

# ARM Cortex-M3 LM3S9B92 開発キットマニュアル

## 株式会社日昇テクノロジー

http://www.csun.co.jp

info@csun.co.jp

2011/12/21



copyright@2011-2012





NO	バージョン	修正内容	修正日
1	Ver1.0	新規作成	2011/11/10
2	Ver1.1	MCU バージョンアップ (C1->C5)	2011/12/21

#### • 修正履歴

※ 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。

[<u>http://www.csun.co.jp</u>]

※ (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 LM3S9B92 開発キット概要5
1.1 LM3S9B92 マイコンの主な特徴5
1.2 基板ボード仕様5
1.3 拡張ボード(LCD 搭載)仕様6
第二章 回路の説明
2.1 Power
2.2 USB Host
2.3 USB Device
2. 4 USB COM
2.5 CAN
2.6 IIC EEPROM
2.7 SPI Flash
2.8 SD Card
2.9 LED
<b>2.10</b> Key
<b>2.</b> 11 JTAG
2.12 EXTEND I/0
<b>第三章 開発ツール KEIL の応用</b> 14
3.1 KEIL のインストール14
3.2 環境の設定16
3.2.1 ダウンロード関連16
3.2.2 デバッグ関連 20
第四章 サンプルソースについて22
4.1 Code¥StellarisWare¥boards¥Smart-9B92-C3¥22
4.1.1 ¥blinky
4.1.2 ¥uart_echo
4.1.3 ¥softuart_echo 25
4.1.4 ¥timers
4.1.5 ¥watchdog
4.1.6 ¥sd_card 26
4.1.7 ¥usb_dev_serial
4.1.8 ¥usb_dev_msc



	4.1.9 ¥usb_dev_mouse	29
	4.1.10 ¥usb_dev_keyboard	30
	4. 1. 11 ¥usb_dev_bulk	30
	4. 1. 12 ¥usb_dev_chidcdc	31
	4.1.13 ¥usb_host_mouse	32
	4.1.14 ¥usb_host_keyboard	33
	4.1.15 ¥usb_host_msc	34
	4.1.16 ¥enet_uip	35
	4.1.17 ¥enet_lwip	37
	4.1.18 ¥enet_ptpd	38
	4. 1. 19 ¥qs-blox	39
	4.1.20 ¥idm-checkout	41
	4.1.21 ¥hello_lcd	<b>19</b>
	4.1.22 ¥hello_widget	50
	4.1.23 ¥calibrate	50
	4.1.24 ¥scribble	52
	4.1.25 ¥lang_demo	53
	4.1.26 ¥showjpeg	53
	4.1.27 ¥i2s_demo	54
4	2 Code¥ek-1m3s9b92-C3¥RL¥TCPnet¥	54
	4.2.1 ¥Http_demo	55
	4.2.2 ¥LEDSwitch	58
	4. 2. 3 ¥DNS_demo	59
	4.2.4 ¥Telnet_demo	30
付籤	と: OpenLink で実行ファイルの書き込む	33
	1 ドライバのインストール	33
	2 J-FLASH ARM で実行ファイルを書き込む	33



#### 第一章 LM3S9B92 開発キット概要

ARM コア新型プロセッサーCortex-M3 を採用した TI 社の LM3S9B92(100MHz 周波数、256KB Flash、96KB SRAM)。

標準外付け: JTAG/SWD、Ethernet、CAN 2.0、USB2.00TG/Host/Device、USBシリアル(TTL) 変換、SDIO、EEPROM、SPI、TFT LCDなど。

#### 1.1 LM3S9B92 マイコンの主な特徴

ARM コア新型プロセッサーCortex-M3 を採用、最大周波数100MHz

256KB Flash、96KB SRAM 内蔵

最大 65 個の GPI0

10/100M Ethernet MAC及びPHY

CAN 2.0 A/B

USB 2.0 OTG/Host/Device

- 2 UARTs、IS07816 サポート
- 2 SPI
- 2 IIC
- 1 IIS
- 8 PWM出力
- 2 10-bit A/D
- 8/16/32ビット専用パラレルバス、SDRAM、SRAM、Flash、FPGAs、CPLDをサポート 関数ライブを内蔵

#### 1.2 基板ボード仕様

JTAG/SWD デバッグ用インタフェース(10pin、2.54mmピッチ) 10/100M Ethernetインタフェース(DP83848) USB2.0デバイスインタフェース(OTG) USB2.0ホストインタフェース USBシリアルTTL変換インタフェース(CP2102) CAN2.0インタフェース x 1 SDカードインタフェース シリアルEEPROM(24C04) SPIシリアルFLASH (W25Q80) TTL LCDインタフェース 外部バスインタフェース ューザーボタン x 1 ューザーLED x 2



5V/1A DC外部電源インタフェース、極性:センタープラス 外形寸法: 74×64(mm) ※突起物は除く

#### 1.3 拡張ボード(LCD搭載)仕様

タッチパネル付きLCD搭載 Audioプレイ機能搭載(WM8510)、イヤホン搭載 8MB SDRAM(MT48LC4M16A2P-7E) 外形寸法: 77×69(mm) ※突起物は除く



第二章 回路の説明



#### 2.1 Power







入力:5V DC 出力:3.3V DC 本ボードはJ2のDC2.1で給電できます。



或いは J7、J8の設定で Mini USB より給電できます。J8 はデフォルトでショートしており ます。USB ケーブルで"USB-DEVICE"の Mini USB からボードに給電できます。



2.2 USB Host





LM3S9B92 の USB Device と USB Host は共用していますので、J10 と J11 で選択します。Host として使用時は J10、J11 の 2-3 をショートします。



#### 2.3 USB Device



J8はデフォルトで0 $\Omega$ の抵抗を設置しております。このインタフェースによりボードに給 電できます。

LM3S9B92のUSB DeviceとUSB Hostは共用していますので、J10とJ11で選択します。Device として使用時はJ10、J11の1-2をショートします。



#### 2.4 USB COM







U3はUSBシリアル変換用でシリアルポートないPCなどでシリアルプログラムのデバッグができます。

J7 はデフォルトでオープン。ボードに電源供給はできません。

#### 2.5 CAN





#### 2.6 IIC EEPROM







#### 2.7 SPI Flash

+3.	. <u>3</u> V	U9		+3.	<u>3</u> V
FLASH_CSn	1	CS	VCC	8	
SSIORX	2		HOLD	7	
	3	$\overline{WP}/\Lambda CC$	SCLK	6	SSI0CLK
	4	GND	SU/SIO0	5	SSI0TX
		UND	51/5100		
		MX25L3205	D/W25Q80I	BVSSIG	
GI	ND				



#### 2.8 SD Card



SPI インタフェースで SD カードと通信します。

#### 2.9 LED



I/Oを節約する為、イーサネットの緑色と黄色 LED を普通の LED と共用します。



2.10 Key



RSTと標示しているのはReset キー。SW は普通のユーザーキー、IIC インタフェースと共用。

#### 2.11 JTAG





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

#### 2.12 EXTEND I/0



本ボードは LM3S9B92 の大部分の I/0 (イーサネット及び USB 以外)を引き出しています。



#### 第三章 開発ツールKEILの応用

本章ではご参考まで KEIL の無償評価版 MDK315B でインストール手順を説明します。 Keil社のHP (<u>http://www.keil.com/</u>) から最新版がダウンロード出来ます。

3.1 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。 Setup RealView Microcontroller Development Kit V3.15b

Welcome to Keil IÉVision3 Release 11/2007	An ARM <sup>®</sup> Company
This SETUP program installs:	
RealView Microcontroller Development Kit V3.15b	
This SETUP program may be used to update a previous product installation. However, you should make a backup copy before proceeding.	
It is recommended that you exit all Windows programs before continuing with SETUF	P.
Follow the instructions to complete the product installation.	
Keil µVision3 Setup	
<< Back	Next>> Cancel

