

ARM7/TDMI LPC2148 Ethernet ENC28J60

モジュール

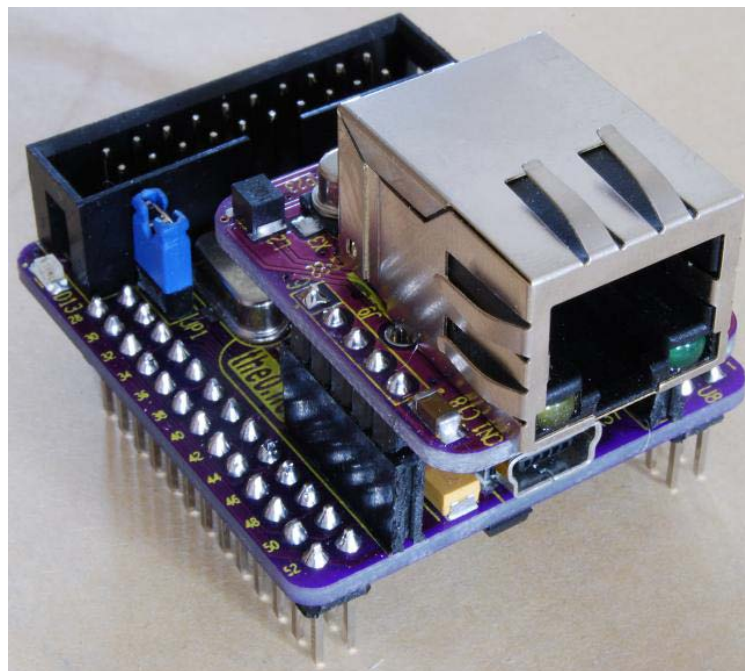
マニュアル

株式会社日昇テクノロジー

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

info@csun.co.jp

2009/12/28



[copyright@2009](http://www.csun.co.jp)



修正履歴

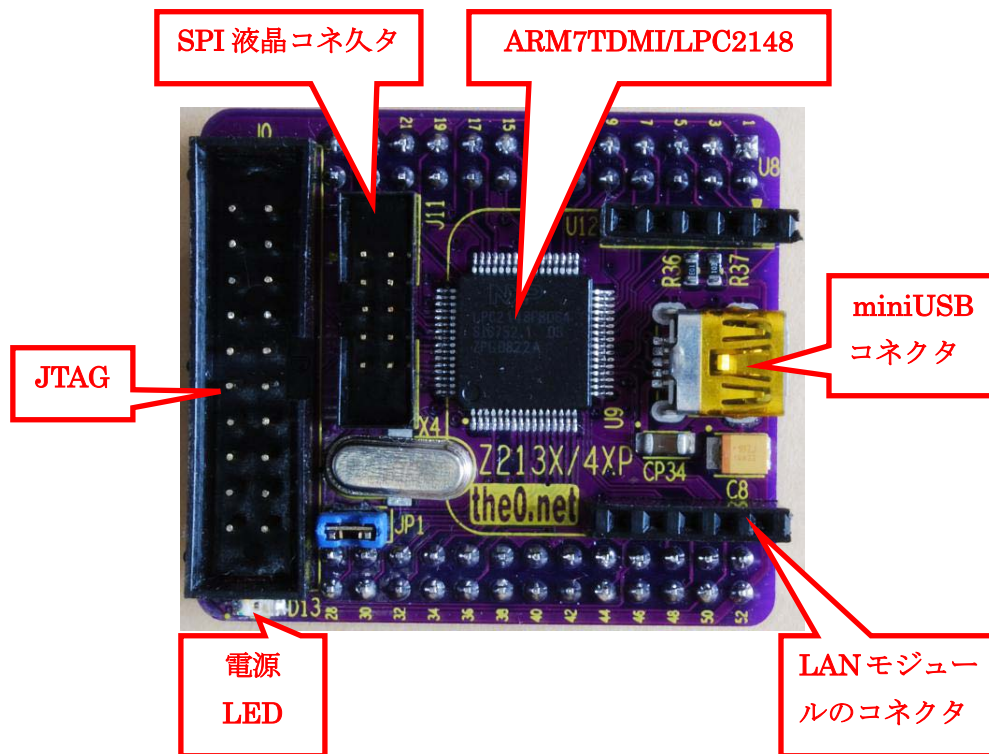
NO	バージョン	修正内容	修正日
1	Ver1.0	新規作成	2009/3/24
2	Ver1.1	USB ダウンロード方法修正	2009/12/6
3	Ver1.2	USB ダウンロード方法追加	2009/12/27



第一章 ARM7TDMI/LPC2148 モジュールの概要.....	4
第二章「LPC214x」シリーズの概要.....	6
第三章 USB ダウンローダ.....	8
3.1 出荷状態.....	8
3.2 ユーザ・プログラムを書き換える手順.....	9
3.2.1 OpenJTAG を利用し書き込む.....	9
3.2.2 Open Link を利用し書き込む.....	13
第四章 開発ツール KEIL のインストール.....	14
4.1 KEIL のインストール.....	14
4.2 既存のプロジェクトから.....	16
第五章 SPI イーサネットモジュール ENC28J60	19
第六章 SPI グラフィック液晶.....	22

※ 使用されたソースコードは<http://www.csun.co.jp/>からダウンロードできます。

第一章 ARM7TDMI/LPC2148 モジュールの概要



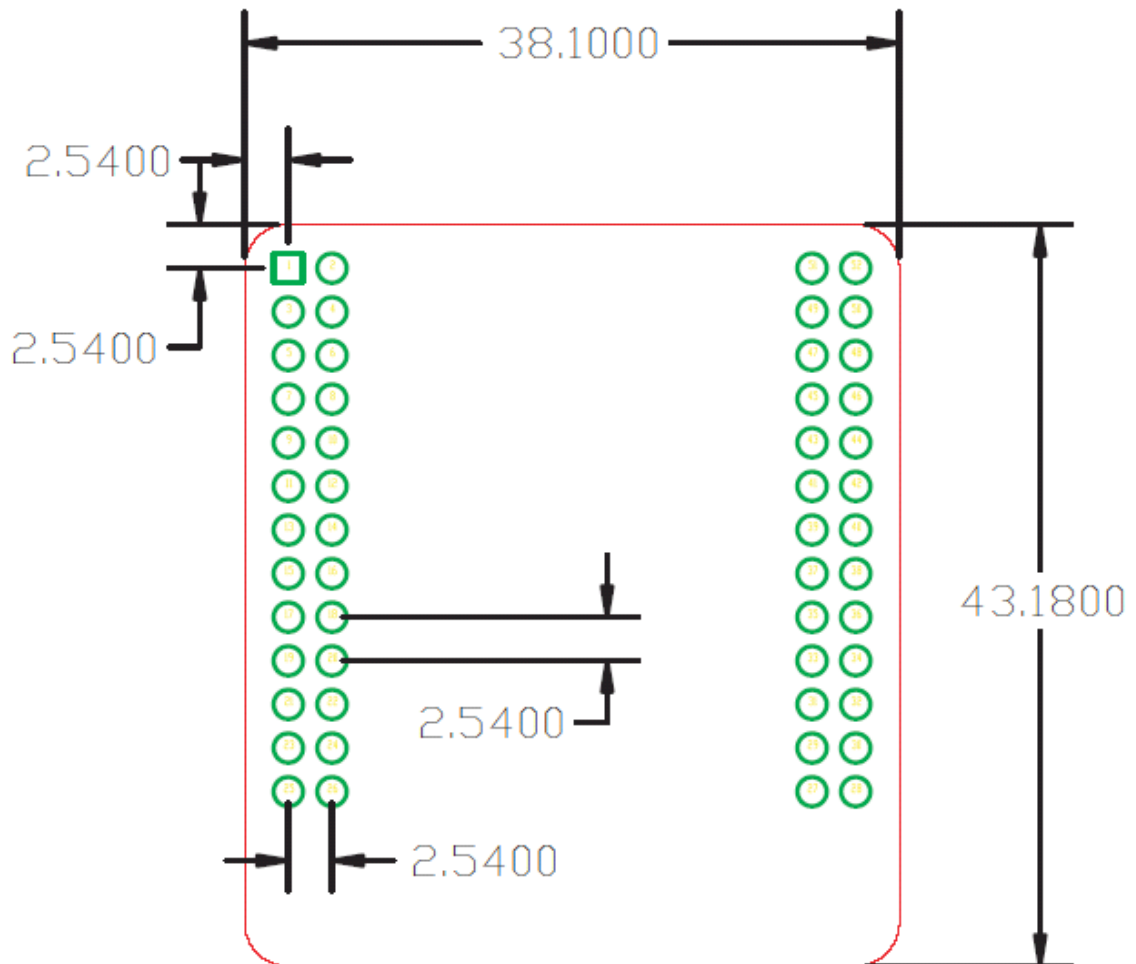
CPU プロセッサ

- ARM7TDMI-S コアを採用した NXP 社の LPC2148(周波数 60MHz,512KB FlashROM, 42KB SRAM, 2×SPI, 2×I2C, USB2.0 デバイス, PWM, 2×10 ビット ADC 4ch, 10 ビット DA×1, 2×USART, 2×32 ビット・タイム, RTC)。
- I/O ポートは 5V トレラントのため、5V 系ロジックを直結できます。

