다. $MSE=J(n)=E[e^2(n)]$, 여기서 E[]는 expectation operator이고, $e(n)=d(n)-W^T(n-1)X(n)$이다. 그 문제는 stochastic gradient 방법을 사용하여 다음의 식을 푼는 것과 같다.

$$y(n)=W^T(n-1)X(n)$$
$$W(n)=W(n-1)+\mu e(n)X(n) \quad (9)$$

여기서 $e(n)=d(n)-y(n)$, $d(n)$는 원하는 signal이며 $y(n)$은 filter 출력 시그널, μ는 stepsize를 말한다. 만약 그 stepsize를 아주 작게 하면 그 LMS 알고리즘은 최적에 가까워진다.

N-tap adaptive filter에서 어떤 filter는 power-up 또는 power-down을 시킬 수 있으며 각각의 filter tab에 대하여 control signal $\alpha_i\in\{0,\ 1\}$, $i=1,\cdots,N$을 정의한다.

여기서 $t_{ij}C_j$을 아키텍쳐 레벨에서 예측하기는 쉽지 않다. B_x bit의 입력과 B_c bit 상수 w_k를 곱하는 $B_x\times B_c$bit multiplier에 대한 전력소모는

$$p_m=B_x[\log_2(|w_k|)]\ C_bV_{dd}^2f_s \quad (10)$$

와 같다. 여기서 C_b는 primitive block cell의 정전용량이며 $B_x[\log_2(|w_k|)]$는 그 곱셈을 수행하기 위해 필요한 primitive block cell의 수를 말한다.

여러 개의 tab으로 이루어진 FIR filter 설계시

if $\alpha_i=0$ then the i^{th} tap is powered down.

if $\alpha_i=1$ then the i^{th} tap is powered up.

$$P_D=(\sum_{i=1}^{N}\alpha_i(\sum_{j=1}^{M}t_{ij}C_j)+T_{oh}C_{oh})V_{dd}^2f_s$$

where M is number of hardware units in each tap

C_j is the average switching capacitance for j^{th} hardware unit

C_{oh} is the overhead capacitance not considered in C_js

t_{ij} is the transition activity in the j^{th} hardware unit in the j^{th} tap

T_{oh} is the average transition activity

to minimize power dissipation, power down these taps which maximize the $\sum_{j=1}^{M}t_{ij}C_j$

mean square error(MSE)

$$e(n)=d(n)-\sum_{i=1}^{N}\alpha_i w_i x(n-i+1)$$

$J_{\min}=\sigma_d^2-\sum_{i=1}^{N}\alpha_i|w_i|^2r(0)$ where σ_d^2 is the power in the desired signal d(n)

r(0) is the power in the input signal x(n)

보통 multiplier의 개수를 줄이기 위하여 time-multiplexed multiplier를 사용한다. 연속된 두 개의 filter 상수는 비슷한 크기를 갖고 있기 때문에 그 두 개의 상수의 크게 다르지 않다면(high correlated) fixed input multiplier와 비교하여 비슷한 전력을 소모한다[29]. [14]은 decimation filter에서 operation minimization, multiplier elimination and block deactivation을 통한 전력 감축 기법을 제안하였다.

Carry-lookahead adders, Wallace multipliers는 면적이 적은 ripple adder나 carry save multiplier 보다 저전력을 소모한다. Booth encoded multiplier는 partial product수를 줄이기 위하여 two's complement operand (i.e.,multiplier)를 사용하여 수행속도와 면적을 향상시킨다. Multiplier를 Y, multiplicand를 X라고 하면 recorded digit는 $Y_{i-1}+Y_i-2Y_{i+1}$ (with $Y_{-1}=0$)이고 partial product selection은 다음과 같다.

Yi+1	Yi	Yi−1	Recoded digit	Operation on X
0	0	0	0	0X
0	0	1	+1	+1X
0	1	0	+1	+1X
0	1	1	+2	+2X
1	0	0	−2	−2X
1	0	1	−1	−1X
1	1	0	−1	−1X
1	1	1	0	0X

Partial product 발생은 다음의 operations에 준하여 수행된다.

Recoded digit	Operation on X
0	Add 0 to the partial product
+1	Add X to the partial product
+2	Shift left X one position and add it to the partial product
-1	Add two's complement of X to the partial product
-2	Take two's complement of X and shift left one position

그 예제는 다음과 같다.

```
(−6)  1 0 1 0  =X
(+6)  0 1 1 0  =Y
                          operation        bits recoded
0 0 0 0 1 1 0 0           −2               1 0 0
1 1 0 1 0 0               +2               0 1 1
1 1 0 1 1 1 0 0  =-36
```

저전력을 위한 메모리 bank 할당 문제는 메모리 bank가 현재 사용 중일 때만 그 bank를 activate 시키는 것이다. Video vector encoder에서 vector quantization[21]의 code-book 메모리를 분할하여 메모리 access를 줄이는 방법이 제안되었다.

