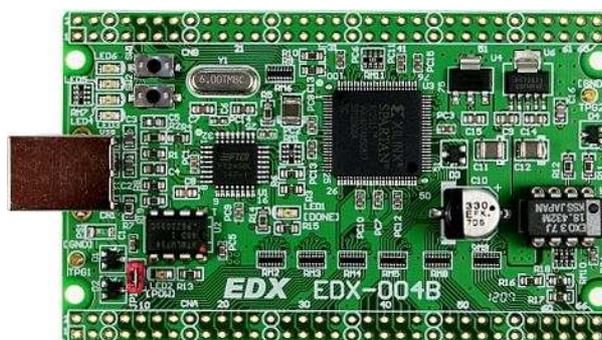


FPGA トレーナ
EDX-004
ユーザーズマニュアル
第 4 版



ヒューマンデータ

目次

はじめに	1
ご注意	1
1. 製品の内容について	2
2. 仕様	2
3. 各部の名称	3
4. ブロック図	3
5. 開発環境	4
6. 電源入力	4
7. ダウンロードケーブル	4
8. クロック	4
9. 押しボタンスイッチ	5
10. 赤色 LED	5
11. FPGA コンフィグレーション	5
12. FPGA ピンアサイン表	5
13. ISE による開発手順	8
14. プロジェクトの新規作成	8
15. VHDL による AND 回路	10
16. 新規ソースの追加	10
17. テストベンチの作成とシミュレーション	13
18. UCF ファイルの作成	16
19. BIT ファイルの作成	17
20. 専用ソフトウェアによる FPGA コンフィグレーション	18
21. 動作の確認	19
22. 製品外形図	20
23. 付属 CD-ROM の内容	20
24. 付属資料	20

はじめに

この度は、FPGA トレーナ EDX-004 をお買い上げいただきまして、誠にありがとうございます。
EDX-004 は、XILINX 社の FPGA である Spartan-II (XC2S15-5VQ100C : 15,000 ゲート) を実装した
トレーナです。汎用 LED、押しボタンスイッチ、クロックモジュール、FPGA コンフィグレーション
に使用する USB インターフェースを実装しておりますので、快適に FPGA 設計を進めることができ
ます。

VHDL や Verilog-HDL による論理回路設計の習得に、ご活用ください。

ご注意

1. 本書の内容は、改良のため将来予告なしに変更することがありますので、ご了承願います。
2. 本書の内容については万全を記して作成しましたが、万一誤りなど、お気づきの点がございましたら、ご連絡をお願いいたします。
3. 本製品の運用の結果につきましては、2. 項にかかわらず当社は責任を負いかねますので、ご了承願います。
4. 本書に記載されている使用と異なる使用をされ、あるいは本書に記載されていない使用をされた場合の結果については、当社は責任を負いません。
5. 本書および、回路図、サンプル回路などを無断で複製、引用、配布することはお断りいたします。

1. 製品の内容について

本パッケージには、以下のものが含まれています。万一、不足などがございましたら、弊社宛にご連絡ください。

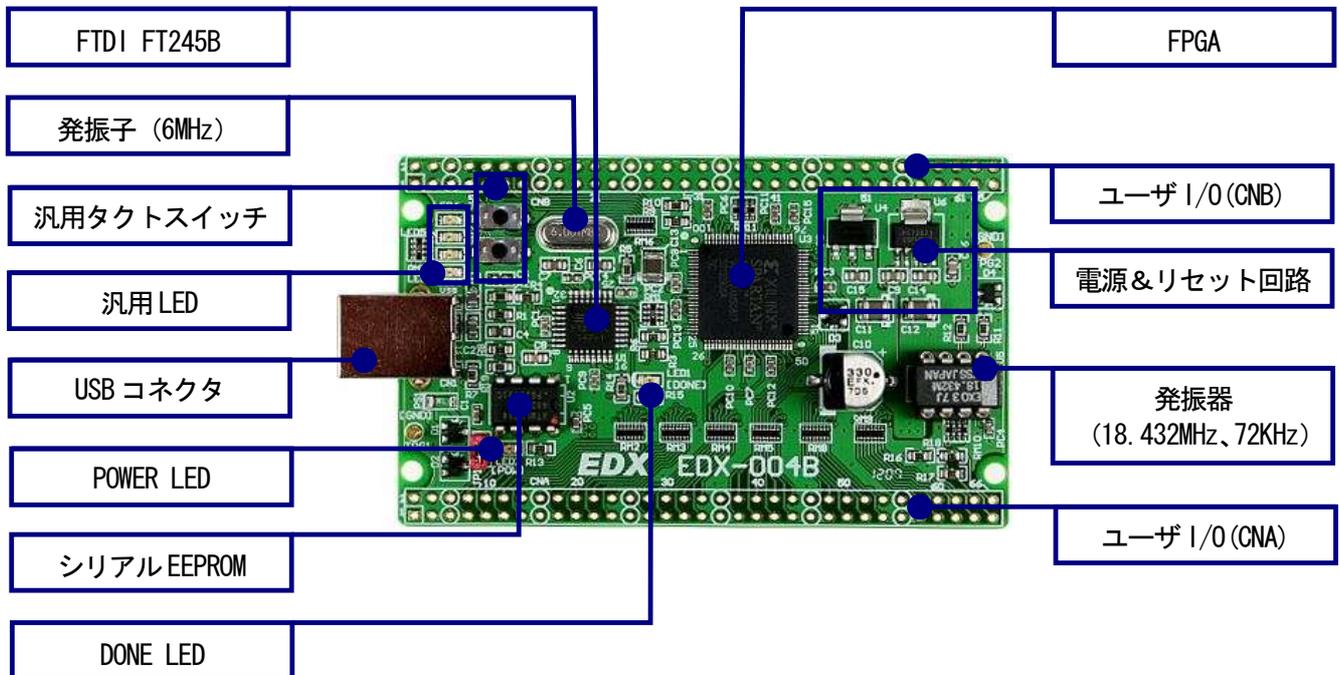
FPGA トレーナ EDX-004	1
付属品 (予備ジャンパなど)	1
マニュアル (本書)	1 *
ユーザー登録はがき	1 *
CD-ROM	1 *
USB ケーブル	1

* オーダー毎に各 1 部の場合があります。(ご要望により追加請求できます)

