

## HD44780 採用の文字 LCD と ADuC702x とのインターフェース MicroConverter ADuC702x 開発システム

### はじめに

今日、多くの LCD ディスプレイが存在しますが、幸いにも、これらの多くの LCD は HD44780U 標準に準拠しています。この標準は LCD コントローラ・チップと呼ばれ、ADuC702x からのデータを入力し、LCD スクリーンと交信します。HD44780 標準の LCD スクリーンは多くのフォーマットで使用可能であり、最も一般的なものは 16×2 フォーマットと 20×2 フォーマットです。このアプリケーション・ノートでは、LCD の基本機能を制御するコマンドについて説明します。

### HD44780 LCD とのインターフェース

HD44780 と MicroConverter® を接続するデータ・バスは 8 ビット幅または 4 ビット幅が可能です。このアプリケーション・ノートでは、8 ビット・データ・バスについて説明します。データ・バスの他に、3 本のコントロール・ラインが必要です。したがって、LCD と MicroConverter とのインターフェースには合計 11 本のピンが必要です。

データ・バスを構成する 8 本のデータ・ラインを、DB0~DB7 と呼びます。

3 本のコントロール・ラインを EN、RS、 $\overline{R/W}$  と呼びます。これらの機能を次に説明します。

#### イネーブル・ライン(EN)

このラインは、LCD コントローラに対するデータ・バイト送信の開始を指示します。送信の開始を表示するときは、このラインをハイレベルにし、送信が終わったときは、EN ラインをローレベルにして、送信が完了したことを表示します。

#### レジスタ・セレクト・ライン(RS)

このラインは、データ・バイトをコマンドとして、またはテキスト・データとしてスクリーン上に表示することを LCD コントローラに指示します。RS ラインがハイ・レベルのときは、データ・バイトを表示するテキストとして扱います。RS ラインがロー・レベルのときは、データ・バイトをコマンドとして扱います。

#### リード/ライト・ライン( $\overline{R/W}$ )

このラインがローレベルのときは、データ・バス上の情報が LCD コントローラへ書き込まれます。このラインがハイレベルのときは、LCD コントローラから読み出すことができます。このラインを使って、LCD のステータスをチェックします。

図 1.HD44780 LCD と ADuC702x の接続に示すように、8 本のデータ・ラインが MicroConverter のポート 1 に接続され、3 本のコントロール・ラインがポート 0.5、ポート 0.7、ポート 2.0 に接続されます。

このインターフェースの定義に使うソース・コードを次に示します。

```
unsigned char Init_MC08_LCD()
{
    GP0DAT = 0xA0000000; // P0.5 = RS, EN = P0.7
    GP1DAT = 0xFF000000; // P1.[7:0] = DB[0:7]
    GP2DAT = 0x01000000; // P2.0 =  $\overline{R/W}$ 

    return 0x1;
}
```

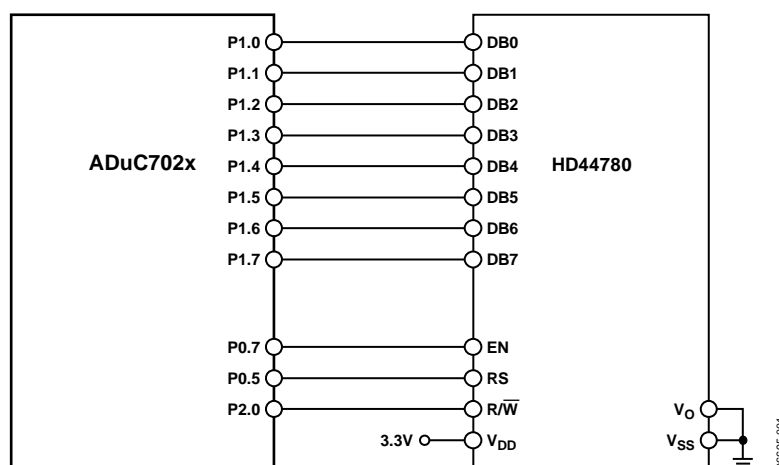


図 1.HD44780 LCD と ADuC702x の接続

Rev. 0

## 目次

はじめに.....	1	入力モードの設定.....	5
HD44780 LCD とのインターフェース.....	1	LCD スクリーンへのテキストの書込み.....	6
LCD スクリーンの設定.....	3	注意事項およびリファレンス電圧.....	7
LCD スクリーンのクリア.....	4		

## LCD スクリーンの設定

LCD スクリーンにテキストを表示するときは、まず LCD スクリーンを設定する必要があります。設定では、LCD コントローラに対して、使用する LCD スクリーンのタイプ、データ・バス・フォーマット、フォントを定義します。使用可能な種々のコマンドの一覧は、HD44780U を採用している LCD モジュールのデータシートに記載してあります。

次に示す例では、LCD を 8 ビット・データ・バスの使用と 5×7 ドットの文字フォントでの表示に設定しています。この設定は、`Select_Function_Set` 関数を使って LCD コントローラへ 0x38 を送信することにより実現されます。

```
unsigned char Select_Function_Set(unsigned char ucFunctionSet)
{
    unsigned long ulFunction = 0;

    delay (10000);
    RdStatus = ReadStatus();
    ulFunction = ucFunctionSet;
    ulFunction = (ulFunction << 16);
    ulFunction |= 0xFF000000;

    GP0DAT = 0xA0000000;           //Clear RS pin(P0.5), clear E pin (P0.7) = R/ $\bar{W}$ )
    GP2DAT = 0x01000000;           //Clear R/ $\bar{W}$  pin (P2.0 = R/ $\bar{W}$ )
    delay (5);

    GP0SET = 0x800000;             // Set E high
    delay (5);                     // Allow min 800 ns setup time
    GP1DAT = ulFunction;           // Write to register
    delay (5);                     // allow hold time of 500 ns min
    GP0CLR = 0x800000;             // Set E low
    delay (5);

    return 0x1;
}
```

