



簡単なプロセッサを組み込んだ FPGA デザインの作成

概要

Tutorial
TU0118 (v1.0) January 29, 2004

このチュートリアルでは、FPGA にプロセッサを組み込んだデザインを作成する方法（FPGA やエンベデッドプロジェクトの作成、C ファイルの作成、プロセッサの設定やコンパイラオプション、ならびに FPGA デバイスへの設定やプログラムの方法）を説明します。

このチュートリアルでは、回路図や、NanoBoard に付属のドータボード Altera Cyclone EP1C12Q240CB チップにダウンロードされる C ソースファイルを含む、エンベデッドプロジェクトを作成します。ビットカウンタの結果は、NanoBoard の LED に表示されます。

PC のパラレルポートに NanoBoard を接続し、スイッチを ON にしてボードの電源を入れます。Altera から Quartus ツール（Web 版）をインストールしていることを確認して下さい。これらのツールは、Altera のウェブサイト (www.Altera.com) からダウンロードすることができます。

FPGA プロジェクトの作成

最初に FPGA プロジェクトを作成します。

1. **File** » **New** » **FPGA Project**（または、**Workspace** ボタンをクリックして **Add New Project** » **FPGA project** を選択します）を選択して、新規の FPGA プロジェクトを作成します。
2. **File** » **Save Project** を選択、または **Projects** パネル内の新規プロジェクト名を右クリックし **Save Project As** を選択して、`Simple_Processor.PrjFpg` という名称でこのプロジェクトを保存します。
3. **File** » **New** » **Schematic**（または、**Project** ボタンをクリックし **Add New to Project** » **Schematic** を選択）を選択して、FPGA プロジェクトに新規の回路図ドキュメントを追加します。回路図を保存します。
4. 図 1 の様に、回路図シートに次の（表 1 にリスト表示された）コンポーネントを配置します。**Place** » **Part** を選択、または、**Wiring** ツールバーの **Place Part** ボタンをクリックします。これらの各コンポーネントは、デフォルトの一般的な FPGA 統合ライブラリから利用できます。その為、このチュートリアルでは、ライブラリを追加する必要はありません。ライブラリの使用についての詳細は、チュートリアル「[統合ライブラリの作成](#)」を参照して下さい。

警告: ファイル名やプロジェクト名には、スペースやダッシュ(-)を使用しないで下さい。必要ならばアンダースコアを使用して下さい。



簡単なプロセッサを組み込んだFPGA デザインの作成

表 1. Johnson Counter 回路図で使用したコンポーネント

コンポーネント名	FPGA 統合ライブラリ
TSK51A_D (プロセッサ)	FPGA Processors.IntLib
RAMS_8x1K	FPGA Memories.IntLib
CLOCK_BOARD	FPGA NanoBoard Port-Plugin.IntLib
TEST_BUTTON	FPGA NanoBoard Port-Plugin.IntLib
LED	FPGA NanoBoard Port-Plugin.IntLib
NEXUS_JTAG_CONNECTOR	FPGA NanoBoard Port-Plugin.IntLib
NEXUS_JTAG_PORT	FPGA Generic.IntLib
OR2N1S	FPGA Generic.IntLib
FPGA_STARTUPx	FPGA Peripherals.IntLib

Place Part ダイアログの Lib Ref の項目でコンポーネント名 (例えば, TSK51A_D) を入力し、**OK** を押します。シート上でクリックしシンボルを配置します。他のパートを選択し、配置するには、右クリックします。パートの配置を終了するには、**Place Part** ダイアログが開いている時に **Cancel** を選択します。

