

2次元 Max-Plus ウェーブレット変換・逆変換実証回路の作製

小野 雅晃^{a)}、延原 肇^{b)}

^{a)} 筑波大学システム情報工学等技術室（装置開発班）

^{b)} 筑波大学システム情報工学研究科知能機能システム専攻

〒305-8573 茨城県つくば市天王台 1-1-1

概要

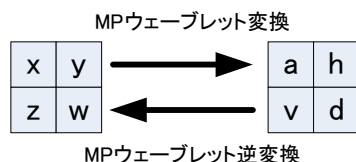
CMOS (Complementary Metal Oxide Semiconductor) カメラで撮影した白黒画像を FPGA (Field-Programmable Gate Array) を使用したハードウェアで、リアルタイムに Max-Plus 代数を使用したウェーブレット変換 (Max-Plus ウェーブレット変換) を行なった。変換結果は見やすい状態に変換して、ディスプレイに表示した。また、閾値処理後に、Max-Plus ウェーブレット逆変換を行って画像を復元し、ディスプレイに表示した。これによって、閾値の値による画像劣化状況を見ることができるようになった。更にハフマン符号化による画像の圧縮率を表示した。

キーワード：Max-Plus 代数、ウェーブレット、FPGA、ハフマン符号

1. はじめに

筑波大学システム情報工学研究科、知能機能システム専攻の延原 肇講師より、Max-Plus ウェーブレット (以下 MP ウェーブレット) 変換実証用ハードウェアの依頼があった。依頼を達成するための第 1 段階として、640×480 ピクセル (VGA 解像度) の CMOS カメラの画像を白黒画像でディスプレイに出力する回路を作製した。この結果は、平成 21 年度高エネルギー加速器研究機構技術研究会で、“FPGA を使用した CMOS カメラ・ディスプレイ回路の製作”^[1] というタイトルで発表した。

今回は、第 1 段階の回路を元に、依頼のあった MP ウェーブレット変換・逆変換回路を作製した。更にハフマン符号化回路を追加して、画像をリアルタイムに圧縮して、圧縮比を表示した。



$$\begin{aligned}
 a &= \min(x, y, z, w); & x &= a + \max(-d, -h, -v, 0); \\
 h &= y - x; & y &= x + h; \\
 v &= z - x; & z &= x + v; \\
 d &= w - x; & w &= x + d;
 \end{aligned}$$

図 1. 2次元 MP ウェーブレット変換式、逆変換式

2. 2次元 MP ウェーブレット変換

MP ウェーブレット変換は、ウェーブレット変換の一種で、演算過程に掛け算、割り算を用いることなく、比較演算 (Max, Min) と加算 (減算) を用いる^[2,3]。2次元 MP ウェーブレット変換の変換式と逆変換式を図 1 に示す。

3. ディスプレイ出力方法

ディスプレイは 640×480 ピクセルの画素を表示している。いわゆる VGA (Video Graphics Array) と呼ばれる画面解像度である。図 2 にそのピクセルの配置を示す。

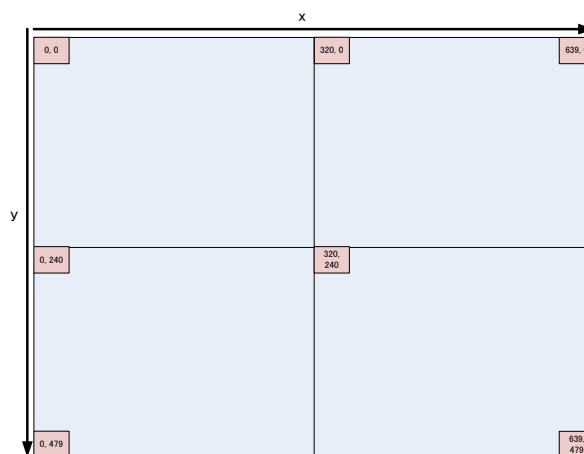


図 2. ディスプレイ上でのピクセルの配置

ここでは、(0,0)、(639,479) などの(x,y) の位置で、ピクセルの位置を示す。ディスプレイは、x 方向に走査を行って、画面を表示している。最初に(0,0)のピクセルを表示して、其の次は(1,0)、ラインの最後には(639,0)のピクセルを表示する。これで 1 ラインの描画が終了する。ラインの最後には水平同期信号が出力されて、次のラインの表示が始まる。次に、(0,1)から(639,1)で表される 2 番目のラインを表示する。2 番目ラインが終了したら 3 番目のラインを表示する。このように、480 番目のラインを表示 (最後のピクセルは(639,479)) して終了となる。その後、垂直同期後に次のフレームを表示し始める。

通常では、フレーム・バッファと呼ばれる RAM (Random Access Memory) にピクセルのデータが一時保存され、VGA ディスプレイ・コントローラがフレーム・バッファから読みだして、ディスプレイに表示する。RAM のアドレスは(x,y)の様に二次元では

