



不可能への挑戦

株式会社日昇テクノロジー

低価格、高品質が不可能？

日昇テクノロジーなら可能にする

ARM Cortex-M3 LPC1343

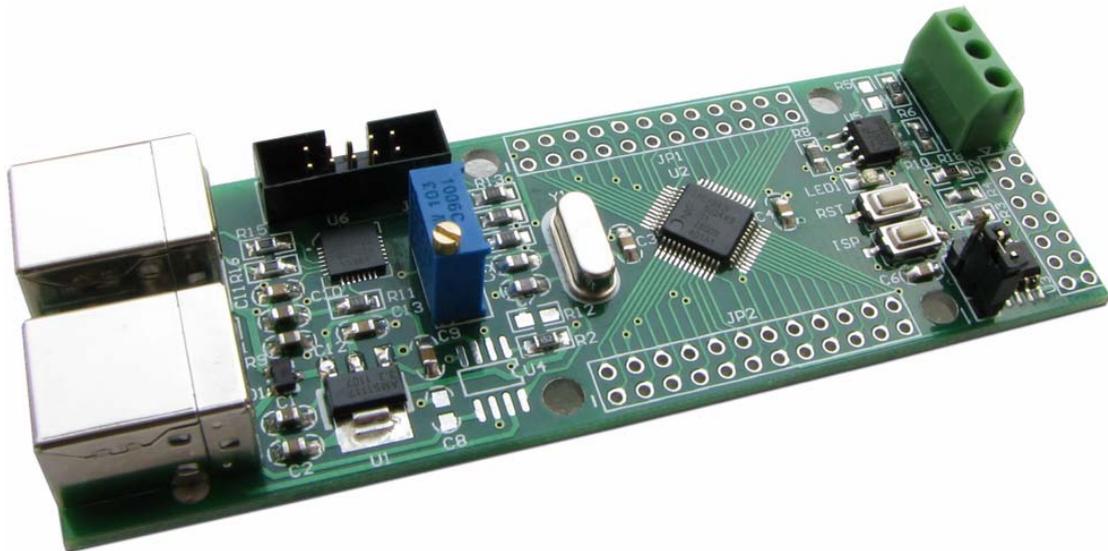
マニュアル

株式会社日昇テクノロジー

<http://www.csun.co.jp>

info@csun.co.jp

2011/08/27



copyright©2011

• 修正履歴

NO	バージョン	修正内容	修正日
1	Ver1.0	新規作成	2010/06/16
2	Ver1.1	改版、回路説明、サンプルソース説明を変更	2011/08/27

※ この文書の情報は、文書を改善するため、事前の通知なく変更されることがあります。最新版は弊社ホームページからご参照ください。

「<http://www.csun.co.jp>」

※ (株)日昇テクノロジーの書面による許可のない複製は、いかなる形態においても厳重に禁じられています。

目次

第一章 概要	5
1.1 主な特徴	5
1.2 豊富なインタフェース	5
第二章 回路の説明	6
2.1 Power Supply (電源供給)	6
2.2 JTAG	6
2.3 USB Device	7
2.4 USB to UART	7
2.5 RS485	8
2.6 IIC EEPROM	8
2.7 LED	9
2.8 ADボリューム調整	9
2.9 KEY	9
2.10 Boot Loader Select	10
2.11 拡張IO	10
第三章 実行ファイルの書き込み	11
3.1 シリアルポートで書き込む	11
3.1.1 FlashMagicのインストール	11
3.1.2 書き込み	15
3.2 ボードの内蔵USBbootloaderで書き込む	18
3.3 OpenLinkで書き込む	20
3.3.1 ドライバのインストール	20
3.3.2 J-FLASH ARMで実行ファイルを書き込む	23
第四章 OpenLinkでデバッグ	28
4.1 J-Link commandでデバッグ	28
第五章 開発ツールKEILの応用	29
5.1 KEILのインストール	29
5.2 既存のプロジェクトから	32
5.3 新しいプロジェクトの作成	41
第六章 サンプルソースの説明	47
6.1 CODE¥LPC13xx_SampleSoftware	47
6.1.1 CODE ¥LPC13xx_SampleSoftware¥Blinky	47



6.1.2	CODE ¥LPC13xx_SampleSoftware ¥UART	47
6.1.3	CODE ¥LPC13xx_SampleSoftware.101 ¥ADC_Single	47
6.1.4	CODE ¥LPC13xx_SampleSoftware.101 ¥URAT_RS485	49
6.1.5	CODE ¥LPC13xx_SampleSoftware.101 ¥i2c	49
6.1.6	CODE ¥LPC13xx_SampleSoftware.101 ¥USBCDC	49
6.1.7	CODE ¥LPC13xx_SampleSoftware.101 ¥USBCDC_RS485	50
6.1.8	CODE ¥LPC13xx_SampleSoftware.101 ¥USBMem	51
6.1.9	CODE ¥LPC13xx_SampleSoftware.101 ¥USBHID	51
6.1.10	CODE ¥LPC13xx_SampleSoftware.101 ¥USBxxx_rom	52

第一章 概要

NXP 社のオンチップ USB ドライバ搭載 LPC1343 コア、LPC1300 シリーズの中、最高性能 4mA@12MHz、17mA@72MHz の低消費電力のマイコン。更に USB Boot loader 内蔵して USB で書き込み出来る。

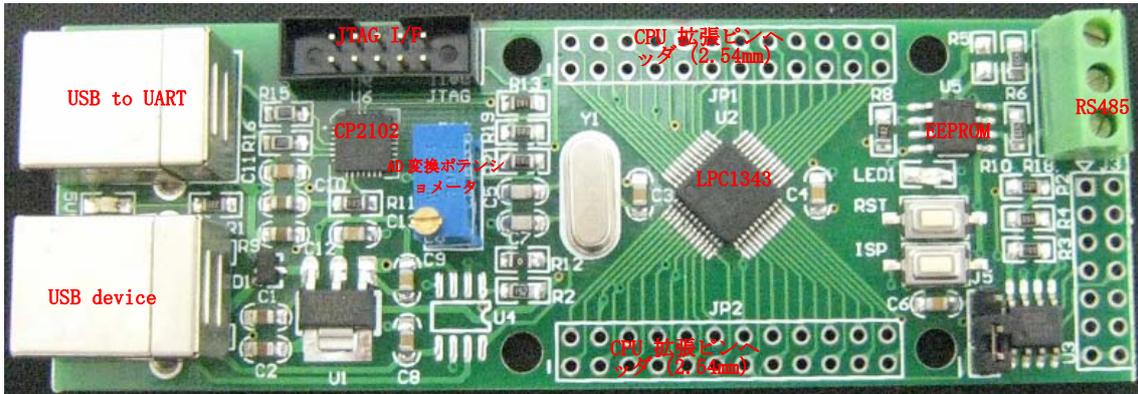
1.1 主な特徴

- パワーフルなARM 32-bit Cortex-M3マイコン、最高周波数72MHz
- 32kB Flash、8KB SRAM
- USB 2.0 全速、USBコントローラはUSB-IF(USBインプリメンターズ・フォーラム)認定のオンチップUSBドライバであり、USBの四つの転送タイプをサポートする。USBドライバは Mass StorageとHuman Interface Deviceの両方を利用できる
- JTAG及びトレース機能付きSWDインタフェース
- Modemインタフェース付きUART

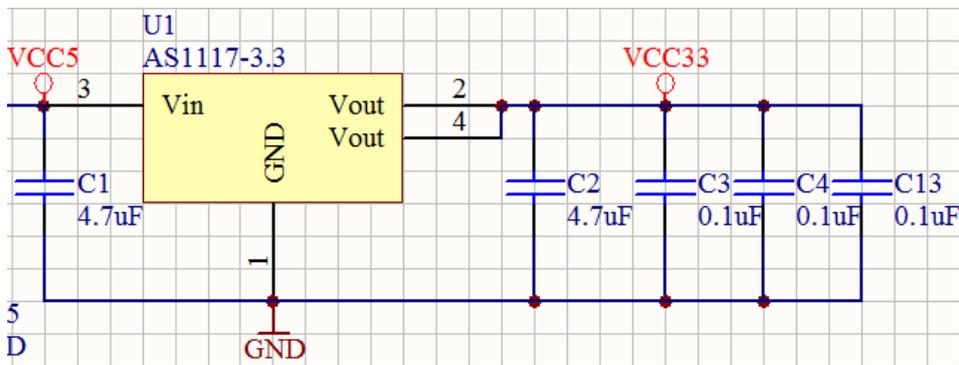
1.2 豊富なインタフェース

- SWDデバッグインタフェース、JLINK/ULINK2をサポートする。
- RS485 x 1
- USB2.0全速インタフェース
- USBからUART変換インタフェース (CP2102)
- 10K可変ポテンシオメータ x 1
- ユーザーLED x 1
- ユーザーボタン x 2
- IIC インタフェース、EEPROM、24C04
- USBポートで給電
- CPU のすべての IO を 2.54mm 拡張ピンヘッダで引き出されている
- 外形寸法: 96×33 (mm) ※突起物は除く

第二章 回路の説明



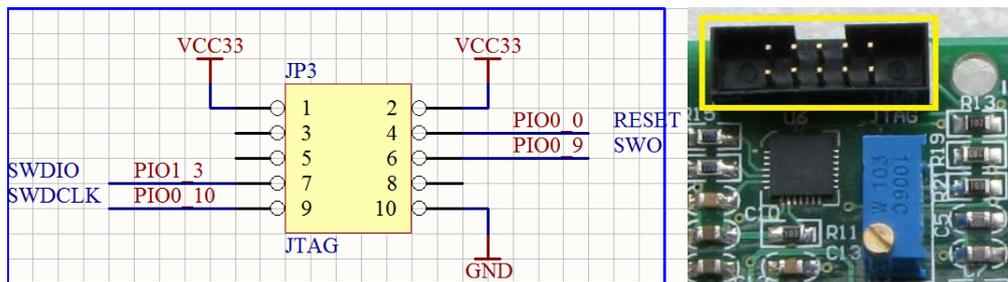
2.1 Power Supply (電源供給)



入力 : 5v

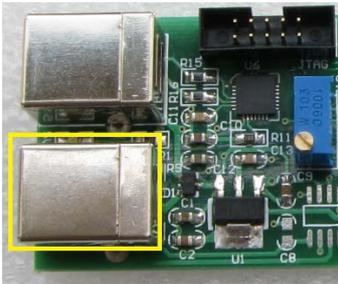
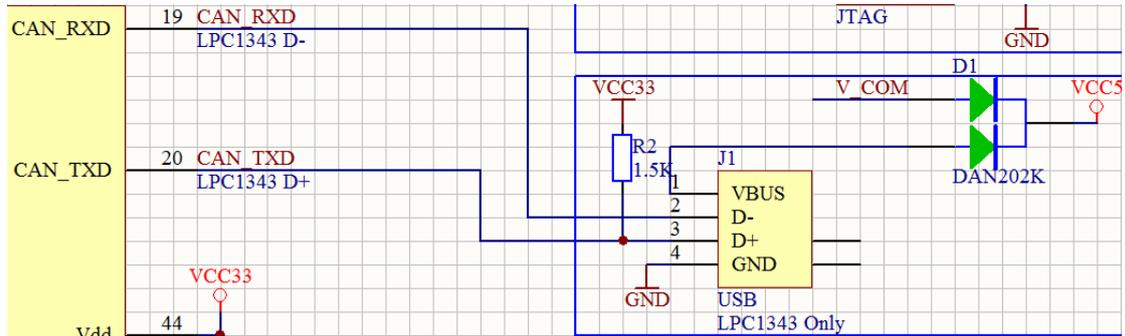
出力 : 3.3v

2.2 JTAG



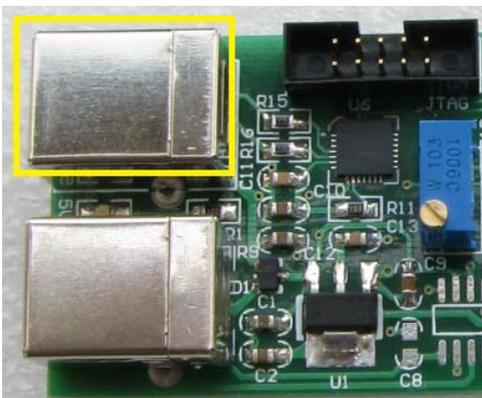
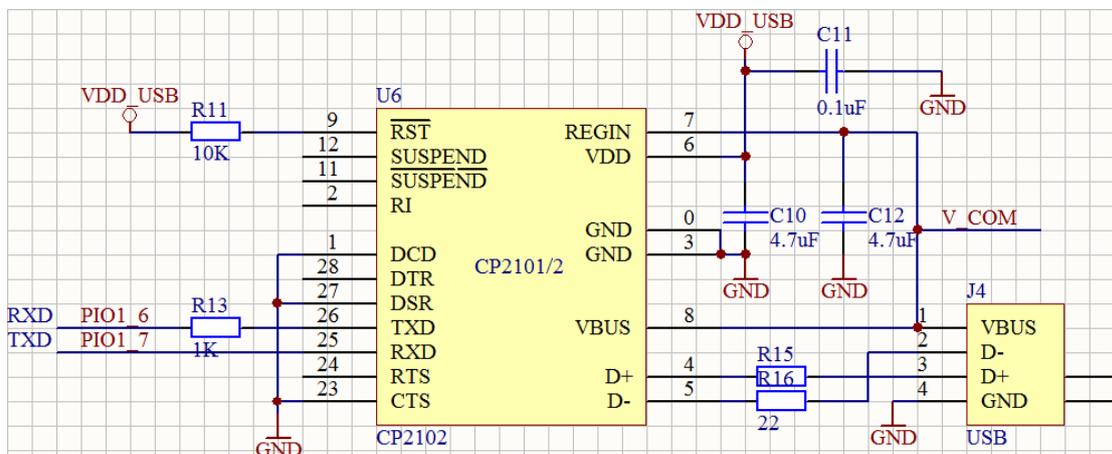
SWD デバッグインタフェースを採用しております。2.0mm ピッチとなっております。

2.3 USB Device

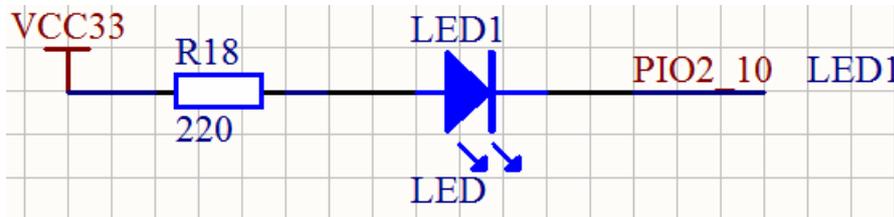


ボードに搭載している二つの USB ポート両方もボードに給電できる。

2.4 USB to UART

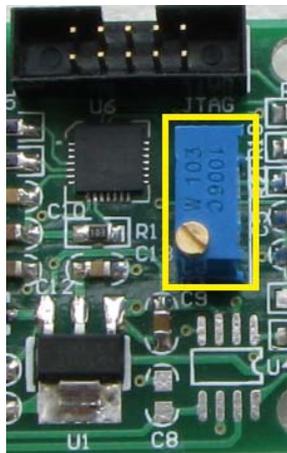
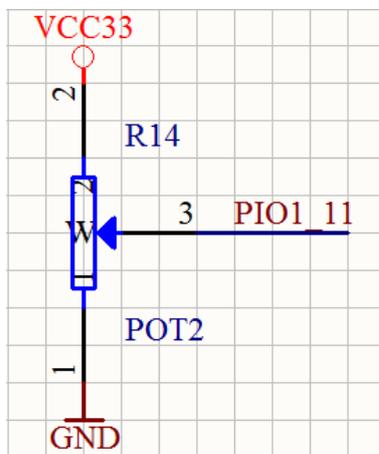


2.7 LED



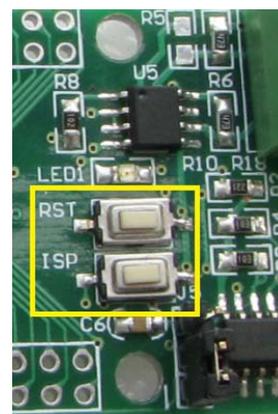
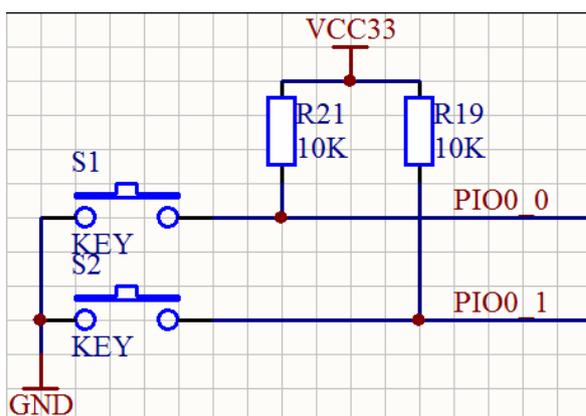
IOポートがLowの場合LED点灯する。

2.8 ADボリューム調整



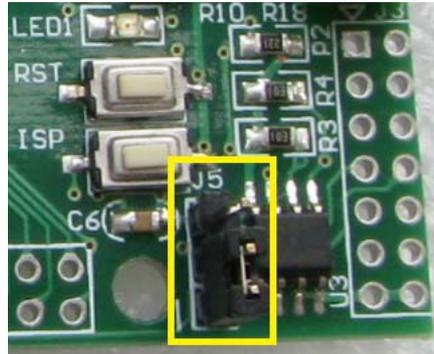
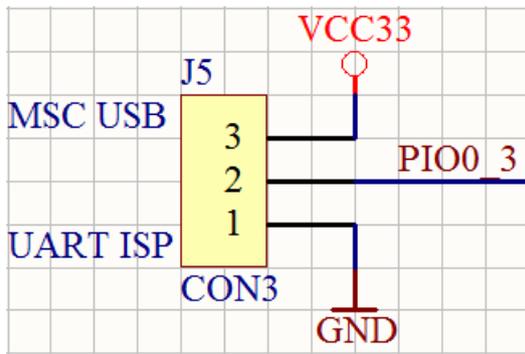
R14は10Kの精密抵抗。

2.9 KEY



RESET キーもユーザーキーとして利用できる。

2.10 Boot Loader Select

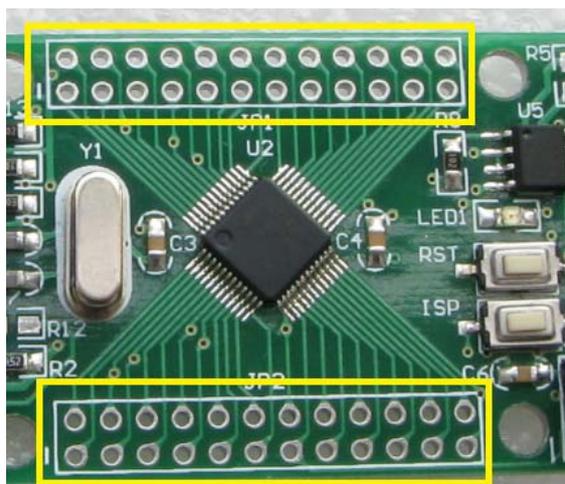
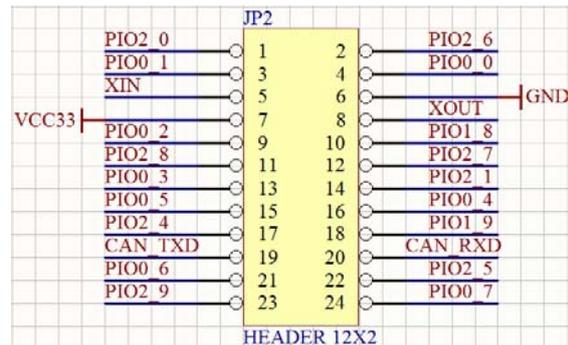
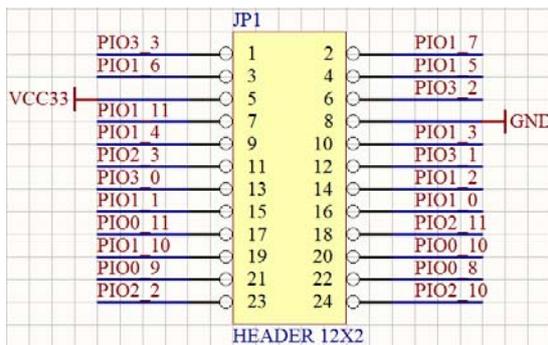


J5 ジャンパーで boot loader モードを設定します。

2-3: 大容量メモリモード(ファイルを U ディスクにコピー) でボードのプログラムを更新。

1-2 (上記画面) : UART ISP モードでボードのプログラムを更新。

2.11 拡張IO



本ボードはCPUの全てのIOを2.54mmピッチで引き出して、デバッグで利用できる。

第三章 実行ファイルの書き込み

3.1 シリアルポートで書き込む

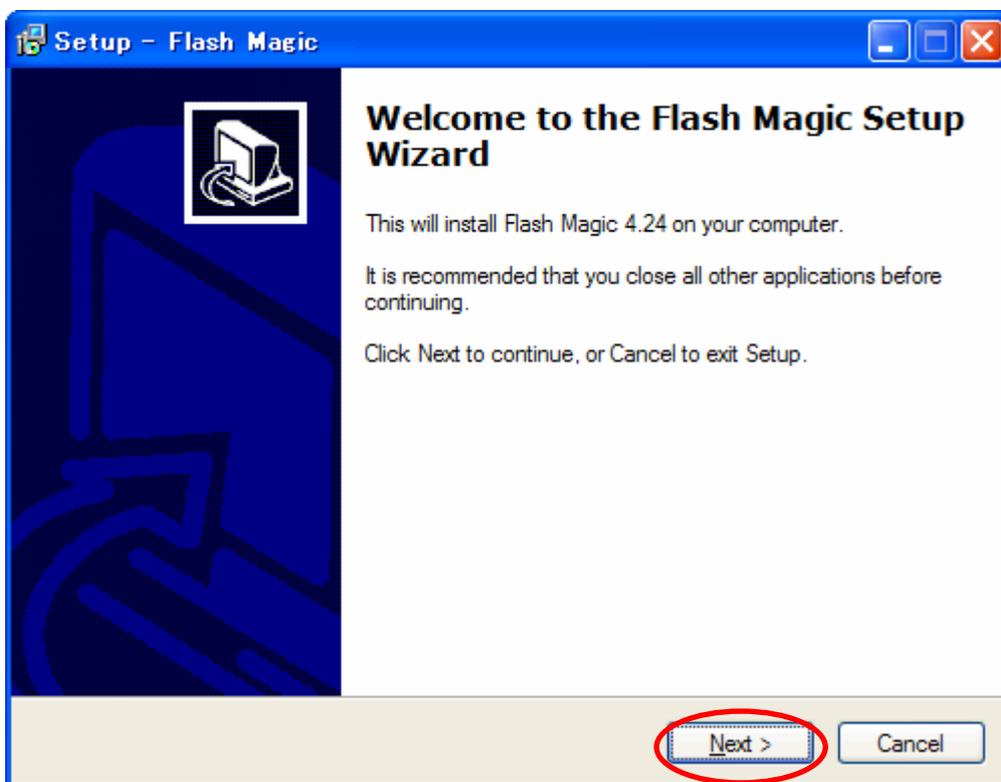
FlashMagic とは

LPC シリーズはフラッシュ ROM を内蔵しているため、ISP (In-System Program) 機能によりユーザ・プログラムを書き込むことができます。そのためのプログラミング・ツールが FlashMagic です。FlashMagic は次の URL からダウンロードできます。

