



不可能への挑戦

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

Cyclone II EP2C20 ボード

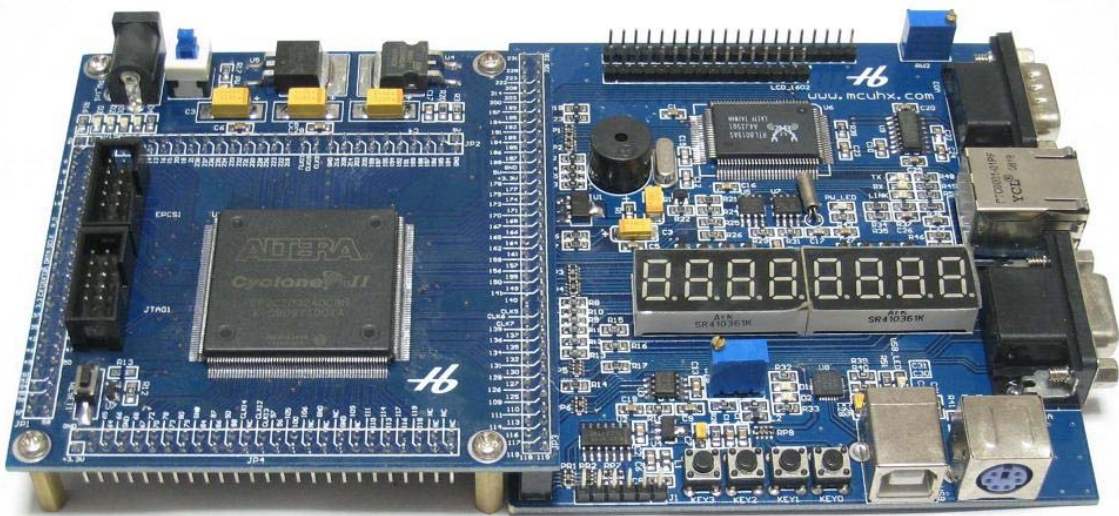
マニュアル

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](http://www.csun.co.jp)



注意事項	4
第一章 Cyclone II /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	12
1.4.1 SOPC¥	13
1.4.2 FPGA¥	14
1.5 USB_URATについて	15
1.5.1 USB_URATのドライバのインストール	15
1.5.2 USB_URATの通信設定	17
第二章 開発ツールのインストール	17
2.1 Quartus II Web Editionをインストールする	18
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	25
第三章 Cyclone II の初体験	31
3.1 Quartus評価版にソースを読み込む	31
3.2 USB-Blasterをインストールする	32
3.3 書き込むソフトウェアを起動する	35
3.4 FPGAのコンフィギュレーションデバイスに書き込む	37
3.5 ByteBlasterIIの使い方	38
3.6 NIOS IIプロセッサの初体験	40



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。

「<http://www.csun.co.jp>」

※ (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、利用していないピンは必ず“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。

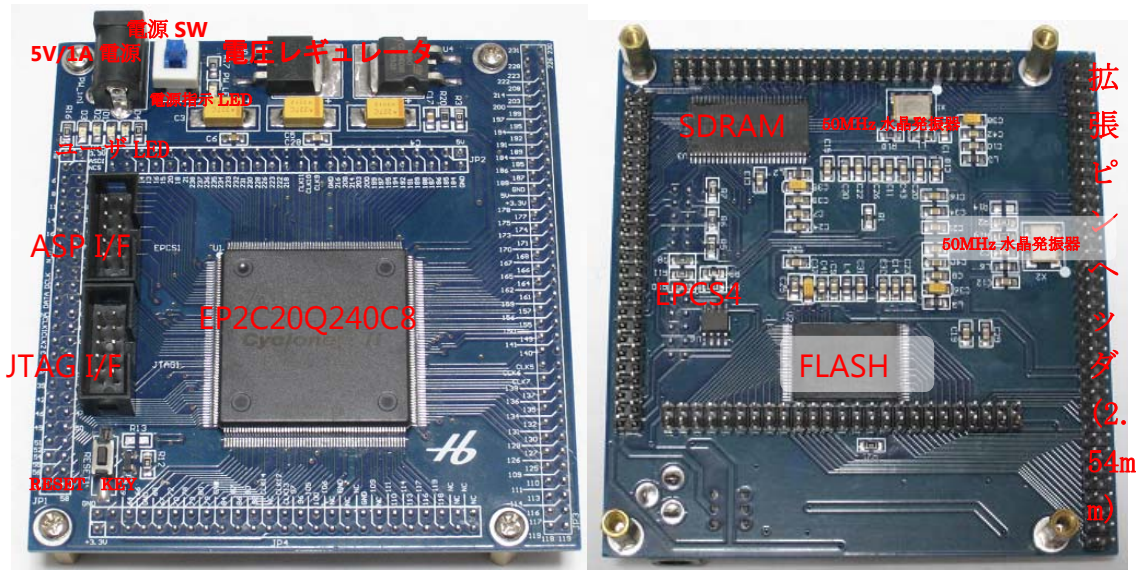
第一章 Cyclone II /EP2C20Q240C8 ボードの概要

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：

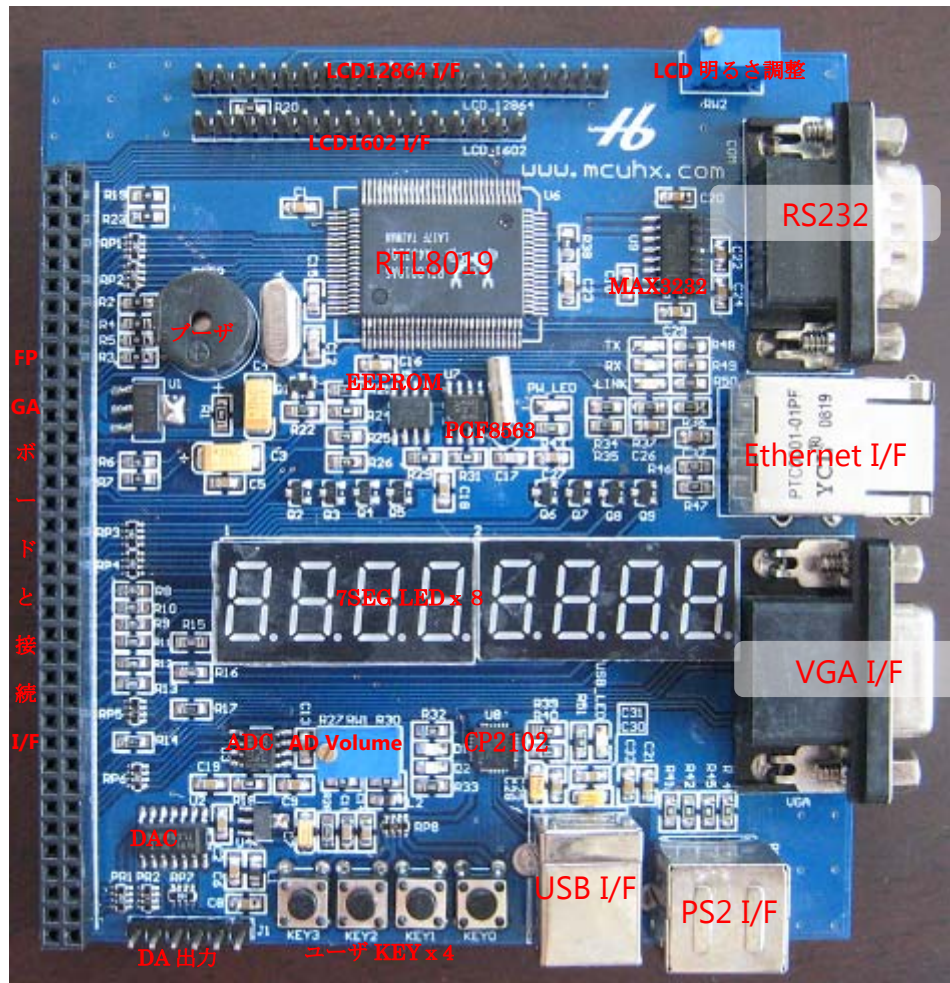
特徴	デバイス			
	EP2C5	EP2C8	EP2C20	EP3C25
ロジック・エレメント数	4,608	8,256	18,752	24,624
RAM 総ビット数	119,808	165,888	239,616	608,256
エンベデッド乗算器数	13	18	26	66
PLL 数	2	2	4	4
I/O 数	143	139	142	148

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、ケーブルを抜いてから、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 IO、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

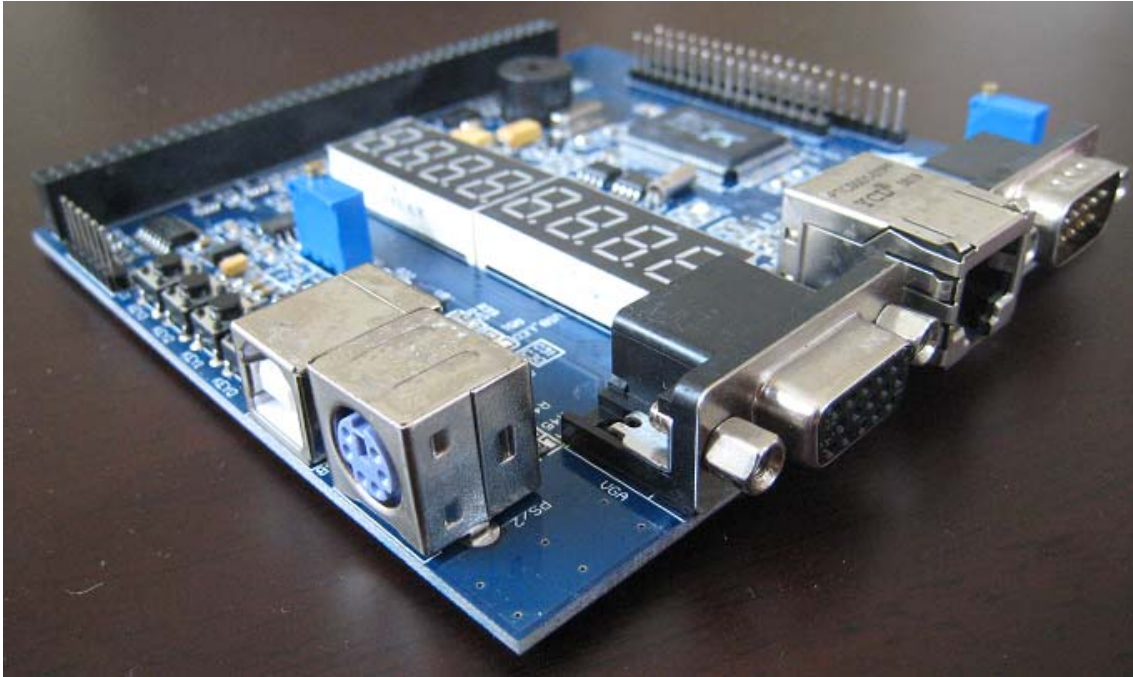


- (1) 10M Ethernet インタフェース (RTL8019)
- (2) 8 ビット ADC (TLC549)
- (3) 4 チャンネル 8 ビット DAC (TLC5620)
- (4) リアルタイムクロック (PCF8563)
- (5) RS232 (MAX3232)
- (6) VGA I/F
- (7) PS2 I/F、キーボード又はマウスを接続可
- (8) 1602 液晶 I/F
- (9) 128*64 LCD I/F
- (10) プザー
- (11) USB ホースと (CP2102)、USB1.1 と USB2.0 両方サポート
- (12) 7 セグメント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

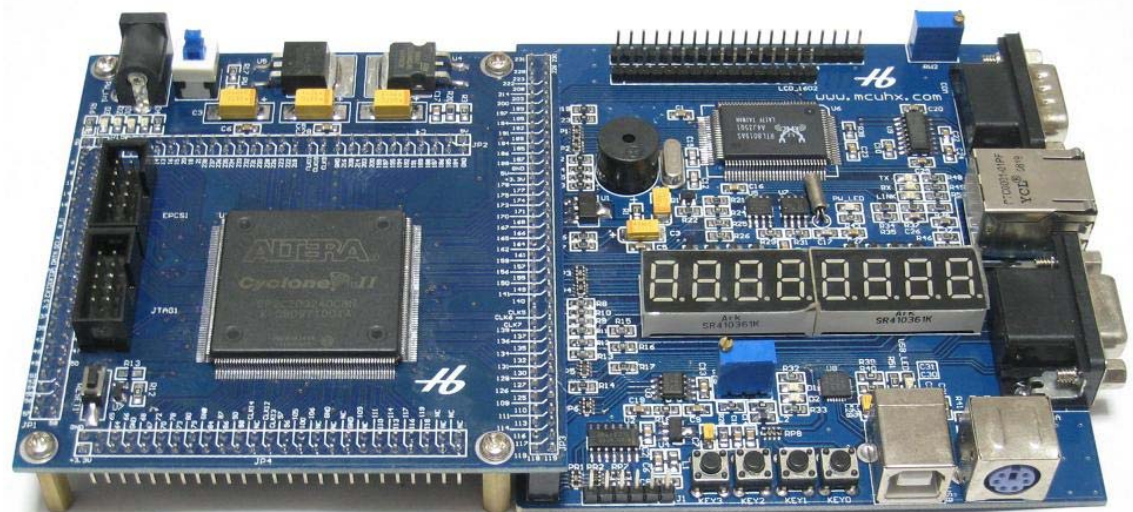
(16) 回路図を提供しております

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：





EP2C20 FPGA ボードのピンリスト：

ネーム	FPGA ピン番号	I/O タイプ	機能
SYS_CLK1	91	I	クロック入力 1
SYS_CLK2		I	クロック入力 2
SYS_nRST	92	I	FPGA リセット
LED0	216	0	四つの LED
LED1	6	0	
LED2	7	0	
LED3	8	0	
SD_DATA0	38	I/O	SDRAM データバス
SD_DATA1	37	I/O	
SD_DATA2	41	I/O	
SD_DATA3	39	I/O	
SD_DATA4	44	I/O	
SD_DATA5	42	I/O	
SD_DATA6	47	I/O	
SD_DATA7	46	I/O	
SD_DATA8	68	I/O	
SD_DATA9	67	I/O	
SD_DATA10	66	I/O	
SD_DATA11	65	I/O	
SD_DATA12	64	I/O	
SD_DATA13	56	I/O	
SD_DATA14	57	I/O	
SD_DATA15	58	I/O	
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	



SD_ADDR8	80	0	
SD_ADDR9	79	0	
SD_ADDR10	97	0	
SD_ADDR11	78	0	
SD_BA0	54	0	SDRAMのBANKアドレス
SD_BA1	90	0	
SD_DQML	50	0	データマスク
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/O	FLASH データバス (拡張ボードのイーサネットチップ、LCD12864、LCD1602 と共用)
FLASH_DQ1	203	I/O	
FLASH_DQ2	200	I/O	
FLASH_DQ3	199	I/O	
FLASH_DQ4	197	I/O	
FLASH_DQ5	195	I/O	
FLASH_DQ6	194	I/O	
FLASH_DQ7	192	I/O	
FLASH_ADDR0	208	0	FLASH アドレスバス (拡張ボードのイーサネットチップと共用)
FLASH_ADDR1	231	0	
FLASH_ADDR2	230	0	
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	



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	書き込み信号(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	I/O タイプ	機能
KEY1	110	I	四つのキー
KEY2	111	I	
KEY3	113	I	
KEY4	114	I	
78LED_COM0	161	0	7SEG LED 共通ポート
78LED_COM1	159	0	
78LED_COM2	164	0	
78LED_COM3	162	0	
78LED_COM4	166	0	
78LED_COM5	165	0	
78LED_COM6	168	0	
78LED_COM7	167	0	
78LED_DATA0	156	I/O	78LED_Da

78LED_DATA1	150	I/O	78LED_Db
78LED_DATA2	140	I/O	78LED_Dc
78LED_DATA3	141	I/O	78LED_Dd
78LED_DATA4	155	I/O	78LED_De
78LED_DATA5	157	I/O	78LED_Df
78LED_DATA6	139	I/O	78LED_Dg
78LED_DATA7	149	I/O	78LED_Dh
KB_DATA	125	I	PS2 データライン
KB_CLK	109	I	PS2 クロックライン
I2C_SCL	173	0	I2C クロックライン
I2C_SDA	174	I/O	I2C データライン
LCD12864_CS1	184	0	LCD12864 左右チップ セクタ信号
LCD12864_CS2	185	0	
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 セクタ信号
RTL8019_INT	177	I	
RTL8019_RST	188	0	イーサネットリセット 信号
RXD_URAT	170	I	RS232
TXD_URAT	171	0	
RXD_USB	130	I	USB URAT I/F
TXD_USB	131	0	
VGA_HS	132	0	VGA I/F
VGA_VS	134	0	
VGA_R	135	0	



VGA_G	136	0	
VGA_B	137	0	
AD_CLK	126	0	ADC (TLC549) I/F
AD_CS	128	0	
AD_DATA	127	I	
TLC5620_CLK	118	0	DAC (TLC5620) I/F
TLC5620_DATA	119	0	
TLC5620_LDAC	117	0	
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
 - [+] 1key1LED
 - [+] 4key4LED
 - [+] VoltingMachine
 - [+] WaterLamp
 - [-] 2.7SegLED
 - [+] DynamicDisplay
 - [+] PlusMinusDisplay
 - [+] StaticDisplay
 - [-] 3.FreqCounter
 - [+] verilogHDL
 - [+] VHDL
 - [-] 4.music
 - [+] verilogHDL_simpleSound
 - [+] VHDL_simpleSound
 - [-] 5.IIC
 - [+] verilogHDL_IIC
 - [+] VHDL_IIC
 - [-] 6.AD
 - [+] verilogHDL_AD
 - [+] VHDL_AD
 - [-] 7.DA
 - [+] VHDL_DA
 - [+] vreilogHDL_DA
 - [-] 8.LCD1602
 - [+] VHDL_LCD1602
 - [+] vreilogHDL_LCD1602
 - [-] 9.LCD12864
 - [+] LCD12864_v
 - [-] 10.UART
 - [+] verilogHDL_uart
 - [+] VHDL_uart
 - [-] 11.VGA
 - [+] verilogHDL_VGA_basic
 - [+] VGA_test
 - [+] VHDL_VGA_basic
 - [-] 12.PS2
 - [+] 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. LED¥4key4LED

