

高速動きベクトル検出アルゴリズムと低電力 CMOS デジタル LSI

High Speed Motion Estimation Algorithms and Low-Power CMOS Digital LSIs

情報工学専攻 小林伸彰
KOBAYASHI Nobuaki

概要

MPEG-4 対応の動画画像符号化 LSI の低電力化を目的に、アーキテクチャを改良した 0.18- μm 累算形差分絶対値和回路とこれに適用する高速動きベクトル検出 (ME) アルゴリズム (BOSA) を開発した。本アルゴリズムは既報の高速 ME アルゴリズム (HS-IBOS) を改良することにより、マクロブロック (M-Blk) 毎のブロックマッチング回数 (N_b) を適応的に制御し、それを最小化できる。テスト画像 “Carphone” (QCIF, 64 kbps, 15 fps, 探索領域サイズ $p = 10$ 画素) を用い、全探索法 (FS) 並の画質を維持したまま、ME 速度を FS の約 11.76 倍以上、高速化できた。BOSA を適用するとアーキテクチャを改良した 0.18- μm 累算形 ADA の動作時消費電力 (P_{AT}) は 29.92 μW (1.23 V, 160 MHz) となり、従来形累算形 ADA の約 2.110% に低減できた。キーワード MPEG-4、動きベクトル検出、CMOS、差分絶対値和回路、消費電力

1 はじめに

携帯機器向け動画画像符号化 LSI、特に、動きベクトル検出 (Motion Estimation; ME) のための差分絶対値和回路 (Absolute Difference Accumulator; ADA) の小形化と低電力化が重要である。中でも ME 処理の高速化は最も効率の良い低電力化手段である。我々はこれまでに全探索 (Full-Search; FS) に代わる各種高速 ME アルゴリズム、つまり、中断法 (Breaking-Off-Search; BOS) [1]、改良中断法 (Improved Breaking-Off-Search; IBOS) [2]、2 ステップ中断法 (2S-IBOS) [3, 4] を開発し、ブロックマッチング (Block Matching; BM) 回数、 N_b を大幅に削減してきた。この結果、ADA を大規模なアレイ形から小型な累算形へ置換えることができた [3, 4, 5]。ME 処理をさらに高速化し、ADA の低電力化を図るために、2003 年 10 月、高速改良中断法 (High Speed IBOS; HS-IBOS) を開発し、報告した [6]。HS-IBOS は最小差分絶対値和、(最小 D_a) の減少率が連続して 0 となる回数が、予め設定された回数、 N_{sc} と等しくなると、最適なマクロブロック (Macro Block; M-Blk) (16 画素 \times 16 ライン) を検出できたとして、BM を終了させる ME アルゴリズムである。QCIF テスト画像の “Carphone” (64 kbps, 15 fps, 探索窓サイズ、 $p = 10$ 画素) を用いた場合、 N_{sc} が 64 で、FS 並の画質を得ることができた。BM 速度は FS の約 5 倍、IBOS の約 2 倍であった。同様に、QCIF の “Coastguard”、CIF の “Mobile & Calendar” を用いると、 N_{sc} をそれぞれ 32, 64 に設定すると、FS 並の画質が得られた。しかし、予め最適な N_{sc} を予測することは不可能であり、 N_{sc} を小さく設定すれば、画質が劣化し、大きく設定すれば、BM 速度が低下する。

このような HS-IBOS の問題を解決するために、M-Blk 毎に最適な N_{sc} を適応的に設定できる新たな ME 法 (Breaking-Off-Search Adaptively minimizing number of block matchings; BOSA) を開発した。第 2 章、第 3 章でそれぞれ BOSA のアルゴリズム、特性を詳述する。第 4 章では回路アーキテクチャの改良および BOSA を適用することにより、ADA の消費電力が大幅に削減できることを述べる。

