

**Application note**

# TE7720

## 1. 概要

TE7720 を使用する際には Xilinx 社ツール (ISE) と併用することを前提としています。また、お客様システムに組み込む場合に、各種プログラム、ユーティリティ・ソフト等を用意しています。本アプリケーション・ノートでは、これらの操作方法について記述します。

TE7720 コントローラの詳細については、以下のデータシートをご参照下さい。

### TE7720 HARDWARE DATASHEET

また、システム設計に関しては、Xilinx 社 FPGA の関連書も参照下さい。

本仕様書中で使用している登録商標、商標は、それぞれの各社および団体に帰属します。  
この仕様書は、信頼性、機能または設計を改善するために予告無く変更を加える事があります。

## 2. 目次

1. 概要.....	1
2. 目次.....	2
3. SVF データの作成.....	3
3-1. SVF データを作成する前に .....	3
3-1-1. Start-Up Clock の設定.....	3
3-1-2. MCS ファイルの作成.....	4
3-2. iMPACT での SVF データ作成(ISE から起動).....	7
3-3. iMPACT での SVF データ作成(単体起動).....	12
4. 容量が 8Mbit 以上の Flash メモリを使用する場合.....	17
5. FPGA を複数個チェーンする場合のダウンロードデータ作成法 .....	19
6. ROM ライタで Flash メモリに書き込む際のデータ作成手順 .....	21
7. XSVF ファイル (バイナリ-フォーマット) 作成方法.....	23
8. XSVF ファイルのダウンロード方法 .....	23
9. バッチコマンド例.....	24
10. FPGA/Configuration Data 以外のダウンロード例 (CPU プログラム等) .....	26
10-1. Binary データをインテルヘキサ・ファイル (MCS) に変換。 .....	26
10-2. iMPACT への読み込み .....	27
11. 参考情報.....	28
11-1. ダウンロード時間 .....	28
11-2. Configuration 時間 .....	28

### 3. SVF データの作成

TE7720 を使用して Flash メモリに FPGA/Configuration Data を書き込む際に必要となるデータの準備、又 SVF データの iMPACT での作成方法を記述します。

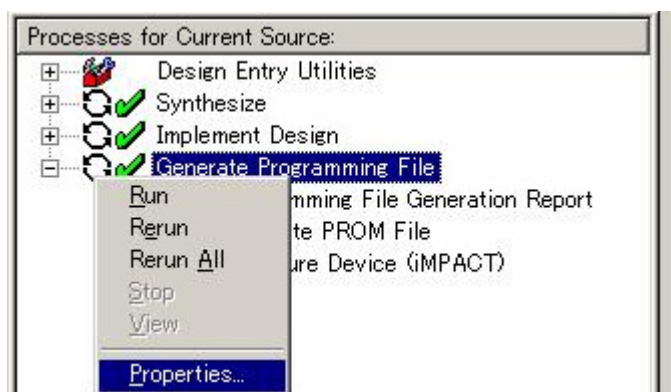
#### 3-1. SVF データを作成する前に

iMPACT で SVF データを作成する前に必要な設定事項、必要なファイルについて説明します。

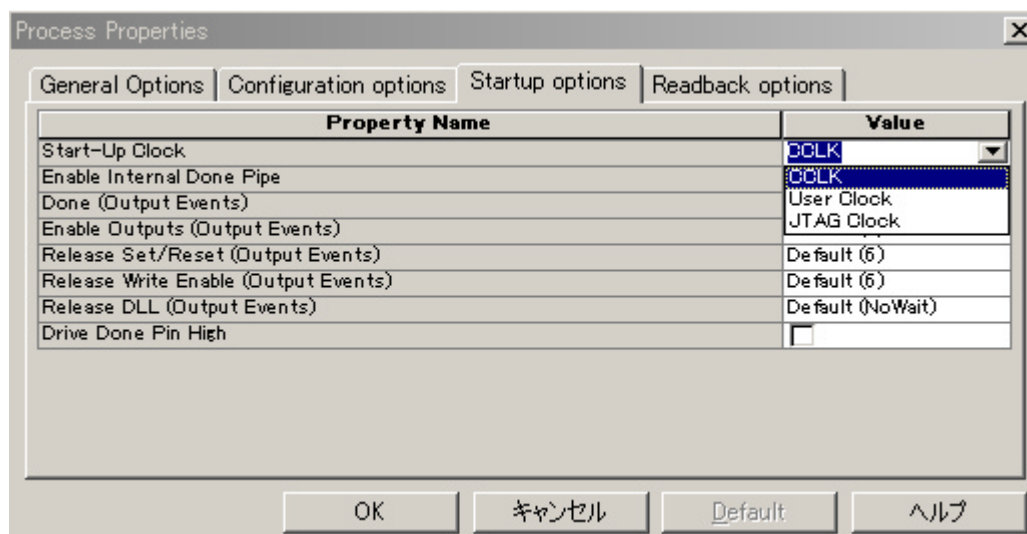
##### 3-1-1. Start-Up Clock の設定

Flashメモリ Little Bear FPGA コンフィギュレーションではコンフィギュレーションクロックに CCLK を使用するマスターシリアルモードで行います。そのため Start-Up Clock を CCLK に設定して下さい。(デフォルトで CCLK になっています。)

Generate Programming File のプロパティを開きます。



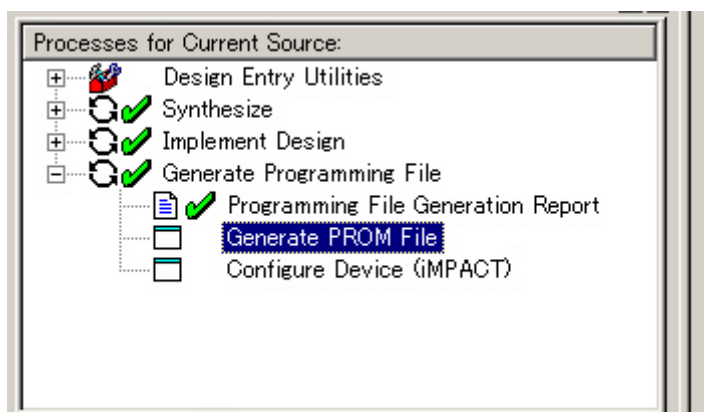
Startup options タブの Start-UP Clock を CCLK に設定します。



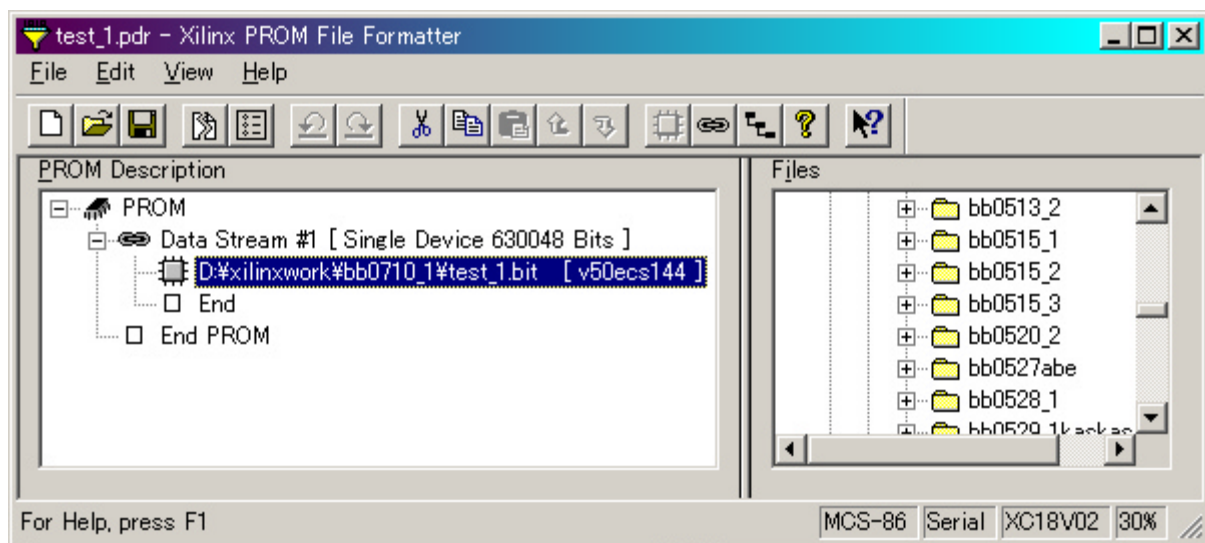
### 3-1-2. MCS ファイルの作成

SVF ファイル作成する場合、直接 Bit ファイルから変換するとデータがビットスワップされていないため、TE7720 を使用した場合に Flash メモリへ書き込んだデータを用いて正常に FPGA を Configuration することができません。ビットスワップさせるため MCS ファイルに変換する必要があります。

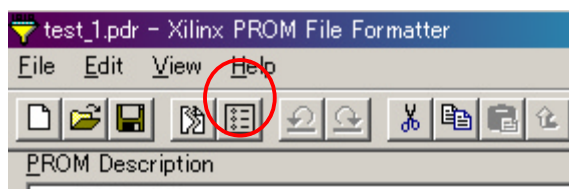
Generate PROM File をダブルクリックして PROM File Formatter を起動します。