なく、1次元であるので、連続的なアドレスが使用される。つまり図2で示した(0,0)のアドレスは0番地に相当し、(639,0)のアドレスは639番地に相当する。2番目のラインの先頭ピクセル(0,1)は640番地、最後の(639,479)は307,199番地に相当する。

図2の様に、画面を4分割すると左上の画面は(0,0)0番地から始まる。右上の画面は(320,0)320番地から始まる。左下の画面は、(0,240)204,800番地から始まる。右下の画面は(320,240)205,120番地から始まる。このように4画面分割して表示するためには、それにあったアドレスに画像データを書きこむ必要がある。2次元MPウェーブレット変換結果の表示に4分割画面を使用している。このアドレス演算が今回、一番デバックに苦労した箇所である。

4. 2次元 MAX-PLUS ウェーブレット変換・逆変換実証装置の概要

2次元Max-Plus ウェーブレット変換・逆変換実証装置(以下MPW実証装置)は、CMOSカメラ、FPGA、SRAM(Static Random Access Memory)、D/A(Digital/Analog)コンバータ、7セグメントLED3個、スイッチ3個とディスプレイで構成される。FPGA、SRAM、A/Dコンバータは、プリント基板に実装されている。その接続図を図3に示す。

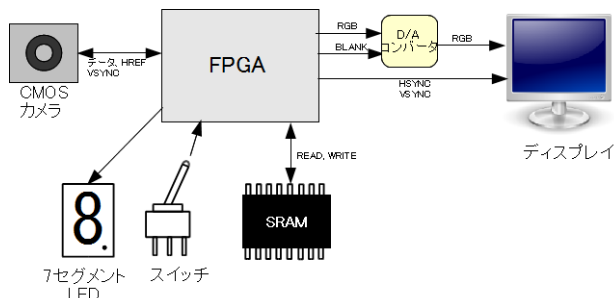


図3. MPW実証装置の接続図

実際の論理回路は、FPGAの内部に構成されている(FPGA内部回路)。FPGA内部回路は、CMOSカメラの白黒画像データをFPGAで受けて、SRAMに書き込む。その際に2次元MPウェーブレット変換及び逆変換を行う。変換・逆変換はスイッチで切り替え可能となっていて、変換なし、変換のみ、変換・逆変換を切り替えることができる。逆変換を行う場合には、閾値処理を施すが、7セグメントLEDは、その閾値を表示するために使用される。閾値処理とは、7セグメントLEDに表示された値以下の値を全て0に置き換える処理である。

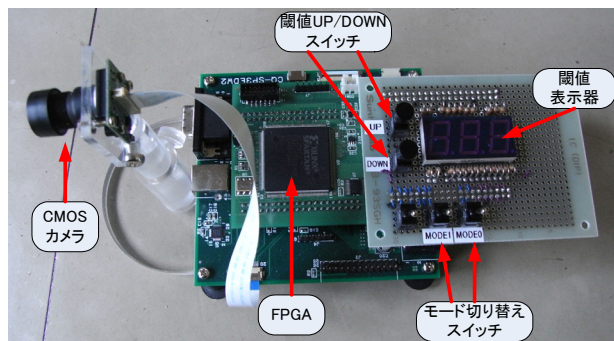


図4. MPW実証装置の写真

VGA解像度のCMOSカメラの8ビット幅の白黒画像データは、SRAMからFPGAへ読み込まれ、RGB信号とHSYNC(Horizontal synchronization signal)、VSYNC(Vertical synchronization signal)に変換されてFPGAから出力される。RGB信号はA/Dコンバータに入力されて、デジタルデータからアナログデータに変換される。A/Dコンバータから出力されたRGB信号はディスプレイに入力される。HSYNCとVSYNCはFPGAからディスプレイに入力される。なお、FPGAから出力されるRGB信号は白黒画像であるため同じ値である。ディスプレイの表示ピクセル数は640×480ピクセルである。

図4にMPW実証装置の写真を示す。ディスプレイは写真から省かれている。CMOSカメラがスタンドに搭載され、CMOSカメラからのケーブルがプリント基板(以下基板とする)に接続されている。SRAMはFPGAと同じ基板に搭載されていて、A/Dコンバータは、その下のベースとなる基板に実装されている。一番上のユニバーサル・ボードに7セグメントLED(閾値表示器)、トグルスイッチ(モード切り替えスイッチ)、プッシュスイッチ(閾値UP/DOWNスイッチ)が搭載されている。