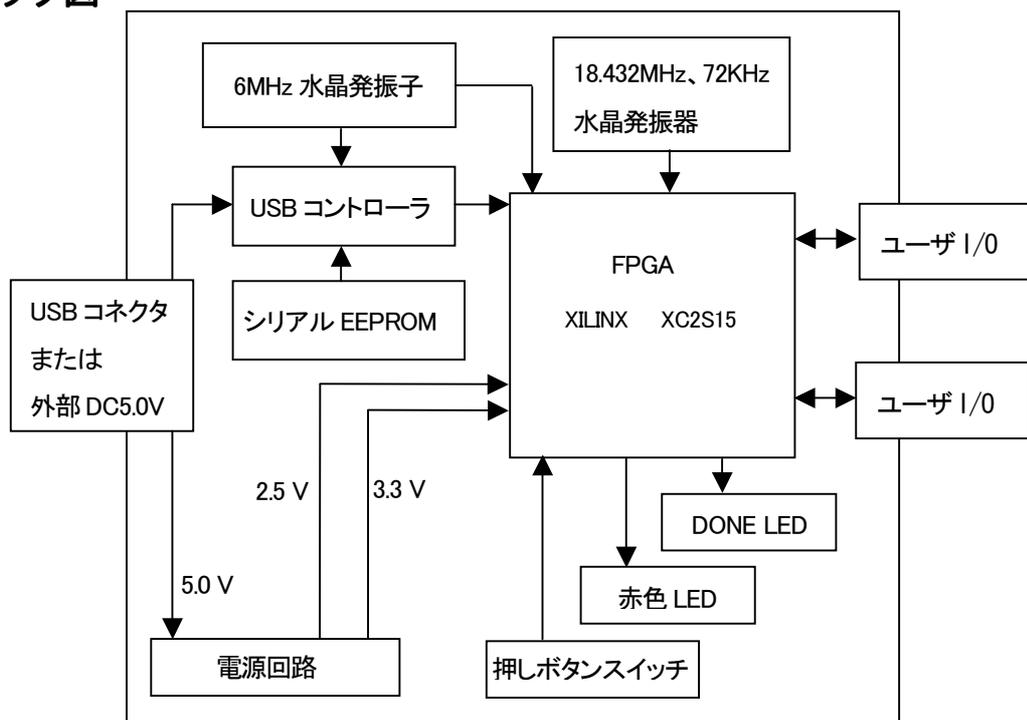
2. 仕様

製品型番	EDX-004
搭載 FPGA	XC2S15-5VQG100C
電源	USB から供給、または外部 DC5V (内部電源はオンボードレギュレータにより生成)
消費電流	N/A (詳細は FPGA データシートご参照)
外形寸法	86 × 54 [mm]
重量	約 25 [g]
ユーザ I/O	59 本
I/O コネクタ	66 ピンスルーホール 0.9[mmΦ] × 2 組 2.54mm ピッチ
プリント基板	ガラスエポキシ 4 層基板 1.6t
クロック	オンボード 18.432MHz、6MHz、72KHz、外部供給可能
リセット回路	内蔵 (200ms TYP)
ステータス LED	2 個 (POWER-LED , DONE-LED)
汎用 LED	4 個
汎用スイッチ	2 個
付属品	2 列 × 40 ピン、ジャンパヘッダ 2 本 (任意にカット可能)

3. 各部の名称



4. ブロック図



5. 開発環境

FPGA の内部回路設計には、回路図エディタやHDL 入力ツール、論理合成ツールが必要です。これらの開発ツールは、XILINX 社が無償配布する WebPack ISE にて可能です。

使用する際には、インターネットによるライセンス登録が必要となります。

【製品ご購入時とマニュアル作成時の開発ツールのバージョンの違いにより操作方法が異なる場合があります】

6. 電源入力

本ボードは、USB からの電源供給、または外部電源 DC5.0V を供給することにより動作します。内部に必要な、3.3V、2.5V はオンボードのレギュレータにより生成されます。

	USB より供給	外部より供給
JP1 ジャンパ	ショート(出荷時)	オープン



外部から電源を供給する場合 5.0V 電源は充分安定して、充分な余裕のあるものをご用意ください。

電源は、CNA、CNB から供給してください。適切な電源を供給してください。

いずれも 5.0V を超えることはできません。

詳しくは FPGA のデータシートや回路図などを参照してください。また電源の立ち上がりは単調増加である必要があります。良質の電源を使用するようにしてください。

7. ダウンロードケーブル

FPGA へのコンフィグレーションには、専用のダウンロードケーブルを必要としません。添付の USB ケーブルをご使用ください。

8. クロック

6MHz 水晶発振器より、USB コントローラと FPGA にクロックを供給します。

また、18.432MHz、72KHz 水晶発振器より FPGA にクロックを供給します。

9. 押しボタンスイッチ

押すと Low になります。このピンは FPGA に対して、入力となりますので、出力しないでください。押しボタンスイッチに対して **FPGA 側から出力するとショートします**ので注意してください。

10. 赤色 LED

赤色 LED は負論理です。Low にて点灯します。完全に消灯させるには、ハイインピーダンスとするか、オープンドレインにてオフとしてください。

11. FPGA コンフィグレーション

FTDI 社提供のダイレクトドライバをインストールすることで、USB 経由で FPGA をコンフィグレーションすることができます。コンフィグレーションが終了し、DONE 信号が H になると 赤色 LED (LED1) が点灯します。

12. FPGA ピンアサイン表

汎用 LED

LED	NET LABEL	FPGA ピン#
LED3	IOB0	P16
LED4	IOB1	P20
LED5	IOB2	P98
LED6	IOB3	P97

汎用スイッチ

スイッチ	NET LABEL	FPGA ピン#
SW1	IOB4	P96
SW2	IOB5	P95

CNA

NET LABEL	FPGA pin #	CNA pin #		FPGA pin #	NET LABEL
	3.3V 出力	1	2	3.3V 出力	
	5.0V 入力	3	4	5.0V 入力	
GND	GND	5	6	GND	GND
IOA0	P3	7	8	P4	IOA1
IOA2	P5	9	10	P6	IOA3
IOA4	P7	11	12	P8	IOA5
IOA6	P9	13	14	P10	IOA7
GND	GND	15	16	GND	GND
IOA8	P13	17	18	P15	IOA9
IOA10	P17	19	20	P18	IOA11
IOA12	P19	21	22	P21	IOA13
IOA14	P22	23	24	P31	IOA15
GND	GND	25	26	GND	GND
IOA16	P32	27	28	P40	IOA17
IOA18	P41	29	30	P43	IOA19
IOA20	P44	31	32	P45	IOA21
IOA22	P46	33	34	P47	IOA23
GND	GND	35	36	GND	GND
IOA24	P52	37	38	P53	IOA25
IOA26	P54	39	40	P55	IOA27
IOA28	P56	41	42	P57	IOA29
IOA30	P58	43	44	P59	IOA31
GND	GND	45	46	GND	GND
IOA32	P60	47	48	P62	IOA33
IOA34	P65	49	50	P66	IOA35
IOA36	P67	51	52	P68	IOA37
IOA38	P69	53	54	P70	IOA39
GND	GND	55	56	GND	GND
IOA40	P71	57	58	P72	IOA41
IOA42	P74	59	60	P82	IOA43
IOA44	P84	61	62	P81	IOA45
IOA46	P80	63	64	P83	IOA47
IOA48	P86	65	66	P87/P36	IOA49/CLKEXT