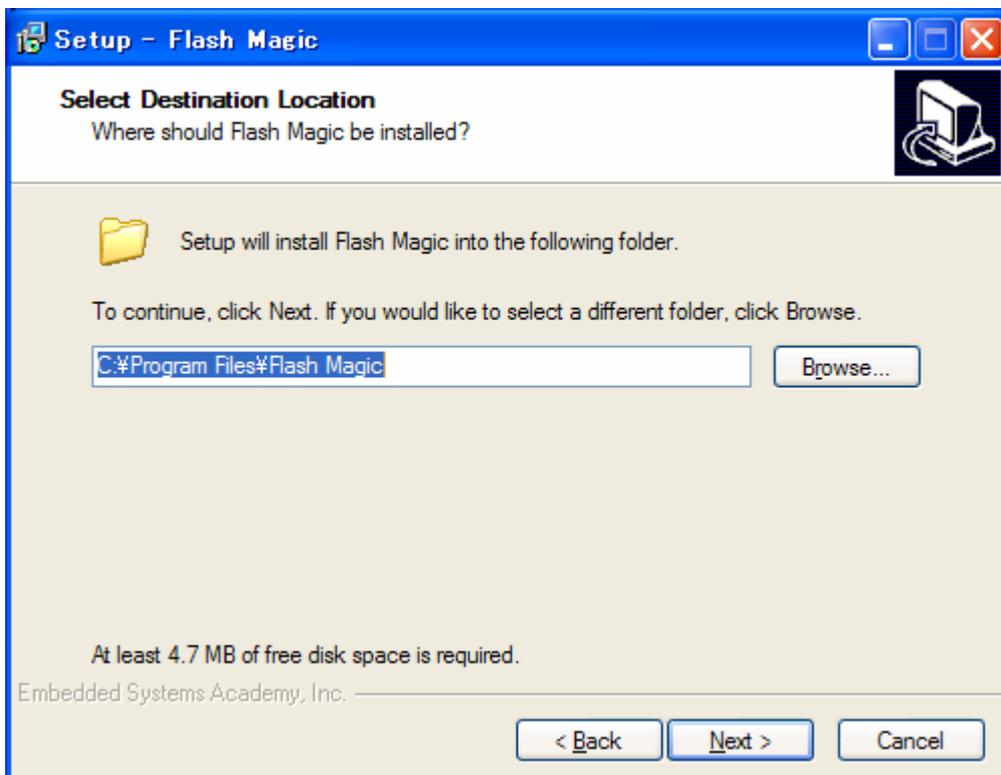
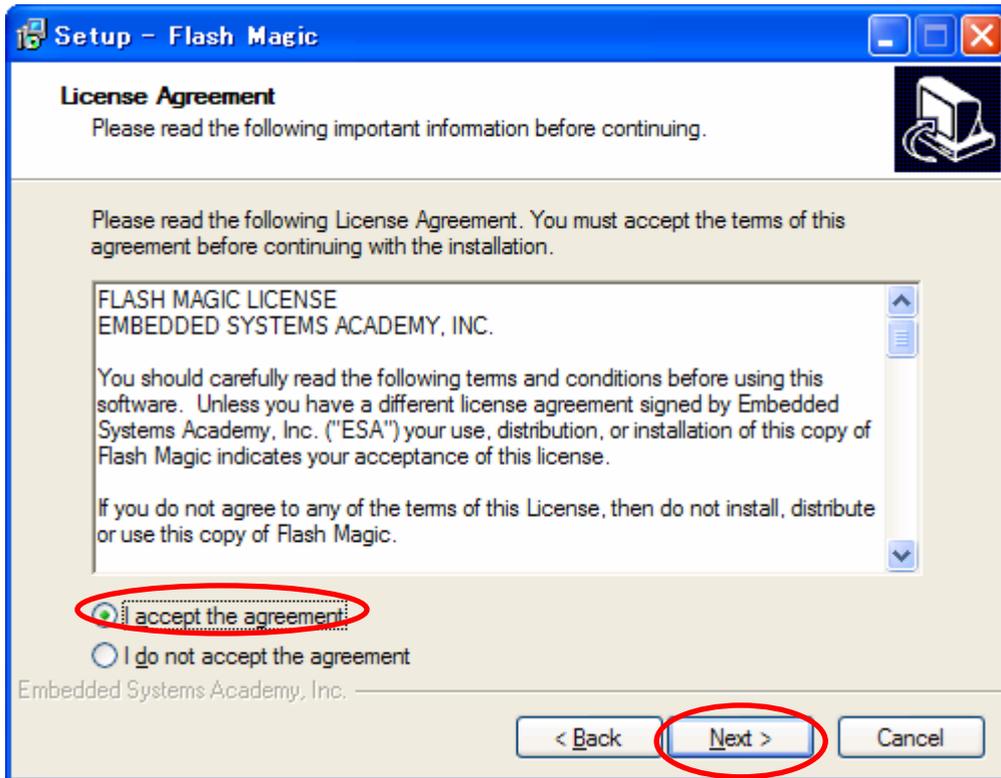
<http://www.flashmagictool.com/>

3.1.1 FlashMagicのインストール

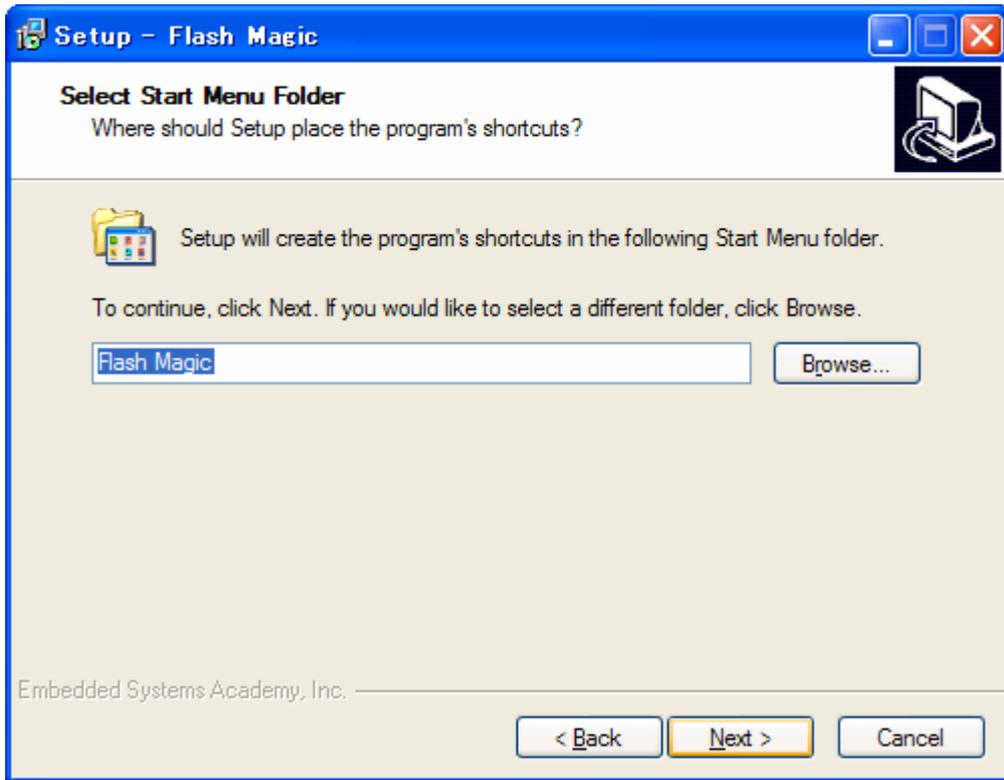
FlashMagic.exe を実行してインストールする。



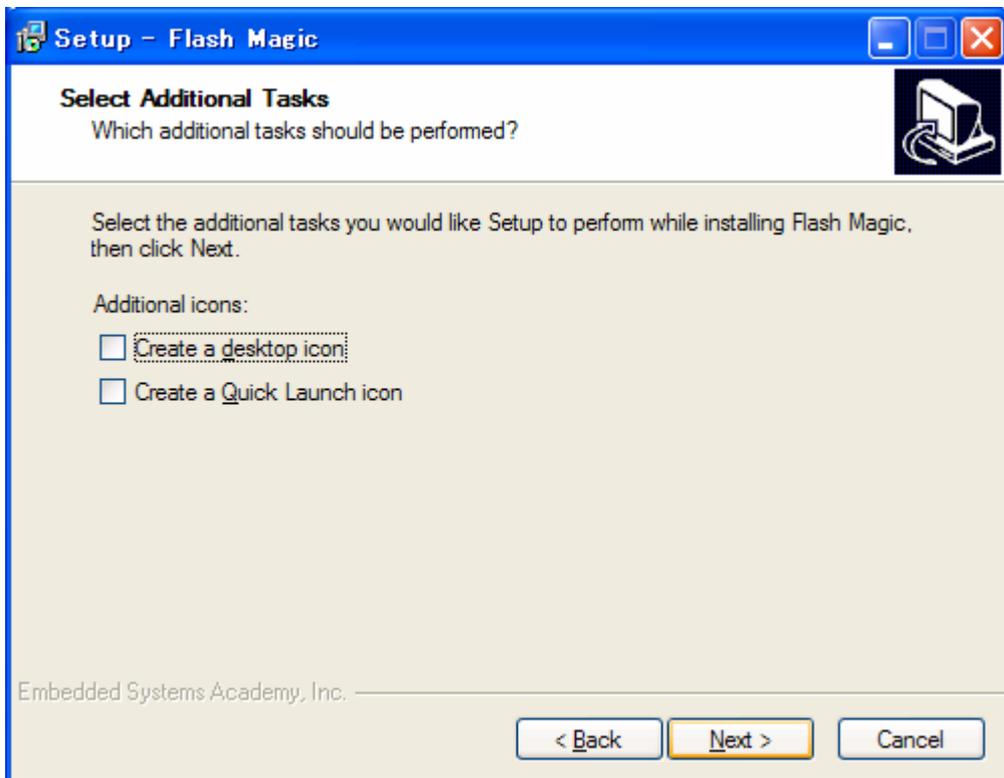
「Next」ボタンを押すと、英文のライセンスが出てきます。同意できる場合は、「I accept the agreement」を選択して、「Next」ボタンを押します。