2 BM 数を適応的に設定する ME 法 (BOSA)

2.1 BM 数を適応的に設定

図 2.1 に FS の BM 処理における対象 M-Blk と探索範囲 (Search Window; SW) 内の各 M-Blk との間で得られる差分絶対値和、 D_a の最小値 (最小 D_a) が推移して行く過程の一例を示す。FS では SW 内のすべての M-Blk に対して BM 処理するので、 N_b が $N_{b,F} (=4p^2)$ となるが、実際には C 点で BM 処理を終了してもよい。これを実現した ME が BOSA である。図 2.1 では同一値の D_a が最も長く保持されている期間、 $N_{b,L} (=N_{b,B} - N_{b,A})$ で、この期間の D_a は $D_{a,A}$ である。実際に、本例のように、 $D_{a,mf}$ に極めて近い値の $D_{a,A}$ が得られた直後に、さらに $D_{a,mf}$ に近い値の $D_{a,B}$ あるいは $D_{a,mf}$ 自身 (この場合、 $D_{a,C}$) が得られることが多い。つまり、 $D_{a,mf}$ に極めて近い D_a やあるいは $D_{a,mf}$ を得るために必要な N_b の値は $N_{b,L}$ より大きな値であると推測できる。図 2.1 から明らかなように、 $N_{b,mf}$ は必ず $N_{b,L}$ より大きいか、あるいは等しい ($N_{b,mf} \geq N_{b,L}$)。そこで、 $N_{b,mf}$ が既知であり、これを本対象 M-Blk の N_{sc} の値として用いることができたとすれば、たとえ、条件を満たすことができなくても、結果として、全探

索することになるので、 $D_{a,mf}$ は得られるはずである。

実際には、対象 M-Blk の $N_{b,mf}$ は未知であるから、これに近い値を既に処理したデータの中から適応的に求めて、これを N_{sc} として設定する以外に方法はない。このように、対象 M-Blk の N_{sc} を適応的に設定し、処理回数を最小化する中断法を BOSA と呼ぶことにする。なお、BOSA では BM が A 点と B 点の間で終了する場合もある。この場合、最小の D_a ($D_{a,mf}$ ではないので、 $D_{a,m}$ と表記する) は $D_{a,A}$ 、 $D_{a,m}$ が得られた時点の N_b ($N_{b,mf}$ ではないので、 $N_{b,m}$ と表記する) は $N_{b,A}$ である。

BOSA [7] は対象 M-Blk を基準に現フレームの左の M-Blk、真上の M-Blk、左上の M-Blk および前フレームの同位置 M-Blk) の $N_{b,m}$ を利用し、これらの中で最大の $N_{b,m}$ 、 $N_{b,Max}$ を用いて、以下のように N_{sc} を量子化して設定する。

$$\begin{aligned} N_{b,mf} \geq N_{b,Max} \geq 256 \text{ の時、} N_{sc} &= 256 \\ 256 > N_{b,Max} \geq 128 \text{ の時、} N_{sc} &= 128 \\ 128 > N_{b,Max} \geq 64 \text{ の時、} N_{sc} &= 64 \\ 64 > N_{b,Max} \geq 32 \text{ の時、} N_{sc} &= 32 \\ 32 > N_{b,Max} \geq 0 \text{ の時、} N_{sc} &= 16 \end{aligned}$$

ここで、 p が 10 の時、 $N_{b,mf}$ は 400 である。

3 BOSA の特性評価

3.1 テスト画像と解析条件

“Carphone” と呼ばれるテスト画像を用いて、画像解析を行う。画像サイズは QCIF (176 画素 \times 144 ライン)、YUV は 4:2:0、フレーム数、 N_f は 382 (I-VOP : 1 フレーム、P-VOP : 381 フレーム) である。1 画素当たりのデータ数は 8 ビット (b) である。以下では、P-VOP 381 フレームに対して、MPEG-4 に準拠したソフトウェアエンコーダを用いて解析する。データレート、 R_d は 64 kbps、フレームレート、 R_f は 15 fps、に設定する。 p を画素数とすると、SW サイズは $2p$ 画素 \times $2p$ ラインで与えられる。以下では p を 10 画素とした [5]。なお、階層探索、多数画素精度探索、半画素精度探索、動きベクトル情報による探索領域のシフト、等は併用しない。また、FS 以外のアルゴリズムでは渦巻き探索を用いている。

3.2 BM 停止条件の設定

図 3.1 (a)、図 3.2 (a) にそれぞれ FS の $N_{b,Max}$ 、BOSA の $N_{b,Max}$ を示す。前者と比較して、後者は等しいかあるいは小さな値を示している。

図 3.1 (b)、図 3.2 (b) にそれぞれ FS、BOSA の N_{sc} (200 フレーム目) を示す (実際には、FS では N_{sc} は用いない。ここでは比較のために示している)。なお、BOSA の N_{sc} は、2 章で述べたように、量子化されている。

3.3 N_b 、最小 D_a の推移 (200 フレーム目)

図 3.1 (c)、図 3.2 (c) にそれぞれ FS、BOSA の N_b を示す (200 フレーム目)。FS では全 M-Blk に対して、 N_b は 400 ($=4p^2$) で一定である。BOSA の N_b は N_{sc} と等しいかあるいは N_{sc} より大きな値を示している。

