

はじめに

本書では、アルテラの MAX® II CPLD を使用した LCD コントローラの実装について説明します。ディスプレイの例では、一般的に使用されている SC1602D デバイスなどの Optrex 16×2 ドット・マトリックス LCD モジュールを使用します。このコントローラは、マイクロプロセッサまたはマイクロ・コントローラからディスプレイ信号を受信し、その信号を LCD モジュールによって解釈されるコマンドに変換します。MAX II デバイスは、業界唯一のユーザ・フラッシュ・メモリ (UFM) を搭載しており、コントロールはそのメモリを使用してパワー・オン時にメッセージを表示します。そのため、外部メモリを使用する必要はありません。また、このコントローラはクロックに CPLD の内部 UFM オシレータを利用するため、外部クロック信号も必要ありません。これらの独自の機能を備えているため、MAX II CPLD はこのようなコントローラ用として最適なターゲット・デバイスとなっています。

LCD

液晶ディスプレイ (LCD) は薄くて平らなディスプレイ・デバイスで、光源やリフレクタの前面に無数のカラー・ピクセルまたはモノクロ・ピクセルが配列されています。これらのディスプレイは電子機器の多くで不可欠なものになっています。LCD が普及する要因となった特徴は、以下のとおりです。

- LCD により、デバッグ機能を備えた便利なインタフェース付きのアプリケーションやプロフェッショナルな外観を持つアプリケーションを実現できます。
- LCD は非常に消費電力が低く、バッテリー駆動装置に最適です。
- LCD は薄型軽量で、他のタイプのディスプレイと比較して実装面積が小さく済みます。

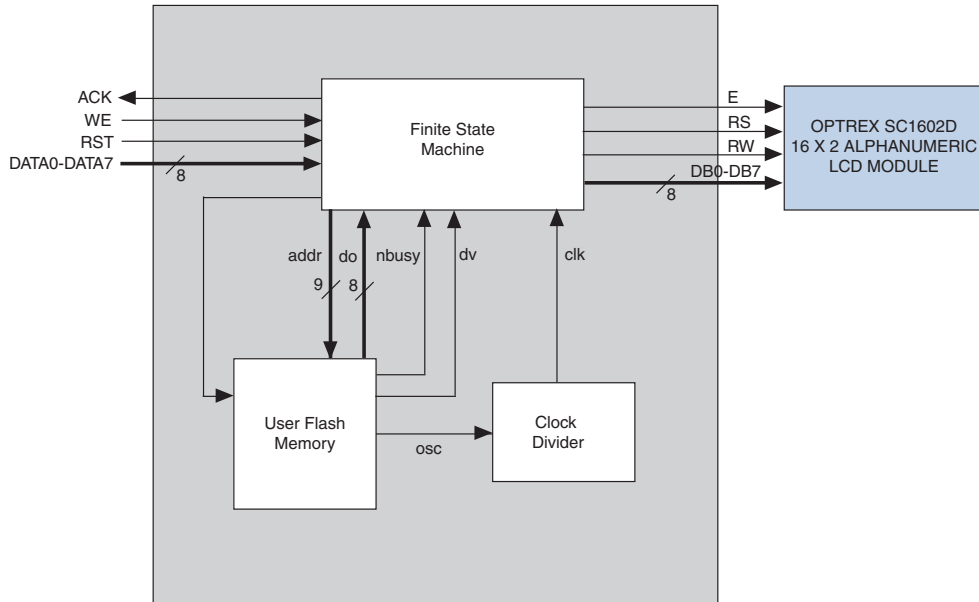
ほとんどの LCD モジュールが、汎用インタフェース、オン・ボード・コントローラ、およびオン・ボード・ドライバを備えています。したがって、LCD を使用してアプリケーションの見栄えを良くすることが簡単になりました。LCD ディスプレイを使用する装置には、腕時計、電卓、ラップトップ、PDA、携帯電話、家電製品、玩具などがあります。