※入力ではありません。

CNB

NET LABEL	FPGA pin #	CNB pin #		FPGA pin #	NET LABEL
	3.3V 出力	1	2	3.3V 出力	
	5.0V 入力	3	4	5.0V 入力	
GND	GND	5	6	GND	GND
IOB0	P16	7	8	P20	IOB1
IOB2	P98	9	10	P97	IOB3
IOB4	P96	11	12	P95	IOB5
IOB6	P93	13	14	P30	IOB7
GND	GND	15	16	GND	GND
IOB8	P34	17	18	N.C	IOB9
IOB10	N.C	19	20	N.C	IOB11
IOB12	N.C	21	22	N.C	IOB13
IOB14	N.C	23	24	N.C	IOB15
GND	GND	25	26	GND	GND
IOB16	N.C	27	28	N.C	IOB17
IOB18	N.C	29	30	N.C	IOB19
IOB20	N.C	31	32	N.C	IOB21
IOB22	N.C	33	34	N.C	IOB23
GND	GND	35	36	GND	GND
IOB24	N.C	37	38	N.C	IOB25
IOB26	N.C	39	40	N.C	IOB27
IOB28	N.C	41	42	N.C	IOB29
IOB30	N.C	43	44	N.C	IOB31
GND	GND	45	46	GND	GND
IOB32	N.C	47	48	N.C	IOB33
IOB34	N.C	49	50	N.C	IOB35
IOB36	N.C	51	52	N.C	IOB37
IOB38	N.C	53	54	N.C	IOB39
GND	GND	55	56	GND	GND
IOB40	N.C	57	58	N.C	IOB41
IOB42	N.C	59	60	N.C	IOB43
IOB44	N.C	61	62	N.C	IOB45
IOB46	N.C	63	64	N.C	IOB47
IOB48	N.C	65	66	N.C	IOB49

※入力ではありません。

13. ISE による開発手順

EDX-004 に搭載されている FPGA をコンフィグレーションするには、bit ファイルが、必要となります。このファイルを生成する手順について説明します。bit ファイルを生成するには、次の開発ツールが必要となります。

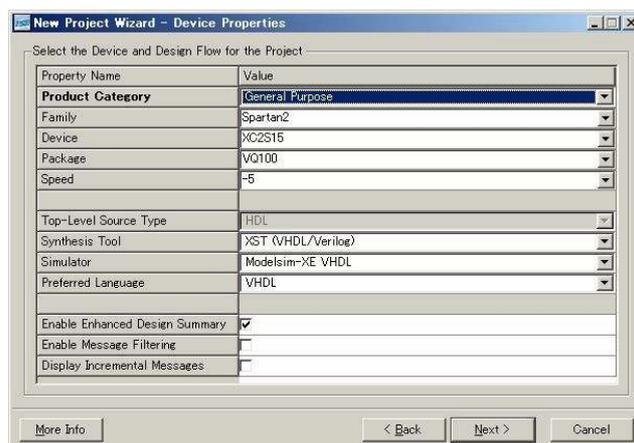
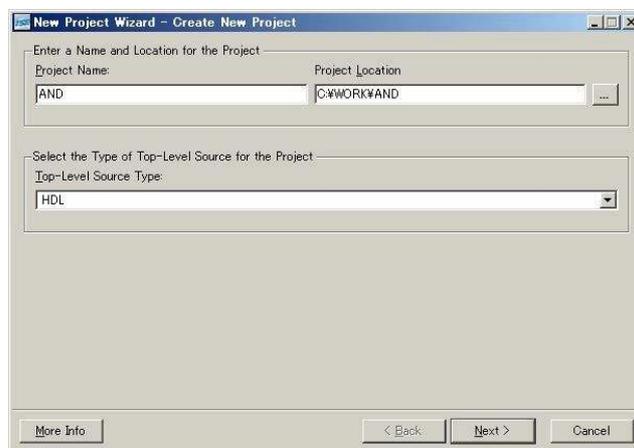
- WebPack ISE もしくは Foundation ISE
- Model Sim

以下、XILINX 社の WebPack ISE 9.1.03i と Model Sim XE III /Starter 6.2c での bit ファイルの生成手順を説明します。

14. プロジェクトの新規作成

プロジェクトを新規作成するには、[File] → [New Project] をクリックします。ここで、ディレクトリとプロジェクト名、ターゲットとなるデバイスも選択します。

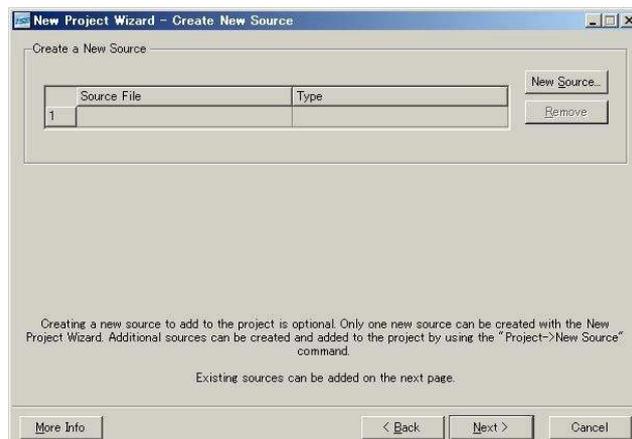
プロジェクトディレクトリは「C: ¥WORK」とします。プロジェクト名を「AND」と入力すると、自動的に「C : ¥ WORK ¥ AND」に変更されます。



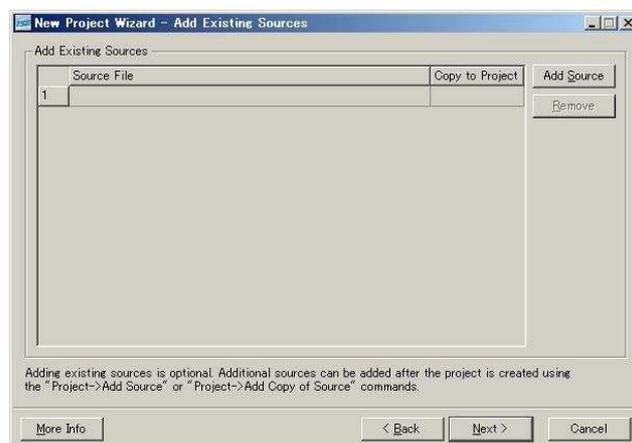
前ページのウインドウのように EDX-004 に搭載されているデバイスに変更します。次の値に変更してください。