5. FPGA内部回路の説明

FPGA内部回路のブロック図を図5に示す。

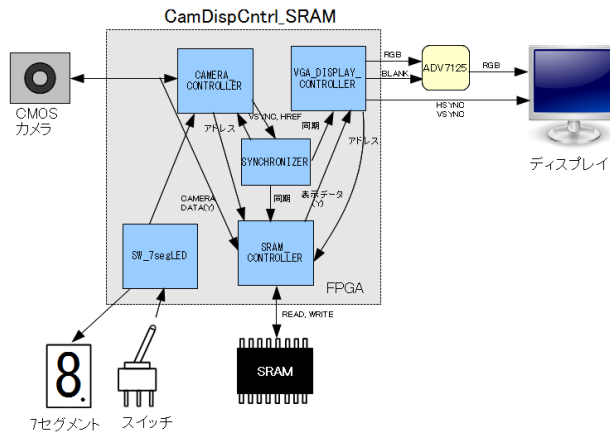


図5. FPGA内部モジュールのブロック図

図5でADV7125は、使用しているD/Aコンバータの実際の型番である。また、CamDispCntrl_SRAMは、FPGA内部回路のトップモジュールの名前を示す。

FPGA内部回路は、CAMERA_CONTROLLER、SYNCHRONIZER、VGA_DISPLAY_CONTROLLER、SRAM_CONTROLLER、SW_7segLEDモジュールで構成される。

最初にMPウェーブレット変換なしの状態を説明する。CMOSカメラから入ってきた映像信号YUV4:2:2は、CAMERA_CONTROLLERで処理され、Y信号(輝度信号)のみがSRAM_CONTROLLERに渡される。SYNCHRONIZERは、CAMERA_CONTROLLERからVSYNC、HREFを受け取って同期信号をCAMERA_CONTROLLER、SRAM_CONTROLLERとVGA_DISPLAY_CONTROLLERに渡す。その同期信号でCAMERA_CONTROLLER、SRAM_CONTROLLERとVGA_DISPLAY_CONTROLLERは、フレイ

ムの最初の垂直映像ラインと水平映像ドットに同期する。

VGA_DISPLAY_CONTROLLERはSRAMからREADした輝度信号をRGBに変換(白黒なので、RGB同じ値にする)し、D/Aコンバータ(ADV7125)に出力する。HSYNC、VSYNCは直接VGAコネクタに出力する。

SW_7segLEDは、モード切り替えスイッチの値で動作モードを切り替える。MODE0、MODE1スイッチの値による動作モードの表を表1に示す。

表1. モード切り替えスイッチによる動作モードの違い

MODE1	MODE0	動作
OFF	OFF	動作モード1 原画像表示モード
OFF	ON	動作モード2 Max-Plusウェーブレット変換表示モード
ON	OFF	動作モード3 Max-Plusウェーブレット変換逆変換表示モード
ON	ON	動作モード1→動作モード2→動作モード3→動作モード1を5秒間隔で繰り返す

表1の動作モード1は、原画像表示モードで、CMOSカメラからの輝度信号を直接ディスプレイに表示するモードである。動作モード2は、MPウェーブレット変換後の画像を4分割してディスプレイに表示するモードである。動作モード3はMPウェーブレット変換後の画像に閾値処理を施してから、もう一度MPウェーブレット逆変換した画像を表示するモードである。このモードは閾値処理によって、どの程度、画像が劣化するかを評価するためである。動作モード4は、表1の一番下に示すモードで、動作モード1から動作モード3までを5秒間隔で繰り返して表示するモードである。

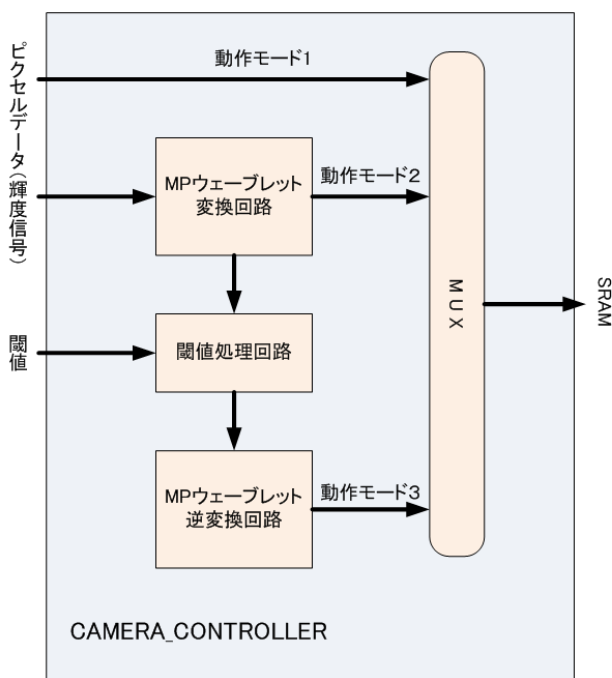


図6. 2次元MPウェーブレット変換・逆変換回路ブロック図

6. 2次元MPウェーブレット変換・逆変換回路

2次元MPウェーブレット変換・逆変換回路(以下、MPウェーブレット変換・逆変換回路)は、CAMERA_CONTROLLERに内蔵されていて、2次元MPウェーブレット変換回路(以下、MPウェーブレット変換回路)、閾値処理回路、2次元MPウェーブレット逆変換回路(以下、MPウェーブレット逆変換回路)に分かれている。そのブロック図を図6に示す。

