

# *ispVM System*

# ユーザーマニュアル

# 目次

1	ispVM System(基本編) .....	4
1.1	概要 .....	4
1.2	ispVM Systemの主な機能・特徴 .....	4
1.3	ispVM systemの起動方法 .....	5
1.4	ダウンロードケーブルの種類 .....	6
1.5	ダウンロードケーブル接続時の注意事項 .....	6
1.6	デバイスのスキャン .....	7
1.7	ダウンロードケーブルとI/Oポートの設定 .....	7
1.8	USBダウンロードケーブルドライバのインストール .....	9
1.9	デバイス情報の編集 .....	12
1.10	プログラミング(オペレーションの実行) .....	14
1.11	チェーン情報の保存 .....	15
2	ispVM System(応用編) .....	16
2.1	概要 .....	16
2.2	Dual Boot .....	16
2.2.1	Primary Image fileの書き換え .....	19
2.3	USERCODE/UES設定 .....	21
2.4	プログラミング中のI/Oステートの設定 .....	22
2.5	書き込み設定の変更 .....	24
2.6	ATEプログラミング・ベクタの生成 .....	26
2.7	SVFファイルの生成 .....	29
2.7.1	デバイス単体のSVFファイルの生成 .....	29
2.7.2	チェーン全体のSVFファイルの生成 .....	30
2.8	VMEファイルの生成 .....	31
2.8.1	VMEファイルの生成 .....	31
2.8.2	エンベデッド用Cソースファイルの生成 .....	33
2.9	CPU エンベデッド ビットストリームの生成 .....	34
2.10	コマンドラインモード .....	36
2.11	Universal File Writer .....	37
2.12	データの圧縮 .....	38
2.13	Application Specific BSDL File Generator .....	39
2.14	Model300 の使用方法 .....	40
3	トラブルシューティング(プログラミング関連) .....	42
3.1	ボード診断機能 .....	42

---

3.2	反復ダウンロード.....	43
3.3	ケーブルデバッグ機能.....	43
3.4	SVFデバッガ.....	45

## 1 ispVM System(基本編)

### 1.1 概要

ispLEVER を使用してプロジェクトをコンパイルした後、Lattice のデバイスをプログラムまたは、コンフィグレーションします。デバイスへのプログラムまたはコンフィグレーションは、ispVM System を用います。ispVM System は ispLEVER インストール時に ispVM System のみ単体でインストールすることも可能です。

ispVM System を単体で使用する場合は、弊社HP【<http://www.latticesemi.com/>】からダウンロードしてご利用ください。ispVM System は無料でご使用頂けます。

### 1.2 ispVM System の主な機能・特徴

- (1) ラティス全 isp デバイスのプログラミング・コンフィグレーション
- (2) 他メーカー混在チェーンの書き込みが可能
- (3) デバイスチェーンの自動スキャン
- (4) シリアル/ターボダウンロードをサポート
- (5) ユーザーコード/UES エディタ
- (6) ボード診断機能
- (7) プログラミング中の I/O ステート指定
- (8) VME ファイルの生成/VME-Processor
- (9) ATE プログラミング・ベクタジェネレータ
  - ・ HP, Genrad, Teradyne 及び Marconi ATE 装置をサポート
  - ・ 先進的 AET ベクタ生成機能
- (10) ビルトイン SVF エディタ
- (11) SVF ファイルのインタープリタとデバッグ
- (12) IEEE1532(ISC)データ・ジェネレータ
- (13) FPGA ローダー
- (14) コマンドラインモードを用意
- (15) XML ベースの新しいチェーン
  - ・ フォーマット(.XCF ファイル)
  - ・ パッケージ・タイプを含むデバイスのフル・ネーム
  - ・ TDI と TDO のチェーン接続・デバイス毎に選択されたオペレーション・ヒューズ
  - ・ チェックサム値やファイルのタイム・スタンプを含む JEDEC ファイル名
  - ・ デバイスのインストラクション・レジスタ長・デバイスの BYPASS オペコード(オール 1 でない場合)
  - ・ SVF ファイルのファイル名とタイム・スタンプ
  - ・ /ispEN、/ENABLE、/TRST、/BSCAN、/EPEN ピンとケーブルとの接続
  - ・ プリロード BSCAN ベクタをサポート(デフォルト I/O 値は HiZ)
  - ・ その他プロジェクト設定

