A Low Dynamic Power 90-nm CMOS Motion Estimation Processor Implementing Dynamic Voltage and Frequency Scaling Scheme and Fast Motion Estimation Algorithm Called Adaptively Assigned Breaking-off Condition Search

Nobuaki Kobayashi and Tadayoshi Enomoto

Chuo University, Graduate School of Science and Engineering
Information and System Engineering Course
1-13-27 Kasuga, Bunkyo-ku, Tokyo 112-8551, Japan

ABSTRACT

A 90-nm CMOS motion estimation (ME) processor was developed by employing dynamic voltage and frequency scaling (DVFS) to greatly reduce the dynamic power. To make full use of the advantages of DVFS, a fast ME algorithm and a small on-chip DC/DC converter were also developed. The fast ME algorithm can adaptively predict the optimum supply voltage (V_D) and the optimum clock frequency (f_c) before each block matching process starts. Power dissipation of the ME processor, which contained an absolute difference accumulator as well as the on-chip DC/DC converter and DVFS controller, was reduced to 31.5 μW, which was only 2.8% that of a conventional ME processor.

Keywords: H.264, motion estimation, DVFS, power dissipation, DC/DC converter, PLL clock driver

1. INTRODUCTION

Power reduction techniques are necessary for battery-driven portable systems such as video encoding LSIs. Two techniques are known to reduce dynamic power (P). One is a power gating technique [1] that reduces the P of a processor by disconnecting the power supply through the use of MOSFET switches whenever the signal processing is completed. The amount of P reduction is proportional to the amount of signal processing reduction (e.g., P is ideally reduced to 1/2 when the amount of signal processing is reduced to 1/2).

The other technique involves using a dynamic voltage and frequency scaling (DVFS) technique [2] for which both the minimum supply voltage (V_D) and the minimum clock frequency (f_c) are supplied to the processor. These minimum values are proportional to the amount of signal processing, so the P reduction is proportional to the cube of the amount of signal processing (e.g., P is reduced to 1/8 when the signal processing amount is reduced to 1/2). Thus, P reduction using the DVFS scheme is much larger than that of the power gating technique.

To use the DVFS technique effectively, a small on-chip DC/DC level shifter and a fast motion estimation (ME) algorithm are needed. The fast ME algorithm must be able to adaptively estimate both the minimum V_D and the minimum f_c before every block-matching (BM) process begins. However, conventional fast ME algorithms [3, 4] can estimate neither the minimum V_D nor the minimum f_c, since they use visual distortion factors (e.g., values of absolute-difference accumulations) as threshold values to stop BM processes. In fact, visual distortion factors are independent of both V_D and f_c. Thus, conventional fast ME algorithms cannot be used in DVFS systems. To solve these problems we have developed a new ME algorithm with a BM-stopping condition that can predict both the required minimum V_D and f_c for each macro-block (M-Blk) for coding. The new ME algorithm, called the "adaptively assigned breaking-off condition search" (A^2BCS) can maintain the same visual quality as that of a full search (FS) algorithm.

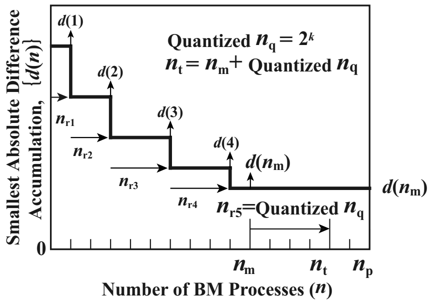

Fig. 2.1 Motion estimation process for A^2BCS.

We fabricated a 90-nm CMOS ME processor that employs the DVFS technique and the A^2BCS algorithm. The P of the processor was 31.5 μW, a significant reduction in P that was equivalent to only 3% of that of a conventional processor.

