

MAX II/Cyclone II EPM240/EP2C5/EP2C8 ボード

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

2009/11/17



copyright@2009



第一章 CPLD/FPGAボードの概要	3
1.1 MAX II/EPM240 ボード	3
1.2 Cyclone II/ EP2C5-EP2C8 ボード	3
1.3 CPLD/FPGA の実験用I/Fボード	3
第二章 開発ツールをインストール10)
2.1 Quartus II Web Editionをインストールする10)
2.2 Nios Ⅱ エンベデッド・デザイン・スイートをインストールする17	7
第三章 MAX II/Cyclone IIの初体験	3
3.1 Quartus II評価版にソースを読み込む	3
3.2 USB-Blaster をインストールする25	5
3.3 書き込むソフトウェアを起動する	3
3.4 FPGA のコンフィギュレーションデバイスに書き込む30)
3.5 NIOS II プロセッサーの初体験32	2
第四章 CPLD/FPGAの開発入門40)
4.1 プロジェクトを作成する40)
4.2 エディタで回路図を描く47	7
4.2.1 トップ・エンティティを作成する47	7
4.2.2 作画手順)
4.2 書き込み前の二つの作業	2
4.2.1 回路図をコンパイルする52	2
4.2.2 回路図の入出力とCPLD/FPGAの端子を関連づける	3
第五章 NIOS IIシステム・モジュールの設計	1
第六章 NIOS IIのプログラムの設計77	7

※ 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。



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

第一章 CPLD/FPGAボードの概要

1.1 MAX II/EPM240 ボード

MAX® II CPLD ファミリは、CPLD ファミリの中で I/O ピンあたり最も低いコストと最少の消費電 カを実現し、画期的な新 CPLD アーキテクチャをベースにしている不揮発性、インスタント・オン プログラマブル・ロジック・ファミリです。この新しいアーキテクチャは、システムの消費電 カ、スペ ース、そしてコストの低減を可能にします。

MAX II デバイス・ファミリの概要				
特長	EPM240/G/Z	EPM570/G/Z	EPM1270/G	EPM2210/G
ロジック・エレメント(LE)数	240	570	1,270	2,210
標準等価マクロセル数	192	440	980	1,700
最大ユーザ I/O ピン数	80	160	212	272
ユーザ・フラッシュ・メモリビ ット	8,192	8,192	8,192	8,192
デバイス配給状況	出荷中(1)	出荷中(1)	出荷中	出荷中
tpd1(ns)(コーナ対コーナ性能)	4.5	5.5	6	6.5
tpd2(ns)(最高速性能)	3.6	3.6	3.6	3.6



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

MAXII/EPM240 ボードの概要:





- EPM240 搭載
- 50MHz 水晶発振器搭載
- 5V 電源又は USB ポートで給電、電源スイッチと電源指示 LED 付き
- 3.3V レギューレータ搭載
- JTAG コネクタ
- 回路図を提供しております
- サンプルのソースコードを提供しております。



1.2 Cyclone II/ EP2C5-EP2C8 ボード



Cyclone II デバイスは、90-nm テクノロジの優位性(小型ダイ・サイズ、高集積度、および低コスト)と、低コスト FPGA における最速性能を提供します。すべての Cyclone II デバイスは、TSMCの 90-nm プロセス技術と low-k 低誘電材を使用して 300-mm ウェハ上に製造されています。



Cyclone II デバイスは、一般的なデジタル信号処理 (DSP) 機能を実装できる、最大 150 個の 18 ビット x 18 ビット・マルチプライヤを備えています。エンベデッド・マルチプライヤは、ロジック・ エレメント (LE) ベースのマルチプライヤと比較してより高い性能とロジック効率を提供します。



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



	デバイス					
特徴	EP2C5	EP2C8	EP2C20	EP2C35	EP2C50	EP2C70
ロジック・エレメント数	4,608	8,256	18,752	33,216	50,528	68,416
M4K RAM ブロック数	26	36	52	105	129	250
RAM 総ビット数	119,808	165,888	239,616	483,840	594,432	1,152,000
エンベデッド乗計算器数	13	18	26	35	86	150
PLL 数	2	2	4	4	4	4

Cyclone II/EP2C5-EP2C8 ボードの概要:





- EP2C5Q208 又は EP2C8Q208 搭載
- NIOS II ソフトプロセッサ搭載
- コンフィギュレーションデバイス EPCS4(4Mbit)搭載
- SDRAM(64Mbit)搭載
- 50MHz 水晶発振器搭載
- 5V 電源又は USB ポートで給電、電源スイッチと電源指示 LED 付き
- 3.3V/1.2V レギューレータ搭載
- JTAG と ASP コネクタ
- ユーザ LED とユーザボタン
- 回路図を提供しております
- サンプルのソースコードを提供しております。

1.3 CPLD/FPGAの実験用I/Fボード



- LED x 8
- 8ビット DIP スイッチ



- ボタンx8
- PS2x2、キーボード又はマウスを接続可
- VGA x 1
- RS232 x 1
- 1602 液晶 I/F x 1
- 7セグメント LED x 8
- Beep x 1
- 回路図を提供しております

実験用 I/F ボードが EPM240 又は EP2C5Q208 又は EP2C8Q208 ボードと一緒に動く様子。



1/2008



第二章 開発ツールをインストール

CPLD/FPGAの開発には、ALTERAから Quartus II Web Edition という無償版のツール が公開されているのでこちらを利用します。Quartus II には別に製品版があり、Web Edition は使用できるデバイスなどに制限がありますが、MAX II と Cyclone II に関しては、 どのデバイスも使用できるのでまったく問題ありません。Quartus II Web Edition は、総 合開発環境になっており、このソフトウェアだけで、ソース・エディタや I/O ピンのアサイ ンメント、論理合成、デバイスの書き込み用のプログラムなど、PLD/FPGAの開発に必要 な機能がすべて含まれています。また、Nios II エンベデッド・デザイン・スイートは Nios プロセッサー用の開発ツールです。

Quartus II Web Edition と Nios II エンベデッド・デザイン・スイートのダウンロードは、 次の URL から行うことができます。

http://www.altera.co.jp/support/software/download/nios2/dnl-nios2.jsp

なお、ダウンロードする際は、最初に ALTERA のページにサイン・インを行い、ユーザ情報を登録する必要があります。現時点最新版は v8.1 です。インストールした後、ライセンス・ファイルが不要です。



2.1 Quartus II Web Editionをインストールする



「Next」ボタンを押すと、英文のライセンスが出てきます。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。





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

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。

Quartus II 8.1 Web Editio	n Setup	×
Choose Destination Loc	ation	
	Setup will install Quartus II 8.1 Web Edition in the following folder. To install to this folder, click Next. To install to a different folder, click Browse and select another folder. Destination Folder————————————————————————————————————	
	C:¥altera¥81 B <u>r</u> owse	
facet IIC (c)	Cancel (Next)	
Installonelo		
Quartus II 8.1 Web Edition	n Setup	×
Quartus II 8.1 Web Edition Select Program Folder	folder	
Quartus II 8.1 Web Editio Select Program Folder Please select a program	folder.	X
Quartus II 8.1 Web Editio Select Program Folder Please select a program	folder. Setup will add program icons to the Program Folder listed below. You may type a new folder name, or select one from the existing folders list. Click Next to continue. Program Folder: Altera Existing Folders: Adobe Apache HTTP Server 2.2 Cadlus 1/7 +/y DAEMON Tools Lite Hewlett-Packard Microsoft eMbedded Visual C++ 4.0 Microsoft SQL Server 2005 Microsoft Visual Studio 2008 Microsoft Windows CE 5.0 Microsoft Windows SDK v6.0A Mozilla Firefox	



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。



「**Complete**」を選択してください。





間違いがないかどうか確認し、問題がなければ「Next」を押します。



インストール中の画面です。



インストール完了すると、ショートカットをデスクトップに作るかどうか聞かれます。どちらでも選択できます。





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。

インストールされた Quartus II 評価版をさっそく起動してみます。一番最初に起動したと きだけ、次のようなダイアログが現れ、「Run the Quartus II software」を選択してくださ い。「OK」ボタンを押します。

Quartus II 8.1 Web Edition
Thank you for installing the Quartus II Web Edition software - the #1 in performance and productivity. To upgrade to the full featured Subscription Edition please visit www.altera.com
Select one of the following options Run the Quartus II software Add an IP license file (for users who have purchased IP)
OK Cancel



Quartus II の画面出てきます。

Cotting Storted Wit	
Getting Started wit	ouartus II Software
	OUARTUS II
	A REAL PROPERTY AND INCOME.
Start Designing	Start Learning
Designing with Quartus II software requires a project	The audio/video interactive tutorial teaches you the basic features of Quartus II software
Create a New Project (New Project Wizard)	Open Interactive Tutorial
Open Existing Project	
Open Recent Project: ButtonCountAllVHDL	
LED1 ButtonCountV(HDI	
ButtonCount	
Web links: Literature Training	Online Demos Support
Don't show this screen again	
Real	
Quartus II File Edit View Project Assignments Processing Tools W	/indow Help
	_ X ∠ Ø Ø Ø ⊅ K № 10 Ŭ ≿ ⊕ ⊕ ❶ ●
Project Navigator A X	
	Quartus II Software Timing esigner
	Download Software Get a Nios* II Subscription Edition Free trial>> Processor Free!
	EMA Partitionation FButy Now
Ariterarchy Files Provide The Sign Units	
Tasks Flow: Compilation	OUARTUS II
Iask	Version 8.1
	View New Austria II
EDA Natist Writer Program Device (Open Programmer)	Information
< >	Documentation
× Type Message	
System (Processing) Extra Info) Info) Warning) Critic	al Warning) Error) Suppressed) Flag /



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

2.2 Nios II エンベデッド・デザイン・スイートをインストールする

AB MIDS T FUIDE	dded Design Suite 8.1
	 Press Install button to start extraction. Use Browse button to select the destination folder from the folders tree. It can be also entered manually. If the destination folder does not exist, it will be created automatically before extraction.
	Destination folder C:\DOCUME~1\zqing\LOCALS~1\Temp ■ Browse Installation progress
	Install Cancel
Nico II Embo	
AB MOS T CUDE	dded Design Suite 8.1 📃 🔲 🔀
	Image: Sector
	Image: Contract of the second state









株式会社日昇テクノロジー

低価格、高品質が不可能? 日昇テクノロジーなら可能にする

Nios II Embedded Design Suite 8.1 Setup		
Choose Destination Loc Select folder where setu	ation n will install files	
Select forder miere seta		
	Setup will install Nios II Embedded Design Suite 8.1 in the following folder To install to this folder, click Next. To install to a different folder, click Browse and select another folder.	
	Destination Folder	7
	C:¥altera¥81 Browse	
		-
InstallShield	< Back Next > Cancel	
Nios II Embedded Design S Select Program Folder	Guite 8.1 Setup	X
Nios II Embedded Design S Select Program Folder Please select a program	G <mark>uite 8.1 Setup</mark> folder.	
Nios II Embedded Design S Select Program Folder Please select a program	Suite 8.1 Setup folder. Setup will add program icons to the Program Folder listed below. You may type a new folder name, or select one from the existing folders list. Click Next to continue. Program Folder: Altera Existing Folders: Adobe Altera Apache HTTP, Server 2.2 Cadlus 1*-4>h DAEMON Tools Lite Hewlett-Packard Microsoft Office Microsoft Uffice Microsoft Visual Studio 2008 Microsoft Windows CE 5.0 Microsoft Windows SDK v6.0A	

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。



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



「**Complete**」を選択してください。



間違いがないかどうか確認し、問題がなければ「Next」を押します。



低価格、高品質が不可能?