コンポーネントが、まだカーソル上にあり、配置されていない状態の時に

- **X** を押すとコンポーネントが反転します。
- **SPACEBAR** を押すと、90度ずつコンポーネントが回転します。

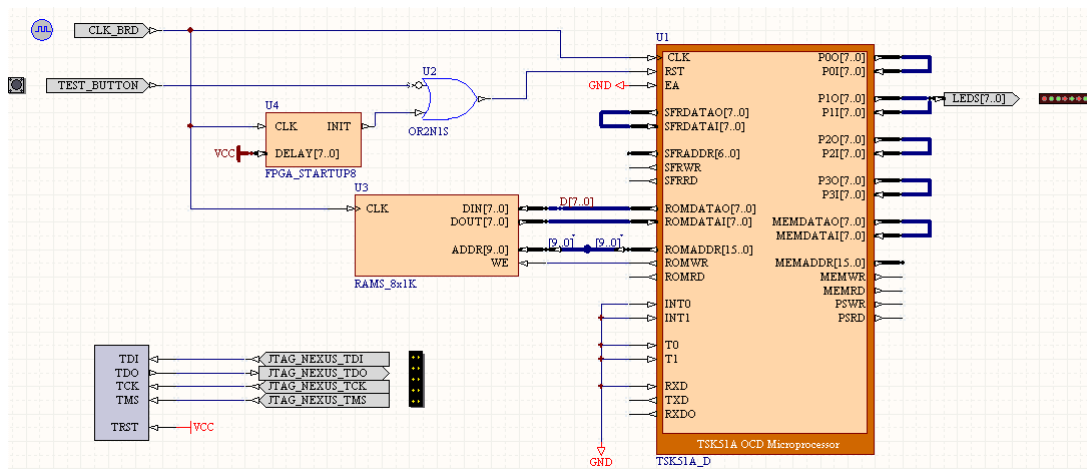

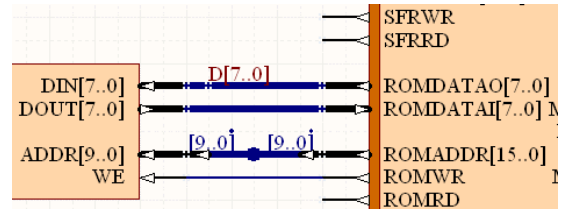


図 1. FPGA デザインの回路図シート

5. **Place » Wire** と **Place » Bus** コマンドを使用して回路図デザインの接続を行います。

 ワイヤやバスの配置についての詳細は、チュートリアル「PCB デザイン入門」を参照して下さい。



6. バスを使用して、RAMS_8x1K の ADDR[9..0] を TSK51A-D シンボルの ROMADDR[15..0] に接続します。バスを連結するには、FPGA Generic.IntLib から JB のシンボルを配置します。**Spacebar** キーを使用して、回転させることができます。文字編集モードを使用して、接合部の両側のパラメータ文字を [9..0] に変更します。バス接合部の上のパラメータ文字を移動します。（移動した場合、文字の下にドットが表示されます。）
 7. **Place » Net Label** コマンドを使用して、適切なネットにネットラベルを配置します。対応するワイヤにラベルを配置する（ネットラベルが正しくネットに接触すると、カーソルが赤い十字に変わります）前に、**TAB** を押して **Net Label** ダイアログでネットラベル名を入力します。
 8. TSK51A_D プロセッサの EA ピンに GND ポートを配置します。**Place » Power Port** を選択し、配置する前に **TAB** を押して、**Power Port** ダイアログでパワーポート名(GND)を入力し、形状（例えば Arrow）を選択します。ピン INT0, INT1, T0, T1, RXD を接続し、更に GND ポートを接続します。
- パワーポートを回転するには、配置中にスペースバーを使用します。
9. Bar 形状を使用して、NEXUS_JTAG_PORT の TRST に VCC パワーポートを配置します。Wiring ツールバーの VCC アイコンをクリックします。FPGA_STATRTUP8 コンポーネントの DELAY[7..0]ピンに VCC Bus パワーポート(**Power Port** ダイアログのネット属性が VCCBUS[...]である必要があります)を配置します。
 10. **Tools » Annotate Quiet** コマンドを使用してデザインをアノテートします。**Confirm Designator Changes** ダイアログに次のメッセージが表示されます。“There are 3 designators requiring update. Proceed with changes?” **Yes** をクリックします。デジグネータは、例えば、U? から U1 に自動で変更されます。
 11. Sheet1.SchDoc が、プロジェクトと同じディレクトリにあることを確認して保存[ショートカット **Ctrl+S**]します。

