

# FPGA を使用した CMOS カメラ・ディスプレイ回路の製作

小野雅晃

筑波大学 システム情報工学等支援室 装置開発班

## 概要

CMOS カメラから出力されたデータを使用して、ディスプレイに表示する回路を FPGA (Field Programmable Gate Array) で作成した。CMOS カメラから出力された YUV422 データのうちの Y (輝度) データを FPGA に取り込んで、一旦 SRAM (Static RAM) に記憶する。その記憶した Y データを再び FPGA に取り込んで VGA (Video Graphics Array) 信号を生成してディスプレイに送り、白黒画像を表示した。

## 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、延原講師から VGA 信号をハードウェアで圧縮したいとの依頼を受けた。そこで、イーエスピー企画の画像ベースボード、デジタル CMOS カメラ、208 ピン Spartan3E XC3S500+2M 高速 SRAM 基板<sup>[3],[5],[6]</sup>を使って CMOS カメラからの画像を加工することにした。その第一段階として、CMOS カメラ出力された YUV422 データのうちの Y (輝度) データを、そのまま白黒画像としてディスプレイに表示してみることにした。CMOS カメラから出力される Y データは 640×480 ドット 30 フレーム/秒なので、640×480 ドット 60 フレーム/秒の VGA 信号にそのままは出力することができない。一旦 CMOS カメラから出力される Y データを SRAM に記憶し、その Y データを SRAM から読み出しながら VGA 信号を生成することにした。