Question	
2	Create Nios II Embedded Design Suite 8.1 shortcut on desktop?
	(はい公) (いいえ(M)

インストール完了すると、ショートカットをデスクトップに作るかどうか聞かれます。どちらでも選択できます。





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。



第三章 MAX II/Cyclone IIの初体験

3.1 Quartus II評価版にソースを読み込む

弊社のウェブサイトで MAX II 又は Cyclone II 用のサンプル・ソース・ファイルをダウン ロードできます。 Cyclone II 用ソース・ファイル: EP2C5.zip, EP2C8.zip MAX II 用ソース・ファイル: epm240.zip

ソース・ファイルを C:¥altera に展開します。その中に、幾つのサンプルがあります。例えば: Beep、LED、VGA、LCD、UART、Key など。一つのサンプルを紹介いたします。

エクスプローラまたはマイ コンピュータを起動して、

C:¥altera¥epm240¥led_water というフォルダを開いてください。

これらの中に、名前が led_water.qpf、Quartus II Project File となっているファイルがあ ります。これをダブル・クリックすると、Quartus II が起動して、led_water というプロ ジェクトが開きます。



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

🖏 Quartus II - C:/altera/FPGA/epm240/led_water/l	ed_water - led_water 📃 🖻 🔀
<u>File Edit View Project Assignments Processing Tools</u>	Window Help
🗋 🖻 🚅 🛃 🎒 🐇 🖿 💼 🗠 🖙 📔 🖬 🖿	<u> </u>
Maxigator × Entity Logic Cells LC Registers L MAX II: EPM240T100C5 Image: Cells Image: Cells Cells	Quartus II Software Download Subscription Edition Free Inabo Free Inabo
Hierarchy E Files d ⁴ Design Units Tasks	
Flow: Compilation	OUARTUS [®] II
✓ □ ► Compile Design 00:00	
	VCISION 0.1
✓ ⊕··· ► Fitter (Place & Route) 00:00 ✓ ⊕··· ► Assembler (Generate programming files) 00:00	
✓ 🗐 ► Classic Timing Analysis 00:00	View New Quartus II
EDA Netlist Writer	
<	Ø Documentation
* Type Message	
Sustan (Procession) Futralisto) Isto) Warning) Cr	ticelWarring & Free & Supressed & Elan /
Message:	



Quartus II の「Processing」メニューから「Start Compilation」を選択します。するとコ ンパイル処理が始まり、プロブレス・バーが働き始めます。コンパイルは数十秒で終了し ます。

🐇 Quartus II - C:/altera/FPGA/EP:	2C8/default_led/div - div
<u>File E</u> dit <u>V</u> iew <u>P</u> roject <u>A</u> ssignments	P <u>r</u> ocessing <u>T</u> ools <u>W</u> indow <u>H</u> elp
] D 📽 🖬 🕼 🎂 % 🖻 🖻 🕫	The Stop Processing Ctrl+Shift+C
Project Navigator	▶ Start <u>C</u> ompilation Ctrl+L
Entity Logic Co	🔊 Analyze Current <u>F</u> ile
Cyclone II: EP2C8Q208C8	Start •
L abd div Fa 54 (54)	Update Memory Initialization File
	Compilation Report Ctrl+R
	Start Compilation and Simulation Ctrl+Shift+K
	Generate Functional Simulation Netlist
	Start Si <u>m</u> ulation Ctrl+I
Historehu 🖹 Filos 🗗 Design Units	Simulation <u>D</u> ebug
	Report Ctrl+Shift+R
Tasks	
Flow: Compilation	
Task 🛒	Simulator Tool
🗸 🖃 🕨 Compile Design	Classic Timing Analyzer Tool
🖌 🖌 🗭 Analysis & Synthesis	💋 <u>P</u> owerPlay Power Analyzer Tool
🖌 🗸 🗄 🕨 🕨 Fitter (Place & Route)	00:00

3.2 USB-Blasterをインストールする

通常、MAX II/Cyclone II にコンフィグレーション・データを書き込むために、アルテラが 発売している専用ダウンロード・ケーブル(ByteBlaster MV や ByteBlaster II や USB 接続 タイプの USB-Blaster など)を購入しなければなりません。





弊社は専用ダウンロード・ケーブル USB-Blaster 同等のデバイスを提供しております。次 に示す手順に従って、USB-Blaster のデバイス・ドライバをインストールしてください。



USB-Blaster を USB ケーブルでパソコンと繋ぐと、自動的にこの画面が現れ、「**いいえ、 今回は接続しません**」を選択してください。

新しいハードウェアの検出ウィザー	۶.
	新しいハードウェアの検索ウィザードの開始
	お使いのコンピュータ、ハードウェアのインストール CD または Windows Update の Web サイトを検索して (ユーザーの了解のもとに) 現在のソフトウ ェアおよび更新されたソフトウェアを検索します。 <u>プライバシー ポリシーを表示します。</u>
	ソフトウェア検索のため、Windows Update に接続しますか?
	 ○はい、今回のみ接続します(Y) ○はい、今すぐおよびデバイスの接続時には毎回接続します(E) ○いいえ、今回は接続しません(T)
	続行するには、 〕ケヘ] をクリックしてください。
	< 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
検索とインストールのオブションを選んでください。
 ● 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 イバがインストールされます。 ■ リムーバブル メディア (フロッピー、CD-ROM など) を検索(M) ♥ 次の場所を含める(Q):
C¥altera¥81¥quartus¥drivers¥usb-blaster ▼ 参照(R) ●検索しないで、インストールするドライバを選択する(D) 一覧からドライバを選択するには、このオプションを選びます。選択されたドライバは、ハードウェアに最適のもの とは限りません。
< 戻る(B) 次へ(N) > キャンセル

 $USB\text{-}Blaster \ \mathcal{O} \ \texttt{F} \ \texttt{?} \ \texttt{?} \ \texttt{?} \ \texttt{S} \ \texttt{?} \ \texttt{S} \ \texttt{S}$





インストール中。

新しいハードウェアの検出ウィザー	k
	新しいハードウェアの検索ウィザードの完了
	次のハードウェアのソフトウェアのインストールが完了しました。
	Altera USB-Blaster
	[完了]をクリックするとウィザードを閉じます。
	< 戻る(B) 完了 キャンセル

インストール完了します。

3.3 書き込むソフトウェアを起動する

Quartus IIの「Tools」メニューから「Programmer」を選択すると、MAX II/Cyclone II



🐇 Quartus II - C:/altera/FPGA/epm240/led_water/led_water - led_water File Edit View Project Assignments Processing Tools Window Help Run EDA Simulation Tool 🗋 🚅 🗐 🎒 👗 🏗 💼 💼 🗠 🖂 led_w Run EDA Timing Analysis Tool Project Navigator = 🗫 Launch Design Space E<u>x</u>plorer Entity Logic Cells LC Registe 💩 MAX II: EPM240T100C5 🚺 TimeQuest Timing Analyzer 🗄 ---- 💦 led_water 52(0) 35 Advisors Ius II Integ e trial>3 🛞 Chip Planner (Floorplan and Chip Editor) 🍓 Design Partition Planner Design Autor Netlist Viewers < 😫 SignalTap II Logic A<u>n</u>alyzer 🛆 Hierarchy 🖹 Files 🗗 Design Units 🛲 In-System Memory Content Editor Tasks 🔳 Logic Analyzer Interface Edito<u>r</u> Flow: Compilation 🙀 In-System Sources and Probes Editor Task 📝 SignalProbe Pins... 🗸 🖃 🕨 Compile Design Programmer 庄 --- 🕨 Analysis & Synthesis \checkmark 📉 Mega<u>W</u>izard Plug-In Manager... \checkmark 🖻 --- 🕨 Fitter (Place & Route) 🕸 -- 🕨 Assembler (Generate programming files) 鴂 SOPC <u>B</u>uilder \checkmark

に回路を書き込むソフトウェア「Programmer ツール」が起動します。





Programmer ツールが起動したら、最初に書き込みケーブルのセットアップを行います。 「Hardware Setup」というボタンを押してください。

Hardware Setup			
Hardware Settings JTAG Settings Select a programming hardware setup to u only to the current programmer window.	use when programmi SB-Blaster (USB-0)	ng devices. This programm	ning hardware setup applies
Available hardware items: Hardware ByteBlaster USB-Blaster	Server Local Local	Port LPT1 USB-0	Add Hardware Remove Hardware
			Close

「USB-Blaster[USB-0]」を選択します。「Close」を押して、Hardware Setup ダイアログ を閉じたら、「Auto Detect」というボタンを押してください。これは、ケーブルの先にあ る CPLD/FPGA を自動認識する操作です。うまく CPLD/FPGA が認識されると、EPM240 又は EP2C5 又は EP2C8 という CPLD/FPGA が発見されるはずです。発見されない場合は、

- ケーブルが正しく接続されているか、
- ・ FPGA の場合は、ケーブルとボードの JTAG ポートを繋ぎますか
- CPLD/FPGA 基板に電源が入っているか

など、これまでの作業に問題がないか再度チェックをしてください。

CPLD/FPGA の認識に成功すると、「Add File」ボタンを押して、書き込みファイルを添加 します。*.pof は CPLD 用書き込みファイル、*.sof は FPGA 用書き込みファイルです。*.pof の右側にある Program/Configure と Verify の欄にチェックを入れて、「Start」ボタンを押 します。プログレス・バーが 100%まで達すれば、書き込み成功です。

CPLD/FPGA 用 I/F 基板上の LED が点滅しているのを確認してください。どうでしょうか?うまく点滅したでしょうか。

3.4 FPGAのコンフィギュレーションデバイスに書き込む

Cyclone II は SRAM ベースの FPGA なので、電源投入直後は中身が空の状態です。製品化



の際や、電源投入後に自動的に動作させる必要がある場合は、専用のコンフィギュレーションデバイス(EPCS4)に回路情報を書き込む必要です。

専用のコンフィギュレーションデバイスに書き込む手順:



先ず、USB-Blaster と FPGA ボードの ASP ポートを繋ぎます。 書き込むソフトウェア「Programmer ツール」が起動します。

💾 Quartus II - (C:/altera/FPGA/EP2	C8/default_led/di	v - div - [di	v.cdf]				×
<u>F</u> ile <u>E</u> dit P <u>r</u> oces	sin g <u>T</u>ools <u>W</u>indow				<u> </u>			
🚖 Hardware Setup	USB-Blaster [USB-0]		Mode: Active	e Serial Programmi	ing 💌 Progress:	C	1%	
🔽 Enable real-time IS	6P to allow background prog	ramming (for MAX II dev	ices)					_
🏴 Start 📐	File	Device	Checksum	Usercode	Program/ Venig Configure	Blank- Check	Examine	Si
📲 Stop	div.pof	EPCS4	06EFF068	00000000				
Auto Detect					\smile			
🗙 Delete					チェッ	ク		
Add File	・ <mark>「Start」</mark> ボ	タンを押す			を入れる			
🎬 Change File								
🗳 Save File								
💕 Add Device								
1 Up								
🔑 Down								
	<							>
For Help, press F1							NUM	11.

「Mode」に[Active Serial Programming]を選択します。「Add File」ボタンを押して、書 き込みファイル*.pof を添加します。*.pof の右側にある Program/Configure と Verify の欄 にチェックを入れて、「Start」ボタンを押します。プログレス・バーが 100%まで達すれば、 書き込み成功です。

書き込み成功した後、USB-Blaster を FPGA ボードの ASP ポートから抜いて、FPGA ボ ードに電源を再投入すると、どうの現象が出てきますか?