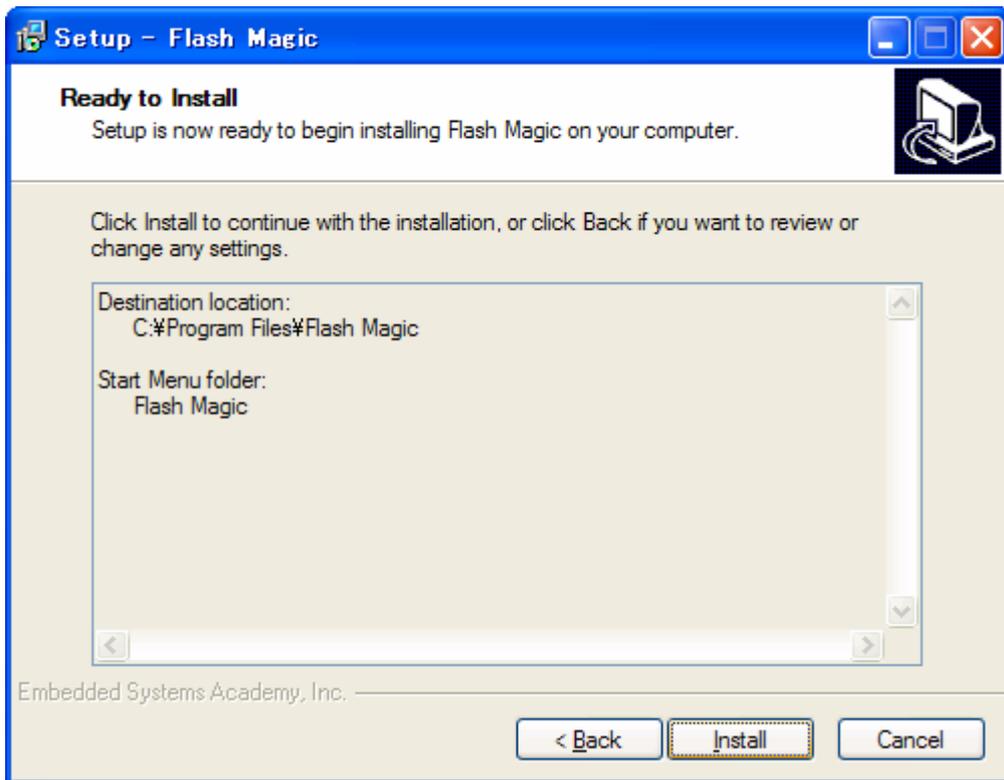
インストール先フォルダを変更せず、そのまま進んでください。



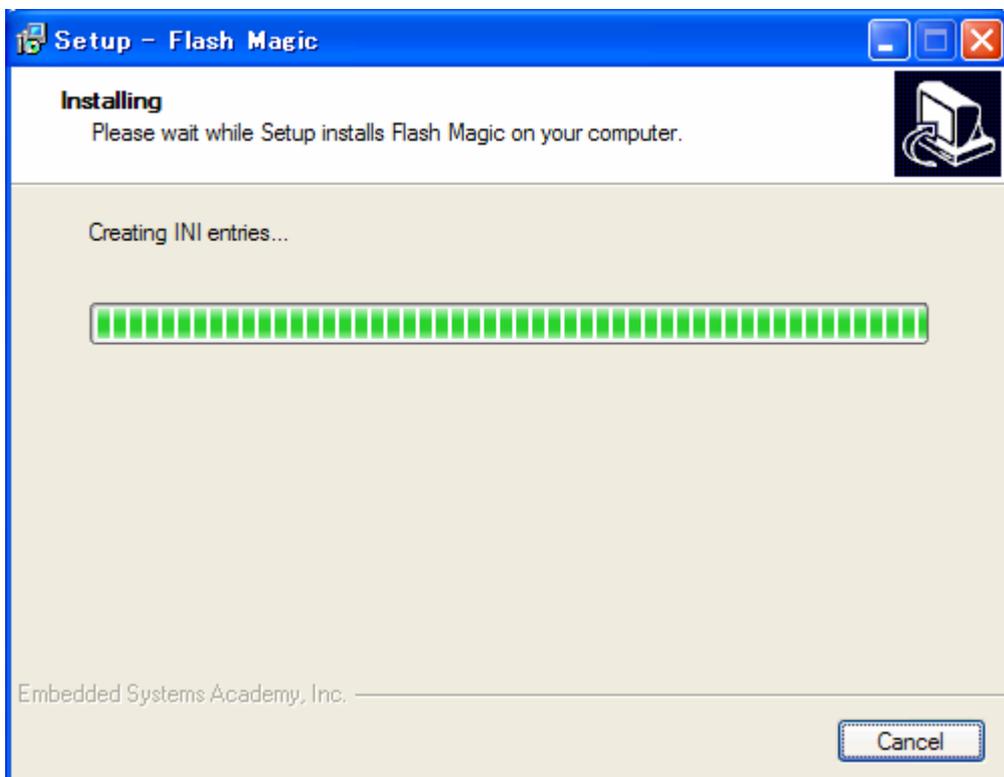
メニュー・フォルダも変更せず、そのまま進んでください。



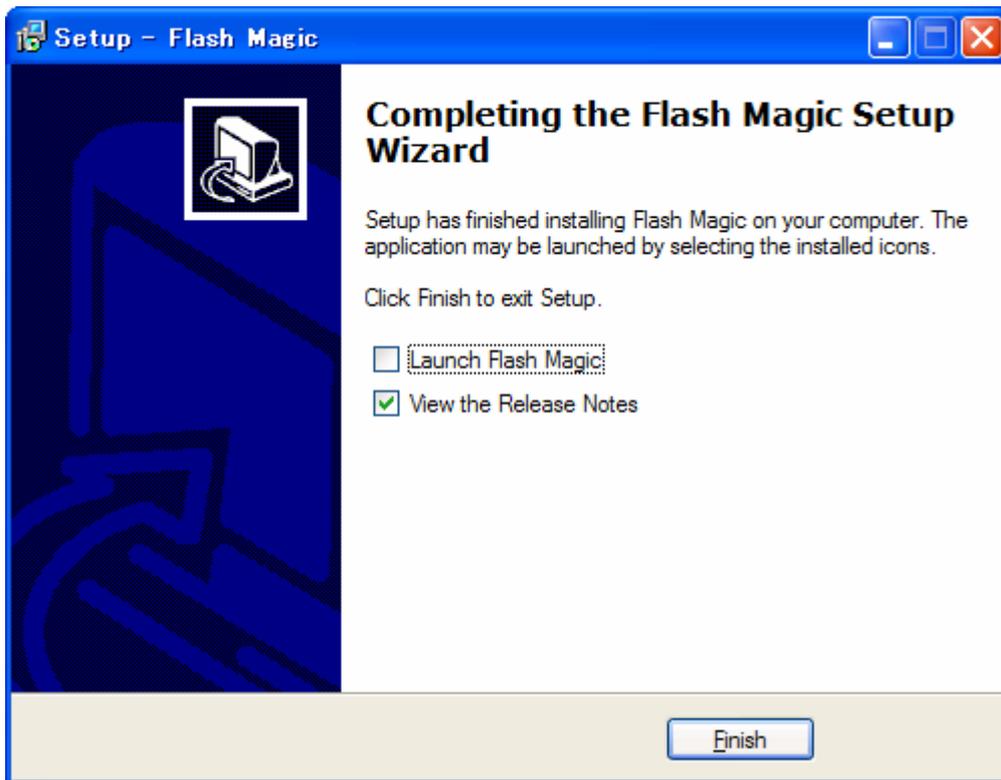
そのまま進んでください。



インストール前の確認、「install」ボタンを押してください。

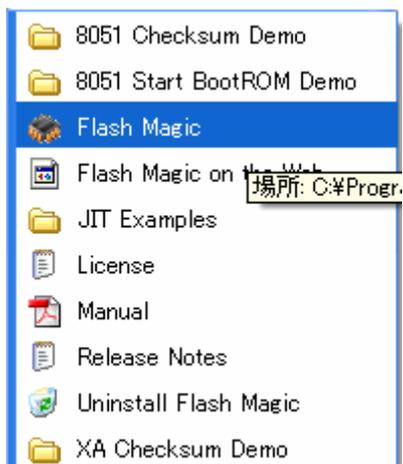


インストール中の画面です。

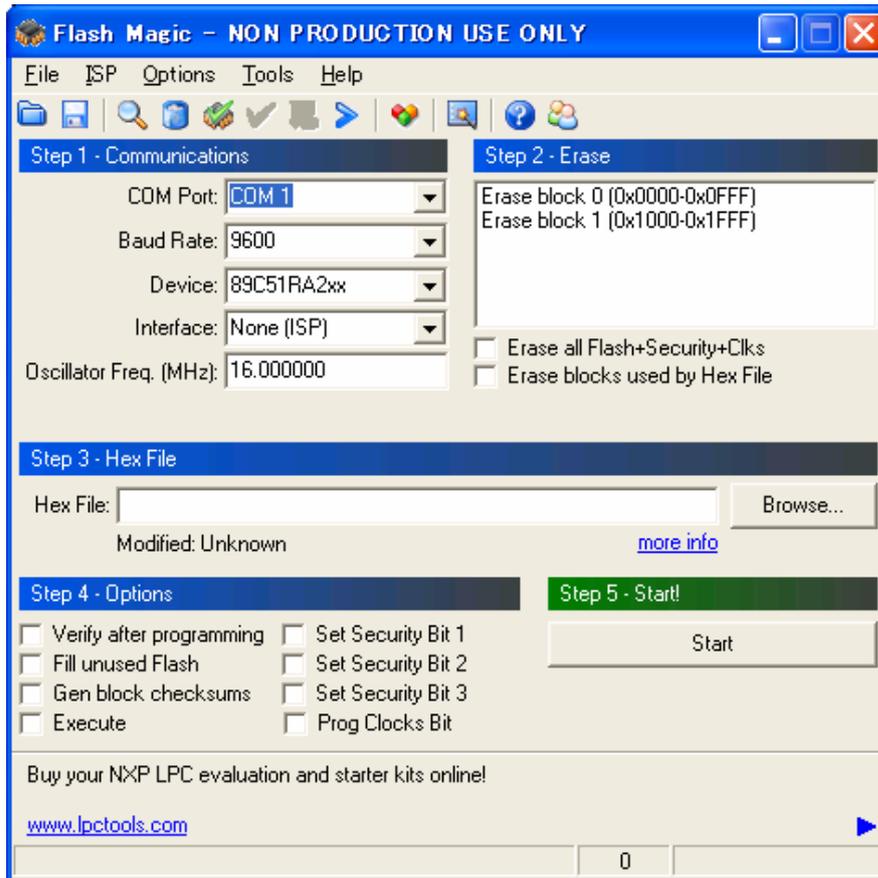


最後に「Finish」をクリックすると、ウィザードが閉じてインストールが終了。

3.1.2 書き込み

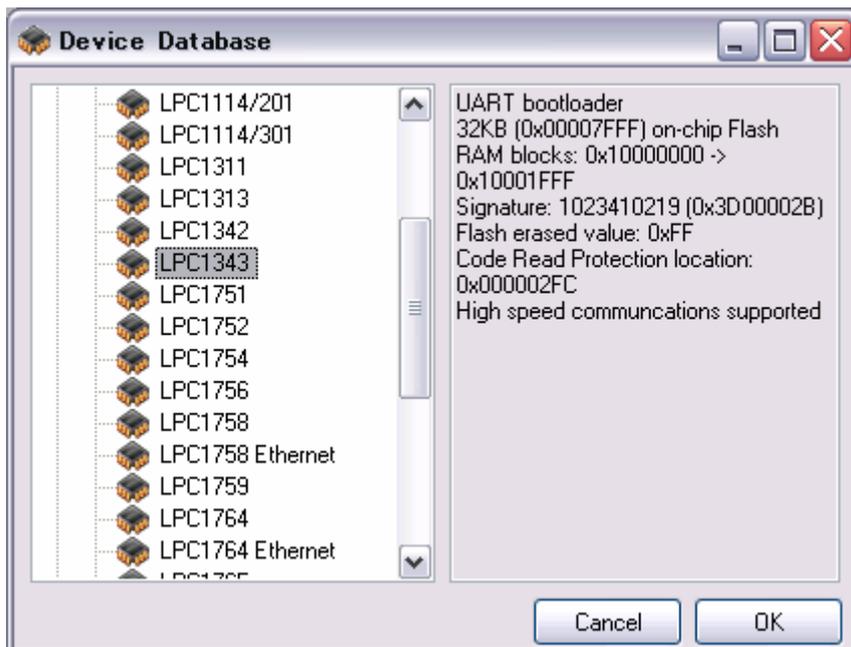


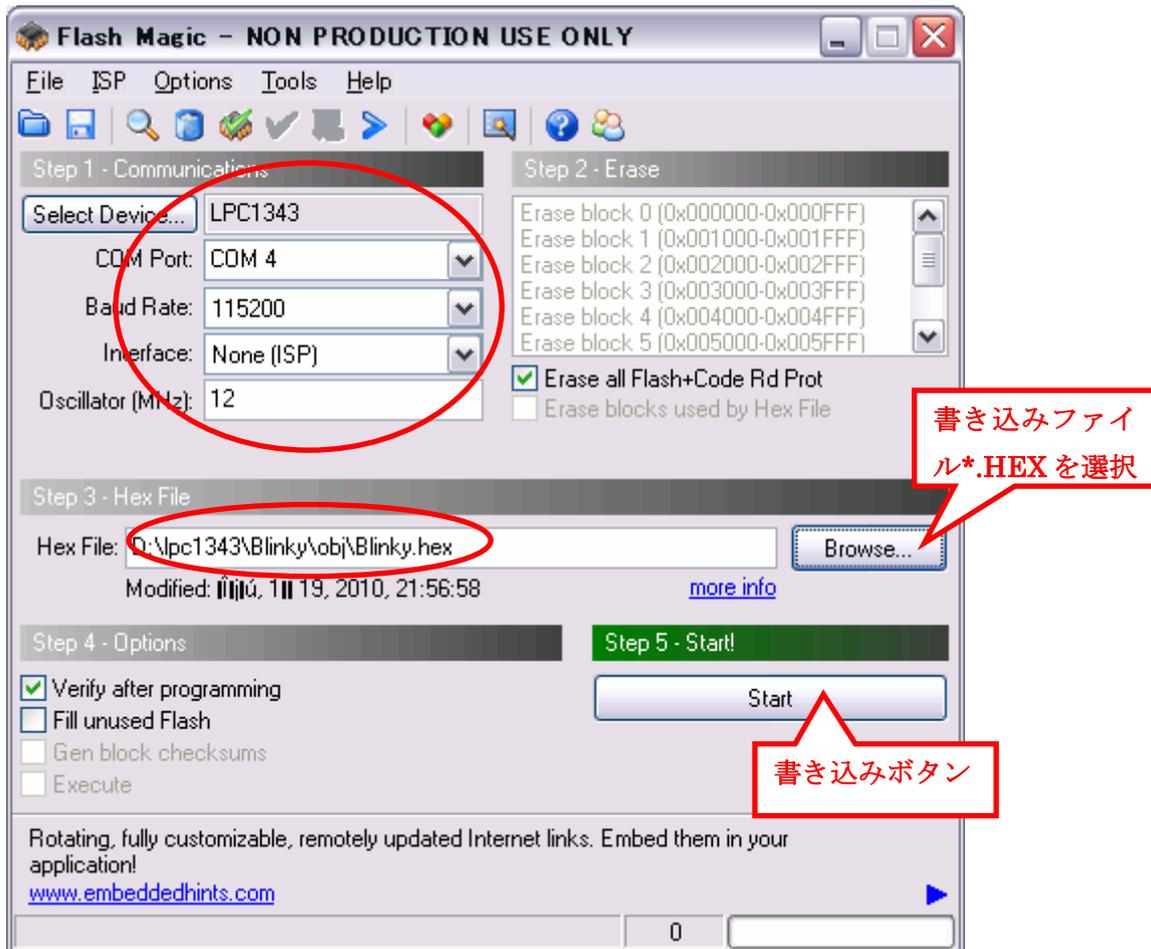
Windows のメニュー「スタート」→「Flash Magic」→「Flash Magic」を選択してください



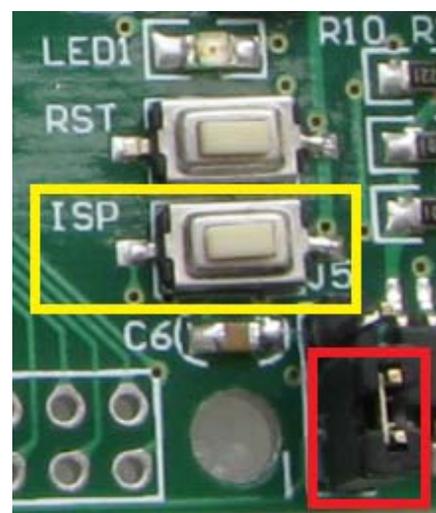
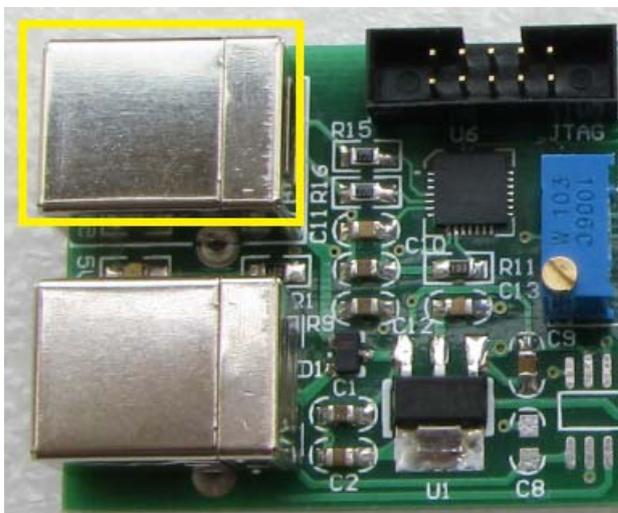
Flash Magic の初画面です。

Device 選択画面で LPC1343 を選択する。





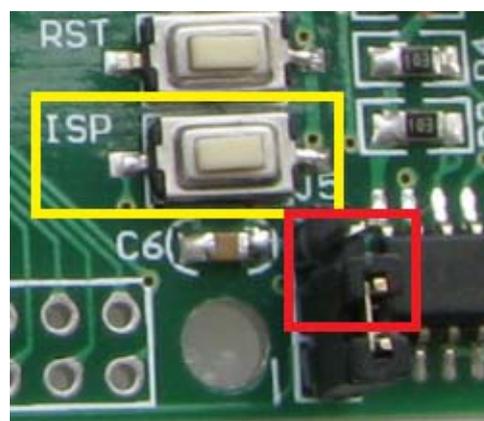
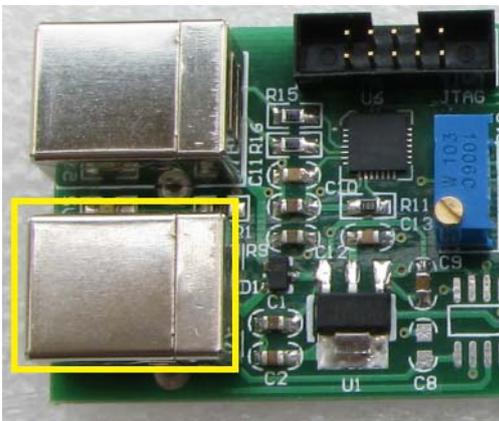
- 1、画面の通りにパラメータを設定して、「Browse」ボタンで書き込みファイル*.hexを選択する。
- 2、tools\CP210x_VCP_Win_XP_S2K3_Vista_7.exe を実行する。
- 3、J5のジャンパーを1-2に設定し、ボード上のISPキーを押しながら、USBケーブルでボードのJ4とPCを繋ぐ。ボードはISP状態に入る。



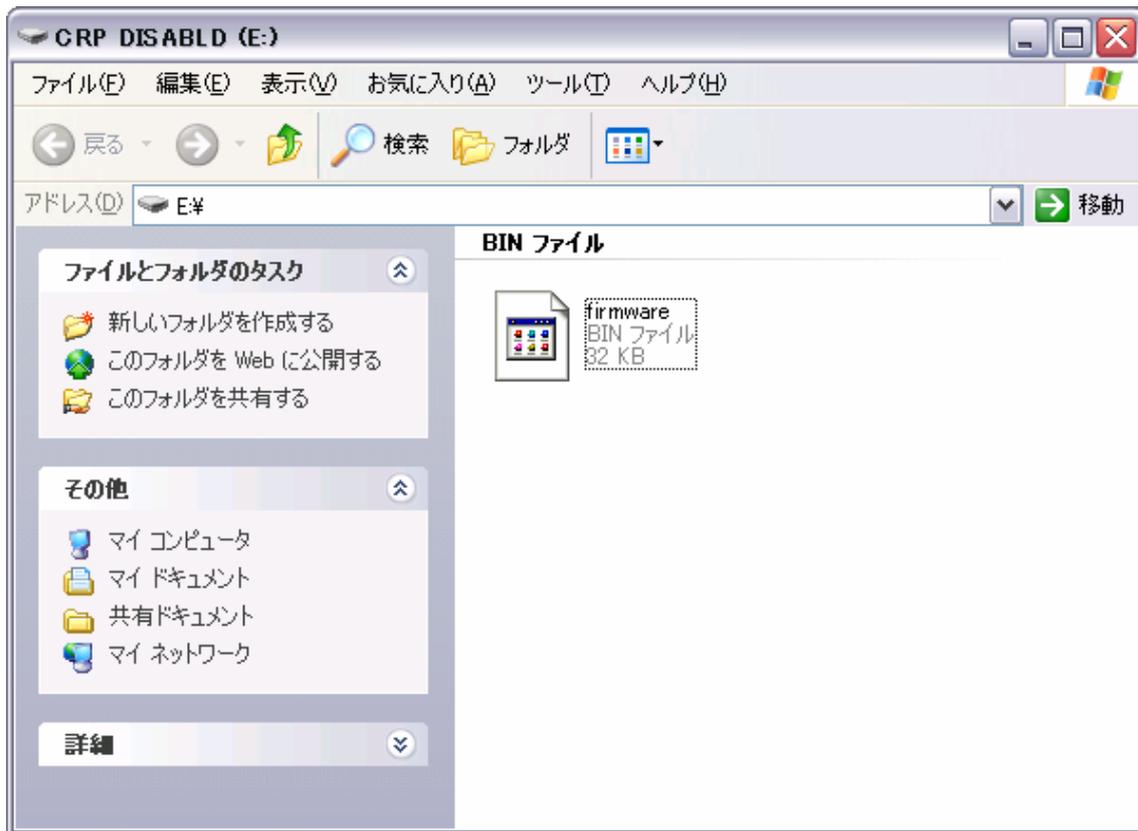
- 4、 デバイスマネージャから配分されたシリアルポート番号を確認してFlash Magic上のCom Portを設定する。
- 5、 Start ボタンをクリックすると書き込み始める。
- 6、 ボードを再起動すると、書き込んだファイルが実行する。

3.2 ボードの内蔵USBbootloaderで書き込む

- 1、 J5 のジャンパーを 2-3 に設定し、ボード上の ISP キーを押しながら、USB ケーブルでボードの J1 と PC を繋ぐ。

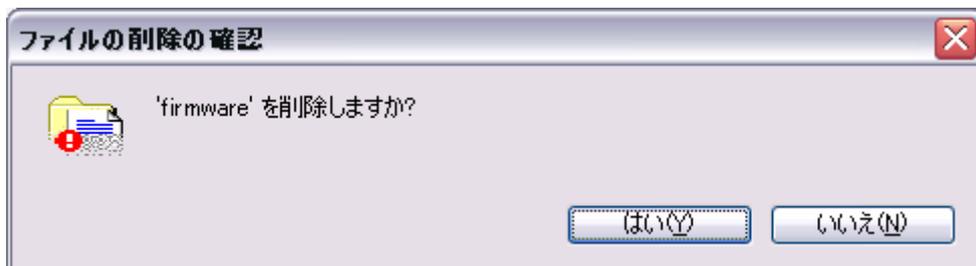


- 2、 ボードは USB ブートローダ状態に入る。初回の場合は新しいハードウェア設備発見されて自動的にドライバのインストールが行う。インストール終了後新しい移動メモリデバイスができる (15 秒ぐらい時間がかかります)。

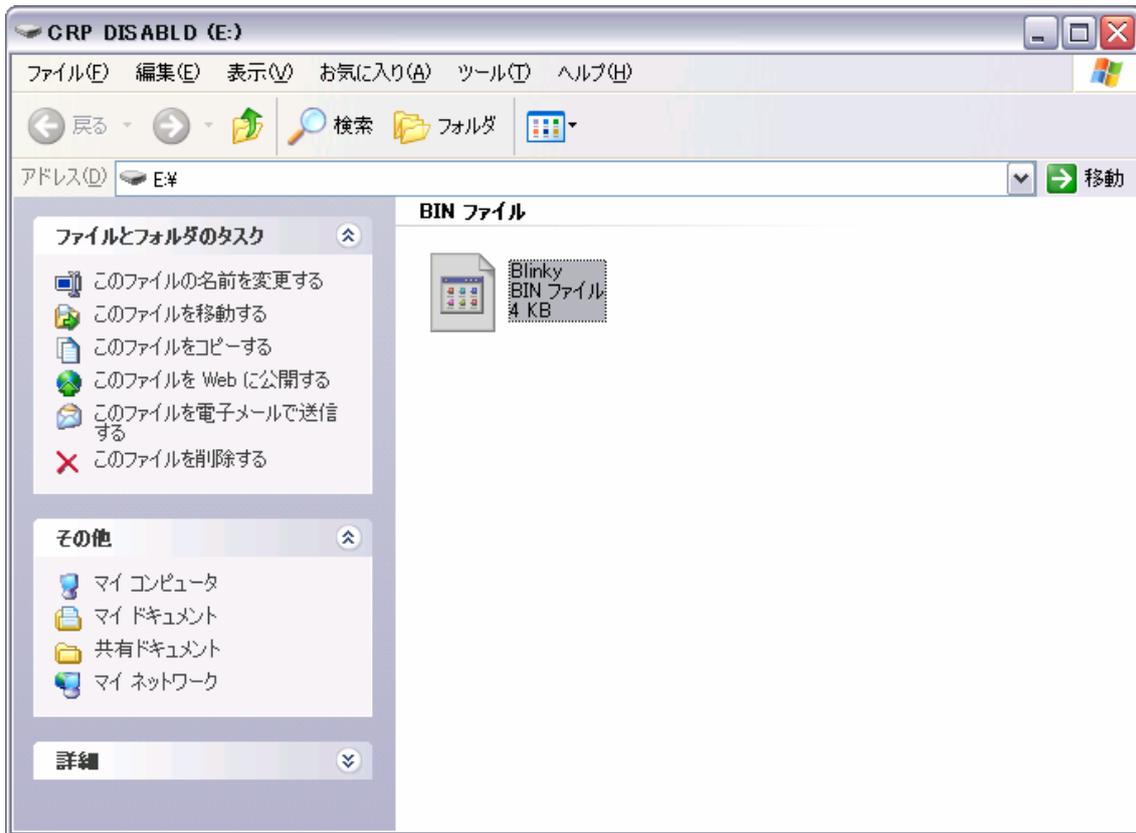


3、デバイスを開くと、firmware.bin ファイルがある。

4、firmware.bin ファイルを削除する。



5、実行したい bin ファイルをこのデバイスにコピーする。これで書き込み終了。



6、ボードを再起動すると、新しくコピーしたファイルが実行する。

3.3 OpenLinkで書き込む

弊社は OpenLink のハードウェアを提供しております（製品紹介 URL: <http://www.csun.co.jp/SHOP/2009121901.html>）。

3.3.1 ドライバのインストール

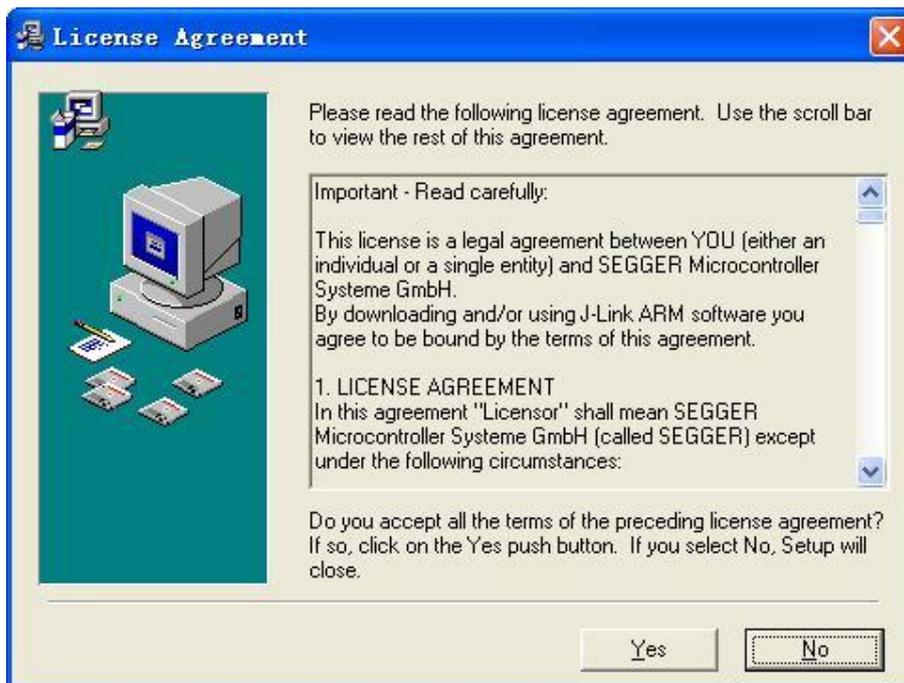
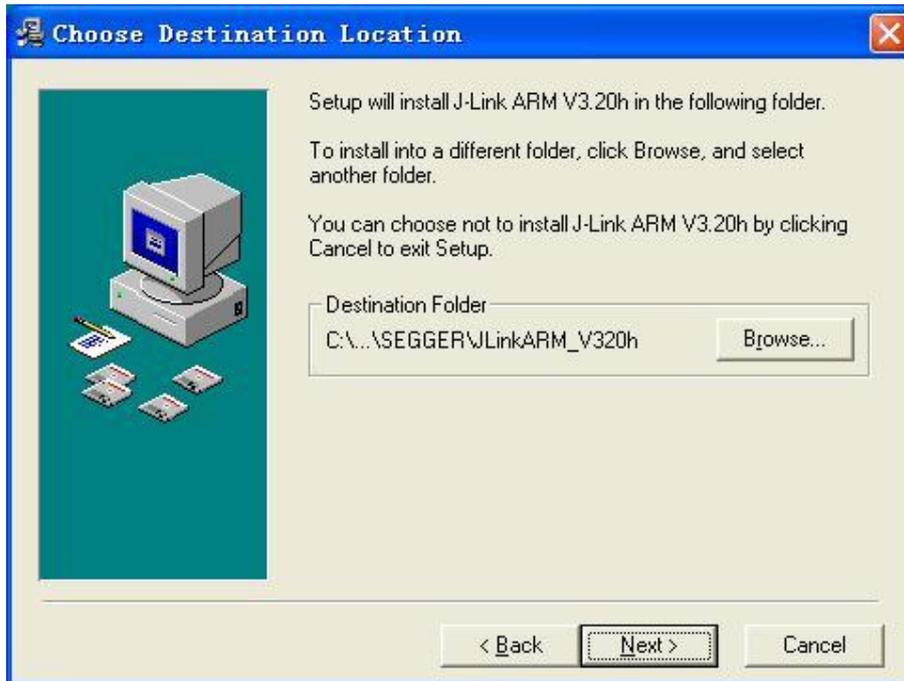
ドライバインストール用のファイルは弊社ホーム下記 URL からダウンロードできる。

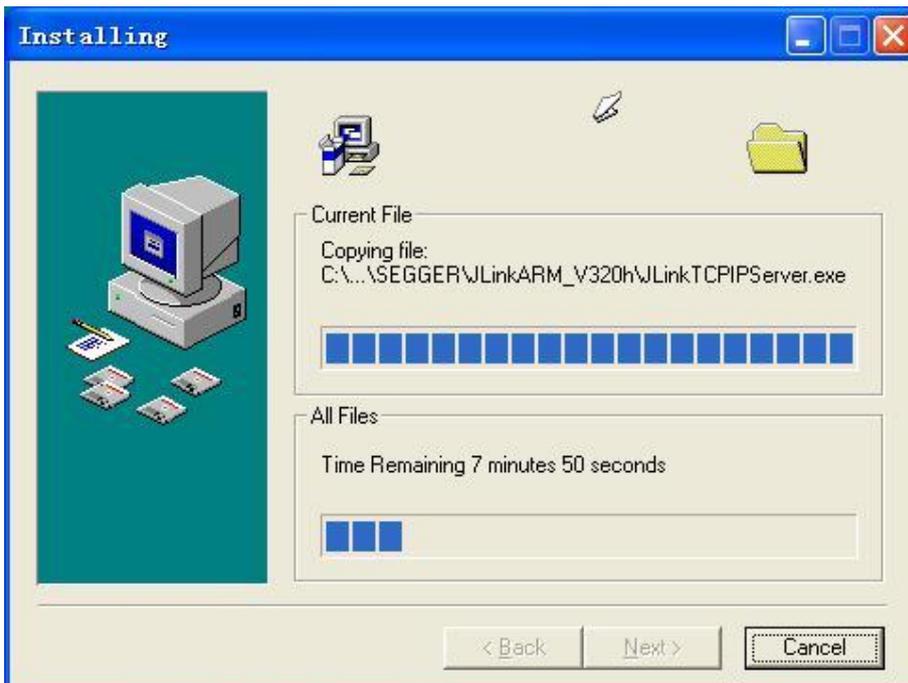
http://www.dragonwake.com/download/open-link/Setup_OpenLinkARM.zip