自動的にデザインの Bit ファイルが読み込まれます。



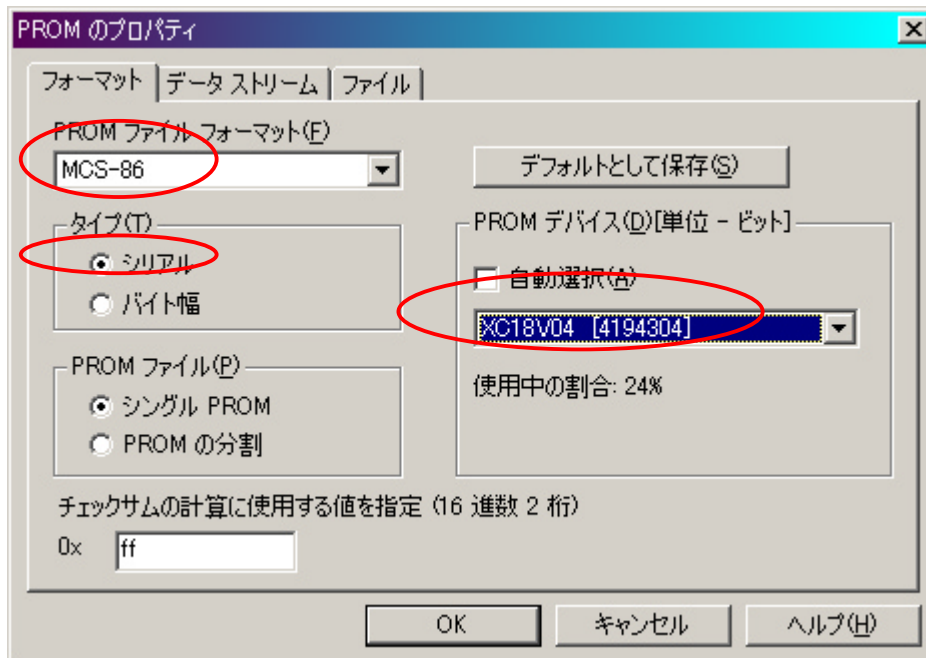
PROMのプロパティアイコンをクリックします。



PROMのプロパティウインドウで以下のように設定します。

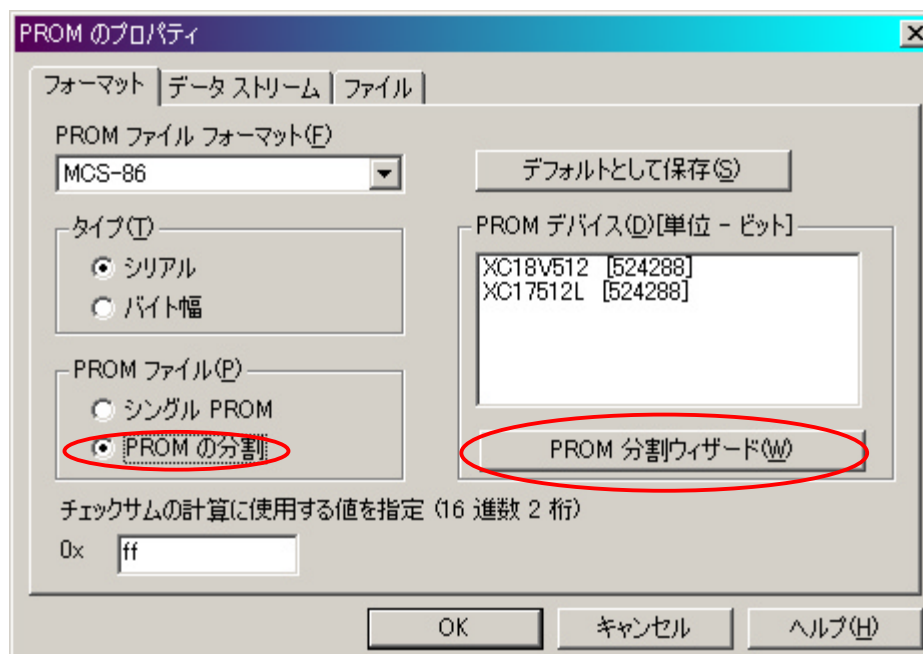
PROM ファイルフォーマット MCS-86  
 タイプ シリアル  
 PROM デバイス 18V04 (必ず 1 8 V04 を指定します。)

OK をクリックします。

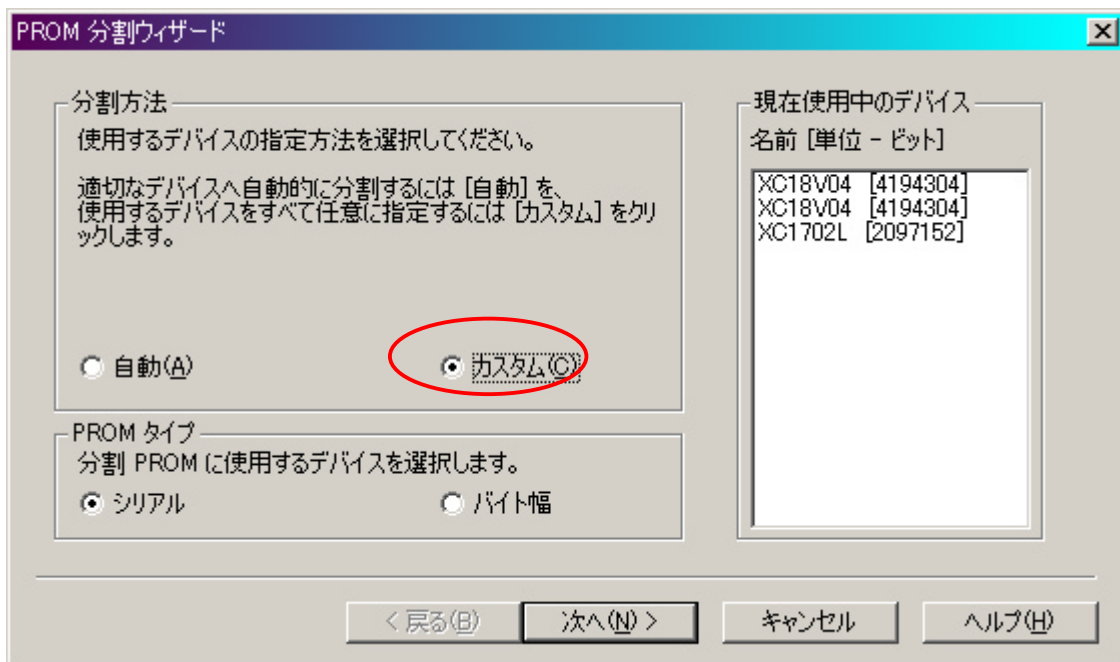


このとき 1 8 V04 で使用率が 100%を超える場合は

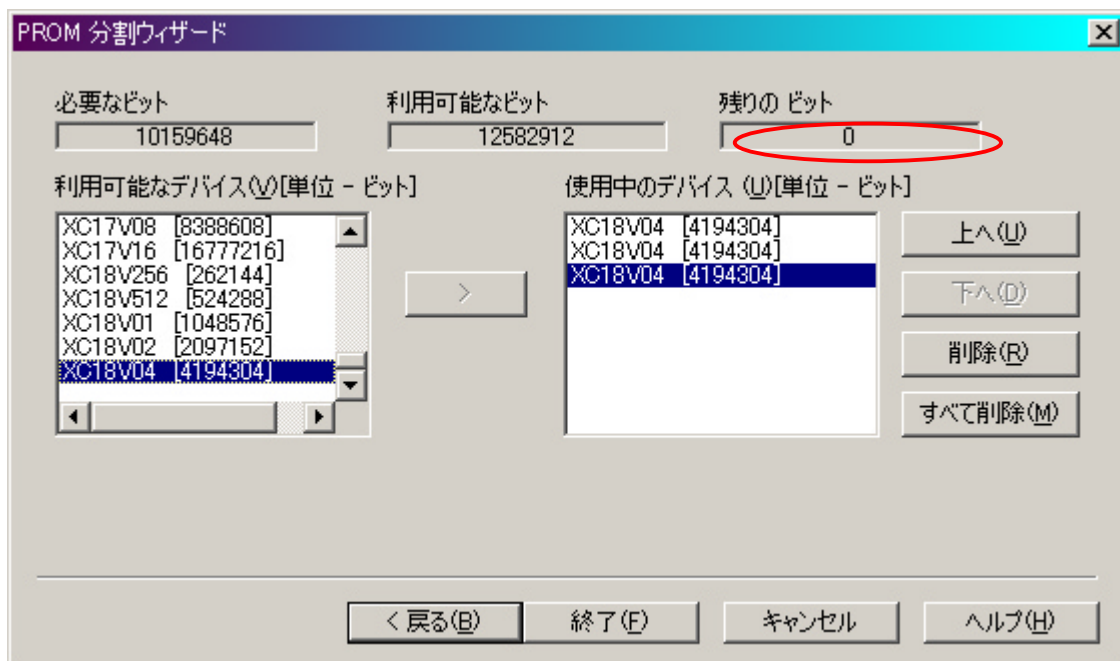
- ・ PROM の分割にチェックを入れ、PROM 分割ウィザードをクリックします。



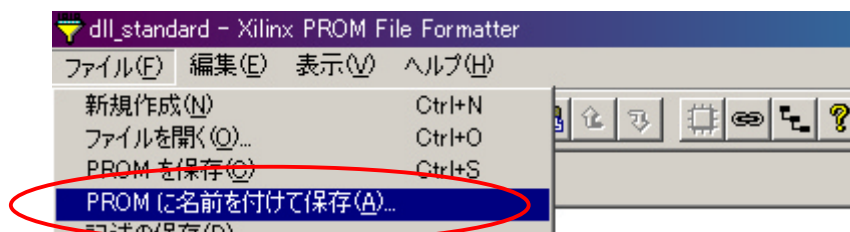
1. PROM 分割ウィザードのウィンドウの分割方法で“カスタム”にチェックを入れ“次へ”をクリックします。