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

ブザーテスト

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

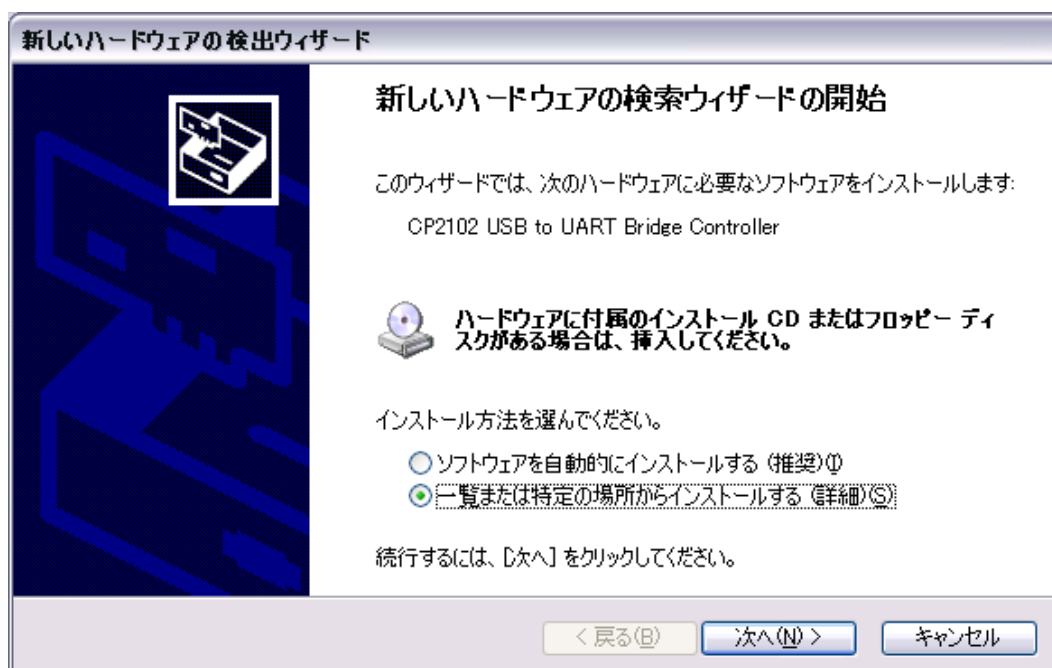
PS2 キーボードテスト

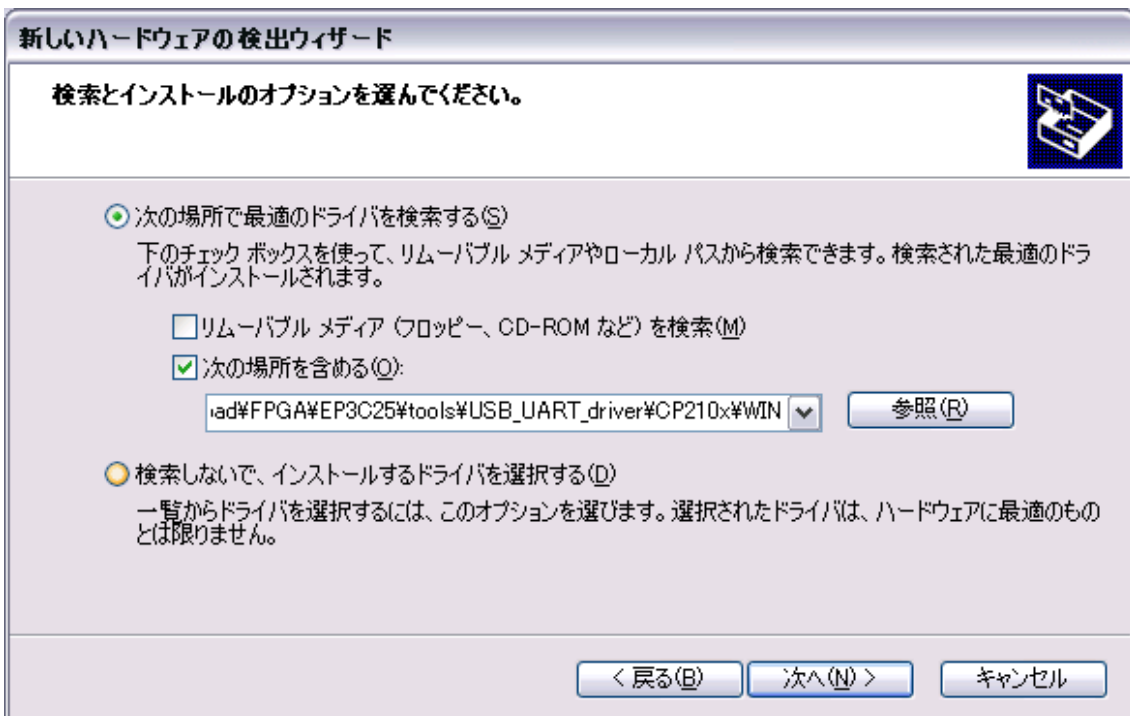
1.5 USB_URATについて

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。

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）。





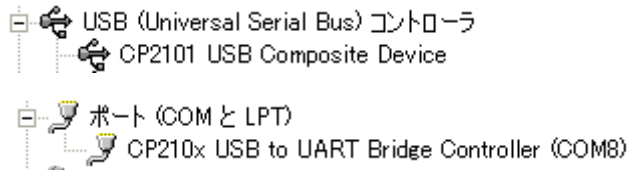
完了すると、自動的に次の検出ウィザード画面が表示される。上記と同じ手順でインストールする。



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)



1.5.2 USB_URATの通信設定

PC側のハイパーターミナル(ここでは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

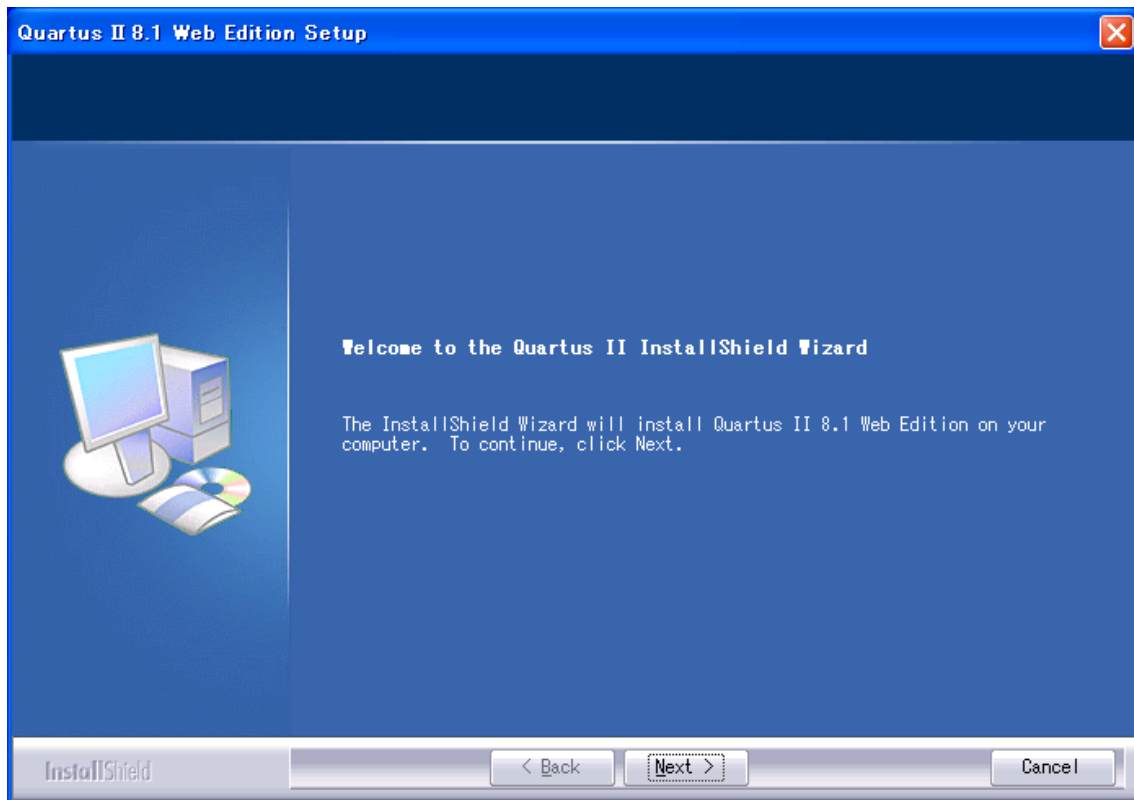
CPLD/FPGA の開発には、ALTERA から Quartus II Web Edition という無償版のツールが公開されているのでこちらを利用します。Quartus II Web Edition は、総合開発環境になっており、このソフトウェアだけで、ソース・エディタや I/O ピンのアサインメント、論理合成、デバイスの書き込み用のプログラムなど、CPLD/FPGA の開発に必要な機能がすべて含まれています。また、Nios II エンベデッド・デザイン・スイートは Nios プロセッサ用の開発ツールです。

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。

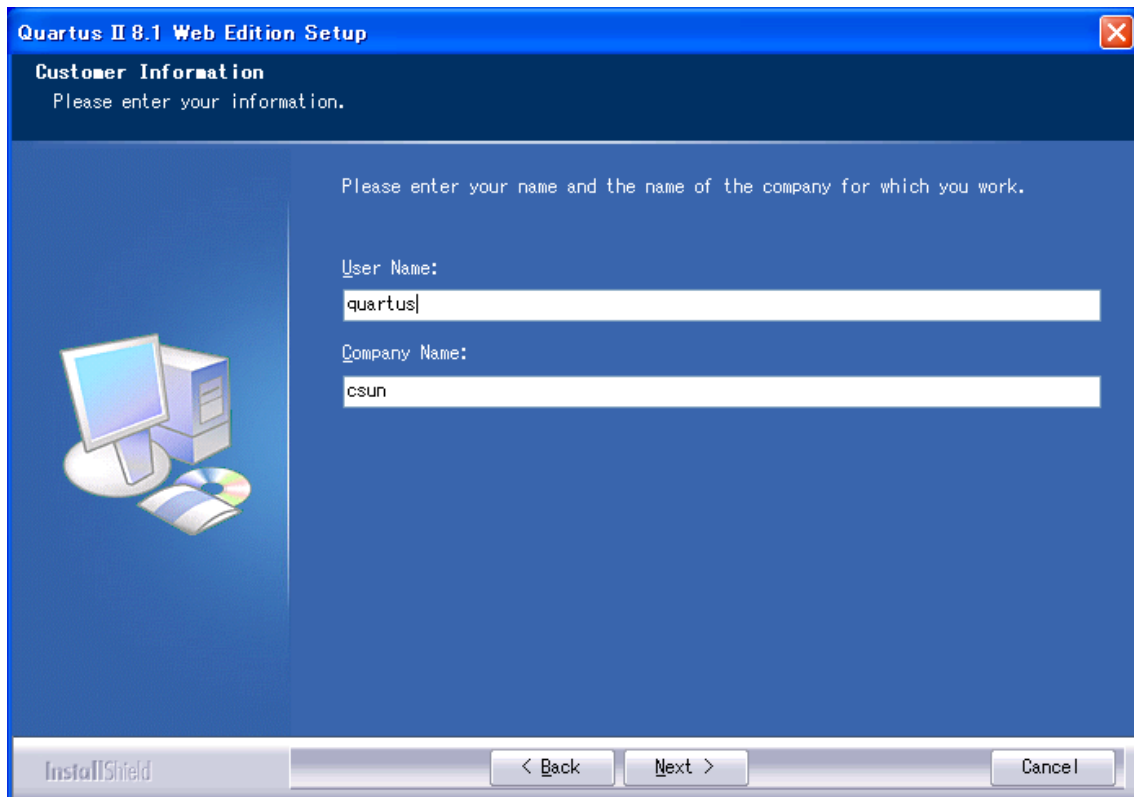
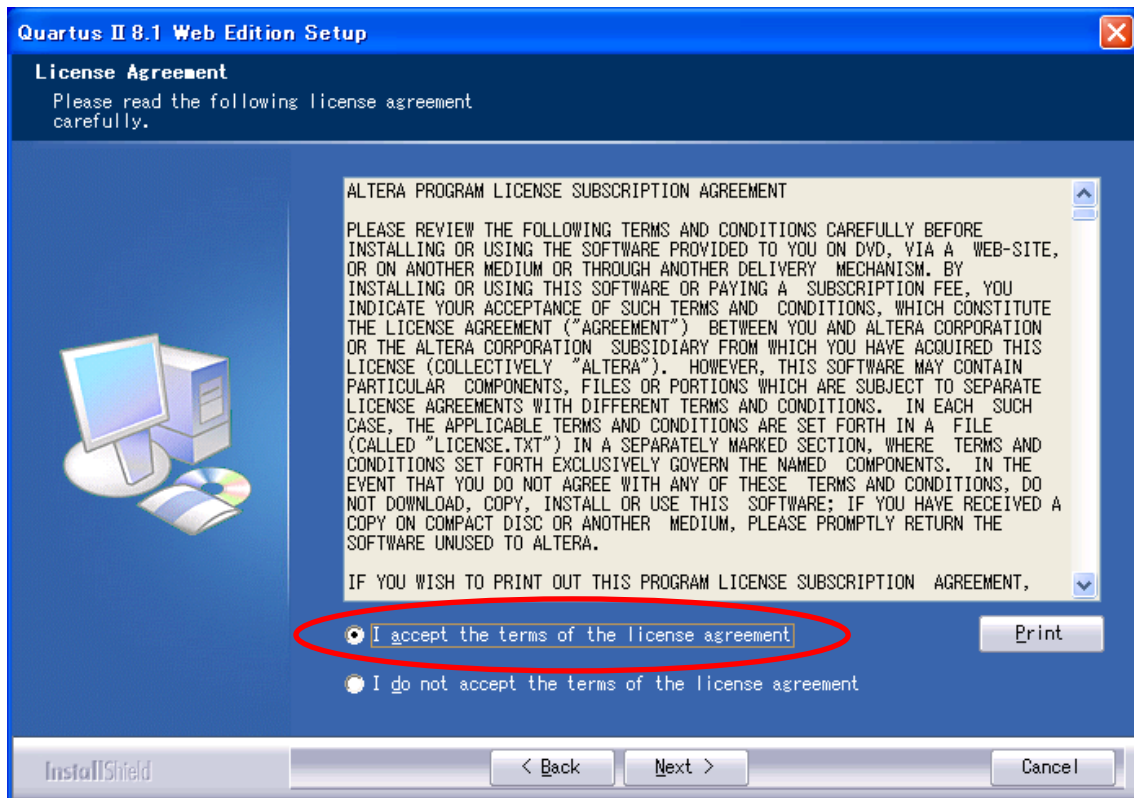
<http://www.altera.co.jp/support/software/download/nios2/dnl-nios2.jsp>

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。

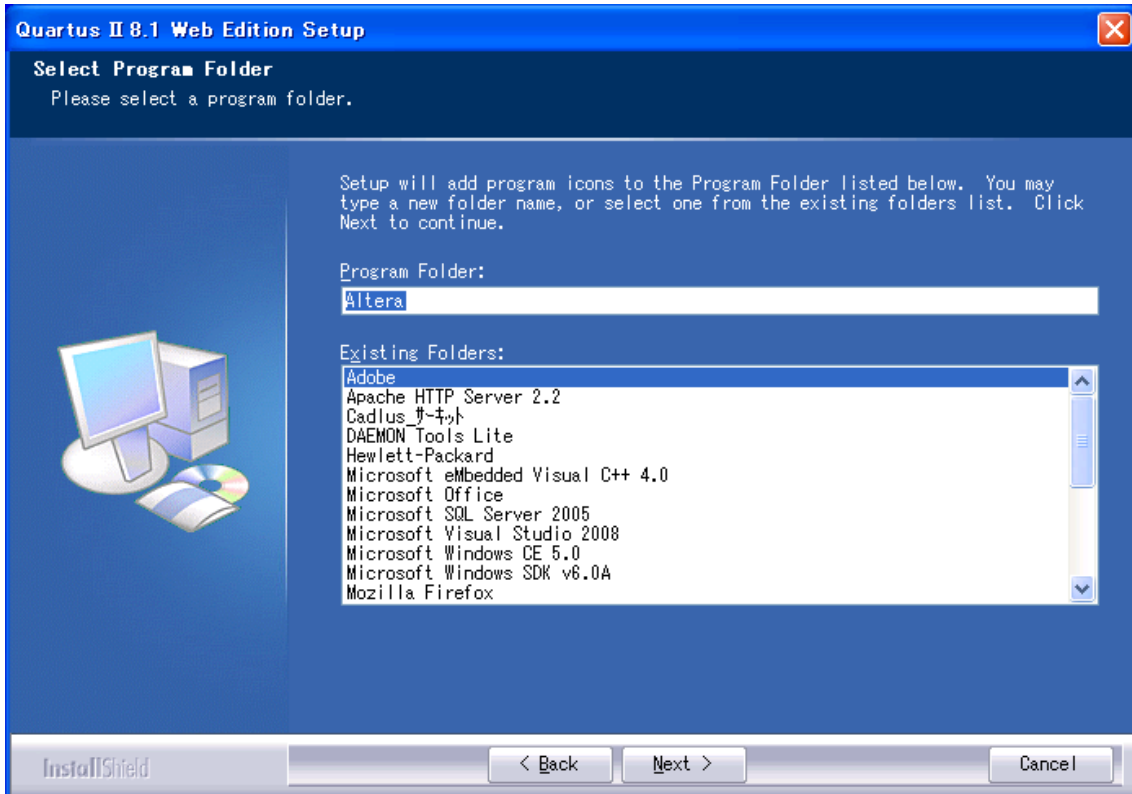
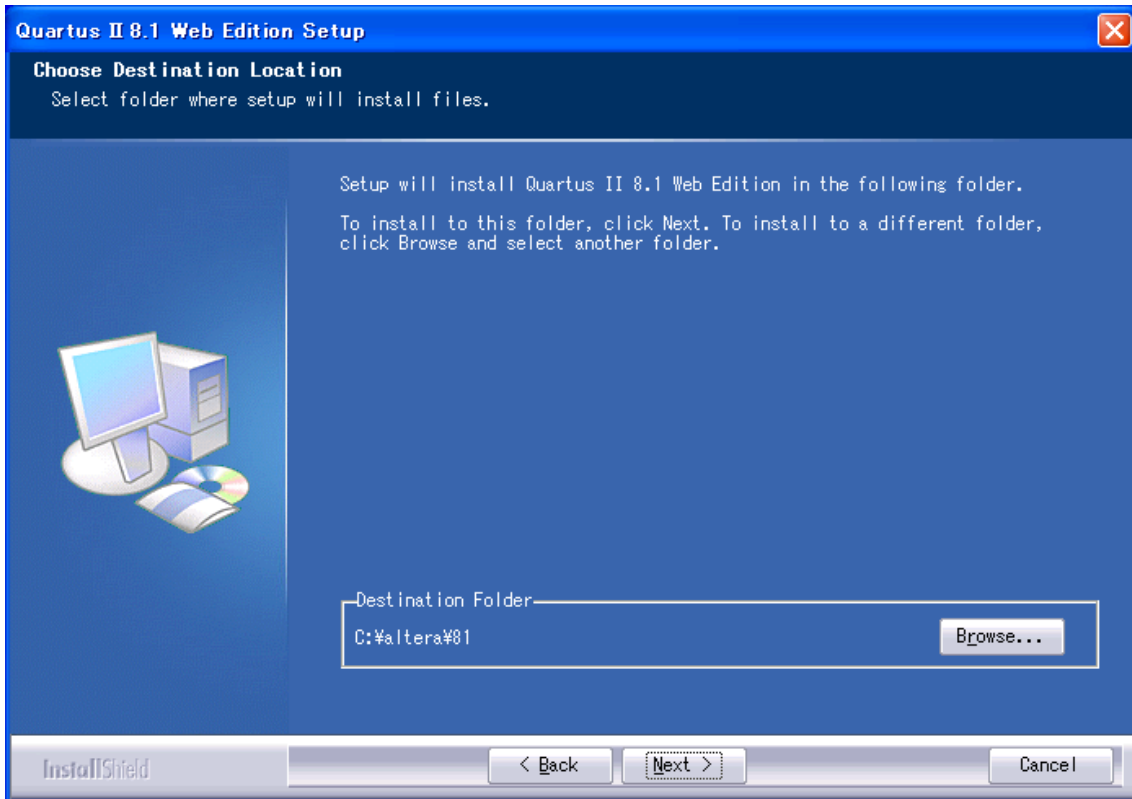
2.1 Quartus II Web Editionをインストールする