VI. 저전력 프로세서

ARM(Advanced RISC Machine) 아키텍쳐는 UK, cambridge의 Advanced RISC machines Ltd. 에 의하여 개발된 32 bit RISC microprocessor이다. Amulet1은 상용 마이크로프로세서 아키텍쳐를 비동기화한 최초 대규모 비동기 회로이다. 비동기회로는 clock없이 작동하기 때문에 적어도 clock에 의해서 소모되는 전력을 줄일 수 있다. 아키텍쳐의 선택이 자유롭고 분배되고 국부적인 control이 가능하고, 공급전원의 채택이 자유롭다. Amulet1은 ARM processor를 변형한 전력 감축을 목적으로 한 비동기 회로이다. 트랜지스터 수와 AMULET1의 die 면적은 ARM6의 2배에 달하였으며 성능 및 전력 효율도 ARM6에 미치지 못하였다[13,11].

비록 AMULET이 기존의 ARM core를 대치할 만한 성능을 보이지는 못하였지만 저전력 목적으로 modular 접근 방식과 함께 많은 연구가 진행되리라 본다.

Philips Research에서는 비동기 회로 설계 자동화를 위하여 handshake circuits에 대한 VLSI-programming, compilation 툴을 개발하였다. 비동기 신호의 조정을 위하여 부가적인 회로(약 5%)를 이용하여 glitch-free 회로로 변환하였다[16]. fully asynchronous, 155k transistor DCC error corrector에 대하여 적용하여 동기회로와 비교한 결과 80%의 전력 감축 효과를 보였다.

Software pipelining[19]은 loop의 instruction-level 병렬 스케쥴을 찾는 기법이며 한 번에 여러 instruction을 수행 (super-scalar processors)하거나 긴 instruction을 사용하는 프로세서(VLIW)에 사용되어 진다. 그 목적은 가용한 Machine resource를 최적화 하여 loop의 한 반복동안 가능하면 많은 병렬처리를 수행하여 수행시간을 줄이는 것이다. 그 문제는 NP-hard로 알려져 있다. Software pipelining[33]는 많은 "register pressure"를 증가시킨다는 단점이 있다. 만약 가용한 register의 수가 부족하다면 register variable을 memory로 보내거나, cycle수를 늘림으로써 결과적으로 스케쥴 산출량은 감소하게 된다. 수행 산출량을 줄이지 않으면서 register pressure를 줄이

는 알고리즘이 필요하다.

VLIW(Very Long Instruction Word) 아키텍쳐는 고속 마이크로프로세서로 머지않아 현재의 프로세서를 대치할 것으로 보인다. High Level 언어를 직접 horizontal microcode로 compiling하여 하나의 긴(537 bits wide in[9]) 명령어(instruction)에서 여러 개의 계산이 병렬(예 : 3 operations/clock)로 수행된다. VLIW의 명령어 decoding은 fixed format을 이용하기 때문에 super scalar보다 쉽다. 그러나 그 고정된 길이의 VLIW 명령어는 operation의 결합에 장애 요소가 된다는 단점이 있다. 같은 ,개념을 이용한 것이 Intel P6이다. CISC 명령어가 결합되어 병렬로 처리된다. 다른 Intel 프로세서인 NexGen 586은 다변길이 x86 code를 고정 길이 RISC 명령어로 변환하여 빠른 RISC core로 수행시킨다.

병렬 Multithreaded 컴퓨터는 데이터 의존성에 기인하는 pipeline bubble을 제거하여 병렬 효과를 극대화하기 위하여 여러 개의 독립적인 tasks(명령어 flows)를 프로세서에 의해서 동시에 수행시킨다. 저전력이 중요시됨에 따라 설계 아키텍쳐의 단순성이 필요하게 되었다. Super-scalar 아키텍쳐는 그 복잡성 때문에 바람직하지 못하며 VLIW 및 Multithreaded 아키텍쳐가 그 간결성 때문에 저전력 프로세서로 바람직 할 것으로 보인다. 또한 모든 명령어를 CPI (clock cycles per instructions)가 1이 되도록 한다면 아키텍쳐가 간결해질 것이다. 그 아키텍쳐는 32-bit 프로세서인 StrongArm에 의하여 구현되었다. 160 MHz, 1.5Volt에 전력소비는 불과 0.5W (150MIPS에서 300MIPS/Watt)이며 0.35㎛ 공정을 이용 2.1 million 트랜지스터를 포함하고 크기는 50㎟이다. 그 아키텍쳐는 5-stage pipeline을 이용하며 execution stage는 3개(execute, buffer/data, write back)의 stage로 나누어진다.