モジュールのインターフェース

- 3.3V レギュレータ内蔵
- サブクリスタル付 (32.768kHz)
- 20P 標準 JTAG 端子
- miniUSB コネクタ USB2.0 device ×1
- EasyNIC(ENC28J60) を直結できます。
- SPI 液晶コネクタ×1

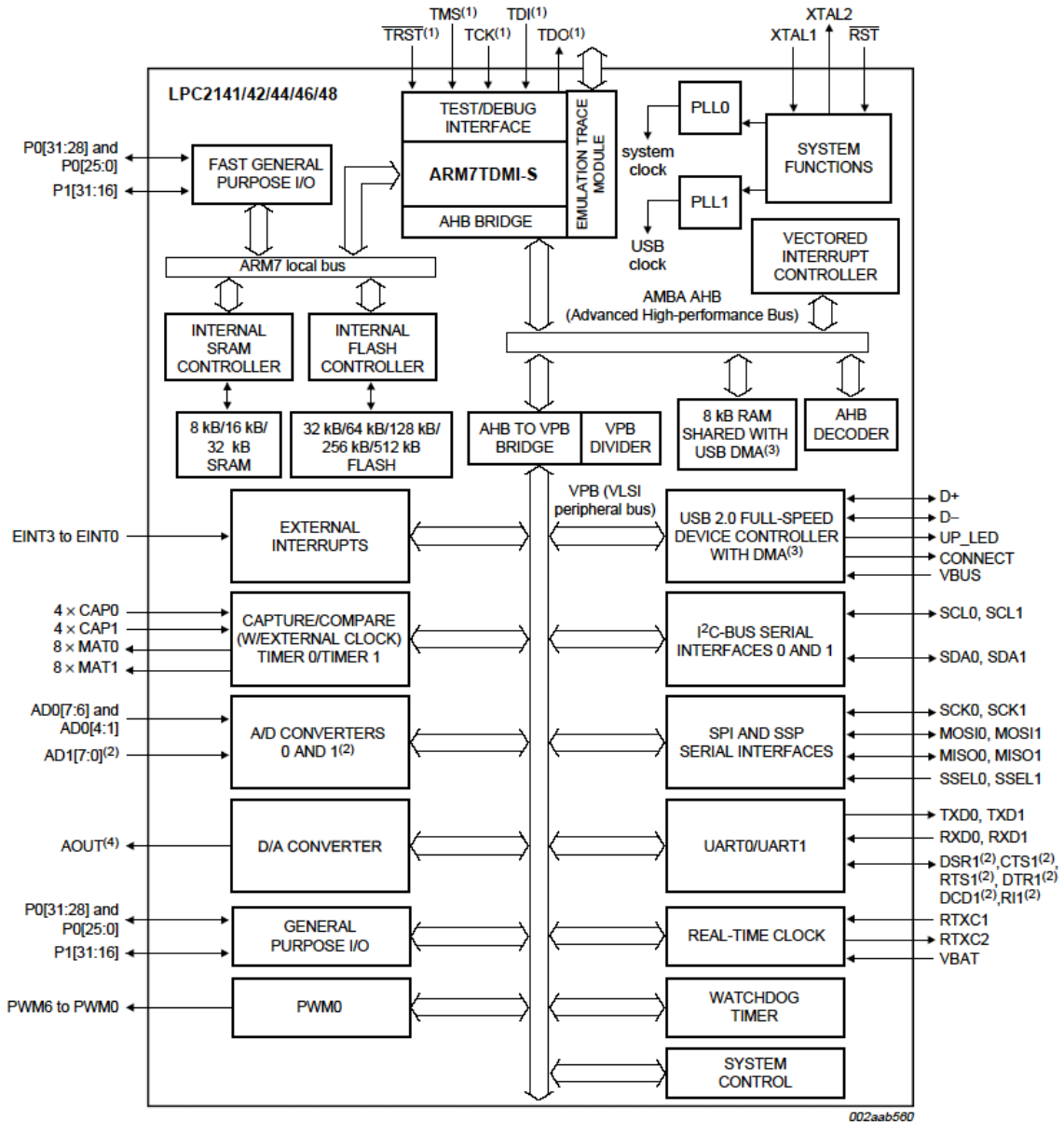
外形寸法

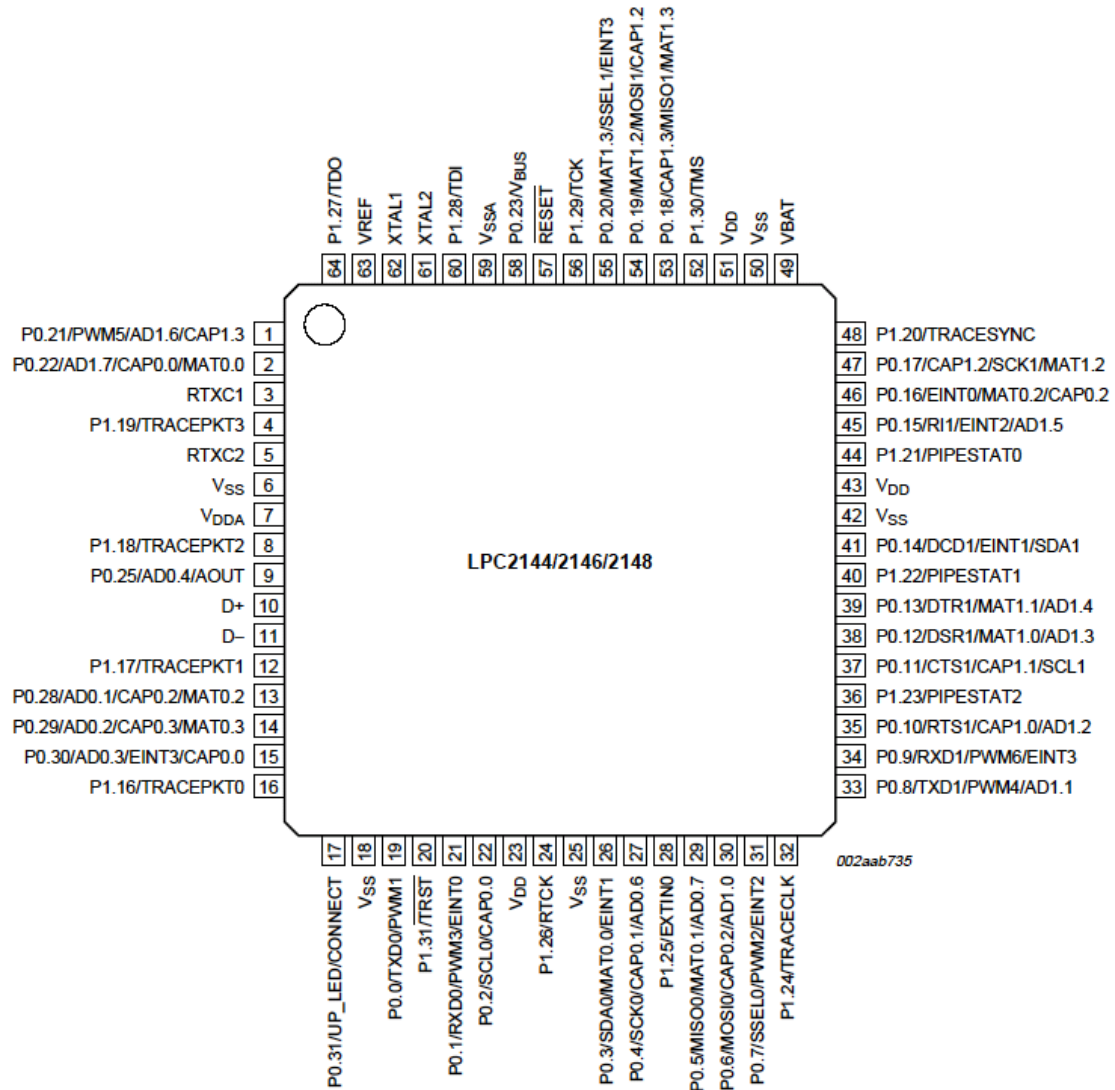


供給電源

- USB ポートで給電、電源指示 LED 付き

第二章「LPC214x」シリーズの概要





第三章 USB ダウンローダ

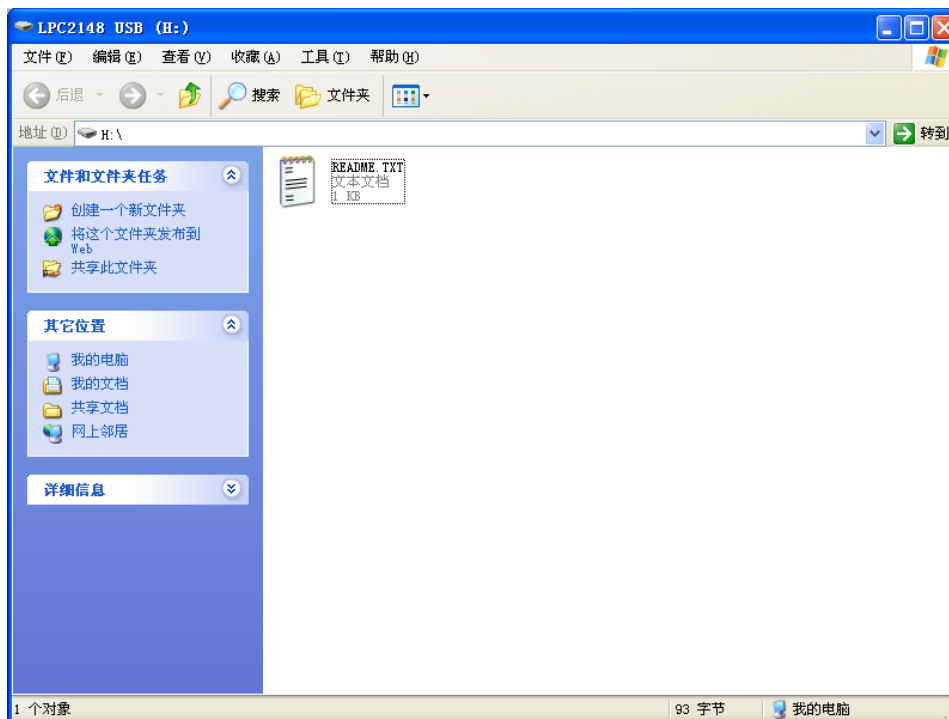
3.1 出荷状態

LPC2148 モジュールには USB ダウンローダが書き込まれています。パソコン上で開発したユーザ・プログラムを USB 経由で LPC2148 のフラッシュ・メモリに書き込むことができます。

USB ダウンローダ領域は 0~0x1FFFF、ユーザ・プログラム領域は 0x2000 からです。

USB ダウンローダはソースも含めて公開されています。Example/USB ISP は USB ダウンローダのプロジェクトです。

LPC2148 モジュールには出荷の際に USB メモリプログラムを書き込まれますので、起動の時、USB ダウンローダモードに入ります。



パソコンは LPC2148 モジュールを USB メモリとして認識します。LPC2148 の USB メモリに「Readme.txt」というファイルがあります。

正し、USB メモリは RAM 上で USB メモリを擬似されるものです、一旦、電源を切れたら、書き込まれたファイルは無くなったです。LPC2148 は再起動の場合、また USB メモリプログラムはデフォルトとして起動されます。

Flash に書き込む必要であれば、下記手順に従って行ってください。

3.2 ユーザ・プログラムを書き換える手順

3.2.1 OpenJTAG を利用し書き込む

1. ユーザー・プログラム作成

*例：USB メモリプログラムを Keil 環境で作成

① KEIL 評価版：

<http://www.dragonwake.com/download/download-stm32/MDK320b.EXE>