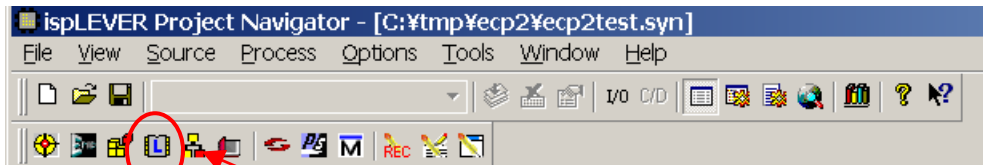
### 1.3 ispVM system の起動方法

Windows のスタートメニューから起動する

[スタート] [プログラム] [Lattice Semiconductor] [ispVM System]で起動します。

ispLEVER Project Navigator から起動する

ispLEVER Project Navigator のアイコンをクリックすることで起動します。



このアイコンをクリック

図 1-1 ispVM System アイコン

上記何れかの方法により、ispVM を起動しますと図 1-2のような画面が立ち上がります。

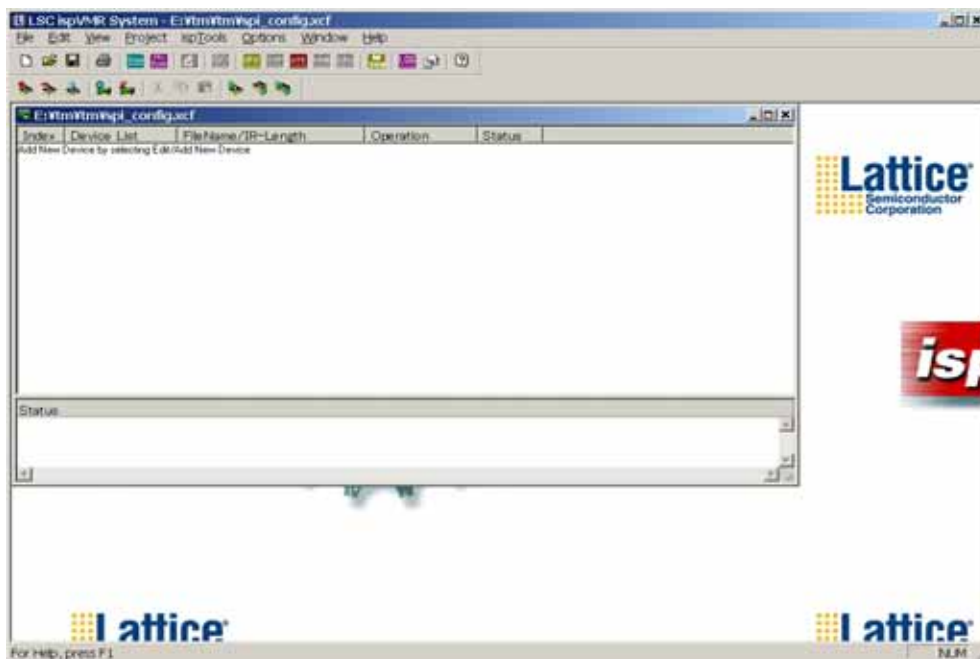


図 1-2 ispLEVER 起動時の画面

### 1.4 ダウンロードケーブルの種類

Lattice のダウンロードケーブルには、PC との接続ポートが USB タイプの物と、25pin パラレルポートタイプの物があります。

以下に、ダウンロードケーブルの外観図を示します。

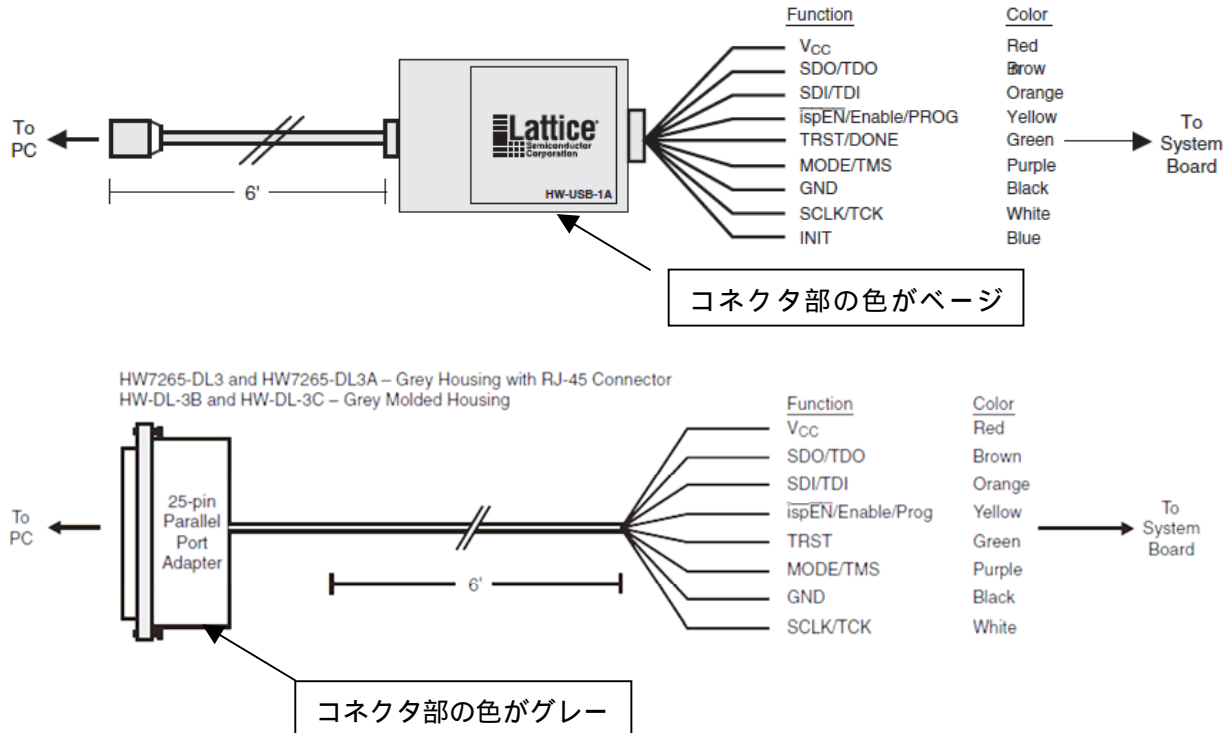


図 1-3 ダウンロードケーブル外観

ダウンロードケーブルの詳細および、旧式のダウンロードケーブルおよび旧 Vantis 製ケーブルについての詳細は、データシート【<http://www.latticesemi.com/lit/docs/devtools/dlcable.pdf>】をご参照ください。

### 1.5 ダウンロードケーブル接続時の注意事項

PC と基板をダウンロードケーブルで接続する際は、以下の点にご注意ください。

1. ダウンロードケーブルの抜き差しは、書き込み対象のボードの電源を切ってから行ってください。電源が供給された状態での抜き差しは、デバイスを破損する恐れがあります。
2. 書き込みが終了次第、ダウンロードケーブルを基板から外して下さい。パラレルポートもしくは USB ポートから何かしらの信号が出力されデバイスを誤動作させる恐れがあります。

## 1.6 デバイスのスキャン

基板と PC をダウンロードケーブルで接続後、基板の電源を投入し、デバイスのスキャンを行います。デバイスのスキャンは、以下の 3 通りの方法で可能です。

- ツールバー上のアイコンをクリック
- メニューの[ispTools] [Device Scan]で起動
- F2 キーを押す。

尚、ラティス ISP チェーンと JTAG - ISP チェーンを混在したチェーンの場合には、Mix Scan ボタンを押します。

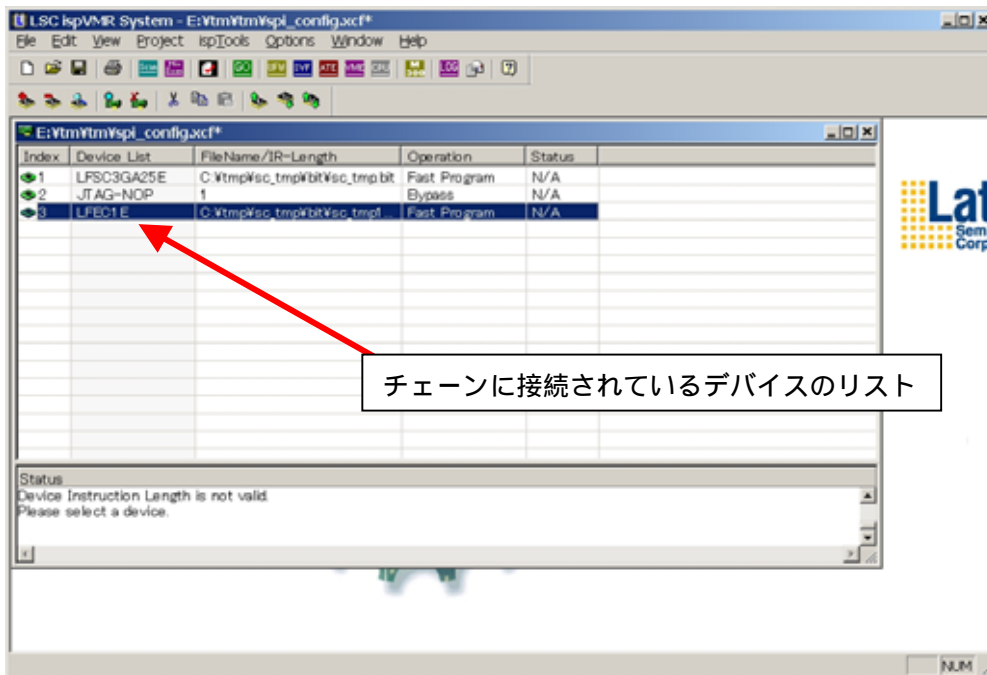


図 1-4 SCAN 実行後の画面

SCAN が正常に行われますと、図 1-4のようにチェーンに接続されているデバイス名が表示されます。

尚、他社デバイスは全て JTAG-NOP と表示されます。

## 1.7 ダウンロードケーブルと I/O ポートの設定

上記でデバイスの SCAN が上手くいかない場合、ダウンロードケーブルの設定がされていない可能性があります。その場合、以下の手順により、設定を行うことができます。

ispVM System のメニューから[Options] [Cable and I/O Setup]を選択します。

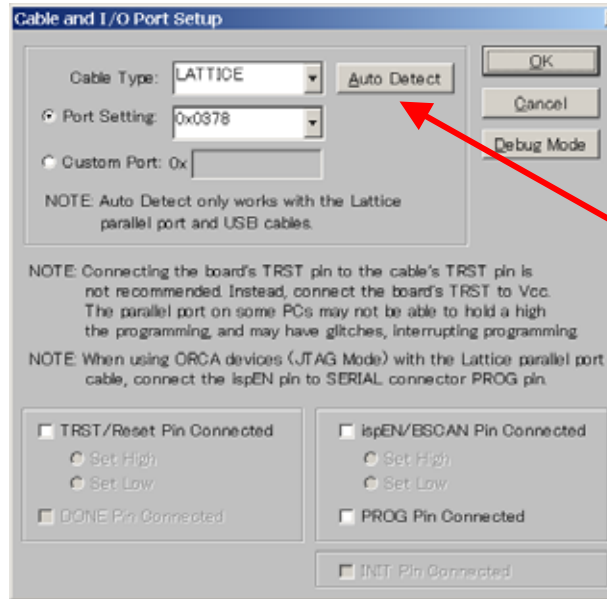


図 1-5 ダウンロードケーブルと I/O ポートの設定画面

各項目の内容および設定内容は、表 1-1を参照ください。

表 1-1 ダウンロードケーブルと I/O ポートの設定項目とその内容

項目	内容	設定内容
Cable Type	使用するダウンロードケーブルのタイプ	Lattice : パラレルポートケーブル Vantis : 旧 Vantis 製ケーブル USB : USB ポートケーブル
Port Setting	ケーブルが繋がっているポートの指定。	Auto Detect により設定可能。 (旧 Vantis 製は自動検出不可)
Custom Port 0x	ケーブルが接続されているポートをカスタム指定	繋がっているポートを指定
TRST/Reset Pin Connected	ダウンロードケーブルに TRST ピンを接続するか、接続しないかの設定	ON : 接続する Set High : Active High Set Low : Active Low OFF:接続しない。
Done Pin Connected	ダウンロードケーブルに DONE ピンを接続するか、接続しないかの設定	ON : 接続する OFF:接続しない。
ispEN/BSCAN Pin Connected	ダウンロードケーブルに ispEN/BSCAN ピンを接続するか、接続しないかの設定	ON : 接続する Set High : Active High Set Low : Active Low OFF:接続しない。
Prog Pin Connected	ダウンロードケーブルに Prog ピンを接続するか、接続しないかの設定	ON : 接続する OFF:接続しない。
INIT Pin Connected	ダウンロードケーブルに INIT ピンを接続するか、接続しないかの設定	ON : 接続する OFF:接続しない。

**Windows XP で USB ダウンロードケーブルを使用する場合、上記方法による自動検出が上手くいかない場合があります。その場合は、次頁に示します手順にてドライバをインストールしてください。**



## 1.8 USB ダウンロードケーブルドライバのインストール

**本作業は必ず Admin 権限で行ってください。**

まず、準備として ispVMsystem を正常にインストールしてください。

インストール後、図 1-6 のような USB ドライバのインストール確認画面が起動しますので、「はい」を押します。



図 1-6 USB ドライバのインストール確認画面

インストールが完了しますと、図 1-7 のような画面が起動します。



図 1-7 USB ドライバインストール完了画面

インストールが完了しましたら、USB ダウンロードケーブルを PC に接続します。自動でのような Wizard 画面が起動します ( Wizard 画面が起動しない場合は、自動で認識しておりますので、以降の作業は不要です )。

