

Cyclone II EP2C20 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

2010/09/30



copyright@2010



注意事項	4
第一章 CycloneⅡ/EP2C20Q240C8 ボードの概要	4
1.1 FPGAボードの概要	5
1.2 拡張ボードの概要	6
1.3 FPGAボードと拡張ボードの接続	7
1.4 サンプルソースについて1	.2
1.4.1 SOPC¥1	.3
1.4.2 FPGA¥1	.4
1.5 USB_URATについて1	.5
1.5.1 USB_URATのドライバのインストール1	5
1.5.2 USB_URATの通信設定1	7
第二章 開発ツールのインストール1	7
2.1 Quartus II Web Editionをインストールする1	.8
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2	25
第三章 CycloneⅡの初体験3	31
3.1 Quartus評価版にソースを読み込む	\$1
3.2 USB-Blasterをインストールする3	12
3.3 書き込むソフトウェアを起動する	5
3.4 FPGAのコンフィギュレーションデバイスに書き込む	57
3.5 ByteBlasterIIの使い方3	8
3.6 NIOS IIプロセッサーの初体験4	0



修正履歴

NO	バージョン	修正内容	修正日
1	Ver1.0	新規作成	2010/09/30

- ※ 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。



注意事項

1、利用してないピンは必ず"As inputs,tri-stated"に設定する(設定手順: Assignments->Device、"Device&Pin Options"表示画面で"Unused Pins"タブを選択し、 "As inputs,tri-stated"を選択する)。

"As outputs, driving ground"或いは"As outputs, driving an unspecified signal"に 設定すると Flash 或いは FPGA を壊す恐れがあります!

また、*.tcl ファイルでピン設定する場合、"set_global_assignment -name RESERVE_ALL_UNUSED_PINS "AS INPUT TRI-STATED""が必要。

2、ダウンロードケーブル差込或いは抜き出し操作をする場合、必ず電源を切ってから操 作しましょう。

3、その他の部品、LCD などを差込或いは抜き出し操作をする場合も電源を切ってから操作 しましょう。

4、長時間利用しなくて、保存する時、埃、湿気、静電気を防止する様にご注意ください。

5、出荷する前、正常に動作するのを検証済みですが、お客様の所で使う前に、下記手順 で動作確認ください:

(1)簡易テスト:出荷前に Flash にテストプログラムを書込み済みなので、電源入れるだけ で、Flash に保存されている Boot Copier プログラムが Flash 上のテストプログラムを SDRAM にコピーして実行します。なので、ボードが正常な場合、LED、7 SEG、LCD などが正常に動 く状態になるはずです。

(2)all_test_3.zip テストプログラム:基本的な SOPC システムで、各機能のテストプログ ラムです。

第一章 Cyclone Ⅱ/EP2C20Q240C8 ボードの概要

FPGA ボードと拡張ボードで構成されている。

各種デバイスのリソース一覧:

	デバイス			
特徴	EP2C5	EP2C8	EP2C20	EP3C25
ロジック・エレメント数	4, 608	8, 256	18, 752	24, 624
RAM 総ビット数	119, 808	165, 888	239, 616	608, 256
エンベデッド乗計算器数	13	18	26	66
PLL 数	2	2	4	4
10 数	143	139	142	148



1.1 FPGAボードの概要



- (1) Cyclone II /EP2C20Q240C8 マイコン搭載
- (2) SDRAM : 8MByte
- (3) Flash: 8MByte
- (4) EPCS4(4Mb)のコンフィギュレーションデバイス搭載

(5) JTAG I/F、SOF ファイルをダウンロードする。直接 FPGA に書き込んで、速度は速いで すが、電源切れたらなくなる。デバッグする時に利用するのをお勧め。

(6) ASP I/F、POF ファイルをダウンロードする。コンフィギュレーションデバイス EPCS4 に書き込む。速度は JTAG より遅いですが、電源切れても保持する。最後のプログラム或い は電源を再起動が必要な場合利用する。※書き込み終了したら、電源を切って、ケーブル を抜けてから、正常に次の操作が出来る。

(7) 二つの 50MHz 水晶発振器搭載

(8) 全ての I0、Avalon バス、コンフィグピンを 2.54mm の拡張ピンヘッダで引き出されて いる

- (9) 四つのユーザ LED
- (10) Reset キー
- (11) 5V 電源で給電、電源スイッチと電源指示 LED 付き
- (12) 外形寸法: 95×90(mm) ※突起物は除く
- (13) 回路図を提供しております
- (14) サンプルのソースコードを提供しております



