

# ARM7/TDMI LPC2148

フリーの開発環境 Eclipse + GCC

## 低価格の OpenJTAG

## 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

2009/8/1



### copyright@2009



第一章 ARM7TDMI/LPC2148 とは	3
第二章 ARM7TDMI/LPC2148 開発キット	6
第三章 初体験	7
3.1 デフォルトのサンプル(テトリス)	7
3.2 書き込みツールのインストール	7
3.3 書き込み	
3.4 OpenJTAGで書き込み	16
3.5 OpenJTAGのドライバをインストールする	
3.6 OpenOCDのダウンロードとインストール	20
3.7 telnetで書き込み	23
3.8 GCCサンプルの紹介	
3.9 USBダウントローダ	25
第四章 クロス開発環境	
4.1 GCCツールチェーン	
4.2 Integrated Development Environment(Eclipse)	
4.3 プロジェクトを作る	
4.4 Eclipseプラグイン(Zylin Embedded CDT)インストール	
4.5 ビルドの設定	
4.6 ビルド	
4.7 GDBの設定	
4.8 OpenOCDの設定	53
4.9 デバッグ	57
4.10 デバッグ終了	61
第五章 SPIイーサネットモジュールENC28J60	63
第六章 USBでARM9(Linux)に接続	65
第七章 微弱無線モジュールnRF24L01(2.4GHz)	

使用されたソースコードは<u>http://www.csun.co.jp/</u>からダウンロー ドできます。 この文書の情報は、事前の通知なく変更されることがあります。 (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。



LPC2148 は ARM7TDMI コアを使った低消費電力・高速 NXP 社のマイコンです。

USB ターゲットを CPU に内蔵しています。USB を使ったシステムの開発・評価に最適です。

H8 や SH と比べると価格も安くて、実にスピードも速い、容量も大きい、消費電力も小さいです。

CPU は 3.3V 動作なのですが 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。

ルネサス SH7144 と比べて

- スピードは 60MHz なので 20%速い
- フラッシュは 512 K バイトなので 2 倍
- RAM は 42K あるので約5倍
- 消費電力は 60MHz フルスピードで 50mA 程度です。約 2~4 分の 1 です。クロックを 落とせばもっと減ります。
- ROM の書き換え回数 10 万回、事実上の無制限、データ保持 20 年。

#### LPC2148 の主な仕様

- ・ ARM7TDMI-S 16/32bit RISC マイコン
- ・ FLASH:512K バイト (H8/3069F と同じメモリ容量)
- RAM:42K バイト(H8/3069Fの約 2.5 倍)
   RAM内蔵の1チップCPUでは最大級
- ・ 60MHz 動作(12MHz × 5 逓倍)
- USB2.0 対応インターフェース内蔵(max 12Mbps)
- ・ 10 ビット AD コンバータ 2 ユニット内蔵、14Ch
- ・ 10 ビット DA コンバータ x1 c h 内蔵
- UART(16C550) x 2, I2C x 2, 32 ビットタイマ x 2, PWM x 6, WDT, RTC など通信系のインターフェース(SPI/SSP)は 2ch 内蔵されています。
- ・ 5V トレラント I/O(C P U は 3.3V 動作ですが、I/O ポートは 5V の入力を受けられます)
- ・ JTAG インターフェース内蔵
- Fast I/O 機能:通常は CPU コアと I/O のクロックが異なるため速い CPU コアでも I/O コントロールにウェイトが入ってしまい、高速に I/O を操作できません。この LPC2148 では Fast I/O 機能により高速に I/O ポートを操作できます。他のCPUの約 3.5 倍のス ピードでポートの操作ができます。













### 第二章 ARM7TDMI/LPC2148 開発キット

#### 開発キットのインターフェース

- RS232(16C550)  $\times 2$
- USB2.0 device  $\times 1$
- JTAG/ICE
- SD カードソケット
- AD テスト用のボリューム
- I2C EEPROM
- ユーザーLED ×1
- ユーザーボタン×1
- ジョイスティック×1
- スピーカー×1
- 1.8 インチ TFT 液晶、分解能 160 × 128

#### 外形寸法

• 外形寸法: 110×70(mm) 突起物は除く

#### 供給電源



 5VDC 電源、プラグ 2.1mmφ、極性はセンタープラス ⊖ー € ⊕ です。電源スイ ッチと電源指示 LED 付き。USB 給電可。

### 第三章 初体験

### 3.1 デフォルトのサンプル(テトリス)



# Example-2148GCC.rar

はLPC2148開発キットのサンプルです。回路図とソースコードも含みます。なかのほかのサンプルを体験してみよう。

### 3.2 書き込みツールのインストール

#### 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/

FlashMagic.exe を実行すると、LPC2148 ボードの書き込みツールをインストールします。 LPC2148 の Flash を更新すれば、ほかのサンプルを体験できます。



「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。

< Back

<u>N</u>ext >

Cancel

At least 4.7 MB of free disk space is required.



🔂 Setup - Flash Magic	$\mathbf{X}$
Select Start Menu Folder Where should Setup place the program's shortcuts?	B
Setup will create the program's shortcuts in the following Start Menu folder.	
To continue, click Next. If you would like to select a different folder, click Browse.	
Flash Magic Browse	
Embedded Systems Academy, Inc.	
< <u>B</u> ack <u>N</u> ext > Cancel	

#### メニュー・フォルダも変更せず、そのまま進んでください。

🕼 Setup - Flash Magic	×
Select Additional Tasks Which additional tasks should be performed?	Z
Select the additional tasks you would like Setup to perform while installing Flash Magic, then click Next. Additional icons: Create a <u>desktop icon</u> Create a <u>Q</u> uick Launch icon	
Embedded Systems Academy, Inc	

#### 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。

🕞 Setup - Flash Magic	
Installing Please wait while Setup installs Flash Magic on your computer.	
Creating INI entries	
Embedded Systems Academy, Inc.	
(	Cancel

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。

### 3.3 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



籋 Flash Magic - NON PRODUCTION					
<u>F</u> ile <u>I</u> SP <u>O</u> ptions <u>T</u> ools <u>H</u> elp					
🖻 🗟   🔍 🗿 🐗 🗸 📕 🔈   😻   🛙	ब 🕜 😂				
Step 1 - Communications	Step 2 - Erase				
COM Port: COM 1	Erase block 0 (0x0000-0x0FFF)				
Baud Rate: 9600 💌					
Device: 89C51RA2xx 💌					
Interface: None (ISP)	Erase all Flash+Securitu+Clks				
Oscillator Freq. (MHz): 16.000000	Erase blocks used by Hex File				
Step 3 - Hex File					
Hex File:	Browse				
Modified: Unknown	more info				
Step 4 - Options	Step 5 - Start				
Verify after programming       Set Security Bit 1       Start         Fill unused Flash       Set Security Bit 2         Gen block checksums       Set Security Bit 3         Execute       Prog Clocks Bit					
Buy your NXP LPC evaluation and starter kits online!					
www.lpctools.com	•				
	0				