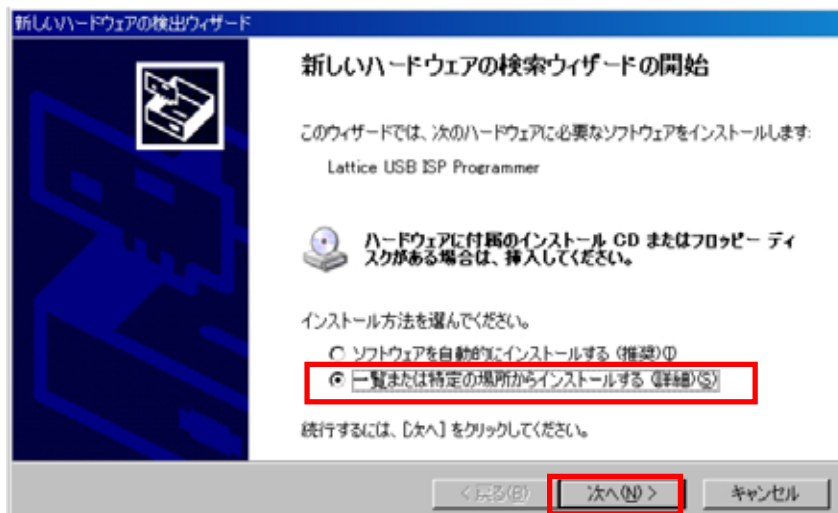


図 1-8 新しいハードウェアの検出ウィザードの開始画面

上記画面が起動しましたら、「一覧または特定の場所からインストールする」を選択し、「次へ」をクリックします。

続いて図 1-9のような画面が起動しますので、「次の場所で最適のドライバを検索する」内の「次の場所を含める」を選択し、「C:\WINDOWS\system32」を指定します。

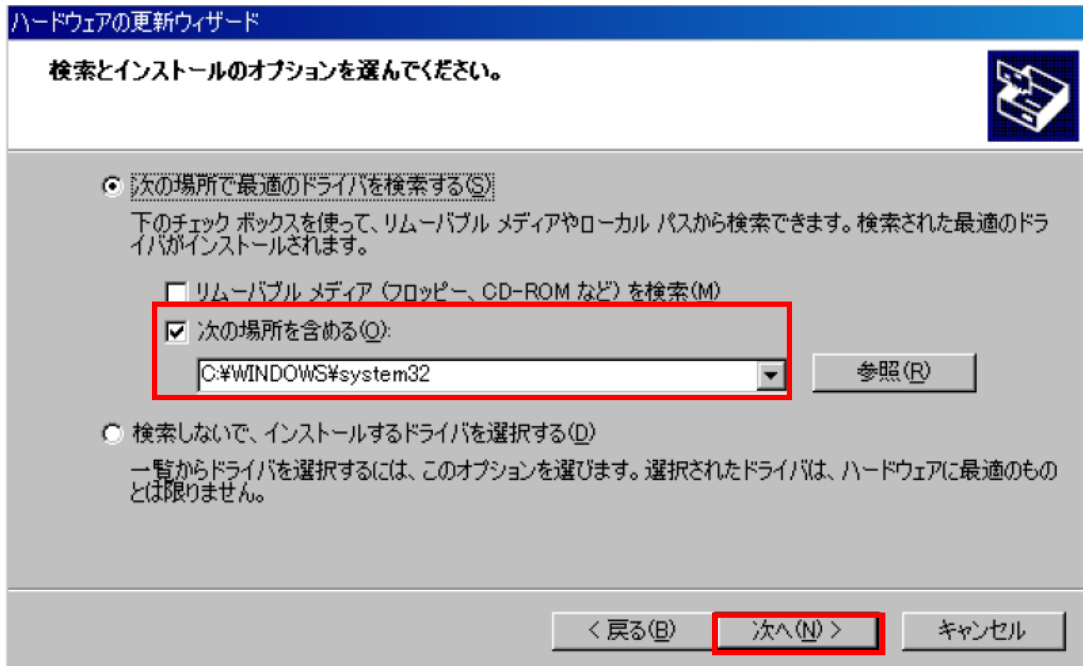


図 1-9 ハードウェアの更新ウィザード

「次へ」をクリックしますと、図 1-10のような画面が起動します。表示に「Lattice USB ISP Programmer」と記載されていることを確認し、「続行」をクリックしてください。

「Lattice USB ISP Programmer」と記載されていない場合は、ispVMsystem インストール時にドライバが正常にインストールされていないか、インストールしていない可能性がございます。

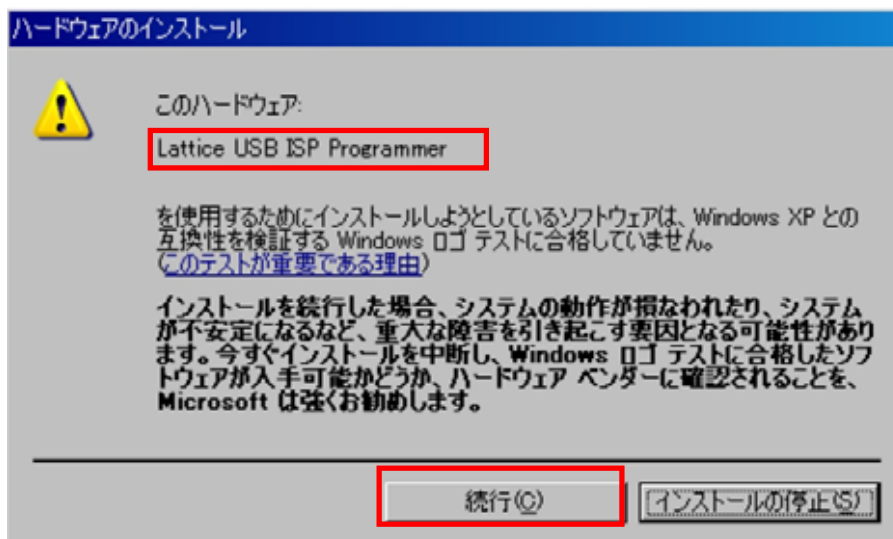


図 1-10 ハードウェアのインストール

「続行」をクリックした後、図 1-11のような画面が起動すればインストールは正常に終了しております。「完了」をクリックして、画面を閉じます。



図 1-11 ハードウェアの更新ウィザードの完了

インストールが完了しましたら、念のため、次に示します手順により確認を行ってください。

**確認手順**

Windows のスタートメニューから「設定」、「コントロールパネル」を選択します。コントロールパネル内の「システム」をクリックします。システムからハードウェアを選択し、デバイスマネージャを開きます。デバイスマネージャ内の USB コントローラを開いて、「Lattice USB ISP Programmer」が表示されていることを確認します。