1.2 拡張ボードの概要



- (1) 10M Ethernet インタフェース (RTL8019)
- (2) 8 ビット ADC (TLC549)
- (3) 4 チャネル 8 ビット DAC (TLC5620)
- (4) リアルタイムクロック (PCF8563)
- (5) RS232 (MAX3232)
- (6) VGA I/F
- (7) PS2 I/F、キーボード又はマウスを接続可
- (8) 1602 液晶 I/F
- (9) 128*64 LCD I/F
- (10) ブザー
- (11) USB ホースと (CP2102)、USB1.1 と USB2.0 両方サポート
- (12) 7 セグメント LED x 8
- (13) EEPROM (I2Cモード、AT24C8)
- (14) 四つのユーザーキー



- (15) 外形寸法: 100×110(mm) ※突起物は除く
- (16) 回路図を提供しております

側面イメージ:



1.3 FPGAボードと拡張ボードの接続 イメージ:





ネーム	FPGA ピン番号	TO タイプ	機能
SVS CLK1	01	I	クロック入力1
SVS_CLK2	51	T	
SVS pDST	0.2	I	FDCA IL TAN L
515_11851	92	1	FFGA J L J F
	010		
LEDO	216	0	
LED1	6	0	四つのLED
LED2	7	0	
LED3	8	0	
SD_DATA0	38	I/0	
SD_DATA1	37	I/0	
SD_DATA2	41	I/0	
SD_DATA3	39	I/0	
SD_DATA4	44	I/0	
SD_DATA5	42	I/0	
SD_DATA6	47	I/0	SDRAM データバス
SD_DATA7	46	I/0	
SD_DATA8	68	I/0	
SD_DATA9	67	I/0	
SD_DATA10	66	I/0	
SD_DATA11	65	I/0	
SD_DATA12	64	I/0	
SD_DATA13	56	I/0	
SD_DATA14	57	I/0	
SD_DATA15	58	I/0	
SD_ADDR0	96	0	SDRAM アドレスバス
SD_ADDR1	105	0	
SD_ADDR2	100	0	
SD_ADDR3	106	0	
SD_ADDR4	88	0	
SD_ADDR5	86	0	
SD_ADDR6	87	0	
SD_ADDR7	84	0	

EP2C20 FPGA ボードのピンリスト:



株式会社日昇テクノロジー

SD_ADDR8	80	0	
SD_ADDR9	79	0	
SD_ADDR10	97	0	
SD_ADDR11	78	0	
SD_BA0	54	0	SDDAM O DANK 7 KLZ
SD_BA1	90	0	SDRAM V) DAINK / F V A
SD_DQML	50	0	データファク
SD_DQMH	70	0	
SD_CS	55	0	チップセレクト
SD_RAS	51	0	行アドレスセレクト
SD_CAS	52	0	列アドレスセレクト
SD_WE	49	0	書込みイネーブル
SD_CKE	73	0	クロックイネーブル
SD_CLK	72	0	クロックインプット
FLASH_DQ0	214	I/0	
FLASH_DQ1	203	I/0	
FLASH_DQ2	200	I/0	FLASH アータバス(拡
FLASH_DQ3	199	I/0	張ホードのイーサイ
FLASH_DQ4	197	I/0	
FLASH_DQ5	195	I/0	LUD12864、LUD1602 と
FLASH_DQ6	194	I/0	共用)
FLASH_DQ7	192	I/0	
FLASH_ADDR0	208	0	FLASH アドレスバス
FLASH_ADDR1	231	0	(拡張ボードのイーサ
FLASH_ADDR2	230	0	ネットチップと共用)
FLASH_ADDR3	228	0	
FLASH_ADDR4	226	0	
FLASH_ADDR5	223	0	
FLASH_ADDR6	222	0	
FLASH_ADDR7	218	0	
FLASH_ADDR8	233	0	
FLASH_ADDR9	232	0	
FLASH_ADDR10	235	0	
FLASH_ADDR11	234	0	



株式会社日昇テクノロジー



FLASH_ADDR12	236	0	
FLASH_ADDR13	21	0	
FLASH_ADDR14	238	0	
FLASH_ADDR15	20	0	
FLASH_ADDR16	18	0	
FLASH_ADDR17	16	0	
FLASH_ADDR18	15	0	
FLASH_ADDR19	14	0	
FLASH_ADDR20	13	0	
FLASH_ADDR21	11	0	
FLASH_ADDR22	9	0	
FLASH_CE	237	0	チップセレクト
FLASH_WE	189	0	書込み信号(拡張ボー
			ドのイーサネットチ
			ップと共用)
FLASH_OE	191	0	読み出し信号(拡張ボ
			ードのイーサネット
			チップと共用)