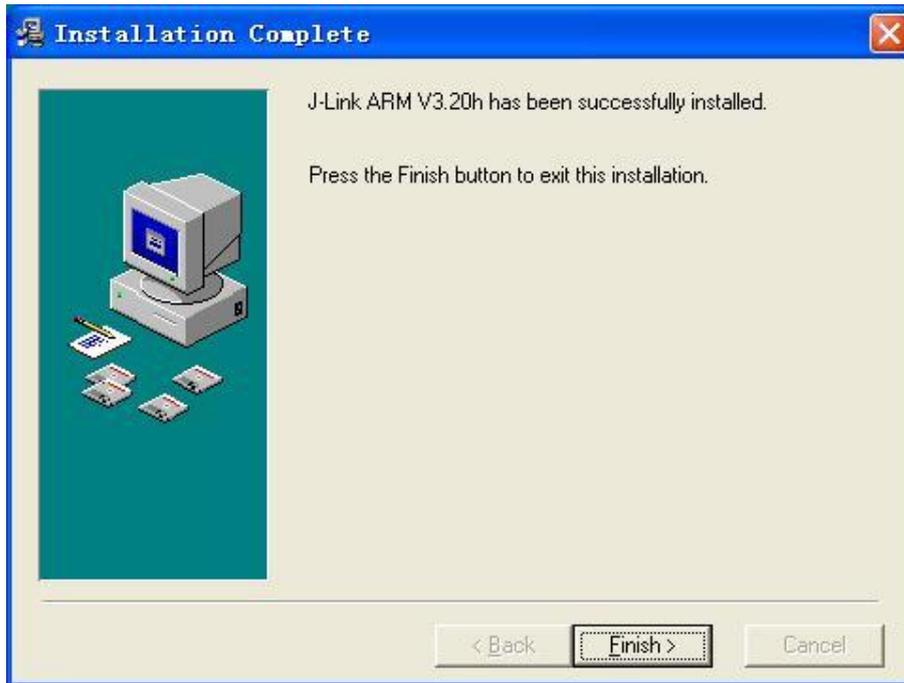
SEGGER 社様のソフトウェアを利用しておりますので、直接 SEGGER 社様ホームページから最新の USB ドライバをダウンロードできる。

<http://www.segger.com/cms/jlink-software.html>

インストールの際に、ダウンロードした ZIP ファイルを解凍し、デフォルトのままで行ってください。

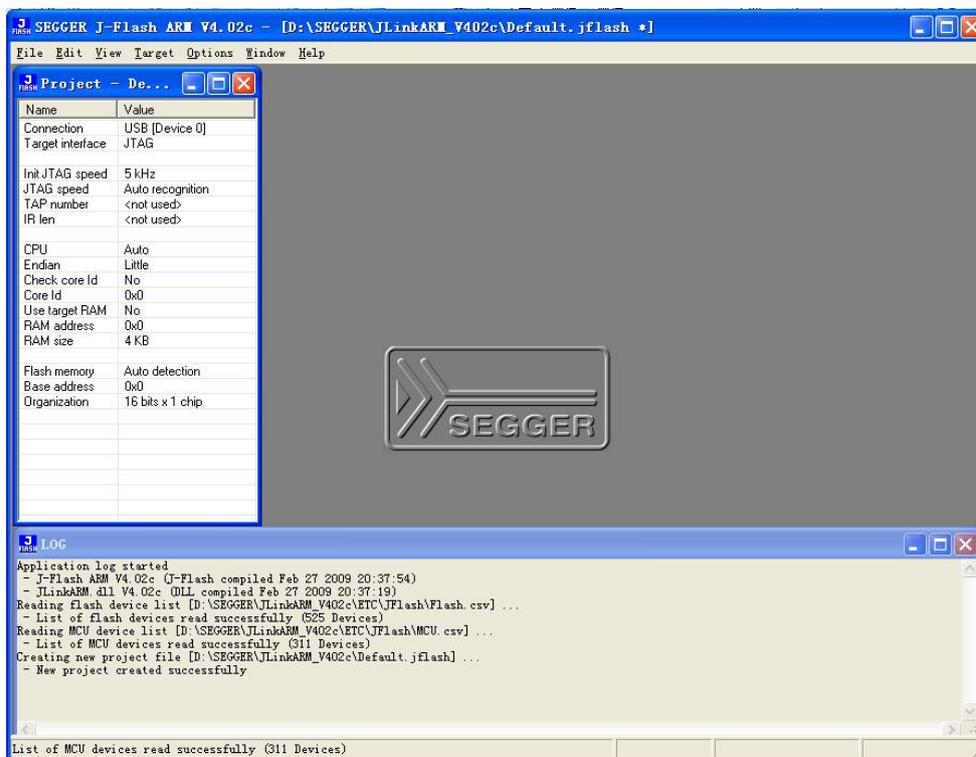




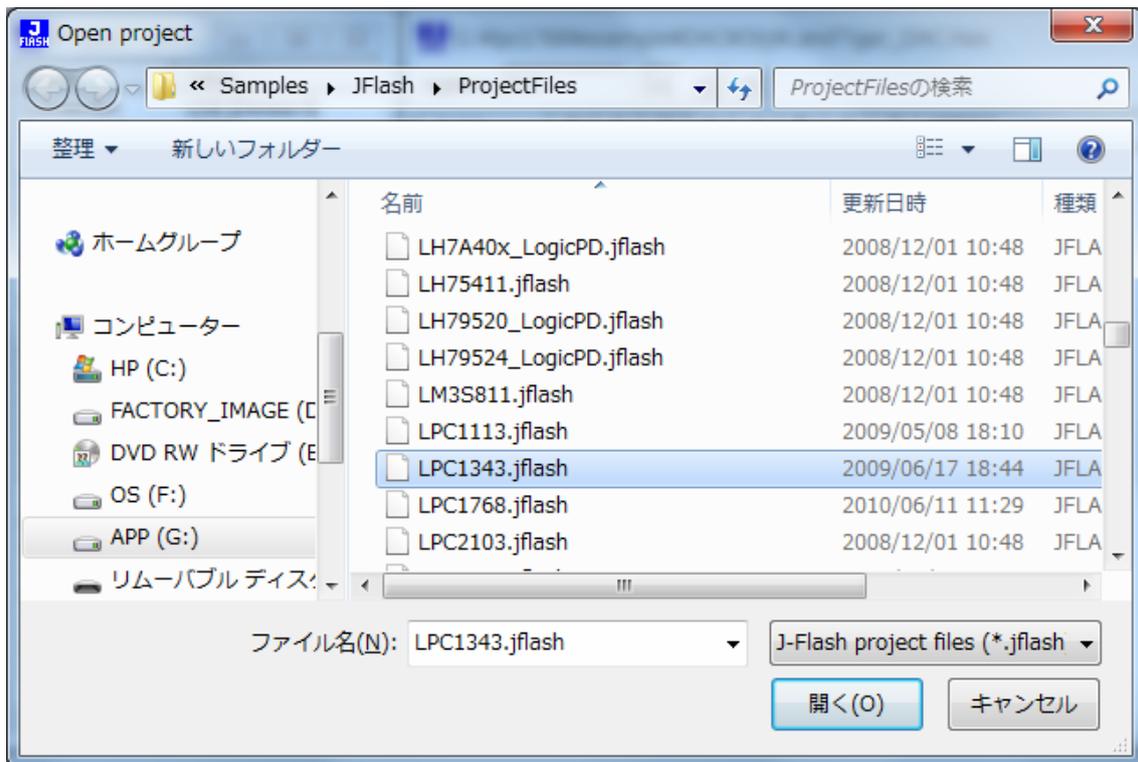


3.3.2 J-FLASH ARMで実行ファイルを書き込む

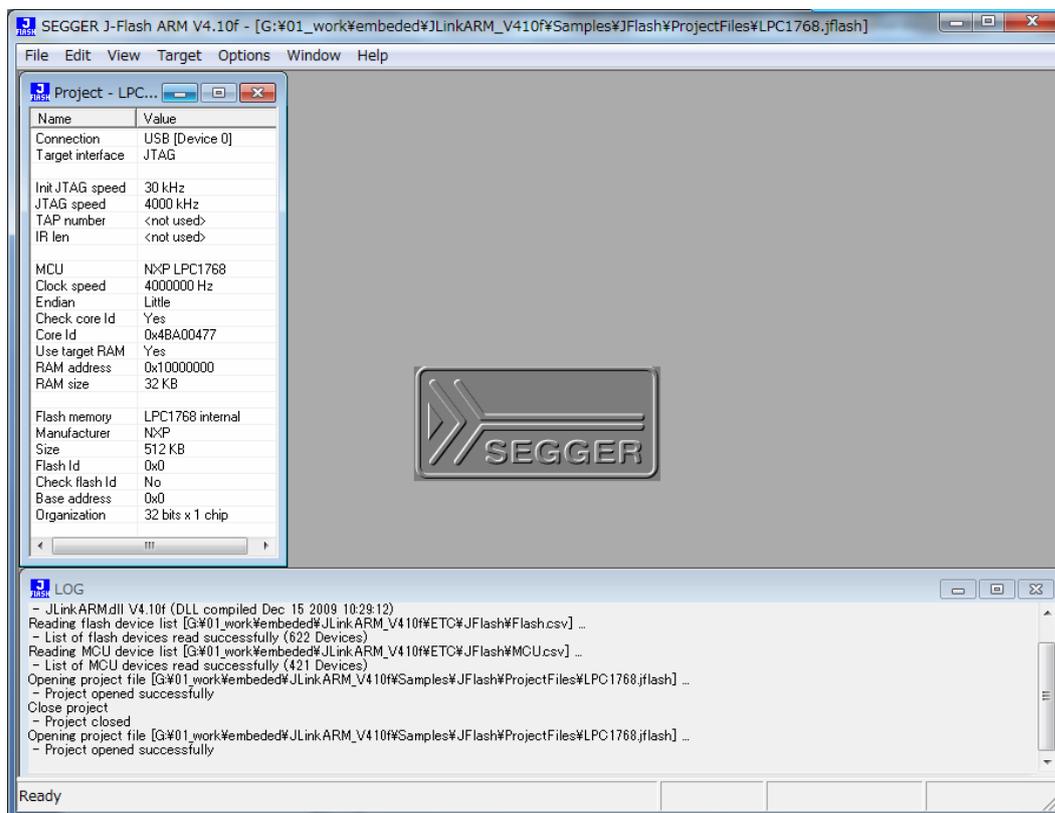
J-FLASH ARM を実行する。



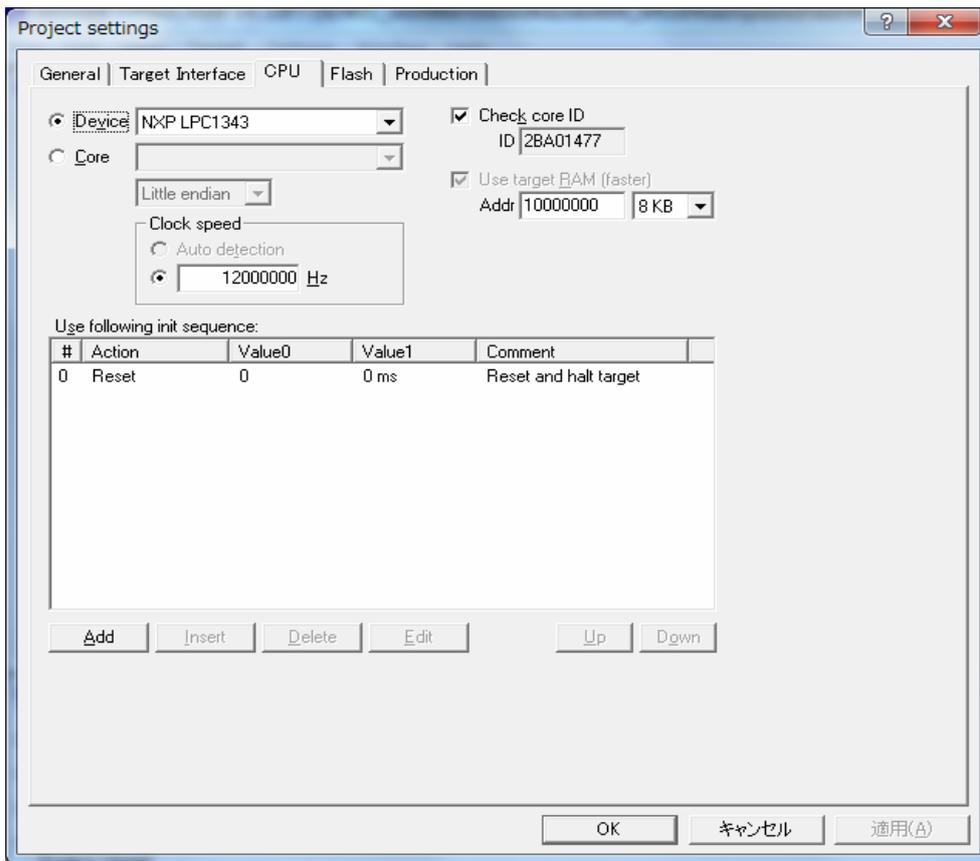
File->Open Project...で LPC1343 を選択する。



書き込む前に必要な設定 (Options->project settings...) :