画像ベースボード、デジタル CMOS カメラ、208 ピン Spartan3E XC3S500+2M 高速 SRAM 基板の写真を図 1 に示す。

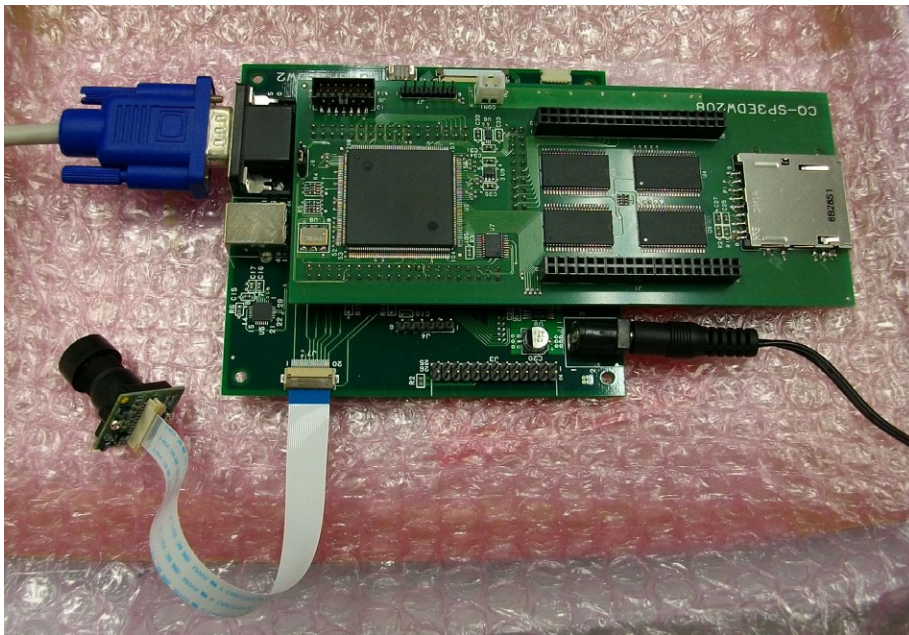


図 1: 画像ベースボード、デジタル CMOS カメラ、208 ピン Spartan3E XC3S500+2M 高速 SRAM 基板の写真

## 2 CMOSカメラの仕様<sup>[1]</sup>

CMOSカメラは、シキノハイテックのCMOSカメラKBCR-M03VGを使用している。このCMOSカメラはCMOSイメージセンサにOmniVision社のOV7640を使用している。OV7640は、SCCBシリアルインターフェースからSCCB設定レジスタを設定することにより、いろいろなモードを選択することができる。今回はリセット後のデフォルト設定を使用している。その設定はVGAモード(640×480, 30フレーム/秒)、YUV422モード、YUVフォーマットはUYVYUYVYモードである。YUVの画像データは8ビット幅で、Uが8ビット、Yが8ビット、Vが8ビット、Yが8ビットというフォーマットで出力される。白黒画像のため、そのうちのYデータのみを利用することにした。HERFのが1の間Y[7:0]には、UYVYの順番に画像データが出力される。その様子を図2に示す。図2のHREFが立ち上がった時に、最初のUデータU0、最初のYデータY0、最初のVデータV0、2番目のYデータY1という順番で出力される。

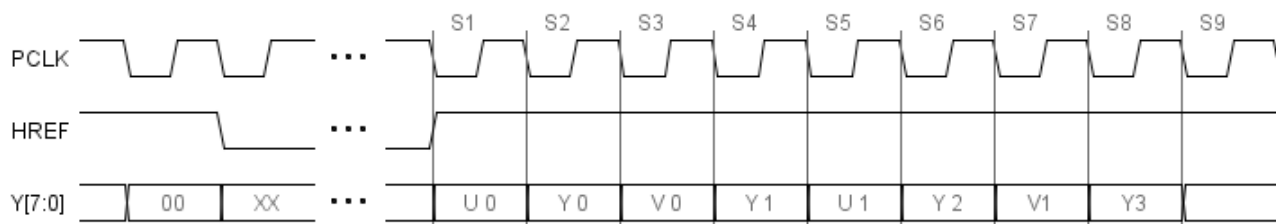


図 2: CMOSカメラからの画像データ出力タイミング<sup>[1]</sup>

CMOSカメラの1フレームのタイミング全体を図3に示す。最初にVSYNCが3 tROW (水平表示期間、1 tROW = 1528 tPCLK)の間1となる。VSYNCが0になってから11 tROW後にHREFが1となる。HREFは1280 tPCLK (PCLK期間、1 tPCLK = 1/24MHz、41.67 ns)の間1になる。HREFが1の間は図2に示すようにCMOSカメラからの画像データが出力される。その後248 tPCLK間隔が開いて、もう一度HREFが1となり、画像データが出力される。最後のHREFが1の期間の終了から31 tROW後にVSYNCが1となり、1フレームが終了する。

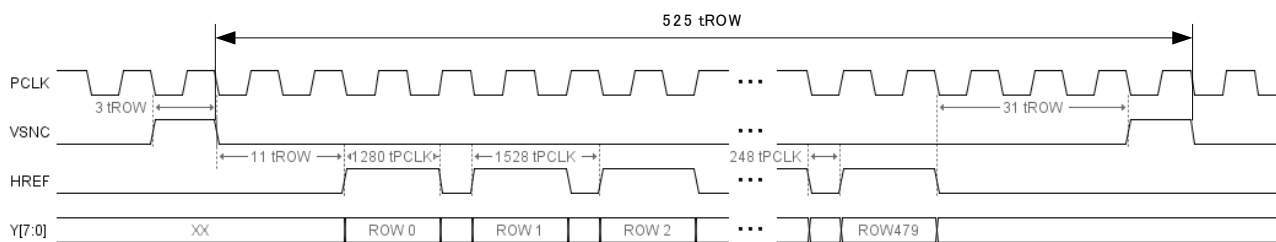


図 3: CMOSカメラの1フレームのタイミング<sup>[1]</sup>

## 3 VGA信号出力のタイミング

VGA信号出力のタイミングは、CMOSカメラの映像信号のタイミングの1/2の時間間隔に設定した。<sup>[4]</sup>このタイミングを用いると、カメラの映像1フレームにつき、VGAが2フレームとなる。カメラのVSYNC後のHREFのタイミングにVGA信号の出力を合わせることにした。VGA信号出力タイミング・パラメータを表1に示す。