拡張ボードのピンリスト:

ネーム	FPGA ピン番号	IO タイプ	機能
KEY1	110	Ι	
KEY2	111	Ι	回つのたー
КЕҮЗ	113	Ι	
KEY4	114	Ι	
78LED_COMO	161	0	
78LED_COM1	159	0	
78LED_COM2	164	0	
78LED_COM3	162	0	7SEC LED 壯语ポート
78LED_COM4	166	0	TOEG LED 英迪尔兰下
78LED_COM5	165	0	
78LED_COM6	168	0	
78LED_COM7	167	0	
78LED_DATA0	156	I/0	78LED_Da



株式会社日昇テクノロジー

78LED_DATA1	150	I/0	78LED_Db
78LED_DATA2	140	I/0	78LED_Dc
78LED_DATA3	141	I/0	78LED_Dd
78LED_DATA4	155	I/0	78LED_De
78LED_DATA5	157	I/0	78LED_Df
78LED_DATA6	139	I/0	78LED_Dg
78LED_DATA7	149	I/0	78LED_Dh
KB_DATA	125	Ι	PS2 データライン
KB_CLK	109	Ι	PS2 クロックライン
I2C_SCL	173	0	I2C クロックライン
I2C_SDA	174	I/0	I2C データライン
LCD12864_CS1	184	0	LCD12864 左右チップ
LCD12864_CS2	185	0	セレクタ信号
LCD12864_E	186	0	LCD12864 イネーブル
LCD1602_E	187	0	LCD1602イネーブル
RTL8019_CS	175	0	イーサネットチップ
			セレクタ信号
RTL8019_INT	177	Ι	イーサネット割り込
			み信号
RTL8019_RST	188	0	イーサネットリセッ
			卜信号
RXD_URAT	170	Ι	DC000
TXD_URAT	171	0	K5232
RXD_USB	130	Ι	
TXD_USB	131	0	USB UKAI 1/F
VGA_HS	132	0	VGA I/F
VGA_VS	134	0	
VGA_R	135	0	
			l.



株式会社日昇テクノロジー

VGA_G	136	0	
VGA_B	137	0	
AD_CLK	126	0	
AD_CS	128	0	ADC(TLC549) I/F
AD_DATA	127	Ι	
TLC5620_CLK	118	0	
TLC5620_DATA	119	0	DAC(TLCE620) = I/E
TLC5620_LDAC	117	0	DAC (110020) 1/F
TLC5620_LOAD	116	0	

1.4 サンプルソースについて

Example_ep2c20.zip に下記サンプルソースが含まれている。





🖃 🚞 Example_ep2c20 🖃 🚞 FPGA 🖃 🚞 1.LED 표 🚞 1key1LED 표 🚞 4key4LED 표 🚞 VoltingMachine 표 🚞 WaterLamp 🖃 🚞 2.7SegLED 표 🚞 DynamicDisplay 표 🛅 PlusMinusDisplay 표 🚞 StaticDisplay 😑 🚞 3.FreqCounter 표 🚞 verilogHDL 🗉 🚞 VHDL 🖃 🚞 4.music 표 🚞 verilogHDL_simpleSound 표 🛅 VHDL_simpleSound 🖃 🚞 5 IIC 표 🚞 verilogHDL_IIC 🗉 🚞 VHDL_IIC 🖃 🚞 6.AD 표 🚞 verilogHDL_AD 🗉 🚞 VHDL_AD 🖃 🚞 7.DA 🗉 🚞 VHDL_DA 표 🚞 vreilogHDL_DA 🖃 🚞 8.LCD1602 🗄 🚞 VHDL_LCD1602 표 🛅 vreilogHDL_LCD1602 😑 🚞 9.LCD12864 표 🚞 LCD12864_v 🖃 🚞 10.UART 표 🚞 verilogHDL_uart 🗄 🚞 VHDL_uart 🖃 🚞 11.VGA 표 🚞 verilogHDL_VGA_basic 표 🚞 VGA_test 🗉 🛅 VHDL_VGA_basic 🖃 🚞 12.PS2 표 🚞 verilogHDL_PS2 🗉 🦳 VHDL PS2