2. ME ALGORITHM FOR DVFS

2.1 ALGORITHM

The ME process for a given M-Blk in a current picture is illustrated as a solid line in Fig. 2.1, where the smallest present value of an absolute-difference accumulation $\{d(n)\}$ is plotted as a function of the number of BM processes (n). The ME process starts from the centre of the search window in a reference picture frame and moves toward the outer area. During this process, $d(n)$ reaches the smallest value, which is denoted by $d(n_m)$, at n of n_m, and then $d(n_m)$ is kept constant as n increases. The most efficient (i.e., the fastest) ME process is thus performed when the BM process is stopped at n of n_m.

If we could determine the value of n_m before the ME process begins for a given M-Blk for coding, we could calculate both the required V_D and f_c that are proportional to n_m. Thus, DVFS can be adopted. However, in fact, there is no way to estimate the value of n_m.

Both n and $d(n)$ are always monitored, so we can start to calculate n whenever the value of $d(n)$ decreases. This n is denoted as n_r in Fig. 2.1. While $d(n)$ changes frequently (i.e., n_rs are small), we should not stop the BM process. However, $d(n)$ keeps the same value for a large number of BM processes; that is, n_r becomes larger and is equal to an assigned number of BM processes (n_q), so that the possibility that $d(n)$ will change is very small. Then we can finish the BM process.

To determine the value of n_q, the latest information

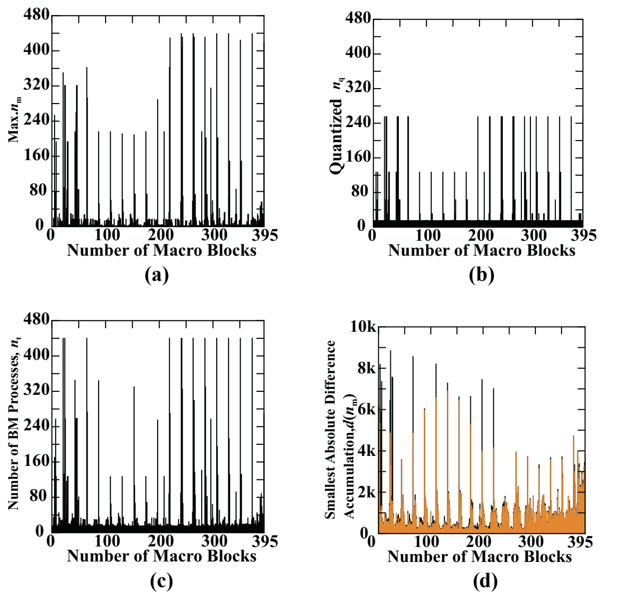

Fig. 2.2 Simulated characteristics of A^{2}BCS for each macro block in the 200th frame. (a) Max. n_ms. (b) Quantized n_q. (c) Total number of BM processes (n_ts). (d) d_ms (gray) of A^{2}BCS, on which d_ms of FS (black) are overlapped.

should be used. It is known that the characteristics of the M-Blk in the current frame resemble those of the M-Blk in the reference frame (i.e., the previous frame for P-pictures) for which both M-Blks are located in the same place. Thus, n_m of the M-Blk in the reference frame was chosen as the value of n_q (this n_m is denoted as n_{m-}). This means that the ME process can be adaptively stopped; consequently, d_m can be determined automatically. Then, the number of BM processes (n_t) for each M-Blk is thus given by the sum of n_m and n_q ($= n_{m-}$).

The encoding performance of the developed algorithm was evaluated by using several test video sequences. It was much faster than the FS algorithm, although the visual quality was slightly degraded; that is, $d(n_m)$ became slightly larger while the average peak signal-to-noise ratio (Ave peak R_{sn}) was slightly smaller than that of FS. This is one reason why the adaptively assigned n_q (i.e., n_{m-}) might be smaller than the optimum values, which results in the BM process stopping earlier than expected and a consequently larger d_m.