表1 VGA信号出力パラメータ

ドットクロック	24MHz (CMOSカメラのPCLKに同期) 41.7ns
水平ライン	764ドットクロック 31.8us
水平表示区間	640ドットクロック 26.7us
水平フロントポーチ	16ドットクロック 0.667us
水平同期信号	64ドットクロック 2.67us
水平バックポーチ	44ドットクロック 1.83us
垂直期間	525ライン 16.7ms
垂直表示区間	480ライン 15.3ms
垂直フロントポーチ	11ライン 0.350ms
垂直同期信号	2ライン 0.0637ms
垂直バックポーチ	32ライン 1.02ms

VGA 信号のフレームレートは、表1の1/(水平期間、16.7 ms) ≒ 60Hzなので、CMOS カメラの画像データのフレームレート30Hzのちょうど2倍となっている。つまり、CMOS カメラから1枚の画像データを取り込むうちに2回ディスプレイを表示する換算となる。

#### 4 CMOS カメラ・ディスプレイ回路のブロック図

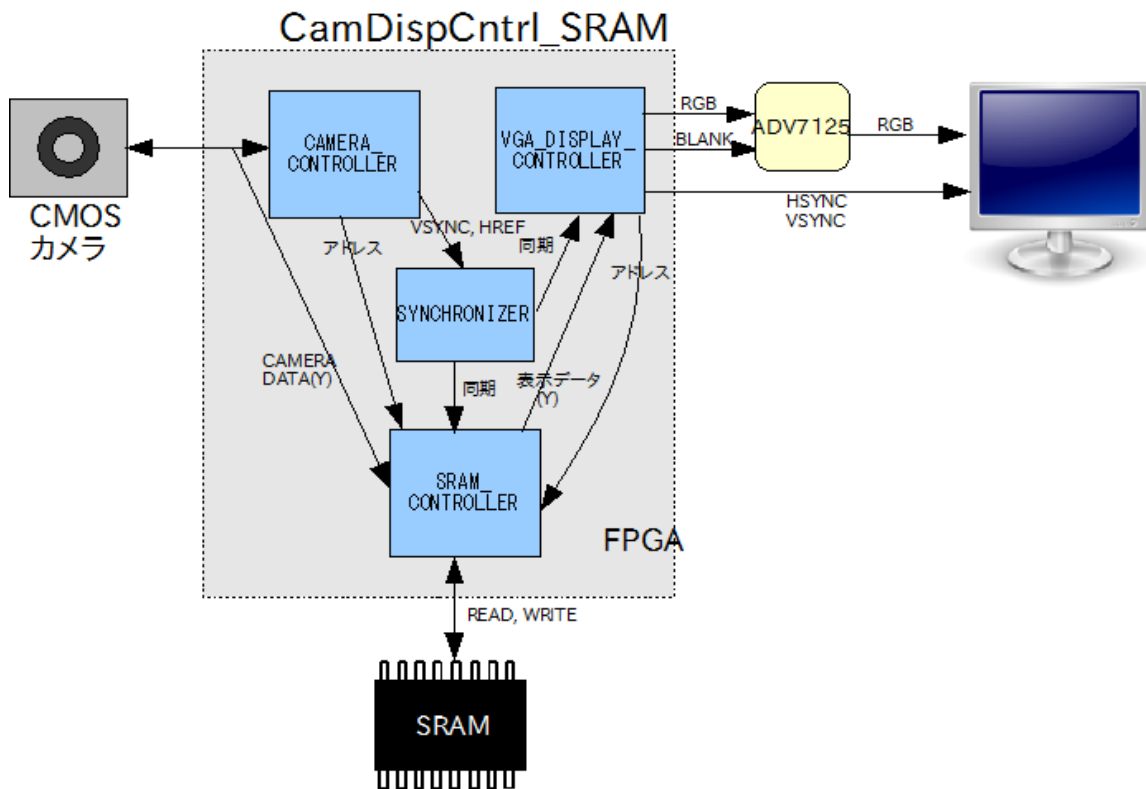


図4: CMOS カメラ・ディスプレイ回路のデータパス、制御部分のブロック図

図4に CMOS カメラ・ディスプレイ回路のデータパス、制御部分のブロック図を示す。図4に示すように CMOS カメラ・ディスプレイ回路は CAMERA\_CONTROLLER, VGA\_DISPLAY\_CONTROLLER, SYNCHRONIZER, SRAM\_CONTROLLER, そして、図5に示す DCM\_module\_24MHz で構成される。CMOS カメラから入ってきた映像信号 YUV422 は 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 SRAM\_CONTROLLER と VGA\_DISPLAY\_CONTROLLER に渡す。その同期信号で SRAM\_CONTROLLER と

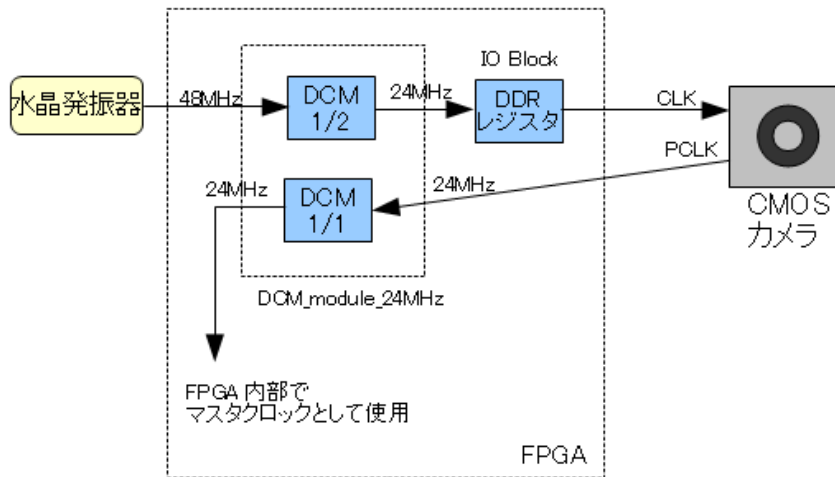


図 5: クロック関連のブロック図

VGA\_DISPLAY\_CONTROLLER は垂直 0 ライン、水平映像信号の 0 ドット目に同期する。VGA\_DISPLAY\_CONTROLLER は SRAM から READ した Y データを RGB に変換（白黒なので、RGB 同じ値にする）し、DAC(ADV7125)に出力する。HSYNC, VSYNC は直接 VGA コネクタに出力する。