LCD コントローラ

このコントローラ・デザインは、どのアプリケーションにでも容易に統合し、最適なディスプレイ・ユニットを作ることができます。このコントローラは、必要なタイミング信号を生成して LCD のタイミング制約を満たすと同時に、マイクロプロセッサやコントローラを LCD の細かな制御から解放することによって、プロセッサやコントローラが他の処理を効率的に実行できるようにします。

図 1 に、LCD コントローラのブロック図を示します。LCD コントローラ内の主要な 3 つのモジュールは、有限ステート・マシン (FSM)、ユーザ・フラッシュ・メモリ (UFM)、およびクロック・ディバイダ・モジュールです。LCD モジュールへのインタフェース信号は、E、RS、RW、および DB0-DB7 です。このデザインでは、LCD コントローラを通じてプロセッサと LCD モジュールを正しくインタフェースさせています。これらの信号について詳しくは、表 1 を参照してください。

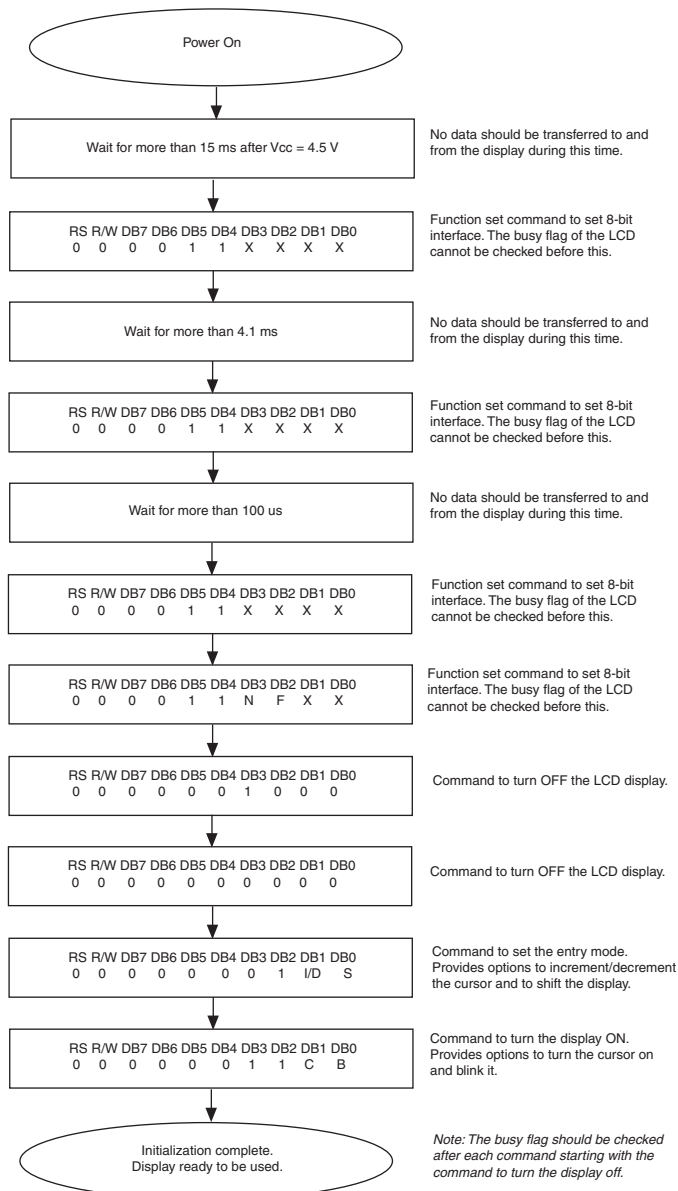
図 1. LCD コントローラ



LCD モジュールと有限ステート・マシンの初期化

FSM は 8 種類のステートで構成されます。FSM は、LCD の初期化、初期化完了後のメッセージの表示、および LCD の書き込みと読み出しに使用されます。図 2 に、LCD モジュールの初期化ステップを示します。