1. 次の画面で、残りのビットが0になるまで18V04を追加(最大4個まで)します。(必ず18V04を選択してください。)終了をクリックします。



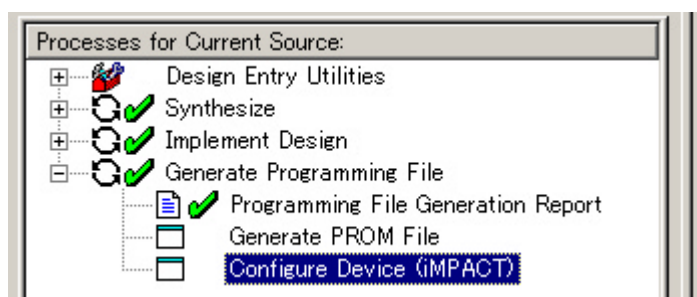
1. 最後に作成したMCSデータを保存してください。



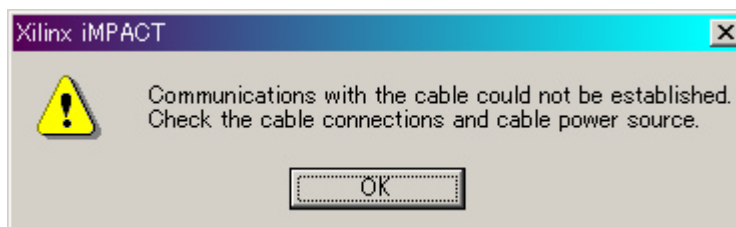


### 3-2. iMPACT での SVF データ作成(ISE から起動)

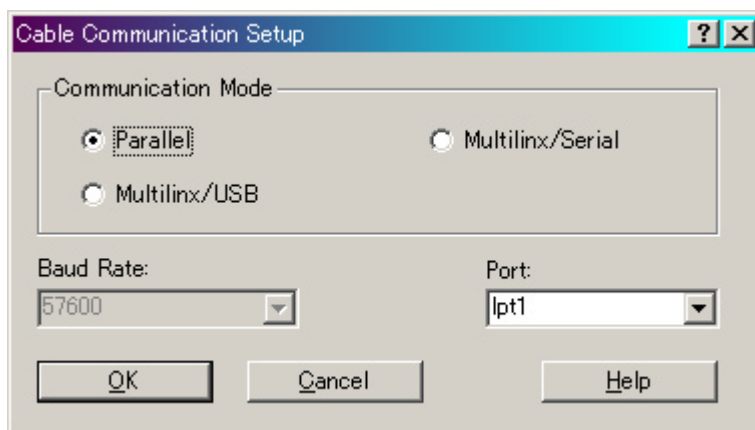
ISE から iMPACT を起動します。



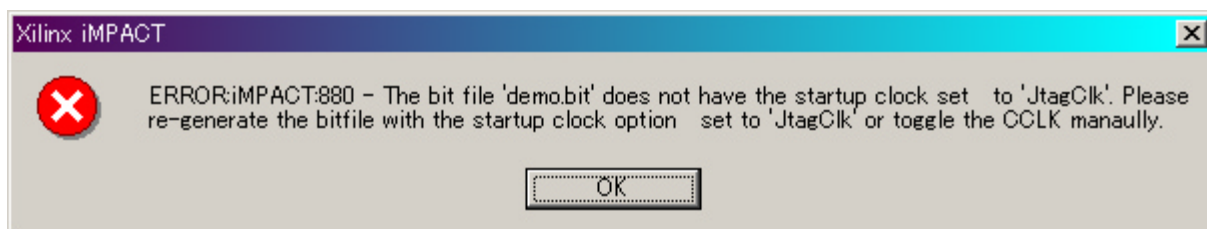
以下のようなエラーが出ますが, SVF ファイルを作成する場合にはケーブルが接続されている必要はありません。無視して OK をクリックしてください。



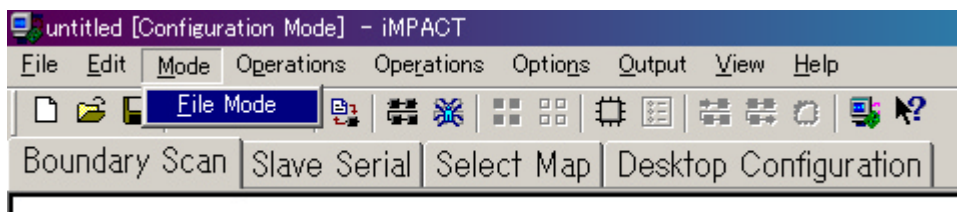
ケーブルセットアップのウィンドウが開きます。上記と同様にケーブルは必要ありませんので Cancel をクリックします。