Flash Magic の初画面です。

Flash Magic のメニュー「Options」→「Advanced Options」を選択してください。

🏀 Flash Magic - NON PRODUCTION USE ONLY					
<u>F</u> ile <u>I</u> SP	Options Tools Help				
🖻 🗟 🕻	Advanced Options		🏹 🕜 😂 👘		
Step 1 - Co	Disable Hints Update		Step 2 - Erase		
	COM Port: COM 1	•	Erase block 0		
В	aud Rate: 9600	-	Erase DIOCK I		
	Device: 89051BA2vv	-			



Advanced Options
Communications Hardware Config Decurity Just In Time Code Timeouts Misc Use DTR and RTS to control RST and PSEN Keep RTS asserted while COM Port open T1: 50 ms T2: 100 ms Assert DTR and RTS while COM Port open
Cancel OK

画面の通りに設定して、「OK」ボタンを押してください。

<u>F</u> ile <u>I</u> SP <u>O</u> ptions <u>T</u> ools <u>H</u> elp	
🖻 🕞   🔍 🗿 🐗 🖌 📕 🔈   🗇   國   😮 😂	
Step 1 - Communications Step 2 - Erase	
CBM Port: COM 1  Erase block 0 (0x000000-0x000FFF) Erase block 1 (0x001000-0x001FFF) Erase block 1 (0x001000-0x001FFF) Erase block 2 (0x001000-0x001FFF)	
Device:         LPC2148	
Interface: None (ISP)	
Oscillator File。(MHz): 12.000000 Erase blocke wed by Hex File 書き込み	ファイ
ル*.HEX	を選択
Step 3 - Hex File       Hex File:     C:\zero\Example-2148GCC\21.Tetris\Tetris.hex       Modified:     Browse   Browse	
Step 4 - Options Step 5 - Start!	
<ul> <li>✓ Verify after programming  Set Code Read Prot</li> <li>✓ Fill unused Flash</li> <li>✓ Gen block checksums</li> </ul>	
■ Execute 書き込みボタン	
On-Line training classes for microcontrollers and embedded networking and Internetworking www.esacademy.com/fag/classes	

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 を選択してください。





書き込みの設定

書き込みボタン「Start」ボタンを押す前に、LPC2148 ボードの JP5/JP8/JP9 設定と RS232 ケーブルの接続を確認してください。「Start」ボタンを押すと、書き込み開始

🏶 Flash Magic - NON PRODUCTION USE ONLY - Using 38 🔳 🗖 🔀			
<u>F</u> ile <u>I</u> SP <u>O</u> ptions <u>T</u> ools <u>H</u> elp			
🗀 🔜   🔍 🖩 🐳 🖊 🗮 >   🎔   🖳   🍞 😂			
Step 1 - Communications Step 2 - Erase			
COM Port:       COM 1         Baud Rate:       38400         Device:       LPC2388         Interface:       None (ISP)         Interface:       None (ISP)         Scillator Freq.       (MHz):         12.000000       Erase blocks used by Hex File         Step 3 - Hex File       Browse			
Ches 4 Options			
Step 9 - Start       Verify after programming       Set Code Read Prot       Fill unused Flash       Gen block checksums       Execute			
Develop your own Flash Magic based applications using our developer's kit!			
www.canopenstore.com/pip/flashmagicdevkit       Programming device (0x00002C00)			

書き込み中の画面です。





実行の時、JP5/JP8/JP9の設定。

### 3.4 OpenJTAG で書き込み



Open-JTAG は ARM 用の USB-JTAG エミュレータで す。 ARM7 、 ARM9 、 Cortex-M3, XSCALE に対応、 OpenOCD をサポートします。 USB-RS232 機能もあります。 COM と LPT ポートがないノ ートパソコンに最適。







LPC2148 モジュールはシリアルポートがありませんので、open-JTAG で書き込みしかできません。



### 3.5 OpenJTAG のドライバをインストールする

OpenJTAG をパソコンの USB ポートに挿入して、下の通りにドライバをインストールし てください。





新しいハードウェアの検出ウィザード	
検索とインストールのオブションを選んでください。	
<ul> <li>次の場所で最適のドライバを検索する(S) 下のチェックボックスを使って、リムーバブルメディイバがインストールされます。</li> <li>リムーバブルメディア(フロッピー、CD-Referred (フロッピー、CD-Referred))</li> <li>(ア)次の場所を含める(G)</li> <li>C:¥OpenJTAG¥usb-driver</li> <li>検索しないで、インストールするドライバを選択するには、このオブションとは限りません。</li> </ul>	(アやローカル パスから検索できます。検索された最適のドラ OM など)を検索(M) を選びます。選択されたドライバは、ハードウェアに最適のもの く 戻る(B) 次へ(N) キャンセル
新しいハードウェアの検出ウィザード	
ソフトウェアをインストールしています。お待ちください。	
100ASK JTAG	
6	B
	(<戻る(B) 次へ(N) > キャンセル

USB ドライバのインストールは3回があります。インストール完了すると、デバイスマネ ージャで三つのデバイスが見えます。



鳥 デバイス マネージャ	
ファイル(E) 操作( <u>A</u> ) 表示( <u>V</u> ) ヘルプ( <u>H</u> )	
Image: Second Sector	
アメンロー1407&PID_5118.DeviceUesc% (CUM4) 夏 ECP デリンクポート (LPT1)	