Property Name	Value
Product Category	General Purpose
Family	Spartan2
Device	XC2S15
Package	VQ100
Speed	-5
Top-Level Source Type	HDL
Synthesis Tool	XST (VHDL / verilog)
Simulator	Modelsim-XE VHDL
Preferred Language	VHDL

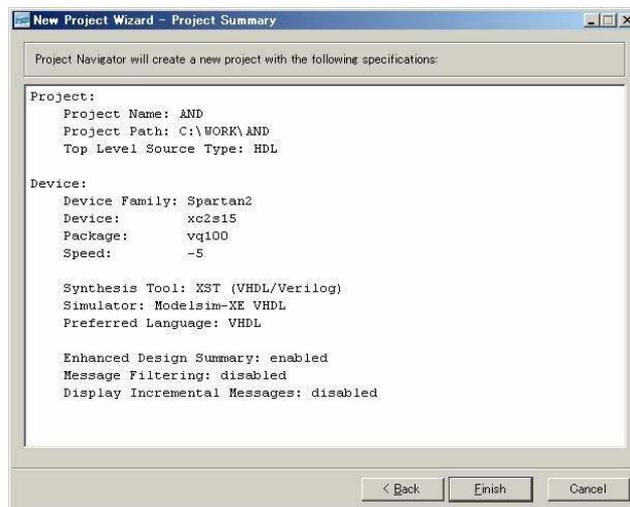
「Next」をクリックすると下のように表示されますので、「Next」をクリックします。



下のように表示されますので、「次へ」をクリックします。

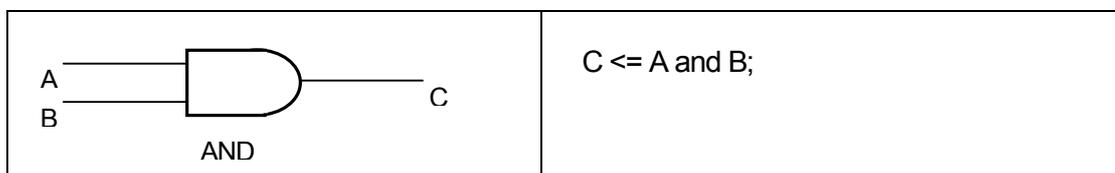


下のような確認画面が表示されますので、間違いがなければ「Finish」をクリックしてください。



15. VHDL による AND 回路

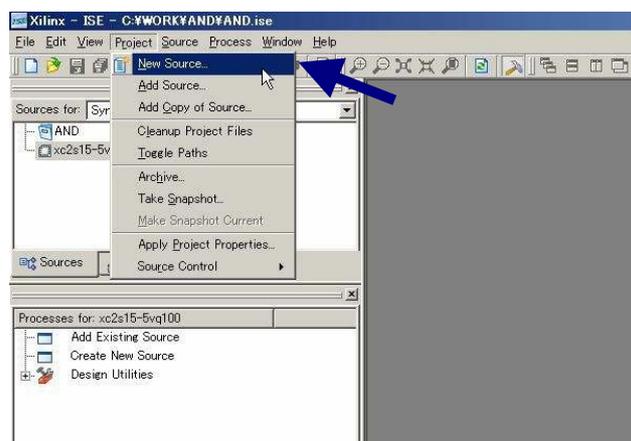
VHDL では、AND 回路は次のように記述します。



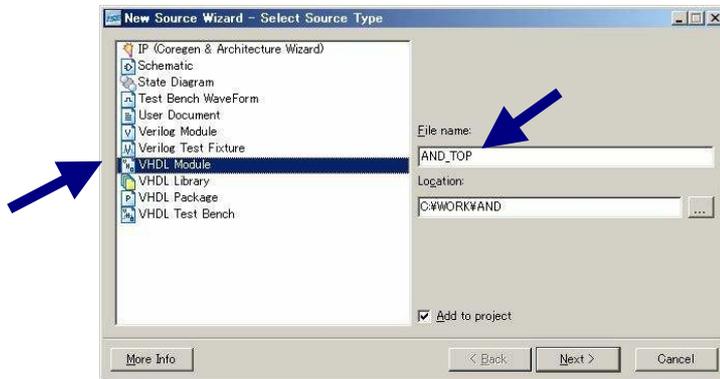
入力 A を「押しボタンスイッチ」の「SW1 (A)」、入力 B を「押しボタンスイッチ」の「SW2 (B)」とします。出力 C は「赤色 LED」の「LED6 (C)」とします。実装する手順について説明します。

16. 新規ソースの追加

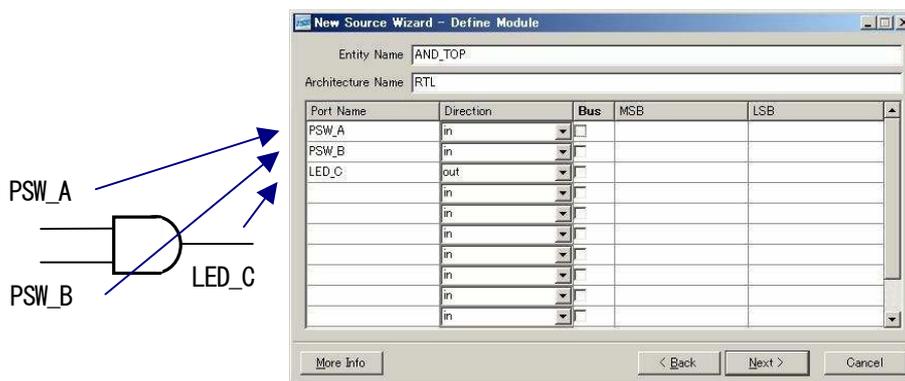
[Project] – [New Source] を選択します。



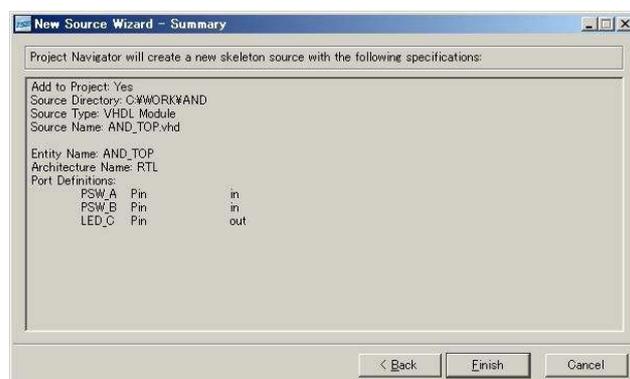
次に[VHDL Module] を選択し、ファイル名を付けます。ここでは「AND_TOP」としています。