図 2. LCD モジュールの初期化フロー・チャート



初期化を簡素化するために、各コマンドの送信後に（ビジー・フラグのステータスをチェックする代わりに）15 ms の遅延が設けてあります。ただし、LCD モジュールへのデータの書き込み中は、各データが書き込まれるたびに、ビジー・フラグのステータスがチェックされます。これにより、書き込み処理が速くなります。



LCD モジュールの初期化および LCD モジュールでサポートされる各種コマンドについて詳しくは、以下のリンクにある Optrex LCD のマニュアルを参照してください。

www.apollodisplays.com/pdf/dmcmman.pdf

ユーザ・フラッシュ・メモリ

MAX II CPLD 内部の UFM は、コントローラの電源がオンになった直後に表示されるキャラクタの ASCII 値を格納するのに使用されます。これは UFM メガファンクションを使用してインスタンス化されます。UFM は、16 進（Intel Format）ファイル（.hex）を利用して、パラレル・インタフェースを経由してプログラムされます。Quartus® II ソフトウェアによる .hex ファイルの作成について詳しくは、以下のリンクを参照してください。

www.altera.co.jp/support/software/nativelink/quartus2/glossary/def_rif.html



MegaWizard® Plug-In Manager による UFM のインスタンス化について詳しくは、「MAX II デバイス・ハンドブック」の「MAX II デバイスでのユーザ・フラッシュ・メモリの使用」の章を参照してください。

クロック・ディバイダ

LCD モジュールのタイミング要件が満足され、コントローラの所要性能が達成するために、クロック・ディバイダ・モジュールによって内部オシレータ出力の周波数を低くします。表 1 に、図 1 に示す LCD コントローラ（MAX II CPLD）ブロック図の信号の説明を示します。

表 1. LCD コントローラ (MAX II CPLD) の信号の説明 (1 / 2)	
信号	説明
DB0-DB7	LCD モジュールとの間でデータとコマンドを転送する 8 ビットの双方向バス。
RW	読み出し動作と書き込み動作を区別する、リード信号とライト信号。 <ul style="list-style-type: none"> ● RW = 0: ライト動作 ● RW = 1: リード動作

表 1. LCD コントローラ (MAX II CPLD) の信号の説明 (2 / 2)

信号	説明
RS	レジスタ選択信号。LCD モジュールのデータ・レジスタとコマンド・レジスタの選択に使用されます。 <ul style="list-style-type: none"> ● RS = 0: コマンド・レジスタ ● RS = 1: データ・レジスタ
E	ストロープ信号をイネーブルします。この信号のネガティブ・エッジは、双方向バス上のデータをデータ・レジスタまたはコマンド・レジスタに、あるいはその逆にラッチします。
DATA0-DATA7	プロセッサおよびコントローラからの 8 ビット・バスで、LCD モジュールに表示するキャラクタの ASCII 値を転送します。
RST	リセット信号。この信号はコントローラをリセットして、LCD モジュールを再初期化するのに使用されます。
WE	ライト・イネーブル信号。この信号はライト・サイクルの開始を示します。LCD モジュールへの書き込みをイネーブルするには、 low に設定する必要があります。
ACK	確認応答信号。この信号は、LCD モジュールにデータが正しく送信され、次のデータを受信する準備ができたことを示します。コントローラは、この信号が Low になる前に送信されたデータはすべて無視します。
addr	UFM の特定のメモリ・ロケーションを選択するのに使用される 9 ビット・アドレス・バス。
nread	アドレス・バスでポイントされるメモリ・ロケーションの内容を読み出すには、この信号を 0 に設定します。
do	データ出力信号。アドレス・バスでポイントされる 16 ビット・メモリ・ロケーションの上位 8 ビットを保持する 8 ビット・データ・バス。
dv	データ有効信号。8 ビット・データ・バス上のデータが有効で、読み出し可能であることを示す信号。
osc	UFM ブロックの内部オシレータの出力。5.5 MHz の信号周波数を出力します。
clk	クロック・ディバイダ・ブロックの出力。オシレータ出力信号の周波数を約 43 kHz に低減します。