OpenJTAG は USB シリアルポートとして使えます。

### 3.6 OpenOCD のダウンロードとインストール

1) 下記のリンクをクリック

 $\underline{http://developer.berlios.de/project/showfiles.php?group\_id=4148}$ 

2) "openocd-0.1.0.msi"をクリック



🖉 BerliOS Developer: Project Fil	elist – Microsoft Internet Ex	(plorer					_ 🗆 🗵
ファイル(E) 編集(E) 表示(⊻)	〒√2) お気に入り(A) ツール(T) ヘルプ(H)					<b>.</b>	
🔆 F3 • 🕥 • 💌 💈	🚮 🔎 検索 🥎 ł	8気に入り 🧭	) 😒 - 📚	w - 🗾	🗶 🔏 🛛		
アドレス(D) 🙋 http://developer.b	perlios.de/project/showfiles	.php?group_id=4	148			🔽 🔁 移動	リンク 👋
Software/Group	Summary   Home Pa Docs   SVN   Files	ge   Forums Screenshot	Bugs   Su s   Memberli	pport   Featu st   Donation	ures   Patche   Admin	es   Lists	-
Search	Below is a list of all fi Release Notes and C	les of the pr hangeLog (a	oject. Before accessible by	downloading, clicking on rel	, you may wa ease version	nt to read ).	
Project: Open On-	Dackage	Release	Filename		Date		
Chip Debugger	Раскаде	& Notes	Fliename	Size	D/L Arch	. Type	
Discussion Forums P Submit Bugs P	openocd						
Request Support P	0.1.0			20	09-01-20 09	9:00	
Request Features M	openocd-0.1.0	)-1.fc10.i386	.rpm	486764	392 i386	.rpm	
Droject Admin N	<pre>copenocd-0.1.0</pre>	).msi 🔿		2516992	1848 i386	Other	
Project AdminiP	openocd-0.1.0	).msi.md5		53	87 Any	Other	
Software	openocd-0.1.0	).tar.gz		985328	1597 Any	Source .gz	
Software Map 🖻	openocd-0.1.0	).tar.gz.md5		62	84 Any	Other	
New Releases Microsom	openocd-0.1.0	).tar.gz.sha1		69	63 Any	Other	
Code Snippet Library P	Project Totals:	1	6	3989268	4071		
BerliOS Developer							<b>•</b>
é						) インターネット	

3)リンクの Download をクリックし Windows 用インストーラパッケージをダウンロード する

🚈 BerliOS Download - The Open Sourc	e Mediator - Microsoft Intern	et Explorer			
ファイル(E) 編集(E) 表示(⊻) お気	に入り(A) ツール(T) ヘルプ	Ē			
🔇 E2 • 🕥 • 💌 💈 🏈	🔎 検索 🥎 お気に入り	) 🥝 🏂	w • 🔜 🎇		
アドレス(D) 🕘 http://prdownload.berlio:	s.de/openocd/openocd=0.1.0.n	nsi			移動 リンク »
<b>berli</b> 0	S BerliOS The Open Source	ce Mediator			FOKUS
Ads by Google Free Download	Is Blog Windows Med	<u>lia Player Fre</u>	eware Firewall	<u>Downloadable F</u>	reeware
Stylevision: Your are requesting file: /openocd/openocd-0.1.0.msi					
	Host	Location	Download		
	download.berlios.de	Berlin, Germany (	Download		
	download2.berlios.de	Berlin, Germany	Download		
	-				
	Copyright ©:	2000-2009 FOKI	JS		
🕑 go to www.altova.com				🔄 📔 🙋 インター	ネット <i>()</i> ,

5)動作確認のため下記を実行します

1. Open-JTAG をターゲット(LPC2148 基板)とPCに接続



- 2.ターゲットに電源を入れます
- 3.コマンドプロンプトでディレクトリを移動 (cd Example-2148GCC¥open-jtag)
- 4.下記のコマンドを入力します

( <code>openocd -f open-jtag.cfg –f lpc2148.cfg</code> )

#### 画面のように"Info: JTAG Tap/device matched"と表示されればOKです

(この時点で ARM LPC2148 と通信が出来ています)

🛤 C:¥WINDOWS¥system32¥cmd.exe - openocd -f open-jtag.cfg -f lpc2148.cfg 🗕 🗖	×
Microsoft Windows XP [Version 5.1.2600] (C) Copyright 1985-2001 Microsoft Corp.	-
C:¥Documents and Settings¥zqing>cd¥zero¥Example-2148GCC¥open-jtag	
C:¥zero¥Example-2148GCC¥oper≤itag>openocd -f open-jtag.cfg -f lpc2148.cfg> Open On-Chip Debugger 0.1.0 (2009-01-21-21:15) Reiease	
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 \$ 500 kHz	
Info : JTAG tap: lpc2148.cpu tap/device found: 0x4f1f0f0f (Manufacturer: 0x787, Part: 0xf1f0, Version: 0x4) Info : JTAG Tap/device matched	
Warn : no telnet port specified, using default port 4444 Warn : no gdb port specified, using default port 3333 Warn : no tel port specified, using default port 6666	
	•



### 3.7 telnet で書き込み

Image: Second
Microsoft Windows XP [Version 5.1.2600] (C) Copyright 1985-2001 Microsoft Corp.
C:¥Documents and Settings¥zqing>cd¥zero¥Example-2148GCC¥open-jtag
C:¥zero¥Example-2148GCC¥open-jtag>telnet localhost 4444
•

新コンソールで telnet localhost 4444 コマンドを入力します。telnet で下記のコマンドを入力します。

halt	ターゲットを停止		
flash write_image	erase Tetris.hex ihex	*.hex ファイルを書き込み	
resume 0	実行		
exit	telnet をクロス		

Image: State	×
Open On-Chip Debugger > halt target state: halted target halted in Thumb state due to debug-request, current mode: Supervisor cpsr: 0xa00000f3 pc: 0x7fffd2d8	
> flash write_image erase usbisp.hex ihex auto erase enabled wrote 7304 byte from file usbisp.hex in 1.359375s (5.247126 kb/s)	
> resume 0	
> exit	
ホストとの接続が切断されました。	
C:¥zero¥Example-2148GCC¥open-jtag>_	-



### 3.8 GCC サンプルの紹介

1. LEI	)	LED 点灯	
2. GPI	O-IN	入力実験	
3. PW	М	PWM で LCD のバックライトを点灯	
4. LCI	)	LCD で英語を表示	
5. LCI	)-Japanese	日本語を表示	
6. LCI	D_BMP	ピクチャを表示	
7. RTC	2	RTC 実験	
8. IIC		IIC の EEPROM のライト・リード	
9. AD(	2	ADC 実験	
10.DA	С	DAC 実験	
11.UA	RT_FIFO	UART のポーリング	
12.UA	RT_INT	UART の割り込み	
13.Tin	ner0-INT	タイマーの割り込み	
14.EX'	TINT	外部の割り込み	
15.AD	_INT	ADCの割り込み	
16.Sou	ınd	DAC で音声	
17.SD	FatFs	SD メモリのデモ	
18.US	Btarget		
	examples/msc.hex	LPC2148 経由 SD メモリが USB メモリとして	
examples/serial.hex		LPC2148 が USB シリアルポートとして使える	
ドライバは USBtarget		t/examples/usbser.inf です。	
19.IAF	D	プログラムで Flash を更新	
20.We	bserver	ウェブサーバ、イーサネット ENC28J60 用	
21.Tet:	ris	デフォルトのサンプル(テトリス)	
22.nR	F24L01	微弱無線モジュール(nRF24L01)の通信実験	



#### 3.9 USB ダウントローダ



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。

この仕組みによって LPC2148 モジュールは、 JTAG デバッグがなくても USB 経由でアプリケーショ

ン・プログラムの書き換えが可能です。ユーザはパソコンと LPC2148 モジュールだけで ARM プロセッサのシステムを開発できます。



誤って0番地に上書きすると、USBブートローダ壊れてしまうので、注意が必要です。この修復にはJTAGデバッグが必要になります。

LPC2148 モジュールが起動の時、裏面の拡張ヘッダの 24,52 ピンをショットすれば、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「firmware.bin」というファイルがあります。サイズは 492KB です。このファイルは ユーザが使える FlashROM のイメージです。使えるサイズは 492KB です。