エンベデッドプロジェクトの作成

FPGA チップにダウンロードするソフトウェアのソースファイルを含む、エンベデッドプロジェクトを作成することができます。

1. **File » New » Embedded Project** を選択（または、**Workspace** ボタンをクリックし **Add New Project » Embedded Project** を選択）して、新規のエンベデッドプロジェクトを作成します。
2. **File » Save Project** を選択して、このプロジェクトを保存します。または、**Projects** パネル内の新規プロジェクト名を右クリックして **Save Project** を選択します。ファイル名にスペースは使用しないで下さい（例えば、Embedded_Project1.PrjEmb）。FPGA プロジェクトと回路図ファイルを同じディレクトリに保存します。
3. **Projects** パネル内のエンベデッド・プロジェクト名を右クリックして **Add New to Project » C File** を選択し、新規の C ファイルを作成します。または、**Project** ボタンをクリックし **Add New to Project » C file** を選択します。

簡単なプロセッサを組み込んだFPGA デザインの作成

- 新規のCファイルに、次のCコードを入力します。

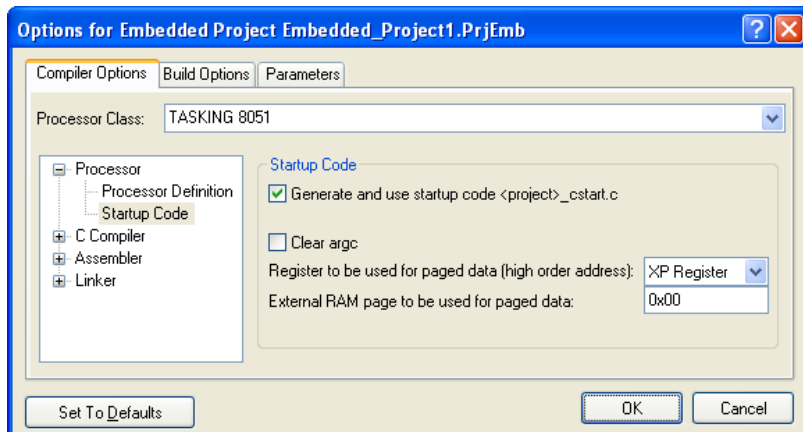
```
void main( void )
{
    unsigned char x = 0;
    unsigned short i;
    for (;;)
    {
        P1 = x++;
        for ( i = 0; i < 0xFFFF; i++ )
        {
            __asm( "nop" );
        }
    }
}
```