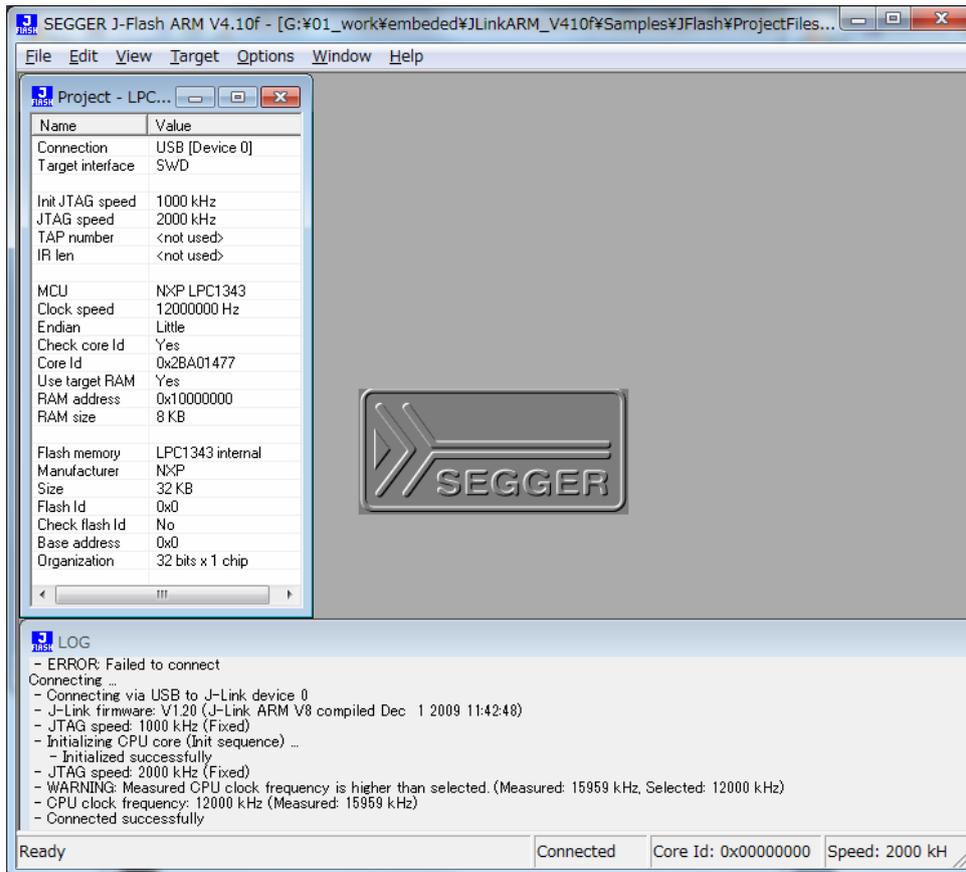
LPC1343 ボードの CPU 型番を選択する。



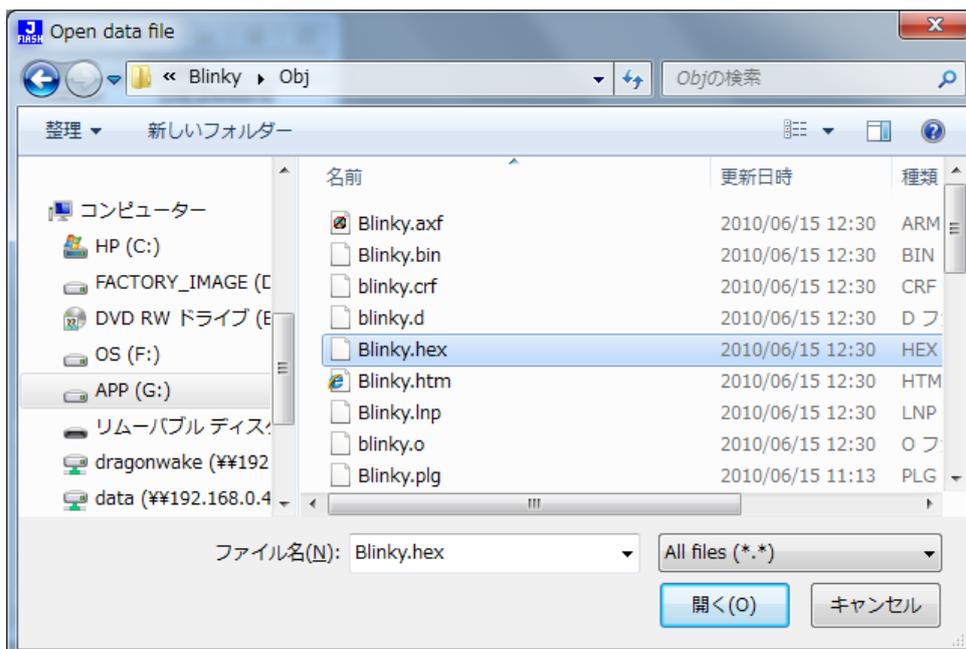
設定後、左側に書き込み情報が表示される。
ボードを接続する。



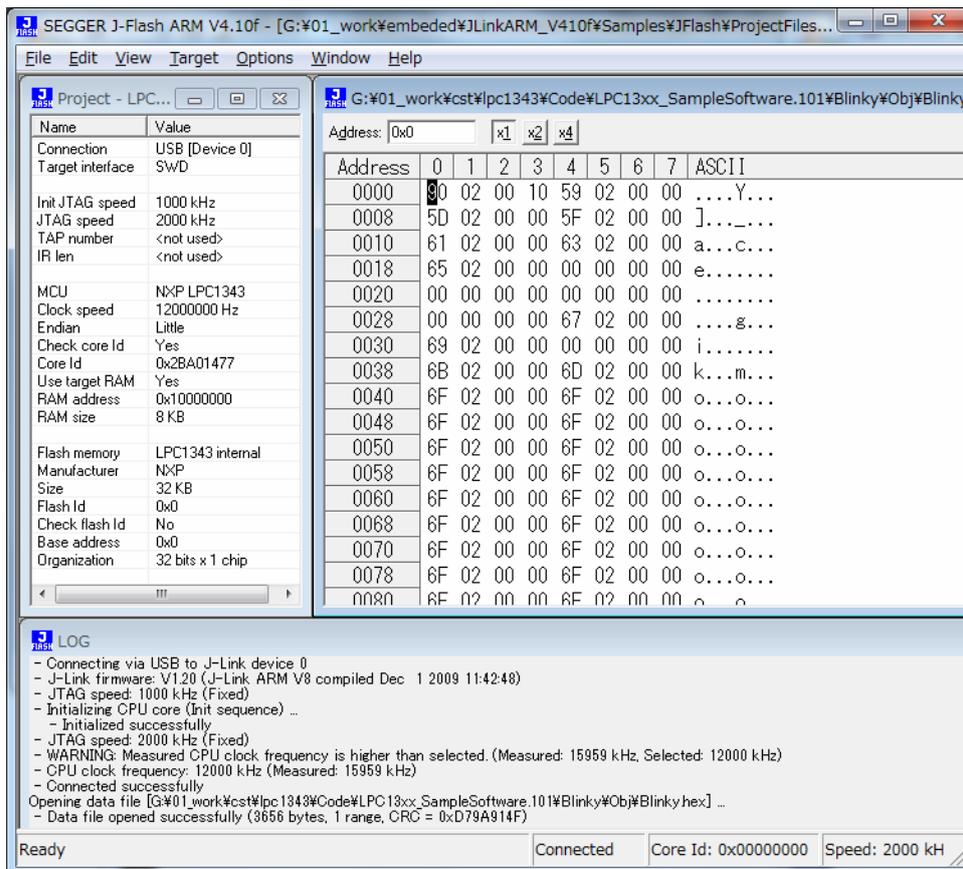
ソフト側も接続する (Target→Connect)。



File->open で実行ファイルを選択する。

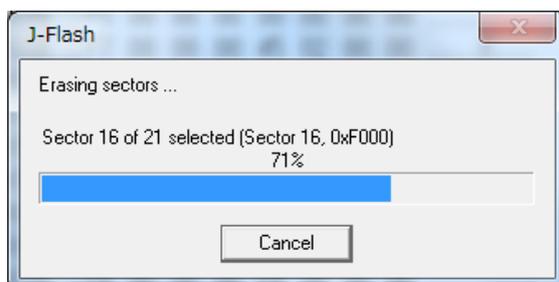


「開く (O)」をクリックする。

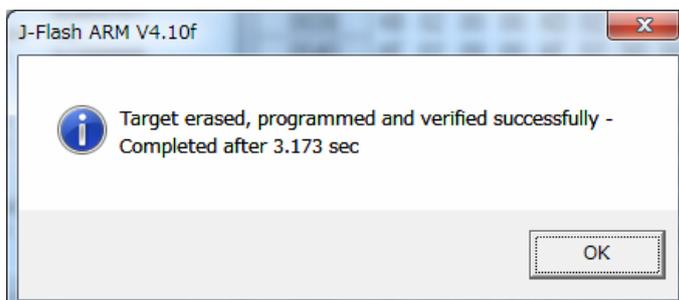


Target→Auto あるいは F7 で書き込み開始する。

書き込み中：



書き込み完了：



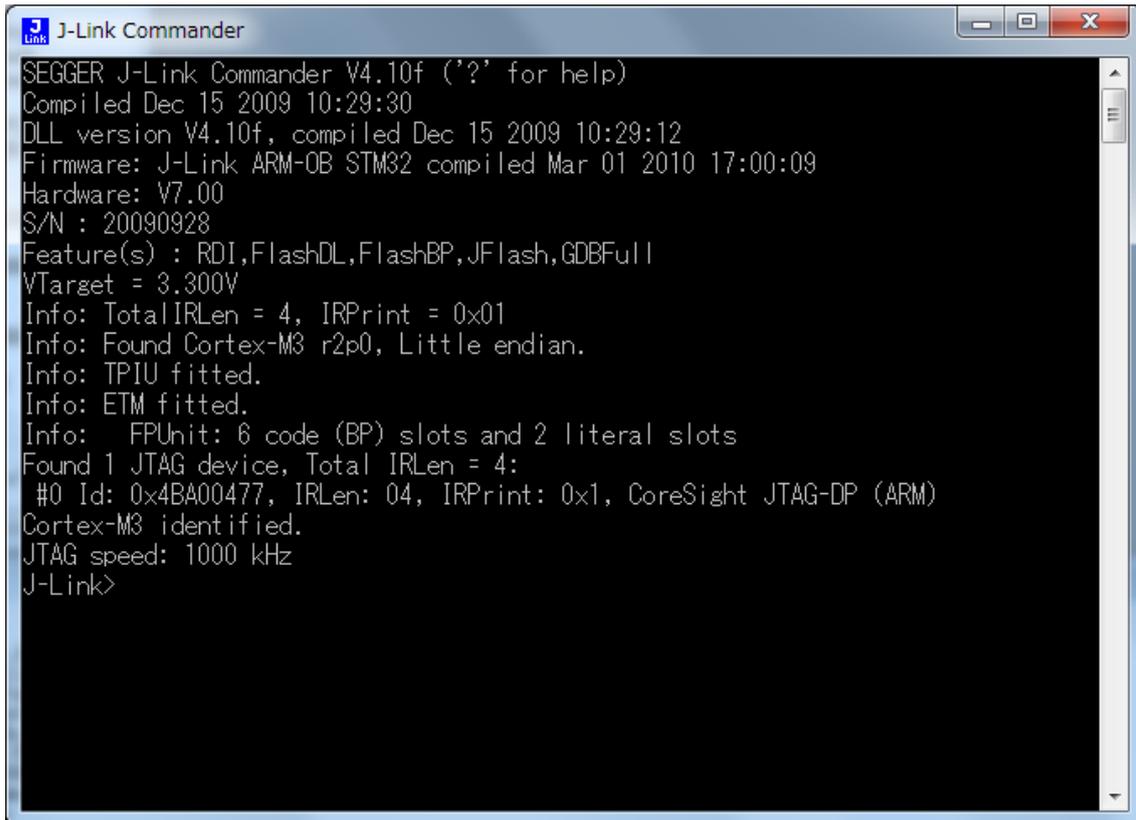
第四章 OpenLinkでデバッグ

使い環境のインストール手順は「3.3.1 ドライバのインストール」をご参照ください。

4.1 J-Link commandでデバッグ

コマンドラインでコマンドを入力して実行する。

J-Link command を起動すると、JLINK のバージョン情報が表示される。ターゲットを接続している場合は、ターゲットの状態と CPU などの情報が表示される。



```
SEGGGER J-Link Commander V4.10f ('?' for help)
Compiled Dec 15 2009 10:29:30
DLL version V4.10f, compiled Dec 15 2009 10:29:12
Firmware: J-Link ARM-OB STM32 compiled Mar 01 2010 17:00:09
Hardware: V7.00
S/N : 20090928
Feature(s) : RDI,FlashDL,FlashBP,JFlash,GDBFull
VTarget = 3.300V
Info: TotalIRLen = 4, IRPrint = 0x01
Info: Found Cortex-M3 r2p0, Little endian.
Info: TPIU fitted.
Info: ETM fitted.
Info:  FPUUnit: 6 code (BP) slots and 2 literal slots
Found 1 JTAG device, Total IRLen = 4:
 #0 Id: 0x4BA00477, IRLen: 04, IRPrint: 0x1, CoreSight JTAG-DP (ARM)
Cortex-M3 identified.
JTAG speed: 1000 kHz
J-Link>
```

J-Link command では豊富なデバッグ、検索などのコマンドを持っている。詳しい内容は J-Link command で ? を入力してエントリすると説明が表示される。

第五章 開発ツールKEILの応用

順番に MDKxxx.exe と rlarm401.exe をインストールする。

rlarm401.exe は RTX のカーネル、ファイルシステム、サンプルソースと LIB のパッケージ。

本章は MDK315B の例でインストール手順を説明します。MDK315B.exe は開発ツール KEIL の無償評価版です。評価版はプログラムにサイズの制限があります。

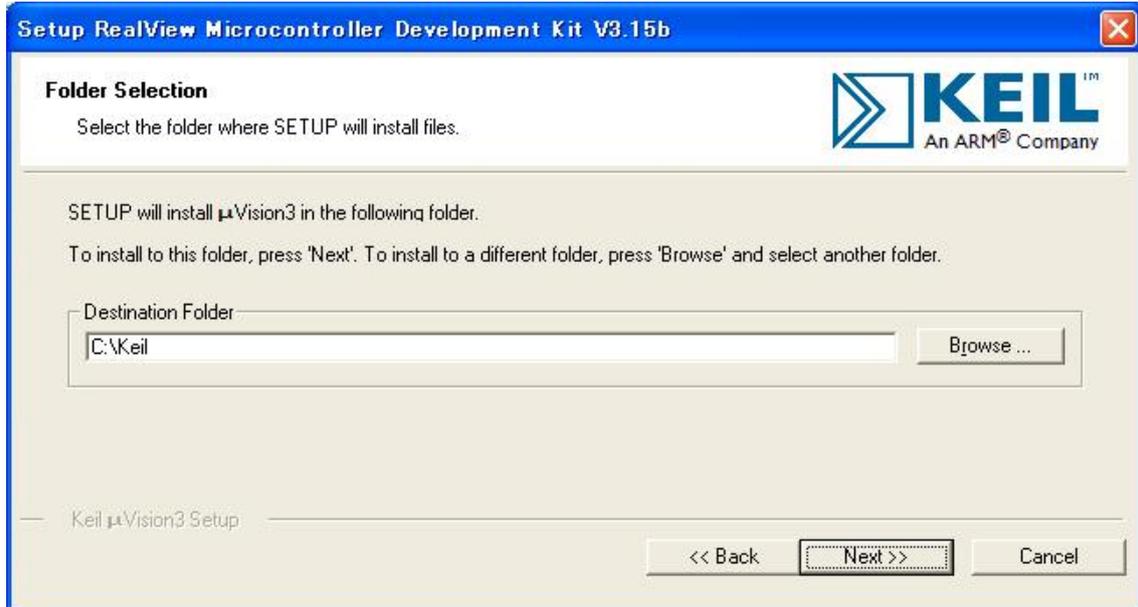
Keil社のHP (<http://www.keil.com/>) から最新版がダウンロード出来ます。

5.1 KEILのインストール

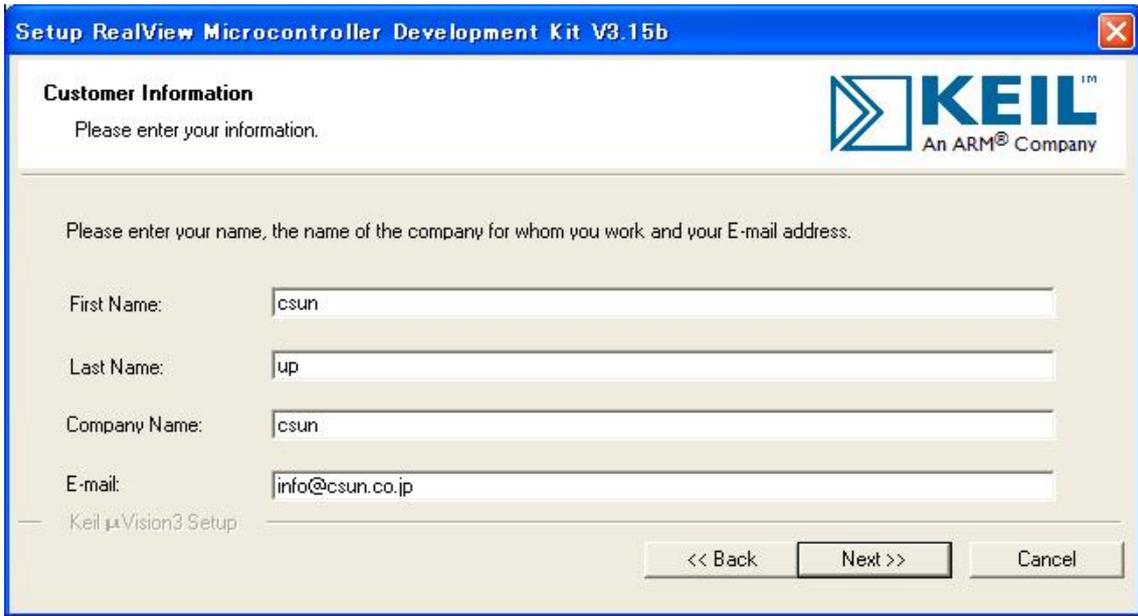
MDK315B.exe を実行して、KEIL3.15 をインストールする。



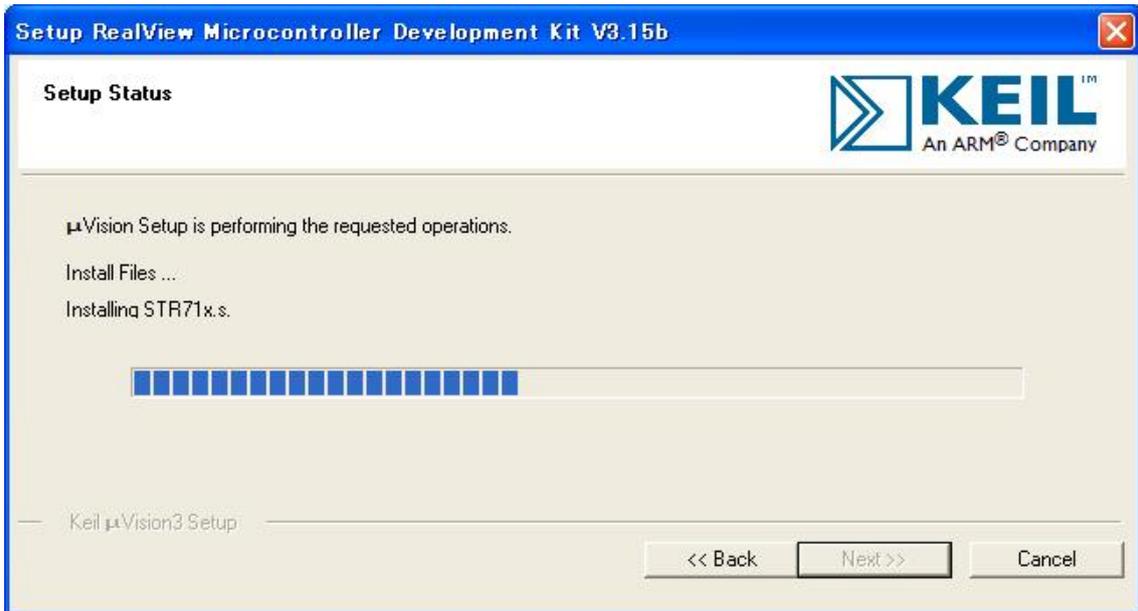
「Next」ボタンを押すと、英文のライセンス契約画面が表示される。同意できる場合は、「I accept the terms of the license agreement」を選択して、「Next」ボタンを押す。



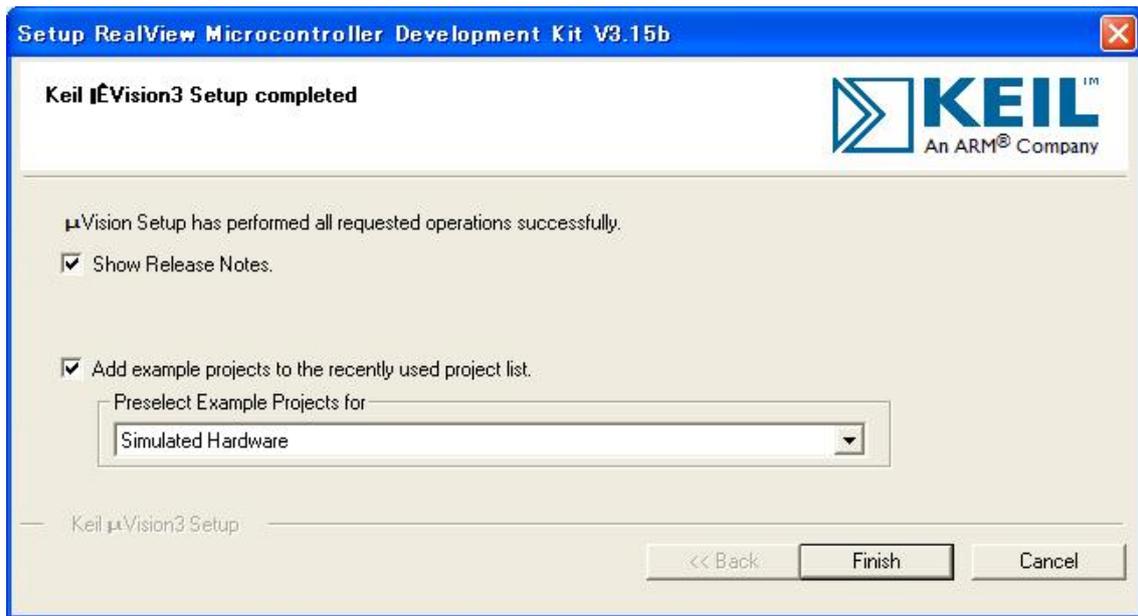
インストール先フォルダを変更せず、そのまま進んでください。



使用者の名前と所属会社名を入力するダイアログが表示される。名前は半角のアルファベットで入力してください。



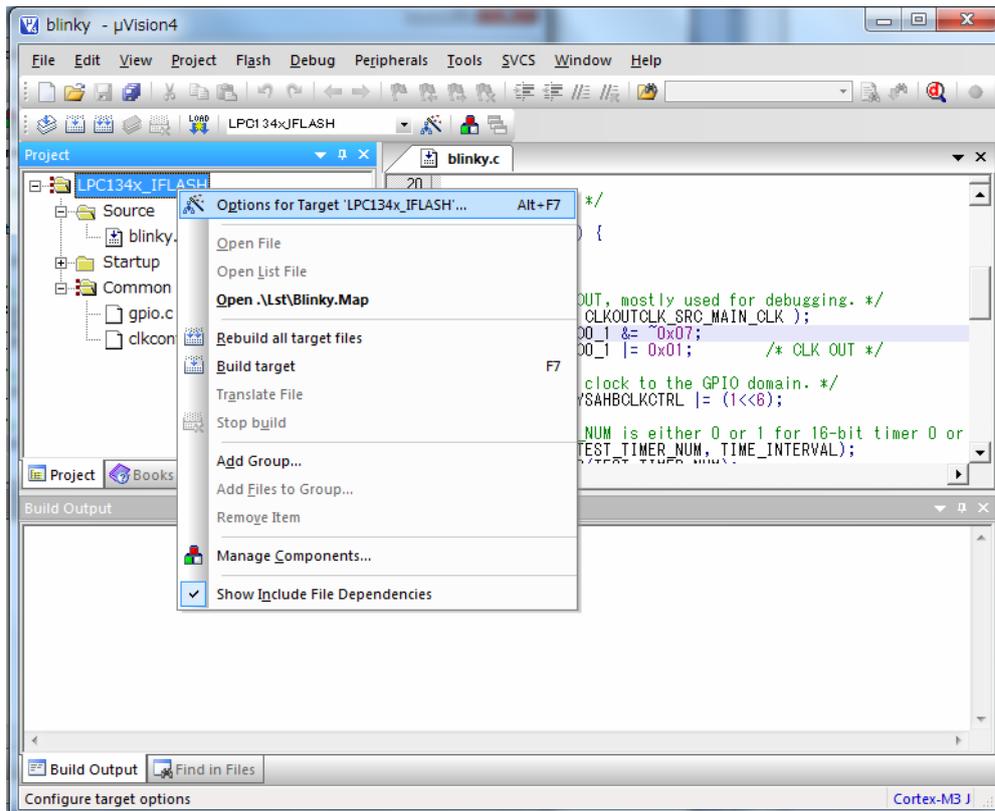
インストール中の画面です。



最後に「Finish」をクリックすると、ウィザードが閉じられてインストール終了。
デモ版ではライセンスがないので、プログラムのサイズ制限があります。ライセンスを取得するにはKeil社の日本代理店と連絡する事。

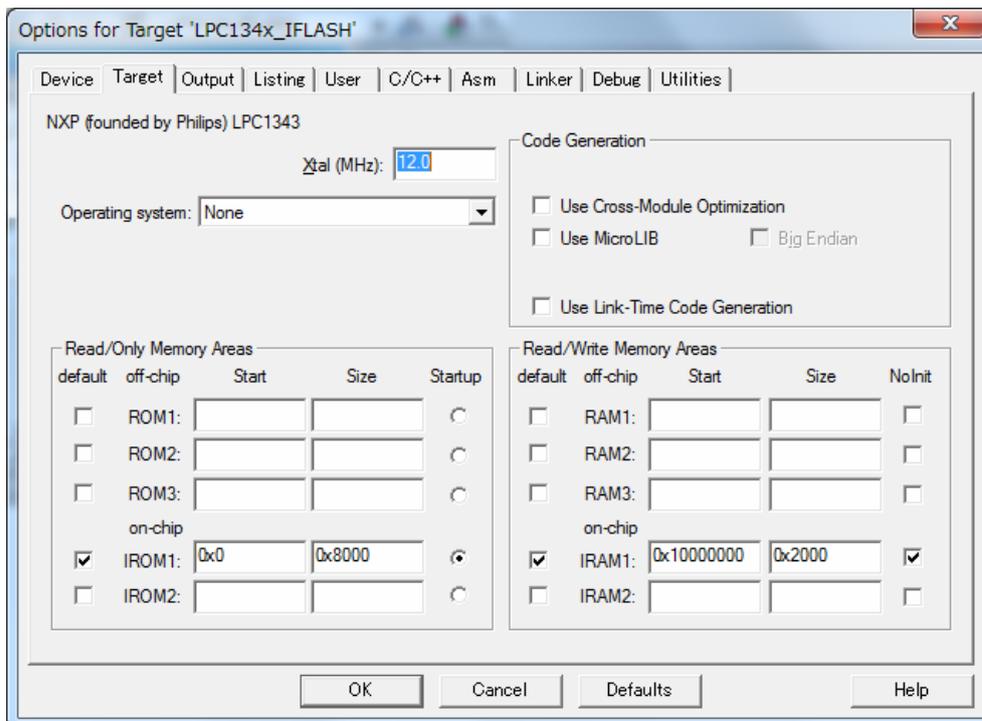
5.2 既存のプロジェクトから

プロジェクトファイルcode/MCB1000/Blinky/blinky.uvprojをダブルクリックする。或いはKEILのメニューでProject→Open Project…でblinky.uvprojを選択する。



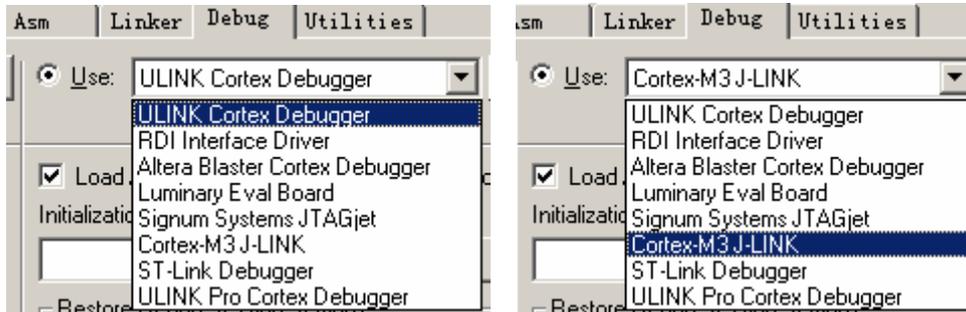
プロジェクト ADC を右クリックして「Options for Target ‘LPC134x_IFLASH’ …」をクリックする。

Options for Target ‘LPC134x_IFLASH’ の画面が出て来る。



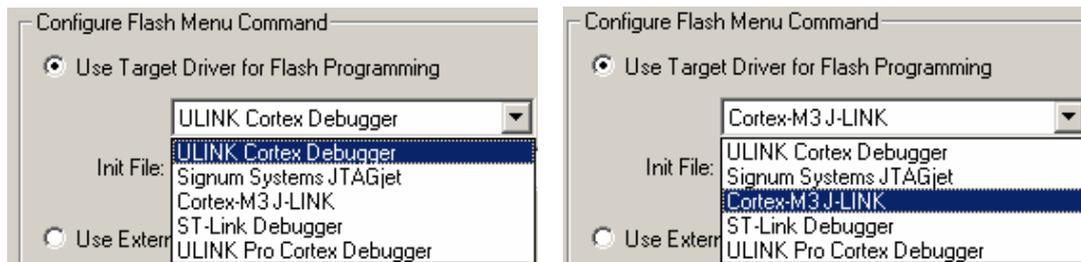
「Debug」タブを選択する。

ULINK2 でデバッグする場合は“ULINK Cortex Debugger”を選択し、JLINK V8 でデバッグする場合は“Cortex-M3 J-LINK”を選択する。