#### ユーザ・プログラムを書き換える手順:

まず、「firmware.bin」というファイルを削除してください。



ユーザ・プログラムのイメージファイル\*.bin を USB メモリにコピーしてください。

☞ H:¥		
ファイル(E) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)		
③ 戻る - ③ - 🎓 🔎 検索 🌔 フォルダ 💷・		
アドレス(D) 🗢 H¥		💌 🄁 移動
▲ 名前 ▲	サイズ 種類	更新日時
ファイルとフォルダのタスク 🍣 🖬 Memory.bin	12 KB BIN ファイル	2009/03/24 16:37
■1 このファイルの名前を変更す		
● る		
CODアイルを1990 つう このつっくしたつやっする		
□ COUPTINESE SIG		
□ COJ7F1ルを電子メールで 送信する		
🗙 このファイルを削除する		
その他 🔹		
2 71 ILI-9		
🔁 共有ドキュメント		
📢 २२ २७२७-७ 💌		
種類: BIN ファイル 更新日時: 2009/03/24 16:37 サイズ: 11.3 KB	11.3 KB 🔡	71 コンピュータ 🔬

LPC2148の拡張ヘッダの24,52 ピン間の短絡線を外して再起動すると、ユーザ・プログラムを実行します。



### 第四章 クロス開発環境

#### 4.1 GCC ツールチェーン

各種ユーティリティ:

 $\underline{http://www.yagarto.de/download/yagarto/yagarto-tools-20070303-setup.exe}$ 

#### GCC ツールチェーン

http://sourceforge.net/projects/yagarto/files/YAGARTO%20for%20Windows/yagarto-bu-2.19.1 gcc-4.3.3-c-c%2B%2B nl-1.17.0 gi-6.8.50 20090329.exe/download

インストールが出来たら make の確認をするためコマンドプロンプトを起動し、右記のコ マンドを入力します(make --version)。画面に下記のメッセージが出てくればOKです。





#### サンプルのコンパイル:

- 1. コマンドプロンプトでディレクトリを移動 (cd Example-2148GCC¥1. LED)
- 2.下記のコマンドを入力します

(make)



コンパイル成功したら、\*.hex ファイルを生成させます。生成された\*.hex ファイルを LPC2148 に書き込みましょう。



### 4.2 Integrated Development Environment(Eclipse)

#### JRE バージョン確認:

確認コマンド: java -version



JRE がなければ、あるいは 1.4.2 以下なら、JRE のインストールが必要です。 <u>http://java.sun.com/javase/downloads/index.jsp</u>

#### Eclipse のインストール:

1)下記のリンクをクリック

http://www.eclipse.org/downloads/index.php

2 ) Eclipse IDE for C/C++ Debelopers(79MB)の Windows をクリック



🖉 Eclinse Downloads – Microsoft Internet Explorer	
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルブ(H)	
③ 戻る ▼ ② - ▼ 2 公 2 検索 ☆ お気に入り ④ ② ● 急 • 急 ■ ▼	13 🖬 🚳
アドレス(D)  個 http://www.eclipse.org/downloads/index.php	<ul> <li>予移動 リンク ※</li> </ul>
Norton · 🛒 🕜 · 🕝 カードとログイン情報 ·	
Google 🚽 👌 🖓 - 🏠 ブックマーク・	» 🔩 ד 🔵 פאלט ד 🍖 י
DAEMON - Q O I O DAEMON Tools Lite - 🎇 AstroBurn 🌄 Products	s ▼ 🔂 News [30/30] ▼ »  🥳
Eclipse IDE for Java Developers (92 MB) The essential tools for any Java developer, including a Java IDE, a CVS client, XML Editor and Mylyn. More Downloads: 120,776	Windows Mac OS X (Carbon) Mac OS X (Cocoa) Linux 32bit Linux 64bit
Eclipse for PHP Developers (139 MB) Tools for PHP developers creating Web applications, including PHP Development Tools (PDT), Web Tools Platform, Mylyn and others. More Downloads: 71,421	Windows Mac OS X (Carbon) Mac OS X (Cocoa) Linux 32bit Linux 64bit
CONTRACT CON	Windows Mac OS X (Cocoa) Mac OS X (Cocoa) Linux 32bit Linux 64bit St
Eclipse for RCP/Plug-in Developers (183 MB)     A complete set of tools for developers who want to create Eclipse plug-ins or Rich Client     Applications. It includes a complete SDK, developer tools and source code, plus Mylyn, an XML     editor and the Eclipse Communication Framework. More     Downloads: 17,776	Windows     M       Mac OS X (Carbon)     Vi       Mac OS X (Cocoa)     Vi       Linux 32bit     Gi       Linux 64bit     Gi
	2



3)画面の下矢印をクリックしダウンロード



4) ダウンロードしたファイル"eclipse-cpp-**galileo**-win32.zip"を解凍し、そのなか の"eclipse"フォルダを適当な場所(C:¥eclipse)へ移動する。

5) Eclipse を起動する。





#### 6) 最初に Workspace の場所を聞いてきます。適当なフォルダに変更してください。

🖶 Worksp	ace Launcher	
Select a v Eclipse sto	<b>vorkspace</b> res your projects in a folder called a workspac	be.
Choose a w	vorkspace folder to use for this session.	230
Workspace:	C:¥workspace	Browse
<mark></mark>	as the default and do not ask again	
$\smile$		OK Cancel

#### 画面の Workbench をクリックします。





€ C/C++ - Eclipse Platform				
Eile Edit Refactor Navigate Search Project	: <u>R</u> un <u>W</u> indow <u>H</u> elp			
= + = =	-   🔨 - 🕲 -   🎄 - 🔘 - 0, -   🝅	2 I A 6 I 5 -	8 /	
Project Explorer X			- 0	🗄 Outlin 🐰 💿 Make 🗖 🗆
				An outline is not available.
	🕞 Problems 💥 🧔 Tasks 🖻 Console	Properties		* ▽ - □
	0 errors, 0 warnings, 0 infos			-91
	Description A	Resource Path	Location	
	I			
J∎◆		]		🐴   奠 💌 💖 🔶

### 4.3 プロジェクトを作る

新規プロジェクトを作成するため"File"→"New"→"C Project"を選択します

ile <u>E</u> dit <u>S</u> ource I	Refac <u>t</u> or <u>N</u> avigate	Se <u>a</u> rch <u>R</u> un <u>P</u> roject <u>W</u> indow <u>H</u> elp	
<u>N</u> ew	Alt+Shift+N	C++ Project	
Open File <u>.</u> .	<	C Project	Ď
<u>G</u> lose	Ctrl+W	Project	
Close All	Ctrl+Shift+W	Convert to a C/C++ Project	
Save	Ctrl+S	Source Folder	
📓 Save <u>A</u> s		Carros File	
😭 Sav <u>e</u> All	Ctrl+Shift+S	C Source File	
Revert		File from Template	
Mo <u>v</u> e		G Class	
Rena <u>m</u> e	F2	= Other Other	
🐑 Re <u>f</u> resh	F5		