図 3.1 (d)、図 3.2 (d) にそれぞれ FS、BOSA の最小 D_a の推移を示す。図 3.2 (d) の $D_{a,m}$ は図 3.1 (d) の $D_{a,mf}$ とよく一致している。しかし、詳細に見ると、BOSA と FS との間に差があるのが分かる。

3.4 N_b 、最小 D_a (2~382 フレーム)

表 3.1 に各アルゴリズムの特性をまとめて示す。全 P-VOP (381 フレーム) の M-Blk 当りの平均 N_b は BOSA が 34 で、FS の $N_b (=400)$ に対してそれぞれ 11.76 倍と大幅に高速化されている。

一方、平均最小 D_a は FS の 1,335 に対して、BOSA が 1,392 (同 4.27% 増) である。これより、BOSA は FS 並の画質 (FS と同等の動き補償画像) を維持できることがわかる。

3.5 再生画像と平均ピーク S/N

図 3.3 (a)、(b) にそれぞれ FS、BOSA を用いた再生画像 (200 フレーム目) を示す。画像間に顕著な差は見えない。FS、BOSA の平均ピーク S/N、 R_{SN} はそれぞれ 32,582, 32,423 [dB] である。以上の結果より、“Carphone” を用いた場合、BOSA は ME 速

度を FS の 11 倍以上高速化しながら、画質を FS 並にすることができるといことが分かる。

4 差分絶対値和回路の低消費電力化

4.1 回路構成

逐次桁上げ伝播加算回路を用いた 16 b 累算形差分絶対値和回路 (従来形) を図 4.1 に示す。本回路は、1 画素分のデータの差分絶対値を得る 8 b 差分絶対値回路 (8-bit ADC) と 1M-Blk 分の差分絶対値を累算する 16 b 累算回路 (16-bit ACC) で構成される。両回路とも逐次桁上げ伝播形の加算回路を用いている [3、4、5、6]。ゲート総数 (G) は 373、クリティカルパスゲート段数 (G_c) は 32 である。

4.2 アーキテクチャの改良

回路の最高動作周波数を維持しながら、動作時消費電力、待機時消費電力を削減するためには G_c を低減して、電源電圧を低減するより方法はない。 G_c を低減するために、8-bit ADC と 16-bit ACC の間にパイプラインレジスタ (PR) を設ける。従来形に PR を加えた、2 段パイプライン構成、逐次桁上げ伝播加算回路を用いた 16 b 累算形差分絶対値和回路 (改良形 1) を図 4.2 に示す。ゲート総数 G は 431 で PR 分、増加する。 G_c は 19 であり、従来形の 59.38% に削減される。

逐次桁上げ方式の上位 8-bit 累算回路 (ACC) は上位 8-bit ACC の保持データに上位 8-bit ACC からの桁上げ信号 C_7 (1bit 分) を加算しているだけである。この機能は C_7 が 0 の時、保持データをそのまま保持し、 C_7 が 1 の時、保持データに 1 を加算することにより実現できる。従って、図 4.3 に示すように、予め 9bit 目に 1 を加算した結果を準備しておく、 C_7 が 1 の時、この結果を選択し、 C_7 が 0 の時、前周期の値を保持する、桁上げ選択方式を用いればよい。上位 8-bit 累算回路に桁上げ選択方式を適用した累算形 ADA を桁上げ選択形差分絶対値和回路 (改良形 2) と呼ぶ。本回路の G_c は 13 段である。従来形の G_c は 32 段であるから、15 段 (59.37%) 削減できたことになる。 G は 463 で、改良形 1 に比べ約 7.4% 回路規模は大きくなる。

16-bit ACC の上位 8-bit ACC は C_7 が 1 となる回数をカウントしているだけである。従って、図 4.4 に示すように、上位 8-bit ACC を 8bit アップカウンタで置き換えてもよい。この累算形 ADA を桁上げカウンタ形差分絶対値和回路 (改良形 3) と呼ぶ。本回路の G_c は 13 段である。改良形 2 と同様、従来形に比べ、15 段削減できたことになる。 G は 463 で、改良形 1 に比べ約 3.2% 回路規模は小さくなる。改良形 1 の $u+1$ サイクル目の ACC 8 bit 目出力信号、 $S_8(u+1)$ は

$$S_8(u+1) = S_8(u) \oplus C_7(u+1)$$

で与えられる。また、 i が 9 以上するとき、 i bit 目の出力信号、 $S_i(u+1)$ は

$$S_i(u+1) = S_i(u) \oplus S_{i-1}(u) \cdots S_9(u) S_8(u) C_7(u+1)$$

で与えられる。周期 $u+1$ の前半で $C_7(u+1)$ 以外を計算しておく、周期 $u+1$ の途中で、つまり、 $C_7(u+1)$ が得られた時点で、全ての $S_i(u+1)$ を同時に計算すれば、全ての累算結果、 $S_i(u+1)$ ($i = 8$ から 15)、が同時に得られる。