次に信号名を付けます。入力信号 (A, B) と出力信号 C を定義します。



「Next」をクリックすると下のような確認画面が表示されますので、間違いがなければ「Finish」をクリックしてください。



自動的に次のような VHDL コードの雛型が生成されます。このコードに追加します。

```

10 -- Tool versions:
11 -- Description:
12 --
13 -- Dependencies:
14 --
15 -- Revision:
16 -- Revision 0.01 - File Created
17 -- Additional Comments:
18 --
19 -----
20 library IEEE;
21 use IEEE.STD_LOGIC_1164.ALL;
22 use IEEE.STD_LOGIC_ARITH.ALL;
23 use IEEE.STD_LOGIC_UNSIGNED.ALL;
24
25 ---- Uncomment the following library declaration if instantiating
26 ---- any Xilinx primitives in this code.
27 --library UNISIM;
28 --use UNISIM.VComponents.all;
29
30 entity AND_TOP is
31     Port ( PSW_A : in  STD_LOGIC;
32           PSW_B : in  STD_LOGIC;
33           LED_C : out STD_LOGIC);
34 end AND_TOP;
35
36 architecture RTL of AND_TOP is
37
38 begin
39

```

次の一行を追加します。

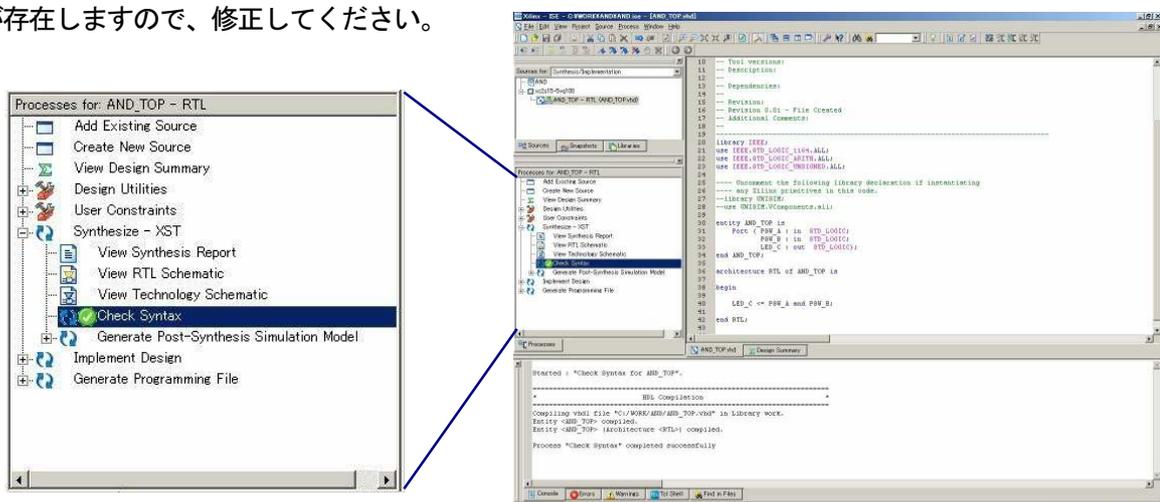
```

29
30 entity AND_TOP is
31     Port ( PSW_A : in  STD_LOGIC;
32           PSW_B : in  STD_LOGIC;
33           LED_C : out STD_LOGIC);
34 end AND_TOP;
35
36 architecture RTL of AND_TOP is
37
38 begin
39
40     LED_C <= PSW_A and PSW_B;
41
42 end RTL;
43

```

追加

プロセスウィンドウの「Synthesize-XST」→「Check Syntax」をダブルクリックし、文法チェックを確認します。緑のチェックマークが付けば正常です。もし赤色の×マークが付けば文法の誤りが存在しますので、修正してください。



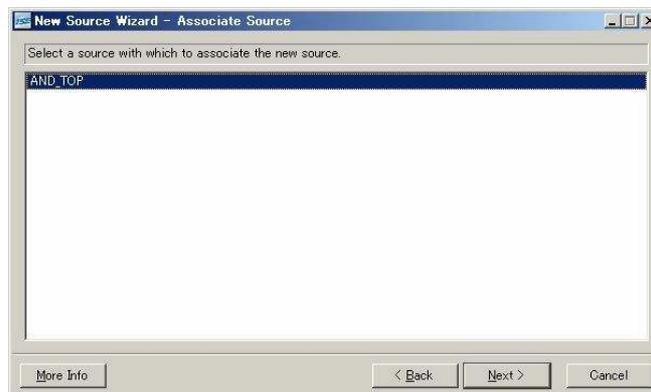
17. テストベンチの作成とシミュレーション

次にシミュレーションに必要なテストベンチを生成します。VHDL ソース同様に[Project]—[New Source] を選択します。「VHDL Test Bench」を選択し、ファイル名を「TB_AND_TOP」として「Next」をクリックします。

(シミュレーションを行うためにはXILINX社のModelSimが、インストールされていることが、必要です。)