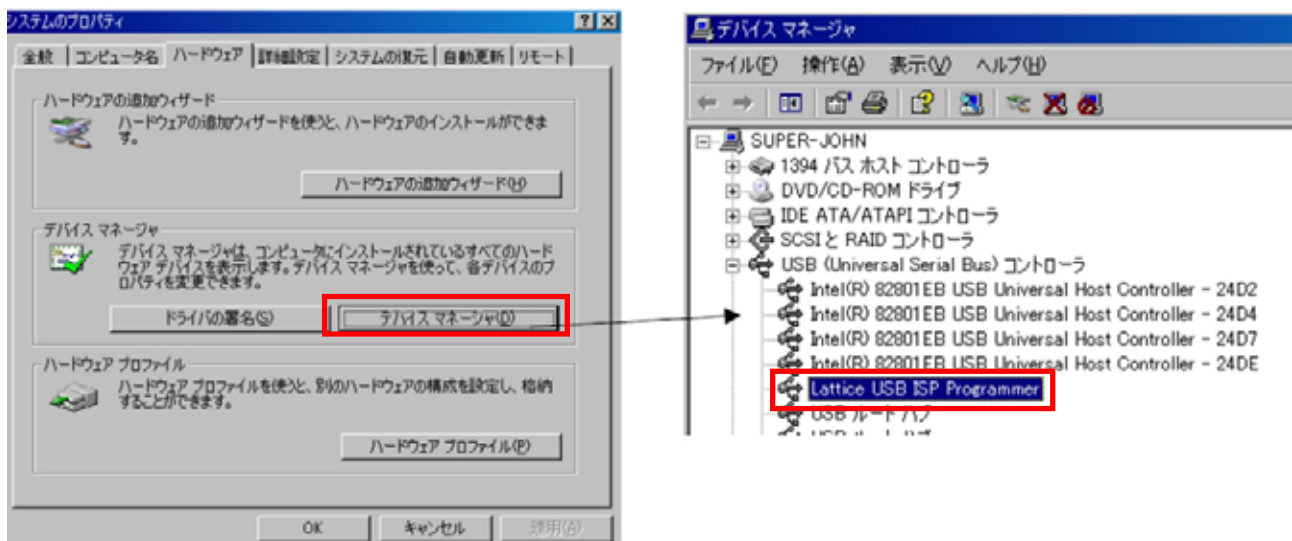


図 1-12 USB ドライバインストール完了の確認

### 1.9 デバイス情報の編集

図 1-4のように SCAN が成功した画面にて、編集したデバイスの行をダブルクリックしますと、図 1-13のようなデバイス情報の編集画面が開きます。

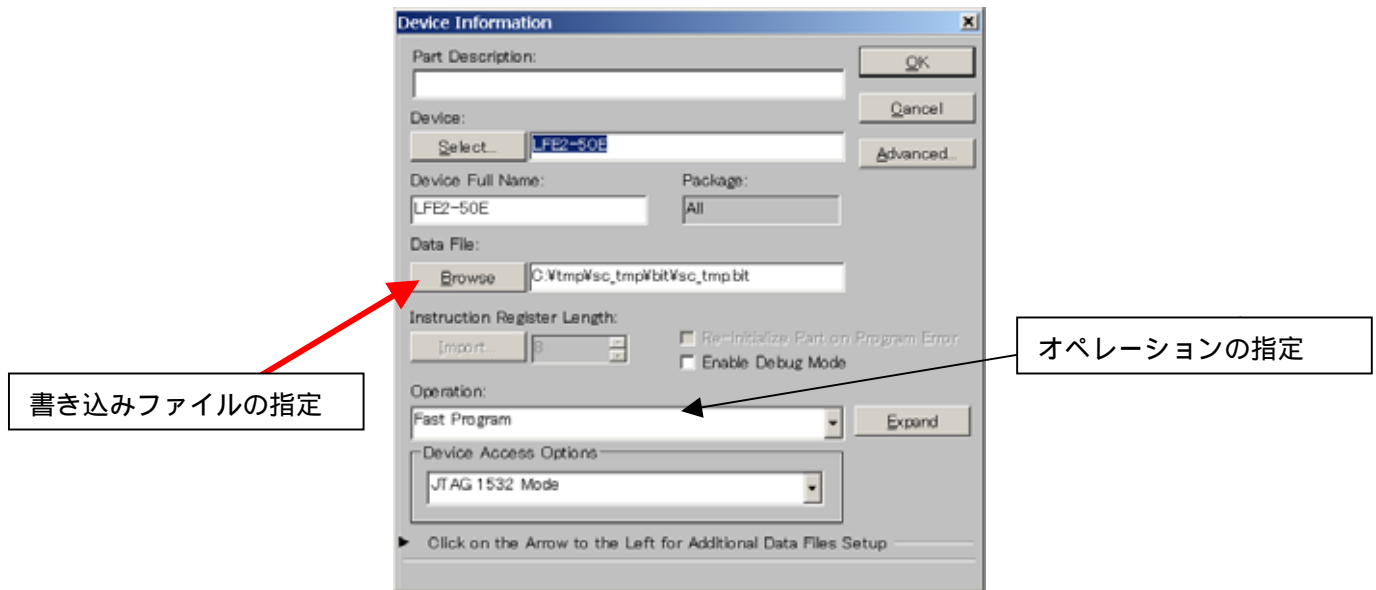


図 1-13 デバイス情報の編集画面

上記画面にて、最初に書き込みファイルに指定を行います。書き込みファイルの指定は、Data File 設定欄の「Browse」ボタンを押し、ファイルの指定を行います。

EC/ECP/ECP2/SC ファミリの場合は、拡張子が bit もしくは rbf を、XP/XO ファミリの場合は、拡張子が isc もしくは jed のファイルを指定します。

書き込みファイルの指定が完了しましたら、オペレーション項目の設定を行います。

主なオペレーションの項目とその内容について表 1-2に示します。

表 1-2 オペレーション項目と内容

オペレーション項目	内容
Fast Program Erase, Programming, Verify	消去、書き込み、ベリファイを実行します
Erase, Programming, Verify, Secure	消去、書き込み、ベリファイを実行後、セキュリティをかけます。(*1)
Verify Only	ベリファイのみ実行します。
Verify ID	デバイスコードをベリファイします。
Erase Only	消去のみ実行します。
Display ID	デバイスからデバイスコードを読み出し、表示します。
Bypass	何も実行しません。
Calculate Checksum	チェックサム値を計算し、表示します。
Verify USERCODE	デバイス内のユーザコードをベリファイします。
Display USERCODE	デバイス内のユーザコードを読み出し、表示します。
Read and Save JEDEC	デバイス内のデータを読み出し、保存します。
Read DONE bit	DONE bit の値を読み出し、表示します。

\*1 セキュリティをかけた場合、読み出しデータにセキュリティがかけられます。これにより、Verify, Checksum の確認ができなくなりますので、ご注意ください。

次に、Device Access Options の設定を行います。設定内容に対する実行内容は、表 1-3～表 1-5の通りとなります。

表 1-3 Device Access Options の設定内容(EC/ECP/ECP2 ファミリ)

項目	内容
JTAG 1532 Mode	デバイスがプログラムモード時に JTAG ポートを使用
Static RAM Cell Background	デバイスがユーザーモード時に、リードまたはベリファイを可能とする
SPI Flash Programming	JTAG 経由で SPI フラッシュメモリの書き込みを実行
Advanced SPI Flash Programming	FPGA ローダ機能を使用して、SPI フラッシュメモリの書き込みを実行
Serial Mode	デバイスのシリアルコンフィグレーションを実行

表 1-4 Device Access Options の設定内容(SC ファミリ)

項目	内容
JTAG Mode	JTAG ポートを使用
Serial Mode	シリアル・コンフィグレーションポートを使用

表 1-5 Device Access Options の設定内容(XP/XO ファミリ)

項目	内容
Flash Programing Mode	内蔵フラッシュメモリのプログラムを実行
Flash Background Mode	デバイスがユーザーモード時に、内蔵フラッシュメモリのバックグラウンドプログラム、再プログラムを可能とする
Static RAM Cell Mode	SRAM にアクセス
Static RAM Cell Backgournd Mode	デバイスがユーザーモード時に、SRAM セルのバックグラウンドリード、確認を実行

### 1.10 プログラミング(オペレーションの実行)

チェーンの設定、デバイスの設定が完了した状態で、メニューアイコンの「GO」ボタンをクリックしますと、プログラミング(設定したオペレーション)が開始されます。

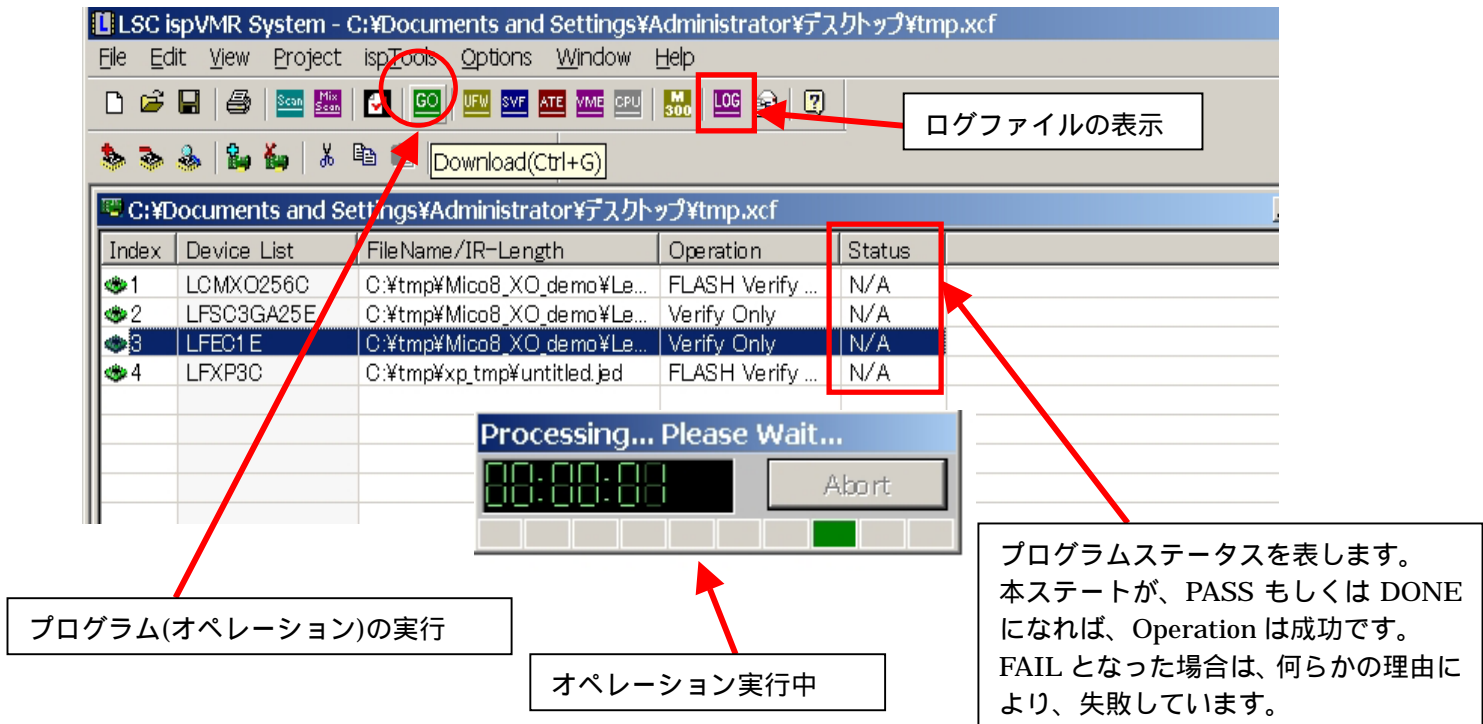


図 1-14 オペレーションの実行

「GO」ボタンをクリック後、ステータス欄に PASS もしくは DONE と表示されれば、プログラミング(オペレーション)が正常に完了しています。FAIL と表示された場合には、何らかの理由により、オペレーションが失敗しています。