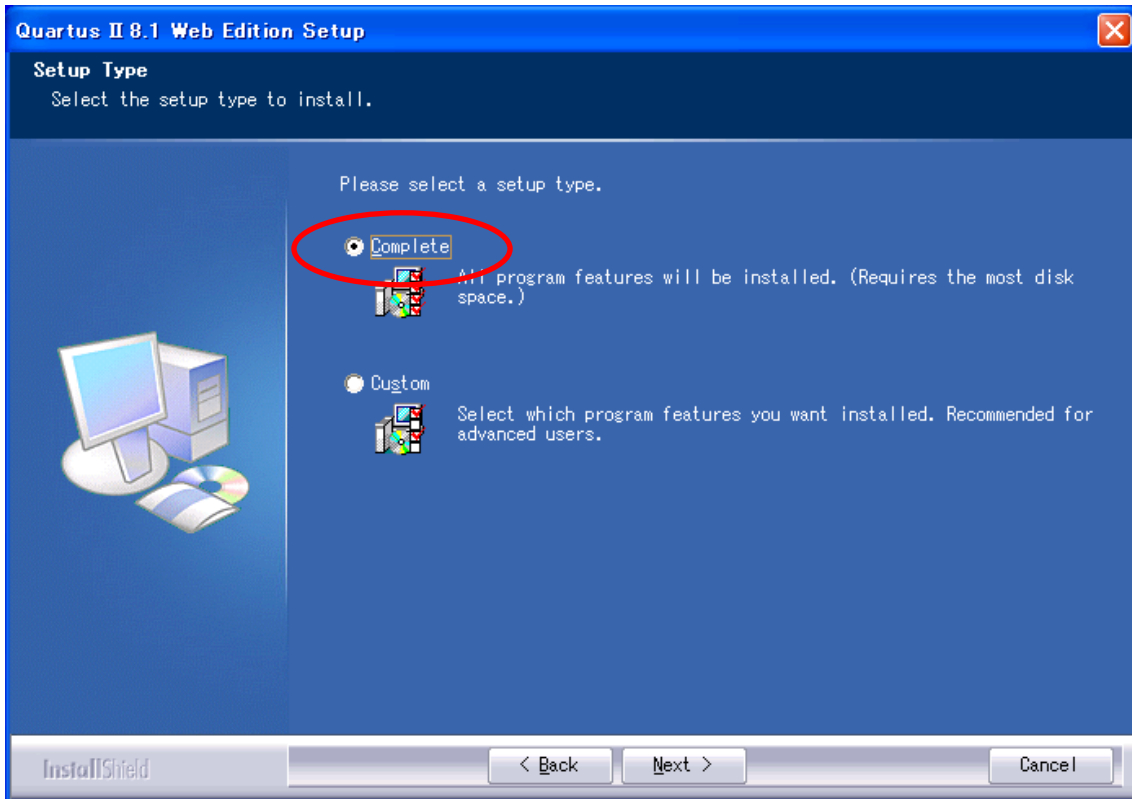
「Next」ボタンを押すと、英文のライセンスが出てきます。同意できる場合は、「I accept the terms of the license agreement」を選択して、「Next」ボタンを押します。



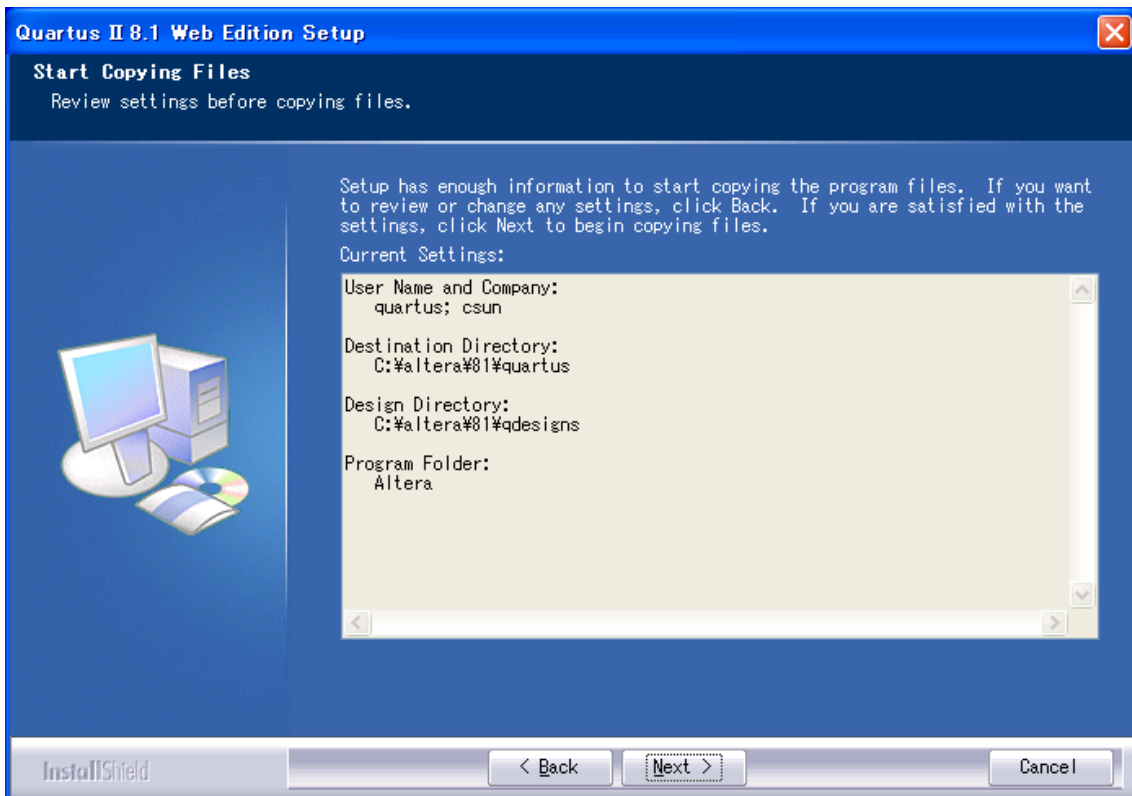
使用者の名前と所属会社名を入力するダイアログが表示されます。名前は半角のアルファベットで入力しましょう。



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。



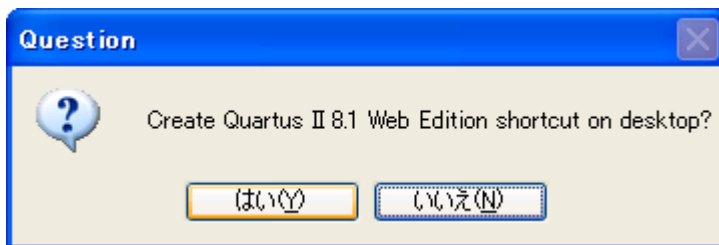
「Complete」を選択してください。



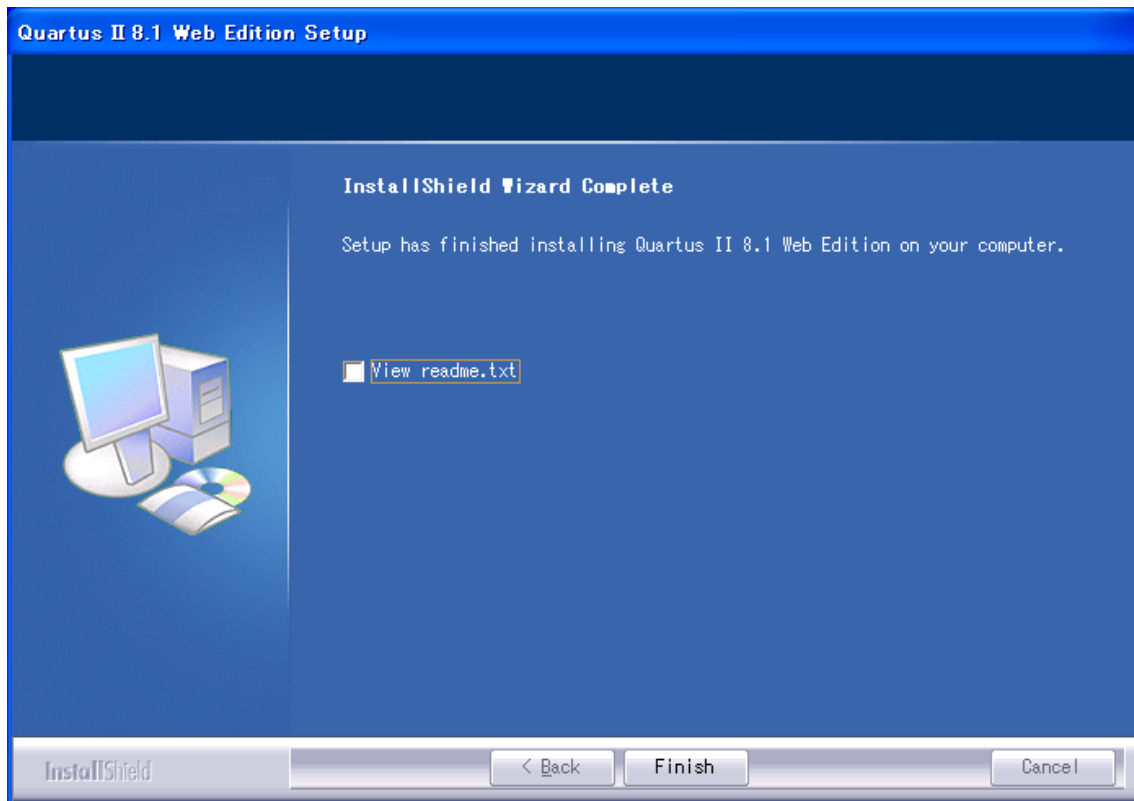
間違いがないかどうか確認し、問題がなければ「Next」を押します。



インストール中の画面です。

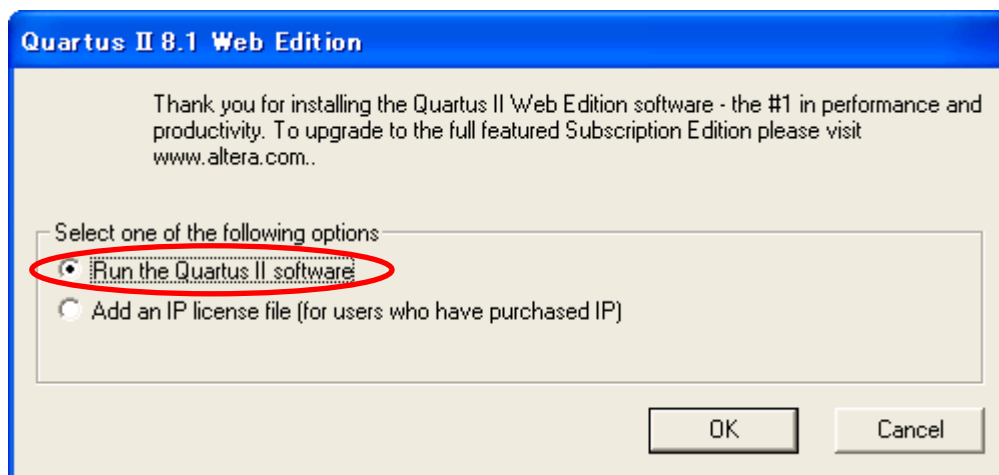


インストール完了すると、ショートカットをデスクトップに作るかどうか聞かれます。どちらでも選択できます。



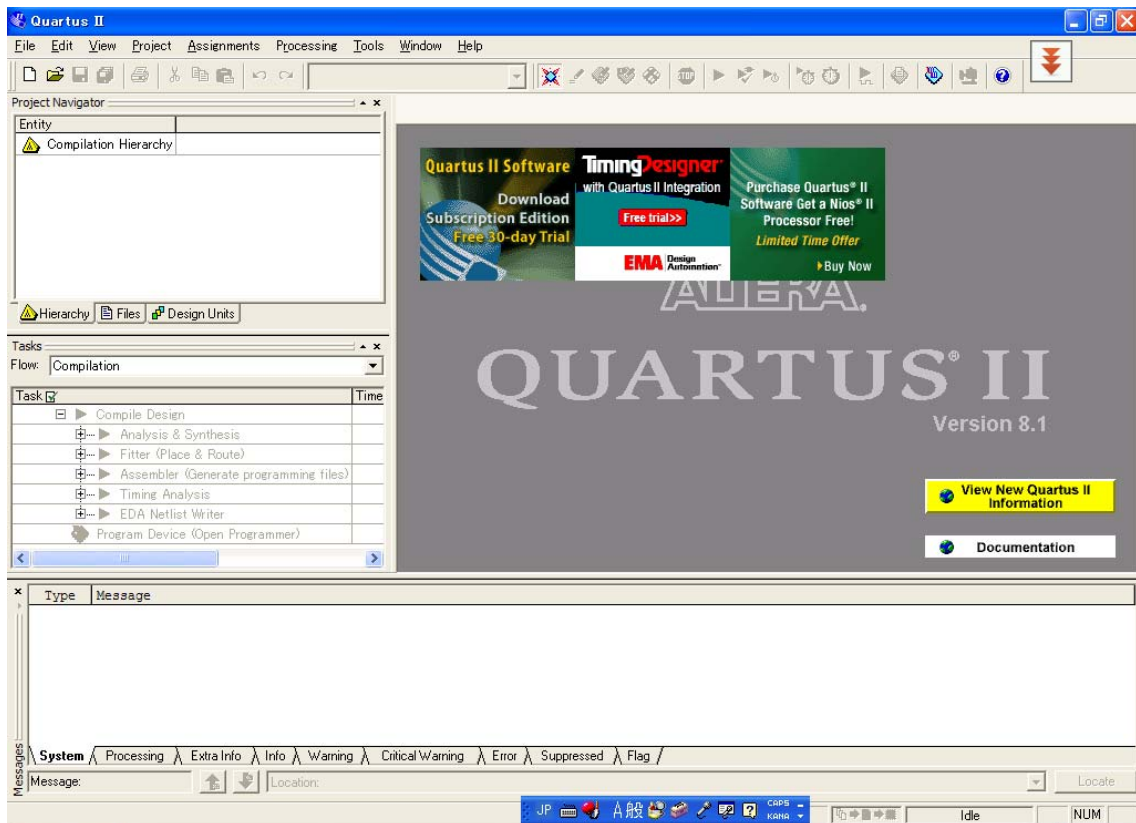
最後に「Finish」をクリックすると、ウィザードが閉じてインストールが終了します。