- 他のプロジェクトファイルと同じディレクトリにコードを保存します。このチュートリアルでは、デフォルト名 (Source1.C) を適用します。

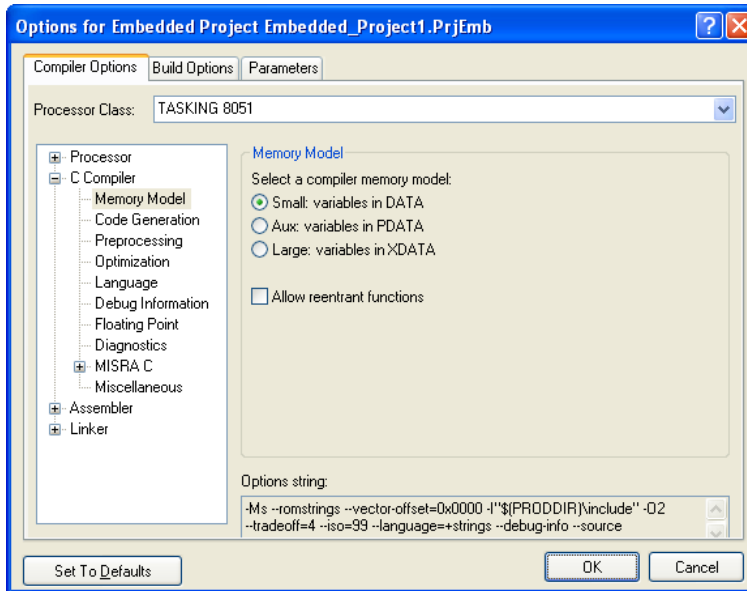
エンベッドソフトウェアのプロジェクトオプション設定

次にプロセッサ開始コードを設定し、メモリモデルとコード生成オプションなどのCコンパイラオプションを設定します。

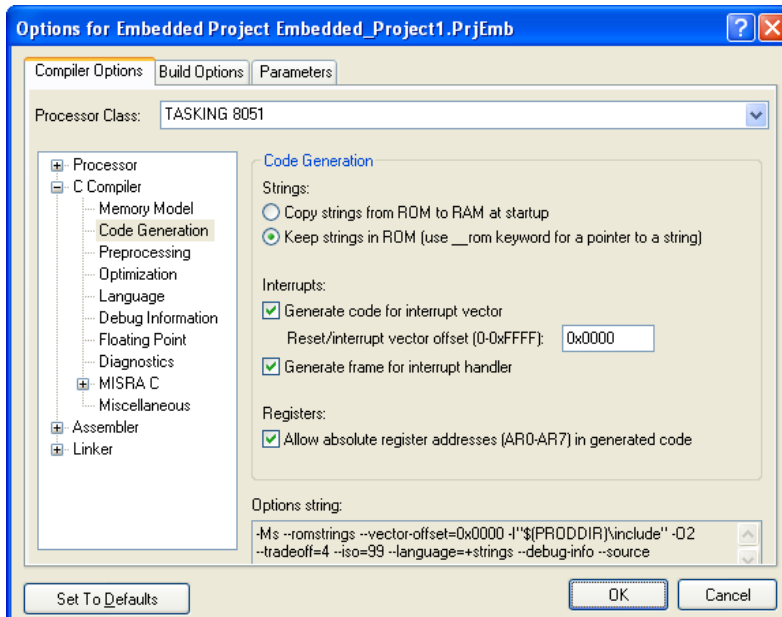
- Projects** パネル内のエンベッドプロジェクトの名称を右クリックし、**Project Options** を選択します。*Options for Project Embedded* ダイアログが開きます。
- Processor** をダブルクリックし **Startup Code** をクリックして、プロジェクトに開始コードを追加します。**Generate and use startup code <project>_cstart.c** が選択されていることを確認して下さい。



- C Compiler** をダブルクリックして **Memory Model** を選択し、**Small: variables in DATA** を選択してメモリモデルを設定します。**Allow reentrant functions** が選択されていないことを確認して下さい。



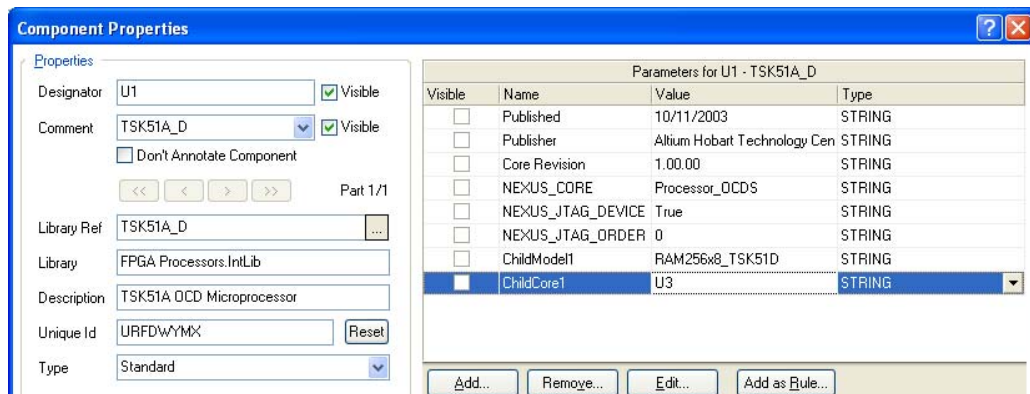
4. C Compiler オプションで、**Code Generation** をクリックし Strings を **Keep strings in ROM** に設定します。Interrupts and Registers オプションが選択されていることを確認して下さい。OK をクリックします。