対象のVHDLコードである「AND_TOP」を選択して、「Next」をクリックします。



下のような確認画面が表示されます。間違いがなければ「Finish」をクリックすればVHDLのテストベンチが自動生成されます。



自動的に生成されたテストベンチの雛型に次の4行を追加します。

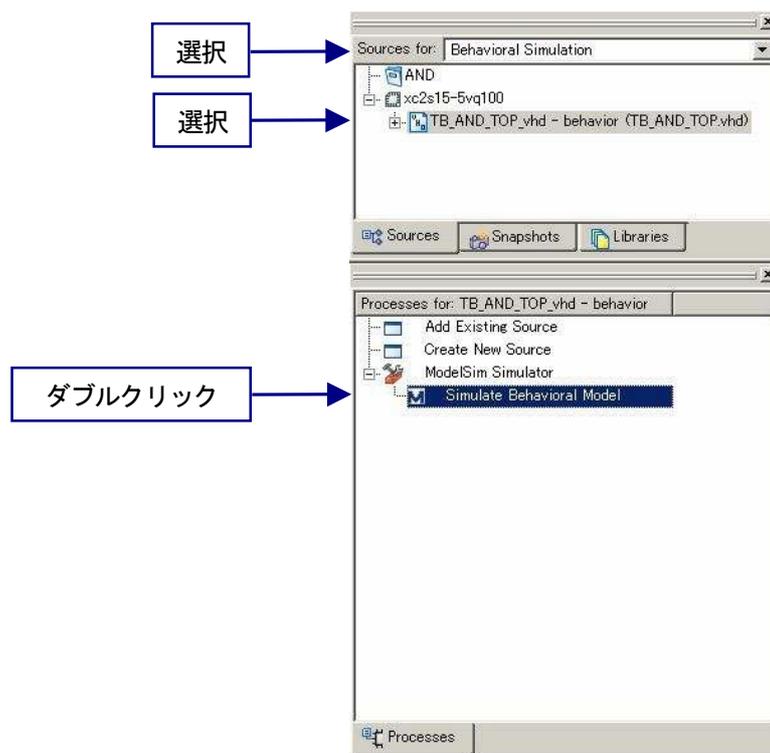
```

55 BEGIN
56
57     -- Instantiate the Unit Under Test (UUT)
58     uut: AND_TOP PORT MAP(
59         PSW_A => PSW_A,
60         PSW_B => PSW_B,
61         LED_C => LED_C
62     );
63
64     tb : PROCESS
65     BEGIN
66     -----
67         PSW_A <= '0'; PSW_B <= '0'; wait for 100 ns;
68         PSW_A <= '0'; PSW_B <= '1'; wait for 100 ns;
69         PSW_A <= '1'; PSW_B <= '0'; wait for 100 ns;
70         PSW_A <= '1'; PSW_B <= '1'; wait for 100 ns;
71     -----
72     -- Wait 100 ns for global reset to finish
73     wait for 100 ns;
74
75     -- Place stimulus here
76
77     wait; -- will wait forever
78     END PROCESS;
79
80 END;
81

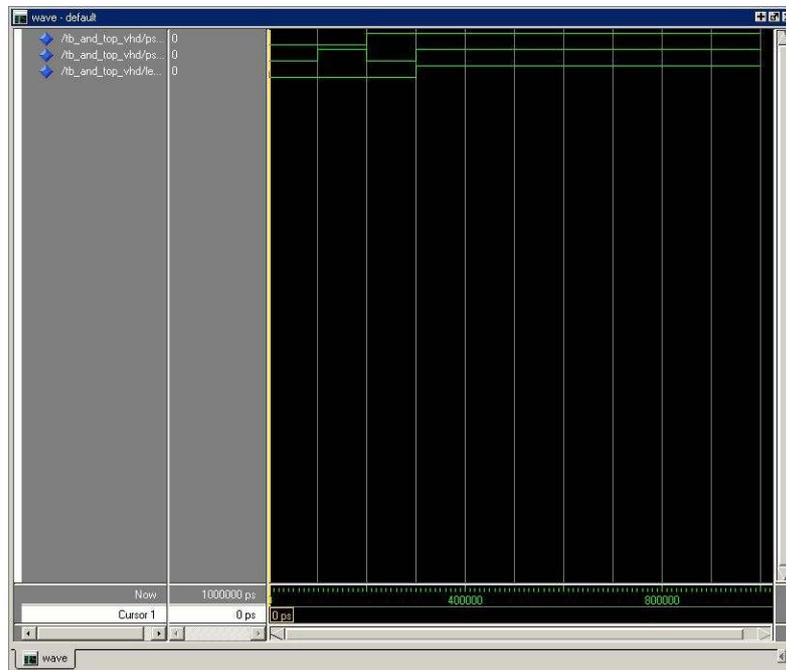
```

追加

Sources タブの Sources for を [Behavioral Simulation] にします。
次に [TB_AND_TOP_vhd-behavior (TB_AND_TOP.vhd)] を選択した状態で、「Simulate Behavioral Model」をダブルクリックします。



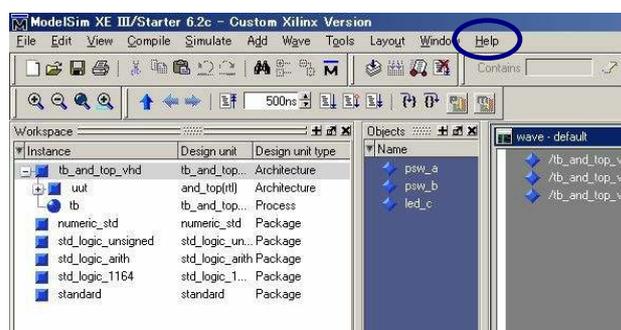
Model Sim が、自動的に起動すると次のように波形の確認ができます。



再実行するにはツールバーの「Restart」をクリックしてください。波形ウインドウのツールバーは次のようになっています。

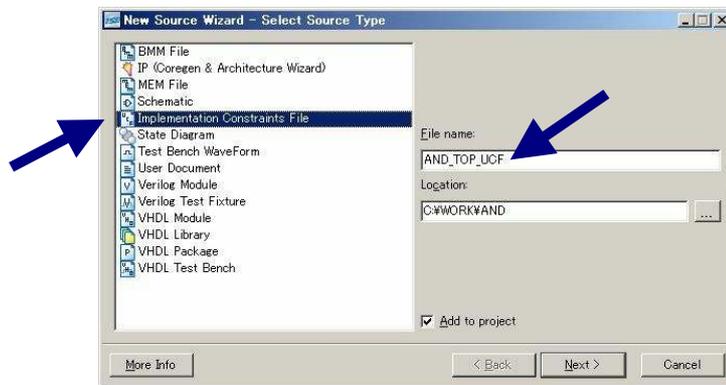


Model Sim は非常に豊富な機能をもっております。詳細につきましては、[Help]-[PDF Documentation]からユーザーズマニュアルを参照してください。