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。

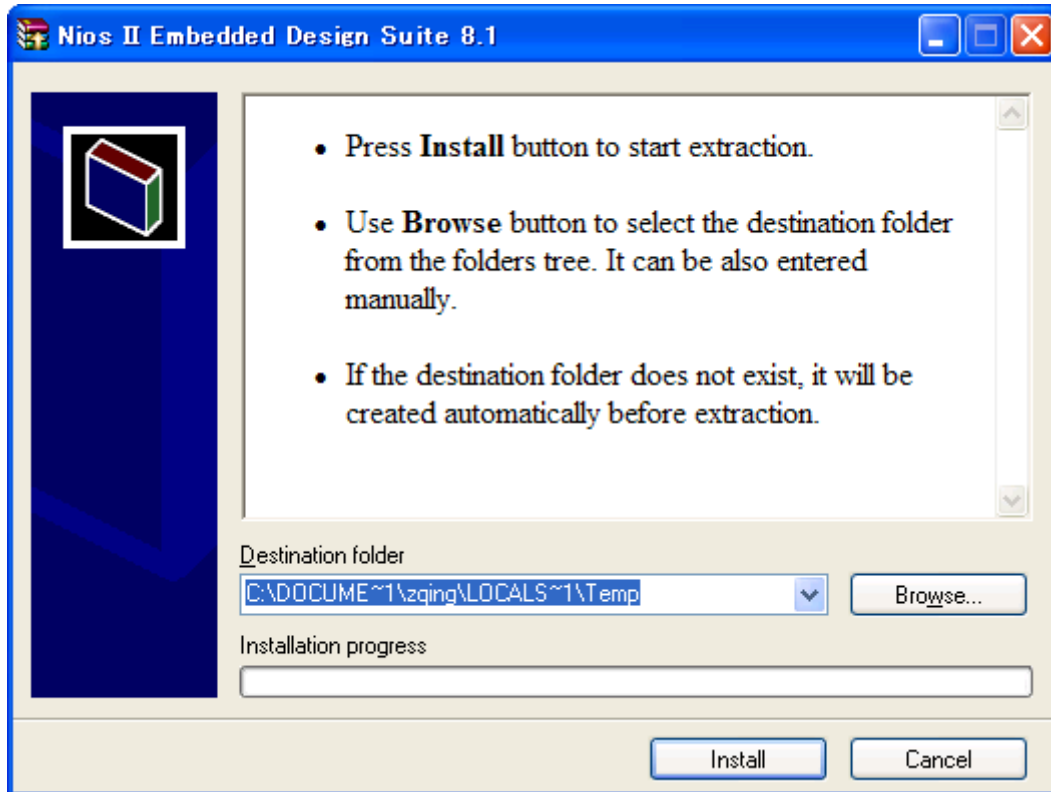


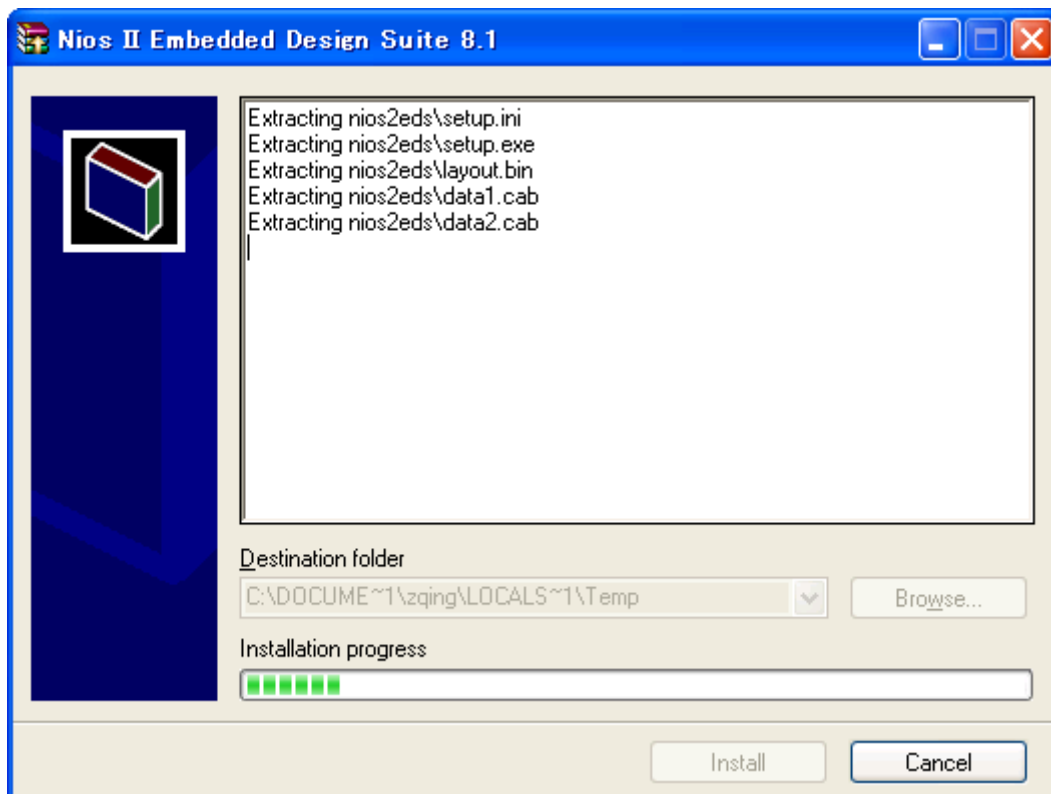


Quartus II の画面出てきます。

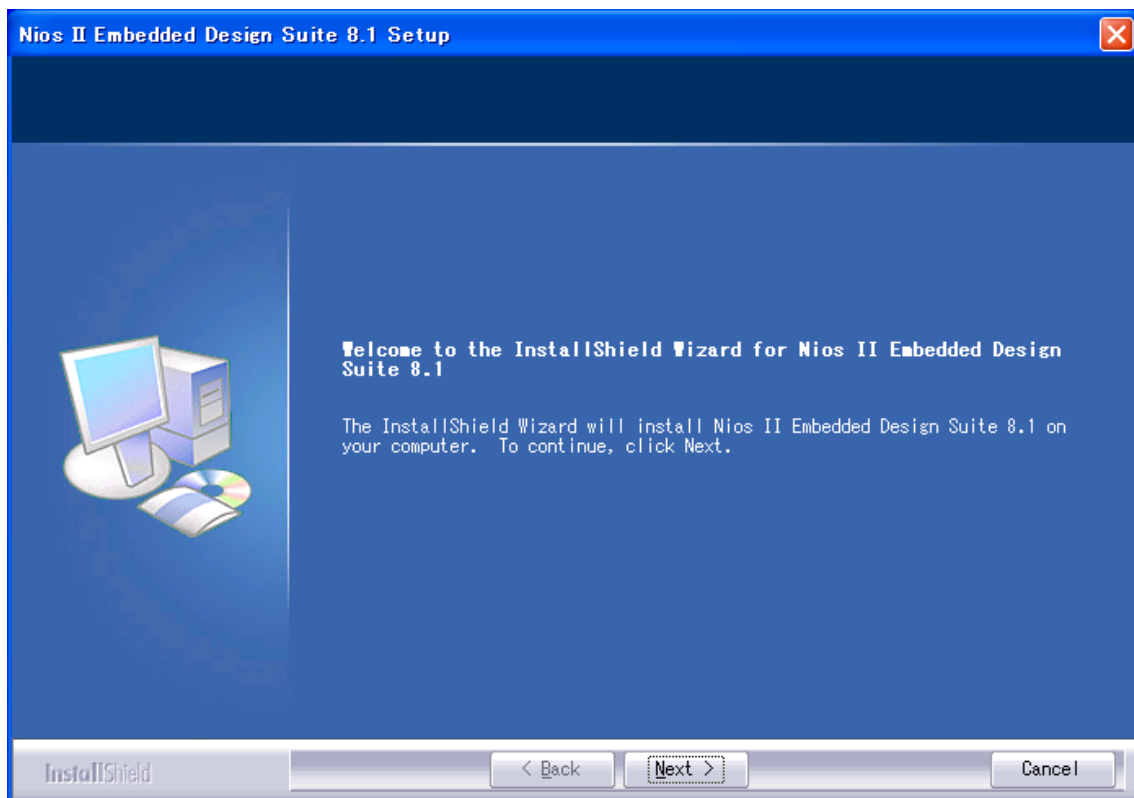


2.2 Nios II エンベデッド・デザイン・スイートをインストールする

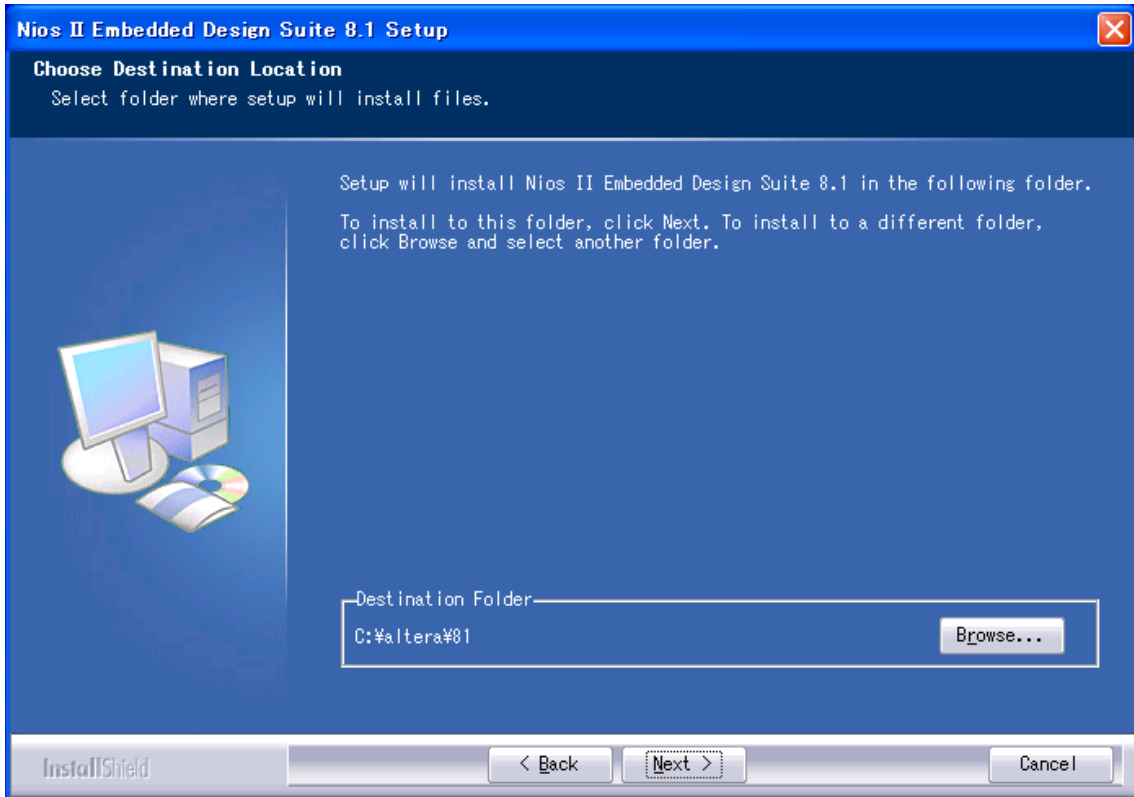
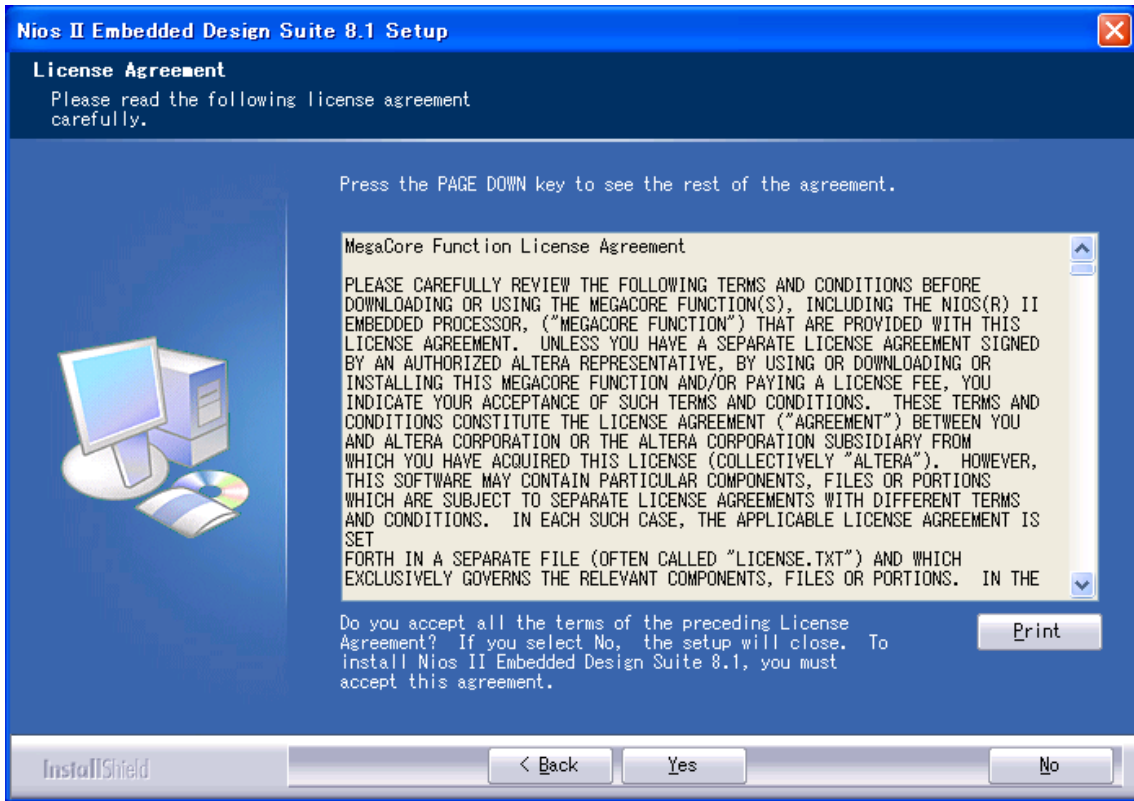