表 3.1 各種アルゴリズムの特性。 ($R_d = 64$ kbps, $R_f = 15$ fps, $p = 10$ 画素)

アルゴリズム	平均 N_b	平均 F_s	平均 D_a	平均 R_{sn} [dB]
FS	400 (100.0)	1.000	1,335 (0.000)	32.582 (0.000)
BOSA	34 (8.5)	11.76	1,392 (+4.270)	32.423 (-0.159)

()内は FS との比較。



図 3.3 動き補償画像 (“Carphone”, $R_f = 15$ fps, $R_d = 64$ kbps, $p = 10$ 画素、200 フレーム目), (a) FS. (b) BOSA.

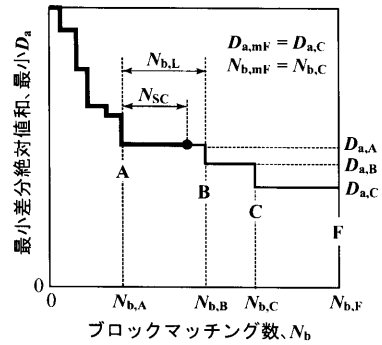


図 2.1 最小差分絶対値和 (最小 D_a) の推移。

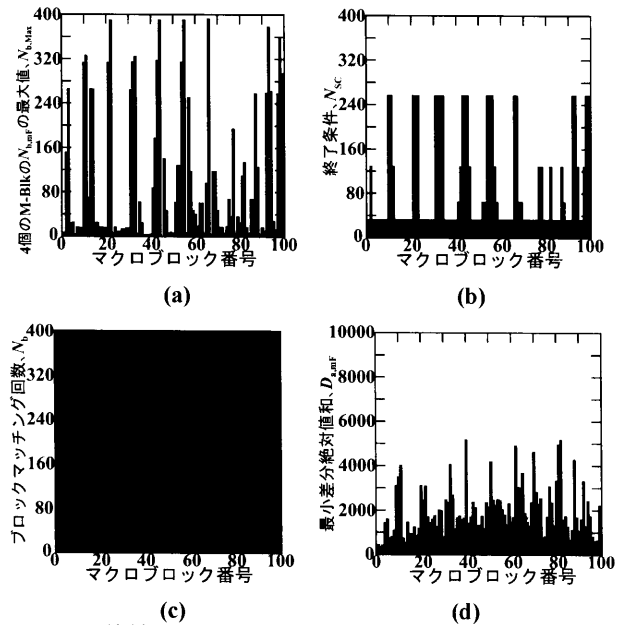


図 3.1 FS の特性 (200 フレーム目). (a) $N_{b,Max}$. (b) N_{sc} . (c) N_b . (d) 最小 D_a .

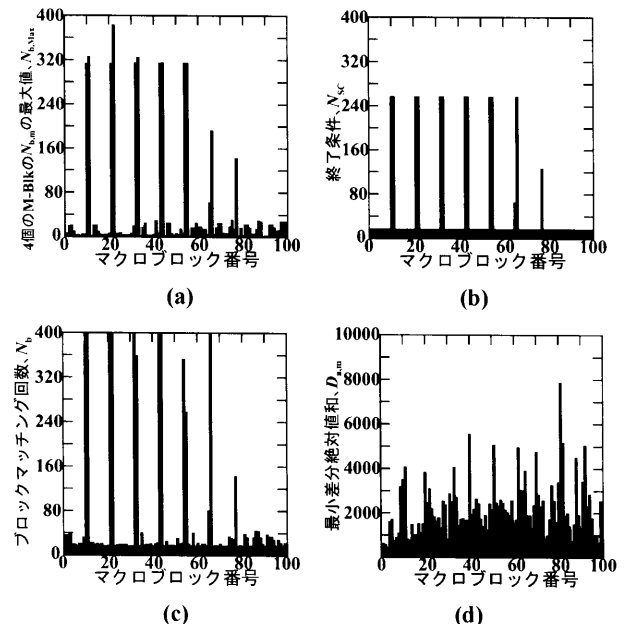


図 3.2 BOSA の特性 (200 フレーム目). (a) $N_{b,Max}$. (b) N_{sc} . (c) N_b . (d) 最小 D_a .