プロセッサ属性の設定

RAM やソフトウェアがドータボードの FPGA チップにダウンロードされる時に、どのサブデザイン（エンベデッド・プロジェクト）を使用するかを回路図上のプロセッサに指示する必要があります。

1. ドキュメントタブをクリックして、Sheet1.SchDoc に戻ります。
2. TSK51A_D プロセッサ (U1) をダブルクリックし **Component Properties** ダイアログを開きます。
3. Parameters の項目内の **ChildCore1** パラメータの **Value** の項目に RAM のデジグネータ（回路図を参照して下さい。このチュートリアルでは U3 です。）を入力します。



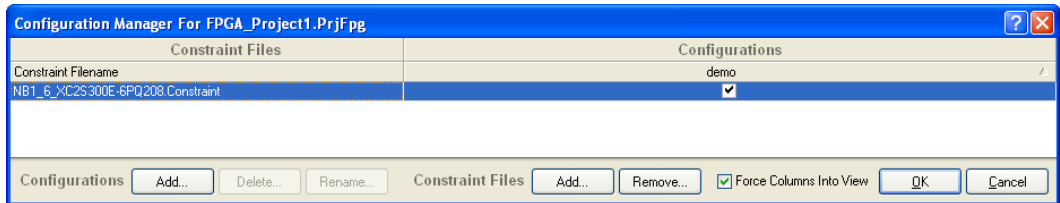
4. プロセッサコンポーネントにエンベデッドプロジェクトをリンクさせる為に、**Projects** パネル内の **Structure Editor** ボタンをクリックして、プロジェクト構成画面を表示します。ハードウェアプロジェクトがコンパイルされ、その構成がツリー表示されることを確認して下さい。**Projects** パネルの下部の項目に **Valid Sub-Projects** リストが表示されます。**Valid Sub-Projects** リストから作成したソフトウェアプロジェクト（例えば、Embedded_Project1.PrjEmb）を選択します。そのアイコンをクリックして、パネルの上部のプロセッサコンポーネント(TSK51A_D)アイコンへ、それをドラッグアンドドロップします。ターゲットが正しくハイライトされているか注意して下さい。リンクが確立されると、整合性を取る為にストラクチャが再コンパイルされます。
5. 回路図と FPGA プロジェクトを保存します。

デザインを FPGA デバイスへ設定

デザインでどの FPGA チップ（例えば、NanoBoard 用のドータボードの Altera Cyclone EP1C12Q240C6 チップ）を使用するか指定する必要があります。これを行う為に、コンフィグレーションとコンストレインファイルを追加します。Constraint ファイルは、デバイスを指定しピン番号を決定します。

1. 新規の Configuration を作成するには、**Projects** パネル内の FPGA Project 名を右クリックし、**Configuration Manager** を選択します。または、メニューから **Project » Configuration Manager** を選択します。
2. **Configuration Manager for project** ダイアログが表示されます。ダイアログの **Configurations** の項目で **Add** ボタンをクリックし、**New Configuration** ダイアログで Configuration 名（例えば、demo）を入力します。そして **OK** をクリックします。

3. **Constraints** の項目で **Add** ボタンをクリックし、**Choose Constraint files to add to Project** ダイアログで NB1_6_EP1C12Q240.Constraint を選択してコンフィグレーションに **Constraints** ファイルを追加します。コンストレインファイルは、Altium2004\Library\FPGA フォルダに保存されています。**Open** をクリックします。
4. **Configuration Manager** ダイアログに戻り **Configuration** チェックボックスを選択し、**OK** をクリックします。