② USB メモリソース

<http://www.dragonwake.com/download/LPC2148/Example.rar>

解凍後：「5.1 - USBMem」

KEIL をコンパイル後、インテル HEX ファイル「Memory.hex」を生成されます。

あるいは弊社ホームページから HEX ファイルを直接ダウンロードできます。

<http://www.dragonwake.com/download/LPC2148/usb-download.rar>

解凍後：「usbMemory¥Memory.hex」

2. OpenJTAG を利用し USB メモリファイルを書き込む

LPC2148 モジュールには JTAG インタフェースがありますので、JTAG インタフェースを利用して USB メモリファイルを書き込むことができます。

① OpenJTAG 取得

弊社は販売しております。

製品紹介ページ：

<http://www.csun.co.jp/SHOP/200905191.html>

OpenJTAG マニュアル：

http://www.dragonwake.com/download/open-jtag/OpenJtag-Arm-All_manual.pdf

② OpenJTAG 設定ファイル取得：

弊社ホームページからダウンロードできます。

<http://www.dragonwake.com/download/LPC2148/usb-download.rar>

解凍後：「openjtag」

③ 設定手順

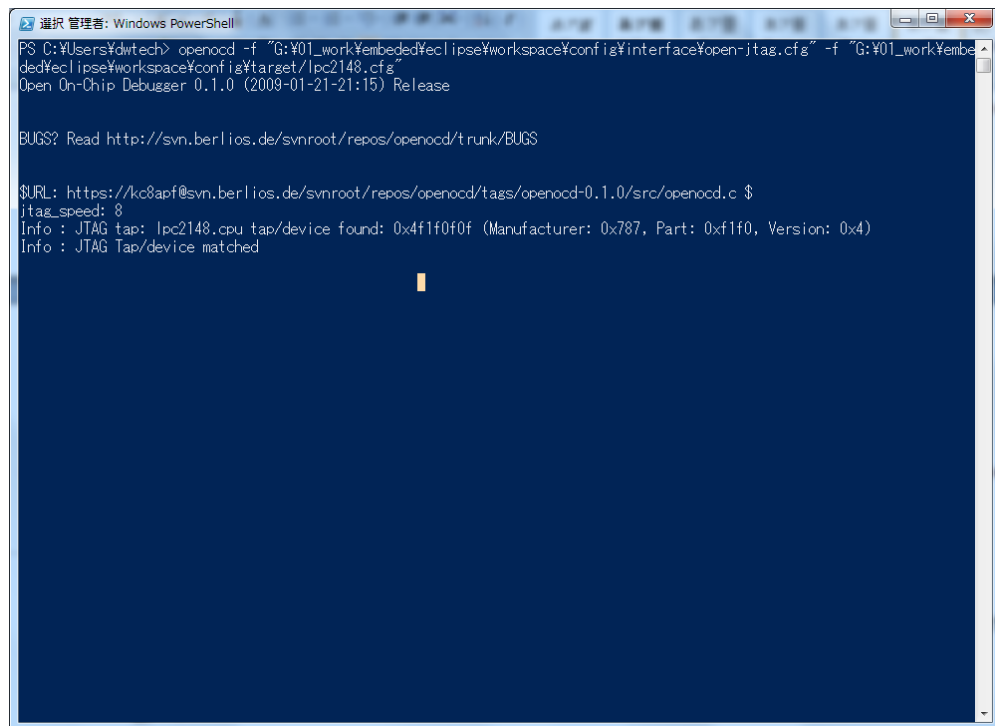
OpenJTAG の手順に従って、ソフトウェアをインストールしてください。書き込む際に、Eclipse にもできますが、LPC2148 モジュールに簡単に書き込むため、コマンドを直接発行し書き込みます。

1) openocd を起動

```
openocd -f "G:\01_work\embedded\ eclipse\workspace\config\interface\open-jtag.cfg" -f "G:\01_work\embedded\ eclipse\workspace\config\target\lpc2148.cfg"
```