オペレーション失敗の原因が不明の場合には、次章 ispVM(応用編)のトラブルシューティングの項を参照ください。

オペレーション終了後、「LOG」ボタンをクリックしますと、ログファイルを参照することができます。また、[Edit] [Clear Log File]でログをクリアすることができます。

### 1.11 チェーン情報の保存

作成した、JTAG チェーンの情報、以下の 2 つの方法により保存することができます。

ispVM Systemu のメニューから [File] [Save] を選択する。  
Save アイコンをクリックする。

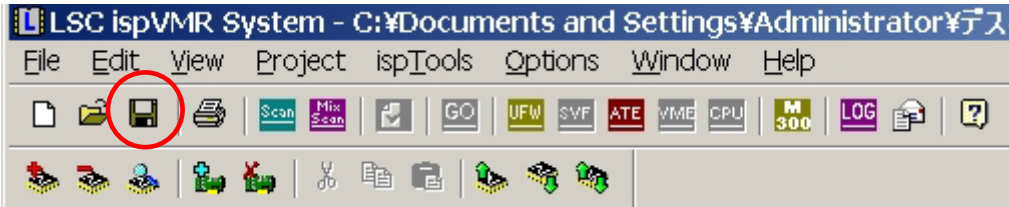
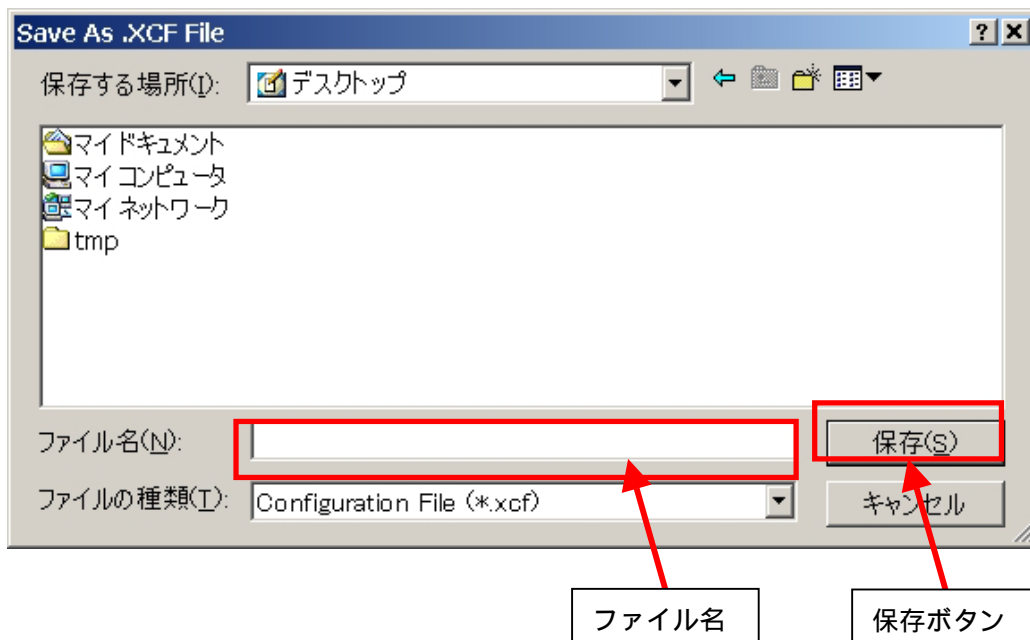


図 1-15 チェーン情報の保存

上記何れかを実行後、ファイル名をつけて保存します。尚、拡張子は xcf となります。



保存したチェーンファイルは、spVM Systemu のメニューから [File] [Save] を選択することで、次回以降同じ設定で使用することが可能となります。

これで、ispVM System 使用ガイド(基本編)は終了です。

ここに記載されていない、応用機能に関しては、次項の ispVM System 使用ガイド(応用編)を参照ください。

## 2 ispVM System(応用編)

### 2.1 概要

本節は、ispVM 機能の応用機能について説明いたします。

### 2.2 Dual Boot

Lattice ECP2/ECP2M ファミリでは、SPI Flash を利用した Dual Boot 機能の使用が可能です。本節では、ispVM System を使用したデュアルブート用書き込みファイルの設定方法、書き込み方法について説明致します。

ボードに電源を供給し、ダウンロードケーブルを接続した状態で ispVM を起動し、SCAN ボタンをクリックしますと、図 2-1のようにボードに実装されている ECP2/M デバイスが Device List に表示されます。

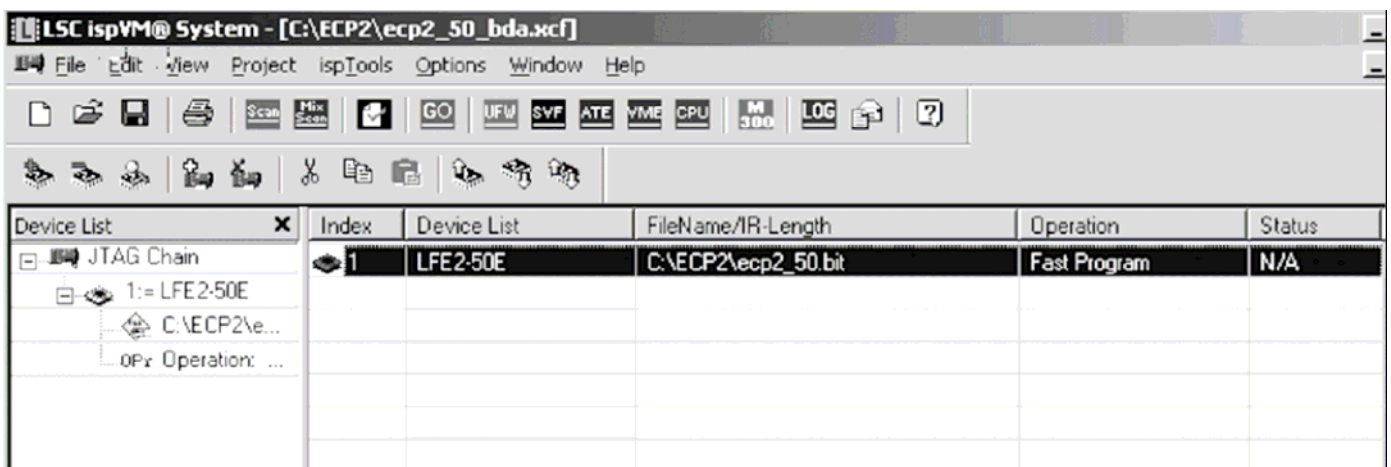


図 2-1 ispVM System メイン画面

デバイスが正常に表示されている状態で該当行をダブルクリックしますと、図 2-2のような画面が起動します。



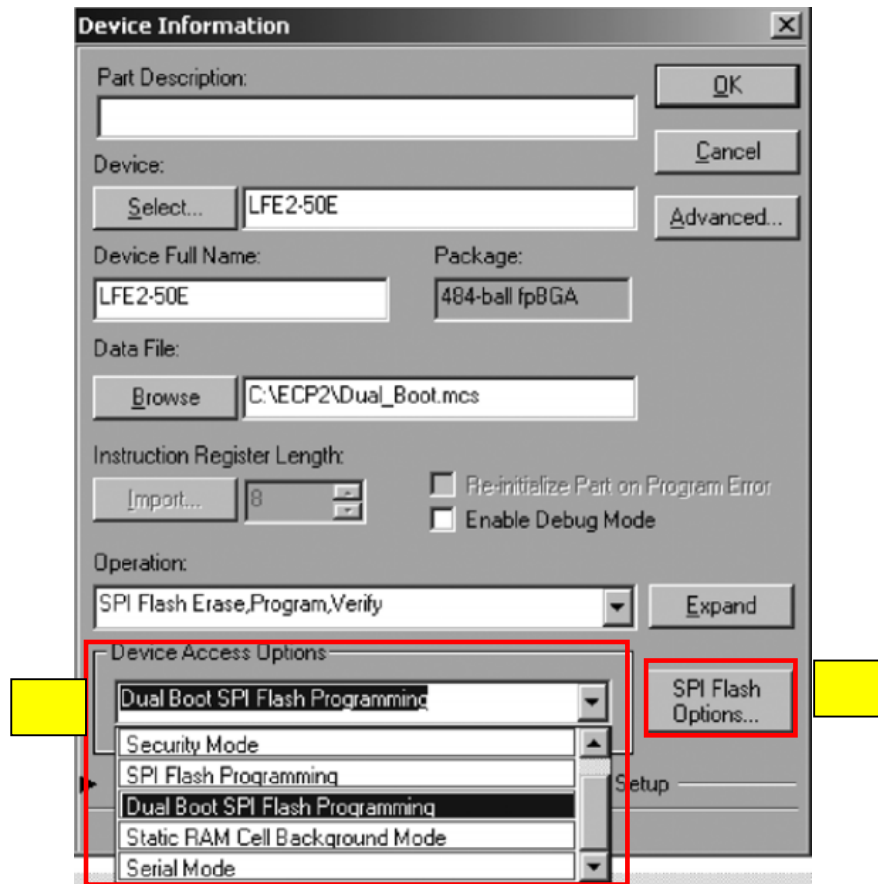


図 2-2 デバイス情報画面

上記画面が起動しましたら、Device Access Options のプルダウンメニューから「Dual Boot SPI Flash Programming」を選択し、「SPI Flash Options」ボタンをクリックします。