5. Settings という名称のフォルダがプロジェクトに追加され、Constraints Files フォルダに Constraint ファイルが表示されます。
6. FPGA プロジェクトファイルを保存します。


Devices ビューを使用して FPGA をプログラム

Devices ビュー (View » Devices) では、FPGA にプログラムを転送する為に必要なワークフロー (左から右へ) を通してプログラムを実行することができます。この画面で以下のことが実行できます。

- プロジェクトのコンパイル (また、エラーの確認)
- 論理合成 (EDIF ネットリストを作成)
- ビルド (EDIF ファイルを翻訳、FPGA にデザインをマップ、FPGA の配置配線、タイミング分析、ビットファイルの作成)
- FPGA のプログラム (ドータボードの FPGA チップ (例えば、Altera Cyclone) へビットファイルをダウンロード)

このワークフローが完了した時、NanoBoard 上の DIP スイッチを切り替えてプログラムを実行することができます。FPGA にデザインをダウンロードするには、

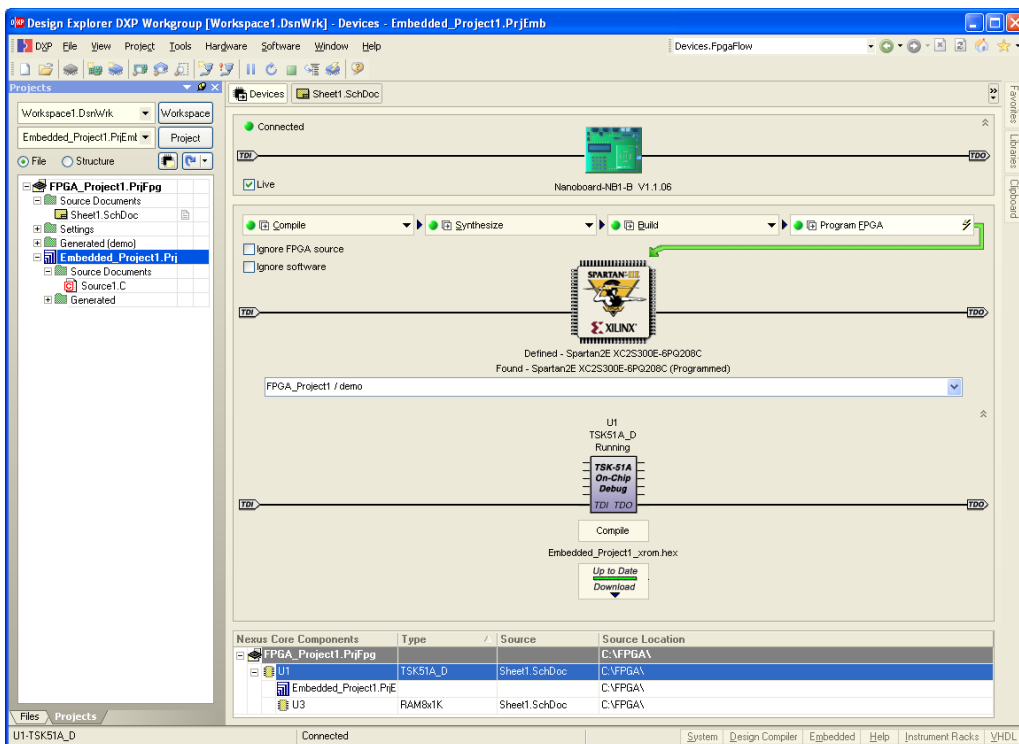
1. **View » Devices** を選択して、**Devices** ビューを開きます。
2. NanoBoard が正しく接続されていることを確認して、スイッチを ON にします。Devices ビューで、**Live** ボタンをクリックし **Connected** インジケータが緑色で表示されていることを確認します。
3. Devices ビューで **Compile** をクリックします。コンパイルが成功すれば、赤色インジケータが緑色に変わります。もし、エラーメッセージが **Messages** パネルに表示される場合、回路図とエンベデッドソースコードに戻り、エラーを修正します。そして、ファイルを保存し再コンパイルを行います。
4. **Synthesize** をクリックします。論理合成が成功すれば、Generated [config_name] というフォルダの中に EDIF, VHDL, 論理合成ログファイルが作成されます。もし、論理合成が完了しない場合は、**Messages** パネルで