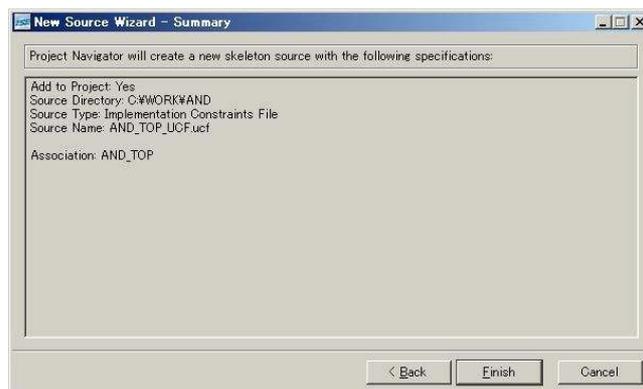


18. UCF ファイルの作成

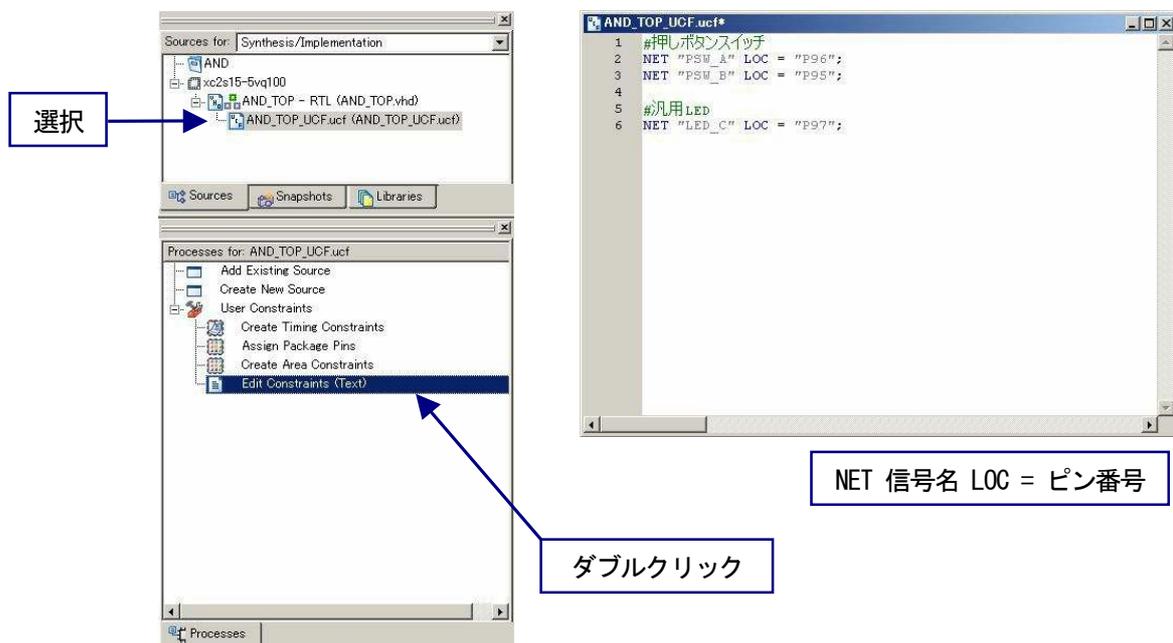
FPGA のピンを固定するために、UCF ファイルを作成します。VHDL ソース同様に[Project] – [New Source] を選択します。[Implementation Constraints File] を選択し、ファイル名を「AND_TOP_UCF」として「Next」をクリックします。



下のような確認画面が表示されますので、間違いがなければ「Finish」をクリックします。

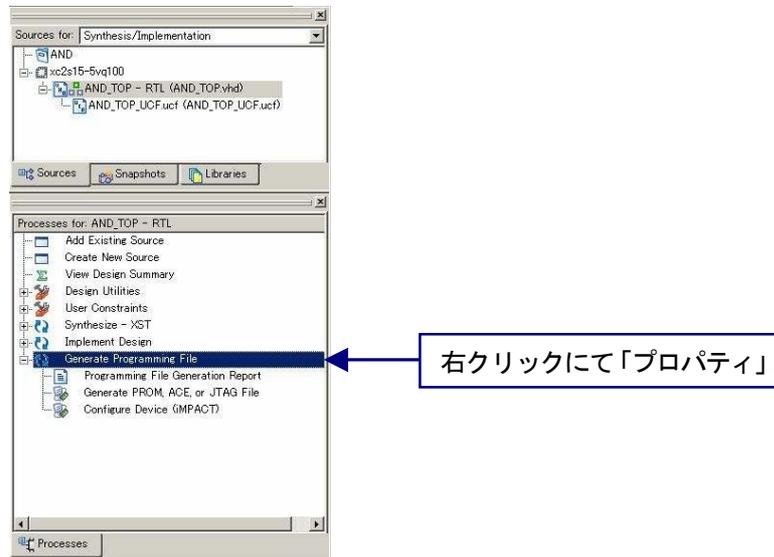


次のように「Edit Constraints (Text)」をダブルクリックし、ピン定義を行います。

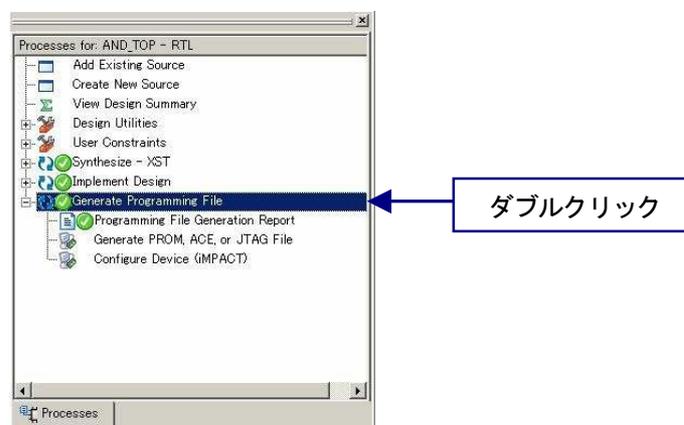


19. BIT ファイルの作成

BIT ファイルの「FPGA Start -Up Clock」は「CCLK」である必要があります。



ダブルクリックにてBIT ファイルが、生成されます。正常に生成された場合には、緑のチェックが、次のウィンドウのようになります。



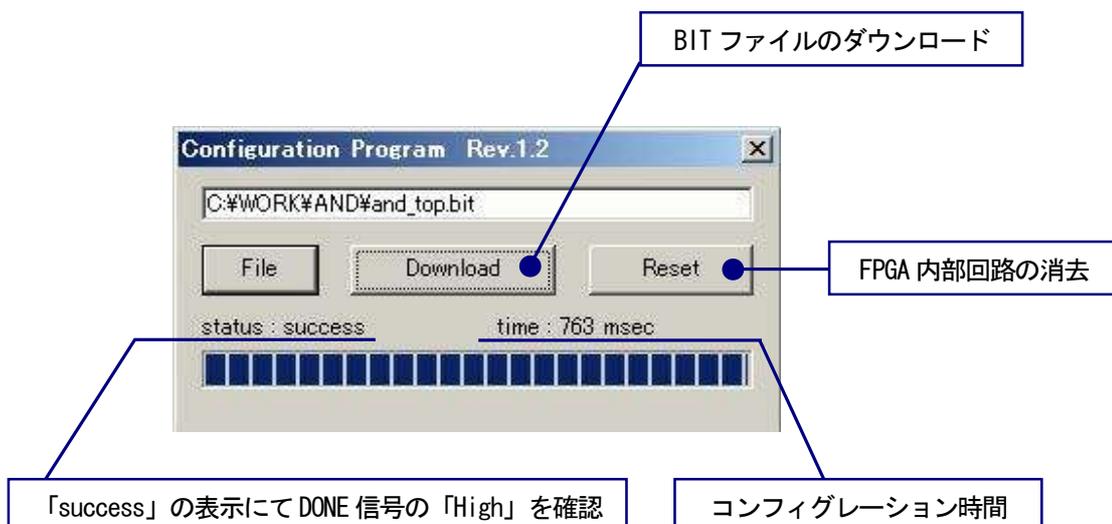
EDX-004 のFPGA へのコンフィグレーションはUSB 経由で行うため、XILINX 社 iMPACT を利用することはできません。付属の専用ソフトウェアにて、USB コンフィグレーションを行ってください。