実装

このデザインは、EPM240G またはその他の MAX II CPLD と、Optrex SC1602D 16×2 英数字 LCD モジュールを使用して実装できます。このデザインのソース・コードを使用して、MAX II CPLD の汎用 I/O (GPIO) ラインにデータ・バスとコントロール・ライン (入力および出力) を割り当てます。内部ユーザ・フラッシュ・メモリを使用して、パワー・オン時に LCD モジュールに表示するデフォルト・メッセージを格納します。Quartus II ソフトウェアのメモリ・エディタを使用して、LCD モジュールにデフォルト・メッセージを格納するためのメモリ初期化ファイルを作成します。

以下では、MDN-B2 デモ・ボードでのこのデザインの実装について詳しく説明します。表 2 に、このデザイン例での EPM240G のピン・アサインメントを示します。

表 2. MDN-B2 デモ・ボードを使用したデザインの実装			
EPM240G のピン・アサインメント			
信号	ピン	信号	ピン
DB[0]	ピン 53	DB[4]	ピン 1
DB[1]	ピン 98	DB[5]	ピン 96
DB[2]	ピン 52	DB[6]	ピン 92
DB[3]	ピン 97	DB[7]	ピン 95
data[0]	ピン 55	data[4]	ピン 61
data[1]	ピン 56	data[5]	ピン 66
data[2]	ピン 57	data[6]	ピン 67
data[3]	ピン 58	data[7]	ピン 68
E	ピン 99	rst	ピン 77
RS	ピン 100	ack	ピン 69
RW	ピン 54	we	ピン 81



未使用ピンは、コンパイル前に Quartus II ソフトウェアのデバイスおよびピン・オプション設定で **As input tri-stated** に割り当ててあります。

デザイン・ノート

MDN-B2デモ・ボードで本デザインのデモを行うには、以下のステップを実行します。

1. JTAG ヘッダ JP5 と従来のケーブル (ByteBlaster™ II、USB-Blaster™) を使用して、デザインを MAX II CPLD ヘダウンロードします。プログラミングを行うには、デモ・ボードのスイッチをオンにする必要があります (スライド・スイッチ SW1)。プログラミングが終了したら、電源をオフにします。
2. メモリ・ファイル **lcd_new.hex** (ソース・コード・ファイルに付属) が Quartus II ソフトウェアのプロジェクト・ライブラリに置かれていることを確認します。
3. 付属のインタコネクタ・ケーブルを使用して、LCD モジュールをボードに接続します。LCD モジュールの CN1 にケーブルの 14 ピン・ソケットを接続し (赤マークがデモ・ボードの CN1 のピン 1 と一致)、デモ・ボードの JP8 に 20 ピン・ソケットを接続します (ソケットの赤マークがデモ・ボードの JP8 のピン 1 と一致)。
4. Optrex SC1602D LCD モジュール (このモジュールもアルテラの NIOS® II 開発キット 2C35N [Cyclone II バージョン] に付属) を使用します。
5. インタコネクタ・ケーブルの DC ソケットを使用して、LCD モジュールに必要な 5 V 電源を供給します。この DC ソケットは標準的な極性が配置されています (内部ピンが正)。3xAA または AAA バッテリーを使用して、LCD モジュールに電源を供給することもできます。
6. まず、LCD モジュールに 5 V 電源を接続します。次に、ステップ 4 で説明したとおりインタコネクタ・ケーブルを接続します。
7. スライド・スイッチ SW1 を使用して、MDN-B2 デモ・ボードの電源をオンにします。
8. LCD モジュールのデフォルト・メッセージを確認します。このメッセージは、**lcd_new.hex** ファイルを使用して、MAX II CPLD の UFM に事前にプログラムされています。LCD コントローラは、このメッセージをパワー・オン時に 1 回だけ読み出します。
9. デモ・ボードのリセット・プッシュ・ボタン SW6 を使用して、LCD モジュールをリセットします。リセット後に、LCD 上のカーソルが 1 行目の第 1 キャラクタの位置で点滅していることを確認します。
10. この演習では、SW5 スwitch (デモ・ボード上の 8 方向デュアル・イン・ライン・パッケージ (DIP) GPIO 入力スイッチ) を使用して、表示するキャラクタに対応する任意の ASCII データを入力できます。このスイッチを ON にすると **logic 0**、OFF にすると **logic 1** が設定されます。スイッチ #1 はこの ASCII データ・バイトの LSB です。