プロジェクト名を聞かれるので適当な名前(LED)を入力し Finish ボタンを押します。

🖨 C Project	
C Project Create C project of selected type	
Project name: LED	
✓ Use <u>d</u> efault location	
Location: C:¥zero¥Example-2148GCC¥18.US	Btarget¥LED Browse
Project type:	Toolchains:
🖃 👝 Makefile project	Other Toolchain
✓ Show project types and toolchains only if	they are supported on the platform
(?) < <u>B</u> ack	<u>N</u> ext > <u>F</u> inish Cancel



Project Explorer にプロジェクト LED が追加されましたが中身が何もないので、"×"がついています。



サンプルのフォルダ Example-2148GCC¥1. LED のなかのファイルを "C:¥workspace¥LED"にコピーしてください。

Eclipse の "File" "Refresh"を選択します。





Project Explorer の"LED"プロジェクトの左にある + をクリックするとファイルの一覧が 表示されます。



### 4.4 Eclipse プラグイン(Zylin Embedded CDT)インストール

€ C/C++ - Eclipse <u>File Edit Source Refactor Navigate Search Run Project Window Help</u> 🗄 📬 🕈 📓 👘 🖸 🕈 🖓 🕶 👩 🗸 🎯 🕶 🔆 🦠 🗸 🔞 💆 👑 elcome Project Explorer 🙁 📃 🗖 (?) Help Contents 3 Search Dynamic Help 🖃 💕 LPC2388\_LED Ctrl+Shift+L 🗄 🔝 crt.s Key Assist... 🗄 h LPC23xx.h Tips and Tricks... 🗄 💽 main.c 💣 Report Bug or Enhancement... 🗄 🚡 typedefs.h Cheat Sheets... -) lpc2388\_ram.ld -]] lpc2388\_rom.ld Check for Updates 🔚 🐻 makefile Install New Software About Eclipse

Eclipse の"Help"→"Install New Software"を選択します



#### Add ボタンを押します。

🖨 Install		
Available Software Select a site or enter the loca	ation of a site.	
Work with: <sup>O</sup> type or select a	site	
	Find more sot	Itware by working with the <u>Available Software Sites</u> preferences.
type filter text		
Name	Version	
Details		
Show only the <u>l</u> atest versio	ins of available software	☐ <u>H</u> ide items that are already installed
Group items by category		What is <u>already installed</u> ?
☑Contact all update sites du	ring install to find required so	oftware
0		<a>Back Next &gt; Einish Cancel</a>

Add Site の"Location"に"http://opensource.zylin.com/zylincdt "と入力し OK ボタンを押す。

🖨 Add	iite	
<u>N</u> ame:		] <u>L</u> ocal
Location:	http://opensource.zylin.com/zylincdt	<u>A</u> rchive
(?)	ОК	Cancel



Install に"http://opensource.zylin.com/zylincdt "が追加されるのでチェックボックスにク リックしチェックを入れて Next ボタンを押す。

🖨 Install	
Available Software Check the items that you wish to install.	
Work with: http://opensource.zylin.com/zylincdt	tware by working with the 'Available Software Sites' preferences
	Invarie by working with the <u>invariable bertiture orice</u> , preferences.
type filter text	
Name	
Details	
Show only the latest versions of available software	Hide items that are already installed
Group items by category	What is <u>already installed</u> ?
☑ <u>C</u> ontact all update sites during install to find required so	oftware
<b>?</b>	< <u>B</u> ack <u>N</u> ext > <u>E</u> inish Cancel



🖨 Install			
Install Details Review the items to be installed.			
Name	Ver	Ы	
🖗 Zylin Embedded CDT	4.8.1	com.zylin.cdt.feature.feature.group	
Size: Unknown Details			
?		< Back Next > Eini	sh Cancel



🖨 Install	
<b>Review Licenses</b> Licenses must be reviewed and accepted before the software can be installed.	
License text:	
ECLIPSE FOUNDATION SOFTWARE USER AGREEMENT March 17, 2005 Usage Of Content	<b>*</b>
THE ECLIPSE FOUNDATION MAKES AVAILABLE SOFTWARE, DOCUMENTATION, INFORMATION AND/OR OTHER MATERIALS FOR OPEN SOURCE PROJECTS (COLLECTIVELY "CONTENT"). USE OF THE CONTENT IS GOVERNED BY THE TERMS AND CONDITIONS OF THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. BY USING THE CONTENT, YOU AGREE THAT YOUR USE OF THE CONTENT IS GOVERNED BY THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF ANY APPLICABLE LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. IF YOU DO NOT AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT AND THE TERMS AND CONDITIONS OF ANY APPLICABLE LICENSE AGREEMENTS OR OF ANY APPLICABLE LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW, THEN YOU MAY NOT USE THE CONTENT.	
Applicable Licenses	
Unless otherwise indicated, all Content made available by the Eclipse Foundation is provided to you under the terms and conditions of the Eclipse Public License Version 1.0 ("EPL"). A copy of the EPL is provided with this	~
OI accept the terms of the license agreement	
OI <u>d</u> o not accept the terms of the license agreement	
Image: Second	Cancel

🖨 Install	
(i)	
Fetching com.zylin.embeddedcdt_4.8.1.jar/plugins/com.zylin.embeddedcdt_4	.8.1.jar
Always run in background	
🛞 Install	
Fetching com.zylin.embeddedcdt_4s/com.zylin.embeddedcdt_4.8.1.jar	ב <b>ו</b>
Run in <u>B</u> ackground Cancel <u>&lt;&lt; D</u> e	tails





インストール完了したら、Yes ボタンを押して、Eclipse を再起動させます。

### 4.5 ビルドの設定

Eclipse の"Project"→"Preferences"を選択する。

🖶 C/C++ - LPC2388_LED/main.c - Ec	lipse		
<u>File Edit Source Refactor N</u> avigate Se <u>a</u>	rch <u>R</u> un	<u>Project Window H</u> elp	
	• • • %	Op <u>e</u> n Project Clo <u>s</u> e Project	B 14
Project Explorer 🛛 🗖 🕞 🗸	main.c	Build <u>All</u> Build Configurations	Ctrl+B
EPC2388_LED	/* F <b>void</b> {	Build Corrigutations Build Project Build <u>W</u> orking Set Clea <u>n</u> Build Auto <u>m</u> atically	▶.EDs
	3	Make Target <u>P</u> roperties	, TM i
Crto - [arm/le]     LPC2388_LED_rom.elf - [arm/le]	/* F void	unction that turns LED_On (char num)	en requeste