* openocd は Openocd をインストールされた後、システムパスを設定されてから直接実行できます。

* 二つファイルは「OpenJTAG 設定ファイル取得」に解凍後「openjtag」にあります。



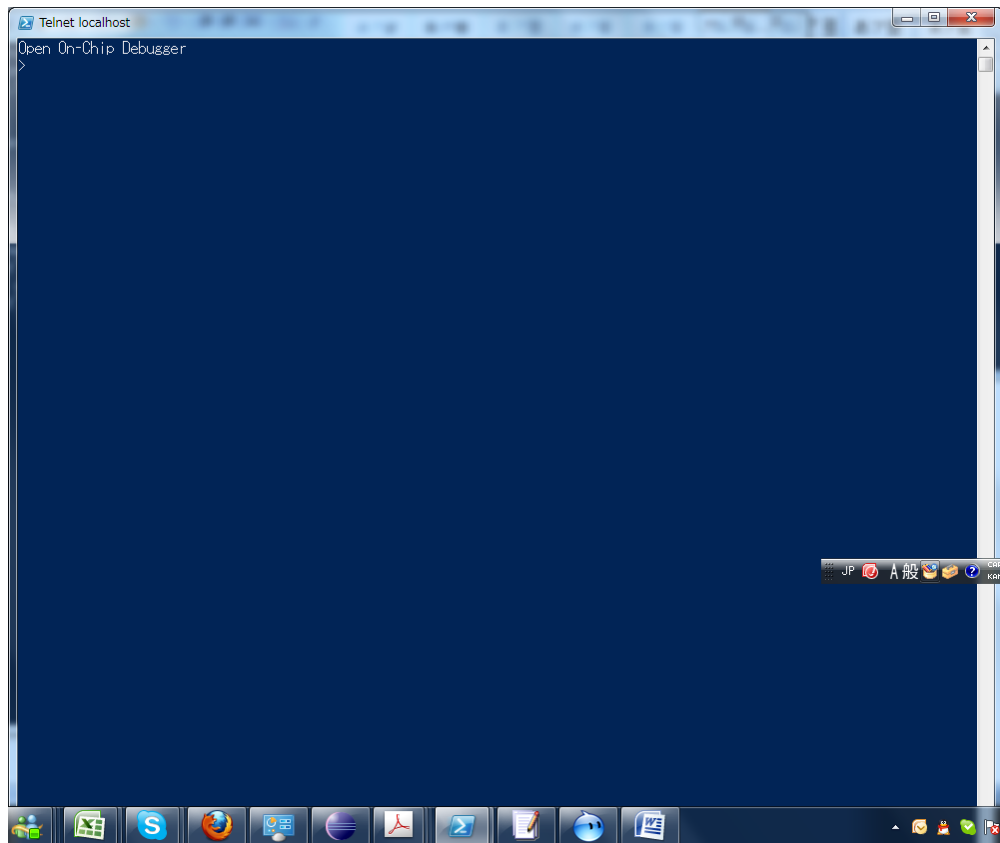
```
選択 管理者: Windows PowerShell
PS C:\Users\dwitech> openocd -f "G:\01_work\embedded\ eclipse\workspace\config\interface\open-jtag.cfg" -f "G:\01_work\embedded\ eclipse\workspace\config\target\lpc2148.cfg"
Open On-Chip Debugger 0.1.0 (2009-01-21-21:15) Release

BUGS? Read http://svn.berlios.de/svnroot/repos/openocd/trunk/BUGS

$URL: https://kc8apf@svn.berlios.de/svnroot/repos/openocd/tags/openocd-0.1.0/src/openocd.c $
jtag_speed: 8
Info : JTAG tap: lpc2148.cpu tap/device found: 0x4f1f0f0f (Manufacturer: 0x787, Part: 0xf1f0, Version: 0x4)
Info : JTAG Tap/device matched
```

2) GDB と接続

telnet localhost 4444



3)LPC2148 モジュール(flash)に書き込む

a)flash info 0

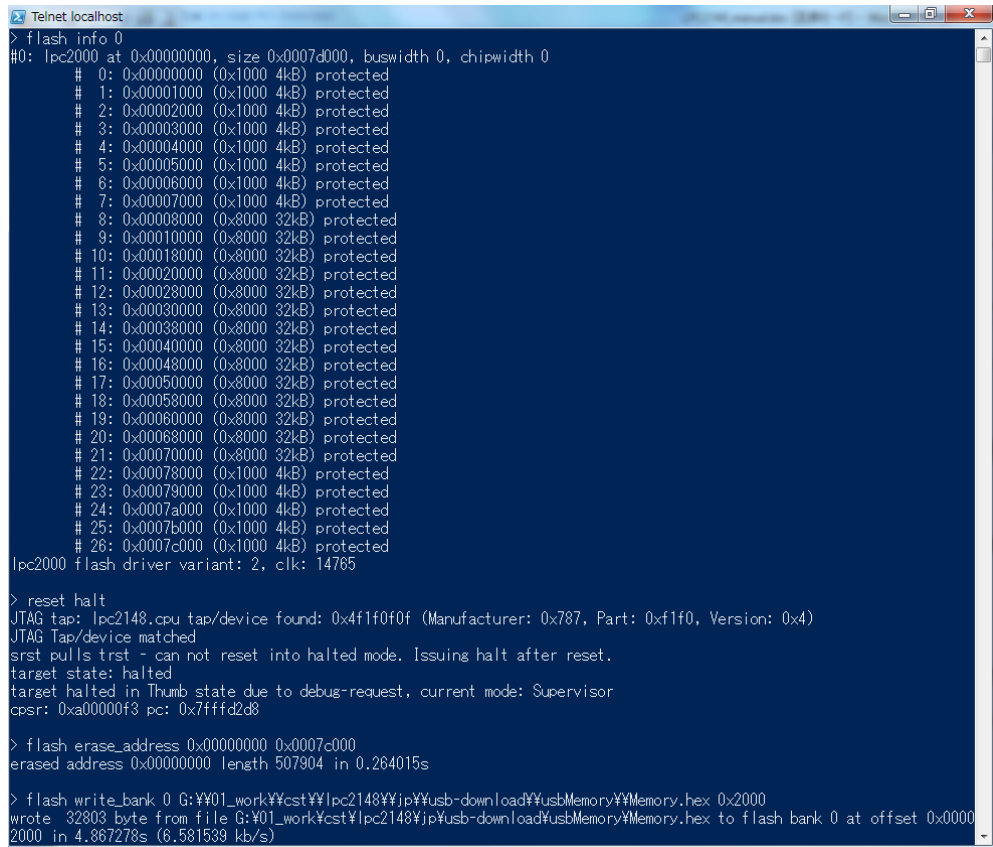
b)reset halt

c)flash erase_address 0x00000000 0x00002000

d)flash

write_image

G:\¥¥01_work¥¥cst¥¥lpc2148¥¥jp¥¥usb-download¥¥usbMemory¥¥Memory.hex 0x00002000 ihex



```
Telnet localhost
> flash info 0
#0: lpc2000 at 0x00000000, size 0x0007d000, buswidth 0, chipwidth 0
# 0: 0x00000000 (0x1000 4kB) protected
# 1: 0x00001000 (0x1000 4kB) protected
# 2: 0x00002000 (0x1000 4kB) protected
# 3: 0x00003000 (0x1000 4kB) protected
# 4: 0x00004000 (0x1000 4kB) protected
# 5: 0x00005000 (0x1000 4kB) protected
# 6: 0x00006000 (0x1000 4kB) protected
# 7: 0x00007000 (0x1000 4kB) protected
# 8: 0x00008000 (0x8000 32kB) protected
# 9: 0x00010000 (0x8000 32kB) protected
# 10: 0x00018000 (0x8000 32kB) protected
# 11: 0x00020000 (0x8000 32kB) protected
# 12: 0x00028000 (0x8000 32kB) protected
# 13: 0x00030000 (0x8000 32kB) protected
# 14: 0x00038000 (0x8000 32kB) protected
# 15: 0x00040000 (0x8000 32kB) protected
# 16: 0x00048000 (0x8000 32kB) protected
# 17: 0x00050000 (0x8000 32kB) protected
# 18: 0x00058000 (0x8000 32kB) protected
# 19: 0x00060000 (0x8000 32kB) protected
# 20: 0x00068000 (0x8000 32kB) protected
# 21: 0x00070000 (0x8000 32kB) protected
# 22: 0x00078000 (0x1000 4kB) protected
# 23: 0x00079000 (0x1000 4kB) protected
# 24: 0x0007a000 (0x1000 4kB) protected
# 25: 0x0007b000 (0x1000 4kB) protected
# 26: 0x0007c000 (0x1000 4kB) protected
lpc2000 flash driver variant: 2, clk: 14765

> reset halt
JTAG tap: lpc2148.cpu tap/device found: 0x4f1f0f0f (Manufacturer: 0x787, Part: 0xf1f0, Version: 0x4)
JTAG Tap/device matched
srst pulls trst - can not reset into halted mode. Issuing halt after reset.
target state: halted
target halted in Thumb state due to debug-request, current mode: Supervisor
cpsr: 0xa00000f3 pc: 0x7ffffd28

> flash erase_address 0x00000000 0x00002000
erased address 0x00000000 length 507904 in 0.264015s

> flash write_bank 0 G:\¥¥01_work¥¥cst¥¥lpc2148¥¥jp¥¥usb-download¥¥usbMemory¥¥Memory.hex 0x2000
wrote 32803 byte from file G:\¥¥01_work¥¥cst¥¥lpc2148¥¥jp¥¥usb-download¥¥usbMemory¥¥Memory.hex to flash bank 0 at offset 0x00002000 in 4.867278s (6.581539 kb/s)
```