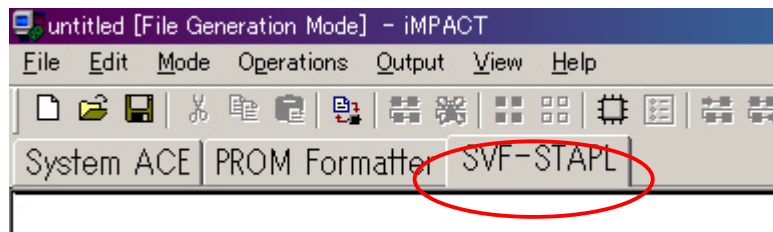
次に以下のエラーが出ますが無視して OK をクリックしてください。



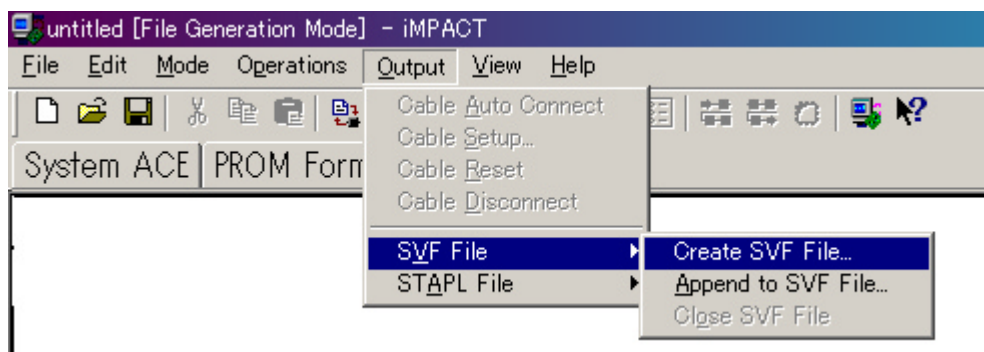
Mode File Mode をクリックします。



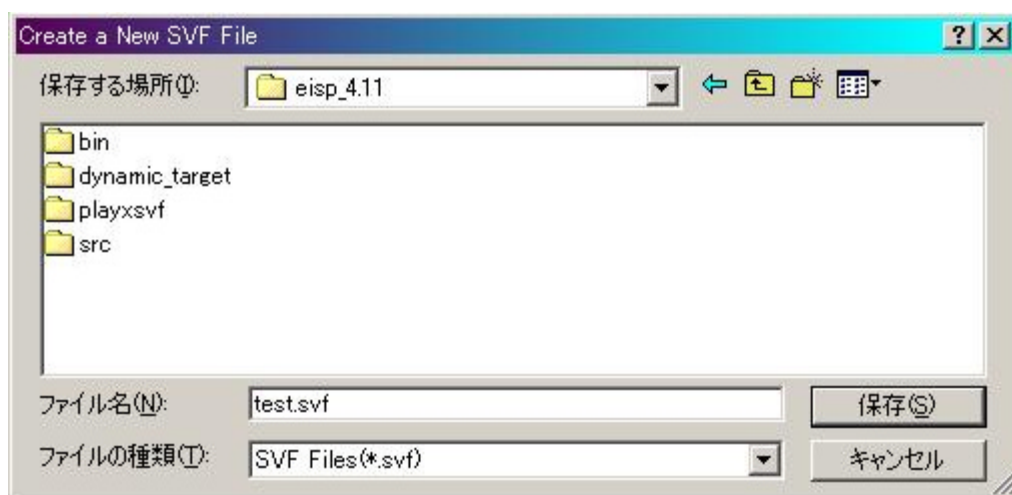
SVF-STAPL タブを選択します。



メニューの Output SVF File Create SVF File をクリックします。

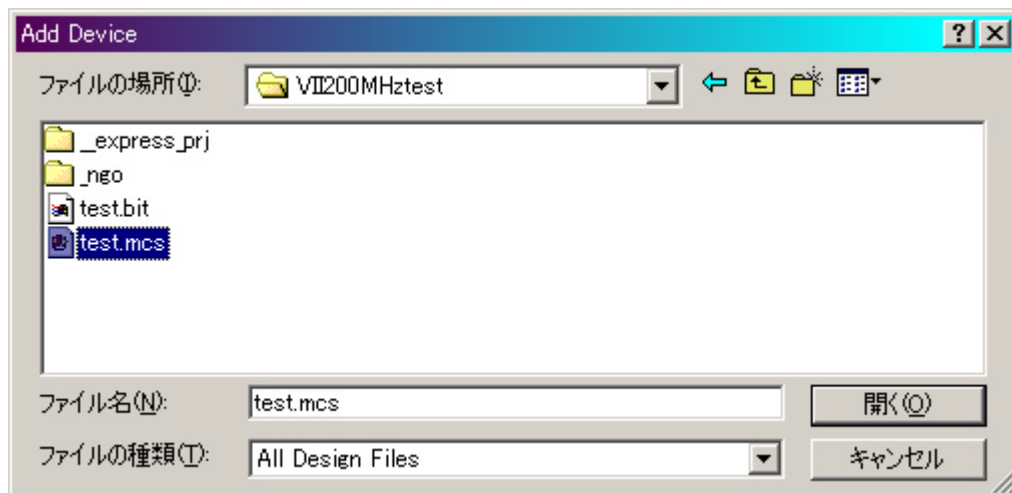


作成するファイル名、場所を指定します。

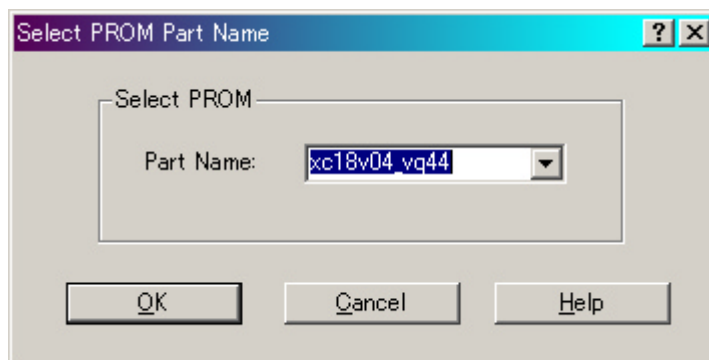




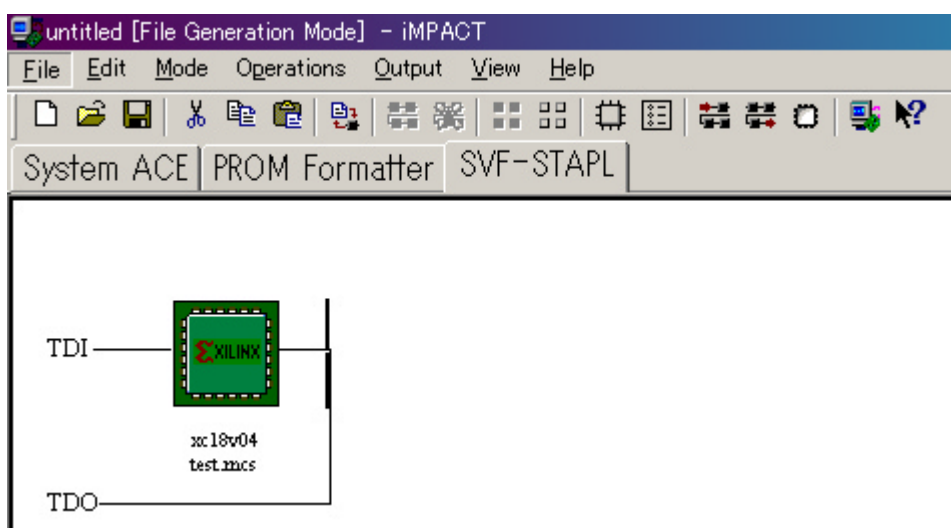
SVFに変換するMCSファイルを選択します。



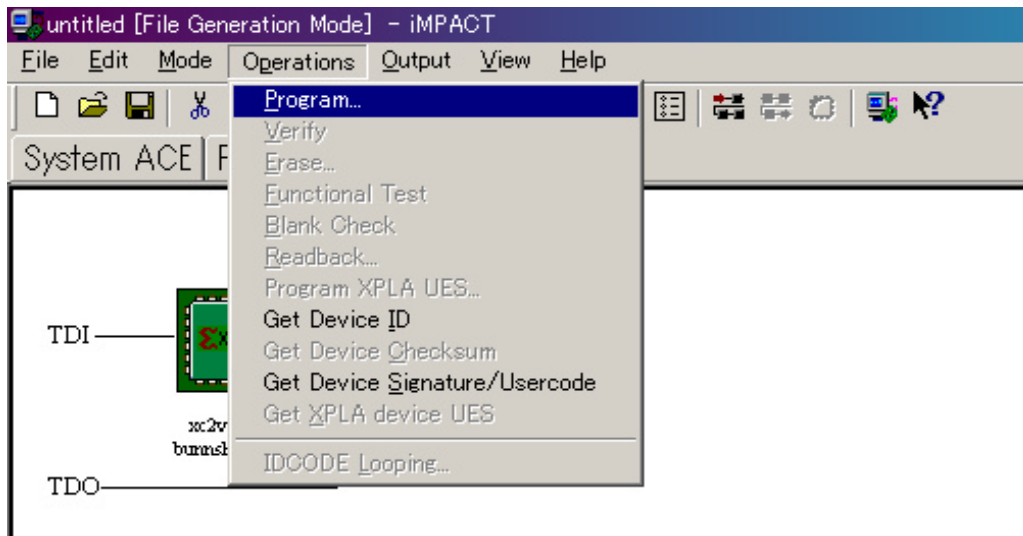
以下の画面が表示されますので必ず XC18V04\_VQ44 を選択します。



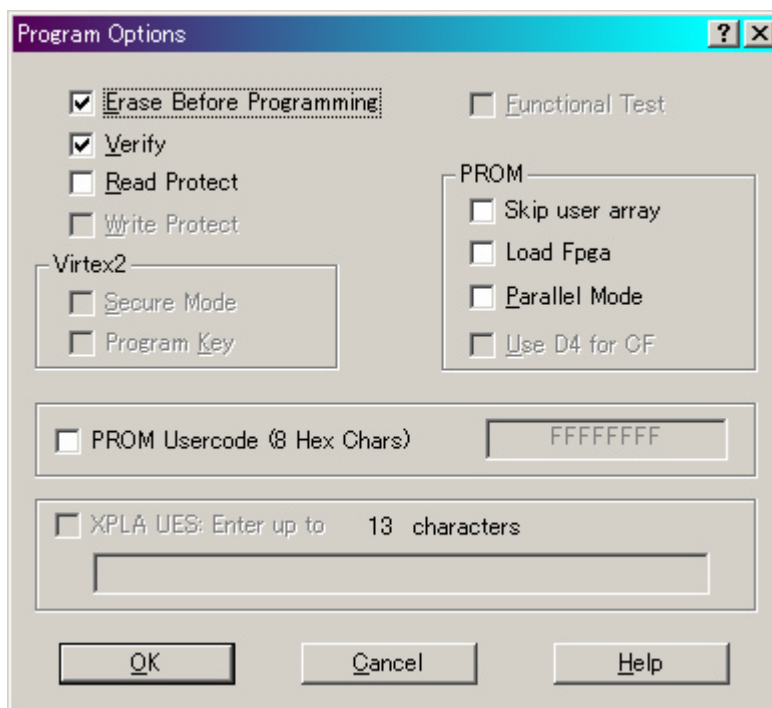
画面上に以下のような図が表示されます。



デバイスを選択(デバイスの図をクリックして反転させる)して,  
“ Operation ” “ Program ” をクリックします。



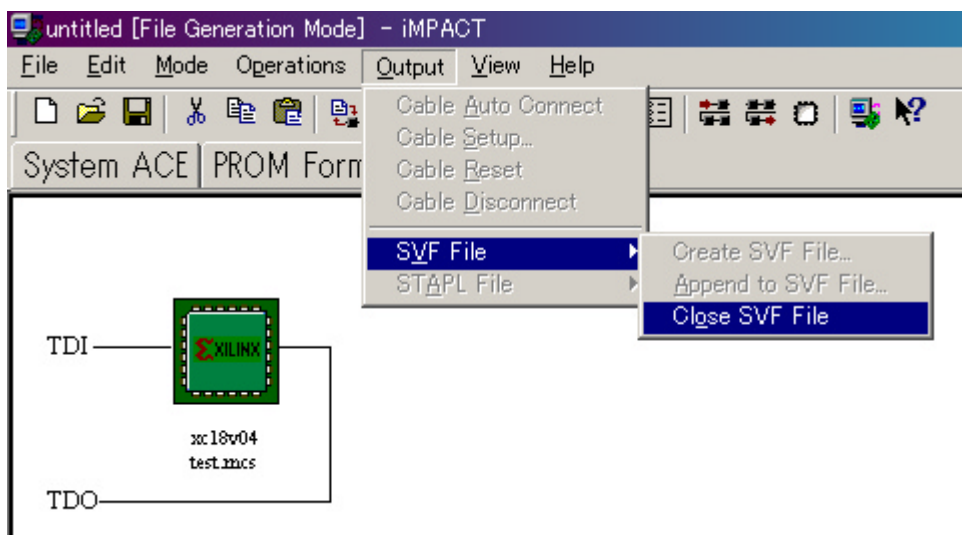
以下のウィンドウが開きます。Verify も同時に行う際は，Verify の項目にチェックを入れます。  
OK をクリックします。



以下のメッセージが表示され SVF ファイルが作成されます。



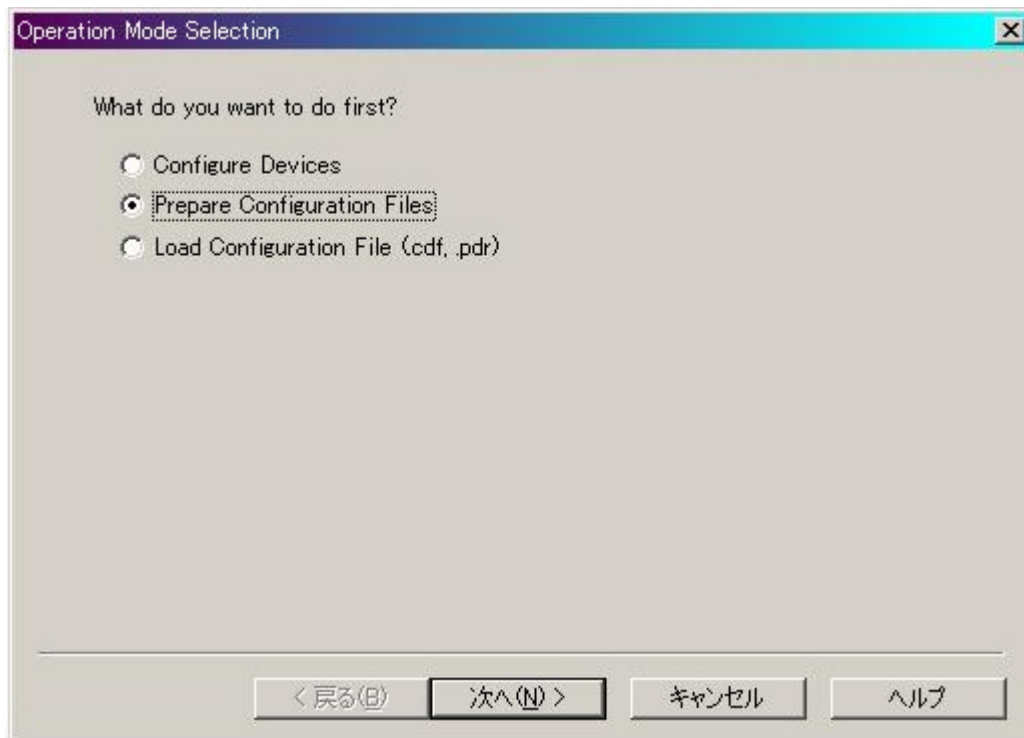
“ Output ” “ SVF File ” “ Close SVF File ” をクリックし SVF 作成作業を終了します。



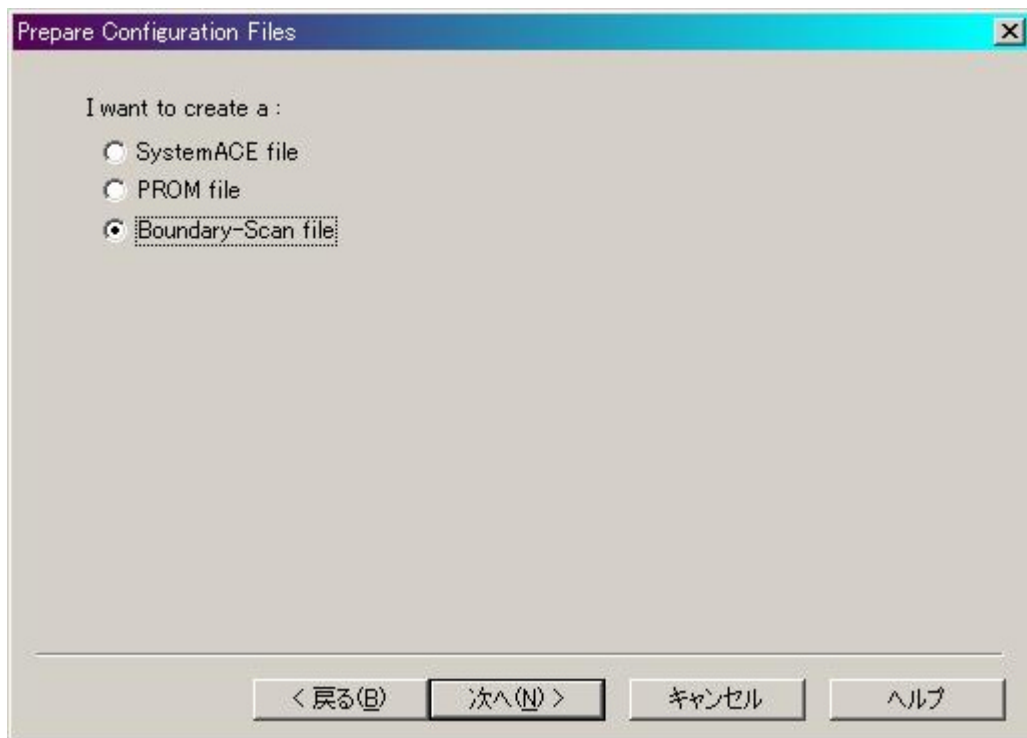
### 3-3. iMPACT での SVF データ作成(単体起動)