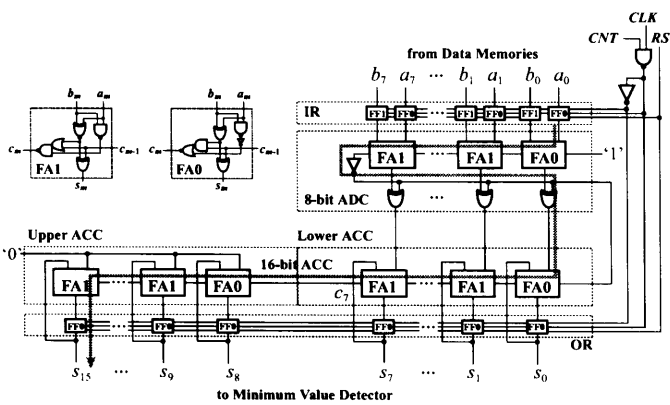


図 4.1 逐次桁上げ伝播形加算回路を用いた 16 b 累算形差分絶対値和回路 (従来形) の構成。

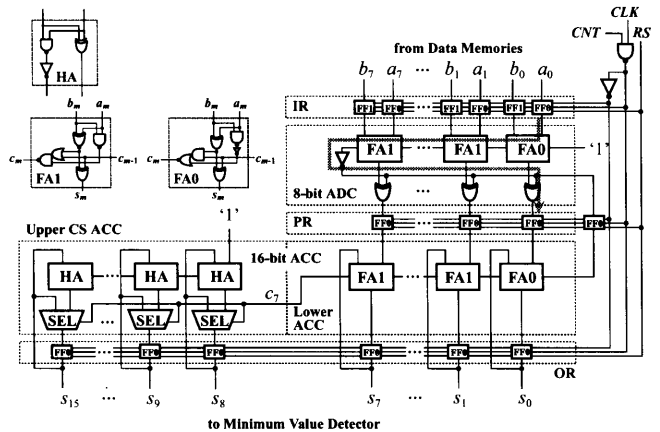


図 4.3 2 段パイプライン構成桁上げ選択方式を用いた 16 b 累算形差分絶対値和回路 (改良形 2) の構成。

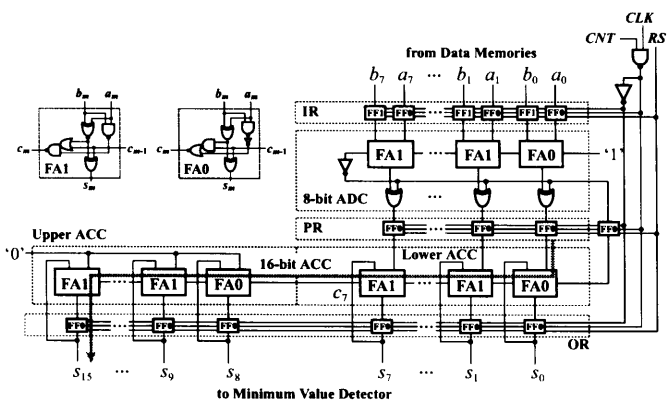


図 4.2 2 段パイプライン構成、逐次桁上げ伝播形加算回路を用いた 16 b 累算形差分絶対値和回路 (改良形 1) の構成。

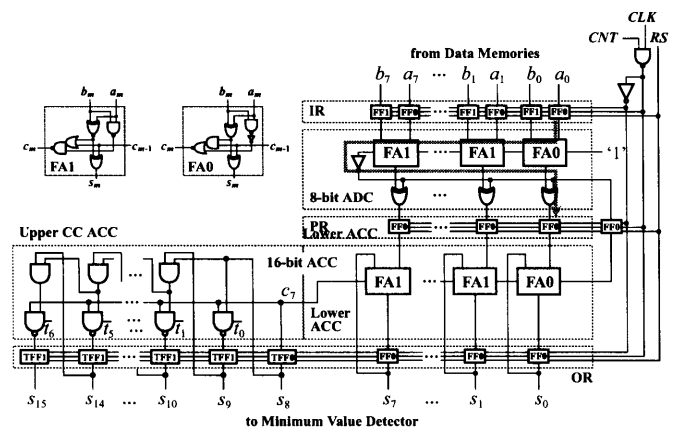


図 4.4 2 段パイプライン構成桁上げカウント方式を用いた 16 b 累算形差分絶対値和回路 (改良形 3) の構成。

4.3 SPICE 解析結果

上述した各累算形 ADA を 0.18- μm CMOS 技術で設計し、最高動作周波数 ($\text{Max}f_c$)、動作時消費電力 (P_{AT}) を SPICE 解析で求めた nMOS、pMOS の閾値電圧はそれぞれ 0.435 V、-0.415 V、標準電源電圧は 1.8V である。