「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。

icense Agreement Please read the following license agreement carefully.		An	ARM <sup>®</sup> Compa
To continue with SETUP, you must accept the terms of the Licer check box below.	nse Agreement. To acce	ept the agreement,	click the
End-User License Agreement for AR Tools	M Keil <mark>S</mark> oftwa	re Develop	ment 🚔
THIS END USER LICENCE AGREEMENT ("LICENCE") (EITHER A SINGLE INDIVIDUAL, OR SINGLE LEGAL E THE USE OF THE SOFTWARE ACCOMPANYING THIS I LICENSE THE SOFTWARE TO YOU ON CONDITION TH THIS LICENCE. BY CLICKING "I AGREE" OR BY INST	IS A LEGAL AGREEN ENTITY) AND ARM L LICENCE, ARM IS O HAT YOU ACCEPT A FALLING OR OTHER	MENT BETWEEN IMITED ("ARM") NLY WILLING TO LL OF THE TERM WISE USING OR	YOU FOR D IS IN
☑ Lagree to all the terms of the preceding License ∆greement			



etup RealView Microcontroller Development Kit V3.	.156
Folder Selection Select the folder where SETUP will install files.	An ARM <sup>®</sup> Company
SETUP will install $\mu$ Vision3 in the following folder.	
To install to this folder, press 'Next'. To install to a different folder, pre	ess 'Browse' and select another folder.
Destination Folder	Browse
Je. wei	
- Keil µVision3 Setup	<pre>&lt;&lt; Back (</pre>

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。

Customer Information Please enter your in	<b>n</b> formation.	An ARM® Company
Please enter your na	me, the name of the company for whom you work and	l your E-mail address.
First Name:	csun	
Last Name:	Jup	
Company Name:	Josun	
Company Name: E-mail: Keil ptVision3 Setup	info@csun.co.jp	

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。



Setup RealView Microcontroller Development Kit \	/3.15b		
Setup Status		An /	ARM <sup>®</sup> Company
$\mu$ Vision Setup is performing the requested operations.			
Install Files			
Installing STR71x.s.			
Keil µVision3 Setup	Kack	Next >>	Cancel

インストール中の画面です。

Keil <b>(</b> ÉVi	sion3 Setup completed		Ar	ARM <sup>®</sup> Company
μVisior	Setup has performed all requested operations su	uccessfully.		
G. Cha	Delesse Meter			
J★ SUD	A melease Notes.			
j∙ snu	W helease Notes.			
le sno	w nelease Notes.			
IV Add	w nelease Notes. example projects to the recently used project list. reselect Example Projects for	5 		
I Add	w helease Notes. example projects to the recently used project list. reselect Example Projects for			
Add	w helease Notes. example projects to the recently used project list. reselect Example Projects for imulated Hardware		<u> </u>	
V Sho	w helease Notes. example projects to the recently used project list. reselect Example Projects for imulated Hardware /ision3 Setup		<u> </u>	

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社の日本代理店とご連絡ください。

#### 3.2 環境の設定

#### 3.2.1 ダウンロード関連

サンプルソースで例を挙げます。

プロジェクトファイルCode¥ek-1m3s9b92¥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。

"Target Options" をクリックする。

inn Configure target options							
Options for Target 'blinky'							×
Device Target Output List	Device Target Output Listing User   C/C++   Asm   Linker   Debug   Utilities						
Luminary Micro LM3S9B92							
Operating system: None		•	Us Vs	e Cross-Moo e MicroLIB	dule Optimizat 	tion Big Endian	
5 //01/10			Us	e Link-Time	Code Genera	tion	
default off-chip Start	Size	Startup	default	off-chip	Areas Start	Size	Nolnit
ROM1:		0		RAM1:			
ROM2:		0		RAM2:			
ROM3:		0		RAM3:			
on-chip	0+40000	_		on-chip	-20000000	0.19000	
	0000	- (			x20000000		
		C		IRAM2:			
	OK	Can	icel	Defau	lts		Help

"Debug" タブを選択して、 "Use Target Driver for Flash Programming" の所デバッグ 型を選択します。ULINKを利用する場合は "ULINK Cortex Debugger" 、JLINKを利用する場 合は "Cortex-M3 J-LINK" を選択します。

C/C++ Asm Linker Debug Utilitie Settings © Use: Cortex-M/R J-LINK/J-Tr	es race     ▼ Settings
<ul> <li>● Use: ULINK Cortex Debugger</li> <li>● ULINK Cortex Debugger</li> <li>■ ULINK Cortex Debugger</li> <li>■ RDI Interface Driver</li> <li>Altera Blaster Cortex Debugger</li> <li>Stellaris ICDI</li> <li>Initializatio</li> <li>Signum Systems JTAGjet</li> <li>Cortex-M/R J-LINK/J-Trace</li> <li>ST-Link Debugger</li> <li>ULINK Pro Cortex Debugger</li> <li>NULink Debugger</li> </ul>	<ul> <li>✓ Use: Cortex-M/R J-LINK/J-Trace</li> <li>✓ ULINK Cortex Debugger RDI Interface Driver</li> <li>Altera Blaster Cortex Debugger Stellaris ICDI</li> <li>Initializatio</li> <li>Signum Systems JTAGjet</li> <li>Cortex-M/R J-LINK/J-Trace</li> <li>ST-Link Debugger</li> <li>ULINK Pro Cortex Debugger</li> <li>NULink Debugger</li> </ul>
"Utilities"タブを選択して、"Use ?	Target Driver for Flash Programming"の所デバ

ッグ型を選択します。ULINKを利用する場合は"ULINK Cortex Debugger"、JLINKを利用す



る場合は"Cortex-M3 J-LINK" を選択します。

Options for Target 'blinky'		
Device   Target   Output   Listing   User	C/C++ Asm Linker Debug Utilities	
Configure Flash Menu Command		
Use Target Driver for Flash Programming		
Cortex-M/R J-LINK/J-Trace	✓ Settings Update Target before	
Configure Flash Menu Command	Use Target Driver for Flash Programming	
Use Target Driver for Flash Programming	Cortex-M/R J-LINK/J-Trace	•
ULINK Cortex Debugger	ULINK Cortex Debugger	
Init File: Stellaris ICDI	Signum Systems JTAGjet	
Cortex-M/R J-LINK/J-Trace	Cortex-M/R J-LINK/J-Trace	
ULINK Pro Cortex Debugger	ULINK Pro Cortex Debugger	
Command: NULink Debugger	Command: NULink Debugger	

「Settings」ボタンを押す。JLINKを利用として時の例を挙げます。

Cortex JLink/JTrace Target Dri	ver Setup			
Debug   Trace   Flash Down	nload			
Download Function	chip 🔽 Program	n	RAM for Algorithm	Size: 0x0800
C Do not Eras	e 🔽 Reseta	and Run		
Programming Algorithm				
Description	Device Type	Device Size	Address Range	
LM3Sxxx 256kB Flash	On-chip Flash	256k	00001000H - 0003FFFFH	

"Flash Download" タブ画面の "Programming Algorithm" にデバイス情報がない場合、 「Add」ボタンを押す。



Cortex JLink/JTrace Target Dr	iver Setup			
Debug   Trace   Flash Down	nload			
Download Function COMP C Erase Full C Erase Sect C Do not Erase Dragramming Algorithm	Chip 🔽 Program ors 🔽 Verify se 🗌 Reset a	n and Run	RAM for Algorithm	Size: 0x0800
Description	Device Type	Device Size	Address Range	
			Start	Size:
	-	Add	Remove	

Add Flash Programming A	lgorithm		X
Description	Device Type	Device Size	A
ATSAM3X GPNVM bits	On-chip Flash	16	
EFM32 Gecko/Tiny Gecko	On-chip Flash	128k	
K8P5615UQA Dual Flash	Ext. Flash 32-bit	64M	=
LM3Sxxx 128kB Flash	On-chip Flash	128k	
LM3Sxxx 16kB Flash	On-chip Flash	16k	
LM3Sxxx 256kB Flash	On-chip Flash	256k	
LM3Sxxx 32kB Flash	On-chip Flash	32k	
LM3Sxxx 64kB Flash	On-chip Flash	64k	
LM3Sxxx 8kB Flash	On-chip Flash	8k	
LPC122x IAP 128kB Flash	On-chip Flash	128k	
LPC11xx/13xx IAP 16kB FI	On-chip Flash	16k	
LPC11xx/13xx IAP 24kB FI	On-chip Flash	24k	
LPC11xx/122x/13xx IAP 32	On-chip Flash	32k	
LPC122x IAP 48kB Flash	On-chip Flash	48k	
LPC122xIAP 64kB Flash	On-chip Flash	64k	
LPC11xx/13xx IAP 8kB Fla	On-chip Flash	8k	-
		001	
	Add	Cancel	