段階ボタンの左側にある矢印アイコン  をクリックして、現在の段階までのワークフローの全ての段階を実行することができます。例えば、**Program FPGA** ボタン上のこのアイコンをクリックすると、最初の以前の段階から全て実行します。

簡単なプロセッサを組み込んだFPGA デザインの作成

エラーを確認して下さい。この例で使用した demo という名称のコンフィグレーションは、Devices ビューの Altera Cyclone アイコンの下に表示されます。

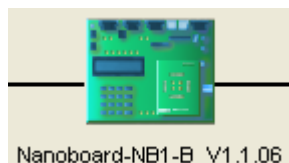
5. **Build** をクリックします。これによっていくつかのプロセスが実行され、FPGA にダウンロードできるビットファイルを最後に作成します。全てのプロセスが完了すると、プロセス名の横にあるボタンが緑色に変わります。必要な全プロセスが完了すると、**Build** ボタンは緑色に変わります。
6. **Program FPGA** をクリックして、ドータボードの Cyclone チップにビットファイルをダウンロードします。プロセスフローを見ると、最後に JTAG バスを通して FPGA のプログラムが行われます。
7. Program FPGA のプロセスが完了すると、LED がビットカウンタとしてフラッシュ表示します。



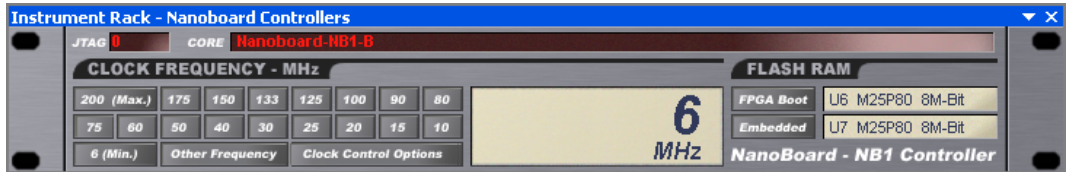
クロック周波数の設定

NanoBoard 上のカウンタスピード（クロック周波数）を遅くしたり、速くすることができます。カウンタ表示を遅らせてみます。

1. Devices ビューで NanoBoard アイコンをダブルクリックします。



Instrument Rack – NanoBoard Controllers が表示されます。デフォルトのクロック周波数は、50MHz です。



2. 別のクロック周波数を選択します（例えば、**6(Min)**ボタンをクリックして、一番遅い(6 MHz)のクロック周波数を選択）。
3. LED 表示の変移がそれに応じて遅くなります。

更新履歴

Date	Version No.	Revision
29-Jan-2004	1.0	New product release

Software, hardware, documentation and related materials:

Copyright © 2004 Altium Limited.

Copyright © 2005 Altium Japan.

All rights reserved. You are permitted to print this document provided that (1) the use of such is for personal use only and will not be copied or posted on any network computer or broadcast in any media, and (2) no modifications of the document is made. Unauthorized duplication, in whole or part, of this document by any means, mechanical or electronic, including translation into another language, except for brief excerpts in published reviews, is prohibited without the express written permission of Altium Limited. Unauthorized duplication of this work may also be prohibited by local statute. Violators may be subject to both criminal and civil penalties, including fines and/or imprisonment. Altium, CAMtastic, Design Explorer, DXP, LiveDesign, NanoBoard, NanoTalk, Nexar, CircuitStudio, nVisage, P-CAD, Protel, Situs, TASKING, and Topological Autorouting and their respective logos are trademarks or registered trademarks of Altium Limited or its subsidiaries. All other registered or unregistered trademarks referenced herein are the property of their respective owners and no trademark rights to the same are claimed.