

NXP LPC1114FN28/102 (ARM Cortex-M0搭載 DIP28)

LPC1114FN28 /102 仕様

- ARM Cortex-M0搭載 低消費・高性能32-bitマイコン
- 50MHz クロックスピード (最大)
- Flash : 32KB
- RAM : 4KB
- RCオシレータ: 12MHz, 1%精度
- 外付けクリスタルも可
- UART (RS-485)
- SSP/SPI
- I2C (Fm+)
- ADC x 6ch
- タイマ : 32-bit x 2 / 16-bit x 2 (PWMモードあり)
systickタイマ
- ウォッチドッグタイマ
- GPIO : 22本
- パッケージ : DIP28

PIO0_8/MISO0/CT16B0_MAT0	1	28	PIO0_7/CTS
PIO0_9/MOSI0/CT16B0_MAT1	2	27	PIO0_4/SCL
SWCLK/PIO0_10/SCK0/CT16B0_MAT2	3	26	PIO0_3
R/PIO0_11/AD0/CT32B0_MAT3	4	25	PIO0_2/SSEL0/CT16B0_CAP0
PIO0_5/SDA	5	24	PIO0_1/CLKOUT/CT32B0_MAT2
PIO0_6/SCK0	6	23	RESET/PIO0_0
VDDA	7	22	VSS
VSSA	8	21	VDD
R/PIO1_0/AD1/CT32B1_CAP0	9	20	XTALIN
R/PIO1_1/AD2/CT32B1_MAT0	10	19	XTALOUT
R/PIO1_2/AD3/CT32B1_MAT1	11	18	PIO1_9/CT16B1_MAT0
SWDIO/PIO1_3/AD4/CT32B1_MAT2	12	17	PIO1_8/CT16B1_CAP0
PIO1_4/AD5/CT32B1_MAT3/WAKEUP	13	16	PIO1_7/TXD/CT32B0_MAT1
PIO1_5/RTS/CT32B0_CAP0	14	15	PIO1_6/RXD/CT32B0_MAT0

マニュアル, 技術情報 及び サンプルコード

LPCZone日本語サイト: www.nxp-lpc.com の「LPC1100」のページに行き、下の方にある「ドキュメント/技術情報」の中にあります。今後、www.nxp-lpc.com内のLPC1100ページのことを「LPC1100」ページと呼びます。最初に必要ものは下記。

- LPC111x(X)Lデータシート: 電気特性があります。
- LPC111x /LPC11Cx ユーザーマニュアル: マイコンの使い方、レジスタの設定方法
最新版は英語のものになります。日本語のものにはLPC1114FN28は載っていませんが、レジスタの設定方法等は同じです、英語版の参照に使えます。
- その他アプリケーションノート、サンプルコードもここから取れます。



開発環境 — LPCXpresso (Windows/Mac OS X/Linux版)

LPCXpresso IDE と LPC-LINK(デバグ)を使用する場合

- LPCXpresso IDEのインストールとアクティベーション、サンプルコードのビルドと実行 について書かれている http://akizuki-denshi.com/download/ds/nxp/LPCXpresso_manual.pdf をダウンロードして参照ください。
- 参照回路図 / LPCXpresso: 「LPC1100」にあります。LPCXpressoのLPC-Linkデバグを使ってデバッグ/プログラムの書き込みを行い場合、配線のために必要になります。
LPCXpressoは何種類ありますが、LPC-Linkは同じです。
- サンプルコード: 「LPC1100」ページ内にあります。「サンプルコード/ LPCXpresso LPC111x, LPC11Cxxx」



その他、IAR EW-ARM, Keil MDK-ARM などの開発環境もあります。

プログラム書き込みツール「FlashMagic」(Windows/Mac OS X版)

LPC-Linkデバグを使用する場合は使いません。

デバグを使わずにPCのCOMポートからLPC1114FN28のUART経由でHexファイルを書き込むことができるツールです。

- FlashMagicのダウンロード先: www.nxp-lpc.comの開発環境/ソリューション ページに「FlashMagic」のリンクがありますので、そこからFlashMagicのページに行き、ダウンロード。
- FlashMagicは、RS-232Cトランシーバを挟んでTxD/RxD以外にものRS-232CのDTR/RTSをマイコンのRST/ISPに接続が必要(ボタンを付けて代用できます)。詳細は、FlashMagicのHelpにあるマニュアルと<http://www.lpcware.com/content/nxpfile/an10302-using-philips-lpc2000-flash-utility-keil-mcb2100> をご覧ください。

CMSIS と サンプルコード

LPC1114FN28はCortex-M0搭載マイコンですので、ARMが提唱しているCMSIS(Cortex Microcontroller Software Interface Standard)に従います。(CMSISの詳細について知りたい方は、<http://www.arm.com>をご覧ください)。