$\text{Max}f_c$ およびその従来比と電源電圧 (V_{DD}) の関係を図 4.5 (a) に示す。各 $\text{Max}f_c$ は V_{DD} にほぼ比例して増加する。0.6 V ~ 2.2 V の間、 $\text{Max}f_c$ はクリティカルパスのゲート段数 (G_c) にほぼ反比例して増加している。

一方、 $\text{Max}f_c$ を一定に維持すると、例えば $\text{Max}f_c$ が 160 MHz の時 (図 4.5 (a))、 G_c の減少にもない、 V_{DD} が低下している (表 4.1)。

P_{AT} およびその従来比と V_{DD} の関係を図 4.5 (b) に示す ($f_c = 160$ MHz)。各 P_{AT} は V_{DD}^2 にほぼ比例して増加する。 f_c を固定すると、 P_{AT} は総ゲート数 (G) にほぼ比例している (表 4.1)。また、 G_c が減少するので、 V_{DD} も低下する (表 4.1)。この結果、改良形 1、改良形 2、改良形 3 の P_{AT} は低減され、それぞれ従来形 (1418 μW) の 51.54%、35.99%、26.06% となる (表 4.1)。

以上の結果を以下に概略する。 f_c を 160 MHz に固定した場合、従来形の 160 MHz 動作を補償する最小電源電圧 V_{DD} は 2.19 V である。一方、改良形 1、改良形 2、改良形 3 は、 V_{DD} をそれぞれ 1.54V、1.23 V に下げても 160 MHz で動作できる。この場合、改良形 1、2、3 の P_{AT} はそれぞれ、731.0 μW 、510.5 μW 、369.6 μW となり、従来形の P_{AT} (1418 μW) の 51.54%、35.99%、26.06% に低減され、 G 、 G_c の削減効果が大いことがわかる。

4.4 チップ試作と実測結果

図 4.6 に改良形 1、改良形 2、改良形 3 を搭載したチップの拡大写真を示す。チップサイズは 3.9 mm 角である。用いた CMOS 技術は 0.18- μm 、ダブルウェル、1 層ポリシリコン、2 層ポリ Si、3 層 Al メタル配線である。nMOS、pMOS のマスク上のゲート長は共に 0.2 μm 、nMOS、pMOS の閾値電圧はそれぞれ 0.435 V、-0.415 V、標準電源電圧は 1.8V である。

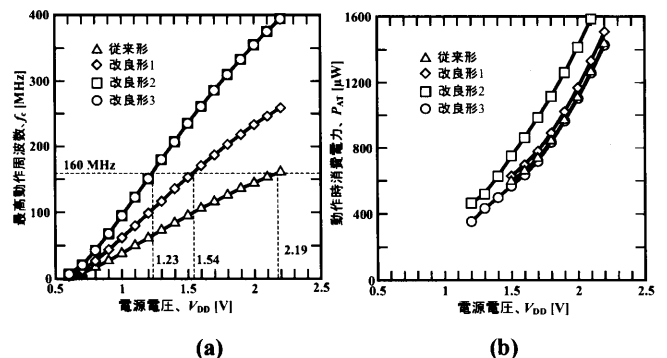


図 4.5 各種累算形 ADA の SPICE 解析結果。(a) 最高動作周波数、 $\text{Max}f_c$ [MHz] (b) 動作時消費電力、 P_{AT} [μW]。

図 4.7 (a) に累算形 ADA の最高動作周波数 $\text{Max}f_c$ の実測結果、図 4.7 (b) に累算形 ADA の動作時消費電力 P_{AT} の実測結果を示す。 $\text{Max}f_c$ 、 P_{AT} の実測結果は共に、SPICE 解析結果とよく一致している。

図 4.8 に 160-MHz 動作している改良形 3 の入出力波形を示す ($V_D = 1.4$ V、 $f_c = 160$ MHz)。(a) はオンチップ PLL の出力信号 (160 MHz) を 1/8 分周した信号波形 (20 MHz) である。(b)、(c)、(d) はそれぞれ累算回路の出力レジスタの 0b 目出力、 s_0 (80 MHz)、1b 目出力、 s_1 (20 MHz)、2b 目出力、 s_2 (5 MHz) である。