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

3.5 NIOS IIプロセッサーの初体験

エクスプローラまたはマイ コンピュータを起動して、

$C: \verb""" altera \verb""" EP2C8 \verb""" nios_led_sch"$

というフォルダを開いてください。

これらの中に、名前が nios_led_sch.qpf、Quartus II Project File となっているファイルが あります。これをダブル・クリックすると、Quartus II が起動して、nios_led_sch.qpf とい うプロジェクトが開きます。

他のプロジェクトと同じ手順でコンパイルして、Cyclone II ボードに書き込みます。 書き込み時、このような情報が出てきます。正式製品なら、アルテラ社からライセンスが 必要です。評価の場合は、そのまま「OK」ボタンを押します。

Quartus	п
٩	File C:/altera/FPGA/EP2C8/nios_led_sch/nios_led_sch_time_limited.sof contains one or more time-limited megafunctions that support the OpenCore Plus feature that will not work after the hardware evaluation time expires. Refer to the Messages window for evaluation time details.

書き込み完了したら、その画面が出てきます。「Cancel」ボタンを押さないでください。その画面をそのまま置いといてください。

OpenCore Plus Status	
Click Cancel to stop using O	IpenCore Plus IP.
Time remaining:	unlimited

Windows の「スタート」→「すべてのプログラム」→「Altera」→「NIOS II EDS 8.1」 から NIOS II 8.1 IDE が起動します。







NIOS II IDE の初起動の画面です。

NIOS II IDE のメニュー「File」→「Switch Workspace」を選択します。



Nios II C/C++ -	Nios II IDE
<u>File E</u> dit <u>N</u> avigate	Se <u>a</u> rch <u>P</u> roject T
<u>N</u> ew	Alt+Shift+N
Open File _e .	
<u>C</u> lose	Ctrl+W
C <u>l</u> ose All	Ctrl+Shift+W
📙 <u>S</u> ave	Ctrl+S
😹 Save <u>A</u> s	
😭 Sav <u>e</u> All	Ctrl+Shift+S
Rever <u>t</u>	
Mo <u>v</u> e	
Rena <u>m</u> e	F2
Re <u>f</u> resh	F5
Con <u>v</u> ert Line Delimi	iters To
👜 <u>P</u> rint	Ctrl+P
Switch Workspace	
N- Import	
A Export	
	A11-5-1
rroperties	Alt+Enter
E <u>x</u> it	

C:¥altera¥EP2C8¥nios_led_sch¥software を入力して、「OK」ボタンを押します。

💽 Workspace Launcher	
Select a workspace Nios II IDE stores your projects in a folder called a workspace. Choose a workspace folder to use for this session.	
Workspace: C:¥altera¥nios_led_sch¥software	▶ Browse
?	OK Cancel

NIOS II IDE が再起動します。





左側の「board_diag_0」でマウスの右ボタンをクリックして、「Build Project」を選択して、 ビルドを開始します。

🌉 Nios II C/C++ -	- board_diag.c - Nios II I	DE
<u>F</u> ile <u>E</u> dit Refac <u>t</u> or	<u>N</u> avigate Se <u>a</u> rch <u>P</u> roject	Tools <u>R</u> un <u>W</u> indow <u>H</u> elp
i 📬 • 🔛 🖆 📾	💼 • 🚳 • 💽 • 🞯 •	🏇 • 🔘 • 隆 • 😕 🔗
Nios II C/C++ Pro	jects 🗙 📃 🖬	board_diag.c 🛛
<	> -> 👰 🖪 🔗 🏹 🗌	/********
표 😂 altera.compon	ents	* Copyright (c) 2006 Alte
🗊 😂 board_diag_0		* All rights reserved. Al
🗄 😂 board_diag_0	Ne <u>w</u>	 ubject to the License
	Go Into	******
	Open in <u>N</u> ew Window	Description
	Rebuild Index	*****
	Active Build Configuration	A program which provid
	Run As	Development Board. Th
	Dahua Aa	- Seven Segment Disp:
	Debug As	- The DO-D7 LEDs (loc
	Build Project	The classic "wall
	Clean Project	- UART test
		Tests UART funct:



株式会社日昇テクノロジー	
--------------	--

低価格、高品質が不可能? 日昇テクノロジーなら可能にする

Nios II C/C++ - board_diag.c - Nios	II IDE
<u>Eile E</u> dit Refac <u>t</u> or <u>N</u> avigate Se <u>a</u> rch <u>P</u> ro	ject Tools <u>R</u> un <u>W</u> indow <u>H</u> elp
i 📬 • 🔚 😐 📾 i 🎯 • 🚳 • 💣 • G)• 🎄• () • 💁 • 😕 🛷 🔂 • ½ - 🖓 - 🌣
Nios II C/C++ Projects 🗙 📃 🗖	ic board_diag.c ⊠ □ 🗄 Our
Image: Constraint of the second s	/*************************************
Build Project Operation in progres Building board_diag_0_syslib	s
	Run in Background Cancel Details >> * * JTAG UART device. * - The LCD Display Bigslave a boott process

ビルド完了すると、左側の「board_diag_0」でマウスの右ボタンをクリックして、「Run As」 →「Nios II Hardware」を選択して、Cyclone ボードにプログラムをダウンロードします。

🛃 Nios II C/C++ - board_diag.c - Nios II IDE				
<u>F</u> ile <u>E</u> dit Refac <u>t</u> or <u>N</u> avigate Se <u>a</u> rch <u>P</u> roject	Tools <u>R</u> un <u>W</u> indow <u>H</u> elp			
i 📬 • 🔚 🖮 i 🎯 • 😂 • 🖸 • 🤡 •	🏇 • 🔘 • 🏊 • 🅭 🖋 📾 • 🖢 - 🖗] - % ⊳ ¢		
💽 Nios II C/C++ Projects 🗙 📃 🖬	board_diag.c 🛛 🗖 🗖	📴 Outline		
← → @ E & ▼ ⊕-@ altera.components	/*************************************			
ereste board_diae New	* All rights reserved. All use (• • •		
B - È board_diag Go Into	- ************************************	• • • •		
Open in <u>N</u> ew Window	*	💿 S		
Rebuild Index	- * Description * ***********	• • • • • • • • • • • • • • • • • • •		
Active Bui <u>l</u> d Configuration	* A program which provides a me	····· •		
Run As	Nios II Hardware			
Debug As	Nios II Instruction Set Simulator			
<u>B</u> uild Project	🗃 Nios II ModelSim	💿 Make T		
Clean Project	📣 Lauterbach Nios II Instruction Set Simulator			
Сору	* * JTAG UART device.	🕀 😂 bo		
💼 <u>P</u> aste	* - The LCD Display	🗄 🗁 po		
Delete	 Displays a short message 			


若しこの情報が出てきたら、

You have more than one JTAG cable available so you must use the --cable option to choose between them (or open the "Run/Run" or "Run/Debug" dialog and go to the "Target Connection" tab). No --cable option was provided (or you selected Automatic)

NIOS II IDE のメニュー「Run」→「Run」を選択します。





「Target Connection」を押して、JTAG cable で「USB-Blaster [USB-0]」を選択します。 「Run」ボタンを押します。

🛃 Run		X
Create, manage, and run con	figurations	
Image: Second system	Name: board_diag_0 Nios II HW configuration Image: Image: Target Connection Image: Target Connection JTAG cable: Image: Target Connection Image: Target Connection Image: Target Connecti	Help Refresh Refresh Refresh
< No. 100 No.		Apply Revert
0		<u>R</u> un Close



Cyclone II ボードにプログラムをダウンロード完了すると、自動的にプログラムを実行しま す。Cyclone II ボードとダイアログするコンソールが出てきます。



コンソールでメニューを選択して、NIOS II システムとダイアログできます。例えば、「a」 と「enter」キーを入力すると、Cyclone II ボードのユーザ LED を点灯します。



日昇テクノロジーなら可能にする

4.1 プロジェクトを作成する

Windows を起動し、「スタート」メニューから Quartus II Web Edition を起動します。

Quartus II 評価版上では、これから作る回路が一つのプロジェクトとして使扱われます。 まずは、新しいプロジェクトを作成しましょう。

新規にプロジェクトを作成するには、Quartus II 評価版の「File」メニューから「New Project Wizard」を選択し、プロジェクト作成ウィザードを起動します。このウィザードを使えば、 ダイアログに表示された質問に答えていくだけで、簡単にプロジェクトを作ることができます。

New Project Wizard: Introduction	
The New Project Wizard helps you create a r	new project and preliminary project settings, including the following:
 Project name and director Name of the top-level des Project files and libraries Target device family and EDA tool settings 	ory sign entity I device
You can change the settings for an existing p the Settings command (Assignments menu). Y functionality to the project.	project and specify additional project-wide settings with You can use the various pages of the Settings dialog box to add
Don't show me this introduction again	
	< Back Next > Finish キャンセル

最初に、New Project Wizard に関する説明が表示されるので、そのまま「Next」を押しま す。すると、プロジェクトの名前や保存場所を聞いてきます。各問いに対して、次のよう に書き入れて「OK」ボタンを押してください。



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

New Project Wizard: Directory, Name, Top-Level Entity [page 1 of 5]
What is the <u>w</u> orking directory for this project?
C:\altera\nios2_sdram_led
What is the name of this project?
nios2_sdram_led
What is the name of the top-level design entity for this project? This name is case sensitive and must exactly match the entity name in the design file.
nios2_sdram_led
Lise Existing Project Settings
<back next=""> Finish キャンセル</back>

パス名やプロジェクト名、モジュール名に漢字や空白などの特殊な記号が含まれていると、 ツールによってはうまく動かないものがあるので、半角のアルファベットや数字などを組 み合わせただけの単純な名前にしてください。

プロジェクト作成ウィザードで入力した三つめの項目は、回路を階層的に設計する場合に 最上位の階層に置くエンティティの名前です。エンティティとは、あるまとまった機能を もった回路のことです。





これから作成する回路のエンティティの階層構造です。FPGA 内部は、nios_sdram_led と いうエンティティで作られています。そして、nios_sdram_led というエンティティの中に、 nios_sdram というエンティティがあります。さらに、nios_sdram というエンティティの 中に NIOS II コアと周辺回路など幾つのエンティティがあります。

※ MAXII は容量が不足ですので、ソフトプロセッサ NIOS II を搭載できません。

このように、小さなエンティティを組み合わせて大きな回路を作って階層構造にすると、 効率良く開発できます。

ダイアログの三つの欄に入力したら「Next」を押します。すると、

Select the design files you want to include in the project. Click Add All to add all design files in the project directory t the project. Note: you can always add design files to the project later.	to
Eile name: Add	
File name Type Library Design entry/s HDL version Add All	
<u>H</u> emove	
<u>Properties</u>	5
Down	
Specify the path names of any non-default libraries.	
くBack Next > Finish キャンセ	2JL

この画面が現れます。プロジェクトに追加したいファイルがある場合はここで追加できま す。今回は不要なので、なにも選択せずに「Next」を押します。



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

New Project Wizard: Fam	ily & Dev	vice Set	tings [pa	ge 3 of	5]			×
Select the family and device you want to target for compilation.								
Device family		Show in 'Available device' list						
Eamily: Cyclone II				-	Pac <u>k</u> age:	An	y QFP	-
Devices: All				-	Pin <u>c</u> ount:	208	3	•
– Target device					Sp <u>e</u> ed grade	e: 8		•
Auto device selected b	y the Fitter					dvanced de	vices	
Specific device selecte	d in 'Availat	ole devices'	list		🔲 HardCop	oy compatib	ile only	
Ausilable devices:				[
A <u>v</u> allable devices.	Core v	1 E o	lleer I/	Memor	Embed	PU	Global	
EP2C5Q208C8	1.2V	4608	142	119808	26	2	8	_
EP2C5Q20818	1.2V	4608	142	119808	26	2	8	
EP2C8U208C8	1.2V	8256 9256	138	165888	36	2	8	
	1.29	0230	150	105000	50	2	0	
Companion device								
HardCopy:								-
☑ Limit DSP & RAM to Ha	rdCopy dev	ice resourc	es					
		< 6	Back	Next	>	Finish	++1)	セル