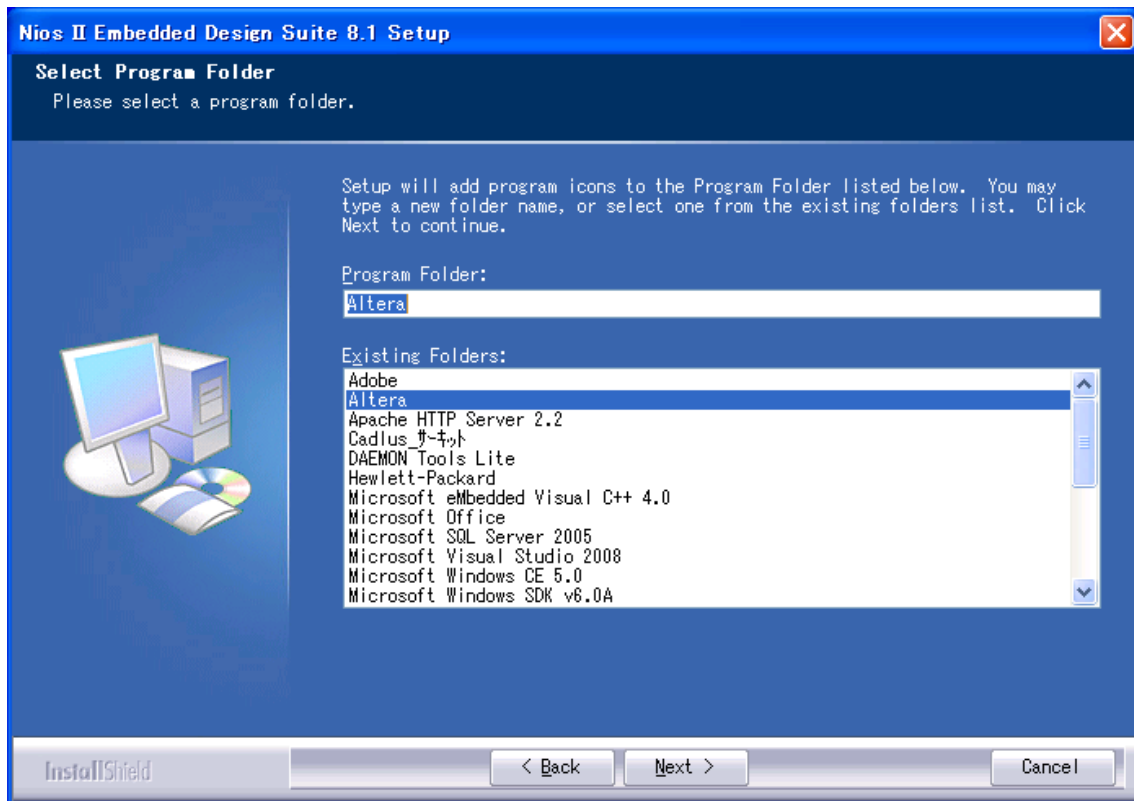


先ず「Install」ボタンを押して解凍します。「Next」ボタンを押します。

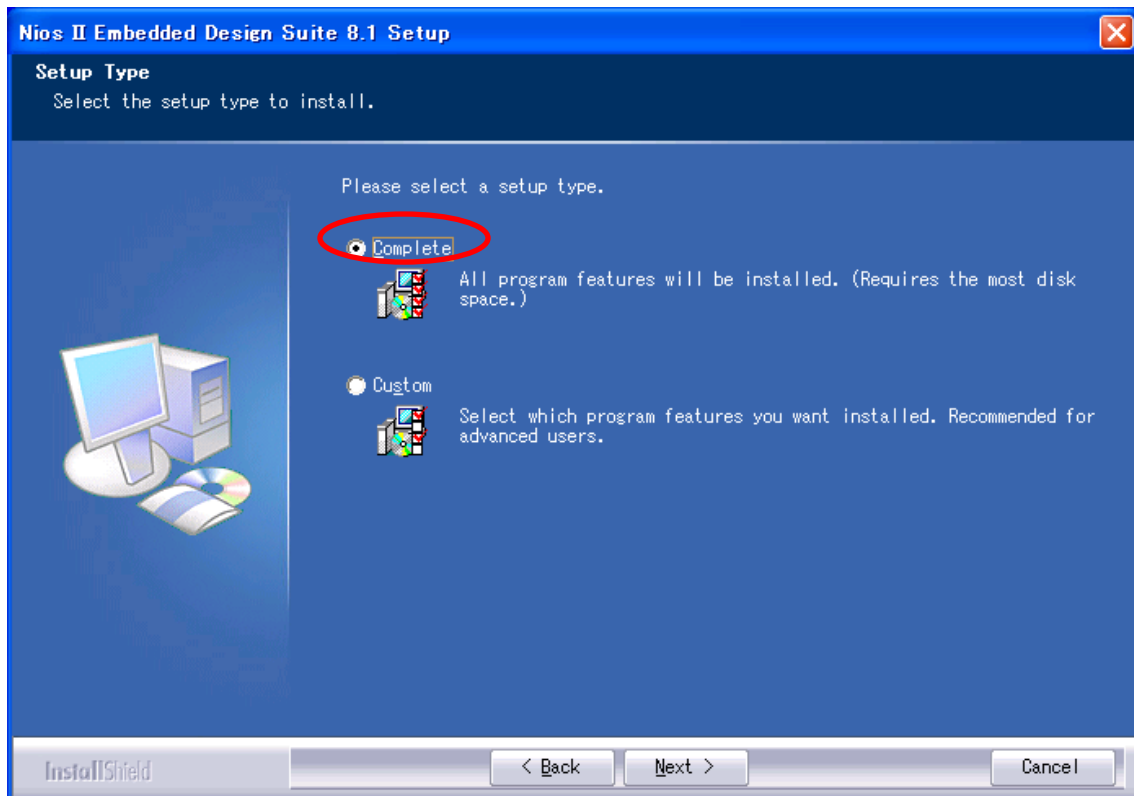


ライセンスを同意すれば、「Yes」ボタンを押します。

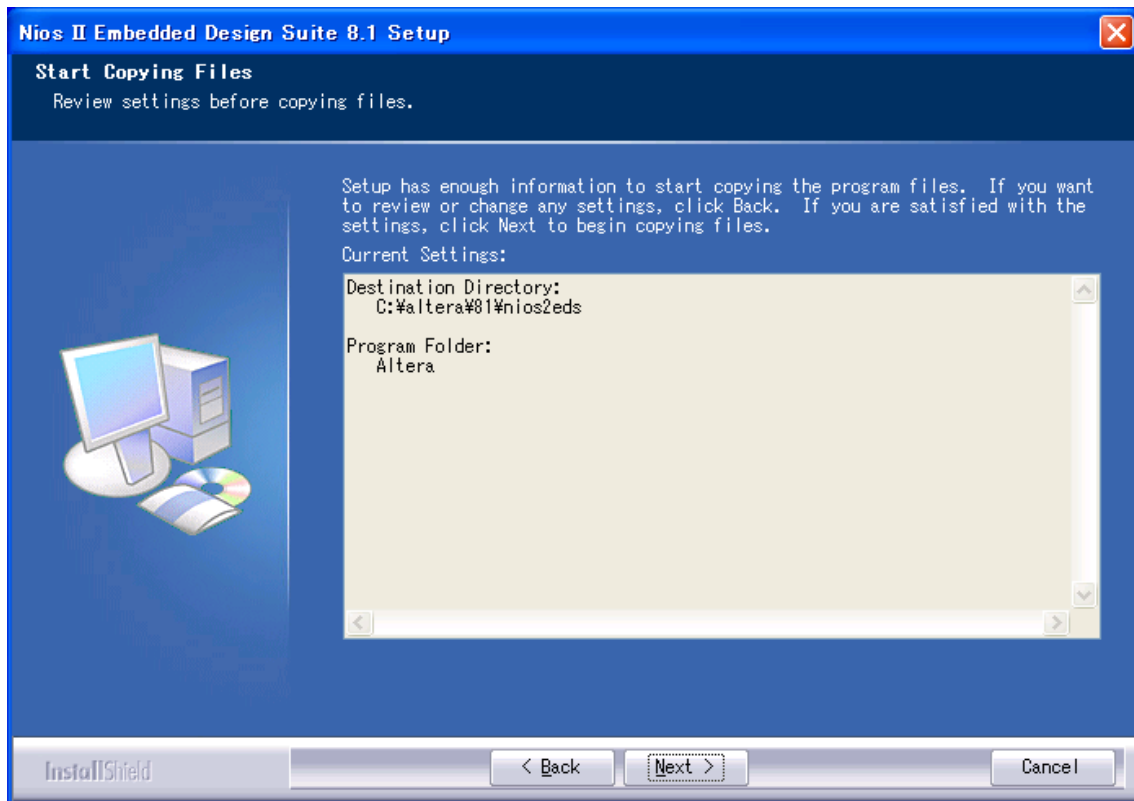




インストール先フォルダを変更せず、そのまま進んでください。



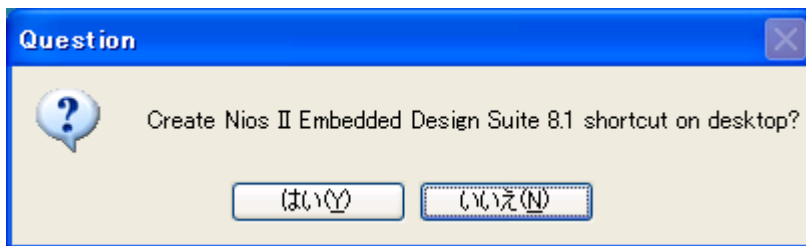
「Complete」を選択してください。



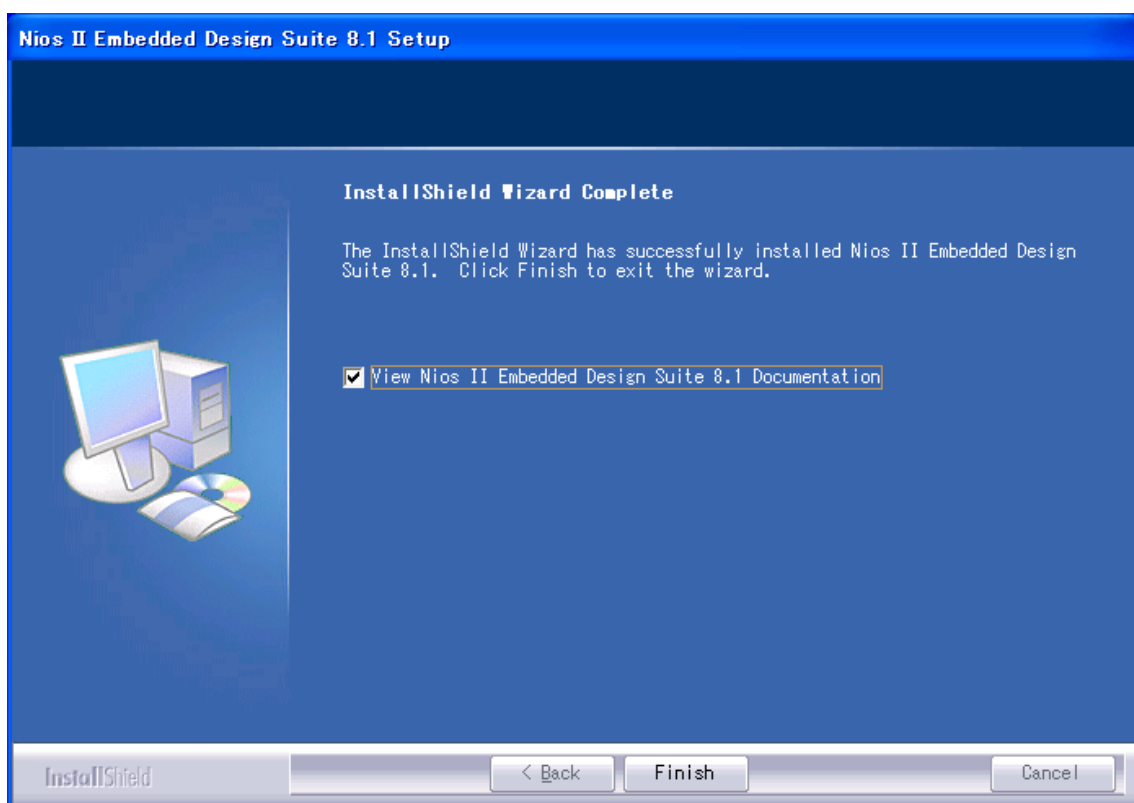
間違いがないかどうか確認し、問題がなければ「Next」を押します。



インストール中。



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

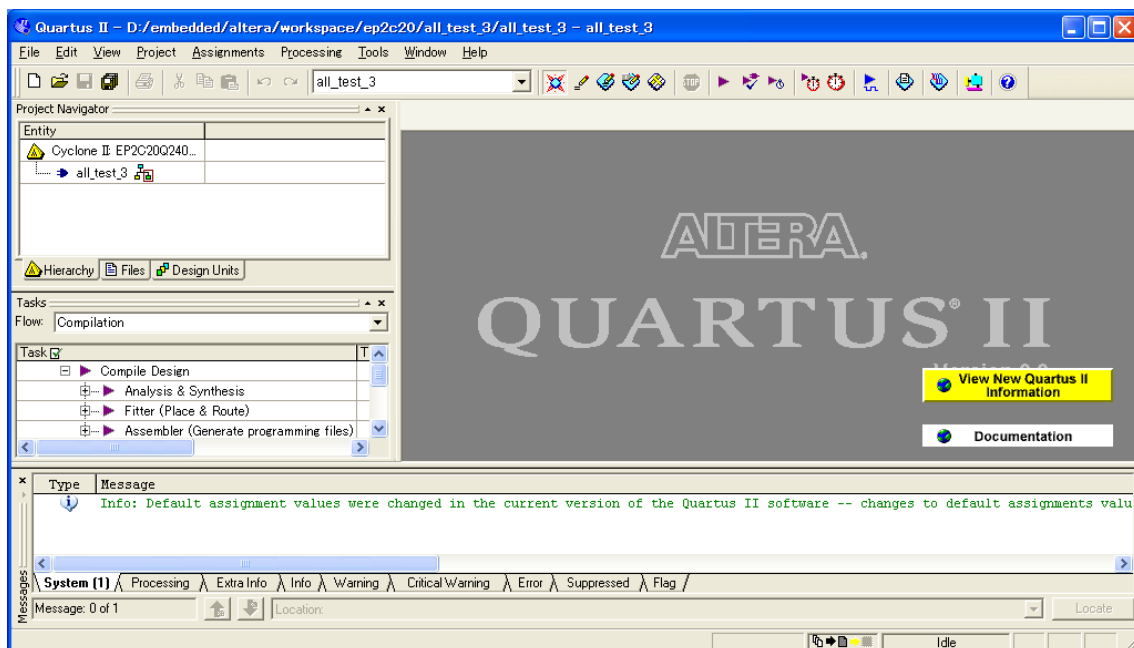
3.1 Quartus評価版にソースを読み込む

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。

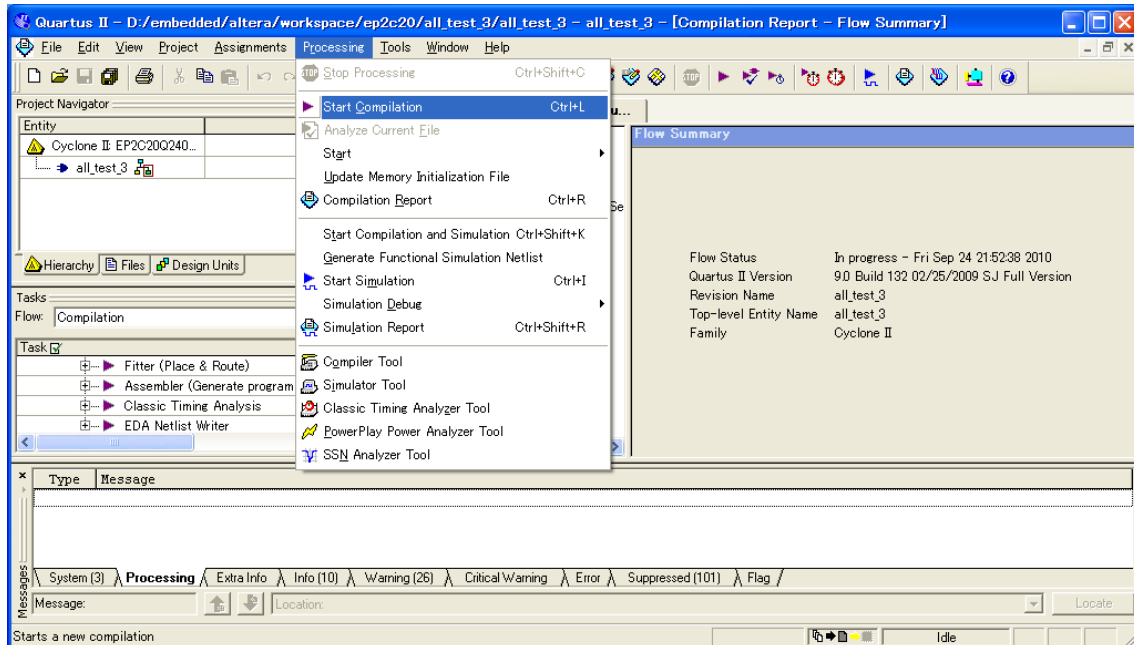
ソース・ファイルを..¥altera に展開します。その中に、幾つのサンプルがあります。具体的には 1.4 節をご参照ください。一つのサンプルを紹介します。

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。

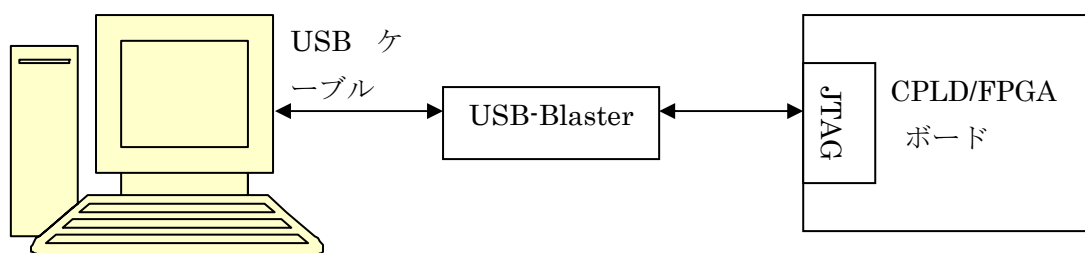


Quartus II の「Processing」メニューから「Start Compilation」を選択します。するとコンパイル処理が始まり、プロGRESS・バーが働き始めます。コンパイルは少し時間が掛かります。



3.2 USB-Blasterをインストールする

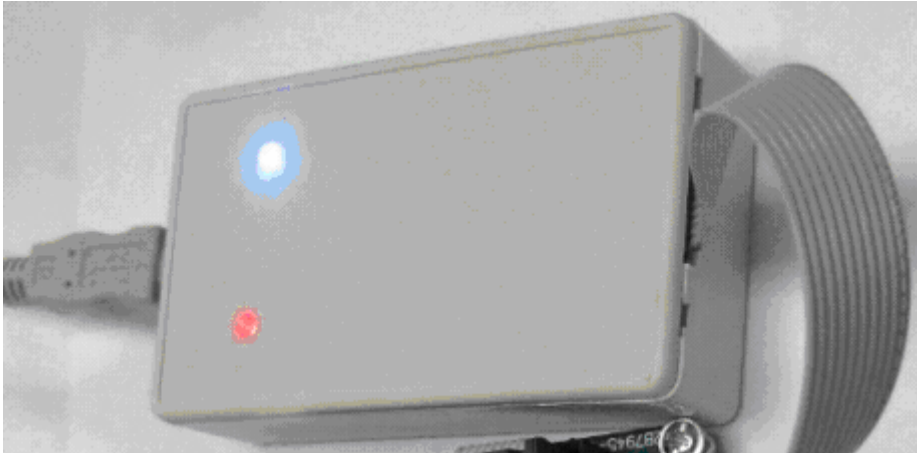
通常、MAX II/Cyclone II にコンフィグレーション・データを書き込むために、アルテラが発売している専用ダウンロード・ケーブル(ByteBlaster MV や ByteBlasterII や USB 接続タイプの USB-Blaster など)を購入しなければなりません。



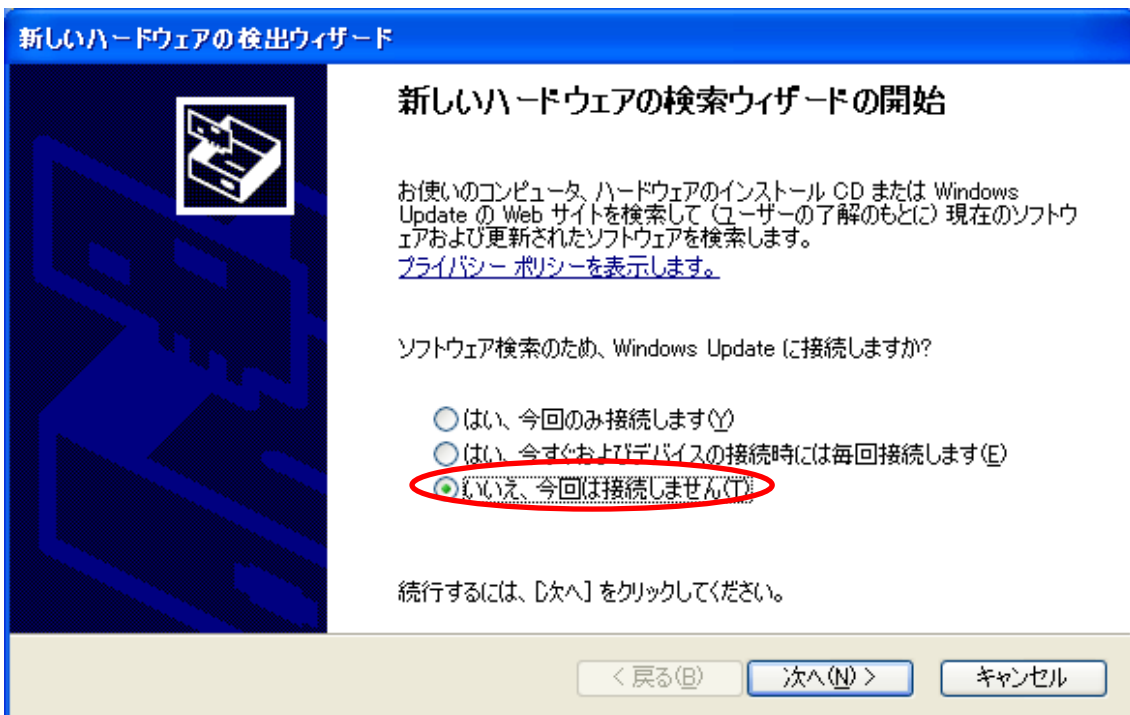
弊社は専用ダウンロード・ケーブル USB-Blaster 同等のデバイスを提供しております。