Preferences の"Build"を選択し"Scanner Configuration Builder"のチェックマークを外し て

Properties for LED		
Properties for LED type filter text Besource Builders C/C++ General Project References Refactoring History Run/Debug Settings Task Repository WikiText	Builders Configure the builders for the project:	
•	OK	Cancel



Preferences の"C/C++ Build"→"Settings"を選択し"Binary Parsers"タブの"GNU Elf Parser"にチェックを入れて OK ボタンを押します





### 4.6 ビルド

Eclipse の"Project"→"Build Automatically"のチックを外してください。

🖨 C/C++ – Eclipse			
<u>F</u> ile <u>E</u> dit <u>S</u> ource Refac <u>t</u> or <u>N</u> avigate Se <u>a</u>	rch <u>R</u> un	Project Window	Help
	• : %	Op <u>e</u> n Project Clo <u>s</u> e Project	Þ
Project Explorer 🕄 🗖 🗖		Build <u>A</u> ll	Ctrl+B
E E LPC2388_LED E S crt.s E LPC23xxh E M LPC23xxh E main.c		Build Configur Build Project Build <u>W</u> orking Clean	rations  Set tically
in typedefs.h ↓ Ipc2388_ram.ld ↓ Ipc2388_rom.ld ↓ Ipc2388_rom.ld ↓ makefile		Make Target <u>P</u> roperties	•

"Project" "Clean"を選択するクリアが行われます。





🖨 Clean	
Clean will discard all build pro build occurs the projects will	oblems and built states. The next time a be rebuilt from scratch.
⊙Clean <u>a</u> ll projects	○Clean projects <u>s</u> elected below
🗹 😂 LED	
Carne - knild immediately	
Build the entire workers	
Build only the selected r	
O band only the selected p	
	UK Cancel

「Start a build immediately」のチェックマークを外して、「Ok」を押します。

🖨 C/C++ - LPC2388_LED/main.c - Ec	lipse		
<u>Eile E</u> dit <u>S</u> ource Refac <u>t</u> or <u>N</u> avigate Sea	arch <u>R</u> un	<u>Project Window H</u> elp	5
1 [] - [] () [] () () () () () () () () () () () () ()	ş -	Op <u>e</u> n Project Clo <u>s</u> e Project	*
Project Explorer 🛛 🖓 🖓	C main.c	Build <u>A</u> ll Build Configuration	Ctrl+B
🖻 🚖 🔝 🏹 🖻 🚰 LPC2388_LED	<b>#inc</b> .	<u>B</u> uild Project Build <u>W</u> orking Set Clea <u>n</u>	FDe
⊞⊶.S crt.s ⊞⊶.h LPC23xx.h ⊞⊶.c main.c	void (	Build Auto <u>m</u> atical Make Target	<u>v</u>
i∎ini typedefs.h i∋i lpc2388_ram.ld	3	<u>P</u> roperties	IM
makefile	/* F void {	unction that tur LED_On (char nu FIO2SET = FIO2SE	ns on request m) T   num;

"Project" "Build All"を選択するとビルドが行われます。



🖨 Building Workspace	
Duilding all	
(**********************************	
Always run in background	
Run in <u>B</u> ackground Cancel	<u>D</u> etails >>

コンパイル中です。



コンパイルが成功すれば、実行ファイル LED.elf と LED.hex を生成されます。

### 4.7 GDB の設定

Eclipse の"Run" "Debug Configurations..."を選択します。



🖨 C/C++ – Eclipse	
<u>E</u> ile <u>E</u> dit <u>S</u> ource Refac <u>t</u> or <u>N</u> avigate Se <u>a</u> rch	<u>Run</u> Project <u>W</u> indow <u>H</u> elp
i 📬 • 🔛 👜   🗟 i 💣 • 🚳 • 💣 • 🐨 •	Q <sub>6</sub> <u>R</u> un Ctrl+F11 ™ <u>5</u> Debug F11
	- Run His <u>t</u> ory ► - Run A <u>s</u> ► Ru <u>n</u> Configurations
in - ∰ Binaries in - ic LED.c in - ☆ LED.elf - Farm /le]	Debug <u>H</u> istory Debug As
ELD.en [ann/le]	Debug Configurations
LED.dmp LED.hex LED.map Makefile	<ul> <li>Toggle Breakpoint</li> <li>Ctrl+Shift+B</li> <li>Toggle Line Breakpoint</li> <li>Toggle Method Breakpoint</li> <li>Toggle Wetebooint</li> </ul>

Debug Configurations の"Zylin Enbedded debug(Native)"を右クリックし"New"を選択す

#### る。

🖶 Debug Configurations		
Create, manage, and run cont	ligurations	Ť.
Image: Second secon	Configure launch settings from this dialog: Press the 'New' button to create a configuration of the selected type. Press the 'Duplicate' button to copy the selected configuration. Press the 'Delete' button to remove the selected configuration. Press the 'Filter' button to configure filtering options. Press the 'Filter' button to configure filtering options. Press the 'Filter' button to configuration by selecting it. Deletered aunch perspective settings from the <u>Perspectives</u> preference page.	



Name に適当な名前を入れる。例、"gdb"と入れます。Main タブの"Project"に"LED"、"C/C++ Appication:"に"C:\#workspace\#LED\#LED.elf"と入力します。

🖨 Debug Configurations		K
Create, manage, and run cor	ligurations 🔅	
Image: Second state st	Nature: [gdb Main	
Filter matched 7 of 7 items	Apply Revert	
?	Debug Close	)



Debugger タブの"GDB debugger:"に"arm-elf-gdb"、"GDB command file:"に何も入力しま せん。

Debug Configurations	
Create, manage, and run cont	igurations to the second se
Image: Second state st	Name: edb Main Debueger: Commands & Source Common Debugger: Encedded GDB Main GDB debugger: C¥Program Files¥yagarto¥bin¥arm-elf-gdb.exe Browse GDB command file: (Warning: Some commands in this file may interfere with the startup operation of the debugger, for example "run") GDB command set: Standard Protocol: mi Use full file path to set breakpoints
Filter matched 7 of 7 items	Apply Revert
?	Debug Close



Commandsタブの"'Initialize' commands"に下記の画面の様に入力します target remote localhost:3333 monitor halt monitor step load break main

nnn	÷.	n	110	
0011	•		40	

Debug Configurations		
Create, manage, and run con	figurations	-
Image: Second system         Image: Second system <td>Name: @db            Main ☆ Debugen © Commands          Help/tips on how to setup GDB init script         'Indicate' commands         target remote localhost:3333         load         break main         continue</td> <td></td>	Name: @db            Main ☆ Debugen © Commands          Help/tips on how to setup GDB init script         'Indicate' commands         target remote localhost:3333         load         break main         continue	
Filter matched 7 of 7 items	Apply	Re <u>v</u> ert
?	Debug	Close