"Reset and Run"の所を選択すると、ダウンロード後すぐ実行します。

Cortex JLink/JTrace Target Driver Setup						
Debug Trace	Flash Download					
– Download F	unction C Erase Full Chip Erase Sectors	<ul> <li>✓ Program</li> <li>✓ Verify</li> </ul>				
	O Do not Erase	Reset and Run				

"Debug" タブを開いて設定します。ダウンロード失敗する時、スピードを下げて、500K或



いは以下に設定してお試しください。

Cortex JLink	/JTrace Ta	rget Driv	ver Setup				
Debug Trace Flash Download							
J-Link / J	-Trace Adap	oter					
SN:		<ul> <li>USB</li> </ul>	#: 0 💌				
Device:							
HW :	V0.00	dll :	V0.00				
FW :	Feb 22	2 2011 20:4	47:17				
Po	Port. Max Clock:						
JTAG V 2MHz V							
			Auto Clk				

RAM for Algorithmのサイズを0x0800になっているか確認する:

Corte	x-M Targe	et Driver Setup				
Deb	ug  Trace	e Flash Down	nload			
	Download Fi	unction			-RAM for Algorithm	
		<ul> <li>Erase Full C</li> <li>Erase Sector</li> </ul>	Chip <b>▼</b> Program ors <b>▼</b> Verify	n	Start: 0x20000000	Size: 0x0800
	<b>b</b> i	O Do not Eras	e 🔽 Reseta	nd Run		
	rogrammin	g Algorithm			1	
	Descriptio	n	Device Type	Device Size	Address Range	
	LM3Sxxx 2	56kB Flash	On-chip Flash	256k	00000000H - 0003FFFFH	

上記設定後、"Dowmload to Flash Memory"を押してダウンロードができます。

🗇 🕮 🎬 🥔 🔜	LOAD	STM32F207 Flash 🛛 💌 🎊 🛔
Project	LOAD	Download t
🗄 🔝 system s	sti	Download code to flash memory

#### 3.2.2 デバッグ関連

「Debug」タブを選択する。

User	C/C++ Asm	Linker	Debug	Utilities		
	Settings	📔 🍽 Use: 🛛	ULINK Corte	ex Debugger	<b>•</b>	Settings

デバッグツールがULINKの場合は"ULINK Cortex Debugger"、JLINKの場合は"Cortex-M3 J-LINK"を選択する。



Output Use: Corte	ex-M/R J-LINK/J-Trace 🗾		Use:	ULINK Cortex Debugger 🗾
ULIN	K Cortex Debugger			ULINK Cortex Debugger
Load A Alter Stell: Initializatio Sign ST-L	nterface Driver a Blaster Cortex Debugger aris ICDI um Systems JTAGjet ex-M/R J-LINK/J-Trace ink Debugger IK Pro Cortex Debugger		Load A	RDI Interface Driver Altera Blaster Cortex Debugger Stellaris ICDI Signum Systems JTAGjet Cortex-M/R J-LINK/J-Trace ST-Link Debugger ULINK Pro Cortex Debugger
Restore NUL	ink Debugger	f r	Restore	NULink Debugger
Hexファイル Device   Tar	主成必要な場合は「Output get Output Listing User	:」タン	ブで設定	宦します。
Sele	ct Folder for <u>O</u> bjects			
Creat	e Executable: .\Flash\Blinky			
<b>▼</b> <u>D</u> €	bug Information			
	eate HEX File			
	b <u>w</u> se mormation			

"Browse Information"の所も選択すると、ソースコードの検索が便利になります。例え ば任意の所でinit\_displayをコールした場合、マウスで右クリックすると "Go To Definition Of init\_display "選択肢があります。

■ <b>static void</b> j /* Add Syst		Select <u>A</u> ll Insert '#include <lpc214x.h>'</lpc214x.h>
<pre>init_io (); init_displa init_TcpNet /* Timer 1 T1TCR = 1; T1MCR = 3; T1MRO = TCN }</pre>	0	Go To Line Insert/Remove Breakpoint Enable/Disable Breakpoint Clear complete Code Coverage Info
L		Go To Definition Of 'init display'
10000 LLL 7		
⊟#else └		Go To Reference To 'init_display'
⊟#else └ □ <b>void</b> init ( <b>vo</b> /* Add Syst		Go To Reference To 'init_display' Outlining

関数のソースと所に飛びます。

```
>/*----- init_display ------
static void init_display () {
    /* LCD Module. 2x16 init*/
#ifdef USE_4BIT_LCD
    LCD_init ();
    LCD_cur_off ();
    upd_display ();
#endif
}
```



#### 第四章 サンプルソースについて

#### 4.1 Code¥StellarisWare¥boards¥Smart-9B92-C3¥

※MCU が C1 のバージョンの場合は Code¥StellarisWare¥boards¥Smart-9B92-C1¥をご参照ください。



rdk-idm-sbc.uvmpw は多プロジェクト管理ファイル、開くと全てのプロジェクトがリストさ れます:



🔣 E:¥lm3s9b92¥Code¥StellarisWare¥b	board:
<u>File E</u> dit <u>V</u> iew <u>P</u> roject Fl <u>a</u> sh <u>D</u> ebug P	e <u>r</u> ipher
	- la
📀 🍱 🎬 🥔 🖳 👯 idm-checkout	× 3
Project	<b>9</b> 🗵
<pre>Image: control of the system of the sys</pre>	

利用するプロジェクトをマウスで右クリックして "Set as Active Project"をクリックすると、そのプロジェクトが利用できます。





#### 4.1.1 ¥blinky

RJ45 に付いている緑色と黄色 LED を点滅させます。

#### 4.1.2 ¥uart\_echo



シリアルポートのテストプログラムです。ハイパーターミナルから入力したデータをそのままシリアルポートから返送されます。

Step1、Mini USBケーブルでUSB-COMインタフェースと繋ぐ。デバイスマネージャで配分さ れたシリアルポートを確認します。例えばCOM6。

📴 🚽 ポート (сом と LPT)

Silicon Labs CP210x USB to UART Bridge (COM6)

Step2、ハイパーターミナルを開いて COM6 ポートを選択して、下記の様に設定します:

сом6のプロパティ	
ポートの設定	
ビット/秒(日): 115200 🗸 🗸	
データ ビット( <u>D</u> ): 8	
パリティ(P): なし 💌	
ストップ ビット(S): 1	
フロー制御(E): なし 💌	
既定値に戻す( <u>R</u> )	
OK キャンセル 適用(A	$\mathbf{v}$

Step3、本テストプログラムをボードにダウンロードすると、ハイパーターミナルに下記情報が提示されます:





Step4、任意の文字を入力するとハイパーターミナルに表示されます:

🦓 usb-com - ハイパーターミナル				
ファイル(E)	編集(E)	表示(⊻)	通信( <u>C</u> )	転ì
🗅 🚔 🍘	🌋 🗈 č	9 🖻		
Enter text: hello!_				

#### 4.1.3 ¥softuart\_echo

普通の IO でシリアルポートを模擬します。機能とテスト方法は前例と同じ。

#### 4.1.4 ¥timers



タイマー割込みテストプログラムです。タイマー1は1秒毎にシリアルポートにデータを出 力します。タイマー2は0.5秒毎にシリアルポートにデータを出力します。





#### 4.1.5 ¥watchdog

割り込みモードで実現しています。間隔は1秒。割り込みサービスプログラムでは毎回 LED 制御の IO を NOT 演算します。結果としては RJ45の緑色の LED が点滅します。

#### 4.1.6 ¥sd\_card



本テストプログラムには FatFs ファイルシステムを利用し、LCD 或いはハイパーターミナル で SD カード上のフィアルを操作できます。

Step1、シリアルポートの設定は uart\_echo 例と同じ。ハイパーターミナルから下記の情報 が表示されます:



D 🛩 🔊 🚳 🖆 😭

```
SD Card Example Program
Type 'help' for help.
/>
```

Step2、help を入力するとサポートするコマンドが表示されます:

	/> help
	Available commands
	help : Display list of commands h : alias for help ? : alias for help ls : Display list of files chdir: Change directory cd : alias for chdir pwd : Show current working directory cat : Show contents of a text file
S	itep3、ls コマンドを入力すると SD カードの内容が表示されます:
	/> ls
	A 2011/04/11 10:55 6 README.TXT A 2011/04/14 08:44 8 MSREAD <sup>-1</sup> .TXT A 2011/04/14 08:44 0 MSWRIT <sup>-1</sup> .TXT A 2011/05/26 16:09 1542844 YAMEVOY.way

1>

Step4、LCDを接続している場合、下記の様な内容が表示されます:



4 File(s), 1542858 bytes total 0 Dir(s), 7764384K bytes free



#### 4.1.7 ¥usb\_dev\_serial



USB 模擬シリアルポートプログラムです。USB デバイスをシリアルデバイスとして利用でき ますのでホスト側の USB ドライバが必要ないです。

Step1、テストプログラムをボードにダウンロードします。J10、J11を"DEV"側に設定し て、Mini USBケーブルでUSB-DEVICEインタフェースと繋ぐ。PC側は新しいUSBデバイスが検 出された提示が出ます。提示によって、¥Code¥StellarisWare¥windows\_driversにあるドラ イバを指定します。

Step2、デバイスマネージャで配分されたシリアルポートを確認します。例えばCOM4。

ローダポート (COMとLPT)

Step3、もう一本のMini USBケーブルでUSB-COMインタフェースと繋ぐ。デバイスマネージャで配分されたシリアルポートを確認します。例えばCOM6。

📴 🝠 ポート (COM と LPT)

🍠 Silicon Labs CP210x USB to UART Bridge (COM6)

🚽 🖳 Stellaris USB serial port (COM4)

Step4、二つのハイパーターミナルを開いて、同じ属性で設定します。二つの仮シリアルポートがお互いに通信できます。

🖏 usb-com - ハイパーターミナ	+JL 📃 🗖 🔀
ファイル(E) 編集(E) 表示(Y)	通信( <u>C</u> ) 転送(T) ヘルプ(H)
06 88	🦚 usb-com4 - ハイパーターミナル
send data to com6	ファイル(E) 編集(E) 表示(Y) 通信(C) 転送 🗋 🚅 🍘 🖧 🕮 🎦 🖆
	send data to com4



#### 4.1.8 ¥usb\_dev\_msc



USB カードリーダーとして利用できます。

Step1、J10、J11を"DEV"側に設定して、Mini USBケーブルでUSB-DEVICEインタフェース と繋ぐ。PC側は新しいUSBデバイスが検出された提示が出ます。自動的にドライバーをイン ストールします。

Step2、新しい移動デバイスが追加されています。開くと中の内容が見えます。

#### 4.1.9 ¥usb\_dev\_mouse



HID マウスの USB デバイスプログラムです。

Step1、J10、J11を"DEV"側に設定して、Mini USBケーブルでUSB-DEVICEインタフェース と繋ぐ。PC側は新しいUSBデバイスが検出された提示が出ます。自動的にドライバーをイン ストールします。

Step2、PC上のマウスが四角形で移動します。同時にハイパーターミナル側は下記情報が表示されます:







#### 4.1.10 ¥usb\_dev\_keyboard



HID キーボードの USB デバイスプログラムです。

Step1、J10、J11を"DEV"側に設定して、Mini USBケーブルでUSB-DEVICEインタフェース と繋ぐ。PC側は新しいUSBデバイスが検出された提示が出ます。自動的にドライバーをイン ストールします。

Step2、Wordファイルを新規作成して開きます。ボードのSWキーを押すと、下記内容がファ イルに書かれます: "Stellaris…SeeHowFarYourDollarCanGo"。シリアルポートからは下 記情報が表示します:



#### 4.1.11 ¥usb\_dev\_bulk



USB バルク転送プログラムです。

Step1、¥TOOLS¥SW-USB-win-6852.exeを実行してインストールします。インストール完了し たプログラムはUSB Bulk Example.exeです。

Step2、J10、J11を"DEV"側に設定して、Mini USBケーブルでUSB-DEVICEインタフェース と繋ぐ。PC側は新しいUSBデバイスが検出された提示が出ます。提示によって、 ¥Code¥StellarisWare¥windows\_driversにあるドライバを指定します。



\_ 🗆 ×

Step3、USB Bulk Example.exeを実行します。受信したキャラクタを大文字/小文字変換してホストに返送します。

USB Bulk Example

Stellaris Bulk USB Device Example

Version 6852

This is a partner application to the usb\_dev\_bulk example shipped with StellarisWare software releases for USB-enabled boards. Strings entered here are sent to the board which inverts the case of the characters in the string and returns them to the host.

Enter a string (EXIT to exit):

「ABCD」を入力してEnterを押すと、ボードから「abcd」を返送します。



#### 4.1.12 ¥usb\_dev\_chidcdc



本プログラムを実行するとPC側はUSB HIDデバイスとUSB CDCデバイス(仮シリアルポート) が追加されます。

Step1、J10、J11を"DEV"側に設定して、Mini USBケーブルでUSB-DEVICEインタフェース と繋ぐ。PC側は新しいUSBデバイスが検出された提示が出ます。提示によって、

¥Code¥StellarisWare¥windows\_driversにあるドライバを指定します。

Step2、デバイスマネージャでポートとHIDデバイスの所にデバイスが追加された事を確認。 例えばCOM6。

Step3、ハイパーターミナルを開いて、「?」と Enter を押すと、本プログラムがサポートするコマンドを提示します:

02 23 08 2

「可能への挑戦

?
Available commands
help : Display list of commands
h : alias for help
? : alias for help
mouse : Turn (on|off) mouse movements
led : Set LED mode (on|off|toggle|activity)
> \_

Step4、led off、led on、led toggleなどのコマンドを入力してRJ45の緑色のLEDの変化を 見ます。

Step5、mouse onを入力するとマウスが正方形を書きます。mouse offを入力するとマウス が正常に戻します。

#### 4.1.13 ¥usb\_host\_mouse



本テストプログラムは USB マウスを接続して制御できます。

Step1、上記図の様に USB マウスをボードの USB Host インタフェースに差し込む。USB-COM と PC を繋ぐ。プログラムをボードにダウンロードして実行すると、USB マウスの座標がシ リアルポートから出力します:





D 🗃 🍘 🥈 🖻 🗃 😭 X:02. Y:-16. X:09. X:07. Button Pressed 01 Button Released 01 Y:32. X:-2. Y:15. Y:16. Y:16. X:02. X:01. X:01. Button Pressed 02 Button Released 02 X:-2. Y:15.

Step2、LCDを繋いでいる場合、白い点もUSBマウスの動きによって移動します:



#### 4.1.14 ¥usb\_host\_keyboard

本テストプログラムは USB キーボードを接続して制御できます。

Step1、上記図の様に USB キーボードをボードの USB Host インタフェースに差し込む。 USB-COM と PC を繋ぐ。プログラムをボードにダウンロードして実行すると、USB キーボー ドから押したキーがシリアルポートから出力します:

02 83 08

Host Keyboard Application Keyboard Connected ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopgrstuvwxyz 0123456789



Step2、LCDを繋いでいる場合、LCDにも押したキーが表示されます:



#### 4.1.15 ¥usb\_host\_msc



本テストプログラムは USB メモリを接続して制御できます。※全ての USB メモリが動作す る保証はできない。本試験では Kingston 4G の USB メモリを利用しております。 Step1、上記図の様に USB メモリをボードの USB Host インタフェースに差し込む。USB-COM と PCを繋ぐ。プログラムをボードにダウンロードして実行すると、下記情報がシリアルポ ートから出力します:

/> help Available commands  help : Display list of commands b
Available commands  help : Display list of commands b : alias for belp
help : Display list of commands
? : alias for help ls : Display list of files chdir: Change directory cd : alias for chdir pwd : Show current working directory cat : Show contents of a text file

Step2、1s コマンドを入力すると USB メモリの内容が表示されます:



Step3、LCDを繋いでいる場合、LCDにもUSBメモリの内容が表示されます:



#### 4.1.16 ¥enet\_uip



本テストプログラムは uip ネットプロトコルで http ウェブサーバーの機能を実現します。 Step1、上記図の様に LAN ケーブルでボードの ethernet インタフェースとルータを繋ぐ。 USB-COM と PC を繋ぐ。プログラムをボードにダウンロードして実行すると、ボードが自動 配分された IP アドレスがシリアルポートから出力します:



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

Ethernet with uIP MAC Address Not Programmed! Waiting for Link Link Established Waiting for IP address... IP: 192.168.1.101

Step2、IEブラウザに<u>http://192.168.1.101</u>を入力するとボードのウェブ内容を見れます:

株式会社日昇テクノロジー



Step3、enet\_uip.c を変更する事で固定 IP モードを実現できます。下記の図の様に USE\_STATIC\_IP 機能を有効にします:

Step4、MAC アドレスは User Register 0-1 に保存されています。ROM\_FlashUserGet 関数で 取得できます。デフォルトでは MAC アドレスを書き込んでないので、読み出した MAC アド レスは全部 FF です。

```
📩 enet_uip.c
 596
         // Read the MAC address from the user registers.
 597
         ROM FlashUserGet(&ulUser0, &ulUser1);
<₽>598
         if((ulUser0 == 0xffffffff) || (ulUser1 == 0xffffffff))
 599
 600
 601
 602
             // We should never get here. This is an error if the MAC address has
             // not been programmed into the device. Exit the program.
 603
 604
             UARTprintf("MAC Address Not Programmed!\n");
 605
 606 //
               while(1)
 607 //
               -{
 608 //
               -}
 609
        }
```



#### 4.1.17 ¥enet\_1wip



本テストプログラムは lwip ネットプロトコルで http ウェブサーバーの機能を実現します。 Step1、上記図の様に LAN ケーブルでボードの ethernet インタフェースとルータを繋ぐ。 USB-COM と PC を繋ぐ。プログラムをボードにダウンロードして実行すると、ボードが自動 配分された IP アドレスがシリアルポートから出力します:

Ethernet with lwIP MAC Address Not Programmed! IP: 192.168.1.101 Netmask: 255.255.255.0 Gateway: 192.168.1.1

Step2、IEブラウザに<u>http://192.168.1.101</u>を入力するとボードのウェブ内容を見れます:





😥 Stellaris 🕏 LM359892 Evaluation Kit - Windows Internet Explorer		
🚱 🗸 🔹 http://192.168.1.101	👻 👻 😽 🗙 🕅 Bing	. م
👷 收藏夹 💊 Stellaris® LM3S9B9	2 Evaluation Kit 🔄 🔹 🐨 🐨 👘 👻 页面(P) 💌 安全(S) 🕶	工具(Q) • 🔞 • 🦈
× 查找: 3748	上一个 下一个 📝 选项 🔻	
TEXAS	Stellaris® LM3S9B92 Evalua U <b>MENTS</b>	tion Kit
About TI	Stellaris Family Block Diagram	Î
About the Stellaris Family     Block Diagram	2000         Acron           NOC         Contrast "Adap           NOC         Contrast "Adap           NOC         Contrast "Adap           NOC         VOL WHIT           Statement         Fill           Statement         Contrast "Adap           Statement         Fill           Statement         Contrast Adap           Statement         Fill           Statement         Fill           Statement         Fill           Statement         Fill	-
	Copyright $\otimes$ 2007-2009 Texas Instruments Incorporated. All rig	hts reserved.
完成	😜 Internet   保护模式: 启用 🛛 🚭	• 4 125 mm • •

#### 4.1.18 ¥enet\_ptpd



本テストプログラムは lwip ネットプロトコルを元に IEEE1588 プロトコルで同期イーサネ ット機能を実現します。

Step1、上記図の様に LAN ケーブルでボードの ethernet インタフェースとルータを繋ぐ。 USB-COM と PC を繋ぐ。プログラムをボードにダウンロードして実行すると、ボードが自動 配分された IP アドレスがシリアルポートから出力します。赤線の枠内はボード上の日付と 時間です。

Ethernet with PTPd MAC Address Not Programmed! IP: 192.168.1.101 Netmask: 255.255.255.0 Gateway: 192.168.1.1 Thu Jan 1, 1970 00:00:32 (GMT)

Step2、IEブラウザにhttp://192.168.1.101を入力するとボードのウェブ内容を見れます:



Ø About Texas Instruments - Windows Internet Explorer		- • • ×
😋 🔍 🖉 http://192.168.1.101/index.htm	🗝 📴 🍫 🗙 🖉 Bing	• م
🚖 收藏夫 💋 About Texas Instruments	🏠 🕶 🔝 👻 🖃 👘 👻 页面(2) 🖛 g	?全(5)▼ 工具(0)▼ 🔞▼ 🎽
About Texas Instruments		
Texas Instruments (TI) is a global <u>analog</u> and digital <u>semiconductor IC design</u> and manufacturing company. In addition to analog technologies, digital signal processing (DSP) and microcontroller (MCU) semiconductors. TI designs and manufactures <u>semiconductor solutions</u> for analog and digital embedded and application processing. In the microcontroller space, TI offers the broadest range of embedded control products, from ultra-low-power <u>MSP430</u> <sup>™</sup> MCUs and high-performance <u>TMS320C2000</u> <sup>™</sup> real-time controllers, to the 32 bit general-purpose ARM®-based MCUs of the <u>Stellaris</u> <sup>®</sup> line.		
Read more about us on the web at <u>www.ti.com</u> .		
For more details about the PTPd implementation, click here.		
		-

Step3、 "For more details about the PTPd implementation, click here"の "here"を クリックしてイーサネット機能テストに入ります。1秒毎に時間を更新します。

Ø About PTP (IEEE-1588) on Stellaris - Windows Internet Explorer		_ <b>O X</b>
() The second se	🝷 🗟 😽 🗙 🌌 Bing	. م
🚖 收藏夹 🔘 About PTP (IEEE-1588) on Stellaris	🏠 🕶 🖾 👻 📑 🖷 🕶 页面(2) 🕶 安全(5) 🕶	⊥具(0) ▼ 🕜 ▼
About PTPd (IEEE-1	588) on Stellaris	
Texas Instruments has implemented the Open Source PTPd soft implementation uses the IwIP TCP/IP stack, and does not require	ware on its Stellaris family of microcontrollers a an RTOS.	. This
Details on the Open Source PTP Protocol Software can be found at <u>ptpd.sourceforge.net</u> .		
Details on the Open Source IwIP TCP/IP stack can be found at <u>www.sics.se/~adam/lwip/</u> .		
The current IEEE 1588 Time (GMT) is: Thu Jan 1, 1970 00:04:02		
To return to the main page, click here.		
带符 http://192.168.1.101/ptpclock.ssi	Verifier Internet   保护模式: 启用 Verifier	• ₹125% •

#### 4.1.19 ¥qs-blox

本試験はテトリスゲームを実現します。LCDを接続する必要です。 Step1、「Start」を押してゲーム開始します。





Step2、画面下にある5つのボタンで各種操作できます。





#### 4.1.20 ¥idm-checkout



出荷時のデモソースです。

試験前の準備:

- 1、Toolフォルダにあるramfs\_data.binをPCのCドライブのルートフォルダにコピーしま す。
- 2、Tool フォルダにある\*. wav ファイル及び SD Web Sit フォルダ内のファイルを全て SD カードのルートフォルダにコピーします。

Step1、LANケーブル、Mini USB(USB-COM)、USBマウス、SDカード、イヤホンをボードの対応インタフェースに接続。プログラムをボードにダウンロードして電源を入れると、四つのテスト項目が表示されます: "IO Examples"、"Graphics Demo"、"Audio Player"、 "Image Viewer"。

Step2、"IO Examples"をタッチします:





 1、"Ethernet"項目ではボードに配分されたIPアドレス「192.168.1.102」が表示します。
 2、"Board I/O"項目の"Touch"は指先がLCDにタッチしている位置の座標を表示します。
 3、"USB"項目の"Mouse"はマウス移動の座標を表示します。1、2、3ボタンでマウスの 三つのボタンの状態を示します。