4.5 稼働率の低減による回路の低電力化

QCIF 画像、SW サイズ (p) が 10 画素、フレームレート (R_f) が 15 fps の場合、160 MHz 動作の累算形 ADA は FS を用いた ME 処理が可能である。この時、最大 420 回の BM 処理が可能である [4]。各アルゴリズムで得られる平均 N_b をこの最大 BM 回数 (420) で規格化すれば、各アルゴリズムを処理する累算形 ADA の平均稼働率、 α として定義できる。BM が終了した時点

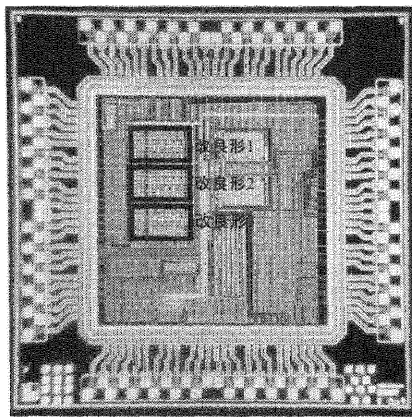


図 4.6 累算形 ADA (改良形 1、改良形 2、改良形 3) を搭載した 0.18- μm CMOS LSI チップの拡大写真。

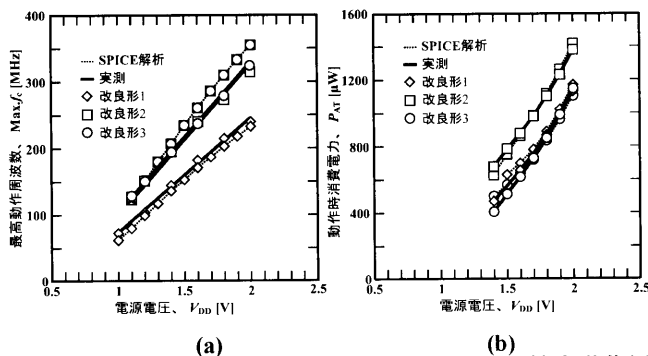


図 4.7 実測結果と SPICE 解析結果の比較。(a) 最高動作周波数、 $\text{Max.}f_c$ [MHz]。(b) 動作時消費電力、 P_{AT} [μW]。実線が実測結果、点線が SPICE 解析結果。

で、クロックパルスの供給を停止して、回路を停止すれば、累算形 ADA の P_{AT} は削減される。具体的には、BM が終了した時点で、図 4.1 から図 4.4 に示す制御パルス (CNT) を "1" として、クロックパルスを停止する ("ゲートドクロック" と呼ばれている)。

従って、各アルゴリズムを採用した累算形 ADA の平均 P_{AT} は 160 MHz 動作時の P_{AT} の α 倍 (αP_{AT}) となる。ここで、BOSA の α は 8.095 % である。BOSA を適用すると、 P_{AT} はそれぞれ 114.8 μW 、59.17 μW 、41.32 μW 、29.92 μW となり、それぞれ従来形の P_{AT} (1,418 μW) の 8.100 %、4.173 %、2.914 %、2.110 % に削減される。表 4.1 に各種アルゴリズムに対応する CMOS 累算形 ADA の P_{AT} をまとめる。

5 おわりに

既報の高速改良中断法 (HS-IBOS) は最小差分絶対値和、 D_a の減少率が連続して 0 となる回数が予め設定した回数、 N_{SC} に達したとき、ブロックマッチング (BM) を終了させる ME アルゴリズムである [6]。

本論文では、HS-IBOS を改良し、 N_{SC} をマクロブロック (M-Blk) 毎に適応的に算出する手法を確立し、さらなる高速化、高画質化を実現する、処理回数を適応的に最小化する高速中断法 (BOSA) を開発した。BOSA を QCIF テスト画像 "Carphone" (64 kbps、15 fps、 $p = 10$ 画素) に適用すると、 D_a は 1.392 となり全探索法 (FS) の $D_a (=1.335)$ と同等の結果が得られた。一方 ME 速度は FS の約 11.76 倍に、高速化できた。

BOSA、"ゲートドクロック" を適用して、4 タイプの累算形差分絶対値和回路 (従来形、改良形 1、改良形 2、改良形 3) の消費電力を低減した。160 MHz 動作の各種累算形 ADA に BOSA を適用すると、動作時消費電力 P_{AT} はそれぞれ 100.1 μW 、59.17 μW 、62.65 μW となり、それぞれ従来形の P_{AT} (1,418 μW) の 8.100 %、4.173 %、2.914 %、2.110 % に削減された。