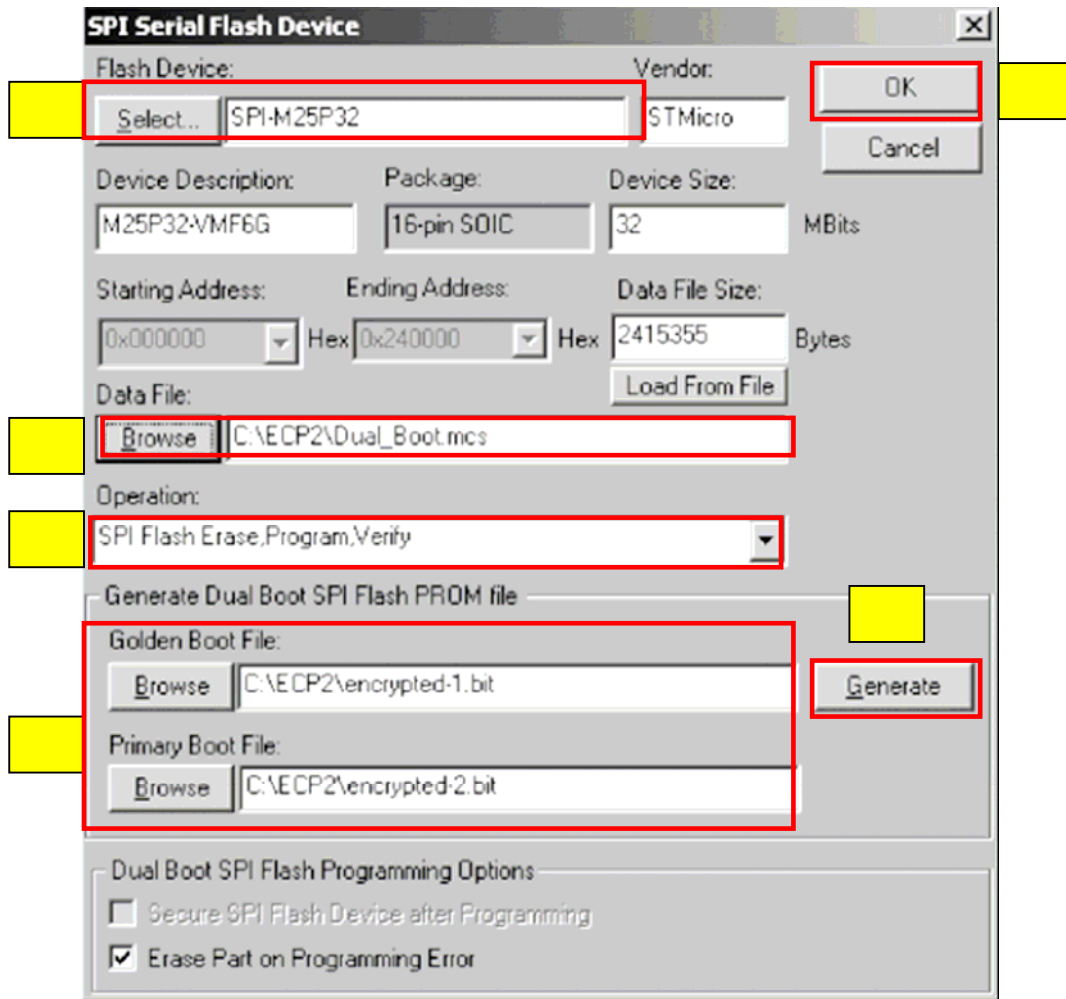


図 2-3 SPI Flash 設定画面

上記画面が開きましたら、下記 6 つの設定を行います。

Flash Device 欄の Select ボタンをクリックし、書き込みを行う SPI Flash の型名を選択します。  
Data File 欄の Browse ボタンをクリックし、書き込みファイルの生成場所、ファイル名を設定します。

Operation を「SPI Flash Erase,Program,Verify」に設定します。

Golden Boot File および Primary Boot File を選択します。

**Golden Boot File : Primary boot file によるコンフィグレーションエラー時に読み込まれる、コンフィグレーションファイル**

**Primary Boot File : 優先的に読み込まれるコンフィグレーションファイル**

Generate ボタンをクリックし、書き込みファイルを生成します。

生成が正常に完了した旨のメッセージが表示されましたら OK ボタンをクリックします。

以上で書き込みファイルの設定は完了です。

## 2.2.1 Primary Image file の書き換え

本節では、ispVM を使用して Primary Image File のみを書き換える方法を示します。前ページに記載の方法で、Dual Boot 用書き込みファイルを生成しますと、Log ファイルに Primary Image と Golden Image のアドレスが記載されます。



図 2-4 LOG ファイルの表示

書き込みファイル生成後、ispVM メイン画面にて「LOG」ボタンをクリックしますと、図 2-5のようにログファイルが表示されます。

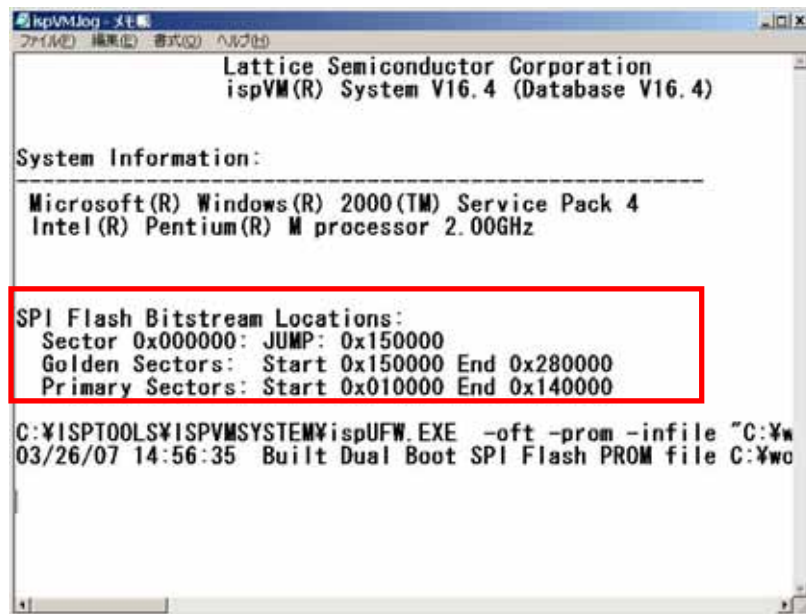


図 2-5 ログファイル

上図のように Golden および Primary の Start アドレスおよび Enc アドレスがレポートされます。Primary のみ書き換えるには、上記アドレスを指定して書き込みを行いますので、Primary のアドレスを記録しておきます。