使う FPGA デバイスの選択ダイアログで、まず「Family」というダウン・メニューで「Cyclone II」を選択し、「Available devices」の中から「EP2C8Q208C8」を選びます。選んだら、「Next」を押します。



株式会社日昇テクノロジー

低価格、高品質が不可能? 日昇テクノロジーなら可能にする

New Project Wizard: EDA Tool Settings [page 4 of 5]	
Specify the other EDA tools in addition to the Quartus II software used with the project.	
Design Entry/Synthesis	
Tool name: None>	-
Format:	3
Run this tool automatically to synthesize the current design	
Simulation	
Tool name: <none></none>	-
Format:	3
Run gate-level simulation automatically after compilation	
Timing Analysis	
Tool name: <none></none>	3
Format:	3
Run this tool automatically after compilation	
(Pack Nevt) Finish Atu'd	
く Back Next / Finish キャクノ	270

この画面では、このプロジェクトで使用したい Quartus II 評価版以外の外部ツールを選択 できます。今回は Quartus II 評価版だけで最後まで設計を行うので、何も選択せずに「Next」 を押します。



New Project Wizard: Summary	[page 5 of 5]	×
When you click Finish, the project will	be created with the following settings:	
Project directory:		
C:/altera/nios2_sdram_led/		
Project name:	nios2_sdram_led	
Top-level design entity:	nios2_sdram_led	
Number of files added:	0	
Number of user libraries added:	0	
Device assignments:		
Family name:	Cyclone II	
Device:	EP2C8Q208C8	
EDA tools:		
Design entry/synthesis:	<none></none>	
Simulation:	<none></none>	
Timing analysis:	<none></none>	
Operating conditions:		
Core voltage:	1.2V	
Junction temperature range:	0-85 ~C	
		_
	K Back Next > Linish キャンセル	

内容を確認して「Finish」を押します。



۵ 🍪	uartu	s II -	C:/alter	a/nios2_sdra	am_led/nios	s2_sdra	m_led -	nios2	_sdram_led		
<u>F</u> ile	<u>E</u> dit	⊻iew	<u>P</u> roject	<u>A</u> ssignments	Processing	<u>T</u> ools	<u>W</u> indow	<u>H</u> elp			
D	2	1 🕼	6 8	B C 10		_sdram	_led		• 💥 .	/ 🏈	ම
Proje	ct Navig	ator —				∃ • ×					
Enti	ty						_				
	Cyclo	ne II: El	P2C8Q208	08							
	- 🏓 ni	ios2_sd	ra 者				Qua	artus l	I Software		IINS
									Download	with (Quart
							Sul	script	ion Edition		Fre
								Free 3	0-day Trial		
									X		-
<u></u>	Hierarch	v 🖹 F	Files 🗗 De	esign Units							
Tasks						- x					
Flow:	Com	pilation				-					
						17					
lask		0	- ile Desie	_		lime			\prec '		
		Dom	pile Desigi Analysis A	n Suudhaain							
	4	P 🏲	Anaiysis &	Synthesis							

これは nios_sdram_led プロジェクト作成直後の画面です。

左上の Project Navigator の中に、デバイス名とトップ・レベル・エンティティ名 (nios_sdram_led)が入れ子になって表示されています。これは「EP2C8Q208C8 というデ バイスの中に nios_sdram_led というエンティティが入っていますよ」という意味です。

このように、Project Navigator 欄を見れば、今開いているプロジェクトがどのような構造 をしているかを確認できます。Project Navigator 欄を閉じてしまった場合は、「View」メ ニューの中の「Utility Windows」という項目の中に「Project Navigator」があるので、こ ちらを選択すればもう一度表示できます。



4.2 エディタで回路図を描く

4.2.1 トップ・エンティティを作成する

「File」メニューから「New」を選択します。



「 Block Diagram/Schematic File」を選択します。「OK」を 押します。



🖏 Quartus II - C:/altera/nios2_sdram_led/nios2_sdr	am_led - nios2_sdram_led - [Block1.bdf]
🔁 Eile Edit View Project Assignments Processing To	ools <u>W</u> indow <u>H</u> elp
📙 🗅 🗃 🖬 🞒 👗 🖡 🛍 🛍 🗠 🗠 🛛 🖬 nios2_sdran	n_led 💽 🔀 🖉 🦑 👘 🕨 🕏
Project Navigator 🛁 🔺 🗙	Block1.bdf
Entity	
🛆 Cyclone II: EP2C8Q208C8	
• nios2 sdra… An	
	A
	Ð □
	「「「同路図モディター」
	🖷 🕘 ::::::::::::::::::::::::::::::::::
🛆 Hierarchy 🖹 Files 🗗 Design Units	🔲 🛤 💠
Tasks 🔺 🔺 🗙	
Flow: Compilation	1
Task II'	
Complete Design	
🖓 🗝 📂 Analysis & Synthesis	
これが同時回来ご,カベナ ここに同時記日	ATT FL インソンド ODID/FDCA 由如の同時

これが回路図エディタです。ここに回路記号を配置していけば、CPLD/FPGA 内部の回路 を設計できます。

Block1.bdf という名前は、Quartus II 評価版が勝手に付けた名前です。これから作りたいのは nios_sdram_led というエンティティです。回路図とエンティティを対応させるために、 ファイル名を nios_sdram_led.bdf として保存します。

8	Qu	artus II	[- C:/	altera/r	nios2_sdram	_led/nios2_	sdran	n_led - ni	os2_sdram_
12	Eil	e <u>E</u> dit	<u>V</u> iew	<u>P</u> roject	<u>A</u> ssignments	Processing	Tool	s <u>W</u> indow	<u>H</u> elp
Pro Er		<u>N</u> ew Open <u>C</u> lose New Pro Open P <u>r</u> Convert Save Pro Clos <u>e</u> Pr	oject <u>W</u> iz oject MAX+P ojec <u>t</u> roject	zard " <u>L</u> US II Pro	Ctrl+N Ctrl+O Ctrl+F4 Ctrl+J pject	r≊ ∫nios2_so	dram_	led Block Block C A - - - - - - - - - - - - -	Second Seco
		<u>S</u> ave Save <u>A</u> s	3	oport Soci	Ctrl+S		[
Tas Flor		Eile Pro	perties	eport 3801	uuri MS	•	×	<u>∧</u>	
Ta		Create <u>2</u>	/ Update	Э	•	Ti	ime		



「File」メニューから「Save As...」を選び、ファイル名を nios_sdram_led としてください。保存する場所は先ほど指定したプロジェクトのワーキング・ディレクトリで、

 $C{:}{\tt ¥altera}{\tt ¥nios_sdram_led}$

です。「Save As...」を選ぶと自動的に、このフォルダが開かれます。

Save As						
保存する場所①:	🗁 nios2_sdram_led		•	← 🗈	r 🗄	
	🛅 db					
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						
S						
マイ ネットワーク						
	ጋ _ሚ ፈ ዚዲ (አ)ኑ	pice2 edram led			-	(保存(S))
						And F and
	ファイルの種類(上):	Block Diagram/Schematic File (*.bdf)		▼	7 77271
	🔽 Ac	ld file to current <u>p</u> roject				

保存が終わると、ウィンドウの名前が nios_sdram_led.bdf に変わります。

🖏 Quartus II - C:/altera/nios2_sdram_led/nios2_sdr	am_led = nios2_sdram_led = [ni
🔁 <u>F</u> ile <u>E</u> dit <u>V</u> iew <u>P</u> roject <u>A</u> ssignments P <u>r</u> ocessing <u>T</u> o	ools <u>W</u> indow <u>H</u> elp
📙 🗅 🚅 🖬 🕼 🕌 🕺 🏝 💼 🔛 🗠 🛛 🗖 nios2_sdran	n_led 💽 🔀 🖉 🍕
Project Navigator 🛁 🔺 🗙	🚏 nios2_sdram_led.bdf
Entity	
A Cyclone II: EP2C8Q208C8	
IIIII → nios2_sdra 🔓	
	K A
	Ð 🔲 💠
	•
	↔
/ Mineralus B Files d Design Units	
Allerarchy El Files Br Design Units	
Taska	
Flow: Compilation	



4.2.2 作画手順

シンボルを置く

「Symbol Tool」というボタンを押すと、このようなダイアログが開きます。



Libraries 欄に、[+]記号をクリックすると、中身が表示されます。中身のシンボルを選択して「OK」ボタンを押します。マウス・カーソルに選択されたシンボルがくっついた状態になるので、シンボルを配置したい場所に移動して、マウスの左クリックを押します。配置できた点で ESC キーを押してカーソルを元に戻してください。



②シンボル同士を接続する

h	_ 💥
🔡 nios_	led_sch.bdf
	PIN_23
Ţ₽ Ę€	Orthogonal Node Tool

「Symbol Tool」の少し下にある「Orthogonal Node Tool」と いう鍵状のアイコンを選択します。マウス・カーソルをシンボ ルのピンに併せて左ボタンを押し、そのまま離さずに他のピン までドラッグします。すると、ドラッグの開始点と終了点が線 で結ばれます。

同様にしてほかの線も配置してください。接続が終わったら SC キーを押し、元のカーソルに戻します。接続を間違えた 合は、一度 ESC キーを押して元のカーソルに戻し、不要な 配線をクリックして Delete してください。

③INPUT 端子と OUTPUT 端子の名前を変える

INPUT シンボルと **OUTPUT** シンボルは、エンティティ外側と接続するための端子です。 端子の名前は pin_name という素っ気ないものになっています。

これでは、エンティティを使うときに、いったい何のための端子なのかわからなくなりま す。そこで、わかりやすい名前に変更しておきます。

端子の名前を変更するためには、ESC キーを押して通常のカーソルに戻した後に、回路図 上の INPUT/OUTPUT シンボルをダブル・クリックします。

Pin Properties		×
General Format		
To create multip enter a comma-s	le pins, enter a name in AHDL bus notation (for example, "name[30]"), or eparated list of names.	
<u>P</u> in name(s):	clik	1
<u>D</u> efault value:	VCC	

Pin name 欄にわかりやすい名前を入力します。

エンティティの回路図入力作業は終わったら、「File」メニューから「Save」を実行して回 路を保存してください。



ある回路図の様子。

4.2 書き込み前の二つの作業

4.2.1 回路図をコンパイルする

入力した回路図から CPLD/FPGA に書き込むデータを生成するためには、作成した回路図 をコンパイルしなければなりません。

Quartus II の「Processing」メニューから「Start Compilation」を選択します。するとコ ンパイル処理が始まり、プロブレス・バーが働き始めます。コンパイルは数十秒で終了し ます。

🖏 Quartus II - C:/altera/FPGA/EP	2C8/default_led/div - div	
<u>File Edit View Project Assignments</u>	Processing Tools Window Help	
D 📽 🖬 🕼 🕼 🕺 🏜 🖻 🗠	Stop Processing Otrl+Shift+C	🥝 🤻
Project Navigator	Start <u>C</u> ompilation Ctrl+L	
Entity Logic C	🛃 Analyze Current <u>F</u> ile	
Cyclone II: EP2C8Q208C8	St <u>a</u> rt 🕨	1
54 (54)	Update Memory Initialization File	
	Compilation <u>R</u> eport Ctrl+R	wim Qu
	Start Compilation and Simulation Ctrl+Shift+K	
	Generate Functional Simulation Netlist	
8	≿ Start Si <u>m</u> ulation Ctrl+I	
Hierarchy 🖹 Files 🗗 Design Units	Simulation <u>D</u> ebug	
	🐏 Simulation Report Ctrl+Shift+R	
Tasks Flow: Compilation	E Compiler Tool	
Taok II	📇 Simulator Tool	
✓ ⊡ ► Compile Design	🖄 Classic Timing Analyzer Tool	
✓ ⊕► Analysis & Synthesis	💋 <u>P</u> owerPlay Power Analyzer Tool	
🗸 🗄 🕨 🕨 Fitter (Place & Route)	00:00	