11. デモ・ボードのボタン SW8 を押して、コントローラをライト・イネーブルにします。これにより、コントローラは SW5 スイッチで設定したフレッシュ・データを取り込むことができます。デモ・ボードは SW8 ボタンが押されている間、約 4.25 kHz のレートで新しいデータを取り込みます。SW8 ボタンを押すと、直後に両行のすべてのキャラクタが更新されます。



このデザインは、Optrex 製 SC1602D LCD モジュール用に作成されたものです。この LCD モジュールでは、LCD の 1 行目の最後のキャラクタと、2 行目の先頭のキャラクタのアドレスは不連続です。このデザインには、この配列に合わせて変更が加えてあります。

ソース・コード

このデザインは Verilog を使用して作成しており、MDN-B2 デモ・ボードを使用したデモとなっています。ソース・コード、テストベンチ、および完成した Quartus II プロジェクトは、以下から入手可能です。

www.altera.co.jp/literature/an/an497_design_example.zip

まとめ

MAX II CPLD は LCD コントローラを実装するための優れた選択肢です。MAX II CPLD は、低消費電力の使いやすいパワー・オン機能やビルトイン・ユーザ・フラッシュ・メモリを備えており、LCD コントローラ・デザインを実装するための理想的なプログラマブル・ロジック・デバイスです。

参考資料

このアプリケーション・ノートでは、以下のドキュメントを参照しています。

- www.apollodisplays.com/pdf/dmcmn.pdf
- www.altera.co.jp/support/software/nativelink/quartus2/glossary/def_rif.html
- 「MAX II デバイス・ハンドブック」の「MAX II デバイスにおけるユーザ・フラッシュ・メモリの使用」の章

関連情報

以下に、このアプリケーション・ノートの関連情報を示します。

- MAX II CPLD ホームページ：
www.altera.co.jp/products/devices/cpld/max2/mx2-index.jsp
- MAX II デバイスの資料ページ：
www.altera.co.jp/literature/lit-max2.jsp
- MAX II パワーダウン・デザイン：
www.altera.co.jp/support/examples/max/exm-power-down.html

- MAX II アプリケーション・ノート :
 - 「AN 428: MAX II CPLD のデザイン・ガイドライン」
 - 「AN 422: MAX II CPLD を使用したポータブル・システムにおける消費電力の管理」

改訂履歴

表 3 に、このアプリケーション・ノートの改訂履歴を示します。

日付 & ドキュメント・バージョン	変更内容	概要
2007 年 12 月 v1.0	初版	—



101 Innovation Drive
San Jose, CA 95134
www.altera.com
Literature Services:
literature@altera.com

Copyright © 2007 Altera Corporation. All rights reserved. Altera, The Programmable Solutions Company, the stylized Altera logo, specific device designations, and all other words and logos that are identified as trademarks and/or service marks are, unless noted otherwise, the trademarks and service marks of Altera Corporation in the U.S. and other countries. All other product or service names are the property of their respective holders. Altera products are protected under numerous U.S. and foreign patents and pending applications, maskwork rights, and copyrights. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