「wrote 11644 byte from file G:\¥¥01_work¥¥cst¥¥lpc2148¥¥jp¥¥usb-download¥¥usbMemory¥¥Memory.hex in 1.907109s (5.962477 kb/s)」 というメッセージが見られれば、成功に書き込みます。

3.2.2 Open Link を利用し書き込む

*OpenJTAG より安い方法

1. ユーザープログラム作成

(「3.2.1 OpenJTAG を利用し書き込む」と同じですので、省略)

2. Open Link を利用し USB メモリファイルを書き込む

OpenLink の利用方法は OpenLink 手順をご参照ください。

① OpenLink 手順 :

http://www.dragonwake.com/download/open-link/OpenLink-Arm_manual.pdf

② Open Link の J-Flash ARM 機能を利用しファイルを書き込む

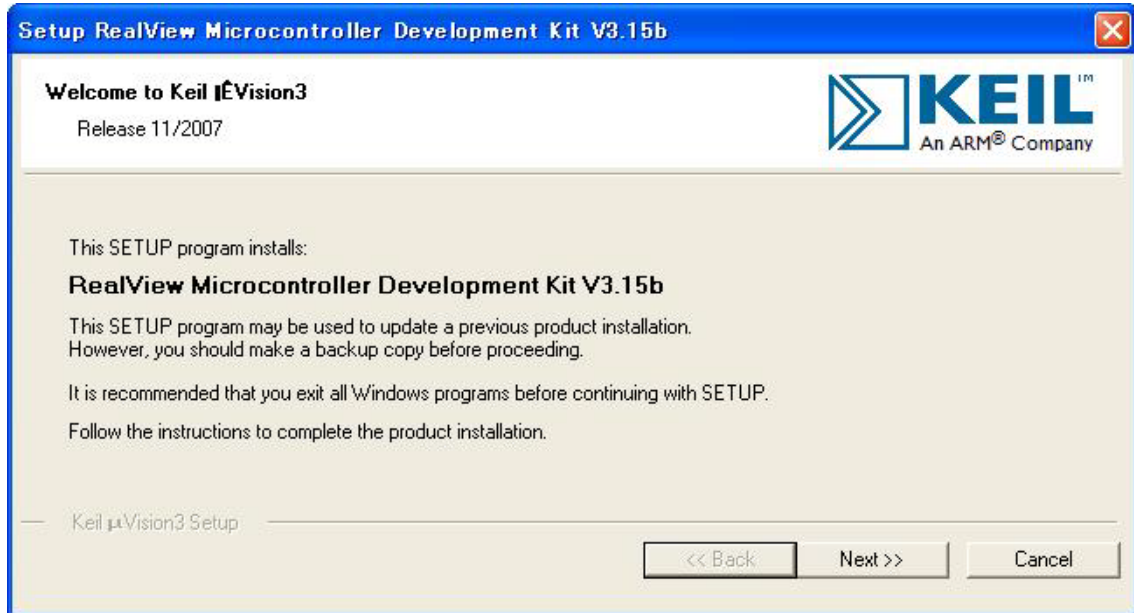
OpenLink 手順の「四、Open Link で J-Flash ARM 使用方法」をご参照のうえ、ファイルを LPC2148 に書き込むできます。

第四章 開発ツール KEIL のインストール

MDK315B.exe は開発ツール KEIL の無償評価版です。

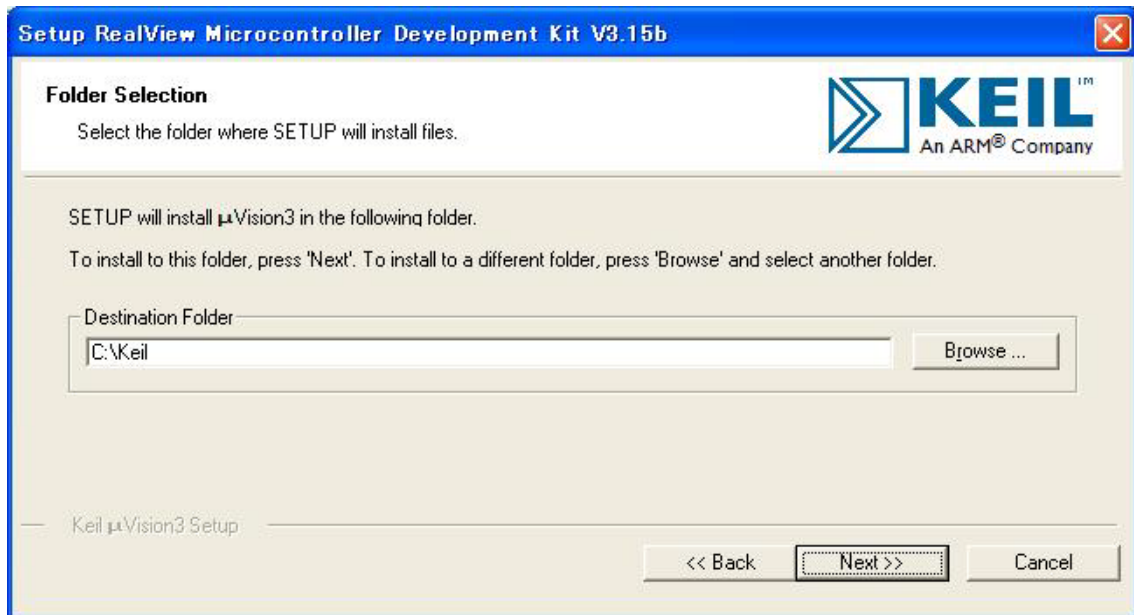
4.1 KEIL のインストール

まず、MDK315B.exe をクリックして、KEIL3.15 をインストールしてください。

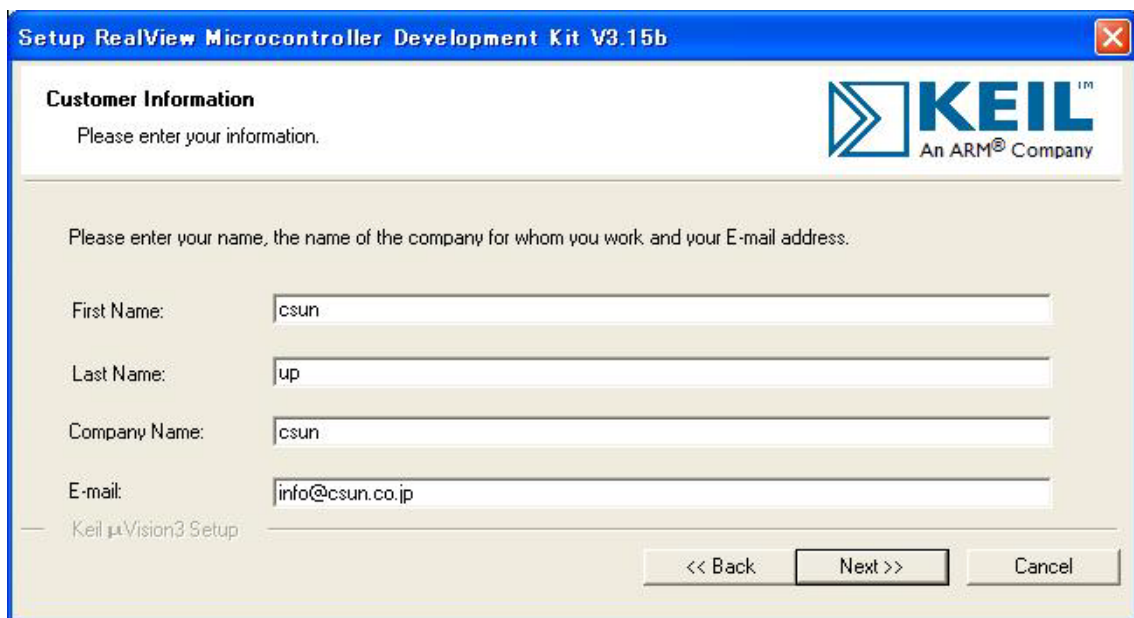


「Next」ボタンを押すと、英文のライセンスが出てきます。同意できる場合は、「**I accept the terms of the license agreement**」を選択して、「Next」ボタンを押します。