To improve visual quality, values of adaptively assigned n_q should increase by using the most recent information obtained by both the M-Blk in the reference frame and M-Blks located at the top, left, and upper left of the given M-Blk in the current frame. We chose the largest n_m (Max.n_m) among the n_m values of these four M-Blks. Then n_q is quantized by using the equation given by

$$2^{k+1} > \text{Max}.n_m \geq 2^k,$$

as follows. When k is larger than K, n_q is set to 2^k; when k is equal to or smaller than K, n_q is fixed at 2^K. As previously mentioned, this ME algorithm is called the adaptively assigned breaking-off condition search (A^{2}BCS) algorithm. The quantized n_q that is larger than n_q ($=n_{m-}$) is expected to improve visual quality of the encoded pictures.

Fig. 2.3 Motion-compensated P-picture ("Foreman", R_f = 15 fps, R_d= 384 kbps, p = 10 pixels, 200th frame). (a) FS. (b) A^{2}BCS.

2.2 Characteristics

An H.264 encoding program, in which A^{2}BCS was programmed, was used for simulation. A quarter-pel search and variable M-Blk size search were not used after A^{2}BCS was completed. The size of the M-Blk was only 16 pixels × 16 lines. The size of the search window was given by {($2p$+16) pixels × ($2p$+16) lines}, where p was the number of pixels and was set at 10. The maximum number of n was 441 (i.e., $4p^2$). Frame rate (R_f) and data rate (R_d) were 15 frames/sec (fps) and 384 kbit/sec (kbps). Encoding performance of the A^{2}BCS algorithm was evaluated by using several test video sequences.

Simulation characteristics of A^{2}BCS with K=4 for a test video sequence called "Foreman" are shown in Fig. 2.2 for the 200th frame. "Foreman" consists of a single I picture and 299 P-pictures with a common intermediate format (CIF) (352 pixels × 288 lines). Figure 2.2(a) shows Max. n_ms of 396 M-Blks in the 200th frame, and (b) plots quantized n_qs, that is, 2^k for k>4 and 16 for k≤4. Figure 2.2(c) shows the numbers of BM processes (n_ts) for each M-Blk. A^{2}BCS is considerably faster (i.e., n_ts is considerably smaller) than that (n_ts=441) of FS. The search speed of A^{2}BCS is 9.6 times faster than FS. Figure 2.2(d) shows d_m (gray) of A^{2}BCS, overlapped with the d_m of FS (black). The d_m of A^{2}BCS agrees well with that of FS, indicating that the visual quality is almost the same.

Figures 2.3(a) and (b) show one of the motion-compensated P-pictures in "Foreman" obtained by FS and A^{2}BCS with K=4, respectively. It is difficult to find a significant difference between these two pictures. Furthermore, the Ave peak R_{sn} of A^{2}BCS is 37.428 dB, exactly the same as that of FS. This means that visual quality was considerably improved by using quantized n_qs.

The performance of A^{2}BCS with K=4 in CIF test-video sequences called "Akiyo" and "Coastguard" was also evaluated. The search speeds of A^{2}BCS for "Akiyo" and "Coastguard" were respectively 23.2 and 20.0 times faster than FS. The Ave peak R_{sn} of A^{2}BCS for "Akiyo" and "Coastguard" was slightly smaller (i.e., 0.010 and 0.039 dB smaller) than that of FS (i.e., the distortion performance of A^{2}BCS is almost the same as that of FS).

3 CMOS Motion Estimation (ME) Processor

To examine the effect of the A^{2}BCS algorithm and the DVFS technique on power reduction, an ME processor

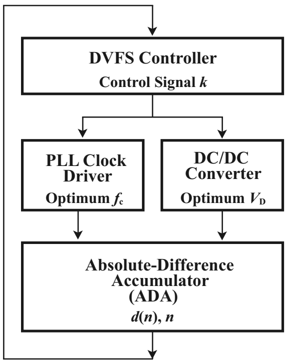

Fig. 3.1 ME processor employing DVFS.