表 4.1 0.18- μm CMOS 各種累算形 ADA の特性。

アーキテクチャ	従来形	改良形 1	改良形 2	改良形 3
パイプライン段数	1	2	2	2
総論理ゲート数 (G)	373	431	463	418
CP ゲート段数 (G_c) (1 段目、2 段目)	32	13, 19	13, 12	13, 13
最小電源電圧 (V_{DD}) [V]*	2.19	1.54	1.23	1.23
動作時消費電力 (P_{AT}) [μW]*	1,418 (100.0%)	731.0 (51.54%)	510.5 (35.99%)	369.6 (26.06%)
BOSA を適用した 動作時消費電力 (P_{AT}) [μW]*	114.8 (8.100%)	59.17 (4.173%)	41.32 (2.914%)	29.92 (2.110%)

*160 MHz 動作時、()内は従来形との比較値。

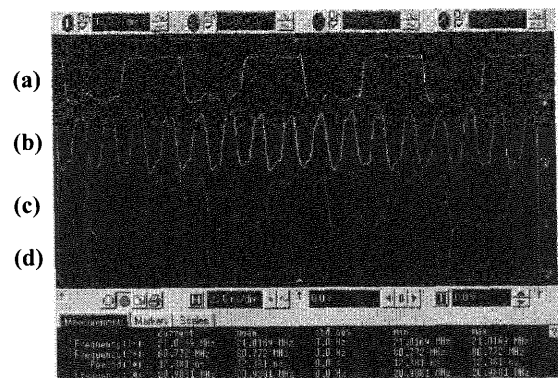


図 4.8 改良形 3 の入出力波形。(a) 1/8 分周したオンチップ PLL の信号 (20 MHz) 波形。(b) 0b 目出力、 s_0 (80 MHz)。(c) 1b 目出力、 s_1 (20 MHz)。(d) 2b 目出力、 s_2 (5 MHz)。

謝辞

榎本忠儀教授には、研究を進めるにあたり御指導頂き、大変お世話になりました。中央大学榎本研究室の皆様にもいろいろ力を貸して頂きました。皆様に、ここに深く御礼を申し上げ、感謝いたします。また、0.18- μm CMOS チップ試作は東京大学大規模集積システム設計教育センターを通し、株式会社日立製作所および大日本印刷株式会社の協力で行われたものであります。

参考文献

- [1] 榎本、笹島、廣部、「中断法動きベクトル検出アルゴリズム」、信学総合大会講演論文集、エレクトロニクス 2、SC-11-6、p.307、1997 年 3 月。
- [2] T. Enomoto and A. Kotabe, "Fast Motion Estimation Algorithm and Low-Power CMOS Motion Estimator for MPEG Encoding", IEICE Trans. Electron., vol. E86-C, no. 4, pp. 535-545, April 2003.
- [3] T. Enomoto and A. Kotabe, "Fast Motion Estimation Algorithm and Low Power 0.13 μm CMOS Motion Estimation Circuits", in Proc. of International Symposium on Circuits and Systems (ISCAS'2001), Sidney, Australia, vol-II, pp. 449-452, May 2001.
- [4] 江井、小田部、原田、榎本、「2 ステップ改良中断法動画像動きベクトル検出アルゴリズムの探索領域最適化と低電力 0.13- μm CMOS 差分絶対値和回路」、信学技報、ICD2001-39, pp. 25 - 32, 2001 年 8 月 2 日。
- [5] 江井、榎本、「MPEG-4 動き補償用 0.13- μm CMOS 差分絶対値和回路の低電力化」、信学技報、ICD2002-58, pp. 73-78、2002 年 8 月。
- [6] 渡邊、榎本、「高速改良中断法動きベクトル検出アルゴリズムと低消費電力 CMOS 動きベクトル検出回路の開発」、信学技報、ICD2003-112, pp. 37 - 42, 2003 年 10 月 23 日。
- [7] 小林、渡邊、榎本、「BM 回数を適応的に制御する高速中断法動きベクトル検出アルゴリズム」、信学総合大会講演論文集、情報・システム 2、D-11-41、p.41、2004 年 3 月 23 日。
- [8] 榎本、「サブ 100-nm デジタル LSI の低消費電力技術」、信学技報、ICD2004-16, pp. 15 - 20, 2004 年 5 月 20 日。