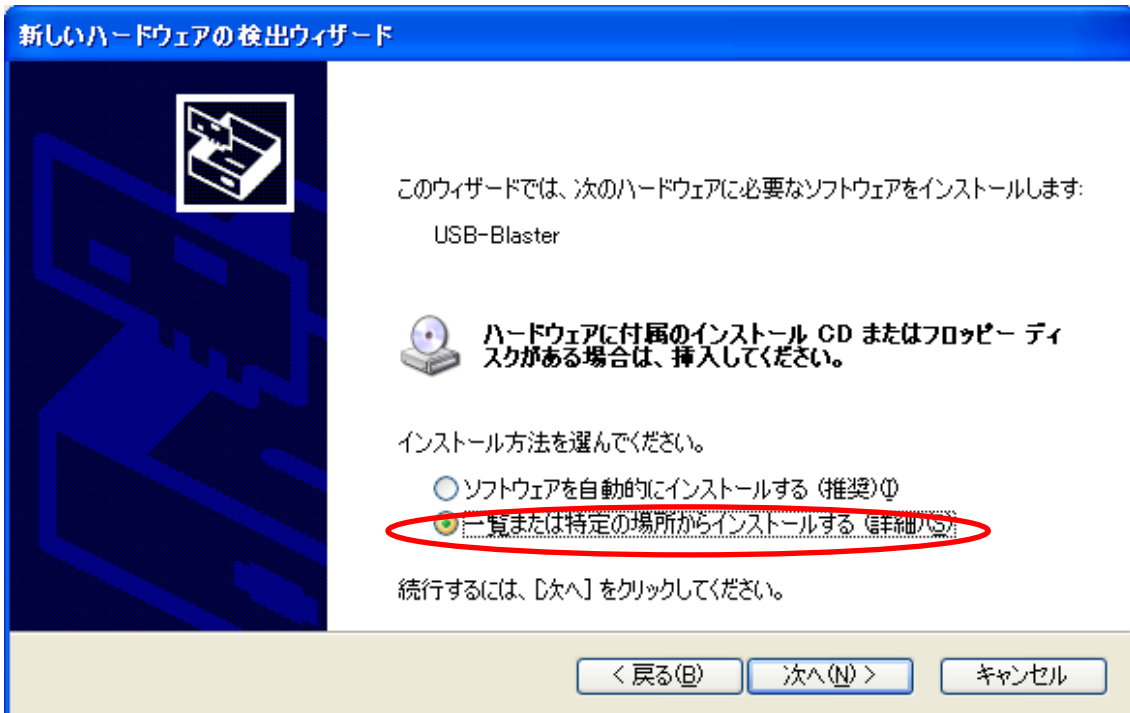
<http://www.csun.co.jp/SHOP/200901025.html>

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。

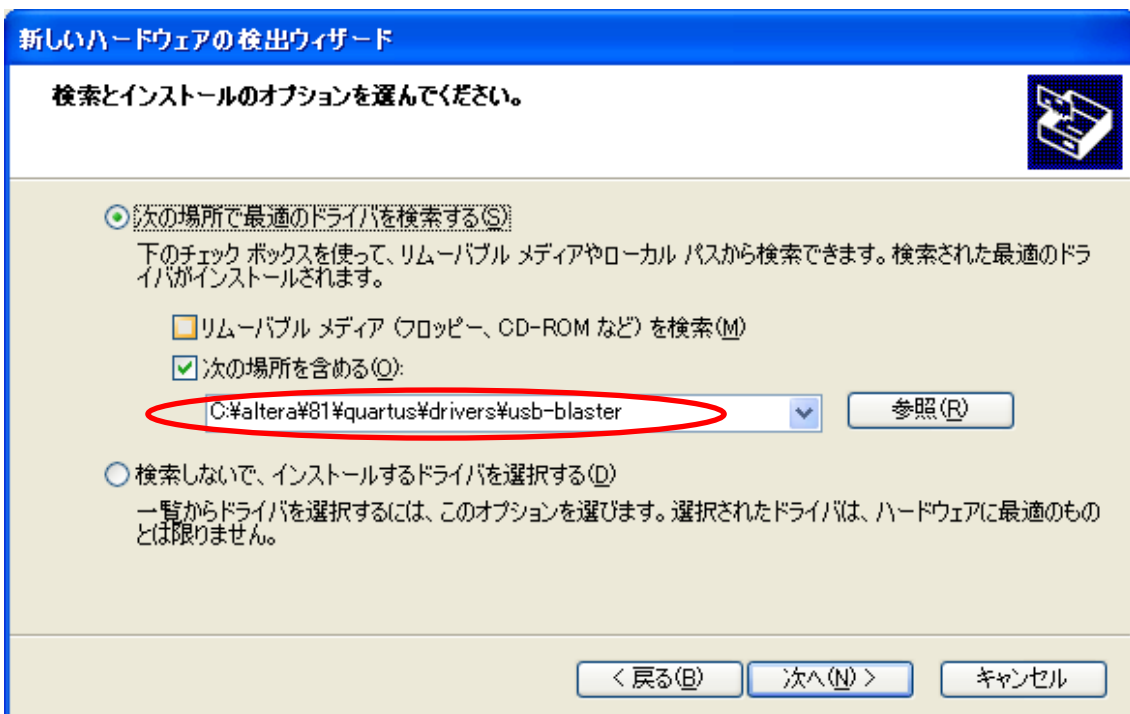


USB-Blaster を USB ケーブルでパソコンと繋ぐと、自動的にこの画面が現れ、「いいえ、今回は接続しません」を選択してください。

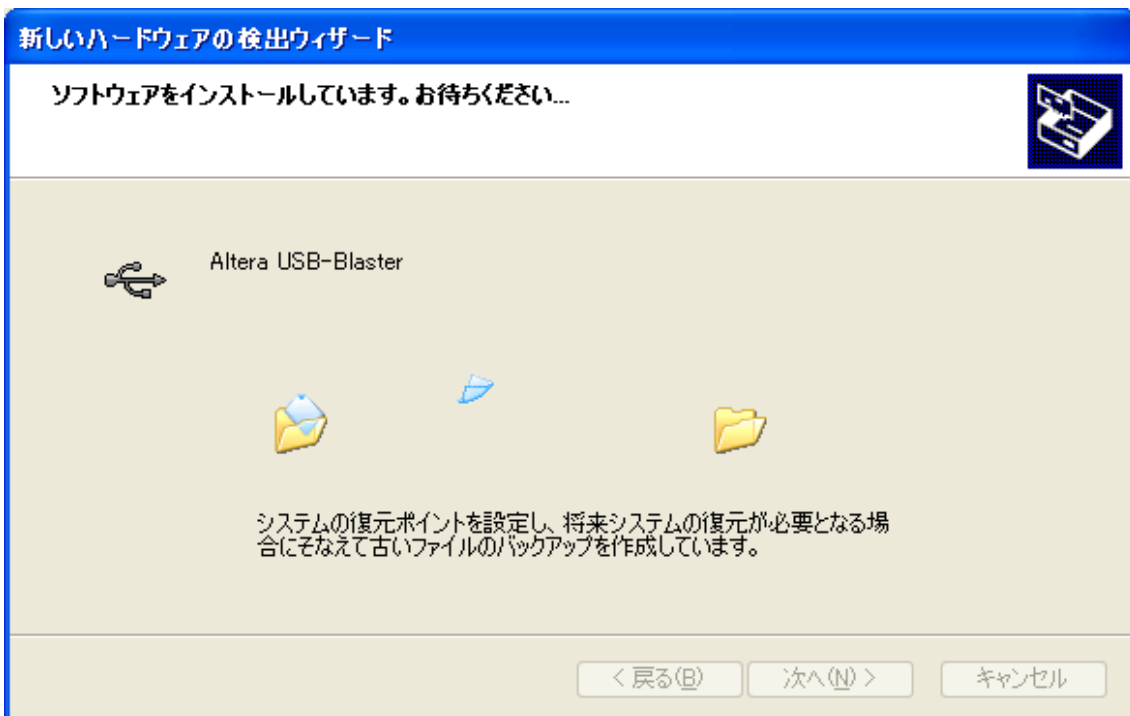




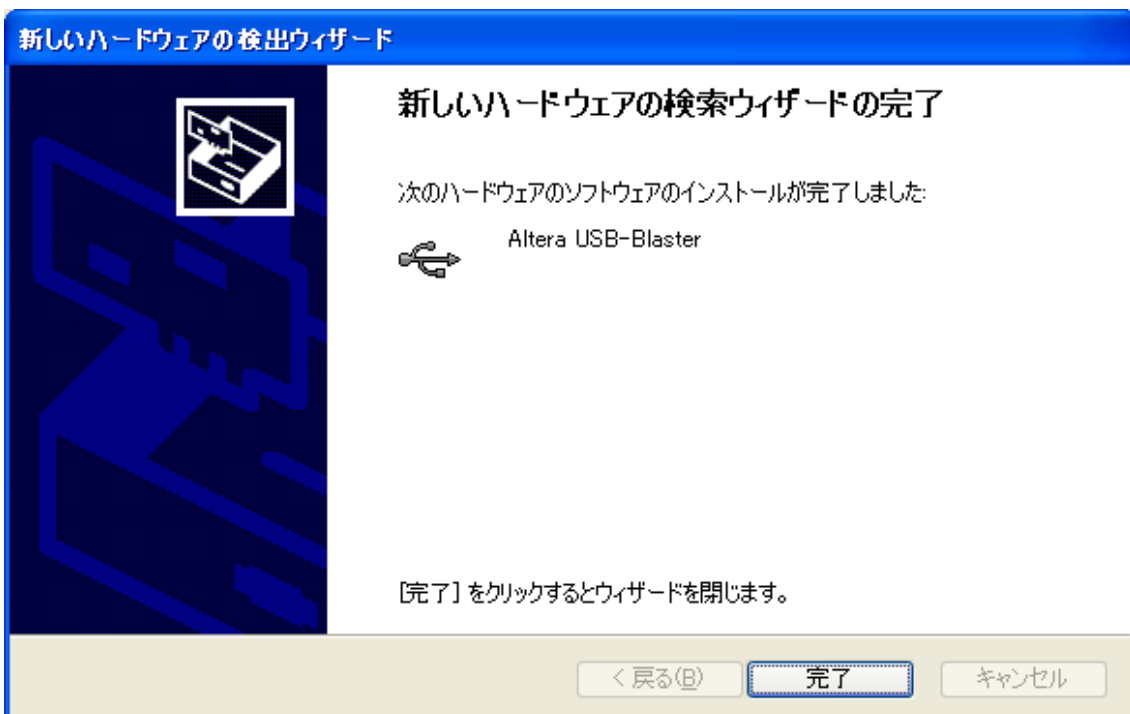
「一覧または特定の場所からインストール」を選択してください。



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



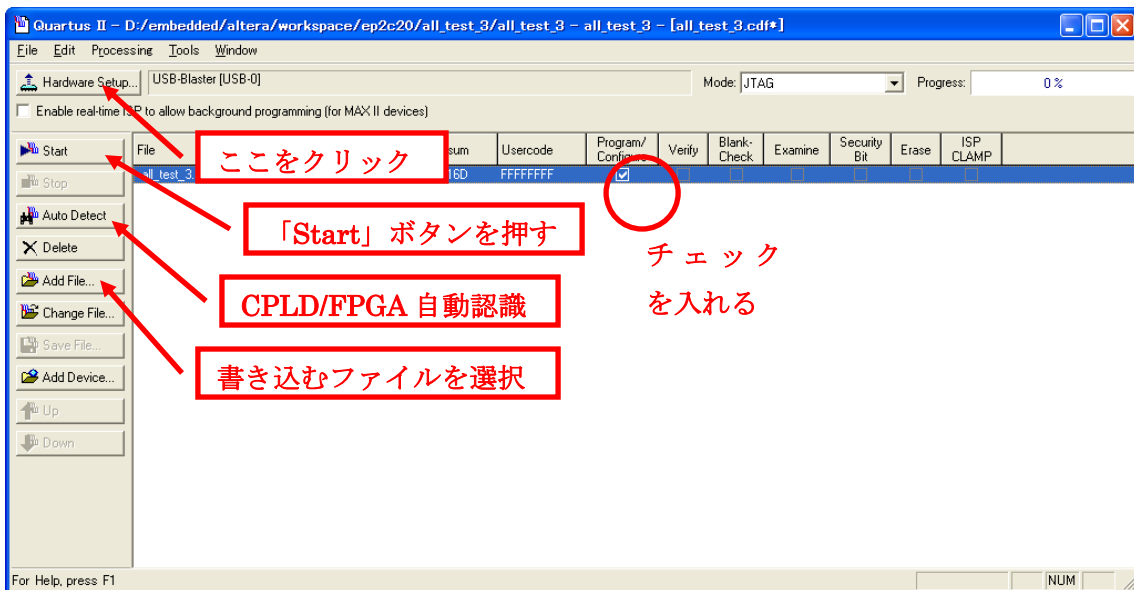
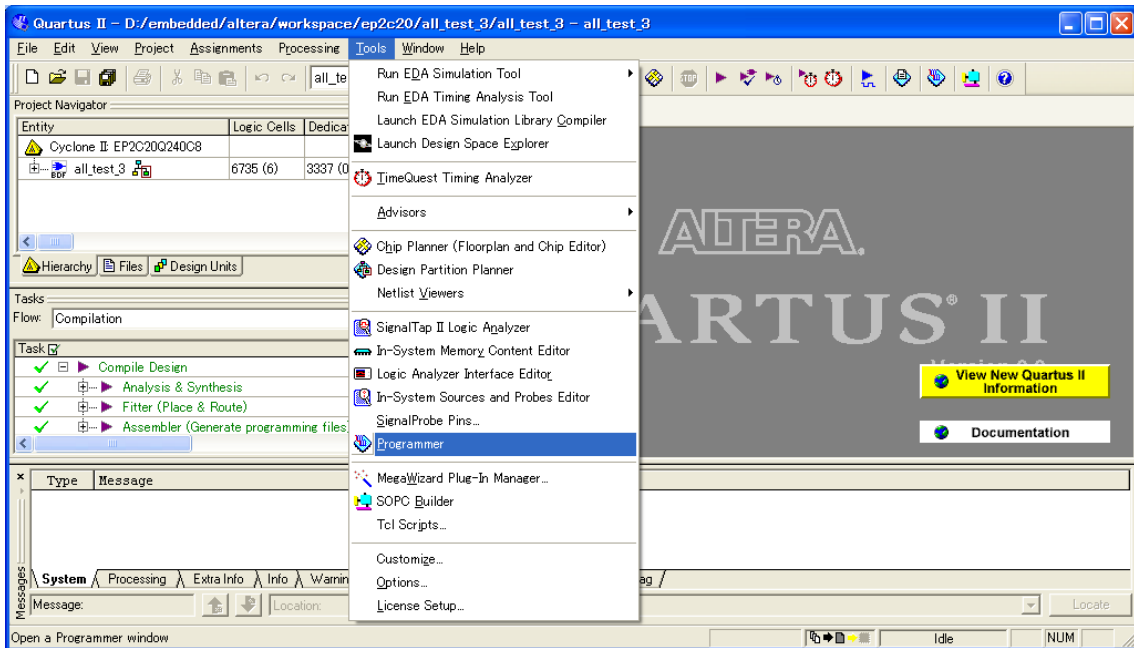
インストール中。



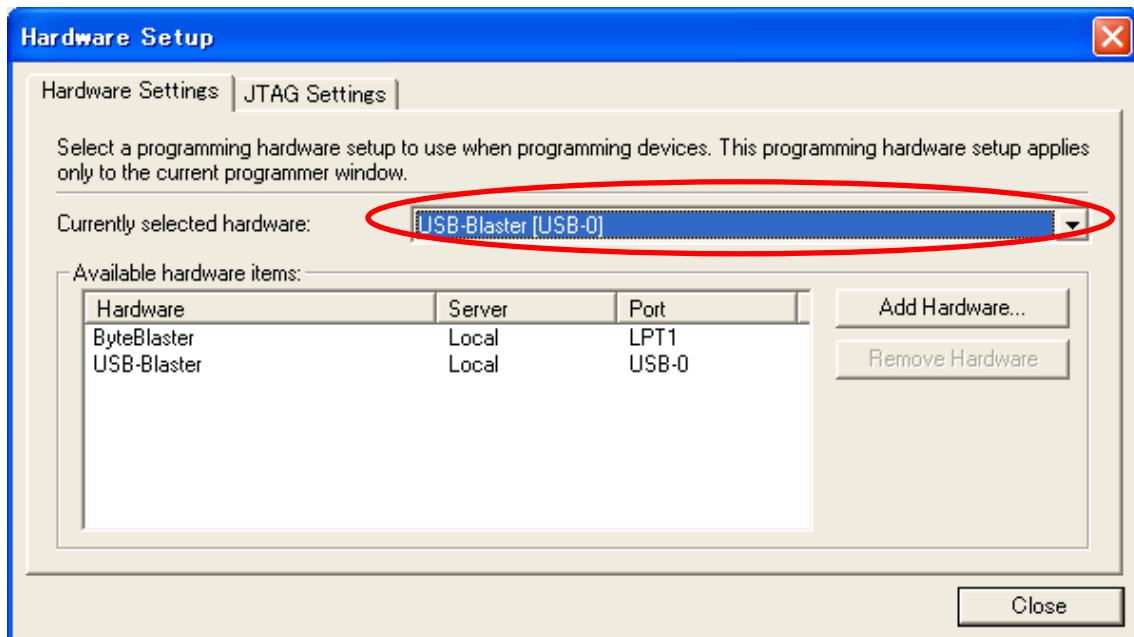
インストール完了します。

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。



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。



「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が、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