正常にコンパイルが終了しなっかた場合は、何か手順を間違えているか、回路図の入力を ミスしている可能性があります。画面に表示されたメッセージを読めば、どのようなミス があるかある程度知ることができます。

4.2.2 回路図の入出力とCPLD/FPGAの端子を関連づける

回路図上の端子名と CPLD/FPGA のピン番号との対応をピン・アサインと言います。ピン・ アサインは、「Assignments」メニューの「Pins」を選択します。

🐇 Quartus II - C:/altera/l	FPGA/EP2C8	FEP2C8_P	ER_2_le	d_water	/led_w
🚹 <u>F</u> ile <u>E</u> dit <u>V</u> iew <u>P</u> roject	<u>A</u> ssignments	Processing	<u>T</u> ools	<u>W</u> indow	<u>H</u> elp
🗋 🗅 🚅 🖬 🎒 🎒 👗 🖻	<u> </u>				•
Project Navigator	ns <u>P</u> ins				
Entity	🔮 <u>T</u> iming And	alysis Settings	S		
♦ Cyclone II: EP2C8Q208C8	🏆 <u>E</u> DA Tool∶	Settings			
🗄 💦 led_water 🏭	_ ∕ <u>S</u> ettings…	Cti	rl+Shift+l	E [::	
	Classic Ti	ning Analyzer	<u>W</u> izard		
	🥩 <u>A</u> ssignmer	t Editor 🛛 Ctr	l+Shift+A	۹ H H	· · · · · ·
	🛤 Pin Planna	 Ots 	l+Shift+N	a tii	

Assignments Editor というウィンドウが開きます。

		Groups			- X			
i = _		Named:	•				10 10 10 10 0 TO 10 10 10 10 10 10 10 10 10 10 10 10 10	000 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
. 🔍 🔳	I 🛃 📜	Nod	e Name	Direction	Loca			
_ ‡ ⊒			0]	Output Group			Top View	
	→ ← ∞ _∞	< <ne< th=""><th>ew node>></th><th></th><th></th><th></th><th>Wire Bond</th><th>1</th></ne<>	ew node>>				Wire Bond	1
	1 🗱 🗄							1
_ ! 6 7 1 @	2							
5 6-8 65 -	6							
1 🏅 🎽	4						Cyclone II	
							EP2C8Q208C8	
	2							24
						- Marchester of the	20070-0-075 2+49262470-0670-00-075 2+45-6	02: TF450
					>	ļ		
Named:	×	💌 «» Edit	XV				Filter: Pins: all	•
	1	Node Name	Direction	Location	I/O	Bank	VREF Group	~
1	🖻 do	ck	Input	PIN_23	1		B1_N0	3.
2	💿 led	[7]	Output	PIN_112	3		B3_N1	3.
3	💿 led	[6]	Output	PIN_106	3		B3_N1	3.
4	💿 led	[5]	Output	PIN_102	4		B4_N0	3.
2		۲ <u>م</u> ا	Output	DTNI QQ	4		R4 NO	2
								-

指定したいのはピンの場所、つまり Location です。行の Location の列をダブル・クリック して、ピンの番号を選択します。



Named:		▼ ≪≫ Edit:	X V PIN_23					Filter: F	^v ins: all	ŀ	•
		Node Name	Direction	Location		I/O Ban	ik	VREF	F Group		~
1		clock	Input	PIN_23	•	1		B1_N0		3.:	
2	•	led[7]	Output	PIN_23	_	IOBANK_1	Dedicate	d Clock	CLK0, LVDS	CLKO	~
3	•	led[6]	Output	PIN_24		IOBANK_1	Dedicate	d Clock	CLK1, LVDSC	CLK0	
4	•	led[5]	Output	PIN_27		IOBANK_1	Dedicate	d Clock	CLK2, LVDSC	CLK1	_
5	.	lad[4]	Output	PIN_28		IOBANK_1	Dedicate	d Clock	CLK3, LVDSC	CLK1	
				PIN_30		IOBANK_1	Row I/O		LVDS7p, DP0	CLK1	
				PIN_31		IOBANK_1	Row I/O		LVDS7n		
				PIN_33		IOBANK_1	Row I/O		LVDS6n		
				PIN_34		IOBANK_1	Row I/O				~

ピン・アサインが完成したら、「File」メニューから「Save Project」を選択してプロジェ クト全体を保存して、再度コンパイルを実行します。

コンパイル成功すれば、書き込むデータ*pof又は*.sofを生成します。データを CPLD/FPGA に書き込みましょう。

 ※ 提供されたサンプルの一部又は全部、回路図で作成したものではありません。VHDL 又は Verilog という HDL(Hardware Description Language: ハードウェア記述言語)を 使って、作成しました。VHDL/Verilog の使い方はほかの資料を参照してください。

第五章 NIOS IIシステム・モジュールの設計

Cyclone II シリーズ FPGA はソフトプロセッサ NIOS II システムを搭載できます。NIOS II は、32bit CPU、命令・データキャッシュ搭載、最大 250MHz 動作します。とてもハイパ フォーマンスなCPUです。開発環境は、ポピューラーなGCCで、無償提供されていま す。開発環境が信じられないほど簡単です。H8・PIC を、お使いの方に使い比べて頂 きたいシステムです。

第四章の CPLD/FPGA の開発入門は定番シンボル(例えば 74 シリーズロジックなど)で回路 図を設計します。今回は回路図に CPU を載せます。

まず、Quartus II を起動して、第四章に基づいて、ある空のトップ・エンティティを作り ます。

「Tools」メニューから「SOPC Builder」を選択し、SOPC Builder を起動します。



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

🐝 Quartus II - C:/altera/nios2_sdram_led/nios2_	sdram_led - nios2_sdram_led - [nios2_s	sdram_led.bdf]
🔁 Eile Edit View Project Assignments Processing	<u>T</u> ools <u>W</u> indow <u>H</u> elp	
🗋 🖸 🚅 🖬 🎒 🎒 👗 🖻 💼 🗠 🖂 🔲 nios2_sc	Run E <u>D</u> A Simulation Tool	👌 💿 🕨 🦻 🙀
Project Navigator	Run <u>E</u> DA liming Analysis lool	
Entity	🐏 Launch Design Space Explorer	
Cyclone II: EP2C8Q208C8	🚺 TimeQuest Timing Analyzer	—
	<u>A</u> dvisors	•
	🗞 Chip Planner (Floorplan and Chip Editor)	
	🝓 Design Partition Planner	
	Netlist <u>V</u> iewers	•
	📓 SignalTap II Logic A <u>n</u> alyzer	
AP Hierarchy E Files AP Design Units	🚗 In-System Memory Content Editor	
Tasks	🔳 Logic Analyzer Interface Edito <u>r</u>	
Flow: Compilation	🙀 In-System Sources and Probes Editor	
Task 🗗	i <u>S</u> ignalProbe Pins	
🖃 🕨 Compile Design	Brogrammer	
🗄 🕨 Analysis & Synthesis	· · · · · · · · · · · · · · · · · · ·	¦```````````````````````````
É▶ Fitter (Place & Route)	📉 Mega <u>W</u> izard Plug-In Manager	
🗄 🕨 Assembler (Generate programming files)	SOPC Builder	
🗄 🕨 Classic Timing Analysis	Tel Scripts	
🗄 🕨 EDA Netlist Writer]	

起動の様子です。

🖞 Quartus II - C:/altera/nios2_sdram_led/nios2_sdram_led - nios2_sdram_led - [nios2_sdram_led.bdf]							
Elle Edit View Project Assignments Processing	Tools Window Help						×
🗋 🖻 🗃 🎒 🎒 👗 🖻 💼 🗠 🖂 🔤 nios2_s	ram_led	- 💢 / 🏈 🦁	🗞 💿 🕨 🕏	1 10 10 1	🧿 📩 🕘	ی 😓 🍪	*
Project Navigator 4	× 💼 nios2_sdram_	_led.bdf					
▲ Cyclone II: EP2C80208C8	₩ A C T T T T T T T T T T						^
A Hierarchy	SOPC Build From Concept of Timer USB	to System in N PCI DMA SDRAM Controll	linutes PU VART	ation ic			
Program Device (Open Programmer)				iii			✓
Type Message System & Processing & ExtraInfo & Info & Warning ;	Critical Warning À Erro	r à Suppressed à Fla	g /				
Message:	- 0						✓ Locate
For Help, press F1	JP 🖮 🗲	🖁 A般 😂 🥔 🧷 🛛	🖉 🕄 сарб 🗧 🖃	22, 284	[h+】+#	Idle	NUM
- スタート 🏟 Outlook Express 🛛 😭 04	altera	🖁 Quartus II - C:/altera		- ペイント			< 🚺 909



ダイアログに NIOS II システムの名前「nios_sram」を入力してください。名前は必ずトッ プ・エンティティの名前と異なります。

😉 Create New System – Altera	a SOPC Builder – unnamed.sopc	: (C:¥altera¥nios2	_sdram_led¥uni	named.sop				
Elle Edit Module System View Iools Help								
System Contents System Generation								
🖳 Altera SOPC Builder	Target	Clock Settings						
Nios II Processor Bridges and Adapters	Device Family: Cyclone II	Name	Source		MHz	Add		
Interface Protocols						Remove		
⊕-Peripherals ⊕-PI I				Olavak.	Deer	5-4		
€-USB	Use Mc L Create New	5ystem		CIOCK	Dase	Eng		
⊞Video and Image Processing	System Name: nios_	sdram						
	Target HDL: 🧿 Veril	og						
		L						
	1							
< >		K Cancel						
	<	(>		
New Edit Add	Remove Edit	Move Up	▼ Move <u>D</u> own	Addr	ess <u>M</u> ap	<u>Filter</u>		
Info: No errors or warnings.								
	Exit Help I Prev Next D Generate							

システムの周波数を入力します。今回の例は 50MHz です。周波数の欄でダブル・クリック して、周波数を入力できます。

💶 Altera SOPC Builder - nios_sdram.sopc* (C:¥altera¥nios2_sdram_led¥nios_sdram.sopc) 🛛 🔲 🔲 🔀						
<u>F</u> ile <u>E</u> dit <u>M</u> odule <u>S</u> ystem <u>∨</u> iew]	<u>T</u> ools <u>H</u> elp					
System Contents System Generation						
Altera SOPC Builder	Target	Clock Settings				
Nios II Processor	Device Family: Cyclone II	Name Source	MHz	Add		
Bridges and Adapters		clk_0 External	50.0	Remove		
Legacy Components						
Memories and Memory Controllers Perinberals						
teripriciais t≢PLL	Use Module Name	Description	Clock Base	End		
USB		- \ ·				
w video and image Processing						
<						
New Edit Add	Remove Edit	▲ Move Up	Address <u>M</u> ap	<u>F</u> ilter		
Info: No errors or warnings						
			_			
	E <u>x</u> it Help	Prev Next Generate				