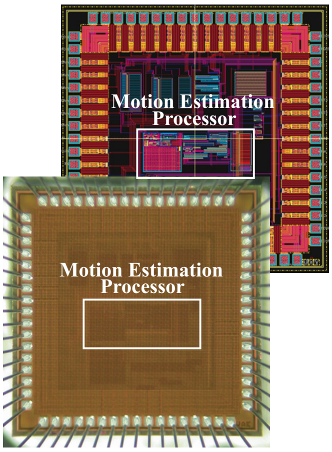

Fig. 3.2 90-nm CMOS LSI including ME processor.

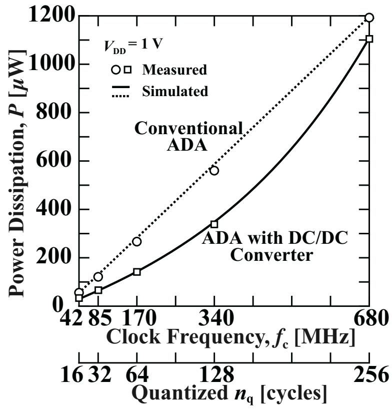

Fig. 3.4 Measured and simulated power dissipations (*P*s) of the 90-nm CMOS ADA as a function of f_c.

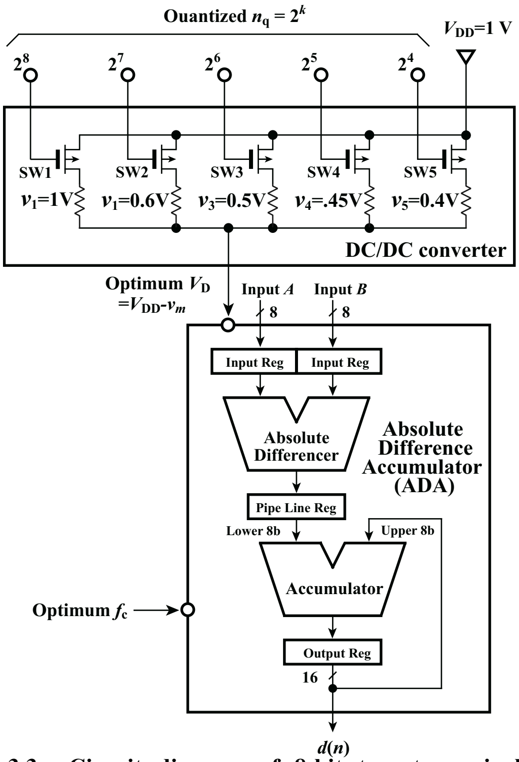

Fig. 3.3 Circuit diagram of 8-bit two-stage pipelined absolute difference accumulator (ADA) with DC/DC converter.

was fabricated using 90-nm, triple-well, six-layer Cu interconnect, CMOS technology. The ME processor consisted of a two-stage pipelined absolute difference accumulator (ADA), a DVFS controller, a DC/DC converter, and a PLL clock driver, as shown in Fig. 3.1. Figure 3.2 shows a photograph and layout of a CMOS LSI in which the ME processor (330 μm × 970 μm) was integrated.

3.1 Absolute Difference Accumulator

Figure 3.3 shows circuit diagrams of the 8-bit ADA with the DC/DC converter. The ADA consists of an 8-bit absolute difference circuit (ADC) and a 16-bit accumulator (ACC). The ADA was designed to calculate *d*(*n*)s for all M-Blks in an entire search window to obtain the best-matching MB having the smallest *d*(*n*). The DC/DC converter consists of five pMOSFET switches {SW_m (*m*=1 to 5)} connected in parallel. One of five switches connects a power supply (V_{DD}) and the ADA on request. When a control signal from the DVFS controller becomes "0," SW_m is turned on. Thus, a virtual supply voltage (= optimum V_D) can be given by V_{DD}-v_m, where v_m is a voltage drop of SW_m.