ですので、ヘッダーファイル(LPC1100.h)、ドライバ、サンプルプログラムも CMSISの書き方になっています。

LPC1100のCMSIS 及び サンプルコードは、「LPC1100」ページ内に「サンプルコード/ LPCXpresso LPC111x, LPC11Cxxx」にあります。

及び、LPCXpressoをインストール時に生成されるサンプル。

LPCXpresso IDEのプロジェクトは、CMSISのバージョン関連しています。例えばCMSIS1.3で生成されたプロジェクトファイルをCMSIS2.0に変更したい場合は、<http://support.code-red-tech.com/CodeRedWiki/cmsis200differences> を参照ください。

LPCXpresso
Now supporting Windows,
Linux, and Mac OS

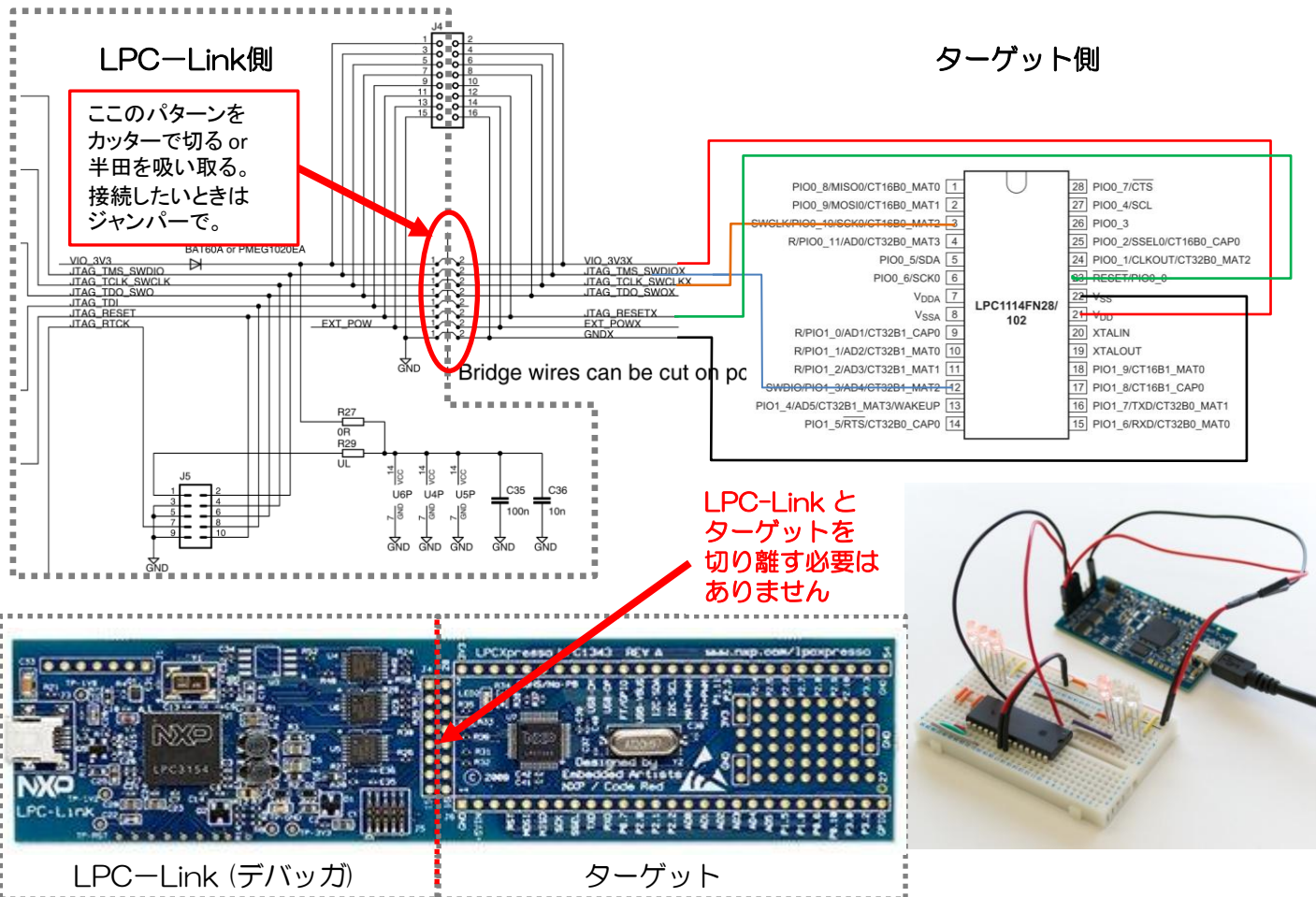


NXP
www.nxp.com

LPCXpressoは 統合開発環境(IDE) と デバグ付きターゲットボードから構成された、低価格 開発ツールプラットフォームです。簡単に開発をスタートできます。LPCXpressoには、開発環境のLPCXpresso IDE と ターゲットボードとデバグが一つになったLPCXpressoボードが含まれます。