1.4.1 SOPC¥

all_test_3

基本的な SOPC システムで、software フォルダに各種機能のテストプログラムが含まれている。IDE に導入してボードを確認できる。

現象:電源いれたらブザーはピピと鳴る、7 Seg 部はカウンタ表示する。LCD1602、LCD12864 接続していれば文字が表示する。COM と USB URAT からも文字列が出力する。





1.4.2 FPGA¥

```
1.LED¥1key1LED
```

キー操作で LED を制御するテスト

 $1.\,\texttt{LED} \$4 \texttt{key} \texttt{4} \texttt{LED}$

キー操作で LED を制御するテスト

1.LED¥VoltingMachine

投票機を模擬するテスト

1.LED¥WaterLamp

ウォーターLED 表現するテスト

2.7SegLED¥DynamicDisplay

7SEGMENT 動的スキャン表示テスト

2.7SegLED¥PlusMinusDisplay

7SEGMENT プラスカウンターとマイナスカウンター表示テスト

2.7SegLED¥StaticDisplay

7SEGMENT 静的スキャン表示テスト

3.FreqCounter¥verilogHDL

verilogHDL 言語で周波数カウンター実現するテスト

3.FreqCounter¥VHDL

VHDL 言語で周波数カウンター実現するテスト

- 4. music¥verilogHDL_simpleSound
- 4. music¥VHDL_simpleSound

ブーザテスト

- 5. IIC¥verilogHDL_IIC
- 5. IIC¥VHDL_IIC

I2C テスト

- 6.AD¥verilogHDL_AD
- 6. AD¥VHDL AD

AC 変換テスト

- 7. DA¥VHDL_DA
- 7.DA¥vreilogHDL_DA

DA 変換テスト

8. LCD1602¥VHDL_LCD1602

LCD1602 液晶テスト

- 8.LCD1602¥vreilogHDL_LCD1602
 - LCD1602 液晶テスト
- 9. LCD12864¥LCD12864_v





lcd12864 液晶テスト 10. UART¥verilogHDL_uart 10. UART¥VHDL_uart シリアル通信受信テスト 11. VGA¥verilogHDL_VGA_basic VGA カラースクリーン表示 11. VGA¥VGA_test 11. VGA¥VHDL_VGA_basic VGA カラースクリーン表示 12. PS2¥verilogHDL_PS2 12. PS2¥VHDL_PS2

PS2 キーボードテスト

1.5 USB_URATについて

本ボードは USB シリアル変換機能(CP2102 で実現)を内蔵していて、USB 関係のデバッグ と開発が便利になっている。弊社 HP からドライバをダウンロードしてインストールくださ い。ドライバをインストールした後、PC 側で仮 Com が一つ増える。この Com で通信が行う。

株式会社日昇テクノロジー

1.5.1 USB_URATのドライバのインストール

USB ケーブルで PC とボードを接続する(拡張ボード上の B type I/F)。

新しいハードウェアの検出ウィザード				
	新しいハードウェアの検索ウィザードの開始			
	このウィザードでは、次のハードウェアに必要なソフトウェアをインストールします: CP2102 USB to UART Bridge Controller			
- Maria	○ ハードウェアに付属のインストール CD またはフロッピー ディ スクがある場合は、挿入してください。			
	インストール方法を選んでください。			
	 ○ ソフトウェアを自動的にインストールする (推奨)④ ④ 一覧または特定の場所からインストールする (詳細)(S) 			
	続行するには、D欠へ] をクリックしてください。			
	< 戻る(四) 次へ(型) > 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):
< 戻る(B) 次へ(N) > キャンセル

完了すると、自動的に次の検出ウィザード画面が表示される。上記と同じ手順でインスト ールする。



完了して、デバイスマネージャ画面で次の二つのデバイスが増えている:



(CP2102は CP2101を基にバージョンアップしたもので、ドライバは共通)

⊟ ⊷Ҿ USB (Universal Serial Bus) ⊐ントローラ │ │ ⊷Ҿ CP2101 USB Composite Device □ - ℊ ポート (COM ≿ LPT)

[™] [™] CP210× USB to UART Bridge Controller (COM8)

1.5.2 USB_URATの通信設定

PC側のハイパーターミナル(ここはCOM8 (115200(B)、8(D)、なし(P)、1(S)、なし(F)))を 起動する。

出荷状態のボードに電源をいれると、ハイパーターミナル画面に通信データが表示される。

第二章 開発ツールのインストール

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 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、CPLD/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。本章には v8.1 でインストールの手順を説明します。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。