20. 専用ソフトウェアによる FPGA コンフィグレーション

付属 CD 内にある「BitCfg.exe」を起動させると、次のウィンドウが起動します。

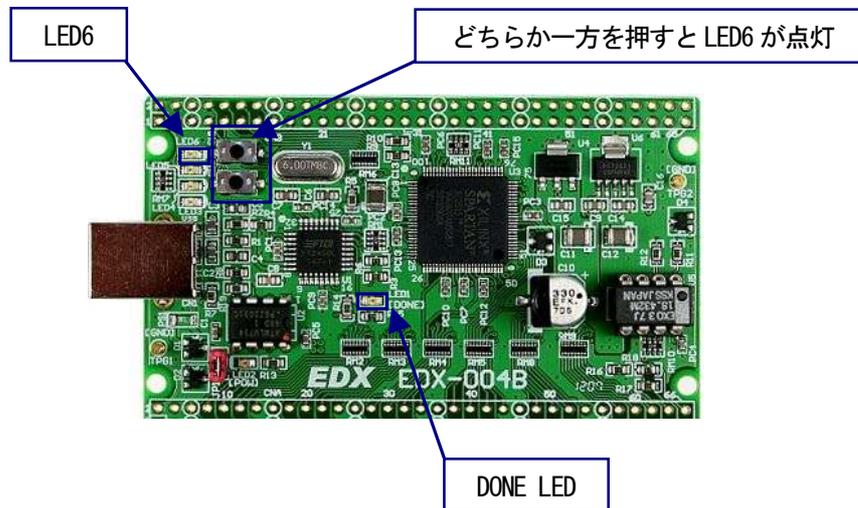


生成した AND 回路「and_top.bit」を選択し、「Download」を選択します。「DONE LED」が点灯し、ウィンドウの「status:success」であれば正常に FPGA にコンフィグレーションであることを意味します。

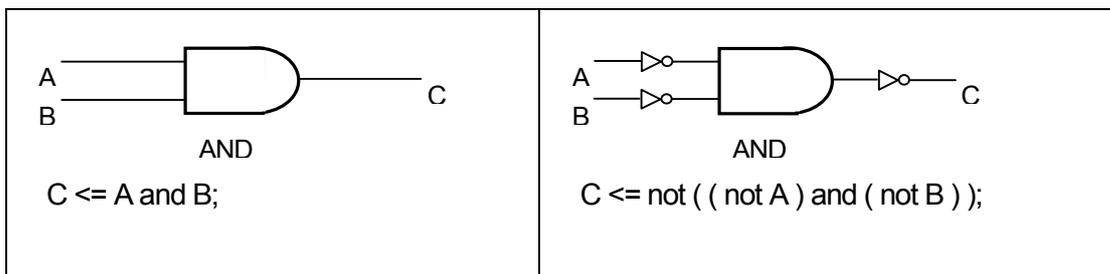


21. 動作の確認

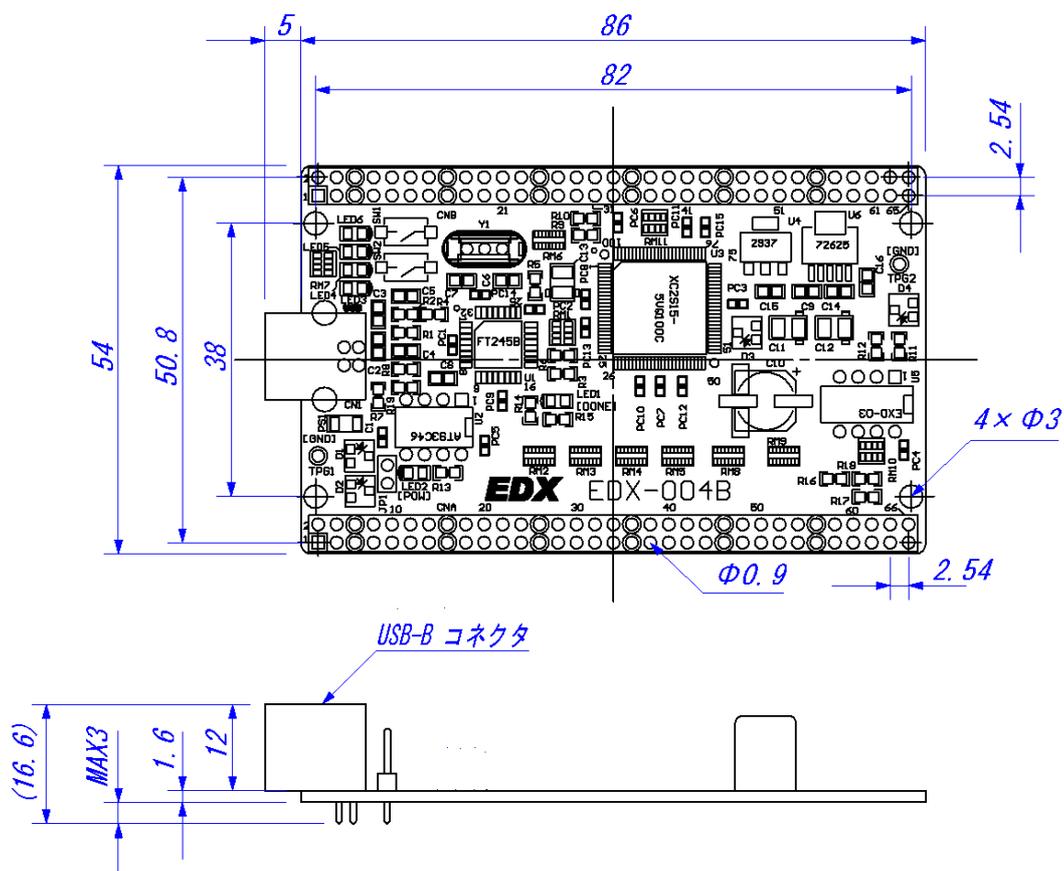
赤色 LED と押しボタンスイッチが負論理であるため、押すと点灯するのではなく両方の押しボタンを押していない状態で消灯し、それ以外は点灯となっています。



両方、押した時だけ点灯させるには、次のように「インバータ」を挿入します。



22. 製品外形図



23. 付属 CD-ROM の内容

CD 内の「readme.txt」をご覧ください。

最新のデータシートは、各社のホームページからダウンロードしてください。

XILINX 社のホームページ <http://www.xilinx.com/>

FTDI 社のホームページ <http://www.ftdichip.com/>

24. 付属資料

1. 基板回路図 (別紙)

FPGA トレーナ (カードサイズ)
EDX-004

ユーザーズマニュアル

2007/04/13 初版

2007/04/27 第2版

2008/06/25 第3版

2008/08/05 第4版

有限会社ヒューマンデータ

〒567-0034

大阪府茨木市中穂積1-2-10

ジブラルタ生命茨木ビル

TEL 072-620-2002

FAX 072-620-2003

URL <http://www.hdl.co.jp/>

このマニュアルは2色で印刷されております