ピクセルデータ(輝度信号)は、マルチプレクサ(MUX)とMPウェーブレット変換回路に入力される。動作モード1の場合は、MUXに入力されたピクセルデータが直接SRAMに書き込まれる。動作モード2の場合はMPウェーブレット変換回路で変換されたデータがSRAMに書き込まれる。動作モード3の場合はMPウェーブレット変換回路で変換されたデータが閾値処理回路を通して閾値処理をした後で、MPウェーブレット逆変換回路に入力され、逆変換された画像がSRAMに書き込まれる。SRAMに書きこまれたデータはVGA_DISPLAY_CONTROLLERで読み出されてディスプレイに表示される。

6.1 MPウェーブレット変換回路

MPウェーブレット変換回路はCAMERA_CONTROLLERの中に構成されていて、動作モード2と動作モード3(表1参照)で使用される。MPウェーブレット変換の変換画像を図7に示す。図7の左上が図1の変換式でのa、右上がh、左下がv、右下がdの値を集めた画像になっている。通常では、2×2ピクセルのデータごとに変換値が出るため、意味のある画像にはならないが、a、h、v、dそれぞれについて、書きこむ番地を変更することで、図7のようなMPウェーブレット変換画像を得ることができる。具体的には、図7のaが4つのピクセルの最小値、hが横の差分、vが縦差分、dが斜めの差分を表す。(図7の写真は差分の画像に露出を合わせたので、aが白飛びしているが、実際は綺麗に表示されている)

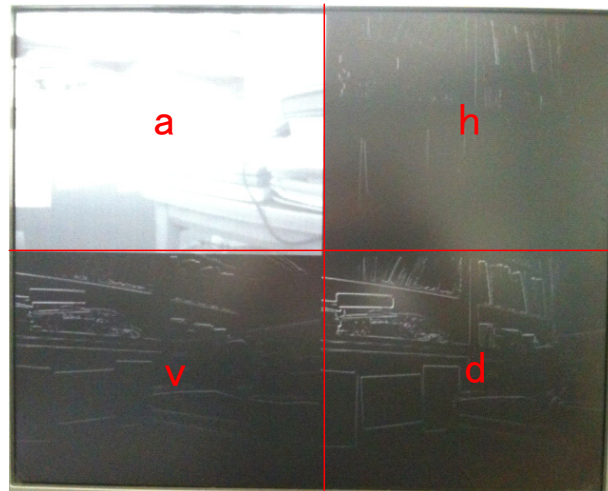


図7. MPウェーブレット変換後の画像

画像を図1の変換式でMPウェーブレット変換するためには、1ライン分(640ピクセル)のライン・

バッファが必要となる。このライン・バッファには、FPGA に内蔵された Block RAM を使用した。下に MP ウェーブレット変換の概念的なブロック図を図 8 に示す。

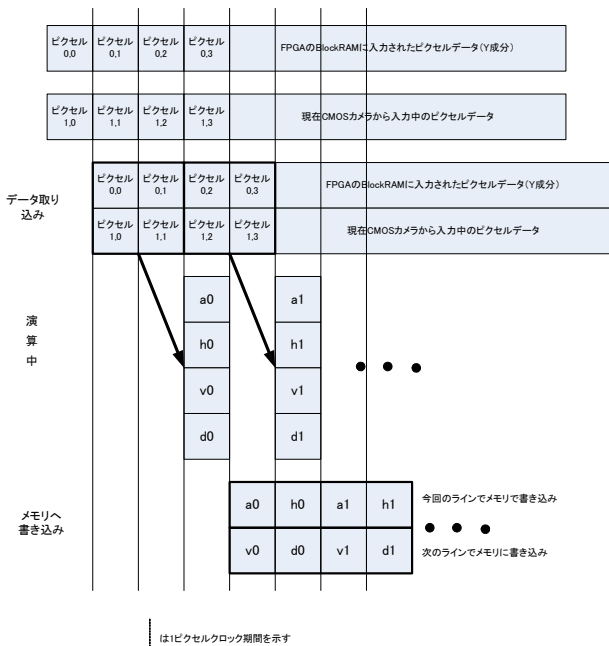


図 8. MP ウェーブレット変換の概念的なタイミングのブロック図

CMOS カメラの映像データの 0 ライン目は、Block RAM に 1 ライン分の画像データを読み込む。次に、0 ライン目の画像データ (Block RAM から読みだして用意したデータ) と現在の CMOS カメラから入力される 1 ライン目の映像データを使用して、MP ウェーブレット変換を行う。それらのデータは FF (Flip-Flop) に入力され、データ取り込みフェーズで FF から出力されている。2 クロック目にピクセル 0,0、ピクセル 0,1、ピクセル 1,0、ピクセル 1,1 が揃うので、MP ウェーブレット変換を行うためのデータが揃うことになる。