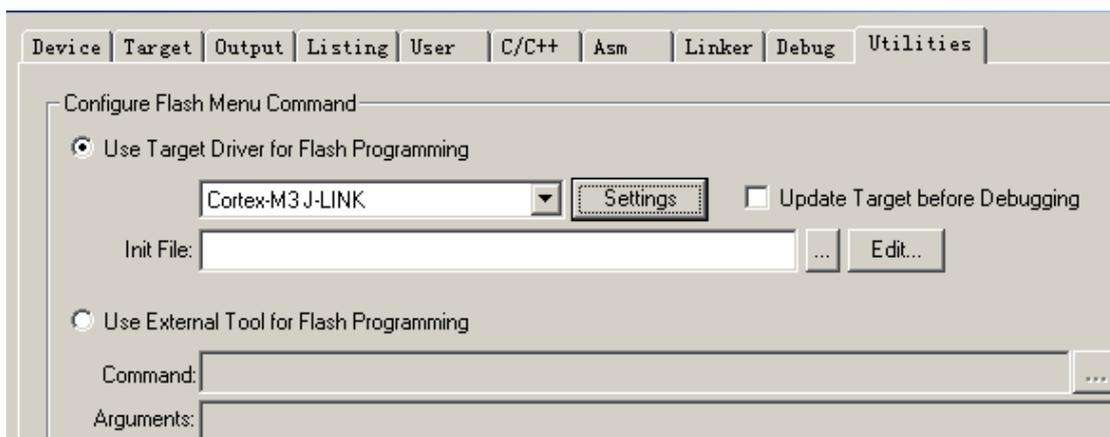


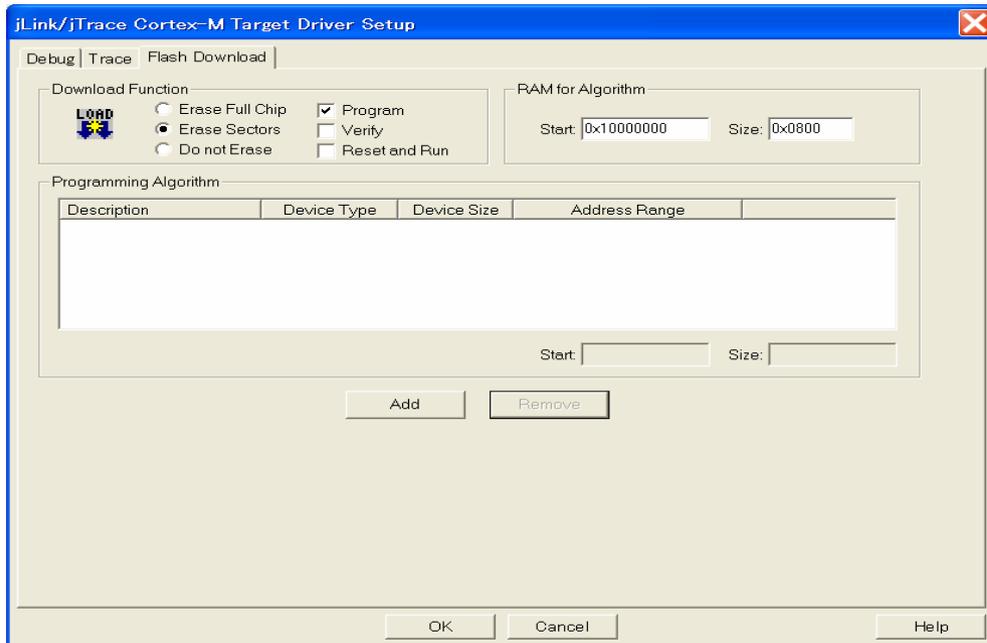
「Utilities」タブを選択する。

ULINK2 でデバッグする場合は“ULINK Cortex Debugger”を選択し、JLINK V8 でデバッグする場合は“Cortex-M3 J-LINK”を選択する。

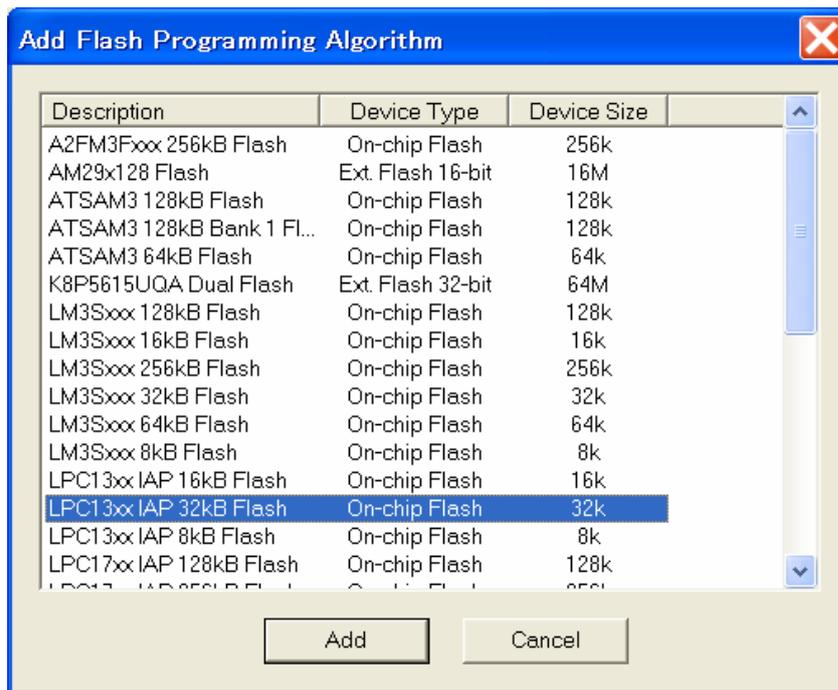


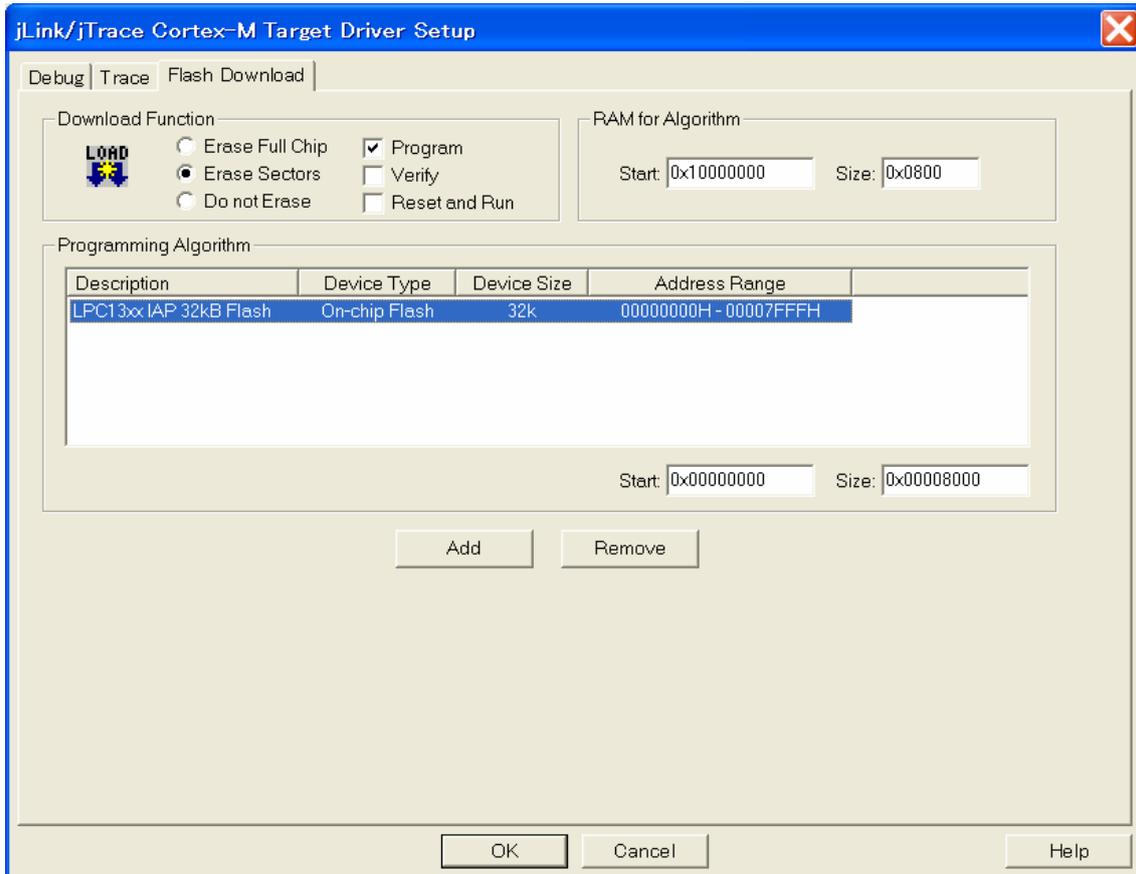
“Settings” をクリックして、デバッグのパラメータを設定する。JLINK V8 でデバッグする例で説明します。





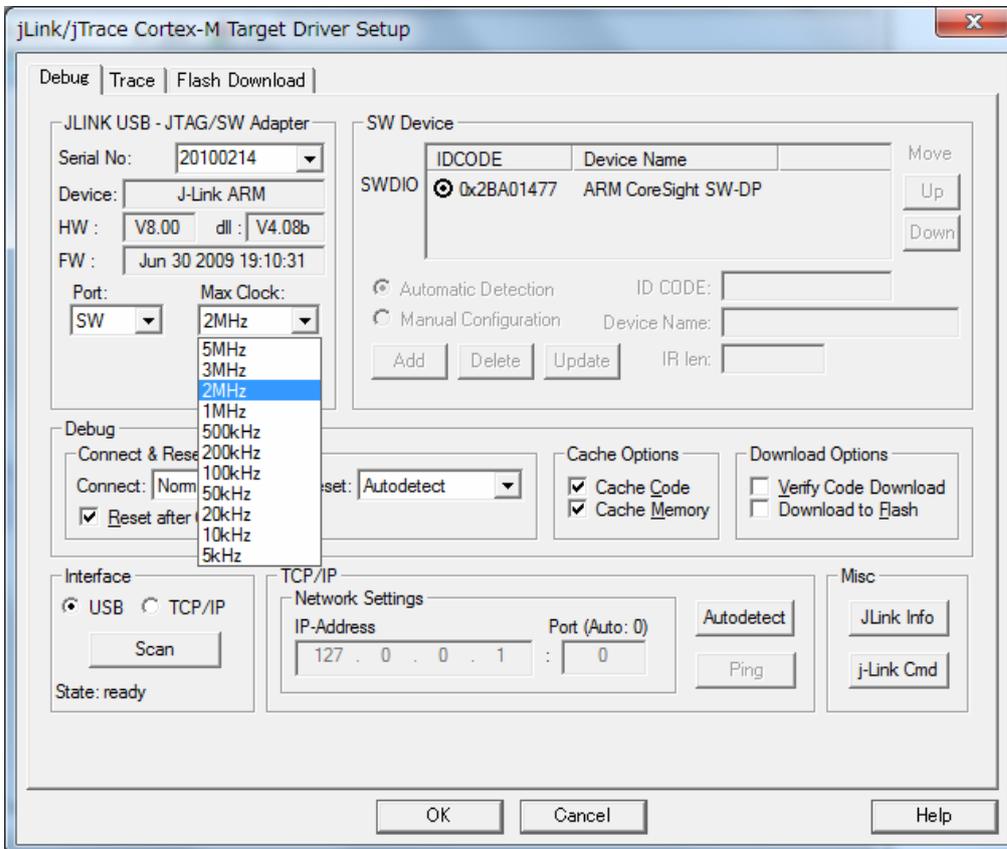
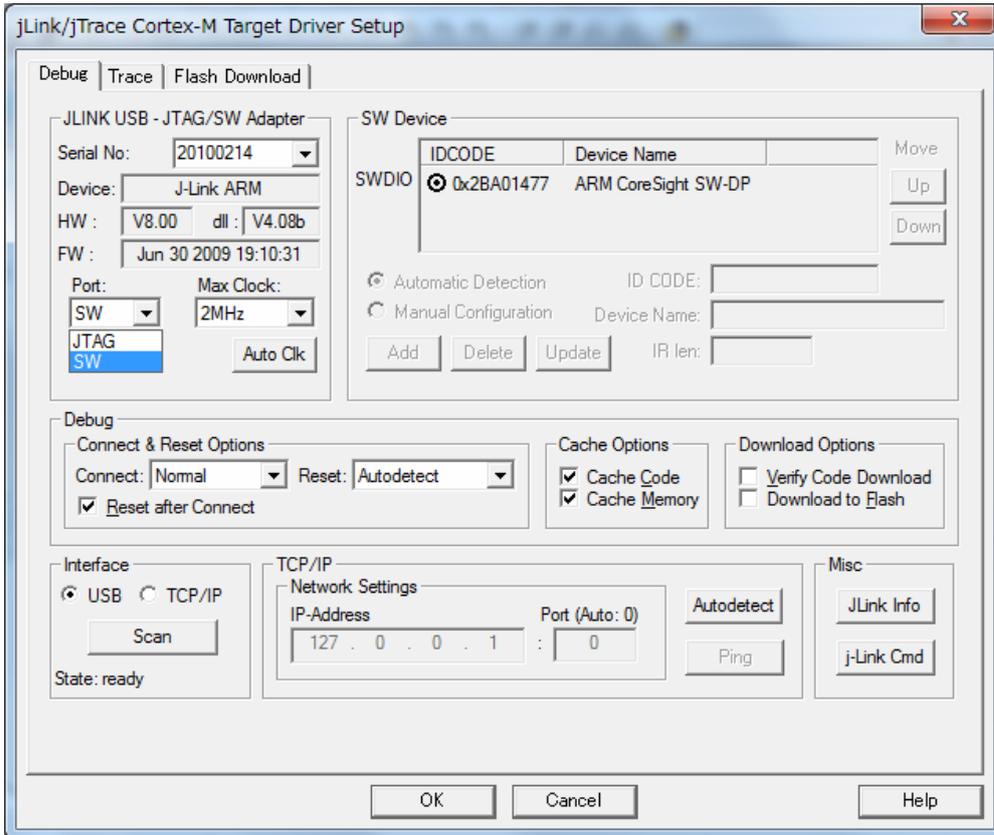
“Add” をクリックして追加する。



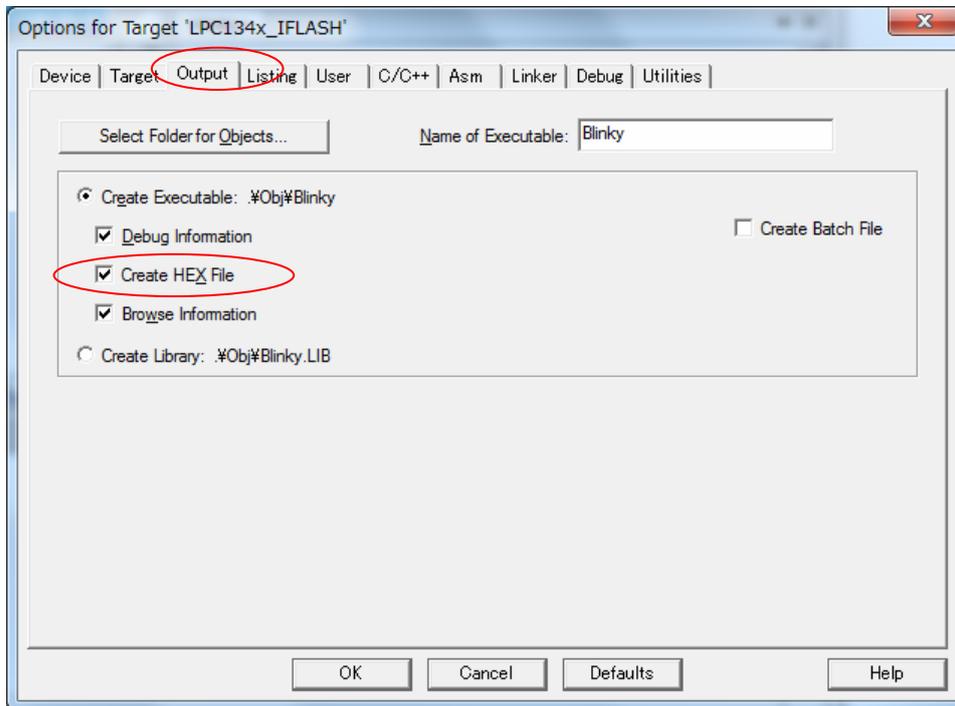


“Reset and Run” の所にチェック入れるとダウンロードしてからすぐ実行する。

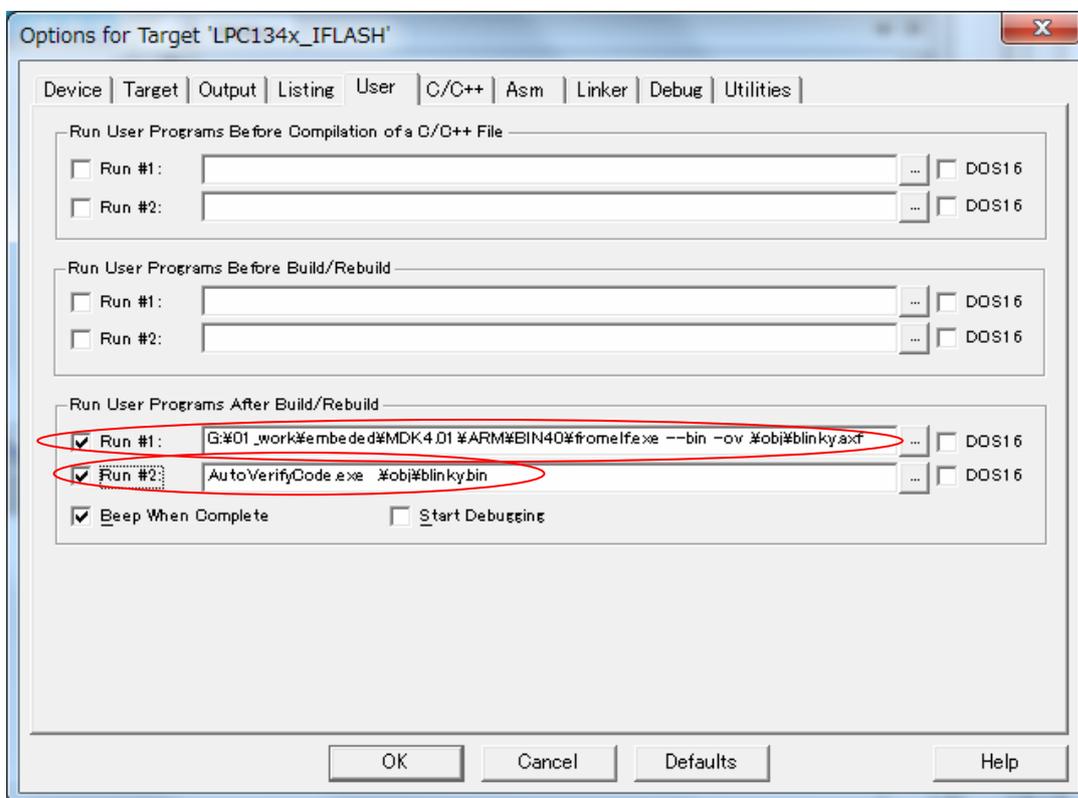
“Debug” タブをクリックして、エミュレータのモードを SW に設定する。またクロックはデフォルトでは 2M ですが、場合によって少し遅い速度に設定しても良い。