使用者の名前と所属会社名を入力するダイアログが表示されます。名前は半角のアルファ ベットで入力しましょう。







インストール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。

Quartus II 8.1 Web Edition	i Setup	×
Start Copying Files Review settings before co	opying files.	
	Setup has enough information to start copying the program files. If you wan to review or change any settings, click Back. If you are satisfied with the settings, click Next to begin copying files. Current Settings: User Name and Company: quartus; csun Destination Directory: C:¥altera¥01¥quartus Design Directory: C:¥altera¥01¥qdesigns Program Folder: Altera	t.
InstallShield	< Back Next > Cancel	

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。

Questio	n 🛛 🔀
2	Create Quartus II 8.1 Web Edition shortcut on desktop?
	(はい)() (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。

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。

	×
Getting Started Wit	th Quartus® II Software
	OUARTÚS II
	A REAL PROPERTY AND A REAL PROPERTY A REAL PRO
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	Open Interactive Tutorial
Open Existing Project	
Open Recent Project:	
ButtonCountAIIVHDL	
ButtonCountVHDL	
Web links:	
Literature Training	Online Demos Support
Don't show this screen again	
Quartus II File Edit View Project Assignments Processing Tools W	(indow Help
□ 🗳 🖬 💋 🍜 Å 🖻 🛍 🕫 🕫 Project Navigator	
Entity	
	Quartus II Software Timing esigner With Quartus II Integration Download With Quartus II Integration Software fet a Nins* II Software fet a Nins* II
	Subscription Edition Free trial>>> Processor Free! Free 30-day Trial Limited Time Offer
Tasks	
Flow: Compilation Task G' Time	QUARTUSII
Compile Design Analysis & Synthesis Firm Fitter (Place & Route)	Version 8.1
E► Assembler (Generate programming files) E► Timing Analysis	View New Quartus II
EDA Netlist Writer Program Device (Open Programmer)	Documentation
× Type Message	
System & Processing λ Extra Info λ Info λ Warning λ Critic	al Warning) Error), Suppressed), Flag /
Message:	Locate



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

📴 Nios II Embedd	led 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		



📅 Nios II Embed	ded Design Suite 8.1	
	Extracting nios2eds\setup.ini Extracting nios2eds\setup.exe Extracting nios2eds\layout.bin Extracting nios2eds\data1.cab Extracting nios2eds\data2.cab	
	Destination folder	
	C:\DOCUME~1\zqing\LOCALS~1\Temp Bro	o <u>w</u> se
	Installation progress	
	(*****	
	Install C.	ancel

先ず「Install」ボタンを押して解凍します。「Next」ボタンを押します。



ライセンスを同意すれば、「Yes」ボタンを押します。











インストール先フォルダを変更せず、そのまま進んでください。



「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。

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

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 Cyclone Ⅱの初体験

3.1 Quartus評価版にソースを読み込む

弊社の HP で Cyclone II 用のサンプルソース (Example_ep2c20.zip) をダウンロードできます。

ソース・ファイルを..¥alteraに展開します。その中に、幾つのサンプルがあります。具体的には 1.4 節をご参照ください。一つのサンプルを紹介します。

エクスプローラまたはマイ コンピュータを起動して、 ..¥all_test_3というフォルダを開いてください。

これらの中に、名前が all_test_3.qpf、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、all_test_3 というプロジェクト が開きます。





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。コンパイルは少し時間が掛か ります。

🖞 Quartus II – D:/embedded/altera/workspace/ep2c20/all_test_3/all_test_3 – all_test_3 – [Compilation Report – Flow Summary]			
Eile Edit View Project Assignments	Processing Tools Window Help		
D 🗳 🖬 🗿 🎒 👗 🖻 🖻 🗠 🗠	🔟 Stop Processing Ctrl+Shift+C 💈 🥙 🚳 👘 🕨 🤡 🏷 🍓 👘 🔛 😵 😓 🚳		
Project Navigator	Start Compilation Ctrl+L		
Entity	Analyze Current Eile Flow Summary		
Cyclone II: EP2C20Q240	Start •		
i ⇒ all_test_3 者	Update Memory Initialization File		
	Compilation Report Ctrl+R		
A Hierarchy Files B ^P Design Units Tasks Flow: Compilation Task g	Start Compilation and Simulation Ctrl+Shift+K Generate Functional Simulation Nettist Start Simulation Ctrl+I Simulation Debug Simulation Report Ctrl+Shift+R Compiler Tool Simulator Tool Simulator Tool PowerPlay Power Analyzer Tool Simulator Tool		
× Type Message			
System (3) \ Processing \ Extra Info \ Info (10) \ Warning (26) \ Critical Warning \ Error \ Suppressed (101) \ Flag \ Message:			
Starts a new compilation	™⇒ ∎ Ide		