2.2節と同様の手順にて SPI Flash の設定画面を開きます。

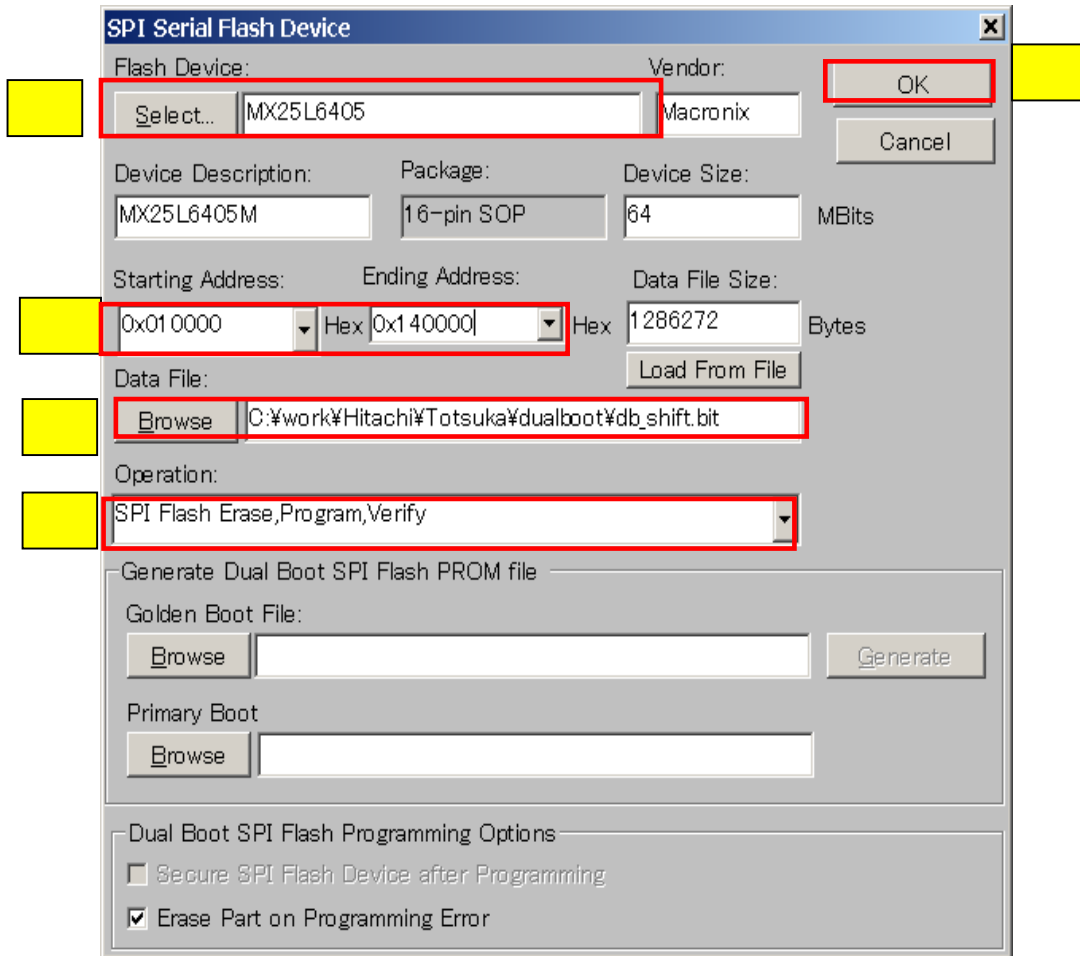


図 2-6 SPI Flash 設定画面

上記画面が開きましたら、下記 6 つの設定を行います。

Flash Device 欄の Select ボタンをクリックし、書き込みを行う SPI Flash の型名を選択します。

Data File 欄の Browse ボタンをクリックし、書き込みファイルの生成場所、ファイル名を設定します。

Operation を「SPI Flash Erase,Program,Verify」に設定します。

ログファイルの情報をもとに、Starting Address および Ending Address を指定します。

OK ボタンをクリックします。

以上で設定は完了です。

### 2.3 USERCODE/UES 設定

ラティス FPGA では、デバイス内にユーザー固有のコード、USERCODE を格納することが可能です。USERCODE の設定は、下記手順により行います。

- 1 .ispVM System を起動し、メニューの[Option] [Display USERCODE/UES Options]で、基数を設定します。
2. Device Information 画面を起動し、[Advanced]をクリックし、[Advanced Device Information]画面を開きます。
3. [USERCODE/UES Field]に で設定したフォーマットで、UES を記述し、[Apply UES] をクリックします。
4. [Auto-Increment After Program by]を 1 以上の数に設定しますと、プログラムごとに設定した数が自動でインクリメントされます (0 の場合は毎回同じです)。

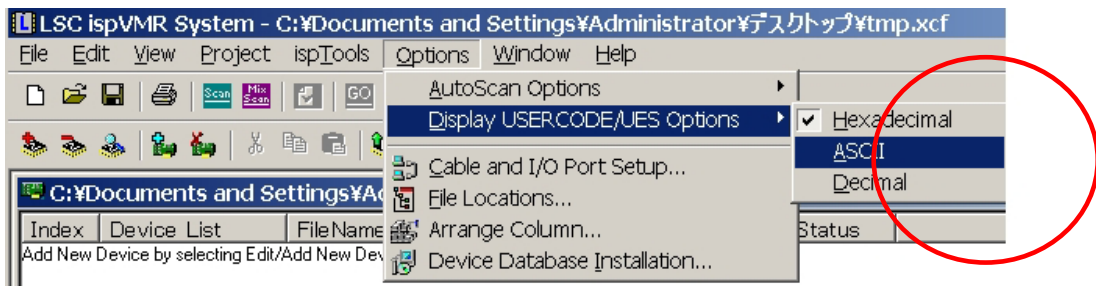


図 2-7 USERCODE/UES のフォーマット設定画面

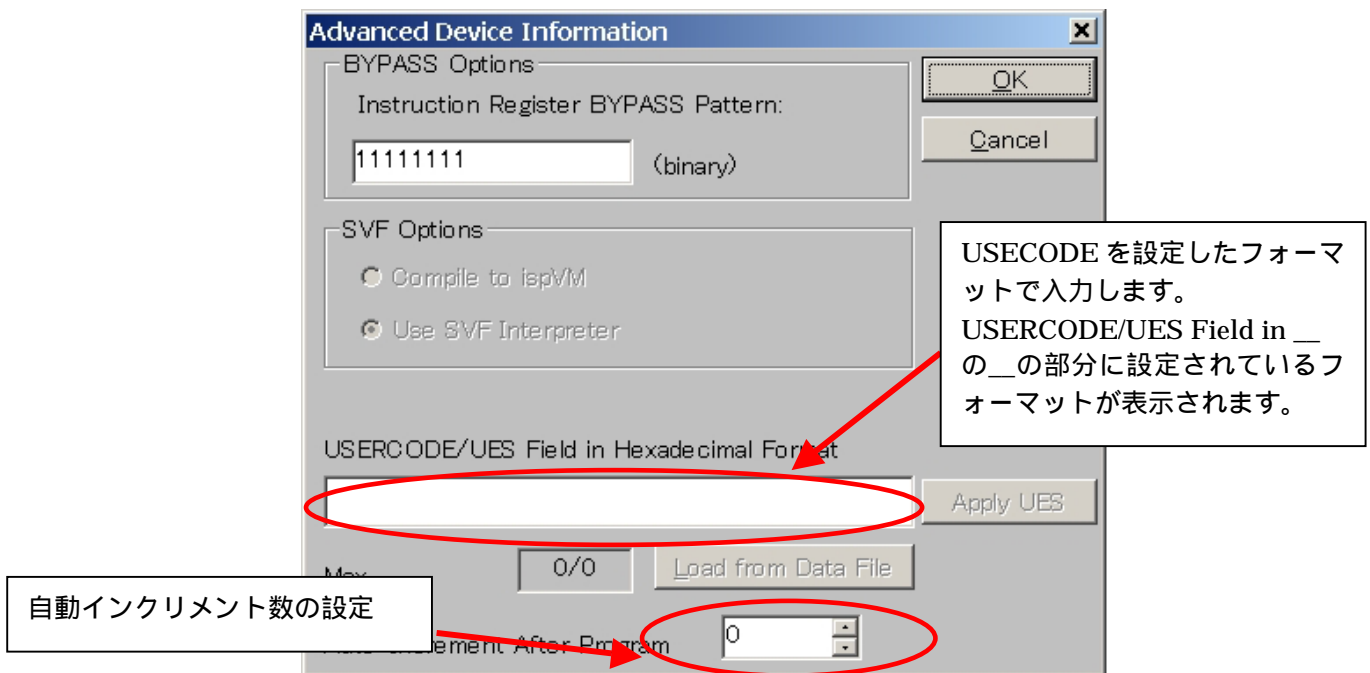


図 2-8 Advanced Device Information 画面

## 2.4 プログラミング中の I/O ステートの設定

下記手順により、プログラミング中の I/O の指定が可能です。

1. ispVM System を起動し、[Device Information]画面を開きます。
2. [Expand]をクリックします。
3. [Device Information]画面が拡張しましたら、[I/O Status]をクリックし、プルダウンから、プログラミング中の I/O 状態を設定します。

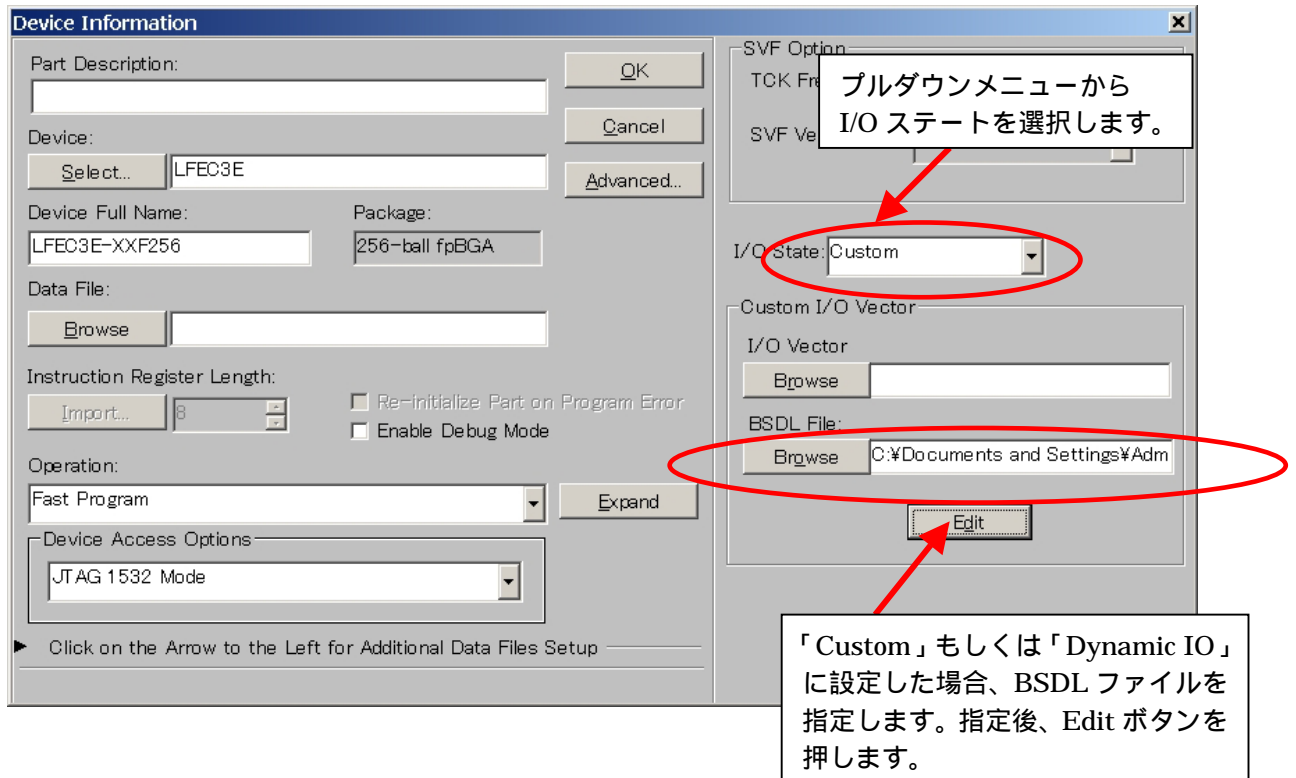


図 2-9 Device Information 画面

各設定項目に対する、I/O 状態を表 2-1に示します。

表 2-1 Device Access Options の設定内容(EC/ECP/ECP2 ファミリ)

