

FRマイコン基板を使った デジタル・フォト・フレームの 設計事例

ハードウェア編



関連データ

飯島 幸太

本誌2008年5月号付録FRマイコン基板を使用しデジタル・フォト・フレームを製作した。480×272ピクセル表示が可能なフルカラーTFT液晶の制御を、FRマイコンだけで行っている。また、汎用入出力ポートを利用し、画像表示している。今回はハードウェアを、次回はソフトウェアを解説する。
(編集部)

本誌2008年5月号付録のFR60基板を用いた応用例として、24ビット・フルカラー・デジタル・フォト・フレーム (Digital Photo Frame. 以降DPF) を製作しました (写真1)。携帯ゲーム機にも採用されている横480×縦272ピクセルの表示が可能な、フルカラーTFT (Thin Film Transistor) 液晶を使用しています。本来ならば、液晶ドライブ用のICやFPGA (Field Programmable Gate Array) を利用してドライブするのですが、液晶の制御方法の勉強も兼ねて、マイコンのみで制御することにしました。FR60には、OSDC (On Screen Display Control) 機能がありますが、表現力は512色です。簡単なアニメーションやメールなどの表示には十分ですが、フルカラー画像は表示させられません。そのため、汎用入出力ポート (GPIO: General Purpose Input/Output) を利用して表示させました。

システム構成を図1に示します。DPF本体と、画像を書き換え可能なWindowsアプリケーションの二つで構成されています。DPFのフラッシュROMはフルカラー・データを2画像分保持でき、書き換えをWindowsアプリ

ケーションで行います。もちろんFR60の特徴であるUSBターゲットの通信機能を利用し、画像データをDPFに転送しています。家族・恋人・ペットなど、お気に入りの画像で癒されてください (笑)。

全ソース・実行ファイル一式は、本誌のWebサイト (<http://www.cqpub.co.jp/interface/>) からダウンロード可能です。また、ダウンロード・サイトでシステムの動画も公開するので、興味のある方は液晶表示を確認してみてください。

1. システムの概要

表1に、デジタル・フォト・フレームの仕様を示します。内蔵フラッシュROM (以降フラッシュ) へ書き込まれた24ビット・フルカラー画像を液晶へ転送^{注1}し、画像を表示するのが一番の仕事です。保存された2画像 (A面・B面) の表示切り替えは、物理的な手動スイッチで切り替

注1: 液晶ドライバを利用していないので、表示し続けるためには、転送し続ける必要がある。



写真1 製作した Digital Photo Frame

アクリル・フレームを利用し、見た目も意識した。写真内の映像はイラストだが、24ビット・フルカラー表示可能なので、デジタル・カメラで撮影したお気に入り画像の表示も可能。

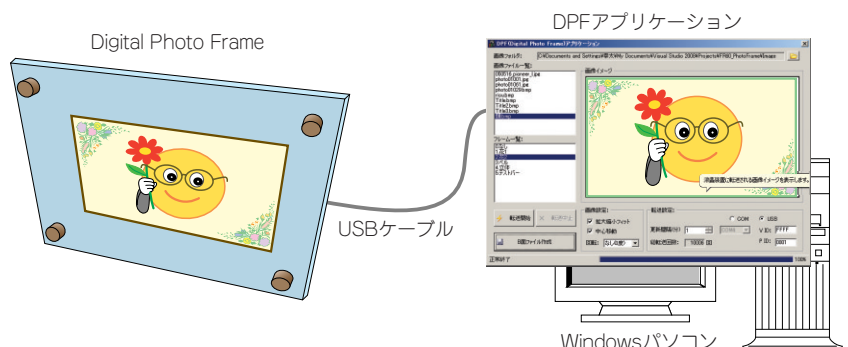


図1 システム構成図

Digital Photo Frameは、WindowsパソコンとのUSB接続を想定して開発した。専用アプリケーションを介し、仮想COMポートもしくはUSB経由での画像の書き換えが可能。もちろん、DPFアプリケーションなしでもDPF本体の動作が可能。

えます。A面は、フラッシュの書き換え手順が公開されているので、USBのバルク通信、もしくはUART通信機能を利用して書き換えます。

表2に、DPFアプリケーション(以降DPFアプリ)の仕様を示します。汎用USBドライバを経由して書き換え指示を行います。お気に入りの画像が格納されたフォルダと時間を指定しておけば、スライドショーで表示させることもできます。まるでRAMのようにフラッシュROMを頻繁に書き換えることとなりますが、付属FRマイコンのフラッシュは、1万回程度書き換え(実際には消去)が行えるので、このような使い方も可能です。1日8時間の利用で30分に1回書き換えと仮定すると、約20ヵ月利用できます(コラム1参照)。

図2に、フラッシュ配置イメージを示します。マイコンには、合計1Mバイトのフラッシュが搭載されており、二

注2：フラッシュ1は書き換え方法が公開されているので変更できるが、フラッシュ2は公開されていないので、提供ツールによる書き換えになる。

つの領域(フラッシュ1とフラッシュ2)に分かれています。画像サイズが480×272×24ビット(RGB各8ビット)=382.5Kバイトであることから、それらに別画像を格納する十分な領域があります。フラッシュ1(A面)は、DPFアプリからの指示で書き換えを行い、フラッシュ2(B面)は、フラッシュROMプログラマで書き換えます^{注2}。DPFアプリに、フラッシュROMプログラマで書き込むことができるB面用のファイル生成機能を実装したので、間接的に書き換えが可能です。

2. ハードウェアの設計

●(割と)簡単な回路で表示できる？

図3は、システム・ブロック図です。『付属FR60基板』、『DC-DCコンバータ』、『TFT液晶』が主な部品です。実は、それほど多くの部品を必要とせずに表示できました。端子割り当てを表3に、回路図を図4に示します。

表1
Digital Photo Frameの仕様

項目	内容	備考
電源	USBバス・パワー駆動	5V単一電源
表示領域サイズ	横480ピクセル×縦272ピクセル	-
保存可能画像枚数	2枚(スイッチによるA/B面切り替え)	A面は直接書き換え可能
通信方式	USB：12Mbps / COMポート：57,600bps	-
リフレッシュ・レート	約6フレーム / 秒	-

表2
DPFアプリケーションの仕様

項目	内容	備考
開発環境	Visual Studio 2008 Express Edition	VC++
動作環境	.Net Framework 2.0	Windows XPを推奨
転送可能ファイル形式	JPEG, BMP	-
画像転送	フラッシュ1へ画像転送	USBで約10秒(消去含む)
画像ファイル作成	B面用モトローラSファイル生成	フラッシュROMプログラマ専用
ライブラリ	汎用USBドライバとDLL	Uusbdl.sys, Uusbdl.dll

コラム1 なぜフラッシュROMを利用するのか？

画像表示目的には、本来なら高速に読み書きできるRAMを利用するべきです。内蔵RAMは16Kバイトと、VRAMとしては容量が足りません。次に、外部RAMの追加を検討しましたが、MB91310シリーズは外部バスがMPU内部で閉じているため追加できませんでした。

しかし、フラッシュROMの容量が1Mバイト(512Kバイ

ト×2領域)と、1チップ・マイコンとしては比較的大容量だったこと、また書き換え用の手順が公開されていたことから、今回の仕様に落ち着きました。もちろん、書き換え可能回数が1万回という、比較的回数の多い製品だったのも理由の一つです。