Figure 3.4 plots the experimentally measured power dissipation (*P*) of the ADA with the DC/DC converter (squares) along with the SPICE-simulated *P* (solid line) as a function of the clock frequency (f_c) at V_{DD} of 1.0 V. The measured *P*s agree well with the simulated *P*s. It is clear that *P* of the ADA with the DC/DC converter is much smaller than *P* of the conventional ADA (circles and dotted line).

3.2 DVFS Controller

The DVFS controller consists of a maximum data detector, a minimum data detector, a quantized n_q generator, a comparator, several counters, SRAMs, etc. The DVFS controller was designed not only to detect $d(n_m)$ and n_m, but also to generate the quantized n_q.

Figure 3.5 depicts the clock timing of the BM process for the *n*th M-Blk for coding. After the BM process for (*n*-1)th M-Blk is finished, the DVFS controller starts to calculate the Max. n_m and to estimate the quantized n_q. Then, for the *n*th M-Blk, the DVFS controller estimates the optimum f_c, the optimum V_D, and n_p. The n_p is the maximum number of BM processes that can be carried out for the *n*th M-Blk at the given optimum f_c. Only several clock periods are needed to obtain these values. The quantized n_qs (=2^k) and corresponding optimum f_cs, optimum V_Ds, and n_ps are summarized in Table 3.1. The *P*s of the ADA with the DC/DC converter at the given quantized n_qs are also listed in Table 3.1. The optimum f_c and the optimum V_D are respectively generated by the PLL clock driver and the DC/DC converter and then supplied to the ADA. The BM process to generate *d*(*n*) is stopped, whenever n_r reaches the quantized n_q (Figs. 2.1 and 3.5).

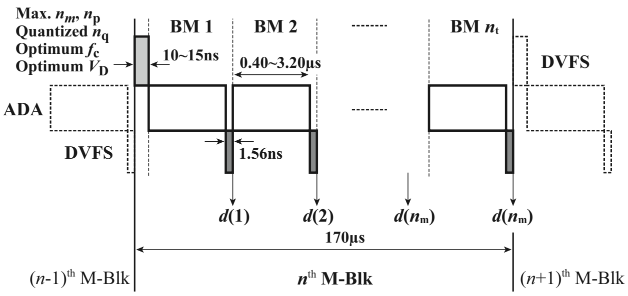

Fig. 3.5 Clock timing of the BM process for the *n*th M-Blk for coding.

Table 3.1 Quantized n_q, optimized V_D, optimized f_c, n_p and P.

Quantized $n_q=2^k$	Optimum f_c [MHz]	Optimum V_D [V]	n_p	P_{AT} [μW]
$2^8 = 256$	680	1.00	450	1,111
$2^7 = 128$	340	0.60	225	344.1
$2^6 = 64$	170	0.50	112	146.1
$2^5 = 32$	85	0.45	56	65.15
$2^4 = 16$	43	0.40	28	26.12

4 Power Dissipation of ME Processor

Figure 4.1(a) and (b) show both the optimum f_cs and the optimum V_Ds, respectively for each M-Blk in the 200th frame of "Foreman" (A²BCS with K=4). They are adaptively assigned for each M-Blk by the quantized n_qs that are shown in Fig. 2.2(b).

Figure 4.1(c) shows n_ps (black), on which n_ms (gray) are overlapped. The n_ps are also adaptively assigned by the corresponding quantized n_qs. To maintain excellent visual quality, such as that of FS, the best-matching M-Blk must be found before n_m reaches n_p (i.e., $n_m<n_p$). All M-Blks shown in Figure 4.1(c) satisfy this condition ($n_m<n_p$). This means that there is no degradation of visual quality due to the introduction of DVFS.

Figure 4.1(d) shows P, which is consumed by the ADA with the DC/DC converter, at each M-Blk. By employing the DVFS technique, the P values of most M-Blks are reduced to less than 65 μW, that is, about 7% of the maximum P. This means that the DVFS technique with the A²BCS algorithm is very effective to reduce P.