データが、揃ったら図 1 の式で 2 次元 MP ウェーブレット変換を行う。それが図 8 の演算中の矢印にあたる。演算後に a0, h0, v0, d0 が出力される。出力された値は演算によって a0 以外は、マイナスの場合が含まれるので、8 ビット幅から 9 ビット幅に拡張されているが、その絶対値を取って 8 ビット幅の値

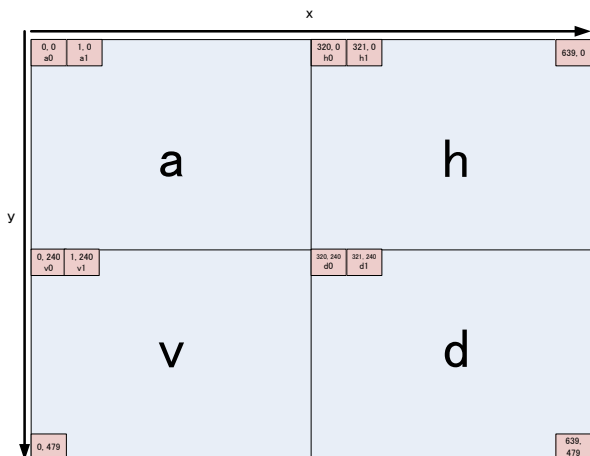


図 9. a, h, v, d のディスプレイでの表示位置

としてメモリ (SRAM) に書き込まれる。SRAM は 16 ビット幅なので、a0 と a1 を同じアドレスに書き込むことができる。

この時、図 7 に示したようにディスプレイに表示するためには、a, h, v, d のそれぞれに異なったオフセットのアドレスで SRAM に書きこむ必要がある。具体的なディスプレイの表示位置を図 9 に示す。

a0 のアドレスは 0 番地、a1 のアドレスは 1 番地、h0 のアドレスは 320 番地、h1 のアドレスは 321 番地、v0 のアドレスは 204,800 番地、v1 のアドレスは 204,801 番地、d0 のアドレスは 205,120 番地、d1 のアドレスは 205,121 番地となる。

更に、SRAM に書きこむ際に a と v, h と d のアドレスが異なるため同時に書くことはできない。そのため、v と d は、次のラインで SRAM に書きこむことにした。次のラインで書き込むために一時的に FPGA 内のライン・バッファ (Block RAM) に書きこんで、次のラインで読み出しながら SRAM に書きこむことにした。幸いなことに偶数ラインはピクセルデータのライン・バッファへの書き込みのみを行って、MP ウェーブレット変換の演算を行っていないため、v と d を SRAM に書きこむタイミングが存在する。

次に、この概念モデルを実際の回路に落とす作業を行った。図 8 では、SRAM は書き込み専用になっているが、実際には、Write/Read を 1 クロックごとに交互に繰り返している。Read は VGA_DISPLAY_CONTROLLER が 2 バイト (16 ビット) 分のピクセルデータを SRAM から読みだして、ディスプレイに出力するために使用される。SRAM の Write/Read を考慮したブロック図を図 10 に示す。

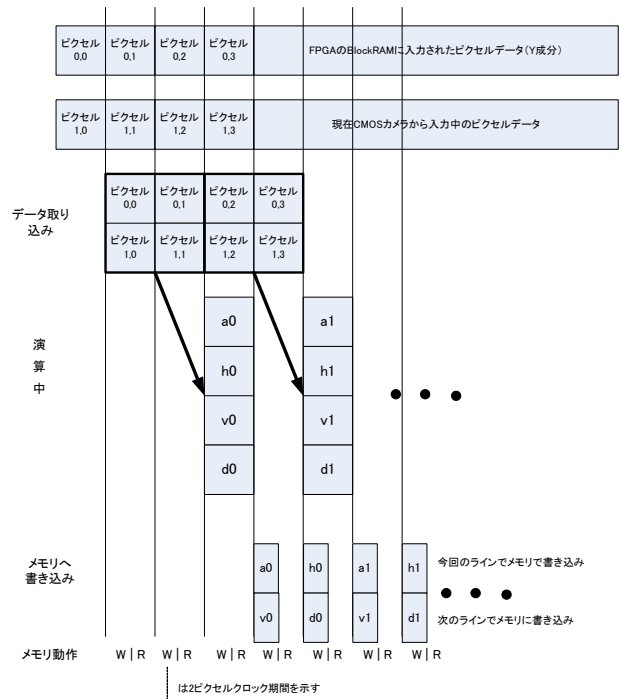


図 10. MP ウェーブレット変換のタイミング・ブロック図

図 10 は図 8 に SRAM の Read タイミングを追加し、より実際の動作に近いタイミング・ブロック図である。表 1 に示した動作モード 2 の場合は、以上述べ