MPACT を単体起動します。

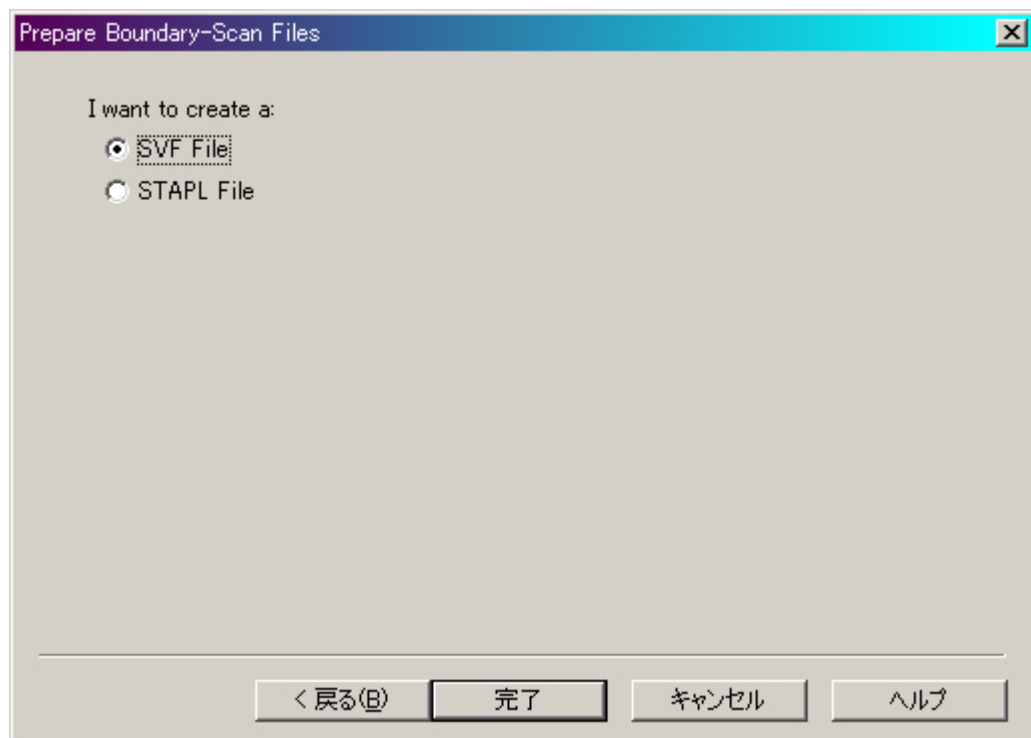
起動後 “Operation Mode Selection” ウィンドウが開きます。  
ここで, “Prepare Configuration Files” にチェックをいれ次へをクリックします。



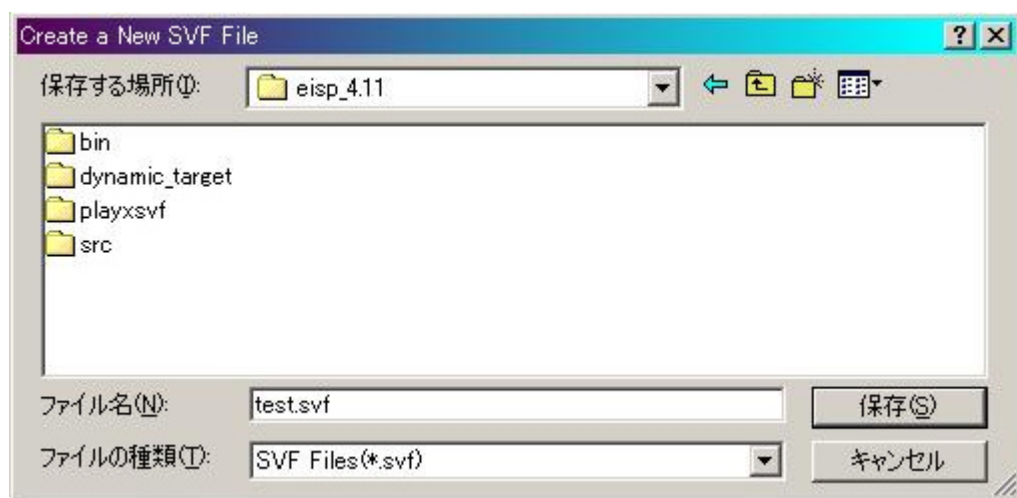
次のウィンドウで “Boundary-Scan File” にチェックを入れ次へをクリックします。