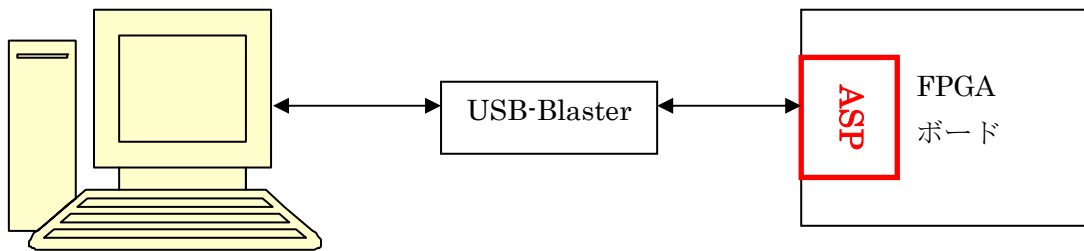
など、これまでの作業に問題がないか再度チェックをしてください。

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。

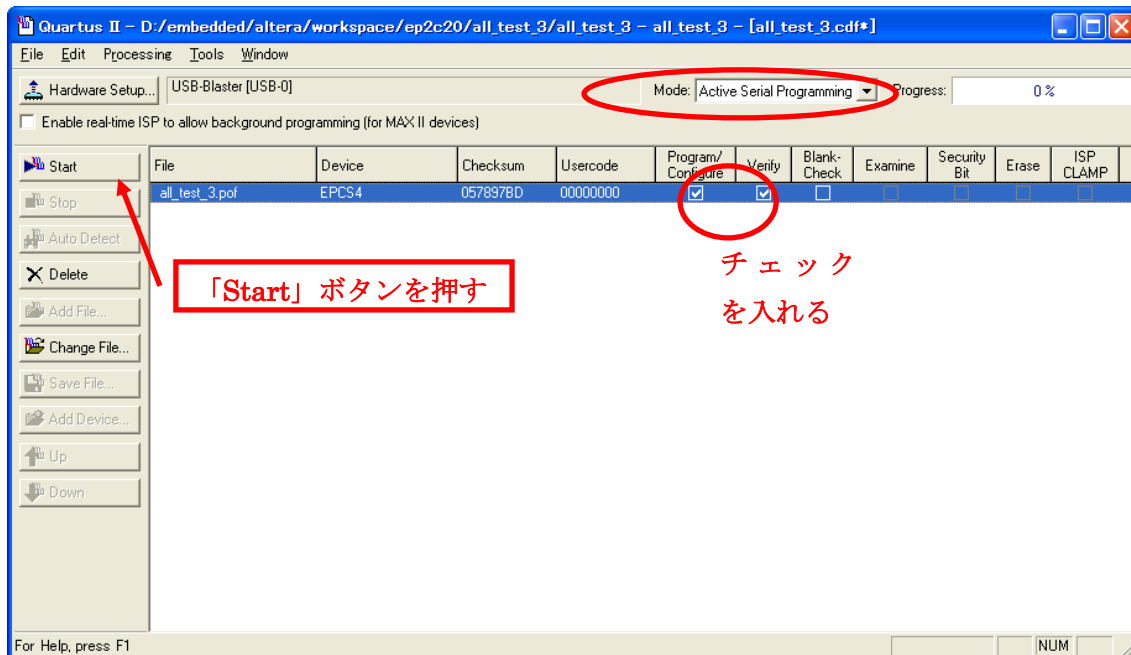
ブザーがピピと鳴って、FPGA ボード上の LED が点滅して、拡張ボード上の 7SegLED がカウンタ表示しているのを確認してください。

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：



まず、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。

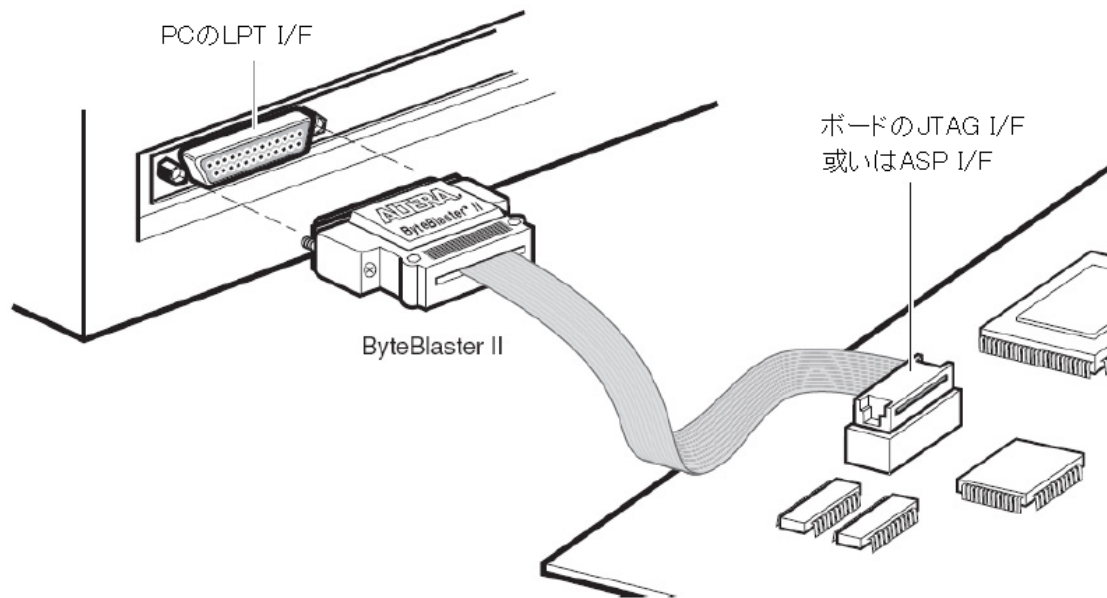


「Mode」に[Active Serial Programming]を選択します。「Add File」ボタンを押して、書き込みファイル*.pof を添加します。*.pof の右側にある Program/Configure の欄にチェックを入れて、「Start」ボタンを押します。プログレス・バーが 100%まで達すれば、書き込み成功です。

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？

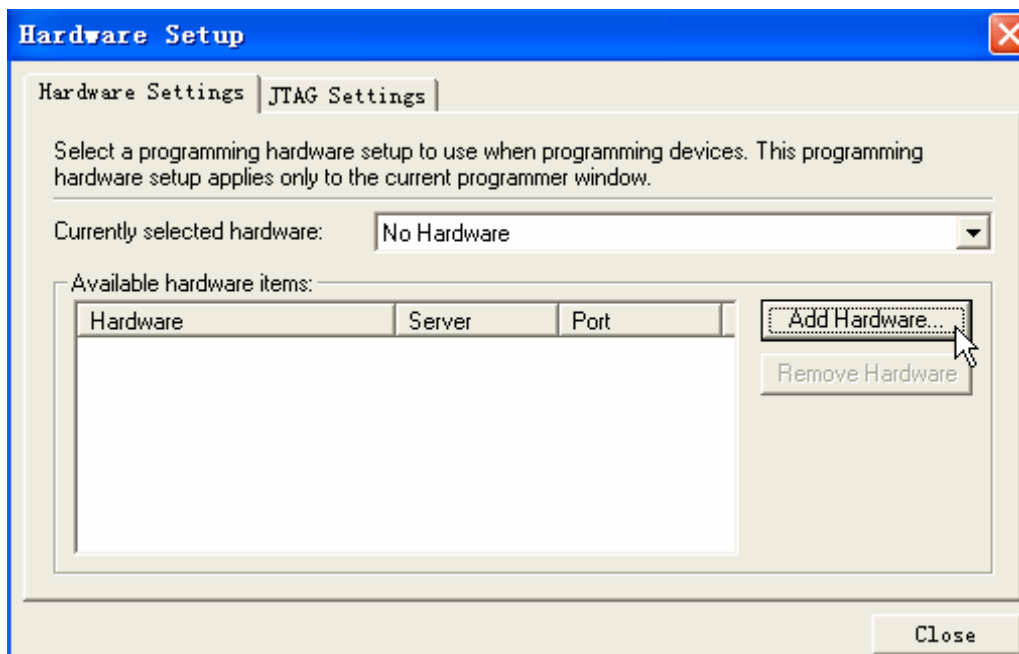
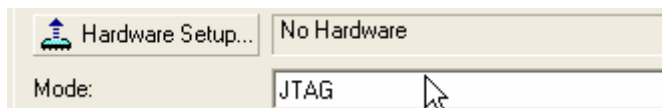
3.5 ByteBlasterIIの使い方

PC との接続：

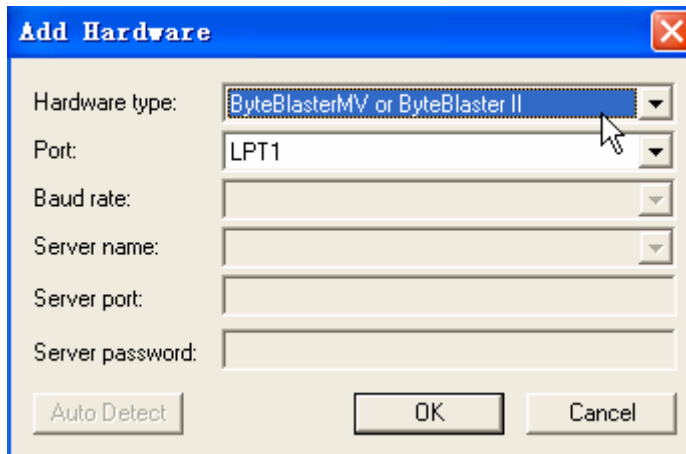


Quartus II の「Tools」メニューから「Programmer」を起動する。

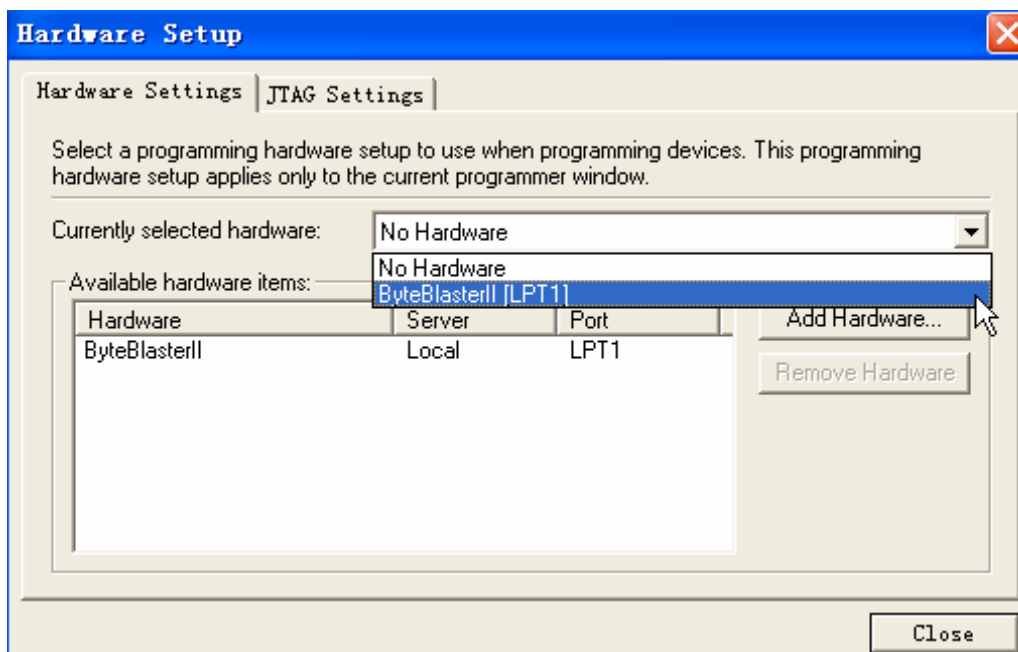
下記のように No Hardware の状態であれば、“Hardware Setup…”をクリックする。



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



OK をクリックする。



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



次の手順は USB Blaster と共通です。

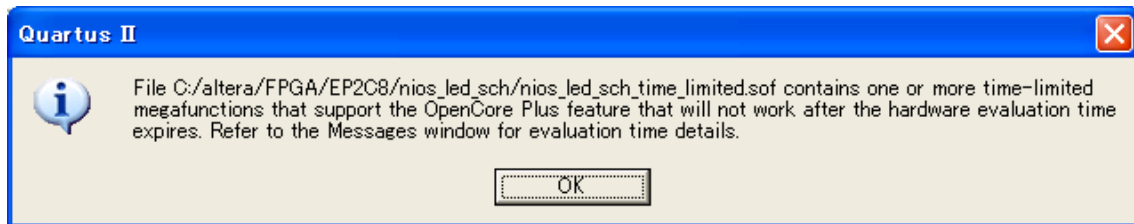
3.6 NIOS IIプロセッサの初体験

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。

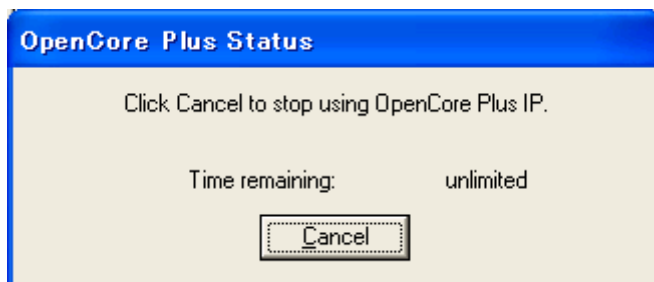
これらの中に、名前が all_test_3.qpf ファイルがあります。これをダブル・クリックすると、Quartus II が起動して、プロジェクトが開きます。

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。

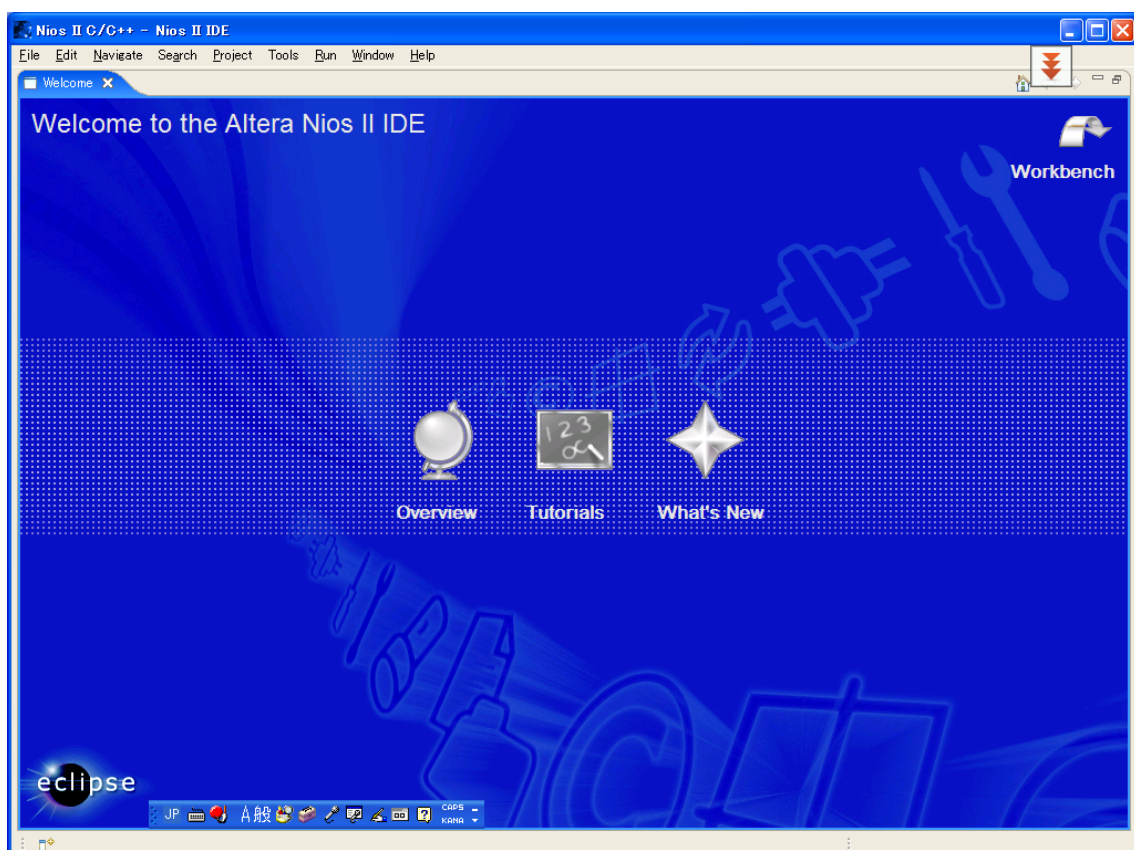
書き込み時、このような情報が出てきます。正式製品なら、アルテラ社からライセンスが必要です。評価の場合は、そのまま「OK」ボタンを押します。