てきたように SRAM に MP ウェーブレット変換されたデータが書き込まれて、ディスプレイに表示される。

6.2 閾値処理回路

図 7 の MP ウェーブレット変換後の画像を見ると分かるが、 h , v , d は隣り合ったピクセル同士の差分であるため、値は小さい方に偏っている。表示された物体のエッジは輝度が大きく変わるので、白く表示されている。このように、差分データは、小さい値には、重要なデータが少ない。そこで、この部分を閾値処理して、ある一定数までのデータを 0 に落とすと、データが少なくなり、データをより圧縮することが可能となる。閾値は図 4 に示す閾値表示器に表示され、閾値 UP/DOWN スイッチで増減することができる。

6.3 MP ウェーブレット逆変換回路

MP ウェーブレット逆変換回路は、ピクセルデータを MP ウェーブレット変換し、閾値処理をしたデータを MP ウェーブレット逆変換して元のピクセルデータに戻す処理である。MP ウェーブレット変換・逆変換は可逆変換であるため、閾値処理でデータを落とさなければ、品質を落とさないままに元のデータに戻すことができる。MP ウェーブレット逆変換のブロック図を図 11 に示す。

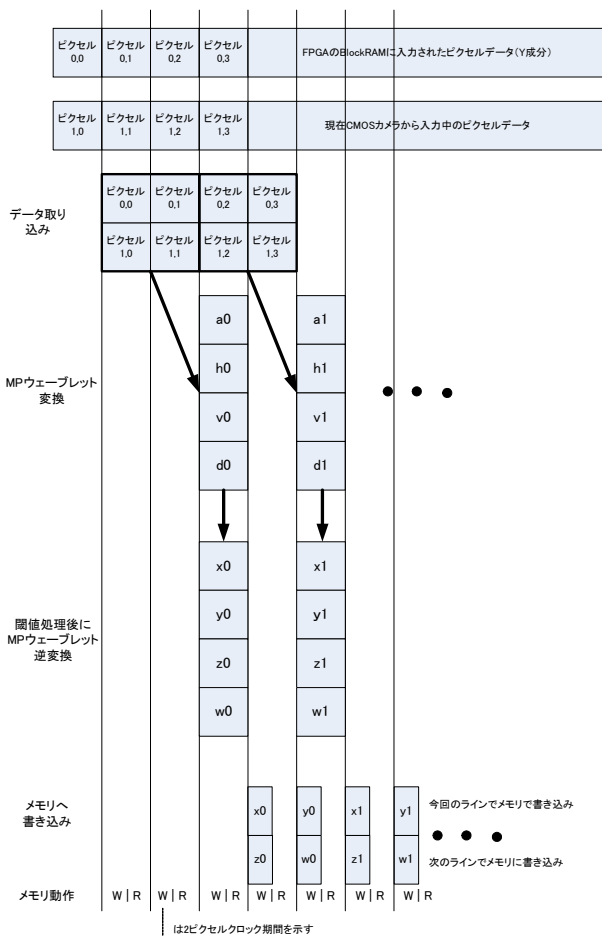


図 11. MP ウェーブレット逆変換のブロック図

MP ウェーブレット逆変換の画像を表示するモードは表 1 では動作モード 3 となる。

MP ウェーブレット変換を画像圧縮に使用するのであれば、MP ウェーブレット変換後に閾値処理をし、ハフマン符号化などのエントロピー符号化をして、他の機器に伝送する。そこで、エントロピー符号化を復号してから、MP ウェーブレット逆変換を行って画像を復元するという使い方をするのが普通である。ここでは、閾値処理による画像の劣化を見るために、同じ機器上で、リアルタイムに閾値処理、MP ウェーブレット逆変換を行って、画像を復元している。

7. ハフマン符号化回路の追加

説明してきた MP ウェーブレット変換・逆変換回路にハフマン符号化回路を追加した。ハフマン符号とは、1952 年にデビット・ハフマンによって開発された符号で、コンパクト符号やエントロピー符号の一つである^[4,5]。簡単に言うと出現確率の高い値に短い符号長の符号を割り当てる方式である。出現確率の低い値は符号長が長くなる。このハフマン符号を用いて、画像の圧縮を行った。以下に示す手順で、ハフマン符号化回路を実現した。

1. レナ像を図 4 の MPW 実証回路で撮影し、同時に MP ウェーブレット変換を行って、1 フレーム分の a , h , v , d の値を集計した。
2. 1. の値を ChipScope Pro の VIO (Virtual Input/Output) 機能を使用してデータをパーソナル・コンピュータ(PC)に取り込んだ。
3. 2.のデータを元にソフトウェアでハフマン符号化を行った。
4. ハフマン符号をソフトウェアで処理してハフマン符号テーブルと逆ハフマン符号テーブル