図 5 に示すように水晶発振器から出力された 48MHz クロックは、FPGA の DCM で 24MHz に変換されて IO ブロックにある DDR レジスタを駆動する。DDR レジスタから出力された 24MHz のクロックは CMOS カメラの CLK に入力される。CMOS カメラから出力された PCLK (24MHz) は FPGA の別の DCM で受けた後 FPGA 内部のマスタークロック (mclk) として使用される。mclk は FPGA 全体で使用するクロックである。

なお、DCM1/2 の LOCKED 信号を DDR レジスタのイネーブル信号として使い、DCM1/2 がロックしない場合は CMOS カメラにクロックが供給されないように工夫した。DCM1/1 の LOCKED 信号は、論理を反転して、FPGA 回路全体のリセット信号として使用する (reset はアクティブハイ)。

それぞれのモジュールは VHDL(VHSIC(Very High Speed Integrated Circuits)Hardware Description Language)) で記述されている。

## 5 CMOS カメラ・ディスプレイ回路の動作タイミング

CMOS カメラ・ディスプレイ回路の動作タイミングの説明方法として SRAM をアクセスするタイミングを中心に説明する。2M バイト高速 SRAM は、アクセスタイム 10ns、16 ビット幅の SRAM (IS61LV25616AL) を使用している。アクセスタイムは 10ns だが、非同期動作なので、周期が 10ns クロックで動作することはできない。現実的には Spartan3E ボードの 48MHz クロックを 1/2 分周した 24MHz で使用する。Write 用のパルスは DDR レジスタにより出力される。DDR レジスタを使用することにより 1/2 クロック周期分アクティブ(0) とすることができた。SRAM のデータは 16 ビット幅なので、8 ビットの Y データを格納する際は最初に上位 8 ビットに Y データを書き込む。次の Y データは下位 8 ビットに書き込む。つまり 24MHz クロック 2 クロックに 1 回、8 ビットずつ書き込みを行う。図 6 に SRAM をアクセスするタイミングチャートを示す。