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。 <u>http://www.csun.co.jp/SHOP/200901025.html</u>

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ド			
	新しいハートウェアの検索ウィサートの開始		
	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 (こ接続しますか?		
	○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、 D欠ヘ] をクリックしてください。		
	〈戻る(四) 次へ(11) > キャンセル		





「一覧または特定の場所からインストール」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): ○ Xaltera¥81¥quartus¥drivers¥ush=blaster
 ○検索しないで、インストールするドライバを選択する(D) → 暫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取りません。
< 戻る(B) 次へ(N) > キャンセル

USB-Blaster のドライバは¥altera¥81¥quartus¥drivers¥usb-blaster にあります。



インストール中。

新しいハードウェアの検出ウィザード			
	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、回路を書き込むソフト ウェア「Programmer ツール」が起動します。



🐇 Quartus II - D:/embedded/altera/workspace/ep2c20/all_test_3/all_test_3 - all_test_3			
<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>Tools Window H</u> elp		
D 📽 🖬 🗿 🎯 🕺 the 💼 🗠 🗠 🕅 all_te	Run E <u>D</u> A Simulation Tool Run <u>E</u> DA Timing Analysis Tool		
Entity Logic Cells Dedica	Launch EDA Simulation Library <u>C</u> ompiler		
Cyclone II: EP2C20Q240C8	Launch Design Space Explorer		
⊞	🚯 TimeQuest Timing Analyzer		
	<u>A</u> dvisors		
	🗞 Chip Planner (Floorplan and Chip Editor)		
Afterarchy Eriles A Design Units	Design Partition Planner		
Tasks	Netlist ⊻iewers ►		
	SignalTap II Logic A <u>n</u> alyzer	ARIUSII	
lask⊻ ✓ ⊡ ► Compile Design	- In-System Memory Content Editor		
✓ ⊕► Analysis & Synthesis	In-System Sources and Probes Editor	Information	
 ✓ I∄ ► Fitter (Place & Route) ✓ I∄ ► Assembler (Generate programming files) 	SignalProbe Pins	Desumentation	
	Programmer		
× Type Message	📉 Mega <u>W</u> izard Plug-In Manager		
М.	1 SOPC Builder		
	Tcl Scripts		
	Customi <u>z</u> e		
Silv System A Processing A Extra Info A Info A Warnin	Options	ag /	
δ Message: Location:	License Setup	Locate	
Open a Programmer window			



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 Setting: Select a programming hardware setu only to the current programmer wind Currently selected hardware:	s up to use when progra ow. USB-Blaster [USB	mming devices. This 3-0)	s programming 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、 EP2C8 又は EP3C25 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 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。

ブザーがピピと鳴って、FPGAボード上の LED が点滅して、拡張ボード上の 7 SegLED がカウ ンタ表示しているのを確認してください。

3.4 FPGAのコンフィギュレーションデバイスに書き込む

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 – I	D:/embedded/	altera/workspace/ep	2c20/all_test_	8/all_test_3	- all_test_3 - [all_test_3.cdf*]	
<u>File E</u> dit P <u>r</u> oces	ssing <u>T</u> ools <u>W</u>	(indow				
🔔 Hardware Setup	D USB-Blaster [I	USB-0]		$\boldsymbol{<}$	Mode: Active Serial Programming rogres	s: 0%
Enable real-time I	SP to allow backgr	ound programming (for MAX II	devices)			
🏓 Start	File	Device	Checksum	Usercode	Program/ Verify Blank- Configure Verify Check Examine	Security Bit Erase CLAMP
齢 Stop	all_test_3.pof	EPCS4	057897BD	00000000		
Auto Detect					\bigcirc	
X Delete	\		_		チェック	
	` 「St	art」ボタンを	押す		ナートフ	
Add File					を入れる	
👺 Change File						
🗳 Save File						
📽 Add Device						
🜓 Up						
🐌 Down						
For Help, press F1	_*					

「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 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、どの現象が出てきますか?