左側の「Nios II Processor」をダブル・クリックして、Nios II CPU コアを添加します。

¼ Altera SOPC Builder – nios_sdram.sopc* (C:¥altera¥nios2_sdram_led¥nios_sdram.sopc)							
<u>E</u> ile <u>E</u> dit <u>M</u> odule <u>S</u> ystem <u>V</u> iew <u>T</u> o	ools Nios II <u>H</u> elp						
System Contents System Generation							
Afera SOPC Builder Nios Il Processor Bridges and Avapters Generative face Protocols Legacy Components Memories and Memory Controllers	Clock Settings Device Family: Cyclone II Name Source MHz clk External 50.0			Add Remove			
B - Video and Image Processing	Use Con Module Name	Description Nios II Processo Avalon Memory	or Mapped Master	Clock Base	End		
	data_master jtag_debug_mode	Avalon Memory ule Avalon Memory	Mapped Master Mapped Slave	0 x	IRQ 0 00000800 0x00000fff		
	<						
New Edit Add	Remove Edit	Move Up	Move Down	Address <u>M</u> ap	<u>Filter</u>		
To Do: cpu_0: No reset vector has been specified for this CPU. Please parameterize the CPU to resolve this issue To Do: cpu_0: No exception vector has been specified for this CPU. Please parameterize the CPU to resolve this issue Warning: cpu_0: Reset vector and Exception vector cannot be set until memory devices are connected to the Nios II processor							
	Exit Help	Prev Next	Generate				



Nios II は性能の違う 3 種類の CPU コアが提供されていますが、今回はデフォルトの高性 能のグレート(Nios II/f)を使用します。このグレードは、デフォルトでハードウェア乗算器 と命令・データキャッシュ、JTAG デバッガが組み込まれています。今回はこれをそのまま 使用します。「Finish」ボタンを押してください。

😃 Nios II Processor	- cpu_0					
Mios	II Processor			About Documentation		
Parameter Settings						
Core Nios II Cache	es and Memory Interfaces >	Advanced Features > MM	1U and MPU Settings 🔰 J1	TAG Debug Module > Custom Instructions >		
Core Nios II						
Select a Nios II core:						
	○Nios II/e	○Nios II/s	● Nios II/f			
Nios II Selector Guide Family: Cyclone II f _{system} : 50.0 MHz cpuid: 0	RISC 32-bit	RISC 32-bit Instruction Cache Branch Prediction Hardware Multiply Hardware Divide	RISC 32-bit Instruction Cache Branch Prediction Hardware Multiply Hardware Divide Barrel Shifter Data Cache Dynamic Branch Predictio	n		
Performance at 50.0 MHz	Up to 5 DMIPS	Up to 25 DMIPS	Up to 51 DMIPS			
Logic Usage	600-700 LEs	1200-1400 LEs	1400-1800 LEs			
Memory Usage	Two M4Ks (or equiv.)	Two M4Ks + cache	Three M4Ks + cache			
Hardware Multiply: Embe	dded Multipliers	Y Hardware Divide				
Reset Vector: Memo Exception Vector: Memo	ory:	✓ Offset: 0x0 ✓ Offset: 0x20				
🗌 Include MMU						
Only include the MMU who	en using an operating system th	nat explicitly supports an MMU				
Fast TLB Miss Exception	Vector: Memory:	~ O	ffset: _{0×0}			
Include MPU						
A Warning: Reset vector	and Exception vector cannot b	e set until memory devices are	connected to the Nios II proce	issor		
		🧯 UP' 🖮 🔫	A般 🗳 🥔 🧷 🕺	Cancel < Back Next > Finish		



EP2C8 基板の SDRAM を使うため、SDRAM コントローラを Nios II システムに組み込み ます。左側の「SDRAM Controller」を選択し、ダブル・クリックして SDRAM コントロ ーラを添加します。

🚇 Altera SOPC Builder – nios_sdram.sopc* (C:¥altera¥nios2_sdram_led¥nios_sdram.sopc) 🗧 🔲						
Elle Edit Module System View Iools Nios II Help						
System Contents System Generation						
Altera SOPC Builder	Target	Clock Settings				
Nios Il Processor Bridges and Adapters	Device Family: Cyclone II 💙	Name Source	MHz	Add		
		cik External	50.0	Remove		
Memories and Memory Control						
tation between the second sec						
	Use Con Module Name	Description	Clock Base	End		
		Nios II Processor	-11			
ODR SDRAM Cont	data master	Avalon Memory Mapped Master	IBQ)		
DDR SDRAM High DDR2 CDRAM Car	jtag_debug_module	Avalon Memory Mapped Slave	<u>∎</u> 0×0000800	0x00000fff		
DDR2 SDRAM Cor DDR2 SDRAM High	Sdram_0	SDRAM Controller				
BORD SURAWING	s1	Avalon Memory Mapped Slave	cik = 0x0200000	0x027fffff		
SDRAM Controller	sdram () s	1				
±-SRAW ⊻	Avalon M	emory Mapped Slave [avalon_slave 8.1]				
				<u>></u>		
New Edit Add	Remove Edit	▲ Move Up	Address <u>Map</u> <u>F</u> ilter.			
To Do: cpu_0: No reset vector has be	een specified for this CPU. Please parameterize	the CPU to resolve this issue				
To Do: cpu 0: No exception vector has been specified for this CPU. Please parameterize the CPU to resolve this issue						
	Exit Help	Prev Next Cenerate				



それぞれのメモリごとにデータのビット幅、チップ・セレクトの本数、バンク数、アドレスのロウ・カラムの本数、およびACスパックをデータ・シートから読み取り、間違いのないように入力します。

🖷 SDRAM Controller – sdram_0
SDRAM Controller
Parameter Settings
Memory Profile Timing
Presets: Custom
Data width
Bits: 16
Architecture
Chip select: 1 Banks: 4
Address widths
Row: 12 Column: 8
Share pins via tristate bridge
Controller shares dq/dqm/addr I/O pins
Tristate bridge selection:
Generic memory model (simulation only)
✓ Include a functional memory model in the system testbench
Memory size = 8 MBytes 4194304 x 16 64 MBits
Cancel K Back Next > Finish

EP2C8 基板は4ビットのLED があります。LED を制御するポートを Nios II システムに



組み込みます。左側の「PIO(Parallel I/O)」を選択し、ダブル・クリックして LED コント ローラを添加します。

🤷 PIO (Parallel I/O) – pio_0	
PIO (Parallel I/O)	About Documentation
Parameter Settings	
Width	
Width (1-32 bits) : 4	
Direction	
O Bidirectional (tristate) ports	
O Input ports only	
O Both input and output ports	
 Output ports only 	
Output Port Reset Value Reset Value: 0x0	
Cance	I < Back Next > Finish

ビット幅に4を入力してください。「Output ports only」を選択します。

ホストと会話するため、JTAG-UART も組み込むことが必要です。左側の「JTAG_UART」 を選択し、ダブル・クリックして JTAG-UART を添加します。



🛯 Altera SOPC Builder – nios_sdram.sopc* (C:¥altera¥nios2_sdram_led¥nios_sdram.sopc)							
<u>F</u> ile <u>E</u> dit <u>M</u> odule <u>S</u> ystem <u>V</u> iew <u>T</u> i	ools Nios I	II <u>H</u> elp					
System Contents System Generation							
Attera SOPC Builder Nios Il Processor Bridges and Adapters Interface Protocols BrASI Ethernet Ethernet	Target Device F	Family: Cyclone II	Clock Settings Name clk	Source External	MHz 50.0	Add Remove	
I III PCI I	Use C	Con Module Name	Description	Clock	Base	End	
Serial Avalon-ST JTAG I Avalon-ST JTAG I Avalon-ST JTAG I Avalon-ST JTAG I JTAG UART SPI (3 Wire Serial) UART (RS-232 Se He-Legacy Components Memories and Memory Control Peripherals		Construction_master data_master instruction_maste data_master itag_debug_modu s1 pio_0 s1	Nios II Processor Avalon Memory Mapp Avalon Memory Mapp SDRAM Controller Avalon Memory Mapp PIO (Parallel I/O) Avalon Memory Mapp PIO (Parallel I/O)	ed Master clk ed Master ed Slave clk ed Slave Avalon Memory M ed Slave clk	IBQ I 0x00000800 ier flapped Master [avalon_m 0x00000000	0 0×000000fff haster 8.1] ff	
		avalon_itag_slave	Avalon Memory Mapp	ed Slave clk	₽ 0x0000010	0x00000017	
	<					>	
New Edit Add		Remove Edit	Move Up	Move Down Addre	ss <u>M</u> ap <u>F</u> ilter.		
To Do: cpu_0: No reset vector has been specified for this CPU. Please parameterize the CPU to resolve this issue To Do: cpu_0: No exception vector has been specified for this CPU. Please parameterize the CPU to resolve this issue Exit Help							
		Exit Help	Prev Next	Generate			



🖷 JTAG UA	RT – jtag_uart_0	
MegaCoro	JTAG UART	About Documentation
Parameter Settings		
Configuration] > Simulation >	
Write FIFO (Buffer dept	(Data from Avalon to JTAG) h (bytes): 64	IRQ threshold: 8
Constru	ct using registers instead of memory blocks	
Read FIFO	(Data from JTAG to Avalon)	
Buffer dept	h (bytes): 64	IRQ threshold: 8
Constru	ct using registers instead of memory blocks	
		Cancel < Back <u>N</u> ext > <u>F</u> inish

デフォルトの設定のまま進んでください。

EP2C8 基板は一つの 50MHz の水晶発振器しかありません。ほかの周波数又は位相クロッ クのため、PLL を Nios II システムに組み込みます。左側の「PLL」を選択し、ダブル・ク リックして PLL を添加します。



🖷 PLL – pll_	0		X
MegaCore'	PLL		About Documentation
Parameter Settings			
PLL Settings	> Interface >		
PLL configura	ition		
The Avalon PLL ALTPLL megafu When you finisi on the SOPC Bu Launch Atte	configuration wizard creates a unction. Use the ALTPLL MegaW h configuring the PLL, the PLL cl uilder System Contents tab.	i component wrapper lizard to configure the ock output appears in	around an Altera e PLL settings. n the clock table
 Info: Selecte Info: Input cl 	ed device family is different from lock frequency conflicts with the	the PLL parameteriza PLL input frequency	ation. Please reparameterize the settings. Please change the inpr
<			
		Cance	I < Back Next > Einish

[Launch Altera's ALTPLL MegaWizard]ボタンを押して、PLL を設定します。





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

ALTPLL	
Transfer	About Documentation
1 Parameter 2 Output 3 EDA Clocks	4] Summary
General/Modes $ ightarrow$ Inputs/Lock $ ightarrow$ Clock swi	tchover
altplipil_0	Currently selected <u>d</u> evice family: Cydone II Match project/default
inclk0 inclk0 frequency: 50.000 MHz Operation Mode: Normal	Able to implement the requested PLL
Clk Ratio Ph (dg) DC (%) c0 1/1 0.00 50.00	General
Cyclone	Which device speed grade will you be using?
	Use military temperature range devices only
	What is the frequency of the inclock0 input? 50.00 MHz V
	Set up PLL in LVDS mode Data rate: 300.000 Mbps
	PLL type
	Which PLL type will you be using?
	○ Fast P∐.
	○ Enhanced PLL
	Select the PLL type automatically
	Operation mode
	How will the PLL outputs be generated?
	Ouse the feedback path inside the PLL
	In Normal Mode
	In Source-Synchronous Compensation Mode
	O In Zero Delay Buffer Mode
	<u>when to compensation</u> <u>verter no compensation</u> <u>verter no compensation</u> (External Feedback Mode)
 」P □ ● ▲ A般 😂 🏈 / 🗵 🛛 🗍	Which output clock will be compensated for?
	Cancel < Back Next > Einish