Common タブの"Display in favorites menu"の Debug にチェックを入れます。全てを入力 し終えたら"Apply"ボタンを押し、"Close"ボタンを押します。



Debug Configurations	
Create, manage, and run con	figurations
Image: Second state st	Name: gdb         Main St Debugger Commands Source Common         Save as         O Lgcal file         Shared file:         Display in favorites menu         Observe         Observe         Observe         Other         Standard Input and Output         Allocate Console (necessary for input)         File:         Workspace         File:         Workspace         File System
Filter matched 7 of 7 items	Apply Revert
?	Debug Close

全てを入力し終えたら"Apply"ボタンを押し、"Close"ボタンを押します。



### 4.8 OpenOCD の設定

Eclipse の"Run" "External Tools." "External Tools Configrurations..."を選択します。

🖨 C/C++ - LPC2388_LED/main.c - Eclipse			
<u>File Edit Source Refactor Navigate Search</u>	Run Project Window Help		
: C*	ଷ୍ଟ୍ର <u>R</u> un Ctrl+F11 *©Debug F11	• 0 • 9 •   😕 😂 🛷 •   🛃	
Project Explorer 🛛 🖓 🖓 🏹	Run His <u>t</u> ory Run A <u>s</u> Ru <u>n</u> Configurations…		
	Debug <u>H</u> istory Debug As De <u>b</u> ug Configurations	*/	
	Toggle Breakpoint Ctrl+Shift+B     Toggle Line Breakpoint     Toggle Added Declarint	interface, enable LEDs */	
	<ul> <li>Foggle Watchpoint</li> <li>Skip All Breakpoints</li> <li>Remove All Breakpoints</li> <li>Breakpoint Types</li> </ul>	ed LED */	
LPC2388_LED_rom.map	🤼 External Tools	(no launch history)	
È lpc2388_ram.ld È lpc2388_rom.ld È main.lst È makefile	<pre>void LED_Off (char num) {     FIO2CLR = FIO2CLR   num; }</pre>	Barr As External Tools Configurations. Organize Fayorites	



External Tools Configrurations の"Program"を右クリックし、"New"を選択します。

🚝 External Tools Configuratio	ins X
Create, manage, and ru Run a program	n configurations
Filter matched 1 of 1 items	Configure launch settings from this dialog:
0	<u>Run</u> Close



Main タブの"Name"に適当な名前を入力してください。私は " OpenOCD"と入れました。 "Location:"に"C:¥Program Files¥OpenOCD¥0.1.0¥bin¥openocd"、

"Working Directory:"C:¥Example-2148GCC¥open-jtag",

"Arguments:"にfopen-jtag.cfg -flpc2148.cfg と入力します。

🖨 External Tools Configurat	ions 🛛 🗙
Create, manage, and run cor Run a program	nfigurations O
Image: Second system         Image: Second system	Main       Refresh       Build       Environment       Qommon         Doctor:       C:\Program Files\#OpenOCD\#0.1.0\#bin\#openocd.exe       Browse Work.space       Browse File System       Variables         Working       Director:       C:\Program Files\#OpenOCD\#0.1.0\#bin\#openocd.exe       Browse Work.space       Browse File System       Variables         Working       Director:       C:\Program Files\#0penocd.exe       Browse Work.space       Browse File System       Variables         Areuments:       -f open-jtag.cfg - f lpc2148.cfg       Image: Common test and test
Filter matched 2 of 2 items	
?	Run Close



🖨 External Tools Configura	tions	
Create, manage, and run co Run a program	nfigurations	
Image: Second system         Image: Second system	Mame: OpenOCD  Main Refresh Build Environment Common Build before launch  The grotere workspace The grotect containing the selected resource Specific projects  Include referenced projects	Projects
Filter matched 2 of 2 items		Apply Reyert

Build タブをクリックし"Build before launch"にチェックを外れます。



🖨 External Tools Configurat	ions 🛛 🗙
Create, manage, and run cor Run a program	nfigurations Organizations
Image: Second system         Image: Second system	Name: openOCD Main Refresh Build Environmett Common Save as O Lgcal file Shared file: Display in favorites menu Oberault - inherited (MS932) O ther Standard Input and Output Standard Input and Output Allocate Console (necessary for input) File: Workspace File System Variables Append Launch in background
Filter matched 2 of 2 items	Apply Revert
?	Bun Close

Common タブをクリックし"Display in favarites menu"の"External Tools"にチェックを入 れます。全てを入力し終えたら"Apply"ボタンを押し、"Close"ボタンを押します。

### 4.9 デバッグ

電源投入

- 1. OpenJTAG をターゲット(LPC2148 ボード)とパソコンに接続
- 2. ターゲットに電源を入れます



External Tools の▼ボタンをクリックし、OpenOCD を選択



Console ウインドに下記のメッセージが出力



Debugの▼ボタンをクリックし、"gdb"を選択。







Yesボタンを押して、デバッグが開始します。







ステップ実行において良く使われる操作の一覧を以下に示します。

操作名	ショートカットキー
再開	F8
ステップイン	F5
ステップオーバー	F6
ステップ・リターン	F7

ステップ実行とは関係ありませんが、前回起動したクラスを再度実行したデバッグする場合は、 以下のショートカットキーが便利です。

操作名	ショートカットキー
前回の起動を実行	Ctrl + F11
前回の起動をデバッグ	F11

ブレークポイントでプログラムが中断した状態から、次のブレークポイントまで実行させたり、1行ずつ実行させたりできます。コードを繰り返して実行することにより、LED ランプが点滅します。





ダブルクリックで breakpoint を設置/キャンセルします。

### 4.10 デバッグ終了

1)gdbの停止

Debug ウインドウの gdb の Thread を選択し、停止ボタンと押します



2) OpenOCD の停止

Debug ウインドウの OpenOCD の Thread を選択し、停止ボタンと押します





3)電源停止

ターゲットの電源を停止

4) OpenJTAG をターゲットから取り外す

5)上記が面倒であれば Eclipse を終了しターゲットの電源停止、open-JTAG を取り外し でも OK です。

### ほかのサンプルは Example-2148GCC.rar を参照してください。



### 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。10BASET-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出来 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。10BASET-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: WebServer/WebServer.bin



このサンプルを LPC2148 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.j p」を入力すると

🥹 Mozilla Firefox		
ファイル(E) 編集(E) 表示(V) 履歴(S) ブックマーク(B) ツール(T) ヘルブ(H)		
C X 🟠 ( http://192.168.3.250/csun.co.jp/	☆ •	
🖻 よく見るページ 🀢 Firefox を使ってみよう <u>気</u> 最新ニュース 🗋 HotMail の無料サービス 🗋 Windows Media 📄 Windows 🗋 リンクの変更		
Output is: ON [refresh status]		
Switch off		