4、"Status"項目欄はファイルシステムを表示します。SDカード、SDRAM上のファイル及 びUSBマウスのカレント状態を表示します。

5、"SDRAM fs" について、画像ファイルはSPIインタフェースのFlashに保存されていて、 電源入れるとFlashからSDRAMにコピーして解凍処理を行います。SPI Flashに何のファイル もない場合は"SDRAM fs absent"と表示します。

6、画像ファイル"ramfs\_data.bin"はTFTPツールでボードにアップロードできます。WinXP でコマンドウィンドウでCドライブのルートフォルダに入って、"TFTP -i 192.168.1.102 PUT ramfs\_data.bin EEPROM"コマンドでファイルをボードのSPI Flashにアップロードし ます(およそ20秒かかります。Win7ではサポートしません)。



7、再度ボードに電源を入れて"Status"項目欄の"SDRAM fs absent"が"SDRAM fs present"に変更します。

8、IE ブラウザで http://192.168.1.102 を入力 (Ethernet 項目に表示している IP アドレス) してボードのウェブサーバーに入ります:





9、"SDCard Web Site"をクリックして SD カードに保存しているウェブファイルが見えま

す: 🝘 Stellaris® LM3S9B92 Evaluation Kit - Windows Internet Explorer \_ **— X** 🔹 🖹 🍫 🗙 🛃 Bing 🔄 🕞 💌 http://192.168.1.102/sdcard/index.htm ρ. 👷 收藏夹 🛛 ♦ Stellaris® LM3S9B92 Evaluation Kit 🟠 🔹 🔝 👻 🛲 🔹 页面(P) 🗸 安全(S) 🖛 工具(Q) 🕶 🔞 🕶 Smart LM3S9B92 Website in SD **IEXAS** Card INSTRUMENTS About TI Texas Instruments (TI) is a global analog and digital semiconductor IC design and manufacturing company. In addition to analog technologies, digital signal processing (DSP) and microcontroller (MCU) semiconductors, TI designs and manufactures <u>semiconductor solutions</u> for <u>About the Stellaris</u>
 <u>Family</u> Block Diagram analog and digital embedded and application processing. In the microcontroller space, TI offers the broadest range of embedded control products, from ultra-low-power MSP430<sup>™</sup> MCUs and high-performance TMS320C real-time controllers, to the 32 bit general-purpose ARM®-based MCUs of the <u>Stellaris</u>® line. Copyright © 2007-2009 Texas Instruments Incorporated. All rights reserved. 😌 Internet | 保护模式: 启用 🛛 🖓 ▾ 🔍 125% ▾

10、8の画面から"Photo Gallery"をクリックすると、TFTP からアップロードした画像フ ァイル(ramfs\_data.bin)が見えます:







11、右下にあるボタン ▼ ● 125% ▼ 」で閲覧モードを変更できます。

12、"IO Examples"項目テスト完了後、"Home"をタッチしてメイン画面に戻ります。 Step3、"Graphics Demo"アイコンをタッチして、TI グラフィック機能をテストします。



















Step4、"Audio Player"アイコンをタッチしてオーディオ機能をテストします。





プレイする曲 (.wav ファイル) をタッチして、"Play"をタッチします。右側のヴォリュ ームバーをタッチして音量を調整できます。



"Home"をタッチしてメイン画面に戻します。

Step5、"Image Viewer"をタッチして TFTP から SPI Flash にアップロードした画像ファ イル(ramfs\_data.bin)を見えます。









#### 4.1.21 ¥hello\_lcd

本テストは簡単な LCD テストプログラムです。LCD 上"Hello World!"を表示します。



#### 4.1.22 ¥hello\_widget

本テストは簡単な LCD 及びタッチパネルテストプログラムです。電源を入れると LCD に"Show Welcome"の表示があります。指先でタッチすると下に"Hello World!"が表示 される同時に"Show Welcom"が"Hide Welcome"に変更します。



#### 4.1.23 ¥calibrate

本テストはタッチパネルのキャリブレーションプログラムです。LCDの提示通り白い点をタ ッチして、補正後のデータが表示されます。このデータで元のデータ(M0-M6)に切り替え る事でタッチ効果がより精確になります。



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

	touch.c	
<₽>118	#ifdef LANDSCAPE	
119	#define M0	85888
120	#define M1	-384
121	#define M2	-18005888
122	#define M3	264
123	#define M4	-76008
124	#define M5	66659640
125	#define M6	193184





calibrat	e
Touch the box	
MAC: ff-ff-ff-ff-ff-ff-ff	IP: 169.254.254.255
calibrat	e
calibrat CALIBRATION DATA:	ie
$\frac{\text{calibrat}}{\text{Calibration data:}}$ $M0 = 72960$	e
calibrat CALIBRATION DATA: M0 = 72960 M1 = -512	e
calibrat CALIBRATION DATA: M0 = 72960 M1 = -512 M2 = -14885696	ie
calibrat CALIBRATION DATA: M0 = 72960 M1 = -512 M2 = -14885696 M3 = -72	ie
$\begin{array}{r} \text{calibrat} \\ \text{CALIBRATION DATA:} \\ \text{M0} = 72960 \\ \text{M1} = -512 \\ \text{M2} = -14885696 \\ \text{M3} = -72 \\ \text{M4} = -78024 \\ \end{array}$	ie
$\begin{array}{r} \text{calibrat} \\ \text{CALIBRATION DATA:} \\ \text{M0} = 72960 \\ \text{M1} = -512 \\ \text{M2} = -14885696 \\ \text{M3} = -72 \\ \text{M4} = -78024 \\ \text{M5} = 64673952 \\ \text{M6} = -168462 \end{array}$	ie
$\begin{array}{r} \text{calibrat} \\ \text{CALIBRATION DATA:} \\ \text{M0} = 72960 \\ \text{M1} = -512 \\ \text{M2} = -14885696 \\ \text{M3} = -72 \\ \text{M4} = -78024 \\ \text{M5} = 64670050 \end{array}$	ie
$\begin{array}{r} \text{calibrat} \\ \text{CALIBRATION DATA:} \\ \text{M0} = 72960 \\ \text{M1} = -512 \\ \text{M2} = -14885696 \\ \text{M3} = -72 \\ \text{M4} = -78024 \\ \text{M5} = 64673952 \\ \text{M6} = 168462 \\ \end{array}$	ie

#### 4.1.24 ¥scribble

本テストは描画プログラムです。タッチペンで LCD に画像を描けます。





#### 4.1.25 ¥lang\_demo

本テストはグラフィックライブラリのストリングの使用を表示します。



#### 4.1.26 ¥showjpeg

本テストは JPEG 画像を表示します。JPEG 画像ファイルは SDRAM にコピーして解凍処理を行います。画像の表示範囲が LCD より大きい場合、スクロールで移動して閲覧できます。





#### 4.1.27 ¥i2s\_demo

本テストはボードのオーディオ機能を実現します。SD カードに保存している.wav ファイル をプレーします。右側のヴォリュームバーを移動して音量を調整できます。

i2s demo		
README.TXT MSREAD"1.TXT MSWRIT"1.TXT YAMEVOY.wav	YAMEVOY.wav 11025 Hz 16 bit Mono 1:09/1:09	
	Play	
MAC: ff-ff-ff-ff-ff-ff IP: 169.254.254.255		

#### 4.2 Code¥ek-1m3s9b92-C3¥RL¥TCPnet¥

※MCUがC1のバージョンの場合はCode¥ek-1m3s9b92¥RL¥TCPnet¥をご参照ください。





この図はC5 バージョンです。

このフォルダにあるサンプルソースは Keil の lib で提供しているネットプロトコルを利用 しています。一部のプログラムでは RTX の 0S モードを選択できます。

LM3S9B92	🖌 👗 🛸
LM3S9B92	HTTP_demo.
LM3S9B92 Debug	
LM3S9B92 RTX	/* Init:

4.2.1 ¥Http\_demo

本テストは http サーバープログラムです。ウェブからボード上の LED 制御、キー状態の取得、ネット接続状態の取得などが出来ます。

Step1、デフォルトでは DHCP 機能でルータから自動で IP アドレスを取得しますが、取得できない場合は Net\_Config.c に設定している固定 IP アドレスを利用します。