入力クロックは 50MHz を入力します。「Next」を押します。



🍰 ALTPLL				
Total			About	Documentation
Parameter 2 Output 3 EDA Clocks	Summary			
General/Modes Inputs/Lock Clock swit	chover			
General/Modes Inputs/Lock Clock swit altplipil_0 incik0 frequency: 50.000 MHz Operation Mode: Normal Cik Ratio Ph (dg) DC (%) c0 1/1 0.00 50.00	Able to implement the requested PLL Optional inputs Co Co Co Co Co Co Co Co Coeate an 'pllena' input to selectively enable th Coeate an 'pllena' input to selectively enable th Coeate an 'pfdena'	ne PLL et the PLL the phase/f	req. detecto only s unters are n	or not supported
2 JP m 🌒 A 般 🧐 🌮 🧷 🕺 👔 Kan		Cancel	< Back	Next > Finish

そのまま「Next」を押します。



ALTPLL			
T		<u>A</u> bout	<u>D</u> ocumentation
1 Parameter 2 Output 3 EDA 4	Summary		
General/Modes Inputs/Lock Clock switch	hover		
CEREB/MODES INDUS/COC LOOS SINCE altplipII_0 incik0 frequency: 50.000 MHz Operation Mode: Normal Cik Ratio Ph (dg) DC (%) c0 1/1 0.00 50.00 Cyclone I	Able to implement the requested PLL Clock switchover Clock switchover Clock switchover Clock switchover Create a 'inclk1' input for a second input clock What is the frequency of the indock1 input? Create a 'diswitch' input to manually select between (The diswitch input will behave as an input clock sele (Allow PLL to automatically control the switching betw (The diswitch input will behave as a manual override Input clock switch Perform input clock switch when the input clocd Perform the input clock switchover after Create a 'glkswitch' input to indicate the input (0 indk0 is being used/1 indk1 is being used) Create a 'clkbad' output for each input clock (0 input clock is toggling/1 input clock is not togg	100.00 the input doc control input goes bad between input input dock it dock being u k switchover is ling)	MHz v dks input) ks) it docks cydes used s initiated) Eisich
	Cancel		Mext > Finish

そのまま「Next」を押します。





🏂 ALTPLL			
and the second sec		About	<u>D</u> ocumentation
1 Parameter 2 Output 3 EDA			
dk c0 > dk c1 > dk c2 >			
	c0 - Core/External Output Clock		
altpllpll_0	Able to implement the requested PLL		
inclk0 inclk0 frequency: 50.000 MHz	<u>co</u> <u>Use this dock</u>		
Operation Mode: Normal	Clock Tap Settings	Requested settings	Actual settings
Clk Ratio Ph (dg) DC (%) c0 1/1 0.00 50.00	Enter output clock frequency:	100.0000000 MHz V	50.000000
	Enter output dock parameters:		
Cyclone	Clock multiplication factor	1	1
	Clock division factor	1 <<< Copy	1
		v	
	Clock p <u>h</u> ase shift	0.00 🔶 ps 🗸	0.00
	Clock duty cycle (%)	50.00	50.00
	More Details >>		
		Per Clock Feasibility	Indicators
		c0 c1 c2	
IP	20 21 CAPS -		
	C R KANA +	Cancel C Back	lext > Einish
			Text - Emisu

そのまま「Next」を押します。





沟 ALTPLL	-	
TODA -		About Documentation
1 Parameter 2 Output 3 EDA 4		
dk c0 > dk c1 > dk c2 >		
	c1 - Core/External Output Clock	
altplipII 0	Able to implement the requested PLL	
inclk0 inclk0 frequency: 50.000 MHz	<u>c0</u> <u>U</u> se this dock	
Operation Mode: Normal	Clock Tap Settings	A 1 1 10
Clk Ratio Ph (dg) DC (%)	Requested settings	Actual settings
c1 1/1 -63.00 50.00	C Enter output clock frequency: 100.0000000 MHz	S0.00000
	Enter output dock parameters:	
Cyclone	Clock multiplication factor	< Copy
	Clock <u>d</u> ivision factor	
	Clock phase shift -63.00 🚔 deg	-63.00
	Clock d <u>u</u> ty cycle (%) 50.00	50.00
	More Details >>	
	-Per Clock	k Feasibility Indicators
	c0	c1 c2
: JP 🖮 🌒 A 般 😂 🥔 🥕		
	Cancel	Back Next > Finish

新しい出力クロックを添加します。クロックの位相は-63 に設定してください。このクロックは SDRAM 用の制御クロックです。「Next」を押します。



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

🏂 ALTPLL					
Contract of				<u>A</u> bout	<u>D</u> ocumentation
1 Parameter 2 Output 3 EDA 4 Settings Clocks	Summary				
altpllpll_0 inclk0 Operation Mode: Normal Clk Ratio Ph (dg) DC (%) c0 1/1 0.00 50.00 c1 1/1 -83.00 50.00	CQ c1 C1 C1 C1 C1 C1 C1 C1 C1 C1	n on the files you wish to genera omatically generated, and a red sh to generate the selected files sequent MegaWizard Plug-In Ma MegaWizard Plug-In Manager or ctory: altera¥nios2_sdram_led¥	ite. A gray chi chedkmark ind . The state of nager session reates the sele	eckmark indic dicates an op each checkb is. ected files in	ates a file that is tional file. Click ox is maintained in the following
	Fi	le	Description	n	
		(altplipII_0.v (altplipII_0.ppf) altplipII_0.mp (altplipII_0.cmp (altplipII_0.bsf) altplipII_0_inst.v (altplipII_0_bb.v (altplipII_0_waveforms.html ialtplipII_0_wave*.jpg	Variation fi PinPlanne AHDL Inci VHDL con Quartus II Instantiatic Verilog HE Sample w. Sample w.	ile r ports PPF fi nponent decl symbol file on template fi)L black-box aveforms in s aveform file(s	le aration file le file ummary)
🐨 日版 📚 🥓 🧶 😕 😫 кана 🗸			Cancel	< <u>B</u> ack	Next > Einish

最後の確認です。「Finish」を押します。

Sysid モジュールは、Altera 社のソフトウェア開発環境 IDE を用いてソフトウェアをダウ ンロードする際に、ハードウェアとの整合性の確認に利用する ID 情報を格納するモジュー ルです。ここで読み出せる値は、SOPC Builder でロジックを生成するときに与えられるも のです。左側の「System ID Peripheral」を選択し、ダブル・クリックして Sysid を添加 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

😃 Altera SOPC Builder – nios_sdra	am.sopc* (C:¥altera¥nios2_sdram	_led¥nios_sdram.sop) (ac)		[
<u>Eile Edit M</u> odule <u>S</u> ystem <u>V</u> iew <u>T</u> ools	Nios II Help					
System Contents System Generation						
Altera SOPC Builder	Target	Clock Settings				
Nios Il Processor	Device Family: Cyclone II	Name	Source	MHz		Add
Bridges and Adapters		clk	External			Add
⊡nterface Protocols		pll_0_c0	pll_0.c0	50.0		emove
H-Memoriae and Memory Controllere		pll_0_c1 pll_0.c1		50.0		
□-Peripherals						
Debug and Performance	Use Con Module Name	Description		Clock	Base	En
···· Avalon-ST Test Pattern	itag debug module	Avalon Memory Mar	ped Slave		0x00000800	0x0(🔨
····· Avalor CT Test Pattern	Sdram_0	SDRAM Controller				
Performance Counter System ID Peripheral		Avalon Memory Map	ped Slave	cik	e 0x0200000	0x02
. Display	I I pio_0	PIO (Parallel I/O)				
FPGA Peripherals	→ s1	Avalon Memory Map	oped Slave	cik	. 0x0000000	UXUL
Microcontroller Peripherals	avalon itag slave	Avalon Memory Max	ped Slave	cik	e 0x0000010	0x0(
Interval Timer		PLL				
PIO (Parallel I/O)	→ s1	Avalon Memory Mag	oped Slave	cik	e 0x0000020	0x0(
	Sysid_0	System ID Periphera	al			
	control_slave	Avalon Memory Mag	ped Slave	pll_0_c0	. 0x0000018	
New Edit Add	Remove Edit	▲ Move Up	love <u>D</u> own	Address Map.	<u>F</u> ilter	
	Exit Help I Pre	V Next	Generate			
u System ID Peripher	ral – sysid_0				X	
System ID: 1181291759 Time stamp: 1231633784 A unique ID is assigned ever	m ID Periphe	e ral	About	<u>D</u> ocumen	tation	
🕐 Warning: The System ID	component must be named	"sysid" to be co	mpatible with	the Nios I	drivers a	
				Cancel	<u>F</u> inish	

[Finish]を押して、Sysid を生成します。

これまで、必要なモジュールをすべて組み込みました。



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

Use	Con	Module Name	Description	Clock	Base	End	IRQ
	G	cpu instruction_master data_master itag.debug.module	Nios II Processor Avalon Memory Mapped Master Avalon Memory Mapped Master Avalon Memory Mapped Slave	pll_c0	IRQ (■ 0×00000800) IRQ 3: 0×00000fff	ا
✓	Ļ	sdram sled_pio	SDRAM Controller Avaion Memory Mapped Slave PIO (Parallel I/O)	pli_c0	■ 0×02000000	0×027fffff	
		s1 □ j tag_uart avalon_jtag_slave	Avalon Memory Mapped Slave JTAG UART Avalon Memory Mapped Slave	pll_c0 pll_c0	 0x0000000 0x00000010 	0x000000017	
		sysid control_slave pll	System ID Peripheral Avalon Memory Mapped Slave PLL	pll_c0	0x0000018 0x00000018	0x0000001f	

モジュールの名前はわかりやすい名前に変更します。モジュールの名前欄で右クリックし

て、出てきたメニューから「Rename」を選択します。



モジュール用のクロックを選択します。[Clock]欄でダブル・クリックして、PLLのクロック pll_c0 を選択します。PLL モジュールは外部クロック「clk」を使用します。

Use	Con	Module Name	Description	Clock	Base	End	IRQ
V		🗆 cpu	Nios II Processor				
	\sim	Instruction_master	Avalon Memory Mapped Master	💙 00 lla			
	\neg	data_master	Avalon Memory Mapped Master	clk	IRQ O	IRQ 31	\leftarrow
		jtag_debug_module	Avalon Memory Mapped Slave	pll_c0	Ox0000800 Solution Solution	0x00000fff	
~		🗆 sdram	SDRAM Controller	pli c1			
		s1	Avalon Memory Mapped Slave	pll_c0	💣 0x0200000	0x027fffff	
 Image: A set of the set of the		🗆 led_pio	PIO (Parallel I/O)				
	≻→	s1	Avalon Memory Mapped Slave	pll_c0		0x0000000f	
V		⊟ jtag_uart	JTAG UART				
	≻→	avalon_jtag_slave	Avalon Memory Mapped Slave	pll_c0	e 0x0000010	0x00000017	⊳−0
 Image: A set of the set of the		🗆 sysid	System ID Peripheral				
	≻→	control_slave	Avalon Memory Mapped Slave	pll_c0	e 0x0000018	0x0000001f	
V		🗆 pli	PLL				
	$ \rightarrow $	s1	Avalon Memory Mapped Slave	cik		0×0000005f	