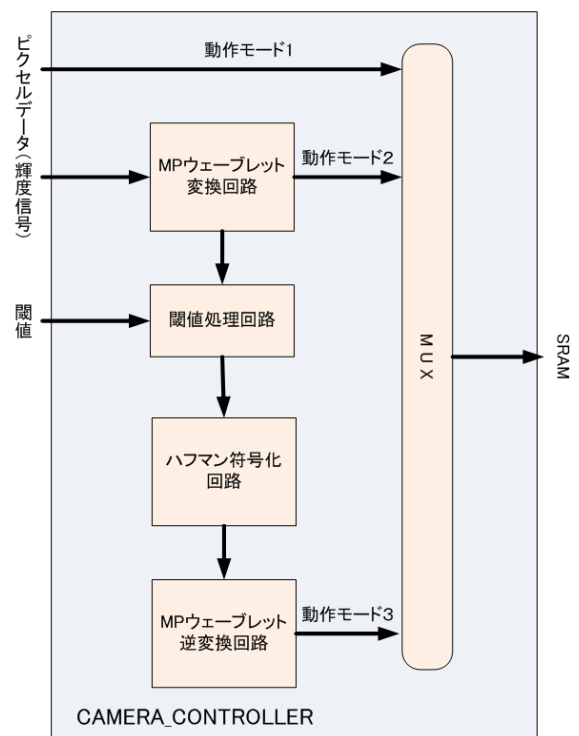


図 12. 構想段階の CAMERA_CONTROLLER 内のブロック図

を作製した。

- ハードウェアを VHDL (VHSIC(Very High Speed Integrated Circuits)Hardware Description Language)で記述した。その際に、生成した ROM にハフマン符号化テーブルを初期値としてロードした。

構想段階のハフマン符号化回路は、図 12 に示すように、閾値処理回路と MP ウェーブレット逆変換回路の間に挿入することにした。

構想段階のハフマン符号化回路のブロック図を図 13 に示す。

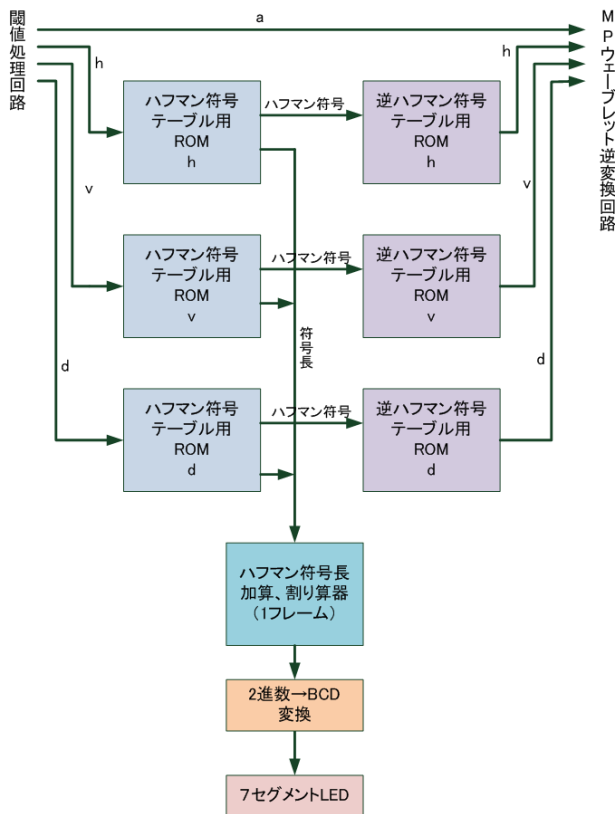


図 13. 構想段階のハフマン符号化回路のブロック図

CMOS カメラで撮影した画像を MP ウェーブレット変換し、ハフマン符号化テーブルでハフマン符号に変換する。ハフマン符号はすぐに逆ハフマン符号テーブルに送られて、MP ウェーブレット逆変換を行って、ディスプレイに表示される。

符号長はハフマン符号長加算、割り算器で 1 フレーム分加算され、1 フレーム終了後に元のデータ長で割られて、パーセント表記に変換される。その後、2進数から BCD (Binary-Coded Decimal) に変換されて、7セグメント LED に表示される。こうすることによって、1 フレームごとに圧縮率をリアルタイムに表示することができる。7セグメント LED は、閾値を表示していたので、図 4 の MPW 実証回路のもう 1 つのスイッチで、閾値と圧縮率のパーセント表示を切り替えることにした。図 13 において、a はハフマン符号化されていないが、これは、明るさ等によって、値が変動しやすいので、ハフマン符号化に向かないと考えたからである。CMOS カメラには AGC (Automatic Gain Control) が付いているが、やはり明るさで値が変動する。

なお、本来の用途では、ハフマン符号化されたデータはパッキングされて、他の機器に伝送され、そこで逆ハフマン符号化されて、MP ウェーブレット逆変換されて画像データに戻されるという手順を踏む。しかし、ここでは、ハフマン符号化のテストのために、同一機器でハフマン符号化-逆ハフマン符号化を行うことにした。