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

サンブルの IP アドレスとバスワードが 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) ~



### 第六章 USB で ARM9(Linux)に接続

弊社の ARM9(MINI2440)は標準 OS に Linux を採用します。Linux には、信頼性が高いネ ットワークスタックが実装され、利用できます。従って、ネットワークに接続する信頼性 の高い遠隔制御機器が、容易に作成できる利点があります。Linux にも USB スタックが実 装され、多種類の USB デバイスを利用できます。例えば、USB プリンター、USB 無線 LAN、 USB メモリ、SD カードなど。パソコンの Linux 上のアプリケーションが ARM9 上で利用 できます。ゼロから開発せずに、例えば GUI と Web サーバなどが組み込み用機器で利用で きるわけで、これは非常に大きな利点といえます。

Linuxの便利さの反面、複雑、重い、反応速度が遅いです。反応速度は大体数十ms ぐらいです。この反応速度は人間との会話に満足できますが、機械制御のリアルタイム性に足りないかもしれません。

ARM7 シリーズはリアルタイム制御に向けのマイコンです。OS なしあるいは簡単な RTOS を搭載します。1us~1ms 以上の反応速度が実現できます。ARM7 は制御のため、豊富な制 御用の周辺機能を持ちます。例えば AD, DA, PWM, GPIO, カウンターなど。

ARM7/LPC2148 には USB ターゲットポートがあります。最大通信速度 12Mbps。LPC2148 は USB デバイスとして使えます。ARM9 は USB ハブを経由すれば、何台分の LPC2148 にも接続できます。拡張性がいいです。パソコンも LPC2148 デバイスを使えます。

システムは ARM9/MINI2440 と ARM7/LPC2148 を同時に採用すれば、Linux の便利な機 能と ARM7 のリアルタイム性を組み合わせ、高度複雑なアプリケーションとリアルタイム 制御が両立できるシステムを作れます。





写真は MINI2440 と LPC2148 を USB で繋ぐ様子。



LPC2148 は CDC ACM デバイスとして動きます。ARM9 側から見ると、LPC2148 は USB シリアルポートです。ARM9 のターミナルで下のコマンドを入力

#### # armcomtest -d /dev/ttyACM0

キーボードで情報を入力します。ARM9はこの情報をARM7へ送信します。ARM7はARM9 が送った情報をそのまま ARM9へ返信します。ARM9はこの情報をコンソールで表示しま す。

LPC2148側のサンプルは**USBtarget/examples/serial.hex** です。

ARM9の Linux のコンフィグの手順:

下のコマンドで Linux のコンフィグに入ります。

\$ cd linux-2.6.29

\$ make clean

\$ cp config\_mini2440\_n35 .config

\$ make menuconfig

	zqing@localhost:~/linux-2.6.29	+ ×
ファイル( <u>F</u> ) 編集(	E) 表示(⊻) 端末(Ĭ) ヘルプ(H)	
.config - Linux	Kernel v2.6.29 Configuration	
Arrow keys Highlighted <m> modular for Search.</m>	Linux Kernel Configuration navigate the menu. <enter> selects submenus&gt;. letters are hotkeys. Pressing <y> includes, <n> excludes, izes features. Press <esc><esc> to exit, <? > for Help, Legend: [*] built-in [] excluded <m> module &lt; &gt;</m></esc></esc></n></y></enter>	
CPU Floa Use Floa Floa	Power Management> ating point emulation> rspace binary formats> er management options> working support>	
V(+)	ice Drivers> e systems nel hacking> urity options> ptographic API>	
	<mark><select></select></mark> < Exit > < Help >	

「Device Drivers」を選択、





「USB support」を選択、

🖻 zqing@localhost:~/linux-2.6.29	× 38
ファイル(E) 編集(E) 表示(V) 端末(I) ヘルプ(日)	
.config - Linux Kernel v2.6.29 Configuration	
USB support Arrow keys navigate the menu. <enter> selects submenus&gt;. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes, <m> modularizes features. Press <esc><esc> to exit, <? > for Help,  for Search. Legend: [*] built-in [] excluded <m> module &lt; &gt;</m></esc></esc></m></n></y></enter>	
<pre>&lt;    OXU210HP HCD support &lt;    ISP116X HCD support &lt;*&gt;    OHCI HCD support &lt;    SL811HS HCD support &lt;    SL811HS HCD support &lt;    R8A66597 HCD support &lt;    Host Wire Adapter (HWA) driver (EXPERIMENTAL) &lt;    Host Wire Adapter (HWA) driver (EXPERIMENTAL) &lt;    Inventra Highspeed Dual Role Controller (TI, ADI,) *** USB Device Class drivers *** &lt;    USB Modem (CDC ACM) support &lt; &gt;    USB Printer support</pre>	
<pre>     v(+)     <select> &lt; Exit &gt; &lt; Help &gt; </select></pre>	

「USB Modem(CDC ACM) support」を選択します。"Exit"& "Save"します。



下のコマンドで Linux カーネルをコンパイルします。

#### \$ make zImage

コンパイル完了すれば、arch/arm/boot フォルダには CDC ACM を含むカーネル zImage を生成します。この zImage ファイルを ARM9 に書き込んでください。詳しい情報は ARM9 のマニュアルをご参照ください。

### 第七章 微弱無線モジュール nRF24L01(2.4GHz)



弊社が販売している微弱無線モジュール nRF24L01(2.4GHz)

- Nordic Semiconductor 社の NRF24L01 を搭載
- 最大伝送速度: 2Mbps、VoIP 応用可能
- 使用周波数带: 2.4GHz ISM 带
- 変調方式: GFSK
- 定格電圧: 1.9~3.6V
- SPI インターフェース
- 使いやすい 2.54mmDIP

#### nRF24L01 モジュールと LPC2148 モジュールの接続:

nRF24L01	LPC2148 モジュール	nRF2401	LPC2148 モジュール
1. GND	26. GND	2. VCC	52. 3.3V
3. CE	43. P1.24	4. CSN	9. P0.7(SSEL0)
5. SCK	6. P0.4(SCK0)	6. MOSI	8. P0.6(MOSI0)
7. MISO	7. P0.5(MISO0)	8. IRQ	44. P1.25(EXTIN0)

始めの数字はモジュールのピン番号です。





受信指示 LED: LPC2148 モジュールの PIN34(P0.31)と PIN27(VBAT/3.3V)

### 送信用のプログラム: nRF24L01/send.bin

### 受信用のプログラム: nRF24L01/rev.bin

この実験は二つのセット(LPC2148+nRF24L01)が必要です。一つは送信です。もう一つは受 信です。送受信成功すれば、受信指示 LED が点滅します。