3.5 ByteBlasterIIの使い方

PC との接続:





Quartus IIの「Tools」メニューから「Programmer」を起動する。

下記の様に No Hardware の状態であれば、"Hardware Setup…"をクリックする。

🔔 Hardware Se Mode:	etup No Hardwa	re Vr			
Hardware Se	tup				×
Hardware Sett:	ings JTAG Setti	ngs			
Select a progra hardware setup	mming hardware set applies only to the (up to use whe current program	n programming de [.] mmer window.	vices. This progra	amming
Currently select	ed hardware:	No Hardware			•
Available har	dware items:				
Hardware		Server	Port	Add Ha	ardware
				Remove	Hardware
					Close

"Add Hardware…" をクリックして、 "ByteBlasterMV or ByteBlasterII" を選択する:



Add Hardware	2		
Hardware type: Port:	ByteBlasterMV or ByteBlaster II		
Baud rate:			
Server name:			
Server port:			
Server password:			
Auto Detect	OK Cancel		

OK をクリックする。

Hardware Setup			\sim
Hardware Settings JTAG Se Select a programming hardware hardware setup applies only to l	ttings setup to use when he current programm	programming device her window.	es. This programming
Currently selected hardware: Available hardware items: Hardware ByteBlasterII	No Hardware No Hardware ByteBlasterii [LP Server Local	11 Port LPT1	Add Hardware Remove Hardware
			Close

"Currently selected hardware:"から"ByteBlasterII[LPT1]"を選択して、Close をク リックする。

🔔 Hardware Setup	ByteBlasterII [LPT1]	
Mode:	JTAG	2

次の手順はUSB Blaster と共通です。

3.6 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、 ¥SOPC¥all_test_3¥ というフォルダを開いてください。



これらの中に、名前が all_test_3. qpf 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、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、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 Op	enCore Plus IP.
Time remaining:	unlimited
Cancel	

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 G/C++ - Nios II IDE	
Eine Eun Mantene Segren Froject tools Kun Window Telp	ta ta ta ta ta ta ta ta ta ta ta ta ta t
Welcome to the Altera Nios II IDE	A
	Workbench
	15 (RD) Y
	.
Overview Tulo	rials What's New
El 1 n	
BA	
eclipse	

NIOS II IDE の初起動の画面です。

NIOS II IDEの project window で右クリックし Import を選択します。



Nios II C/C++ - makefile - Nios II II		
File Edit Navigate Search Project Tools	s <u>R</u> un <u>w</u> indow <u>H</u> elp ☆ ▼	1 Nios II C/C++
Nios II C/C++ Projects X	Image: Second state of the second s	Outline X Ingets X Outline X Ingets
: □ ◆		

「Import」画面で「Existing Projects into Workspace」を選択して、「Next」ボタンを押 します。

📑 Import	
Select Create new projects from an archive file or directory.	2J
Select an import source:	
type filter text	
General Archive File Breakpoints Existing Projects into Workspace File System Preferences Altera Nios I C/C++ C/C++ Team	
	Cancel



🚮 Import			X
Import Projects Select a directory to searc	sh for existing Eclipse proje	cts.	
⊙ Select roo <u>t</u> directory:		(Browse
⊖Select <u>a</u> rchive file:			Browse
Projects:			
			Select All
			Deselect All
			Refresh
✓ <u>C</u> opy projects into wo	orkspace		
0	< <u>B</u> ack <u>N</u> ext >	<u> </u>	Cancel

「Browse」ボタンを押してプロジェクト保存しているフォルダを選択します。

フォ	オルダの参照	
5	Select root directory of the projects to import	
	🖃 🚞 software	^
	E D AD TEST	
	□ 🔁 all_test_v5	
	settings	
	🕀 🛅 Debug	
	⊞ 🛅 all_test_v5_syslib	
	⊞ 🛅 beep_test	
	⊞ 🛅 da_5620_test	
	🗉 🛅 DMA_TEST	~
		<u> </u>
:	フォルダ: all_test_v5	
	新しいフォルダの作成(M) OK キャンセル	

「OK」ボタンを押します。



E Import				
Import Projects Select a directory to search for existing Eclipse projects.				
 Select root directory: Select archive file: Projects: 	D:¥embedded¥altera¥workspace¥Example_e	Browse		
⊽ all_test_v5		Select All		
Copy projects into workspace				
0	< <u>B</u> ack <u>N</u> ext > <u>F</u> inish	Cancel		