もう一つの設定は CPU のリセットと実行アドレスです。モジュール CPU の名前欄でダブ ル・クリックして、CPU 設定のダイアログを再び開きます。

😃 Nios II Processo	r – epu				
Nios MogoCore	II Processor				About Documentation
Parameter Settings					
Core Nios II Cach	ies and Memory Interfaces $>$	Advanced Features 🔪 M	MU and MPU Settings $>$	JTAG Debug Module	Custom Instructions
Core Nios II					
Select a Nios II core:					
	○Nios II/e	○Nios II/s	●Nios II/f		
Nios II Selector Guide Family: Cyclone II f _{system:} 50.0 MHz cpuid: 0	RISC 32-bit	RISC 32-bit Instruction Cache Branch Prediction Hardware Multiply Hardware Divide	RISC 32-bit Instruction Cache Branch Prediction Hardware Multiply Hardware Divide Barrel Shifter Data Cache Dynamic Branch Predic	ction	
Performance at 50.0 MH	z Up to 5 DMIPS	Up to 25 DMIPS	Up to 51 DMIPS		
Logic Usage	600-700 LEs	1200-1400 LEs	1400-1800 LEs		
Memory Usage	Two M4Ks (or equiv.)	Two M4Ks + cache	Three M4Ks + cache		
Hardware Multiply: Emb	edded Multipliers				
Reset Vector: Men	nor:- sdram	Viset: 0x0	0x	02000000	
Exception Vector: Mem	or X sdram	✓ Offset: 0x20	0×0	2000020	
Include MMU Only include the MMU when using an operating system that explicitly supports an MMU Fast TLB Miss Exception Vector: Memory: Offset: 0x0					
Include MPU					
UP 🔒	■ 🌒 A般 😂 🥔 🧷 👳			Cancel	< Back Next > Finish

リセットと実行アドレスを SDRAM に設定します。「Finish」を押します。



株式会社日昇テクノロジー

低価格、高品質が不可能? 日昇テクノロジーなら可能にする

🔹 Altera SOPC Builder – nios_led.sopc (C:¥altera¥nios_led_sch¥nios_led.sopc)							
<u>F</u> ile <u>E</u> dit <u>M</u> odule <u>S</u> ystem <u>V</u> iew <u>⊺</u>	Ejle <u>E</u> dit <u>M</u> odule <u>S</u> ystem <u>V</u> iew <u>I</u> ools Nios II <u>H</u> elp						
System Contents System Generation							
Altera SOPC Builder Altera SOPC Builder Nos II Processor Bridges and Adapters -Interface Protocols -Legacy Components Memories and Memory Control Dentify the factors	Target Device Family: Cyclone II		Clock Settings Name Clk pll_c0 pll_c1	Source External pil.c0 pil.c1	5 5 5	MHz 60.0 50.0 50.0	Add Remove
⊕ Debug and Performance	Use Co	on Module Name	Description		Clock	Base	End
⊡ Display		🗆 сри	Nios II Proces	sor			<u> </u>
FPGA Peripherals Arron optimizer Peripherals	- C	instruction_master	r Avalon Memo	ry Mapped Master	00 lla	¥	
Interval Timer PIO (Parallel I/O) Huttiprocessor Coordinatic		data_master jtag_debug_modul staram	Avalon Memo Avalon Memo SDRAM Contr Avalon Memo	ry Mapped Master ry Mapped Slave oller ry Mapped Slave	pli c0	IRQ (■ 0×00000800 ■ 0×02000000	0x000001
		□ led_pio s1	PIO (Parallel I/ Avalon Memo) 7) 7y Mapped Slave	pll_c0	■ 0×00000000	0×00000(
		⇒ jtag_uart avalon_jtag_slave	JTAG UART Avalon Memo	ry Mapped Slave	pll_c0	■ 0×00000010	0×00000(
	<		joystem ib r er	iprior di	1		>
New Edit Add Remove Edit Move Up Move Down Address Map Eliter							
Info: No errors or warnings.							
	E	E <u>x</u> it Help	Prev Next	Generate			

全部の設定が完了しました。「Generate」ボタンを押して、Nios II システムを生成します。



Nios II システムを生成中です。



😃 Altera SOPC Builder – nios_sdram.sopc (C:¥altera¥nios2_sdram_led¥nios_sdram.sopc)		
<u>Eile E</u> dit <u>M</u> odule <u>S</u> ystem <u>V</u> iew <u>T</u> ools Nios II <u>H</u> elp		
System Contents System Generation		
 Ortions 		
System module logic will be created in Verilog		
Simulation Create project and the simulator files		
Simulation, create project simulator mes. Excitosimulator		
CNics II Tools		
NeelIDE		
Info: Peak virtual memory: 48 megabytes	<u>^</u>	
Info: Processing ended: Sun Jan 11 09:41:04 2009		
Info: Elapsed time: 00:00;01		
Info: Total CPU time (on all processors): 00:00:00		
# 2009.01.11 09:41:06 (*) Completed generation for system: nios_sdram.		
# 2009.01.11 09:41:06 (*) THE FOLLOWING SYSTEM ITEMS HAVE BEEN GENERATED:		
SOPC Builder database : C:/altera/hios2_sdram_led/hios_sdram.ptf		
System HDL Model : C:/altera/hios2_sdram_led/hios_sdram.v		
System Generation Script : C:/altera/hios2_sdram_led/hios_sdram_generation_script		
# 2009.01.11 09:41:06 (*) SUCCESS: SYSTEM GENERATION COMPLETED.		
Info: System generation was successful.	×	
Info: No errors or warnings.		
Exit Help Prev Next Generate		

生成完了すると、「Exit」を押して、Quartus II に戻します。

Quartus II で再びシンボルボタンを押すと、

🖏 Quartus II - C:/altera/nios2_sdram_led/nios2_sdr	ram_led - nios2_sdram_led - [nios2_sdra
🔁 <u>F</u> ile <u>E</u> dit <u>V</u> iew <u>P</u> roject <u>A</u> ssignments P <u>r</u> ocessing <u>T</u> i	ools <u>W</u> indow <u>H</u> elp
📙 🗅 😂 🖬 🕼 🛛 🕹 🖌 🖻 💼 🔛 🗠 🦳 nios2_sdrar	m_led 💽 🔀 🖉 🏈 🖉
Project Navigator 🛁 🔺 🗙	nios2_sdram_led.bdf
Entity	
l Cyclone II: EP2C8Q208C8	
🛄 🔹 nios2_sdra 🖓	:シンボルボタン:::::
	📲 🔍
] 🔲 🏘 💠
🛆 Hierarchy 🖹 Files 🗗 Pesign Units	
	- 4 🔻 🛛 : : : : : : : : : : : : : : : : : :
Tasks 🔺 🔺 🗙	. 😰 🛛
Flow: Compilation	





Libraries 欄に、Project 前の[+]記号をクリックすると、中身が表示されます。生成された Nios II システム「nios_sdram」が見えました!nios_sdram を選択して「OK」ボタンを押 します。マウス・カーソルに選択されたシンボルがくっついた状態になるので、シンボル を配置したい場所に移動して、マウスの左クリックを押します。配置できた点で ESC キー を押してカーソルを元に戻してください。



次の作業は第四章と同じです。Nios II のシンボルに INPUT/OUTPUT などの端子を配置します。端子の名前は分かりやすい名前に変更してください。

	aa adram	
The second secon	os_sulam	
	clk pll c0 out	
	and a set	
	reset_n pil_c1_out	
	out_port_from_the_led_pio[30]	
		······································
	zs addr from the sdram[11_0]	OUTPUT SDRAM_A[110]
	zs_ba_from_the_sdram[10]	
	ze cae n from the edram	OUTPUT SDRAM nCAS
	zs_cas_ii_iroiii_uic_suraiii	
	zs cke from the sdram	OUTPUT SDRAM_CKE
		OUTPUT SDRAM aCSI01 =
	zs_cs_n_trom_tne_soram	
	zs do to and from the sdram[150]	BIDIR SDRAM_D[150]
	zs_dqm_from_the_sdram[10]	
	ze rae n from the edram	OUTPUT SDRAM nRAS
	23_103_1_1011_010_301011	
	zs we n from the sdram	SDRAM_nWE
		·····
ins	t	
	-	

配置が終わったら、コンパイル、ピン・アサイン、再コンパイルを行います。Cyclone II に書き込むデータ*.sof を生成します。このファイルを Cyclone II 書き込みます。

※ EP2C8¥nios_led_sch というフォルダに、すでに完成しているプロジェクトー式があり ます。

第六章 NIOS IIのプログラムの設計

Nios II システムを構築しました。これから、Nios II システムのプログラムを開発します。

Windows の「スタート」→「すべてのプログラム」→「Altera」→「NIOS II EDS 8.1」 から NIOS II 8.1 IDE が起動します。



<u>File</u> <u>E</u> dit <u>N</u> avigate Search <u>P</u> roject Tools <u>R</u> un <u>Window H</u> elp	
New Alter Children N and Device the	
Open File,. 🚿 Nios II C/C++ Application	
Close Ctrl+W 😽 Nios II System Library	
Close All Ctrl+Shift+W 😽 Nios II User-Makefile C/C++ Application	
Save Ctrl+S Nios II C/C++ Library	
Save As	
Source File	
Revert Header File	
Maura Ger Class	
Rename F2 Other	
Refresh E5	
Con <u>v</u> ert Line Delimiters To	
🖻 Print Ctrl+P	

「File」→「New」→「Nios II C/C++ Application」を選択します。



株式会社日昇テクノロジー

低価格、高品質が不可能? 日昇テクノロジーなら可能にする

New Project	
Nios II C/C++ Application SOPC Builder system file must be spec	ified.
Name: board_diag_0 Specify Location Location: Citalterations_led_schtsof Select Target Hardware. SOPC Builder System PTF File: CPU: Select Project Template Bank Project Board Diagnostics Sevent Binary Hello Freestanding Hello World Hello World Small Memory Test Simple Socket Server Web Server	tware Prowse Prowse Perowse Perows
?	< <u>B</u> ack <u>N</u> ext > <u>F</u> inish Cancel

Altera 社は幾つの定番テンプレートを提供しています。これらテンプレートに基づいて、 プログラムを開発しやすいです。もちろん、ゼロ「Blank Project」から開発もできます。 今回は「Board Diagnostics」を選択します。一行のコードも入力することが必要ないです。

「Browse」ボタンを押して、



Select Target H	ardware.				? 🗙
ファイルの場所型:	🚞 nios_led_sch	~	G 🕻	D 📂 🚥	
最近使ったファイル で デスクトップ マイドキュメント	 sopc_builder db incremental_db nios_led_sim software nios_led_ptf 				
ער די					
マイ ネットワーク	ファイル名(N):	nios_led.ptf		~	開((0)
	ファイルの種類(工):	*.ptf		*	キャンセル

Nios II システムの*.ptf ファイルを選択します。



株式会社日昇テクノロジー

低価格、高品質が不可能? 日昇テクノロジーなら可能にする

🛃 New Project 🔀				
Nios II C/C++ Application Click Finish to create application with a default system library as C#altera¥nios_led_sch¥software¥board_diag_0				
Name: board_diag_0 Specify Location Location: C:¥altera¥nios_led_sch Select Target Hardware. SOPC Builder System PTF File:	¥software Browse C:¥altera¥nios_led_sch¥nios_led.ptf I Browse			
CPU:	cpu 💌			
Select Project Template Blank Project Board Diagnostics Count Binary Hello Freestanding Hello MicroC/OS-II Hello World Hello World Small Memory Test Simple Socket Server Web Server	Description Tests peripherals on the development boards Details Board Diagnostics tests the LCD, the LED(D0-D7), and the seven-segment display peripherals, and the JTAG UART. This software example runs on the following Nios II hardware designs: - Standard - Full Featured - Low Cost			
0	< Back Next > Einish Cancel			

「Finish」ボタンを押すと、プログラムとハードウェアに関連するライブラリを自動的に生成します。





次の手順は「3.5 NIOS II プロセッサーの初体験」で紹介いたしました。