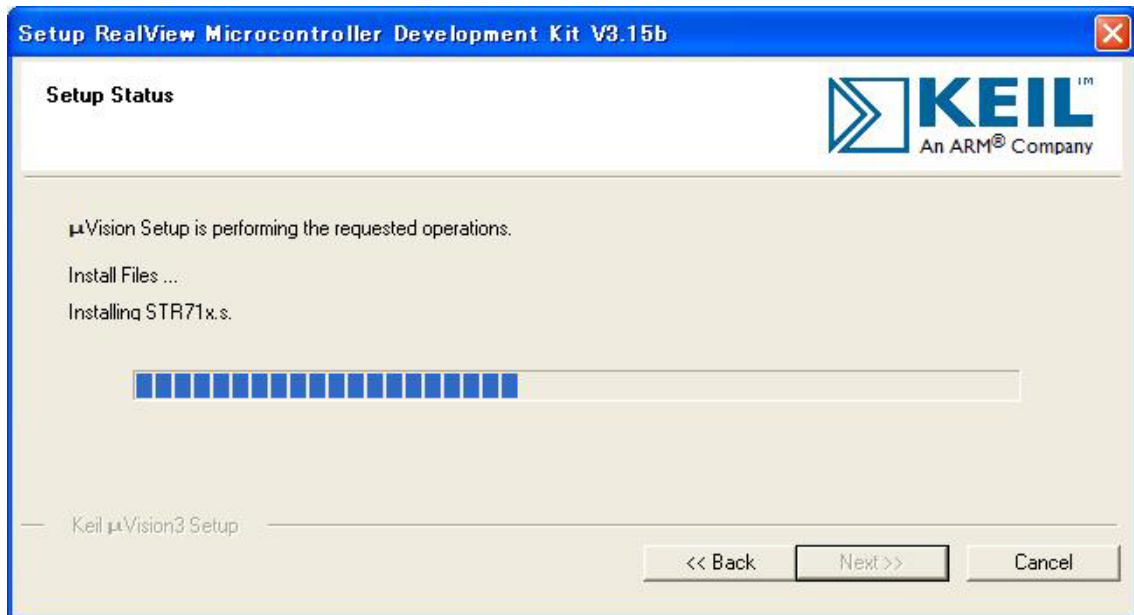




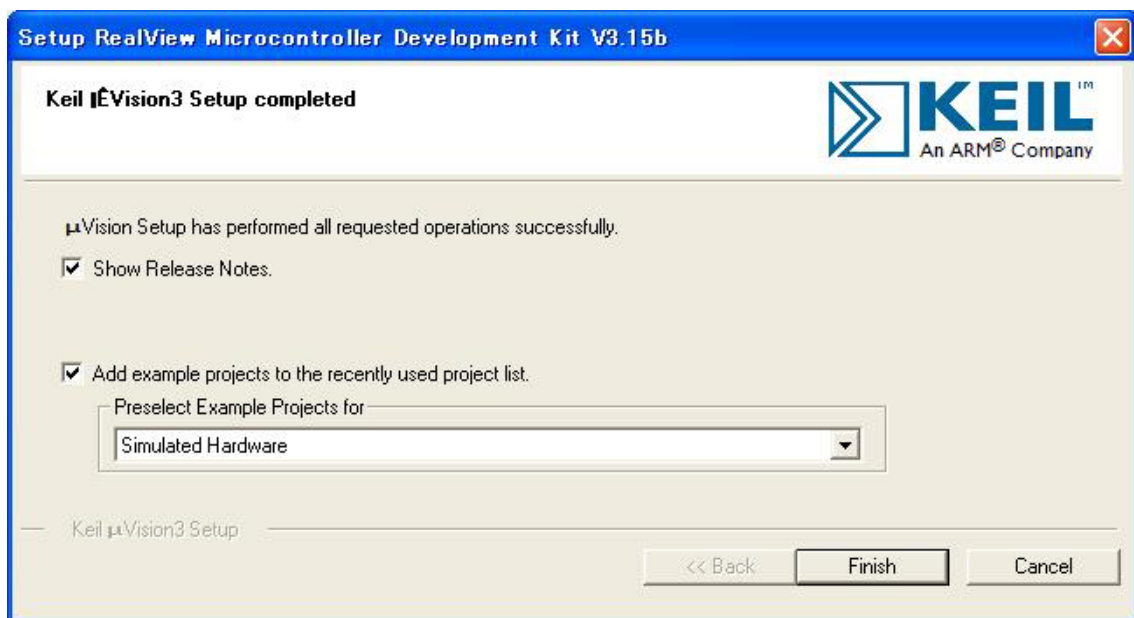
インストール先フォルダを変更せず、そのまま進んでください。



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ます。名前は半角のアルファベットで入力しましょう。



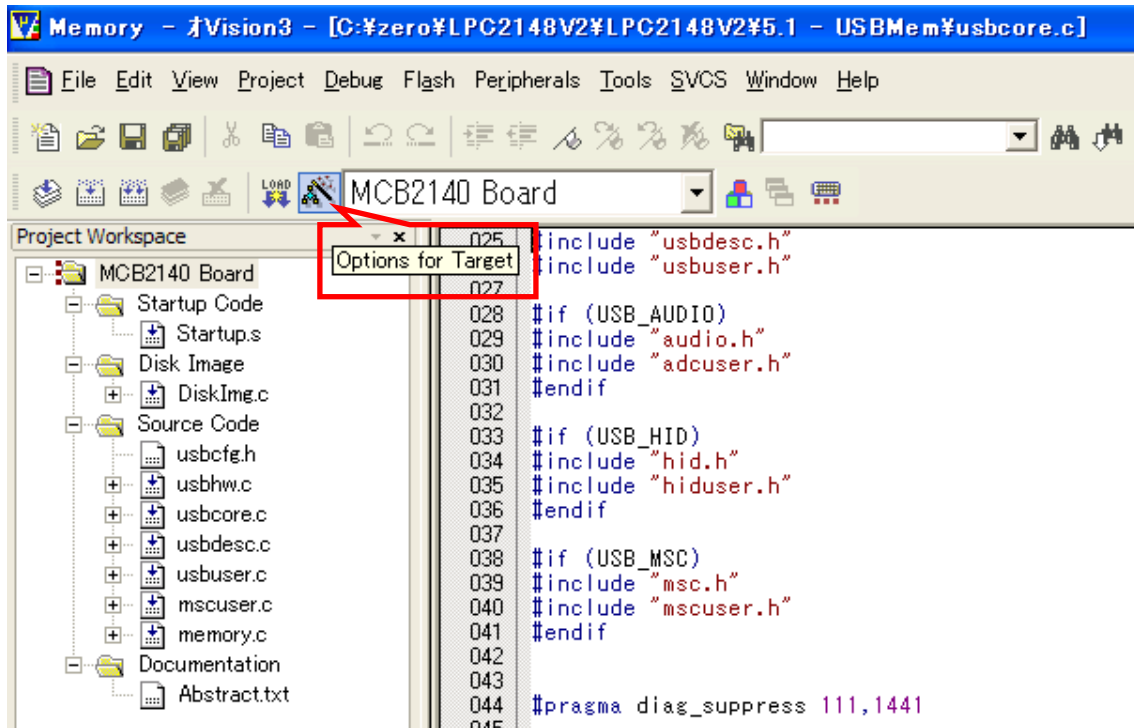
インストール中の画面です。



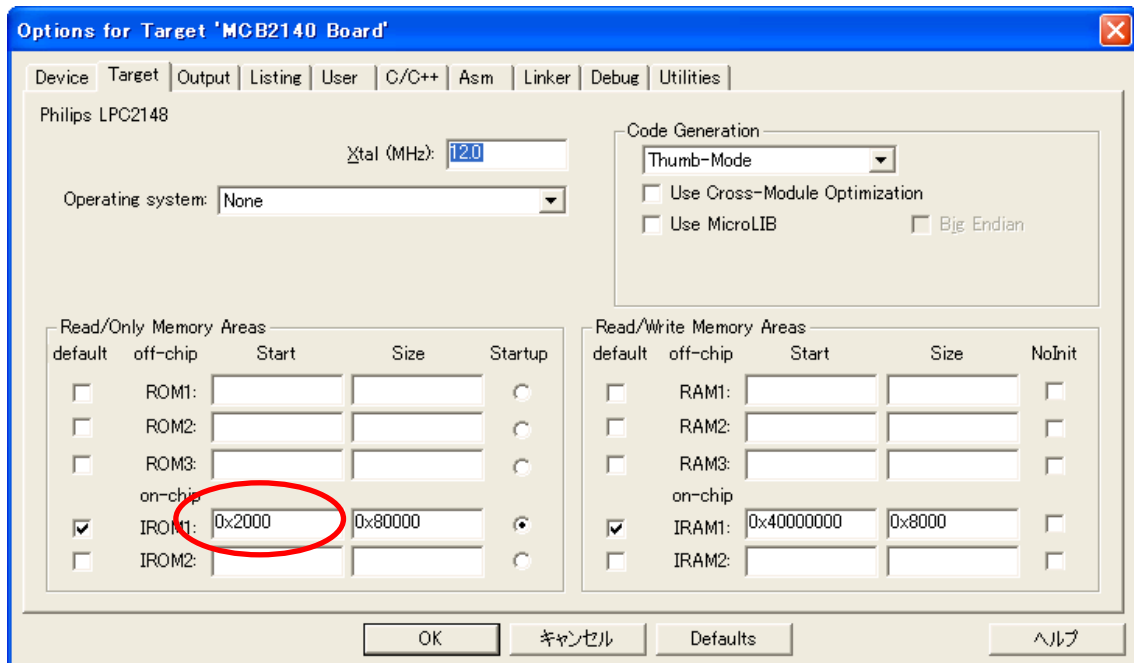
最後に「Finish」をクリックすると、ウィザードが閉じてインストールが終了します。