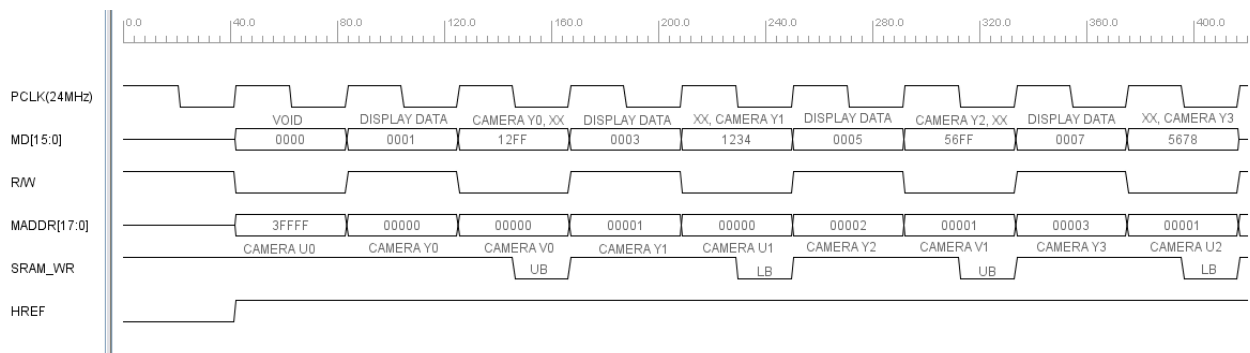


図 6: SRAM をアクセスするタイミングチャート

CMOS カメラに 24MHz クロックを供給し、カメラから帰ってくるのが図 6 に示す PCLK である。PCLK も 24MHz のクロックとなっている。

MD はメモリに読み書きするデータで、DISPLAY\_DATA の時が、表示用のデータをリードしている場合である。この時は 8 ビットの Y データ 2 つ分、2 バイトを読み出してディスプレイに表示する。2 クロックに 1 回の読み出しなので、2 バイトのデータをもらう必要がある。VOID は書き込みのタイミングだが、書き込むデータがそろっていないので書き込めないことを示す。最初の Y データ (CAMERA Y0) が来たら上位 8 ビットに書き込む(CAMERA Y0,XX)。次の CAMERA Y1 は下位 8 ビットに書き込む (XX, CAMERA Y1)。SRAM の UB, LB 信号を用いて上位 8 ビット、下位 8 ビットだけに書き込むことができる。

R/W は Read/Write を表す。MADDR は、SRAM のアドレスを示す。アドレスを使用しない場合は 3FFFF と書かれている。

その下の CAMERA U0, CAMERA Y0...は、CMOS カメラから出力される YUV422 のデータが出てくるタイミングを示している。その下が HREF の信号となる。HREF は CMOS カメラの映像信号の 1 水平フレームを表している。

## 6 CMOS カメラ・ディスプレイ回路のシミュレーション

各モジュールの VHDL ファイル作成後にシミュレーションを行った。その際に、CMOS カメラのシミュレーション用モデル(OV7640\_Model.vhd) と SRAM のシミュレーション用モデル(IS61LV25616\_model.vhd) を作成した。さらに、CMOS カメラ・ディスプレイ回路と作成したモデルを接続するテストベンチファイルを作成して ModelSim でシミュレーションを行った。いろいろな間違いがあったが一つ一つトラブルシューティングしながら正常動作に近づけていった。図 7 に ModelSim の wave 画面のシミュレーション波形を示す。