HTTP_demo.c Ret_Config.c			
Expand All Collapse All	Help		
Option	Value		
Ethernet Network Interface			
MAC Address			
□ IP Address			
Address byte 1	192		
Address byte 2	168		
Address byte 3	1		
Address byte 4	103		
🖻 Subnet mask			
Mask byte 1	255		
Text Editor Configuration Wizard			

Step2、ブラウザで<u>http://stellaris</u>を入力するとボードのウェブに登録できます。ユーザ 一名はadmin、パスワードはないです。





Step3、"Network"をクリックしてボードのネット関連設定を確認及び修正ができます:

Ø Network Settings - Windows Internet Explorer	×						
💽 🗸 🔊 http://stellaris/network.cgi	👻 🔁 😽 🗙 🌌 Bing 🖉						
🚖 收藏夹 🏼 🏈 Network Settings	🏠 🕶 🔜 👻 🥶 🖛 🔻 页面(P) 🕶 安全(S) 🕶 工具(Q) 🕶 🚱 🕶						
Embedded Deve	opment Tools						
<b>Network Settings</b> Here you can change the system <b>Network Settings</b> . After you have changed the IP address, you need to change also the host IP address in you Internet browser to re-connect to target. Make changes with <b>care</b> or you may permanently lose a connection until next hardware reset. This Form uses a <b>GET</b> method to send data to a Web server.							
Item	Setting						
LAN IP Address	192.168.1.101						
LAN Net Mask	255, 255, 255, 0						
Default Gateway	192.168.1.1						
Primary DNS Server	192.168.1.1						
Secondary DNS Server	0. 0. 0. 0						
	Change Undo .						
خور مدر							

Step4、"System"をクリックして登録時のユーザー名とパスワードを修正できます:





Step5、"LED"をクリックしてLED制御画面に入ります。



Running Lights は点滅モード。Brower は一個づつ LED の点灯/消灯を制御できます。 Brower モードを選択して、2 と 3 の所にチックを入れる或いは消す事で RJ45 の緑色と黄色の LED を点灯/消灯させます。

ED Control - Windows Internet Explorer							
🚱 🔍 🕖 http://stellaris/leds.cgi	👻 😒 😽 🗙 🌌 Bing	ب م					
🚖 收藏夹 👩 LED Control	🏠 🔻 🔝 👻 🛁 🖶 👻 页面(12) 🖛 安全(5) 👻	<u>⊥ạ(o)</u> • (?) • "					
接照 Internet Explorer 中的默认设置现在已关闭 Intranet	设置。单击此处查看更多选项	×					
Embedded Develo	pment Tools						
Control LEDs on the board This page shows you how to use the following http form input objects: checkbox, select and button. It uses also a simple Java Script function to check/uncheck all checkboxes and submit the data. This Form uses a POST method to send data to a Web server.							
Item	Setting						
<ul> <li>LED control:</li> <li>LED diode ports [70]:</li> <li>All LED diodes On or OFF</li> </ul>	Browser • 7 6 5 4 V 3 V 2 1 0 ON OFF						
A							

Step6、"Button"をクリックしてボードのキーの状態を制御できます。SWキーは0位と対



応していて、押下時にチックが表示されます。

	8 Button inputs - Windows Internet Explorer	
☆ 松蔭夫 愛 Button inputs 按照 Internet Explorer 中的軟从设置現在已关闭 Intranet 设置,单击此处画着更多透现 Embedded Development Tools Embedded Development Tools Eutrons on the board This page allows you to monitor on board buttons state. Periodic screen update is based on xml technology. This results in smooth flicker-free screen update. Press a button on an evaluation board and observe the change on the screen. Item Status	💽 🗸 🖉 http://stellaris/buttons.cgi	🝷 🗟 🔩 🗙 🌌 Bing 🛛 👂 •
を接肌terret Explorer 中的軟以设置現在已关闭 Intranet 设置,单击此处画着世多透现…   Embedded Development Tools    Embedded Development Tools   Embedded Development Tools   Embedded Development Tools    Embedded Development Tools   Embedded Development Tools    Embedded Development Tools    Embedded Development Tools   Embedded Development Tools    Embedded Development Tools    Embedded Development Tools    Embedded Development Tools  <	🚖 收藏夹 🏼 🍘 Button inputs	🏠 🔻 📉 👻 🖃 👘 👻 页面图 👻 安全(S) 🐂 工具(Q) 👻 🔞 💌
Embedded Development Tools       Image: Company       Image: Company         Buttons on the board       Image: Company       Image: Company <t< th=""><th>● 按照 Internet Explorer 中的默认设置现在已关闭 Intranet</th><th>置。单击此处查看更多选项 &gt;</th></t<>	● 按照 Internet Explorer 中的默认设置现在已关闭 Intranet	置。单击此处查看更多选项 >
Buttons on the board         This page allows you to monitor on board buttons state. Periodic screen update is based on xml technology. This results in smooth flicker-free screen update.         Press a button on an evaluation board and observe the change on the screen.         Item       Status	Embedded Develo	oment Tools
Press a button on an evaluation board and observe the change on the screen.           Item         Status	Bu This page allows you to monitor on board	ttons on the board
Item Status	Press a button on an evaluation board and	observe the change on the screen.
	Item	Status
Buttons [70]:         7         6         5         4         3         2         1         0	Buttons [70]:	7 6 5 4 3 2 1 0
Refresh Periodic:		Refresh Periodic:
● fnck ● Internet   保护模式: 启用 ● 125% ▼		

#### 4.2.2 ¥LEDSwitch

本プログラムはUDP 及び TCP ポートで¥Code¥PC-SoftWare¥LEDSwitch¥Release フォルダにあ る LEDSwitch.exe から発送したコマンドを検出して処理を行います。LEDSwitch.exe からボ ードの LED を点灯或いは消灯制御できます。¥Code¥PC-SoftWare¥LEDSwitch フォルダには VC++ソースコードです。

Step1、手動でボードのIPアドレスとGatewayを変更します。ボードとPCを同じIPアドレス セグに設定する必要です。Net\_Config.cファイルを編集します。

LEDSwitch.c I Net_Config.c	
Expand All Collapse All	Hel
Option	Valu
Ethernet Network Interface	$\checkmark$
H MAC Address	
Address byte 1	192
Address byte 2	168
Address byte 3	1
Address byte 4	103
⊞ Subnet mask	
🖻 Default Gateway	
Address byte 1	192
- Address byte 2	168
- Address byte 3	1
Address byte 4	1

Step2、LANケーブルでルータとボードを繋ぐ。プログラムをボードにダウンロードして、 LEDSwitch.exeを実行します。"TCP/UDP Port"を1001に設定し、"Board IP"を



192.168.1.103に設定します。 対応LEDが点灯、消灯します。

192 . 168 . 1 . 103

TCP/UDP Port

1001

LEDSwitch

Protocol

○ TCP

O UDP

Board IP

0x0C

"Outputs[LEDs]"の2-3位の所にチックを入れるとボードの

X X LEDSwitch Protocol TCP/UDP Port TCP 1001 O UDP Board IP 192 . 168 . 1 . 103 Outputs [LEDs] 76543210 76543210 

#### 4.2.3 ¥DNS\_demo

Outputs [LEDs]

本プログラムではサーバー名でサーバーの IP アドレスを解析します。5 秒毎に以下のサー バーを解析します:

www.google.com

www.keil.com

www.microsoft.com

www.yahoo.com

dns\_cbfunc関数で解析結果を判断して、解析情報をシリアルポートから出力します。

0 🛩 🍘 🔏 🕒 🗃

#### DNS Resolver Demo Example

```
Resolving host: www.google.com
IP Address : 74.125.71.104
Resolving host: www.keil.com
IP Address : 217.140.108.95
Resolving host: www.microsoft.com
IP Address
                 : 207.46.19.190
Resolving host: www.yahoo.com
IP Address : 98.137.149.56
Resolving host: www.notexisting.site
IP Address : 220.250.64.24
Resolving host: www.google.com
IP Address : 74.125.71.104
Resolving host: www.keil.com
IP Address : 217.140.108.95
```