그밖에 저전력(low power)과 고산출량(high-throughput)의 두 가지 목적을 실현하기 위하여 60 MHz, 1.0 V에서 17mW을 소모하는 FPGA를 이용한 programmable DSP가 0.35-micron dual threshold-voltage CMOS process를 사용하여 제작되었다[27].

VII. 결 론

전력은 면적 및 성능과 동시에 고려해야 하므로 더욱 복잡한 3-dimension의 multi-criterion 최적화 문제가 되며 그 해를 구하기 위해서는 새로운 CAD 툴의 개발이 절실히 요구되고 있다. 설계의 전 과정을 통해 전력 감축 노력이 필요하지만 특히 알고리즘 및 아키텍쳐 변환을 통한 전력 감축 효과 및 레이아웃 최적화[4] 및 device scaling을 통한 전력 감축 효과의 영향이 가장 클 것으로 기대된다. CAD 툴은 다음의 세 가지 1) analysis tools, 2) optimization tools, 3)libraries and design management tools로 나누어지며 유용한 CAD 툴은 domain-specific하게 제작되며 DSP 또는 CPU 각각에 적합하도록 제작된다. 저전력 설계 및 CAD 툴은 3년에 1/3의 전력감축, 5년에 1/5의 전력 감축의 목표를 달성하기 위한 필수적인 요건이 되고 있다. 기타 최근의 저전력 설계 자동화에 대한 조사문헌은 [10,28,36]에 상세히 기술되어 있다.

참 고 문 헌

[1] Akira Matsuzawa, Low-Power Portable Design, Matsushita Electric Co., Japan

[2] http : //www.arm.com, Advanced RISC Machines Ltd. Fulbourn Road, Cherry Hinton, Cambridge CB1 4JN U.K.

[3] L. Benini, M. Favalli and G. De Micheli. Design for testability of gated-clock FSMs European Design and test conference 1996.

[4] J. D. Cho and P. Franzon, "High Performance Design Automation for Multichip modules and packages", World Scientific

Pub. Co., Ltd. 1996.

〔5〕 J. D. Cho, "Combinatorial Aspects of Lower Power Clock Networks in VLSI Circuits", 한국정보과학회 SIGTCS news, Vol.7 No.2 pp.21-39, 1996.

〔6〕 J. D. Cho, S. S. Chun, "데이터 전송시 스위칭 동작 회수의 최소화를 통한 전력소비 감축 및 압축률 개선", 학국정보과학회 학술대회, 4. 1997.

〔7〕 J. D. Cho, "소자의 스위칭 동작 최소화를 통한 디지탈 회로 저전력 상위 레벨 최적화에 관한 연구", 서울대학교 반도체 공동연구소 연구보고서, ISRC96-E-2020, 9. 1997.

〔8〕 J. D. Cho, H.S.KIm, "저전력 기술매핑을 위한 게이트 재합성", 대한전자공학회 학술대회, 제 20권 1호 pp.743-746, 1997.

〔9〕 K. Ebcioglu, "Some Design Ideas for a VLIW Architecture for Sequential-Natured Software", IFIP Conf., on Parallel Processing, Pisa, Italy 1988 pp.18-29.

〔10〕 M. Elrabaa, I. S. Abu-Khater, M. I. Elmasry, Advanced Low-Power Digital Circuit Techniques, Kluwer Academic Pub. 1997.

〔11〕 S.B.Furber, Computing without Clocks: Micropipelining the ARM processor, in G. Birtwistle, G. and A. Davis, (eds.) Asynchronous Digital Circuit Design, Springer, ISBN 3-540-19901-2, 211-262.

〔12〕 Y.Fand and A.Albicki, "Joint Scheduling and Allocation for Low Power", Proc. of Int'l Symposium on Circuits and Systems, pp. 556-559, May, 1996.

〔13〕 S.B.Furber, P. Day, J.D.Garside, N.C. Paver and J.V.Woods, The Design and Evaluation of an Asynchronous Microprocessor, Proc. ICCD'94, pp. 217-220.

〔14〕 E. N. Farag, R. H. Yan, M. I. Elmsry, A Programmable Power-Efficient Decimation Filter for Software Radios, DAC, pp 68-71, 1997.

〔15〕 Z. L. He and K. K. Chan and C. Y. Tsui and M. L. Liou, Low Power Motion Estimation Design Using Adaptive Pixel Truncation, International Symposium on Lower Power Electronics and Design, pp. 167-172, DAC, 1997.

〔16〕 Kees van Berkel, "VLSI Programming of Asynchronous Circuits for Low Power" Philips Research Labs, The Netherlands.