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。

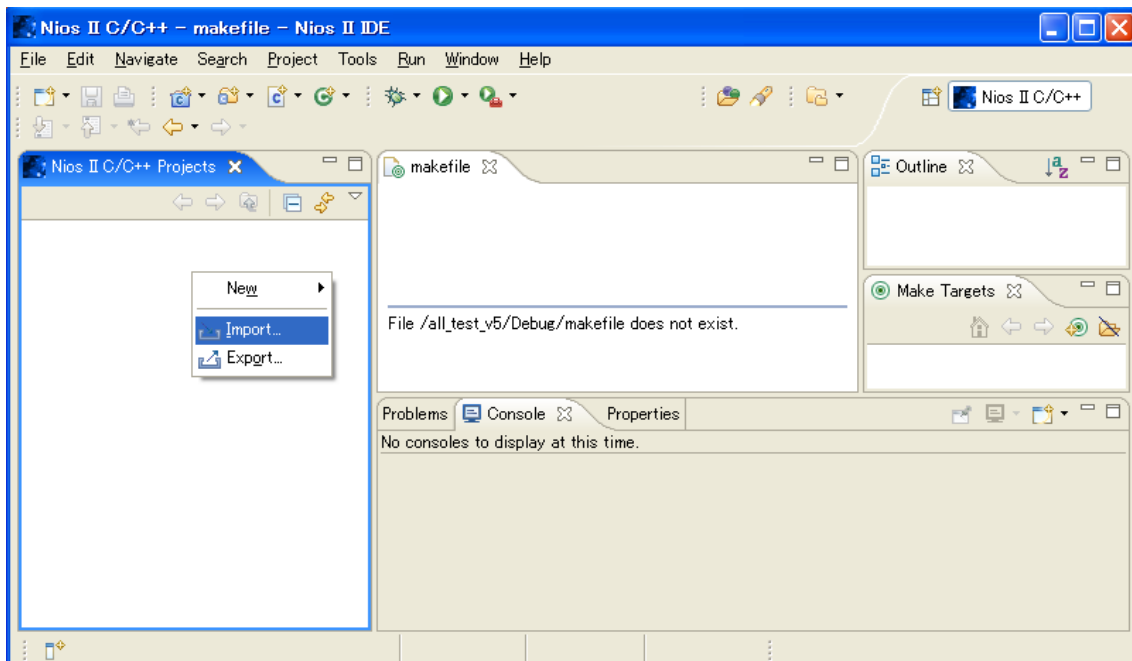


Windows の「スタート」→「すべてのプログラム」→「Altera」→「NIOS II EDS 8.1」から NIOS II 8.1 IDE が起動します。

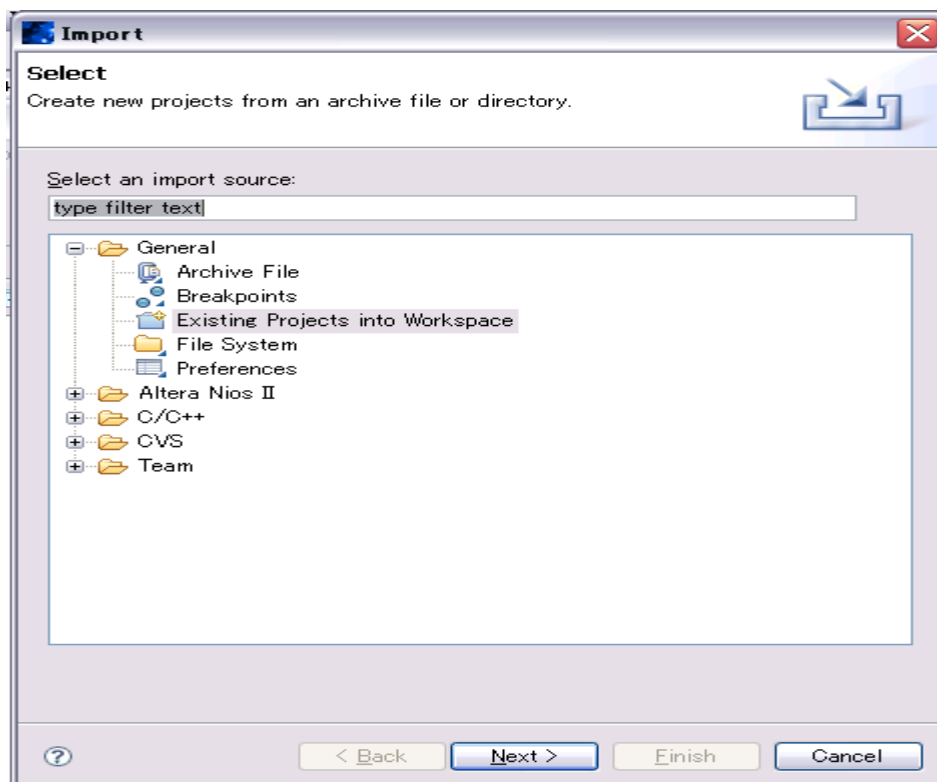


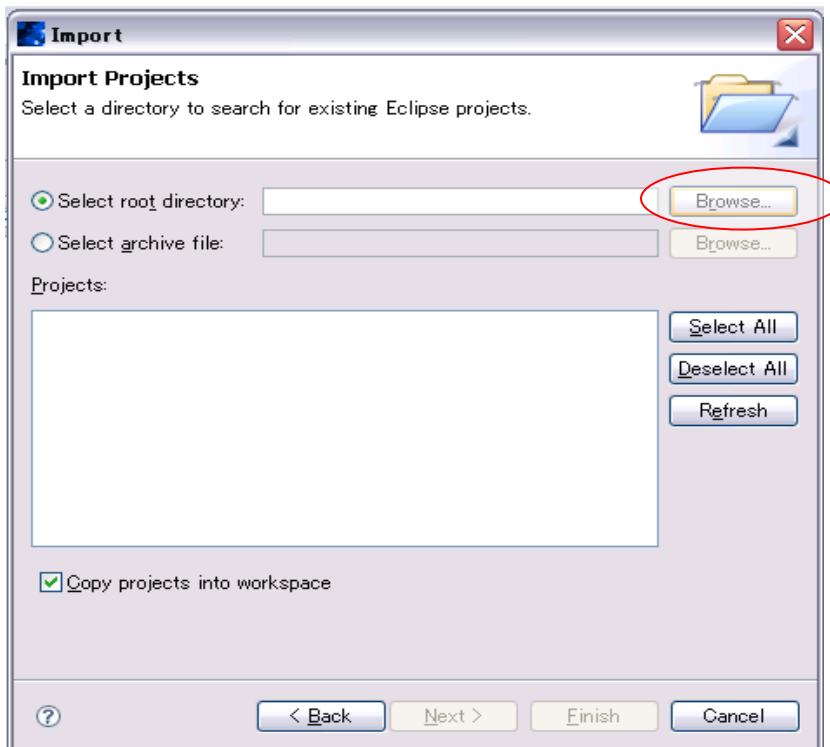
NIOS II IDE の初起動の画面です。

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

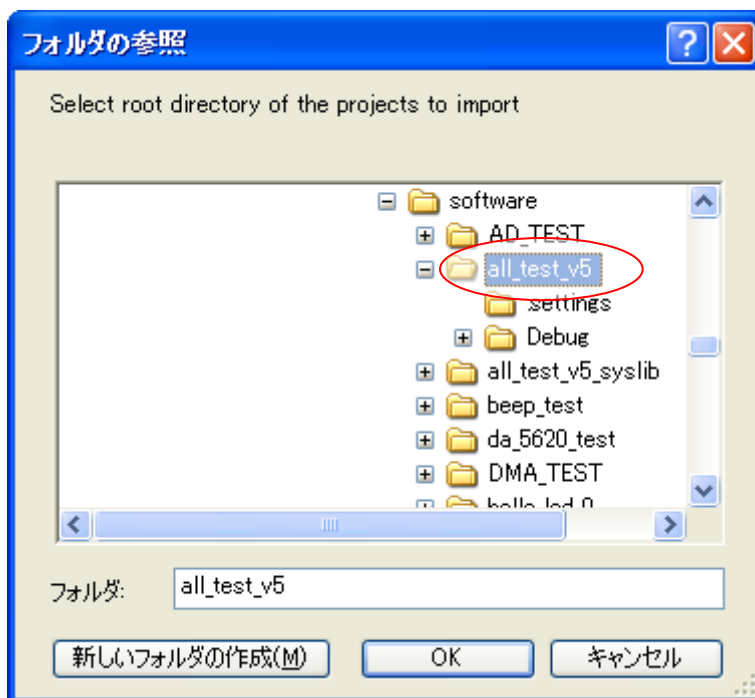


「Import」画面で「Existing Projects into Workspace」を選択して、「Next」ボタンを押します。

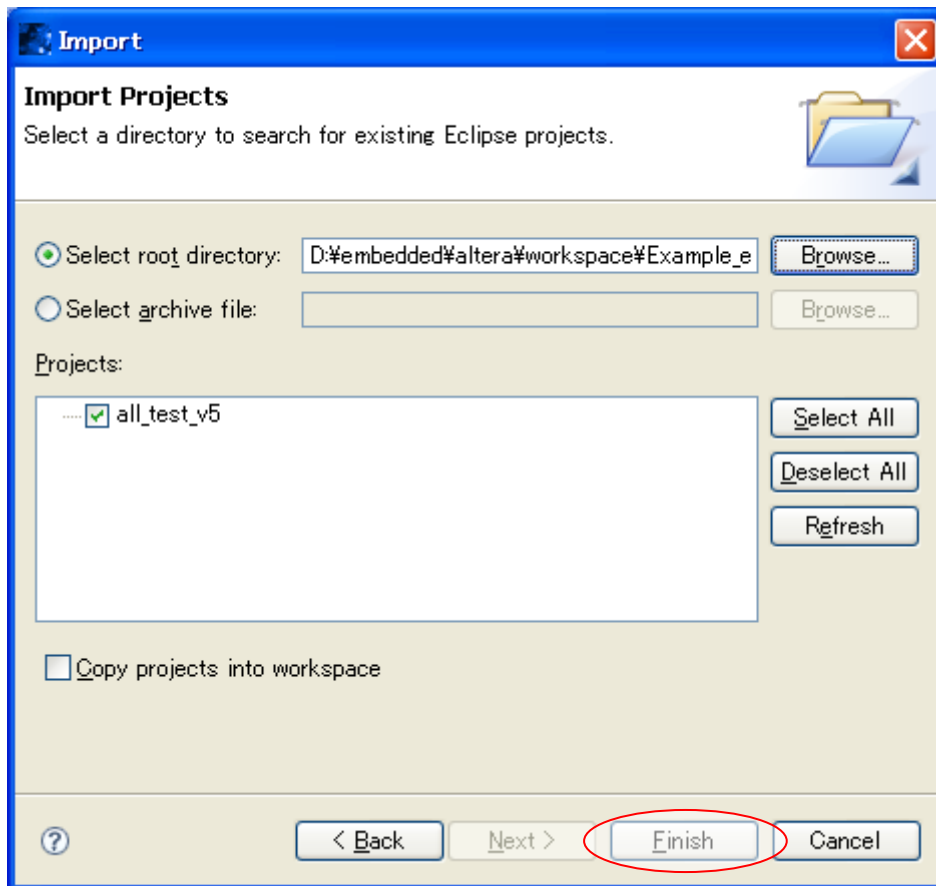




「Browse」ボタンを押してプロジェクト保存しているフォルダを選択します。

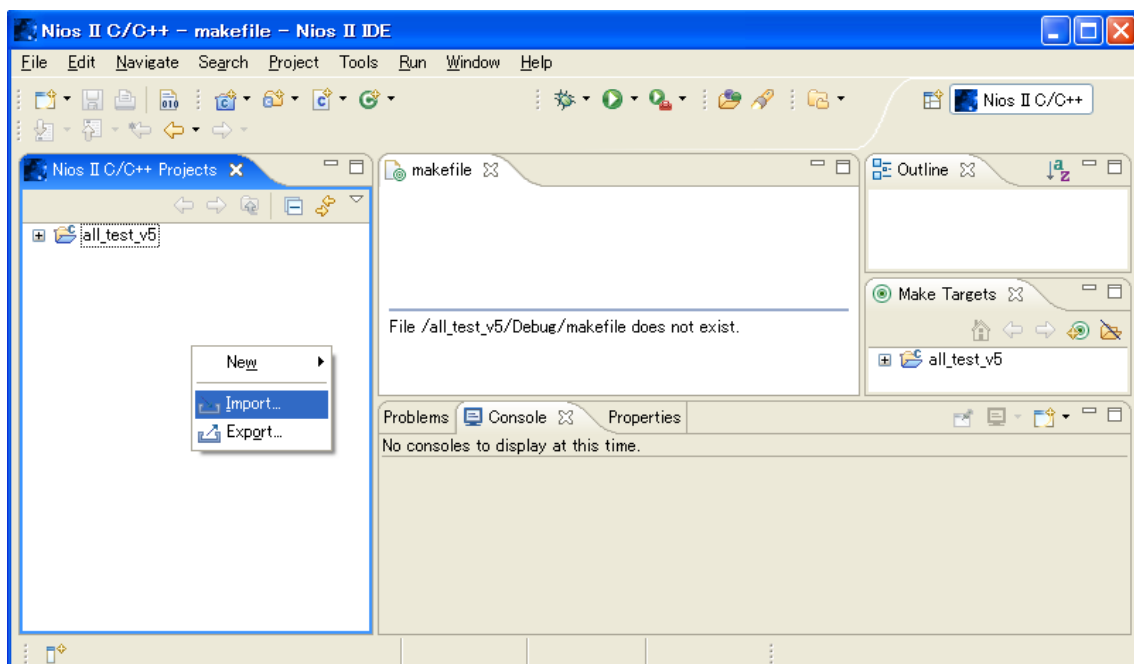


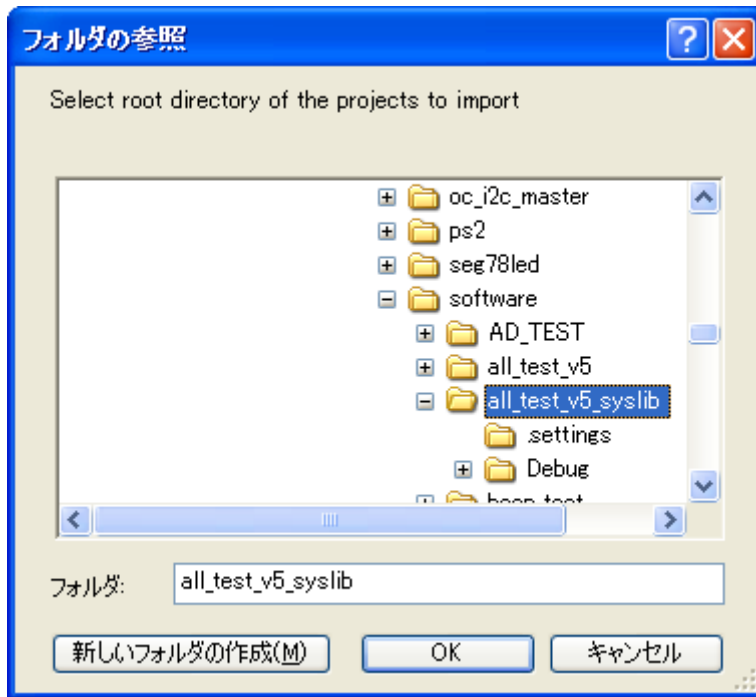
「OK」ボタンを押します。



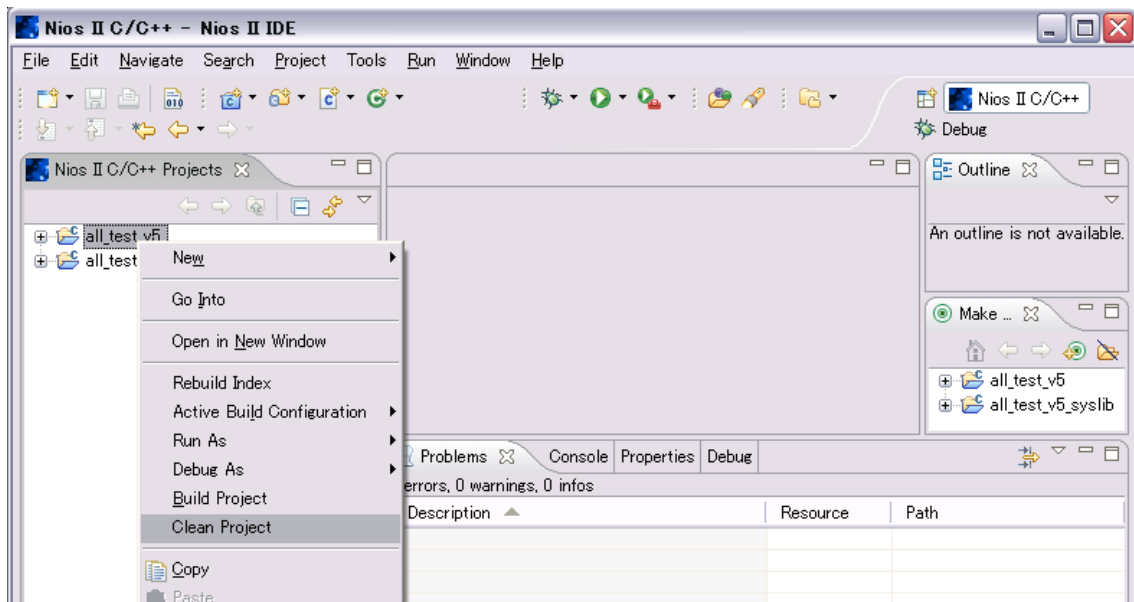
そのまま、「Finish」ボタンを押します。

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。

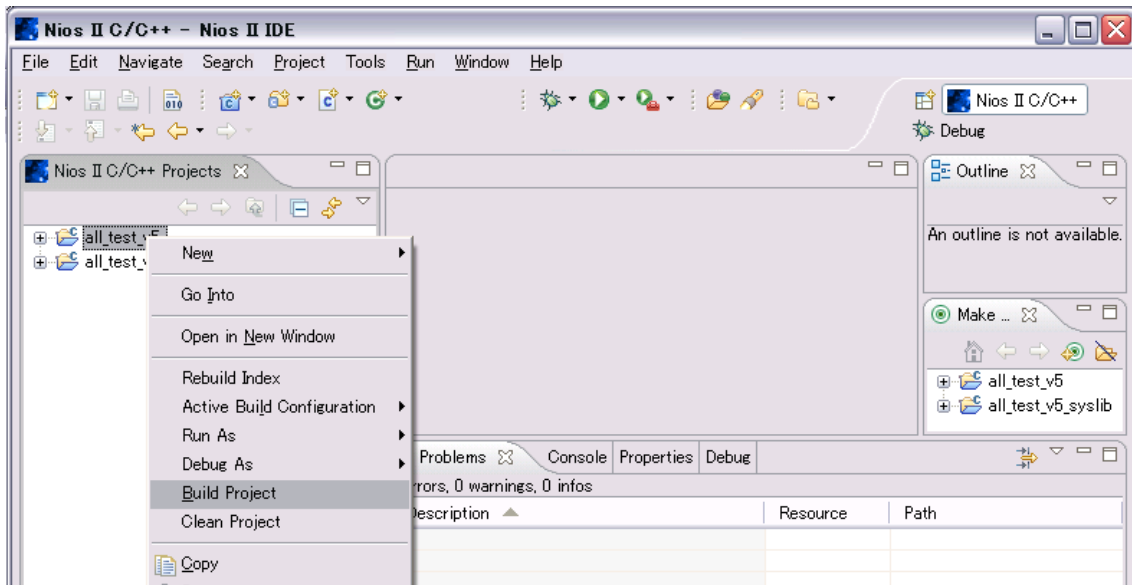




このままビルドすると旨く行かない場合がありますので、一回 Clean Project を実行します。



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。



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。