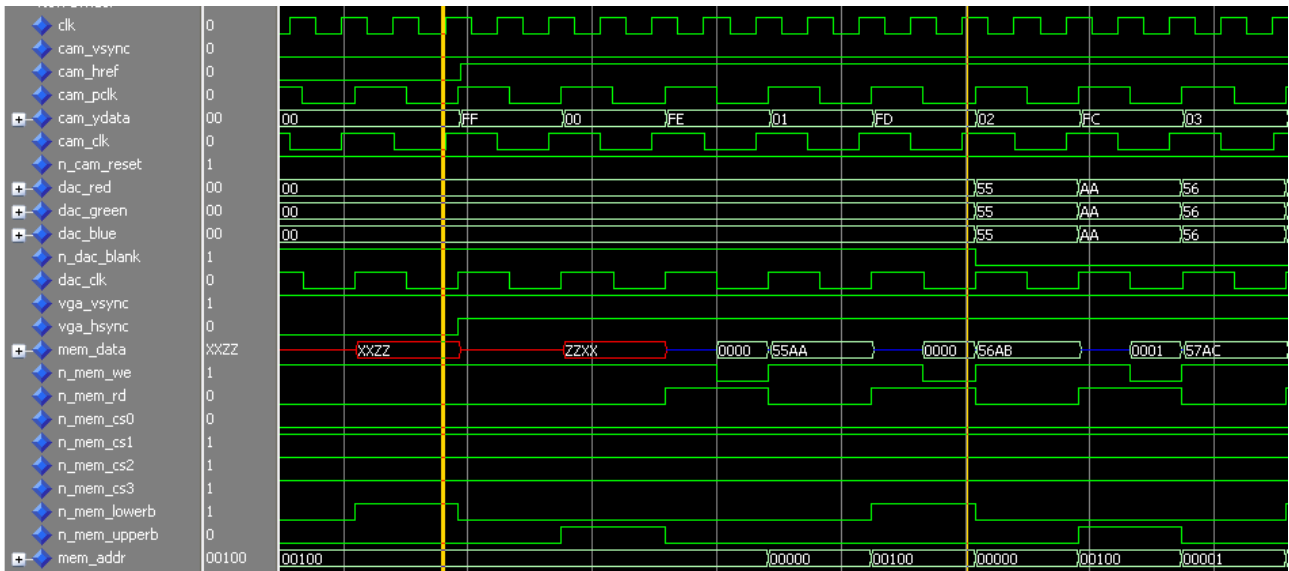


図 7: CMOS カメラ・ディスプレイ回路のシミュレーション波形 (一部)

図 7 のシミュレーション波形は CMOS カメラの VSYNC (cam\_vsync) が 1 から 0 になってから、最初の HREF(cam\_href) が 1 になった時の動作を表している。cam\_vsync が 1 になってから、初めて cam\_href が 1 になって、CMOS カメラの Y データのサンプルや VGA への映像信号の出力が開始される時点が、最初のカーソル (黄色の縦線) の位置である。cam\_ydata を見ると、FF, 00, FE, 01... と出力されている。これは CMOS カメラモデルが、U, Y, V, Y と出力しているので、そのうちの Y データ、つまり、00, 01... というようにキャプチャすればよい。

次に mem\_data を見ると、下の n\_mem\_we が 0 の時が CMOS カメラの Y データの書き込みで、書き込みアドレスは mem\_addr に出力されている。本当は 00000 からなのだが、わかりやすいように CMOS カメラのデータを書き込むアドレスはスタートを 00100 にしてある。最初の書き込みはまだデータが出力されていないのでダミーの書き込みで、次からが Y データの書き込みとなる。2 番目の n\_mem\_we が 0 の時 (CMOS カメラデータのメモリへの書き込み) は、n\_mem\_upperb = '0', n\_mem\_lowerb = '1' となっていて、上位 8 ビットへの書き込みであることが分かる。3 番目の n\_mem\_we が 0 の時は、今度は n\_mem\_upperb = '1', n\_mem\_lowerb = '0' となり、下位 8 ビットへの書き込みであることが分かる。それぞれ、"00XX", "XX01" を書き込んでいる。