- LPCXpresso LPC1769 ・ ・ Cortex-M3 120MHz
Flash : 512KB, RAM : 64KB
Ethernet, USB(Host/OTG/Device) など
- LPCXpresso LPC1347 ・ ・ Cortex-M3 72MHz
Flash : 64KB, RAM : 8KB
USB Device(ROM内にクラスドライバ) など
- LPCXpresso LPC1227 ・ ・ Cortex-M0 30/45MHz
Flash : 128KB, RAM : 8KB
コンパレータx2(555タイマ), DMA, RTC など
- LPCXpresso LPC111U14 ・ ・ Cortex-M0 50MHz
Flash : 32KB, RAM : 6KB
USB Deviceなど
- LPCXpresso LPC11C24 ・ ・ Cortex-M0 50MHz
Flash : 32KB, RAM : 8KB
CANコントローラ+トランシーバなど
(ROM内にCANとCANopenドライバあり)
- LPCXpresso LPC1115 ・ ・ Cortex-M0 50MHz
Flash : 64KB, RAM : 8KB

LPC-Link と LPC1114FN28/102 の接続方法



LPCXpressoでデバッグする時のTips

- LPCXpressoで使用するマイコンを「LPC1114FN/102」に変更する。
LPCXpressoをからLPC-Linkでプログラムをマイコンにダウンロードするとき、LPCXpressoで設定したマイコンの型番とターゲットが同じかどうかをチェックしています。LPCXpresso IDEの上部ツールバーに のマーク(MCUと書いてあります)をクリックし、「LPC1114FN/102」を選択してください。
「LPC1114FN/102」がない場合は、最新のLPCXpressoをインストールしてください。
- Clockソースを外部クリスタルからIRCに変更する(外付けで12MHzのクリスタルを付けている場合は不要)。
リセット後プログラムはmainに行く前に、まずスタートアップファイル(ここではcr_startup_lpc1114.c)内の ResetISR() から始まります。CMSISを使う場合、SystemInit() でクロックの設定をしています。その後、ユーザプログラムのmain()に飛んできます。
LPC1114FN28/102を使う場合、IRCを使うことが多いと思いますので、IRCをクロックソースにする設定方法を書きに書きます。

system_LPC11xx.c ファイル内を修正する。

- 128行目にある `#define SYSPLLCLKSEL_Val` の値を「0」に変更する
`#define SYSPLLCLKSEL_Val = 0x00000001; → 0x00000000;` (PLLのクロックソースを外付けクリスタルからIRCに変更)
- これはやらなくても動作的には問題なし。
`void SystemInit (void) 内`
414行目: `LPC_SYSCON->PDRUNCFG &= ~(1 << 5);` をコメントアウトする (外付けクリスタル用のオシレータを起動させているが、IRCはReset後 defaultで起動するので必要なし、コメントアウトしなくても問題ないが。)

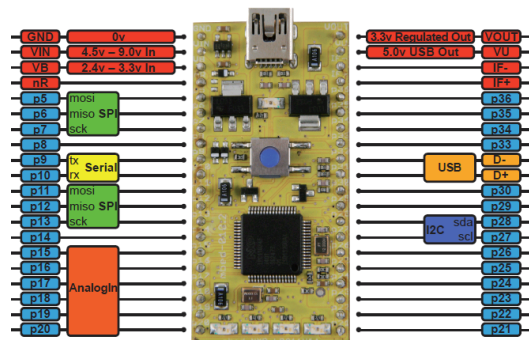
以上を変更することで、LPCXpressoのLPC1114用のサンプルコードが、LPC1114FN28/102のIRCで動くようになります。

LPCXpressoの使い方に関しては、<http://lpc123.blog.fc2.com/> も参照ください。

mbed

mbedはARMが運営しているWebサイト上で開発する高速プロトタイプツールです。開発環境のPCへのインストールは不要です。Webにアクセスできる環境であればどこからでも開発でき、60秒で「Hello World」が可能です。
mbedには直感的な分かりやすいAPIと豊富なサンプルコード、ドライバ、ライブラリがあり、組み込み初心者でも簡単に始めることができます。高度に抽象化されたライブラリ群を用いることで、下層レイヤに時間を取られず、高機能なアプリケーションに集中して取り組めます。
Webサイト上には日本語情報も豊富で、問題発生時はフォーラム上で共有、解決が可能です。

- 高性能 mbed LPC1768 ・ ・ Cortex-M3 98MHz
- 低消費電力 mbed LPC1114 ・ ・ Cortex-M0 48MHz



mbed LPC1114U24