上記設定終わって、「OK」ボタンをクリックして、「Output」タブを選択して Create HEX File の所のをチェックを入れる。



最後に bin ファイル生成する必要な場合は「User」タブを選択する。

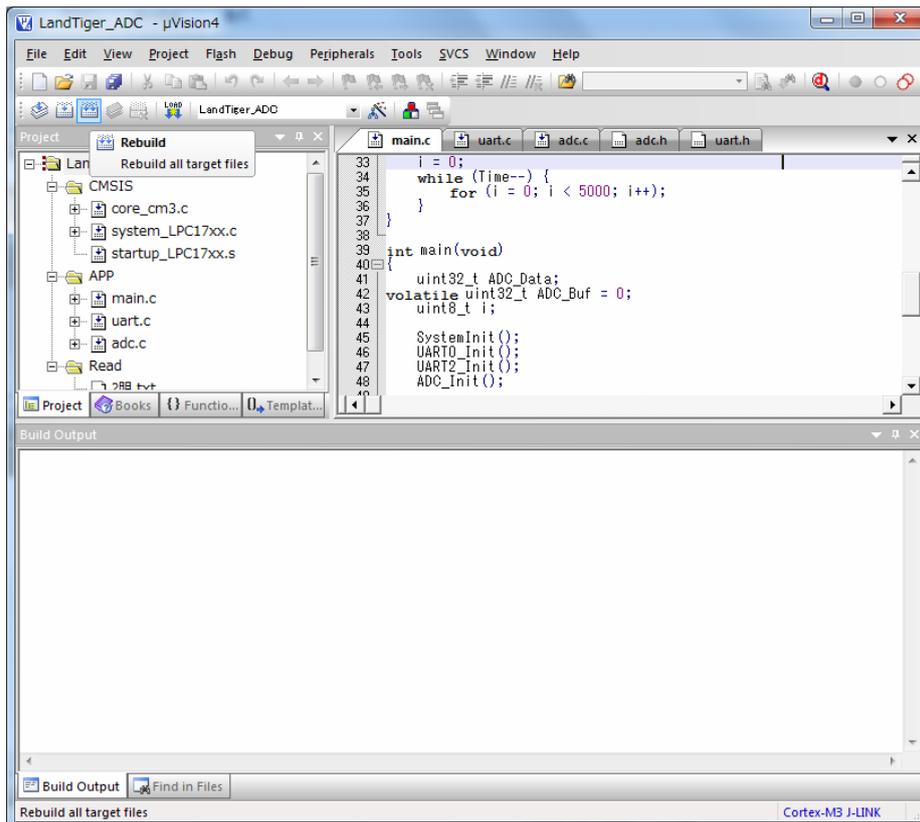


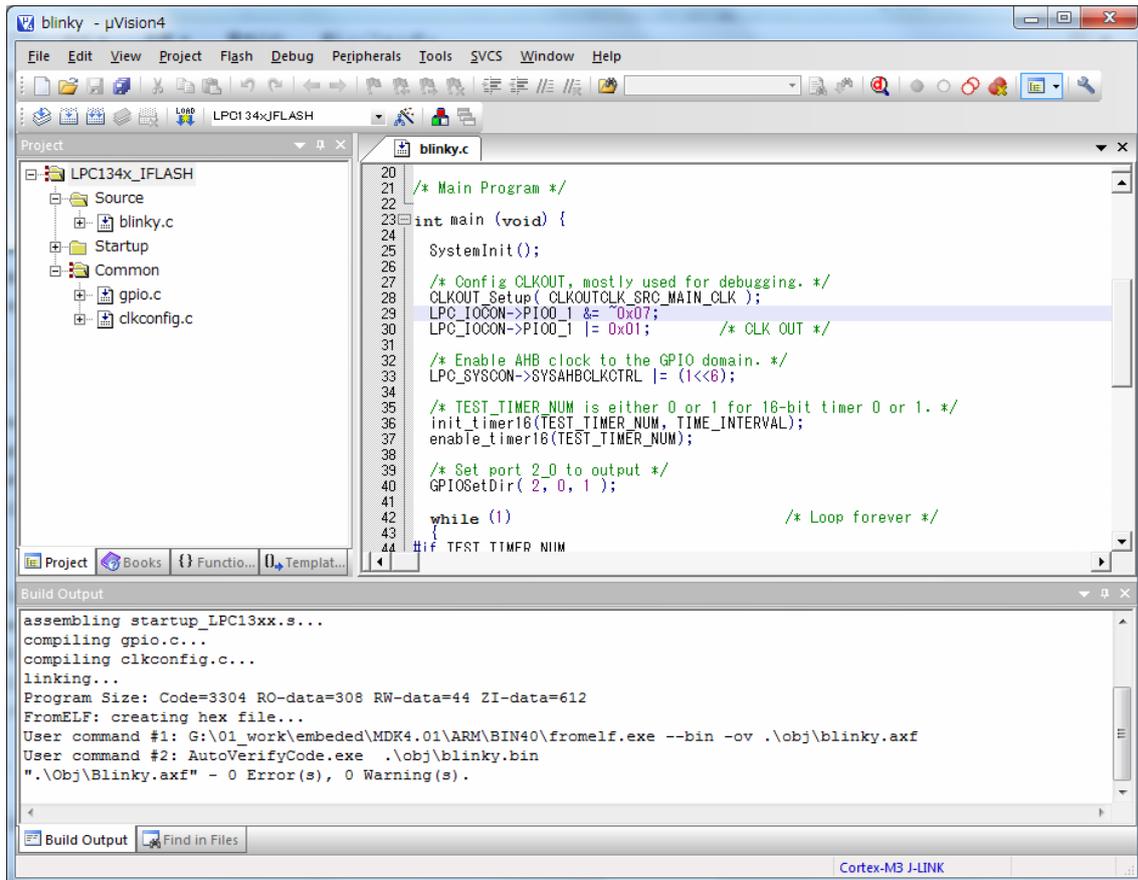
上記例の様に MDK の場所よりフォルダを指定する。

AutoVerifyCode.exe ファイルは弊社 HP の tools.zip から Blinky フォルダにコピーする。

“OK” ボタンをクリックする。

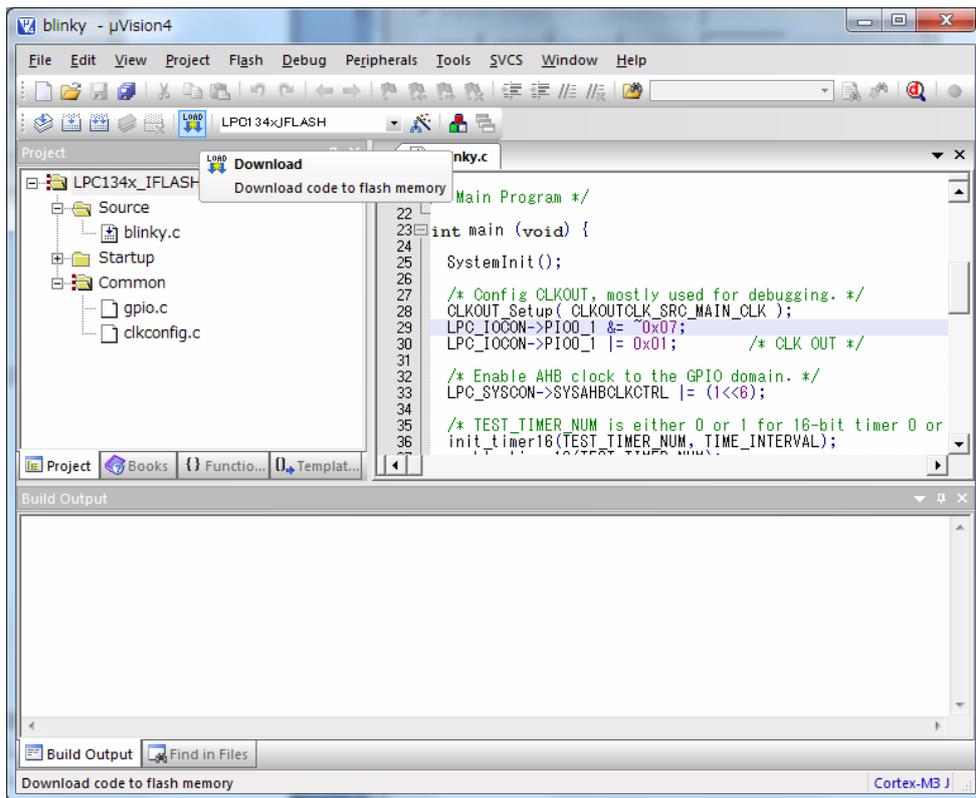
メイン画面に戻して、ツールバーの「Rebuild all target files」を押すと、ビルドが開始する。





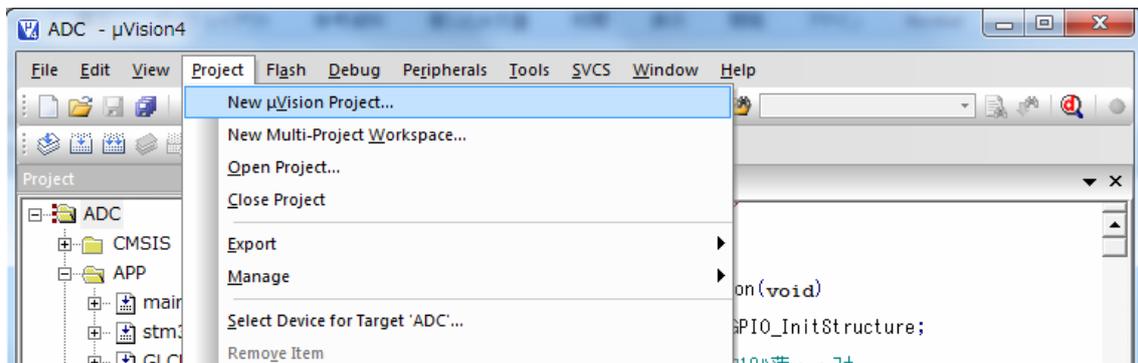
ビルドが成功したら、プロジェクトのobjフォルダにBlinky.hexファイルが生成される。このHEXファイルをボードに書き込む。

“Download to Flash Memory” でエミュレータでダウンロードが出来る。

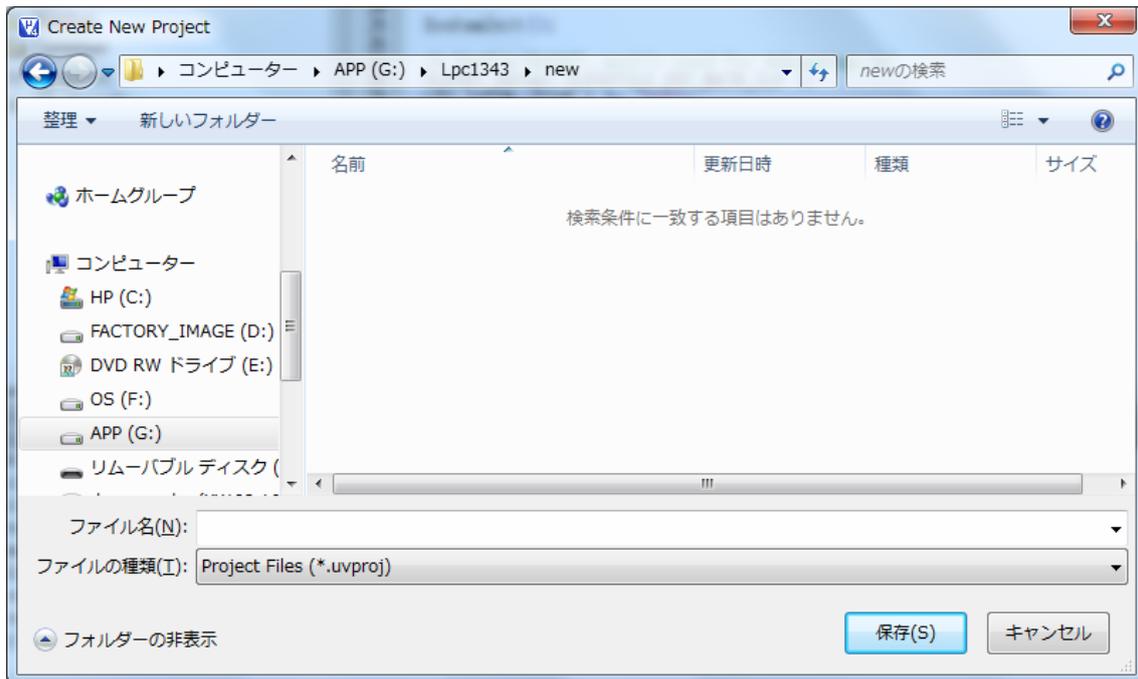


5.3 新しいプロジェクトの作成

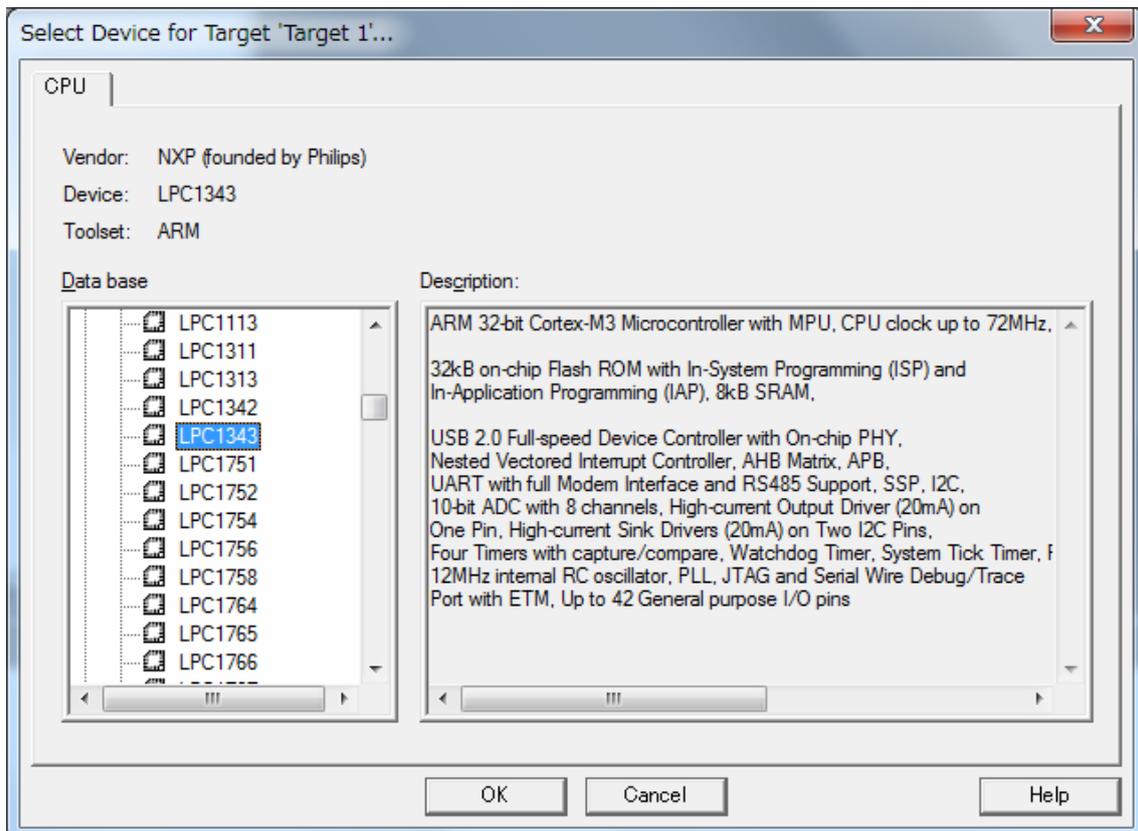
Keil のメニュー「Project」→「New uVision Project…」を選択する。



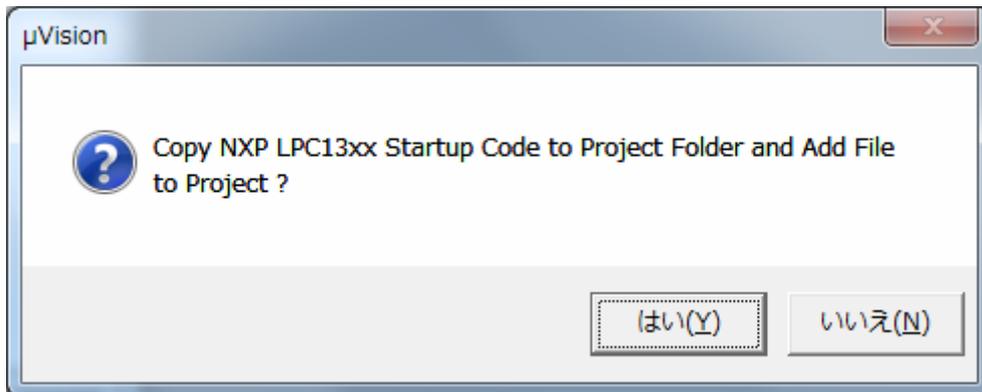
プロジェクト名を入力して、保存する。



CPU 選択画面が出て来る。選択肢 NXP を開いて LPC1343 を選択する。

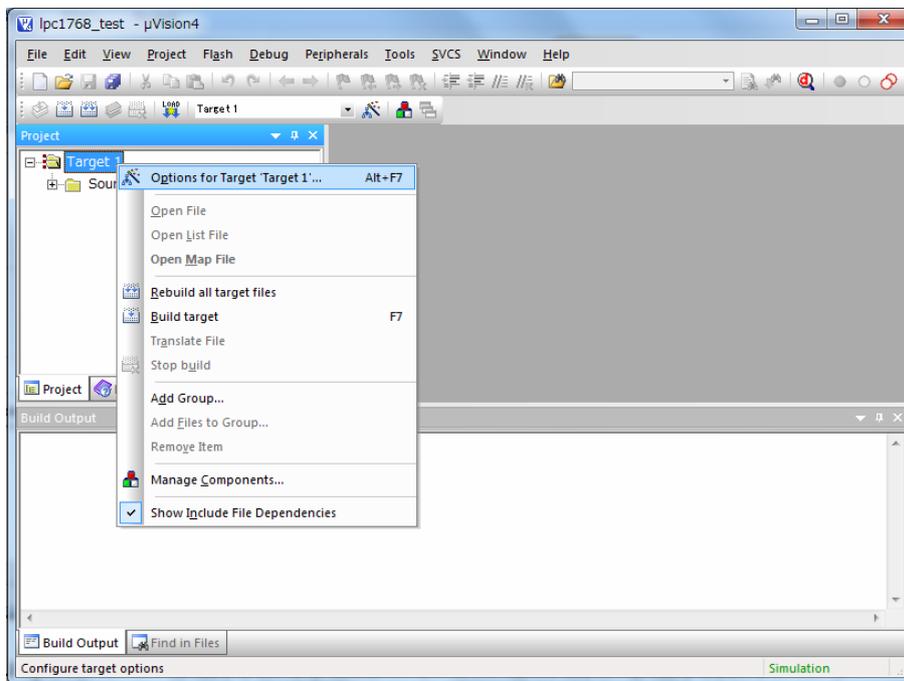


「OK」 ボタンをクリックすると下記画面が表示される。

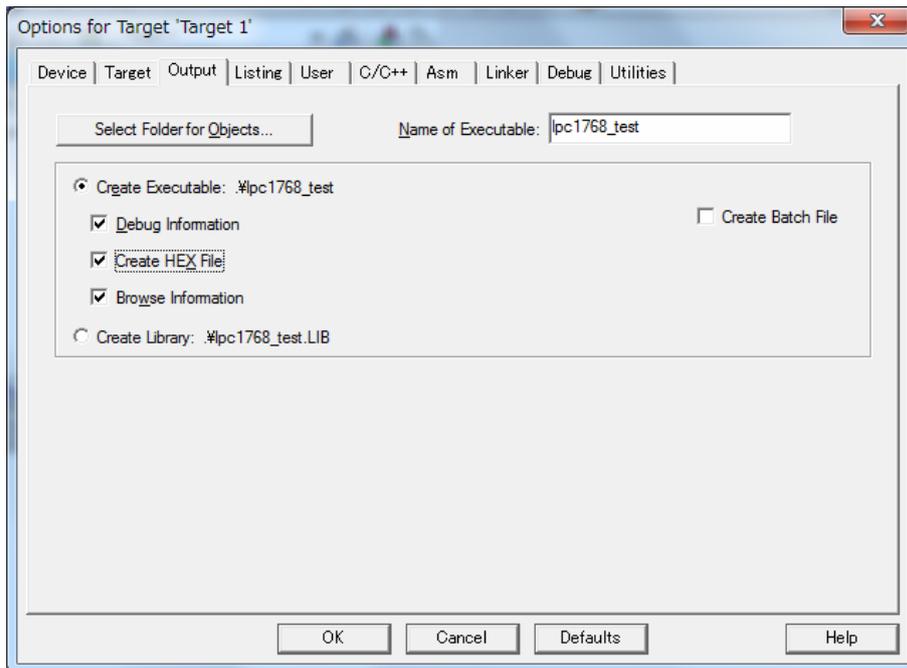


「はい (Y) 」 ボタンを押す。

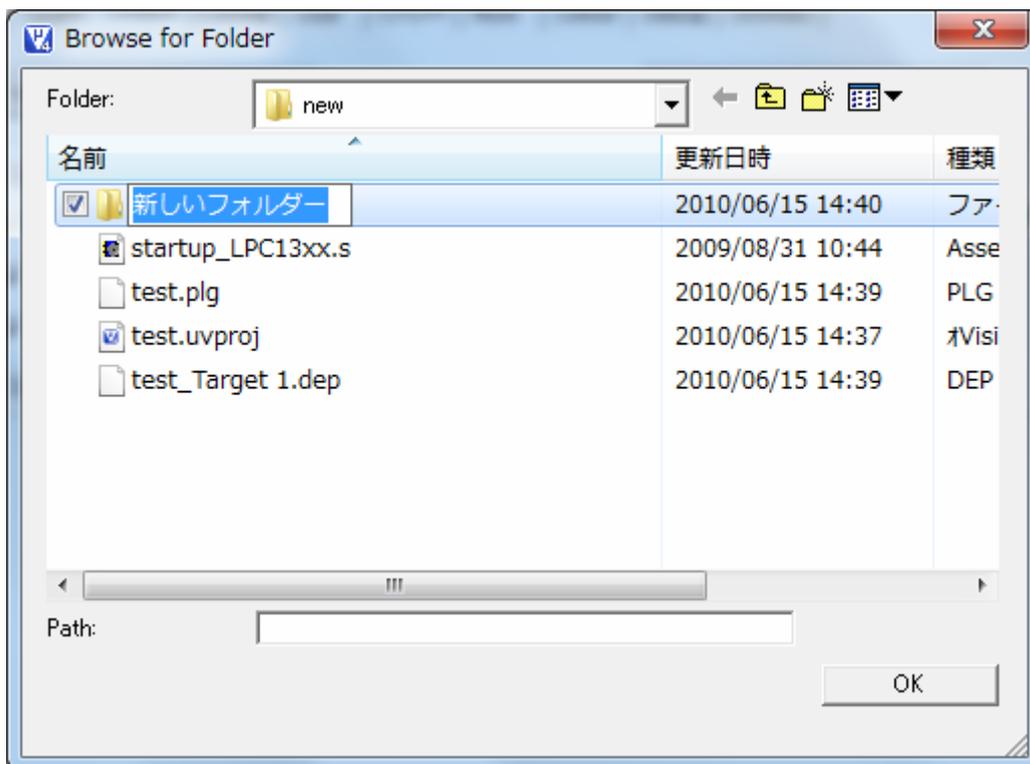
プロジェクトのオプションを設定する。



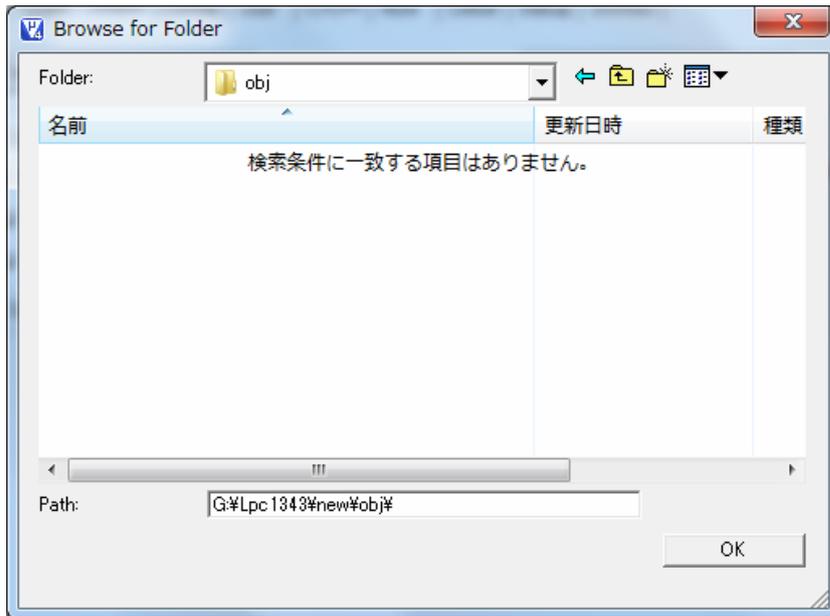
オプション設定画面で「output」タブを選択して、Hex ファイルを作成する選択肢にチェックを入れる。