n\_mem\_rd が 0 の時には、SRAM モデルの mem\_data が、55AA からスタートして、上位、下位 8 ビットごとに +1 したデータを出力している。mem\_data を見ると、n\_mem\_rd が 0 の時のデータが "55AA", "56AB"... と上位、下位 8 ビットごとに +1 されているのが分かる。このデータは、VGA\_Display\_Controller.vhd でサンプルされて dac\_red, dac\_green, dac\_blue に同じ値が出力されている (2 番目のカーソルの位置)。

## 7 CMOS カメラ・ディスプレイ回路のインプリメントと実機でのテスト<sup>[2], [7]</sup>

CMOS カメラ・ディスプレイ回路を Xilinx 社の FPGA ツール ISE11.3 でインプリメントし、実機にダウンロードしてテストを行った。その結果、CMOS カメラの VSYNC や HREF が出力されなかった。ChipScope Pro (FPGA 内のロジックアナライザ) を用いて調査を行い、CMOS カメラの RESET の極性を勘違いしていたことが分かった。CMOS カメラのデータシートには RESET の極性が書いてなかったため、RESET の極性を 0 と思いこんでいたが、実際は 1 であることが分かった。RESET を 0 に固定したところ HREF, VSYNC やデータが出力され、正常な表示ではないが、図 8 に示すようにディスプレイに表示が出始めた。



図 8: 正常な表示でないときのディスプレイの画像

いろいろトラブルシュートを試みたが、バグの原因はVGA\_DISPLAY\_CONTROLLERのアドレスカウンタだった。VGA\_DISPLAY\_CONTROLLERのアドレスカウンタのバグを修正後、CMOSカメラからの画像を正常にディスプレイに表示することができた。ディスプレイに表示された画像を図9に示す。



図 9: 正常なディスプレイの画像

## 8 まとめ

イーエスピー企画の画像ベースボード、デジタルCMOSカメラ、208ピンSpartan3E XC3S500+2M高速SRAM基板を使用して、CMOSカメラの輝度データをディスプレイに表示する回路をVHDLで作成した。その回路の表示画素数は640×480ドットで、60フレーム/秒のVGA信号を出力して、CMOSカメラの白黒画像をディスプレイに表示することができる。

いろいろなバグがあったがバグを修正して、CMOSカメラの輝度データをディスプレイに正常に表示することができた。これからは、独自の画像変換回路をFPGA上に構成してディスプレイにその画像変換結果を表示する予定である。

## 参考文献

- [1] OV7640 Color CMOS VGA (640 x 480) CAMERACHIP™ データシート  
(<http://www.alldatasheet.jp/datasheet-pdf/pdf/121707/ETC/OV7640.html>)
- [2] Spartan-3 ジェネレーション FPGA ユーザーガイド  
([http://japan.xilinx.com/support/documentation/user\\_guides/j\\_ug331.pdf](http://japan.xilinx.com/support/documentation/user_guides/j_ug331.pdf))
- [3] 江崎 雅康、” 7月号付属 Spartan-3E ボードで始める画像回路入門”、Design Wave Magazine 2007年8月号、p20 - p28
- [4] 江崎 雅康、” VGA デジタル CMOS カメラ・モジュールからの入力回路を作ろう”、Design Wave Magazine 2007年8月号、p51 - p67
- [5] 江崎 雅康、” 画像フレーム・メモリと FPGA を使った画像プラットフォーム”、Design Wave Magazine 2007年10月号、p66 - p72
- [6] 江崎 雅康、” フレーム・メモリを備えた画像処理回路の設計”、Design Wave Magazine 2007年10月号、p85 - p96
- [7] FPGA の部屋まとめサイト、画像処理、([http://marsee101.web.fc2.com/image\\_processing.html](http://marsee101.web.fc2.com/image_processing.html))