The average P of the ADA for 299 P-pictures of "Foreman" was 86.2 μW, that is, 7.37% of P (1,170 μW) of the conventional ADA. Similarly, employing A²BCS and the DVFS technique significantly reduces the P of the ADA for other test video sequences. They were 29.5 μW for "Akiyo" and 29.6 μW for "Coastguard"; these values are 2.52% and 2.53% of P for the conventional ADA, respectively.

The values of P of the ME processor varied from 31.5 to 88.2 μW depending on the test video pictures. These were the sums of P of the ADA and P of the DVFS controller. The DVFS controller operated at the clock frequency of 680 MHz. However, it was stopped most of the time (Fig. 3.5) by using a gated clock technique Therefore, the P of the DVFS controller was dominated by leakage currents, and was 1.95 μW.

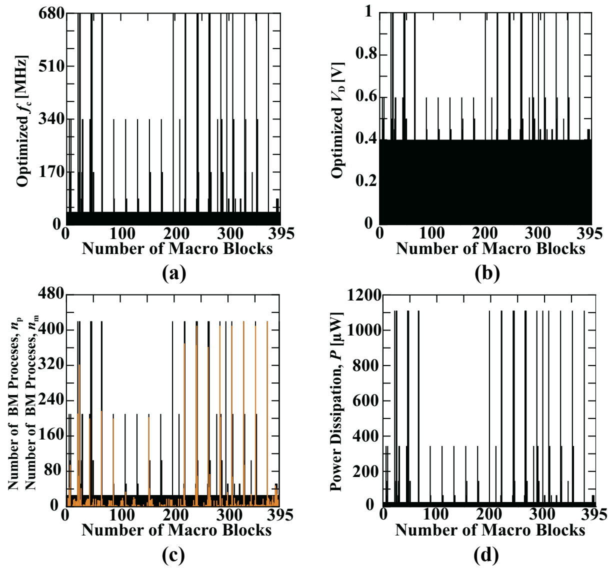

Fig. 4.1 Simulated characteristics of each M-Blk in the 200th frame. (a) Optimized f_cs. (b) Optimized V_Ds. (c) n_ms on n_ps. (d) Ps of ME processor.

5 Summary

A motion estimation (ME) processor that employs dynamic voltage and frequency scaling (DVFS) was developed using 90-nm CMOS technology. To make full use of the advantages of DVFS, we developed a fast motion estimation algorithm called the adaptively assigned breaking-off condition search (A²BCS). The A²BCS algorithm can predict the optimum clock frequency and the optimum supply voltage. The ME processor consists of an absolute difference accumulator with a small DC/DC converter, a minimum value detector, a DVFS controller, and a PLL clock generator. Power dissipation of the ME processor was significantly reduced and varied from 31.5 to 88.2 μW, only 3 to 8 % of the power dissipation of a conventional ME processor, depending on the test video pictures. Thus, DVFS is one of the most useful power reduction techniques for future video picture coding applications.

References

[1] S. Kim, S. V. Kosonocky, D.R. Kneble, and K. Stawlasz, "Experimental measuremnt of a novel power gating structure with intermediate power saving mode," Proc. ISLPED, pp. 20-25, Sept. 2004.

[2] V. Gutnik and A Chandrakasan, "An efficient controller for variable supply-voltage low power processing," Symp. on VLSI Circuits, pp. 158-159, June 1996.

[3] C. H. Cheung and L. M. Po, "Novel cross-diamond search algorithms for fast block motion estimation," IEEE Tran. on Multimedia, vol.12, no. 12, pp. 1168-1177, Dec. 2005.

[4] T-H Tsai and Y-N Pan, "Novel cross-diamond search algorithms for fast block motion estimation," IEEE Tran. on Circuits. Syst. Video Technol., vol.16, no. 12, pp. 1542-1549, Dec. 2006.