〔17〕 Nand Kumar, Srinivas Katkoori, Leo Rader and Ranga Vemuri "Profile-Driven Behavioral Synthesis for Low Power VLSI Systems", IEEE Design & Test of Computers, Fall Issue, pp.70-84, 1995.

〔18〕 C. Knapp, P.J. Kindlmann, and M.C. Papaefthymiou. Implementing and Evaluating Adiabatic Arithmetic Units . In Proceedings of the IEEE 1996 Custom Integrated Circuits Conference, May 1996.

〔19〕 J. R. Lorch, Scheduling techniques for reducing processor energy use in MacOS.

〔20〕 N. Lalgudi and M.C. Papaefthymiou. Fixed-Phase Retiming for Low Power Design . In 1996 International Symposium on Low Power Electronics and Design, August 1996.

〔21〕 D. Lidsky and J. Rabaey, "Lower POwer Design of Memory Intensive applications-Case study : vector quanti-zation", Symp. on Low Power Electron. 1994.

〔22〕 P. Landman and J. Rabaey "Architectural Power Analysis : The Dual Bit Type Model", IEEE Trans. on VLSI Systems, vol 3, no.2, pp.173-187, 1995.

〔23〕 Musoll and J. Cortadella. Low-Power Array Multipliers with Transition-Retaining Barriers. In Proc. of the Int. Workshop on Power and Timing Modeling

Optimization and Simulation (PATMOS), pp. 227-238, October 1995.

[24] H. Mehta, B. M. Owens, and M. J. Irwin, Small Signal Model for Low Power DSP, International Symposium on Lower Power Electronics and Design, 1997.

[25] J. Montanaro et al. "A160Mhz 32b 0.5W CMOS RISC Microprocessor", ISSCC'96, San Francisco, CA. Feb. 1996.

[26] M. Goel, N. Shanbhag, Dynamic Algorithm Transformation for Low Power Adaptive Filter, International Symposium on Lower Power Electronics and Design, 1997, pp. 161-166.

[27] M. Mehendale, S. D. Sherlekar, G. Venkatesh, Coefficient Optimization for Low Power Realization of FIR Filters, IEEE Signal Pocessing VIII, pp.352-361, 1995.

[28] W. Nebel and J. Mermet, Low Power Design in Deep Submicron E;ectronics, NATO ASI Series, 1997.

[29] C. J. Nocol, P. Larsson, Low Power Multiplication for FIR Filters,, pp76-79, 1997.

[30] K.K.Parhi, "Algorithm Transformation techniques for concurrent processors", Proceedings of the IEEE, vol. 77, pp.1879-1895, Dec. 1989.

[31] N. Sankarayya, K. Roy, D. Bhattacharya, Algorithms for Low power and High Speed FIR Filter Realization Using Differential Coefficients, IEEE Trans. on Circuits and Systems-II : Analog and Digital Signal Processing, Vol. 44, No. 6, June 1997.

[32] Sanchez and Jordi Cortadella., Time-Constrained Loop Pipelining. In Proc. of the IEEE/ACM International Conference on Computer Aided Design, pp. 592-596, San Jose (USA), November 1995.

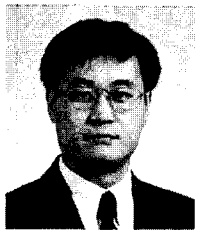
[33] Sanchez and J. Cortadella. Maximum-Throughput Software Pipelining. In 2nd International Conference on Massively Parallel Computing Systems (MPCS'96), pp. 483-490, Ischia (Italy), May 1996.

[34] Wai Lee Texas Instruments Inc., Dallas, Texas.

[35] A. Y. Wu, K. J. Liu, Z. Zhang, K. Nakajima, A. Raghuparthy, S. Liu, Algorithm-Based Low-Power DSP System Design : Methodology and Verification VLSI Signal Processing VIII, pp. 277-286, 1995.

[36] G. Yeap, Practical Low Power Digital VLSI Design, Kluwer Academic, 1997.

저 자 소 개



趙 浚 東

1957年 7月 21日生
1980年 2月 성균과대학교 학사
1989年 9月 Polytechnic University
1993年 6月 Northwestern University

1980年 4月～1983年 6月 대한민국 해병대 통신 장교
1983年 7月～1987年 8月 삼성반도체통신 연구소 CAD 연구원
1993年 8月～1995年 2月 삼성전자 반도체 연구소 CAD 선임 및 수석 연구원
1995年 3月～현재 성균관대학교 전자공학과 조교수
1996年 10月～현재 IEEE Senior Member

주관심분야 : VLSI 설계 최적화 알고리즘, 저전력 설계, Deep submicron 설계