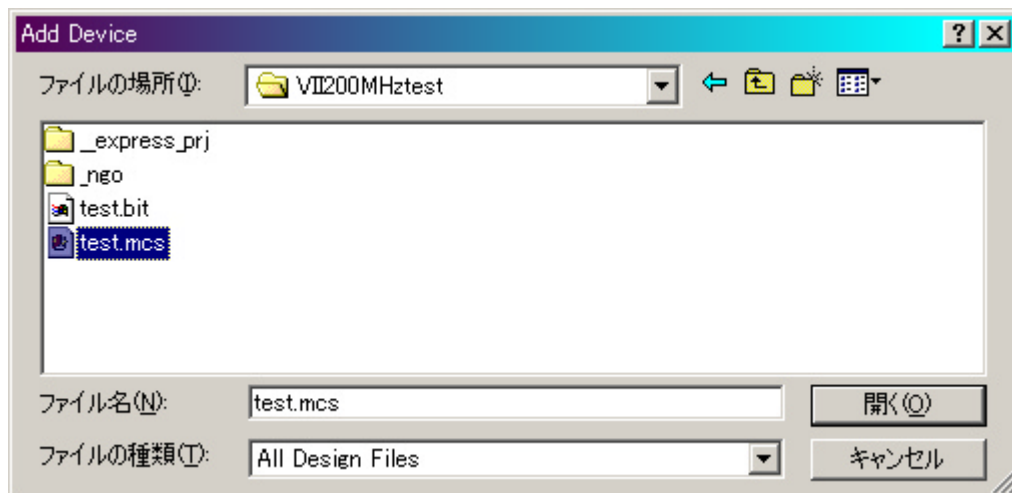
次のウインドウで“SVF File”にチェックを入れ完了をクリックします。



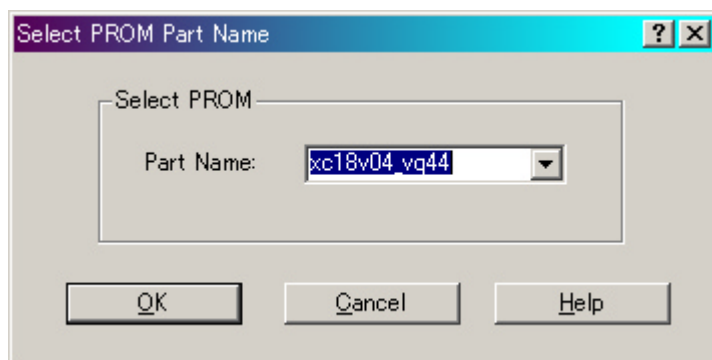
作成するファイル名，場所を指定します。



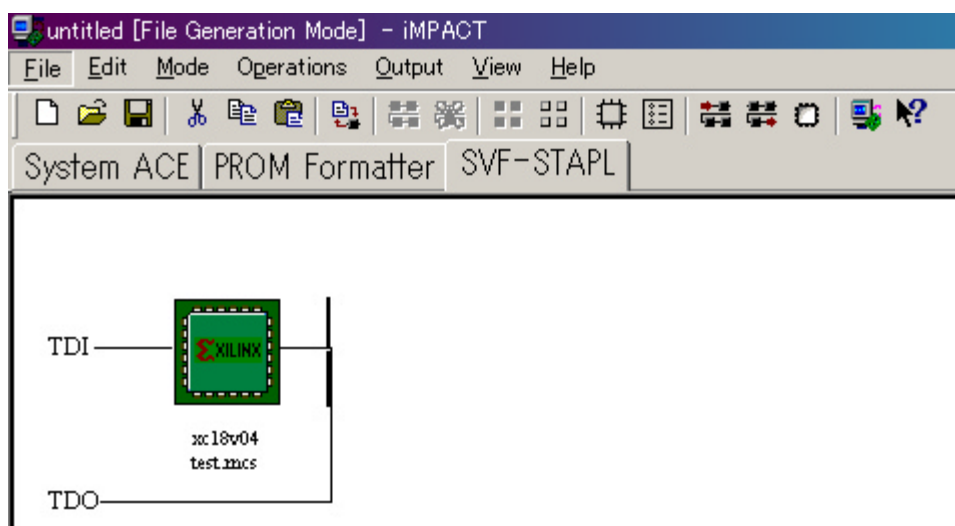
SVFに変換するMCSファイルを選択します。



以下の画面が表示されますので必ずXC18V04\_VQ44を選択します。

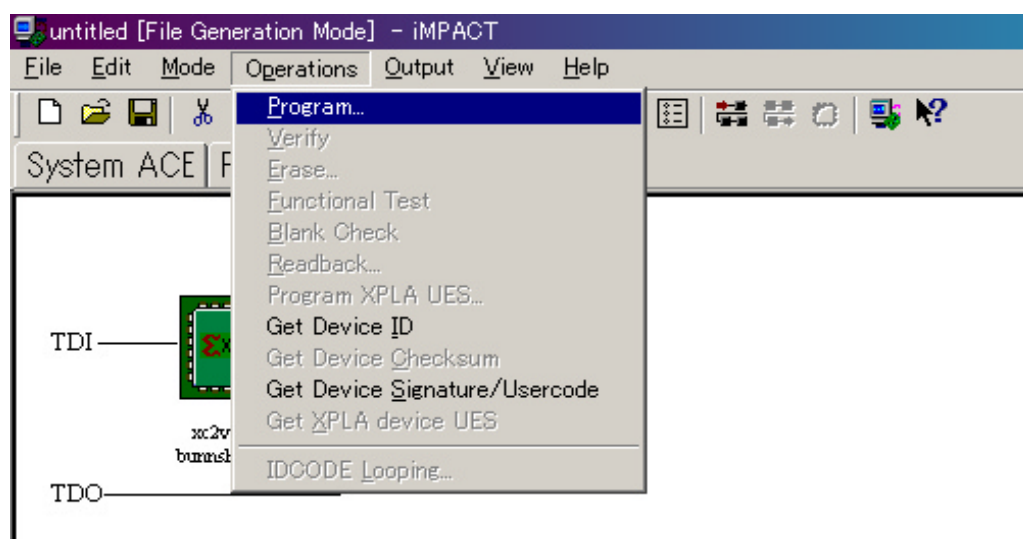


画面上に以下のような図が表示されます。

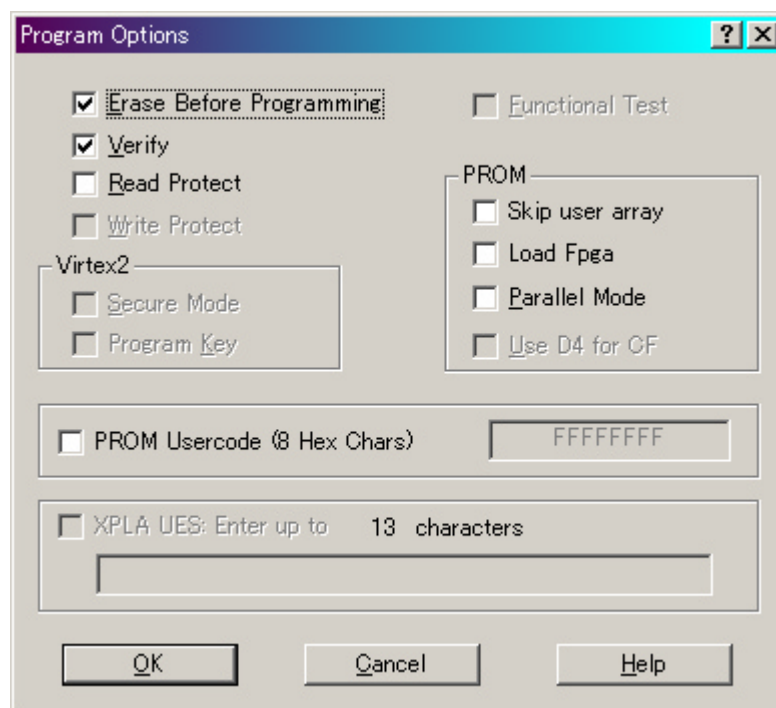




デバイスを選択(デバイスの図をクリックして反転させる)して,  
 “ Operation ” “ Program ” をクリックします。



以下のウィンドウが開きます。Verify も同時に行う際は，Verify の項目にチェックを入れます。  
 OK をクリックします。



以下のメッセージが表示され SVF ファイルが作成されます。

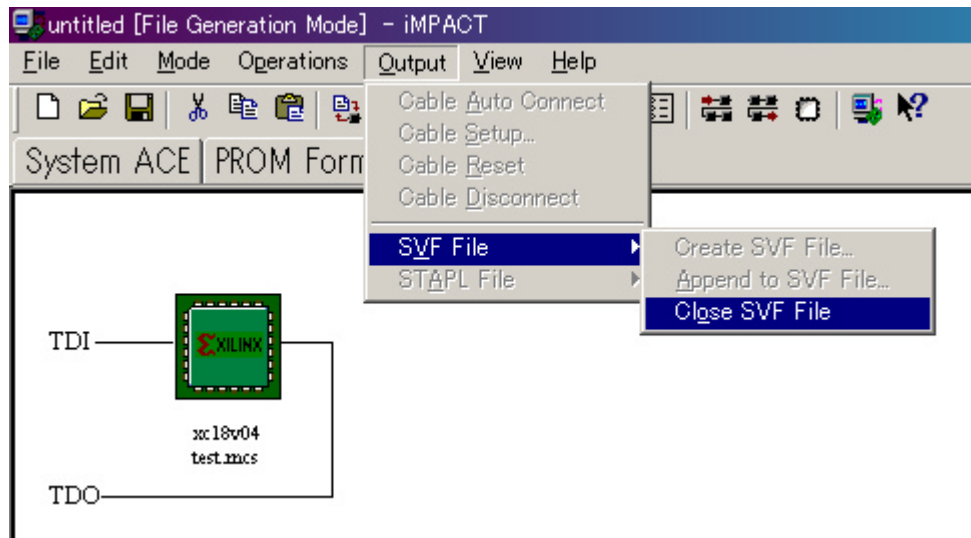
SVF

Programming Succeeded

“Output”

“SVF File”

“Close SVF File” をクリックし SVF 作成作業を終了します。



#### 4. 容量が 8Mbit 以上の Flash メモリを使用する場合

TE7720(Little Bear)は Xilinx 社の開発環境から 18V04 として認識されるよう設計されています。そのため、8Mbit の Flash メモリの場合 18V04 × 2 個、16Mbit 及び 32Mbit の Flash メモリの場合には 18V04 × 4 個を想定して SVF データを作成する必要があります。

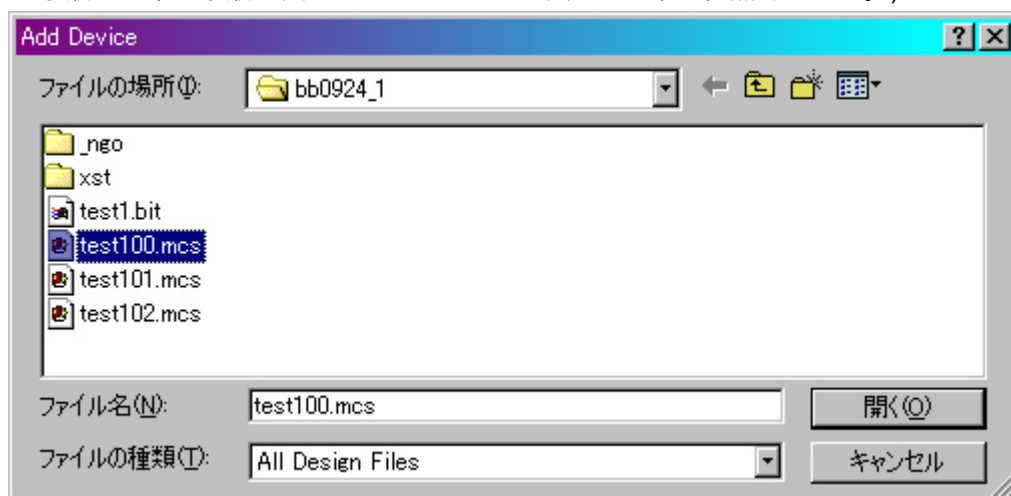
(32Mbit の Flash メモリでは上位の 16Mbit のみ使用します。)

SVF データ作成方法(例：16Mbit の Flash メモリ使用時)

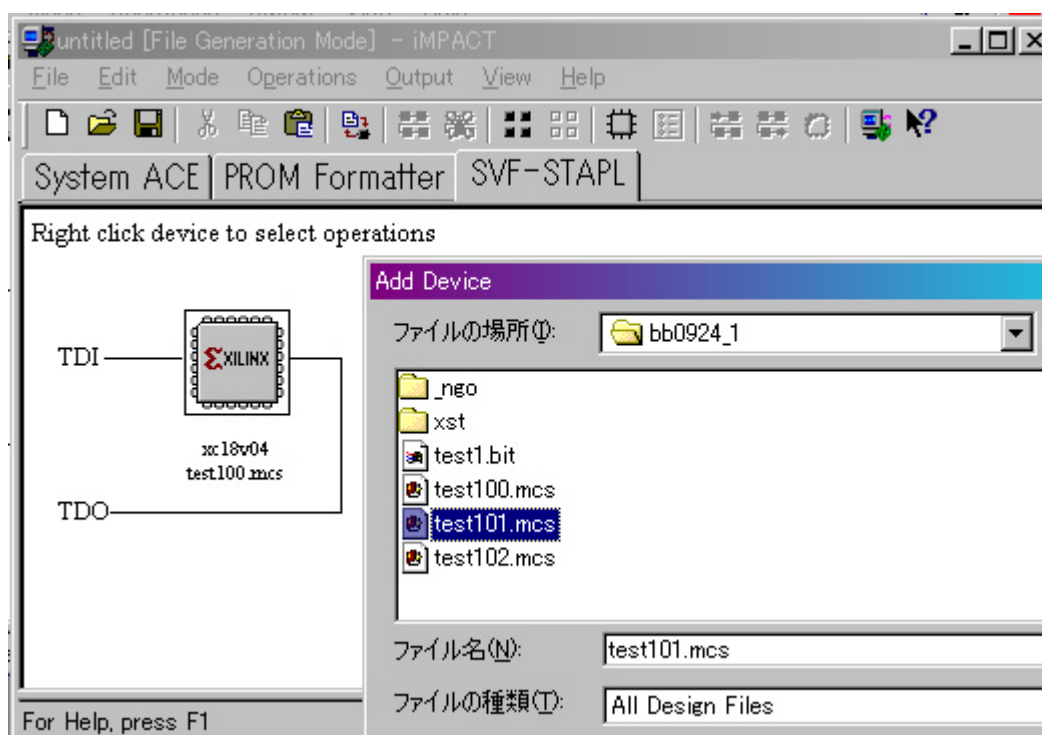
SVF に変換する MCS ファイルを選択します。

PROM ファイルを複数個に分割している場合は最初にデザイン名の後に '00' が付いた mcs ファイルを選択します。

(ここまでの手順は通常の手順と変わりませんので 3.2 又は 3.3 章を参照ください。)