4.2 既存のプロジェクトから

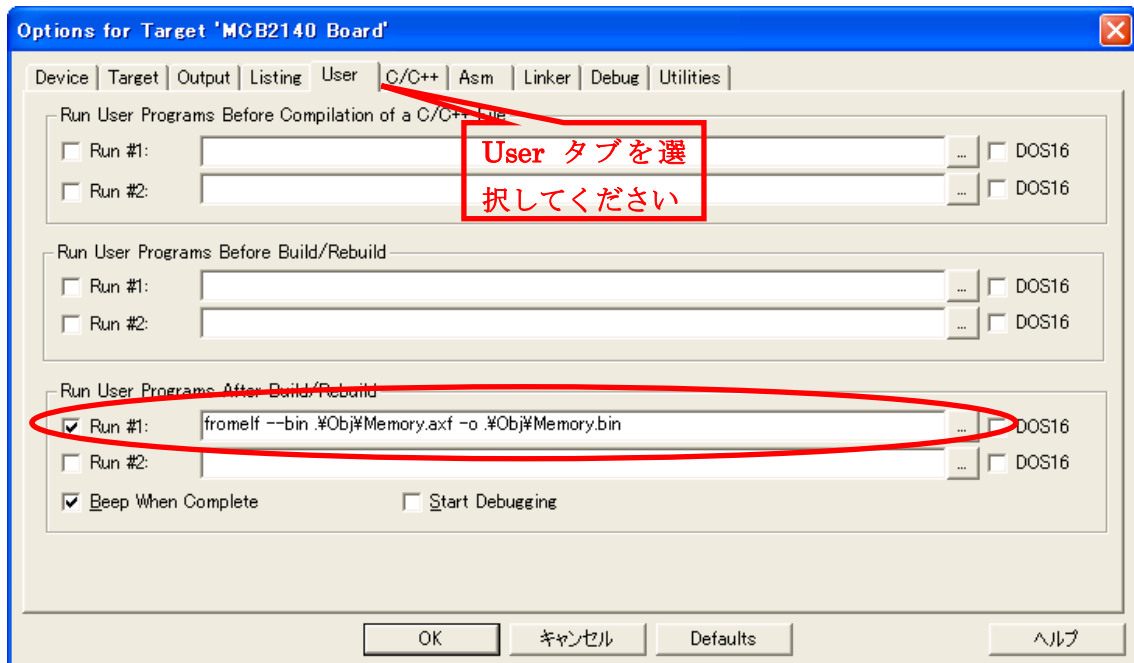
プロジェクトファイル **Example/5.1 - USBMem/Memory.Uv2** をダブルクリックして、開きます。



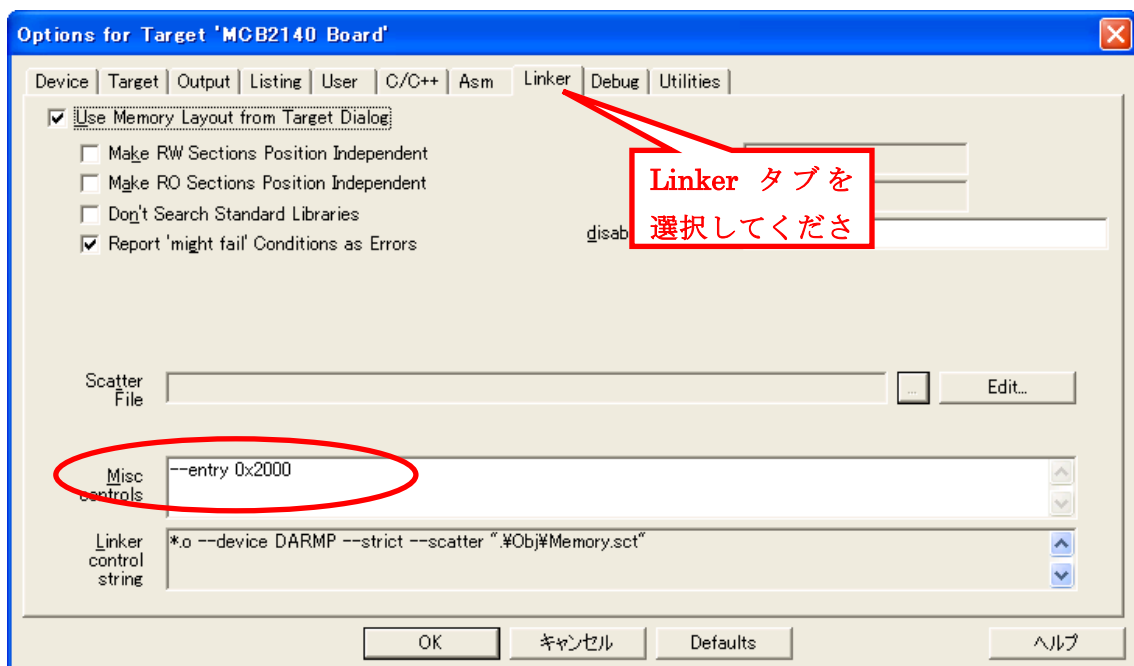
ツールバーの「Options for Target」を押します。



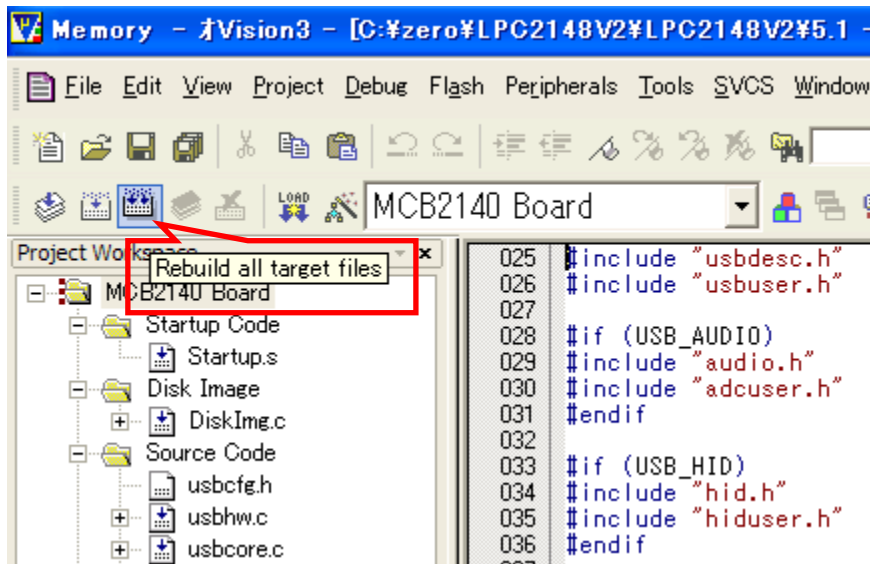
USB ブートローダは内蔵フラッシュ ROM の先頭である 0~0x1FFF を使用しますので、ユーザ・プログラムは 0x2000 番地からです。