上記画面で「Select Folder For Objects」ボタンを押して、出力フォルダを指定する。



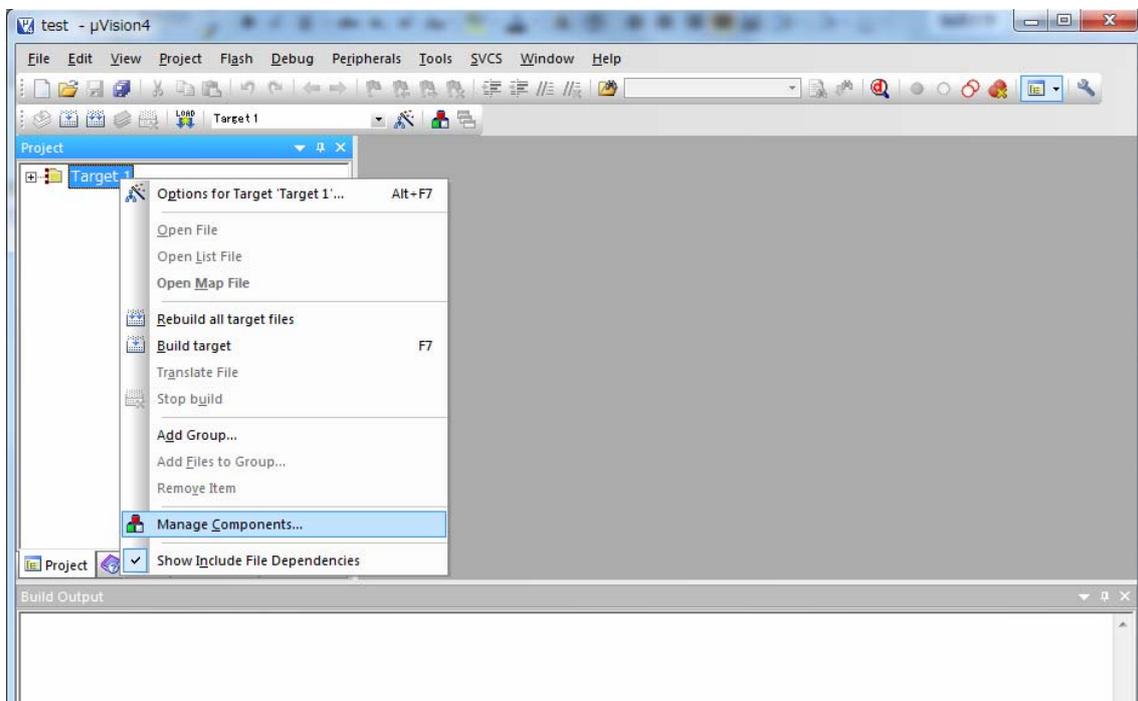
「obj」フォルダを作成して指定する。



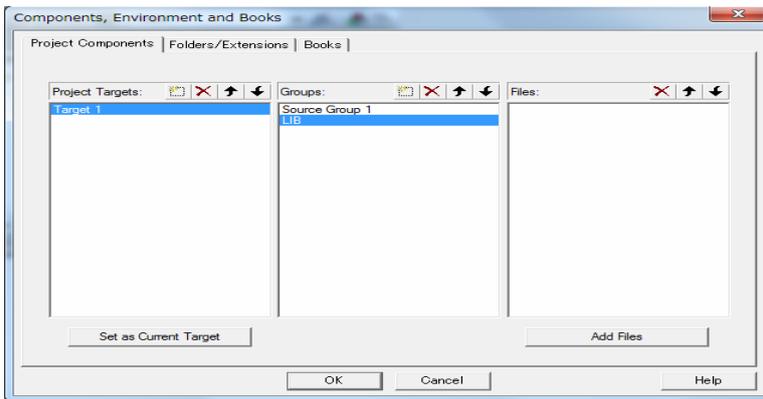
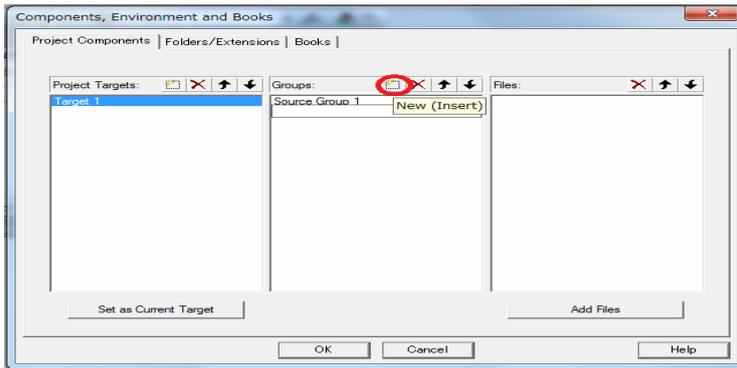
「OK」押してオプション設定画面に戻る。

「Listing」タブを選択して、上記と同じ手順でlistフォルダを作成する。
他の内容は5.2節をご参照ください。

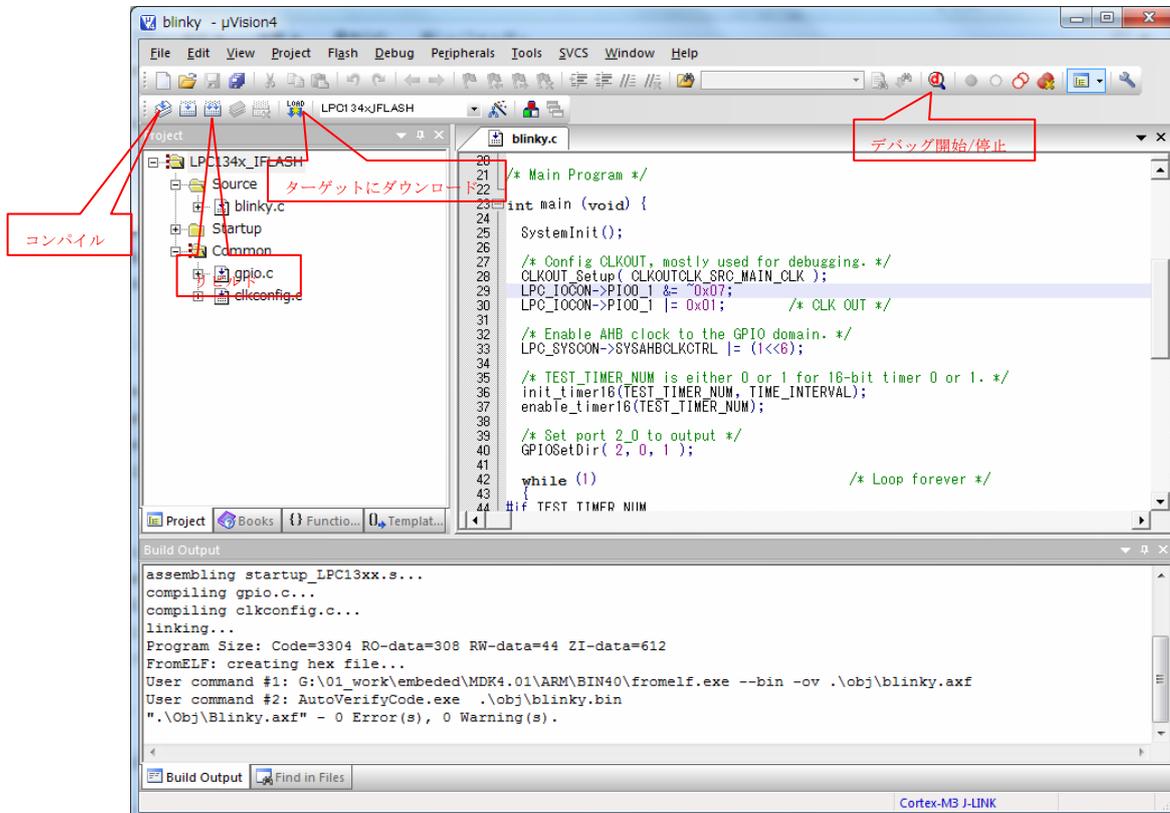
Target1でマウスを右クリックして”Manage Components”を選択する。



必要に応じてグループフォルダを追加する。LIB、APP など。

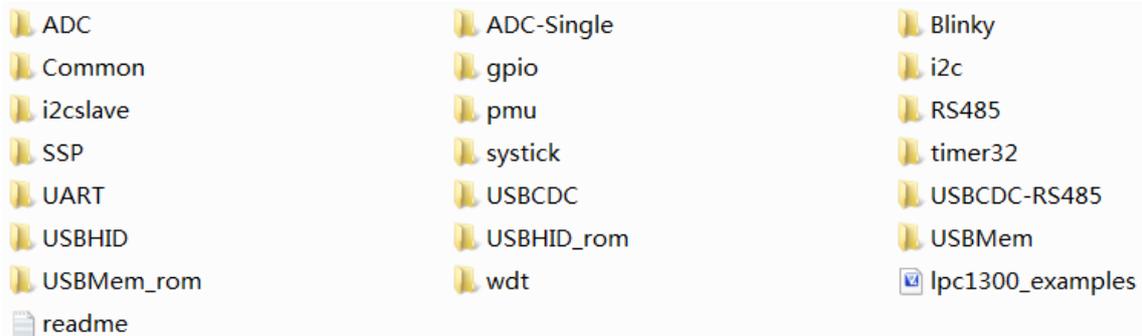


コンパイル、ビルド、ダウンロード、デバッグなどの操作。



第六章 サンプルソースの説明

6.1 CODE¥LPC13xx_SampleSoftware



6.1.1 CODE ¥LPC13xx_SampleSoftware¥Blinky

Blinky は 16bit タイマー0 或いは 1 で LED を点滅させる。

6.1.2 CODE ¥LPC13xx_SampleSoftware¥UART

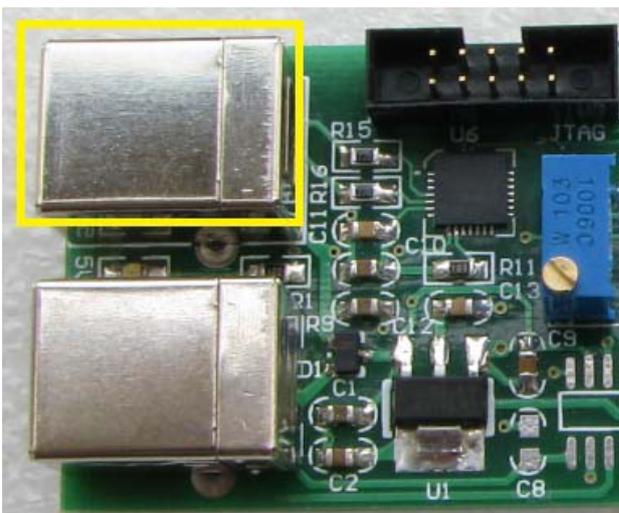
割り込みモードのシリアルテストプログラムです。シリアルテストソフトでボードに発送したデータをボードから返送されます。

受信したデータは UARTBuffer に保存され、UARTCount は計上します。メイン関数は UARTCount をチェックして 0 でない場合 UARTBuffer のデータをシリアルポートに発送します。

6.1.3 CODE¥LPC13xx_SampleSoftware.101¥ADC_Single

定期的な ADC7 チャンネルの AD ボリュームを観測して AD 値をシリアルポートで出力する。

1、ADC_Single¥obj¥adctest.hex をボードに書き込んで USB ケーブルで J1 と PC を繋ぐ。



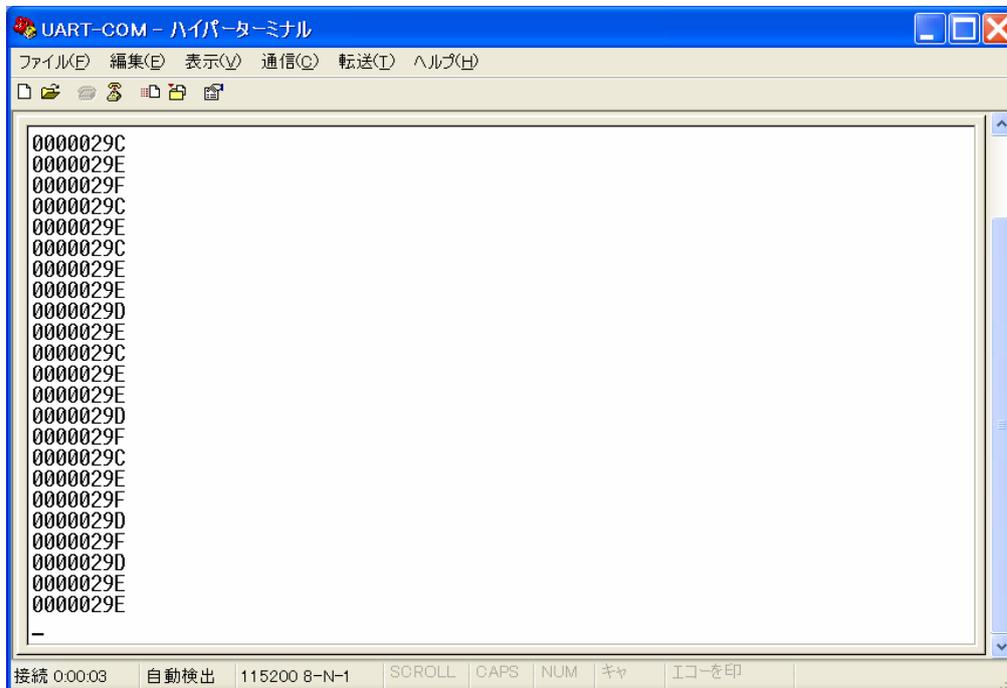
2、デバイスマネージャを開いて、USB から UART 変換のシリアルポートを確認する、例えば COM6。



3、PC側でハイパーミナルシリアルポート COM6、(115200(B)、8(D)、なし(P)、1(S)、なし(F))を開く。



4、ハイパーミナル端末に ADC7 の AD 値が表示される。



6.1.4 CODE¥LPC13xx_SampleSoftware.101¥URAT_RS485

1、標準 EIA-485 のサンプルソースです。“Normal Multidrop Mode” モードを利用しております。スレーブデバイスモードでデバイスアドレスは 0xC0。

2、本テストを実行するにはホストデバイスが必要です。ホストデバイス側で送データのフォーマットは：アドレス データ1 データ2 … データN

※ 8bit のデータを転送する時実際に物理層で 11bit を転送する必要、EIA-485 は parity ビットでアドレスとデータを区別します。

1 2 3 4 5 6 7 8 9 10 11

start bit0 bit1 bit2 bit3 bit4 bit5 bit6 bit7 parity stop

3、スレーブデバイス側は割り込みモードでデータを受信し、“UART RX line status” 割り込みをイネーブルします。データパリティはNONEに設定。ホストデバイスからアドレスを送るとき、スレーブデバイスは“Receive Line Status” 割り込みを発生します。

4、スレーブデバイスは自動方向コントロールに設定。

※本機能は少し難しいので、データシートの参考をお勧めします。

6.1.5 CODE¥LPC13xx_SampleSoftware.101¥i2c

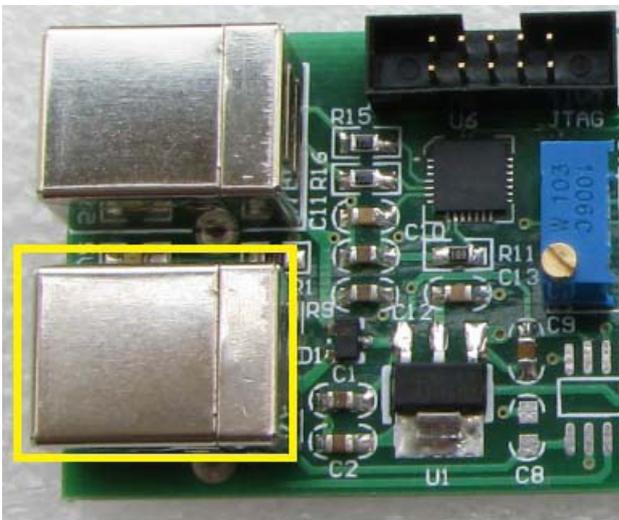
IIC EEPROM の 0-3 に 0x55、0xAA、0x12、0x34 を書き込む。書き込んだデータを読み出してシリアルポートから出力する。

6.1.6 CODE¥LPC13xx_SampleSoftware.101¥USBCDC

USB 仮想シリアルポートプログラム。本テストプログラムをボードに書き込んで、他の USBCDC ドライバ持っているデバイス (PC、ARM9/2440 ボードなど) と接続すると、新しいハードウェアを発見して自動的にドライバをインストールする。

USB 仮想シリアルポートは普通のシリアルポートと同じ様に通信できる。

1、.hex ファイルをボードに書き込んで、USB ケーブルで J1 と PC を繋ぐ。



2、デバイスマネージャを開いてシリアルポート番号を確認する。



3、もう1本のUSB ケーブルで J4 と PC を繋ぐ。同じ様にデバイスマネージャでシリアル番号を確認する。

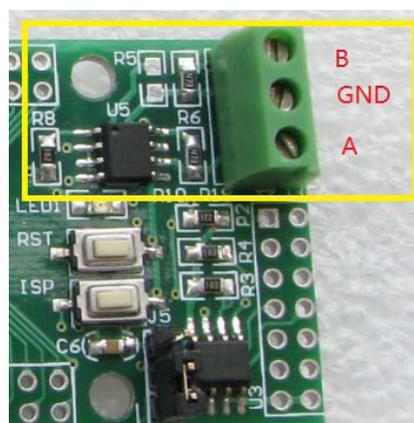
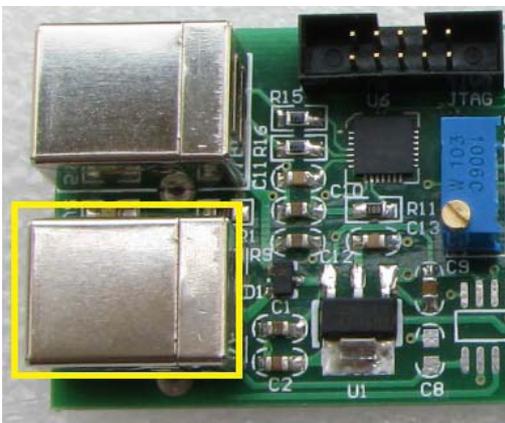


4、ハイパーターミナルで Com6 と Com7 を開いて、同じ設定で開くと、お互いに送信できる。



6.1.7 CODE¥LPC13xx_SampleSoftware.101¥USB CDC_RS485

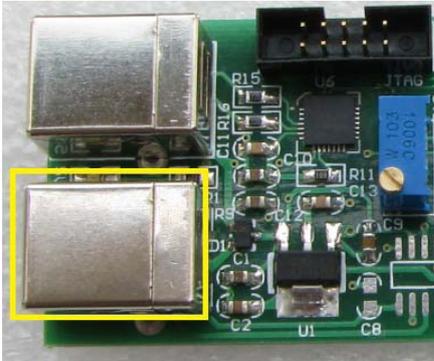
USB から RS485 への変換機能を実現します。機能は上記サンプルと同じで、もう一つの RS485 デバイスを繋いでテストする必要があります。



6.1.8 CODE¥LPC13xx_SampleSoftware.101¥USBMem

ボード内蔵 RAM 領域を USB メモリデバイスとして利用する。

- 1、USBMem¥obj¥Memory.hex をボードに書き込む。
- 2、USB ケーブルでボードの J1 と PC を接続する。PC 側で USB メモリ設備として認識する。



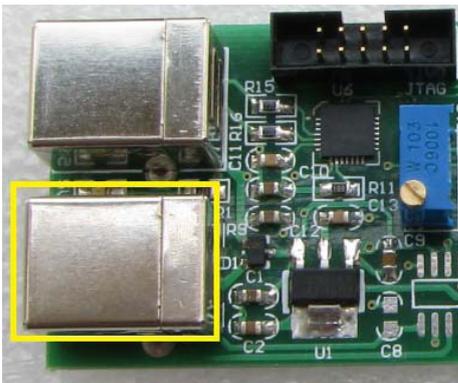
- 3、移動デバイスを開いて見ると、Readme.txt ファイルがある (windows7 ではドライブしない)。

6.1.9 CODE¥LPC13xx_SampleSoftware.101¥USBHID

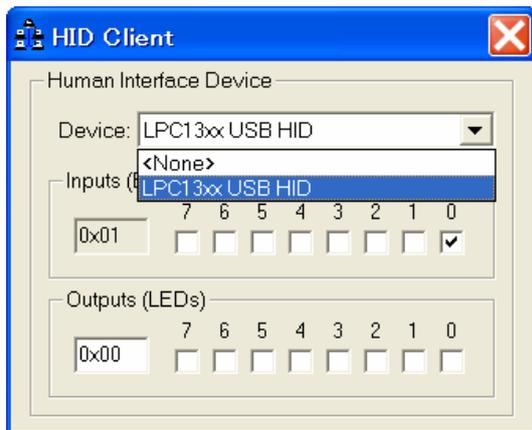
ボードを HID デバイスとして利用する。

¥Code¥HID_Client¥Release にある HIDClient.exe でテストできる。

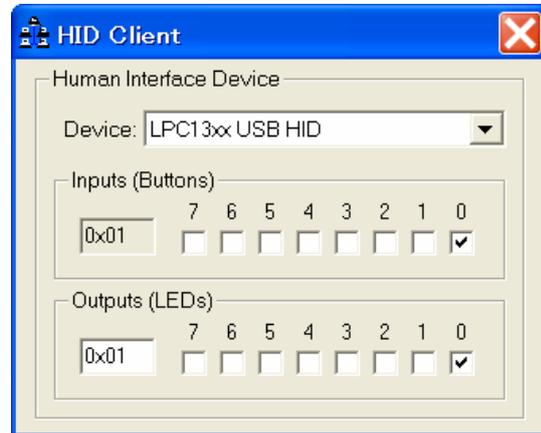
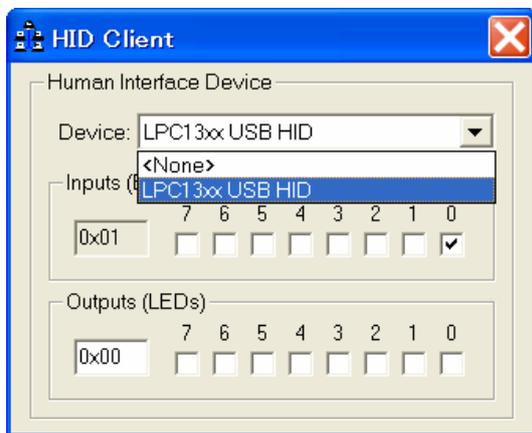
- 1、USBHID¥obj¥HID.hex をボードに書き込む。
- 2、USB ケーブルでボードの J1 と PC を接続する。PC 側で HID 設備として認識する。また LED1 が点滅する。



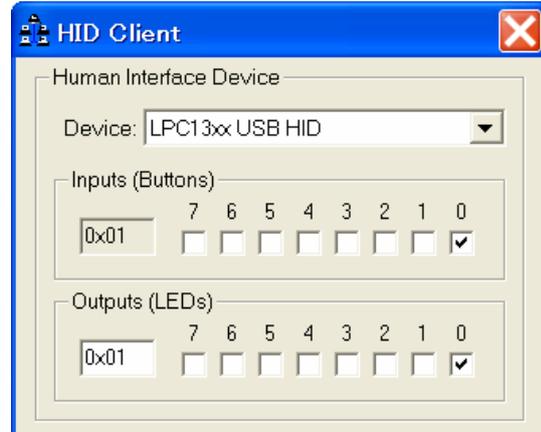
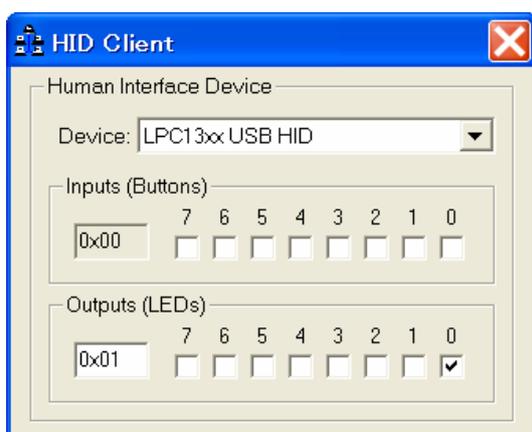
- 3、HIDClient.exe を実行する。Device の選択肢に “LPC13xx USB HID” がある。これを選択する。LED1 が消灯する。



4、Outputs(LEDs)にチェックを入れ/消すで、ボード上のLEDを点灯/消灯させる。



5、ボード上の“ISP”キーを押したら“Inputs(Buttons)”の0の所にチェックが消えて、離すとチェックが表示される。



6.1.10 CODE¥LPC13xx_SampleSoftware.101¥USBxxx_rom

USBxxx_romはUSBxxxと同じ機能で、メリットとしてはUSBBootloaderのAPI関数を利用し



てFlashメモ리를節約した事。

以上。