PROM ファイルを分割している場合は Add Device をクリックして残りの msc ファイルも順番に選択します。





## 5. FPGA を複数個チェーンする場合のダウンロードデータ作成法

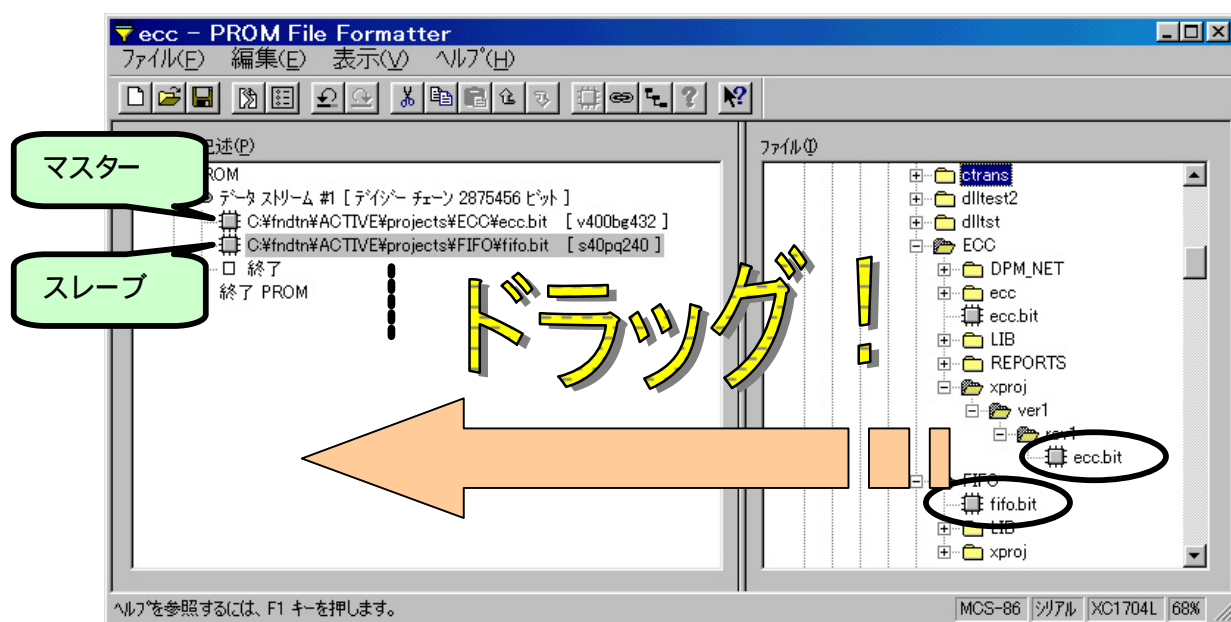
FPGA を複数個チェーンする場合、複数の bit データを一旦まとめてから、そのデータを iMPACT で SVF ファイルに変換する必要があります。

その場合、mcs(インテル HEX)または coe(モトローラ HEX)フォーマットに変換します。使用法を以下に記します。


PROM File Formatter を起動します。

スタート プログラム ザイリンクス ISE4 日本語版又は英語版 Accessories PROM File Formatter を選択。

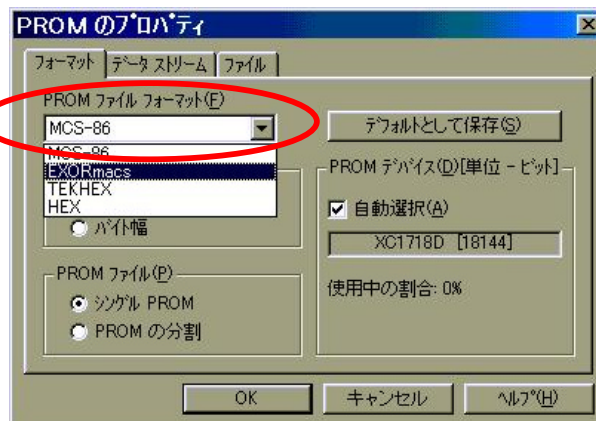
下の画面が立ち上がりますので、右のブラウザより bit データを参照し、左画面にドラッグ又はダブルクリックして下さい。



デバイスが複数個ある場合は、チェーンの順番通り(マスター、スレーブ 1、スレーブ 2...)に上から bit ファイルを並べる必要があります。

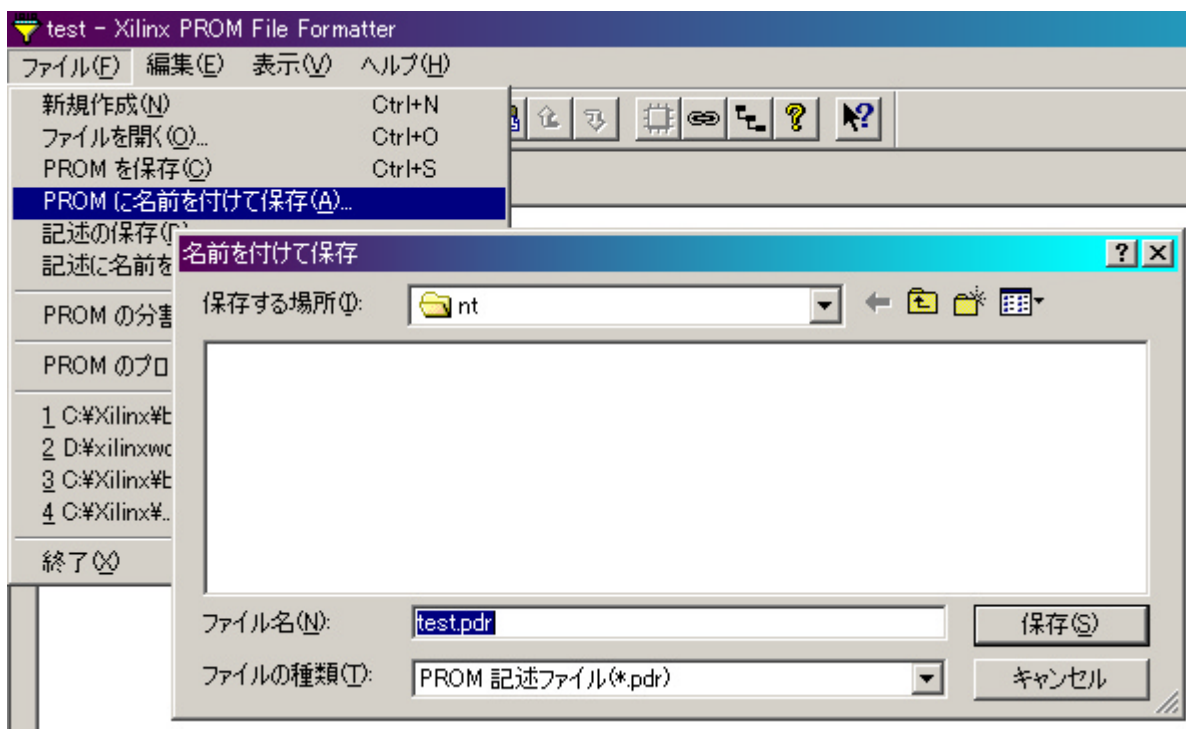
ファイル PROMのプロパティ又はアイコン  をクリックして下さい。

下の「PROMのプロパティ」が開きますので、ここで書きこみデータのフォーマット(MCS-86)を設定します。



PROM デバイスは 18V04 を選択します。

作成したデータを保存します。



後は、このデータを使用して通常通りの手順で SVF データを作成して下さい。

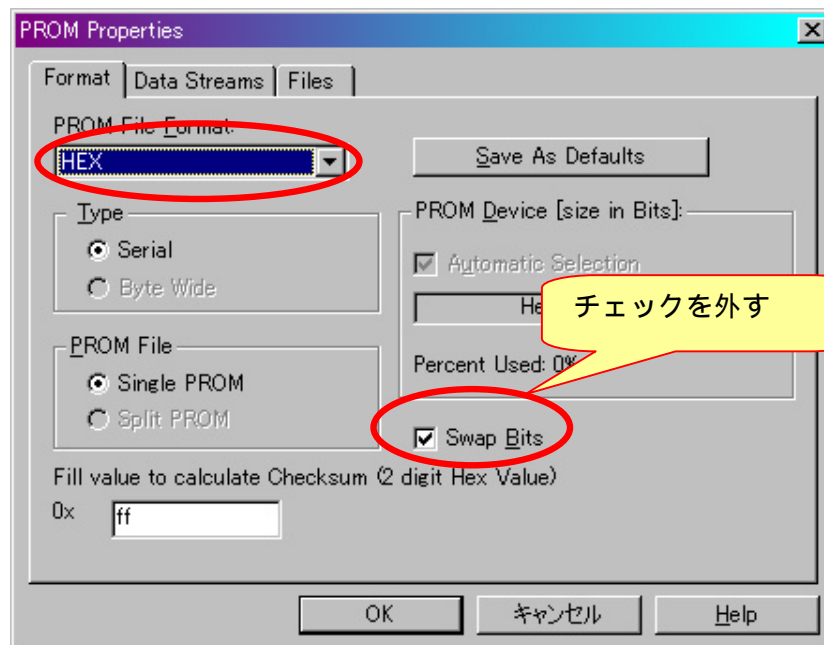


## 6. ROMライターでFlashメモリに書き込む際のデータ作成手順

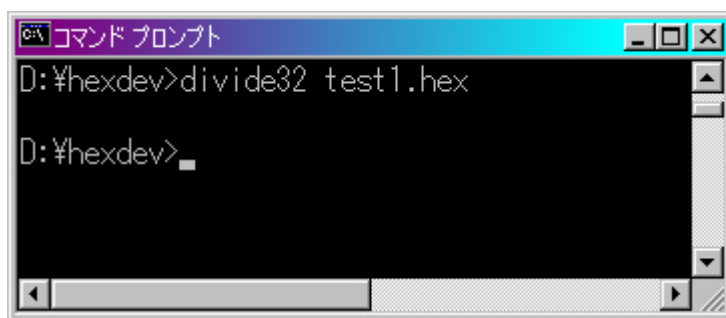
ROMライターでFlashメモリへ直接データを書き込む方法について記述します。

この場合、Xilinx社の開発ツールから出力するファイルフォーマットをHEXファイルとする必要があります。この場合、出力されたHEXファイルにはリターンコードが含まれていないため、弊社から無償提供するユーティリティを使用してHEXファイルを加工する必要があります。

PROM File Formatterの“PROM Properties”で“PROM File Format”を「HEX」に、また“Swap Bits”のチェックを外します。後はこの設定で通常通りROMデータを作成、保存してください。