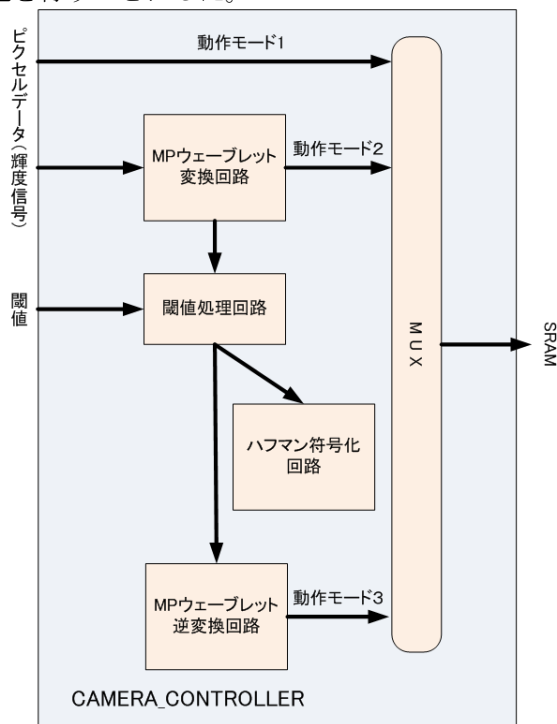


図 14. 実装した CAMERA_CONTROLLER 内のブロック図

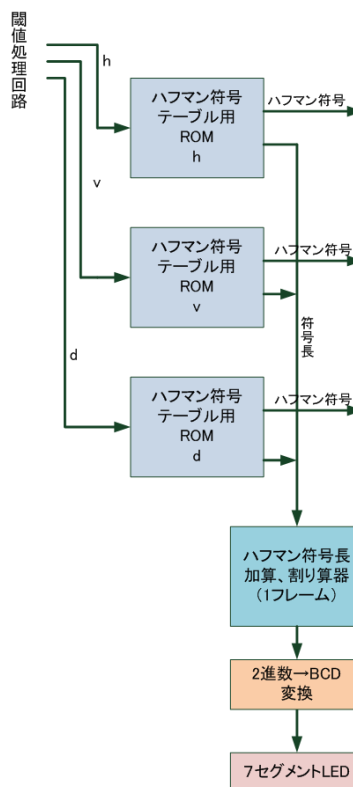


図 15. 実装したハフマン符号化回路のブロック図

ハフマン符号化回路をハードウェア化していくと逆ハフマン符号テーブルが大規模の ROM を必要とするということがわかった。逆ハフマン符号テーブルの通常の実装は FPGA リソースの面からも難しいのでやめることにした。SRAM に書きこむデータは前回のように MP ウェーブレット変換・逆変換したピクセルデータを SRAM に出力することにして、ハフマン符号化回路は圧縮率の計測のみに使用することにした。実際に作成した CAMERA CONTROLLE R 内のブロック図を図 14 に示す。実際にハードウェアにしたハフマン符号化回路のブロック図を図 15 に示す。

図 15 に示すように、ハフマン符号テーブル ROM の符号長のみを積算し、総ビット数で割り算した値を BCD 変換して、7 セグメント LED に圧縮率として表示した。実際に圧縮率を測定したところ、レナ像では約 58%、部屋の中を撮影した結果は約 65% だった。

8. まとめ

CMOS カメラで撮影した白黒画像を FPGA に入力して、MP ウェーブレット変換を行い、それぞれの成分に分けてディスプレイに表示した。また、MP ウェーブレットした画像を閾値処理後に MP ウェー

レット逆変換を行って、ディスプレイに表示して、閾値処理での画像の劣化を確認することができた。

MP ウェーブレット変換後のデータをハフマン符号化して、ハフマン符号化でのデータ圧縮率をリアルタイムに表示することができた。

なお、第 16 回画像センシングシンポジウム、イノベーションジャパン 2010 ほか、国際会議 ISCIT において発表を行っている。

参考文献

- [1] 小野雅晃, FPGA を使用した CMOS カメラ・ディスプレイ回路の製作, 平成 21 年度高エネルギー加速器研究機構技術研究会報告集
<http://www-eng.kek.jp/meeting09/proceedings/pdf/h21g3002.pdf>
- [2] H. Nobuhara, M. Ono, T. Tanabata, B. Bede, 3D-Wavelet Decomposition Based on Max-Plus Algebra and its Application to Video Coding, 10th International Symposium on Communications and Information Technologies (ISCIT2010), Meiji University, Tokyo, Japan, Oct. 26-29, (2010)
- [3] 延原肇, 小野 雅晃, 七夕高也, マックスプラス代数に基づく三次元ウェーブレット変換の提案とコンテンツ・センシングへの展開, 第 16 回画像センシングシンポジウム, パシフィコ横浜, 2010 年 6 月 10 日
- [4] 昌達 K'z, 圧縮アルゴリズム, C magazine
- [5] ウィキペディア, ハフマン符号,
<http://ja.wikipedia.org/wiki/%E3%83%8F%E3%83%95%E3%83%9E%E3%83%B3%E7%AC%A6%E5%8F%B7>