そのまま、「Finish」ボタンを押します。

上記操作をもう一度繰り返して、all_test_v5_syslib プロジェクトを追加します。





フォ	rルダの参照 ? 🔀				
Select root directory of the projects to import					
	🗉 🛅 oc_i2c_master 🛛 🔼				
	🕀 🧰 ps2				
	🗄 🚞 seg78led				
	🖃 🚞 software				
	🗉 🛅 AD_TEST 📃				
	🕀 🛅 all_test_v5				
	🖃 🛅 all_test_v5_syslib				
	🛅 settings				
	🕀 🧰 Debug				
	🗆 🗁 boon toot				
フォルダ: all_test_v5_syslib					
	新しいフォルダの作成(M) OK キャンセル				

このままビルドすると旨く行かない場合がありますので、一回 Clean Project を実行します。

💽 Nios II C/C++ - Nios II IDE				
<u>F</u> ile <u>E</u> dit <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 📸 ▾ 🚱 ▾ 🚱 ▾ 🚱 ▾ 🎧 ▾ 🎧 ▾ 🎧 ▾ 🎧 ▾ 🎧 ▾ 🎦 🖓 i 🔁 ▾ 📰 💽 Nios II C/C++ i ½ - 🖓 - 🏷 🗘 ▾ ↔ -				
💽 Nios II C/C++ Projects 🙁 📃 🗖				
() 🗟 🗖 🤣 🔻		▽		
E-C all_test v5		An outline is not available.		
	_			
Go Into	_	Make S		
Open in <u>N</u> ew Window	_	A (+ -> 🔊 📐		
Rebuild Index		⊕ 😂 all_test_v5		
Active Build Configuration	•	⊕ 🧀 🚰 all_test_v5_syslib		
Run As	Problems 🖄 Console Properties Debug			
Debug As	errors, 0 warnings, 0 infos	-14		
<u>B</u> uild Project	Description 🔺	Resource Path		
Clean Project				
Сору				
Paste				

左側の「all_test_v5」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++ - Nios II IDE				
<u>File E</u> dit <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 📸 ▼ 🗳 ▼ 💣 ▼ 🎯 ▼ i ½ ▼ 🖓 ▼ 🎝 ▼ 🏷 ↔ ↔ ↔ ↔ ↔ ↔ ↔ ↔ ↔ ↔ ↔ ↔ ↔ ↔ ↔ ↔ ↔ ↔				
🖪 Nios II C/C++ Projects 🕱 🦳 🗖 🗍 🔚 🔛 🖓 🖓				
	_ &	~		
⊕ 😂 all_test v=		An outline is not available.		
i il_test_v New	F			
Go Into				
Open in <u>N</u> ew Wi	indow			
Rebuild Index		⊕ ∰ all test v5		
Active Bui <u>l</u> d Co	nfiguration 🕨	⊕ 😂 all_test_v5_syslib		
Run As	•			
Debug As	, Problems 🛛 Console Properties Deb	ue 🏦 🎝		
<u>B</u> uild Project	rrors, 0 warnings, 0 infos	1 - 1 - 1		
Clean Project	Description 🔺	Resource Path		
Сору				

終わりましたら、NIOS II IDEのメニュー「Run」→「Debug As」→「2 Nios II Hardware」 をクリックしてデバッグ開始する。

🛃 Nios II C/C++ - Nios II IDE 📃 🗌 🖂 🖂				
<u>F</u> ile <u>E</u> dit <u>N</u> avigate Se <u>a</u> rch <u>P</u> roject Tool	s <u>R</u> un <u>W</u> indow <u>H</u> elp			
: █ • ☐ ▷ 励 : @ • & • € • 0 : 2 - 7 - * ← ← - → -	Presume ■ Suspend			
💽 Nios II C/C++ Projects 🛛 📃 🗖	Step Into			
	⊜Step Qver _neStep Return	∽ An outline is not available.		
test_v5_systib [nios] te-∰ altera.components	→[Run to Line	● Make ※ □		
	Q ₆ <u>R</u> un Last Launched Ctrl+F11 [™] ₆ <u>D</u> ebug Last Launched F11			
	Run History Run As ►	all_test_v5_syslib		
	F Run	Debug 🔒 🚉 🛃 🖬 🖬 🖓		
	I Debug As	C 1 Debug Local C/C++ Application		
	Debug	2 Nios II Hardware		
] 💁 External Tools 🔹 🕨	🗑 3 Nios II ISS		
	Creating generated_app.sh			
	Post-processing to create flash.flash Herdware simulation is not enabled for the torget SONC			
	Builder system. Skipping creation of hardware simulation			
	model contents and simulation symbol files. (Note: This does 🗏			
	not affect the instruction set simulator.)			
	Post-processing to create onchip_mem.hex			
	parta compresea in iz.107 se			
 ^o /all_test_v5	r			