プログラムのイメージファイル*.binを生成するため、「Run #1」をチェックして、「fromelf --bin %Obj%Memory.axf -o %Obj%Memory.bin」を入力してください。



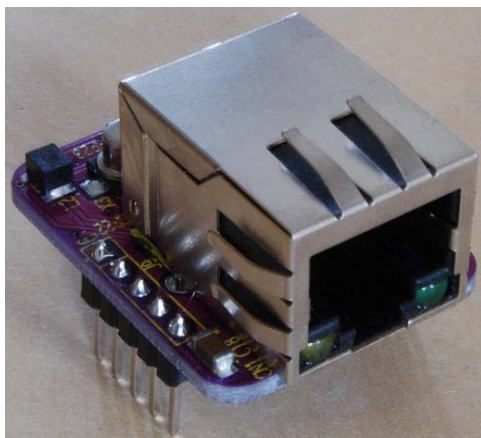
プログラムの番地「--entry 0x2000」を入力してください。設定完了すると、「OK」ボタンを押します。



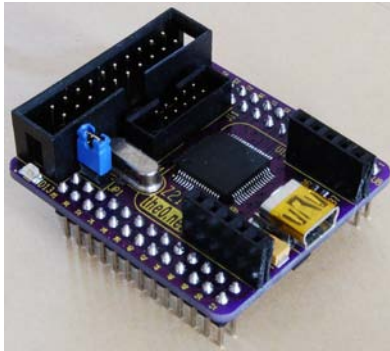
ツールバーの「Rebuild all target files」を押すと、ビルドが開始します。ビルドが成功したら、プロジェクトの Obj フォルダで Memory.bin イメージファイルを生成させます。このイメージファイルを LPC2148 モジュールにダウンロードしてください。

第五章 SPI イーサネットモジュール ENC28J60

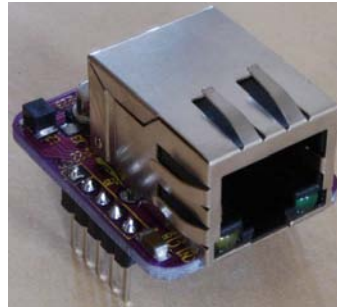
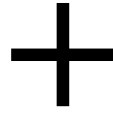
MICROCHIP 社から ENC28J60 という、SPI 接続のイーサネットコントローラ (MAC+PHY)が発売されました。10BASE-T ですので、速度は早くありませんが、SPI インターフェイスでマイコンと接続できるのが特徴です。



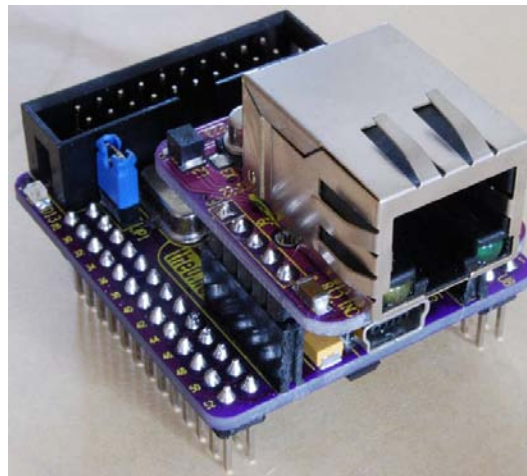
SPI インターフェイスは、SO、SI、SCK、CS の 4 本で構成され、速度も ENC28J60 の場合は 8MHz-10MHz でマイコンと接続することができます。SPI は AVR、PIC、ARM、H8 などのマイコンに標準的なインターフェイスですので、手軽にマイコンと接続出来ます。10BASE-T のイーサネットコントローラとしては RTL8019AS などが有名ですが、こちらはデータバス接続となりますので、20本近くの配線が必要になりますので、ピン数の少ないマイコンには荷が重過ぎます。



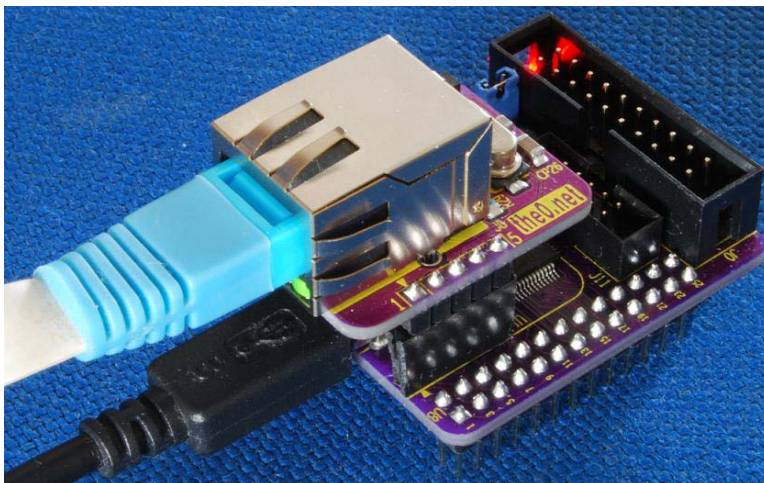
LPC2148 モジュール



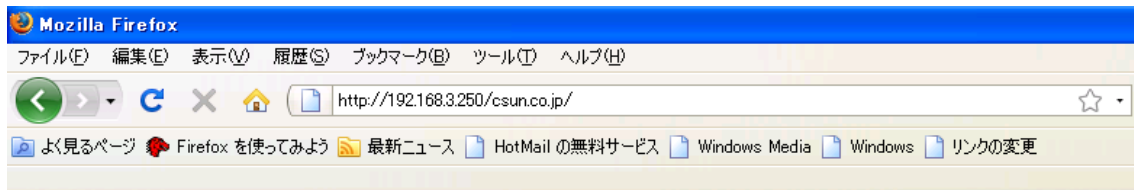
ENC28J60



サンプル : `Example/WebServer.213x.4x/out/WebServer.bin`



このサンプルを LPC2388
モジュールにダウンロード
して、パソコンのブラウザ
ー で URL
「`http://192.168.3.250/csun.co.jp`」を入力すると



Output is: ON [\[refresh status\]](#)

[Switch off](#)

version 2.10, tuxgraphics.org <http://www.csun.co.jp>

サンプルの IP アドレスとパスワードが WebServer.c にあります。自分のネットワークによって直してください。

WebServer.c

```
～ (略) ～

static uint8_t mymac[6] = {0x54,0x55,0x58,0x10,0x00,0x24};
// how did I get the mac addr? Translate the first 3 numbers into ascii is: TUX
static uint8_t myip[4] = {192,168,3,250};
// listen port for tcp/www (max range 1-254)
#define MYWWWPORT 80
//
// listen port for udp
#define MYUDPPORT 1200

#define BUFFER_SIZE 1500//450
static uint8_t buf[BUFFER_SIZE+1];

// the password string (only the first 5 char checked), (only a-z,0-9,_ characters):
static char password[]="csun.co.jp"; // must not be longer than 9 char

～ (略) ～
```

※ STM32 版の ENC28J60 モジュールのサンプルも提供しています。

第六章 SPI グラフィック液晶



サンプル : **Example/RTC_2148/output/RTC.bin**

このサンプルは LPC2148 の RTC を読み込んで、液晶で表示します。