`ReadStatus` 関数は、HD44780 モジュールからのビジー信号出力の読み出しに使われます。

## LCD スクリーンのクリア

スクリーンに書き込む前に、まずスクリーンをクリアする必要があります。これは、`Clear_Display_LCD` 関数を使って次のように行います。

```
unsigned char Clear_Display_LCD(unsigned char ucFunctionSet)
{
    unsigned long ulFunction = 0;

    delay (10000);
    RdStatus = ReadStatus();
    ulFunction = ucFunctionSet;
    ulFunction = (ulFunction << 16);
    ulFunction |= 0xFF000000;

    GP0DAT = 0xA0000000;           //Clear RS pin(P0.5), clear E pin (P0.7) = R/ $\bar{W}$ )
    GP2DAT = 0x01000000;           //Clear R/ $\bar{W}$  pin (P2.0 = R/ $\bar{W}$ )
    delay (5);

    GP0SET = 0x800000;             // Set E high
    delay (5);                     // Allow min 800 ns setup time
    GP1DAT = ulFunction;           // Write to register
    delay (5);                     // Allow hold time of 500 ns min
    GP0CLR = 0x800000;            // Set E low
    delay (2000);                  // Minimum clear time of 1.58 ms

    return 0x1;
}
```

## 入力モードの設定

入力モードも設定する必要があります。この例では、インクリメントをオンにし、シフトなしにしています。この設定は、Set\_Entry\_Mode関数を使ってLCDコントローラへ0x6を送信することにより実現されます。

```
unsigned char Set_Entry_Mode(unsigned char ucFunctionSet)
{
    unsigned long ulFunction = 0;

    delay (10000);
    RdStatus = ReadStatus();
    ulFunction = ucFunctionSet;
    ulFunction = (ulFunction << 16);
    ulFunction |= 0xFF000000;

    GP0DAT = 0xA0000000;           //Clear RS pin(P0.5), clear E pin (P0.7) = R/ $\bar{W}$ )
    GP2DAT = 0x01000000;           //Clear R/ $\bar{W}$  pin (P2.0 = R/ $\bar{W}$ )
    delay (5);

    GP0SET = 0x800000;             // Set E high
    delay (5);                     // Allow min 800 ns setup time
    GP1DAT = ulFunction;           // Write to register
    delay (5);                     // Allow hold time of 500 ns min
    GP0CLR = 0x800000;            // Set E low
    delay (5);

    return 0x1;
}
```

## LCD スクリーンへのテキストの書込み

サンプル・プログラムでは、次のテキストを LCD スクリーンへ書込みます。

Analog Devices

ADuC7020 LCDdemo

これは、スクリーンをクリアした後に、Wr\_Data\_LCD 関数を使って実行します。

たとえば、文字 A を出力するときは、16 進表示の 0x41 を使います。他のすべての文字も 16 進表示を使って、次のように実行される関数へ渡す必要があります。

```
unsigned char Wr_Data_LCD(unsigned char ucFunctionSet)
{
    unsigned long ulFunction = 0;

    delay (10000);
    RdStatus = ReadStatus();
    ulFunction = ucFunctionSet;
    ulFunction = (ulFunction << 16);
    ulFunction |= 0xFF000000;

    GP0DAT = 0xA0200000;           //Set RS pin(P0.5), clear E pin (P0.7) = R/ $\bar{W}$ )
    GP2DAT = 0x01000000;           //Clear R/ $\bar{W}$  pin (P2.0 = R/ $\bar{W}$ )
    delay (5);

    GP0SET = 0x800000;             // Set E high
    delay (5);                     // Allow min 800 ns setup time
    GP1DAT = ulFunction;           // Write to register
    delay (5);                     // Allow hold time of 500 ns min
    GP0CLR = 0x800000;             // Set E low
    delay (5);
    GP0CLR = 0x200000;             // Set RS low

    return 0x1;
}
```

## 注意事項およびリファレンス電圧

このアプリケーション・ノートで説明するすべてのプロセスの実装で使用したドット・マトリクス LCD スクリーンは、Samsung 社の S6A0070 ドライバとコントローラです。これらの製品の詳細については、Samsung 社のウェブサイトをご覧ください。

ADuC702x ファミリー内の特定のアナログ・デバイセズ製品については、表 1 に示す製品リストから選択してください。

表 1. アナログ・デバイセズの ADuC702x シリーズ

Part No.	No. GPIO Pins	No. ADC Channels	No. 12-Bit DAC Outputs	Temp. Range (°C)
<a href="#">ADuC7020</a>	14	5	4	-40 to +105
<a href="#">ADuC7021</a>	13	8	2	-40 to +85
<a href="#">ADuC7022</a>	13	10	N/A	-40 to +85
<a href="#">ADuC7024</a>	30	10	2	-40 to +105
<a href="#">ADuC7025</a>	30	12	N/A	-40 to +105
<a href="#">ADuC7026</a>	40	12	4	-40 to +125
<a href="#">ADuC7027</a>	40	16	N/A	-40 to +125
<a href="#">ADuC7028</a>	40	16	4	-40 to +125

AN06695-0-6/07(0)-J