#### 4.2.4 ¥Telnet\_demo

本テストはTelnet関連アプリのプログラムです。Telnetソフトでボードを制御できます。 Step1、LANケーブルを繋いで電源をいれる。プログラムをダウンロードします。 Step2、PC側でコマンドウィンドウを開いて(CMD.exe)、telnet stellarisを実行します。 ユーザー名はadmin、パスワード無し。







Step3、?を入力してEnterキーを押す。提供している機能を確認します。

Telnet stellaris	
× Keil Embedded Telnet Ser∪er Demo ∪3.00 × Server Demo ∪3.00	***
Stellaris> ? Available Commands:	
<pre>led xx - write hexual xx to LED port led - enable running lights adin x - read AD converter intput x meas n - display n measurements tcpstat - display a tcp status rinfo - display remote machine info passw [new] - change system password passwd - display current password help, ? - display this help bye - disconnect</pre>	
<esc>,&lt;^C&gt; - disconnect <bs> - delete Character left <up><down> - recall Command History</down></up></bs></esc>	
<b>Stellaris</b> >	-



Step4、led 0x0Cを入力してEnterを押すと二つのLED共消灯させます。Led 0x08を入力して Enterを押すと一つのLEDが点灯します。

Telnet stellaris	
<esc>,&lt;^C&gt; - disconnect <bs> - delete Character left <up><down> - recall Command History</down></up></bs></esc>	•
Stellaris> ?	=
Available Commands:	
<pre>led xx - write hexval xx to LED port led - enable running lights adin x - read AD converter intput x meas n - display n measurements tcpstat - display a tcp status rinfo - display remote machine info passw [new] - change system password passwd - display current password help, ? - display this help bye - disconnect</pre>	
<esc>,&lt;^C&gt; - disconnect <bs> - delete Character left</bs></esc>	
<ul> <li><ul> <li><ul> <li>Pecall Command History</li> </ul> </li> <li><ul> <li><ul></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul></li></ul>	
Stellaris> led 0x0C> Running Lights OFF Stellaris> led 0x00 Stellaris> led 0x08 Stellaris>	
	Ŧ



#### 付録: OpenLink で実行ファイルの書き込む

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)。

#### 1 ドライバのインストール

SEGGER 社様のソフトウエ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してください。

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。

#### 2 J-FLASH ARMで実行ファイルを書き込む

J-FLASH ARM を実行する。

SEGGER	J-Flash ARM V4.	34d - [C:¥Program Files¥SEGGER¥JLinkARM_V434d¥Default.jflash *]	
<u>E</u> ile <u>E</u> dit <u>V</u> ie	ew <u>T</u> arget <u>O</u> ptior	ns <u>W</u> indow <u>H</u> elp	
Project -	- D 🔲 🗖 🔀		
Name	Value		
Connection	USB [Device 0]		
Target interf	JTAG		
Init JTAG sp	5 kHz		
JIAG speed	Auto recognition		
I AP number	<not used=""></not>		
IRPre	<not used=""></not>		
CPU	Auto		
Endian	Little		
Check core ld	No		
Core Id	0x0		
Use target R	No		
RAM address	0x0		
RAM size	4 KB		
Flash memo	Auto detection		
Base addre	0x0		
Organization	16 bits x 1 chip		
LOG			
– J–Flash ARI	M V4.34d (J <del>-</del> Flash c	ompiled Sep 13 2011 17:29:11)	^
– JLinkARM.dl	ll V4.34d (DLL comp	iled Sep 13 2011 17:28:37)	
Reading flash o	device list LC:¥Progra	am Files¥SEGGER¥JLinkARM_V434d¥ETC¥JFlash¥Flash.csv]	
- List of flash	i uevices read succe	ssiuny (arz. Devices)	
List of MCU de	vices read successf	ully (1015 Devices)	
書き込む前	前に必要な設定	定(Options->project settings):	

nen SEGGER J-Flash ARM V4.34d - [C:¥Program Files¥SEC									
<u>F</u> ile <u>E</u> dit <u>V</u> i	ew <u>T</u> arget	<u>O</u> ptions	<u>W</u> indow	<u>H</u> elp					
Project -	- D	<u>P</u> roje <u>G</u> loba	ct settings al settings	. Alt-F7					
Name	Value				_				
Connection	USB [Device	: 0]							



#### 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。

	J-Flash ARM	V4.3	84d - [C:
<u>File E</u> dit <u>V</u> ie	ew <u>T</u> arget <u>O</u> p	otions	s <u>W</u> indow
Project -	- D	X	
Name	Value	^	
Connection	USB [Device 0]		
Target interf	JTAG		
Init ITAG en	5 kHz		
JTAG speed	Auto recognition		
TAP number	<not used=""></not>		
IRPre	<not used=""></not>		
MCU	Luminary LM3S9		
Clock speed	Auto recognition		
Endian	Little		
Check core Id	Yes		
Core Id	0x3BA00477		
Use target R	Yes		
RAM address	UX2000000	-	
RAM size	96 KB		
Flash memo	LM3S9B92 intern		
Manufacturer	Luminary		
Size	256 KB	~	
<	>		

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。



Open data file				? 🗙
ファイルの場所①: 最近使ったファイル デスクトップ マイドキュメント マイ コンピュータ マイ ネットワーク	rvmdk	▼ ← €	*	
	ファイル名( <u>N</u> ): ファイルの種類( <u>T</u> ):	uart_echo.hex Data files (*.mot, *.srec, *.s19, *.s, *.hex, *	▼ K.bin.▼	開(( <u>0</u> ) キャンセル

「開く(0)」をクリックする。

RASH SEGGER	J-Flash ARM	V4.:	84d = [C:¥Pr	ogra	m F	iles	(SEC	<b>GE</b>	R¥J	Link	ARN	M_V434d¥Default.jflash <b>*</b> ]	
<u>F</u> ile <u>E</u> dit ⊻i	ew <u>T</u> arget <u>O</u> p	tion	s <u>W</u> indow <u>H</u>	elp									
Real Project	- D	X	FIASH E:¥Im3s9	b923	(Co	de¥{	Stell	a ris I	Vare	e¥bo	ard	s¥Smart-9B92-C1¥uart_echo	💶 🗖 🗙
Name	Value	^	Addres 0x1000	)	×1	<u>×2</u>	× <u>4</u>						
Connection Target interf	USB [Device 0]		Address	0	1	2	3	4	5	6	7	ASCII	<b>_</b>
raigetinen	onna		1000	00	01	00	20	1D	11	00	00		
Init JTAG sp	5 kHz		1008	21	11	00	00	23	11	00	00	!#	
TAP number	<not used=""></not>		1010	25	11	00	00	25	11	00	00	%%	
IRPre	<not used=""></not>	-	1018	25	11	00	00	00	00	00	00	%	
мси	Luminary LM3S9	=	1020	00	00	00	00	00	00	00	00		
Clock speed	Auto recognition		1028	00	00	00	00	25	11	00	00	%	
Endian Check core Id	Little Vec		1030	25	11	00	00	00	00	00	00	%	
Core Id	0x3BA00477		1038	25	11	00	00	25	11	00	00	%%	
Use target R	Yes		1040	25	11	00	00	25	11	00	00	%%	
RAM size	96 KB		1048	25	11	00	00	25	11	00	00	%%	
			1050	25	11	00	00	Δ9	11	00	00	%	
Hash memo Manufacturer	LM3S9B92 intern		1058	25	11	00	00	25	11	00	00	%%	
Size	256 KB	~	1060	25	11	00	00	25	11	00	00	%%	
<		зđ	1068	25	11	ΠΠ	ΠN	25	11	ΠΠ	ΠΠ	% %	<b>_</b>
INSH LOG													
Opening data f - Data file op	ile [E:¥lm3s9b92] ened successful	¥Coo Iv (7)	le¥StellarisWan 36 bvtes 1 ran	e¥bo: ge.C	ards) RC :	¥Sma =0x1	rt-9E 3772	392-0 2D8E	01¥i )	uart_e	echo <sup>3</sup>	¥rvmdk¥uart_echo.hex]	^
				_ / _									*
<													> .:
Ready											1	Not connecte	//

Target->Auto あるいは F7 で書き込み開始する。

#### 書き込み完了:

J-Flash	a ARM V4.34d 🛛 🔀
į)	Target erased, programmed and verified successfully – Completed after 8.894 sec
	OK