作成したHEXデータを弊社から無償提供するユーティリティ「divide32.exe」を使用してリターンコードを挿入します。（Hexファイルに対し32キャラクタごとにリターンコードを入れます。）MS-DOSプロンプトで「divide32.exe」があるディレクトリに移動しコマンド「divide32 ファイル名」でdiv\_ファイル名.hexというファイル名のリターンコードが挿入されたHEXデータが作成されます。



div\_test1.hex というファイルが生成されます。



## 7. XSVF ファイル (バイナリ-フォーマット) 作成方法

これまでのフローにより作成された SVF データを TE7720 を介して Flash メモリに書き込む場合、SVF データを XSVF ファイルにフォーマットした上でダウンロードする必要があります。

DOS プロンプトを起動し、svf2xsvf プログラムを実行して下さい。

<<<コマンド実行例>>>

```
>svf2xsvf -i aaa.svf -o aaa.xsvf
```

-i : 入力ファイルの指定  
-o : 出力ファイルの指定

## 8. XSVF ファイルのダウンロード方法

7.で作成されたファイルをダウンロードすることにより、Flash メモリに対して Erase、Program、(Verify)を行います。ダウンロードには、弊社から提供する LBPlayer (playlb.exe) を使用します。

DOS プロンプト上で、playlb コマンドを実行して下さい。

<<<コマンド実行例>>>

```
>playlb aaa.xsvf
```

注意 : Erase、Program、Verify を個別に行う場合は、3.章に示す SVF データ作成の項で、個別の SVF データを作成する必要があります。

## 9. バッチコマンド例

bit データから MCS データ作成後、SVF データを作成、XSVF ファイルへ変換、TE7720 を介して Flash メモリへのダウンロードをバッチで行う場合の例を示します。各コマンドのオプションについては、Xilinx 社の Web サイトでご確認ください。

ターゲットデバイス : 2s150

slot.bit というファイルを XSVF ファイルへ変換し、FlashROM へのダウンロードを行います。この時、playb コマンドでは、ターゲット基板の準備ができたことを確認するため、Key 入力 (リターン) が必要になります。

<<<コマンド実行例>>>

>bit2svf.bat (リターン)

```
##### bit2svf.bat #####
promgen -p mcs -o b_slot.mcs -w -u 0 slot.bit
impact -batch mcs2svf.cmd
svf2xsvf -i b_slot.svf -o b_slot.xsv
playlb b_slot.xsv
```

```
##### mcs2svf.cmd #####
setmode -bscan
setcable -p svf -file b_slot.svf
addDevice -p 1 -sprom xc18v04 -file b_slot.mcs
program -e -p 1
quit
```

ターゲットデバイス : XC2V3000

slot\_2v3.bit というファイルを XSVF ファイルへ変換し、FlashROM へのダウンロードを行います。この時、playlb コマンドでは、ターゲット基板の準備ができたことを確認するため、Key 入力 (リターン) が必要になります。

<<<コマンド実行例>>>

>bit2vsvf\_2v3k (リターン)

```
#####bit2svf_2v3k.bat#####
promgen -p mcs -x XC18V04 XC18V04 XC18V04 -o b_2v3k00.mcs b_2v3k01.mcs b_2v3k02.mcs -w
-u 0 slot_2v3.bit
impact -batch mcs2svf_2v3k.cmd
svf2xsvf -i b_2v3k.svf -o b_2v3k.xsv
playlb b_2v3k.xsv
```

%%% slot\_2v3.bit という bit ファイルから、3 つの mcs ファイルを生成しています。

```
#####mcs2svf_2v3k.cmd#####  
setmode -bscan  
setcable -p svf -file b_2v3k.svf  
addDevice -p 1 -sprom xc18v04 -file b_2v3k00.mcs  
addDevice -p 2 -sprom xc18v04 -file b_2v3k01.mcs  
addDevice -p 3 -sprom xc18v04 -file b_2v3k02.mcs  
addDevice -p 4 -sprom xc18v04 -file b_2v3k02.mcs  
program -e -p 1  
program -e -p 2  
program -e -p 3  
quit
```

%%% 3つのmcsファイルから、b\_2v3k.svfというsvfファイルを生成しています。この時、4つ目のmcsファイルはダミーでチェーンに入れておき、プログラムは実行していません。

## 10. FPGA/Configuration Data 以外のダウンロード例 (CPU プログラム等)

TE7720 は、Xilinx 社 iMPACT が出力する SVF データを使用して FlashROM にデータをダウンロードします。この時、SVF データは bit データから MCS を作成し、それを SVF データに変換することを前提としていますが、以下のように CPU のプログラムをインテルヘキサ (MCS) タイプのファイルとして ISE ツールに読み込ませることにより、FPGA/Configuration Data に見立てて FlashROM にダウンロードすることが可能です。この時、TE7720 の仕様により、ダウンロードする FlashROM の領域は、4Mbit のバウンダリに限定されます。TE7720 がサポートする FlashROM のバウンダリについては、TE7720\_データシートの「6-4. データセットセレクション機能」を参照して下さい。

一般的な MCS ファイルを Xilinx 社 iMPACT が読み込めるフォーマットに変換するためのユーティリティ・ソフトを用意しています。(cnv3.exe)

### 10-1. Binary データをインテルヘキサ・ファイル (MCS) に変換。

一般的な ROM ライター等で読み込んだ Binary データをインテルヘキサ・ファイルとして保存すると、1 行あたり 32 バイトのデータを含むフォーマットで出力されるようです。

<一般的な MCS ファイル>

```
:2000000000000000010001000200020003000300050005000600060007000700090009009E
:200020000A000A000B000B000D000D000E000E000F000F00120012001300130015001500CE
:200040001600160017001700190019001A001A001B001B001D001D001E001E001F001F00F6
```

:	スタートマーク
先頭 1 バイト	バイト数 (20=32 バイト、10=16 バイト)
次 2 バイト	ロードアドレス
次 1 バイト	レコードタイプ
次 ~ まで	データ
チェックサム	~ を 16 進で加算し、2 の補数をとる (1 Byte)

このファイルを iMPACT が読み込めるような MCS のフォーマットに変換します。DOS プロンプトを開き、以下のコマンドを入力します。

```
>cnv3 BP32_16M.mcs output
```

%%% これにより、BP32\_16M.mcs ファイルをフォーマット変換し、4Mbit を超えるサイズの場合は 4Mbit ごとのファイルに分割します。例えば 16Mbit のサイズの場合、4Mbit のファイルが 4 個できます。

```
◇ output.000
◇ output.001
◇ output.002
◇ output.003
```

それぞれのファイルを test6\_00.mcs、test6\_01.mcs、test6\_02.mcs、test6\_03.mcs のように、ファイル名.mcs と名前を変更して下さい。順番は、iMPACT で読み込む場合のチェーンの順序をあらわしますので、必ず出力された番号とファイル名の番号を一致させて下さい。



## 10-2. iMPACT への読み込み

10-1.で作成した MCS ファイルを iMPACT に読み込みます。方法は、bit データから作成した MCS データを読み込む方法と同じです。3 章「SVF データの作成」の章を参照下さい。

## 11. 参考情報

### 11-1. ダウンロード時間

LBPlayer によるダウンロード時間は、PC の OS 種別、CPU の種類、周波数、また PC の状態により変化します。以下に参考情報として実測値を記述しますのでご参照下さい。

ターゲット : XC2S150 ( Configuration Data:約 1Mbit )

OS	CPU	ダウンロード時間	
		Erase+Program	Erase+Program+Verify
Win2K	Pentium -500MHz	25sec	31sec
Win2K	Pentium -1.5GHz	18sec	24sec
Win2K	K6-500MHz	24sec	30sec
Win98ME	Duron-750MHz	20sec	28sec
Win98SE	K6-500MHz	16sec	21sec
WinXP	Pentium -1.0GHz	16sec	20sec
WinXP	Pentium -1.5GHz	17sec	22sec
WinNT	K6-500MHz	21sec	28sec

ターゲット : XC2V3000 ( Configuration Data:約 10Mbit )

イレース、プログラム、ベリファイ全てを実行

OS	CPU	ダウンロード時間
Win2K	Pentium -1.5GHz	207sec
Win98ME	Duron-750MHz	247sec
Win98SE	K6-500MHz	219sec
Win98SE	Pentium -1.7GHz	137sec
Win98SE	Pentium -1.0GHz	115sec

\* XC2V3000 の場合、イレース+プログラム (ベリファイ無し) のダウンロード時間は上記のおよそ半分になります。

### 11-2. Configuration 時間

TE7720 を用いて FPGA の Configuration を行う場合の実測値です。

ターゲット : XC2V3000 (Configuration Data:約 10Mbit)


CCLK 周波数	Configuration 時間
4MHz	10 秒
34MHz	1.2 秒
41MHz	1.1 秒 (参考値)

## TE7720-APP001 改版履歴

Revision	Author	Date	Note
1.00	木俣	2002/09/27	初版
1.01	木俣	2002/10/11	P1,P2,P28 誤記訂正 TE4300 TE7720
1.02	木俣	2002/10/28	P21 ROMライターでFlashメモリに書き込む際のデータ作成手順の誤記修正。 「“Swap Bits”のチェックを入れます」という記述を「外します」に変更。 P28 ダウンロード時間、Config 時間の実測値（参考値）を記載





 **東京エレクトロン デバイス株式会社**

TOP マーケティンググループ

〒224-0045

神奈川県横浜市都筑区東方町 1 番地

TEL 045-474-7013

FAX 045-474-5617

E-mail top@teldevice.co.jp

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

お問い合わせは下記営業担当までお願い致します。

- ・ この資料の記載内容は、予告なしに変更することがあります。ご使用を検討の際には弊社担当までご確認ください。
- ・ この資料に記載された情報・図面の使用に起因する第三者の特許権、工業所有権、その他の権利侵害について、弊社はその責任を負うものではありません。
- ・ 本製品を他の製品と組み合わせて、または他の製品に組み込んで使用される二次製品、三次製品等について、第三者との間に特許権、実用新案権、回路配置利用権、著作権、その他の知的財産権の関する紛争が発生した場合には、本製品をご使用される方の責任において処理、解決してください。
- ・ 本製品は、生命維持装置、原子力制御装置、航空宇宙機、輸送機器等の極めて高い信頼性が要求される装置用に製造されておりません。