項目	内容
High-Z(Default)	全てのピンを Hi-Z 状態にします。
All 1's	全てのピンを High にします。
All 0's	全てのピンを Low にします。
Leave Alone	ピンの状態をサンプルし、プログラム中はその状態を保持します。
Custom	プログラム前にベクタを作成し、選択したピンごとに I/O 状態を指定します(未設定ピンは High-Z になります)。詳細は、下記を参照ください。
Dynamic IO	“Leave Alone”と “Custom”を混在させたモードです。“Leave Alone”に設定したピンは、サンプル値を保持し、値を指定したピンは、その値となります。
Caputure	ダウンロード前に、Edit ウィンドからの I/O の値をキャプチャし、その後 Dynamic モードになります。

I/O ステートを、「Custom」もしくは「Dynamic IO」に設定した場合、BSDL ファイルが必要となります。BSDL は、ラティスの HP からダウンロードできますので、ダウンロードして使用ください。  
\* 初めて、ダウンロードする場合はユーザー登録(無料)が必要となります。

BSDL ファイルを指定後、EDIT ボタンを押しますと、のような[I/O Vector Editor]画面が開きます。

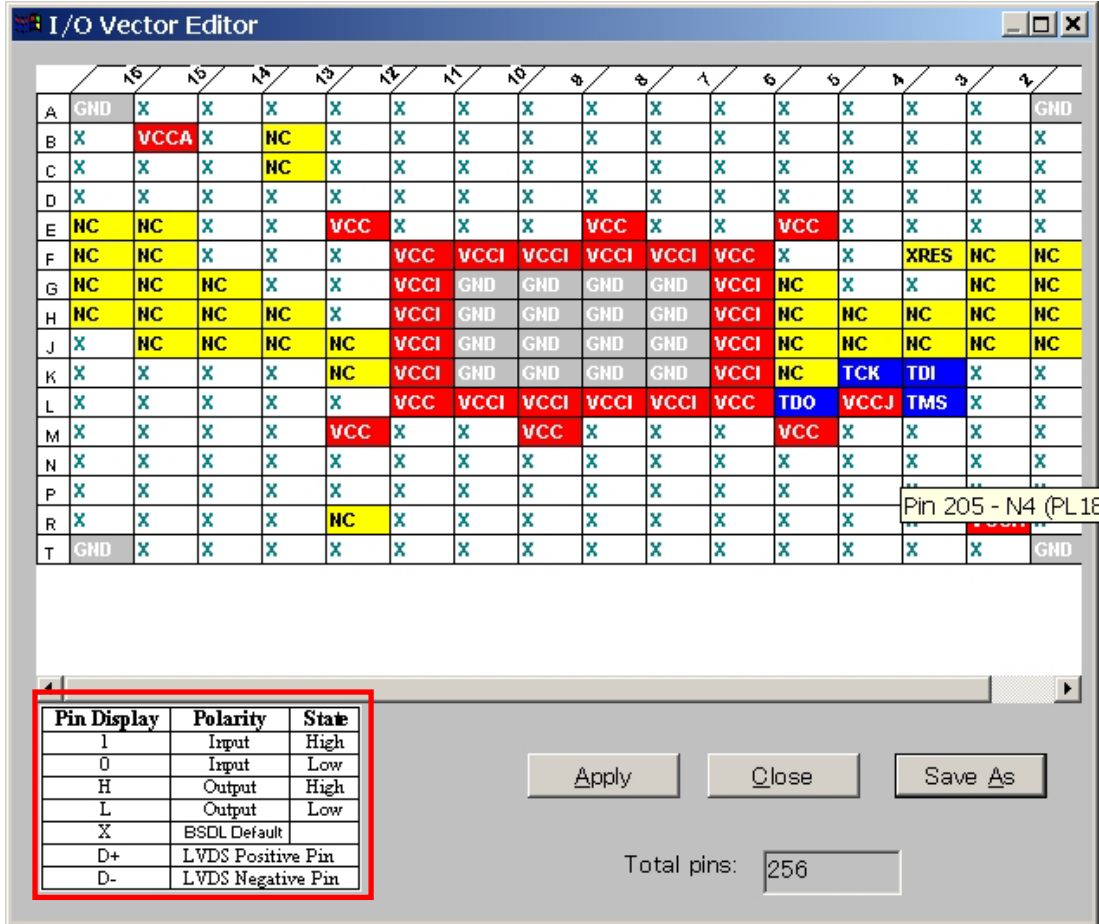


図 2-10 I/O Vector Editor 画面

[I/O Vector Editor]画面上で、各ピンに対してクリックを繰り返すことで、I/O ピンの状態を設定することができます(色のついているピンは設定不可です)。設定内容に対するピンの状態は、左下の表に示されます。

設定が完了しましたら、[Apply]ボタンを押すことで、設定が反映されます。また、[Save As]ボタンを押すことで、I/O ベクタファイル(\*.iov)として保存することができます。

## 2.5 書き込み設定の変更

ispVM System のメニュー画面の[Project] [Project Setting]により、書き込みに関する設定を変更することができます。

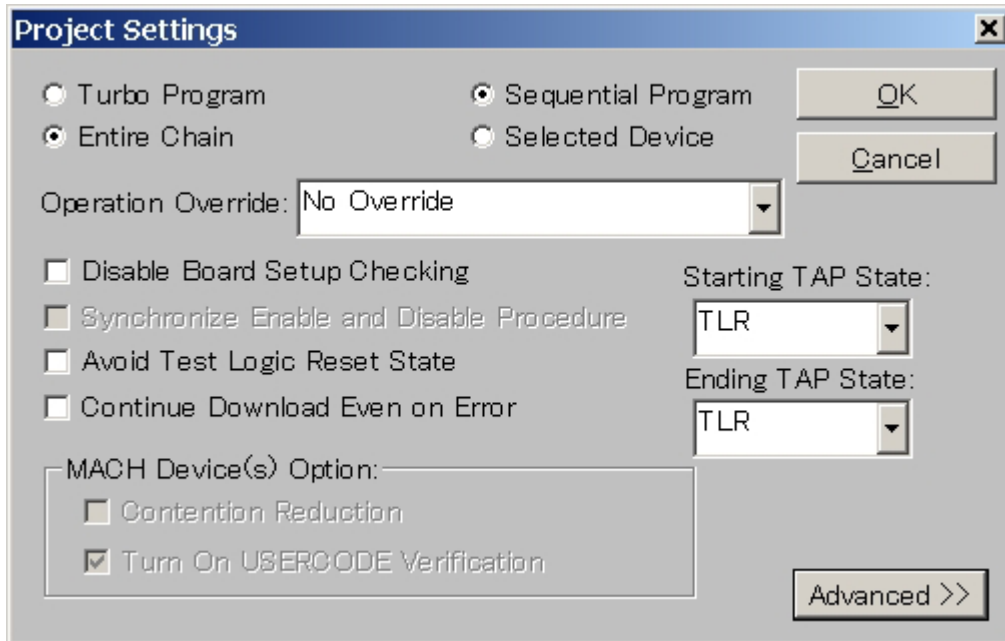


図 2-11 書き込みに関する設定画面

各設定項目に対する、設定内容を図 2-11に示します。

表 2-2 書き込みに関する設定内容

項目	内容
Sequential Program	同一チェーン上のデバイス 1 個ずつ書き込みを実施 (JTAG アルゴリズム)。
Turbo Program	デバイスに同時に書き込みます。これにより書き込み時間が短縮できます (Lattice オリジナルアルゴリズム/特許)。
Entire Chain	チェーン上全てのオペレーションを実施します。
Selected Device	チェーンの中で、選択されたデバイスに対してのみオペレーションを実施します。
Operation Override	プルダウンメニューからオペレーションを選択できます。
Disable Board Setup Checking	書き込み前ボード構成のチェックを実施しません。
Avoid Test Logic Reset State	ダウンロード中に、JTAG の Test Logic Reset 状態を回避します。
Continue Download Even on Error	ダウンロード中に、エラーが発生してもソフト上で無視します。
Starting/Ending TAP State	ダウンロード時、最初の JTAG の状態を指定します。



また、[Advance]ボタンをクリックすると以下のような画面が開きます。

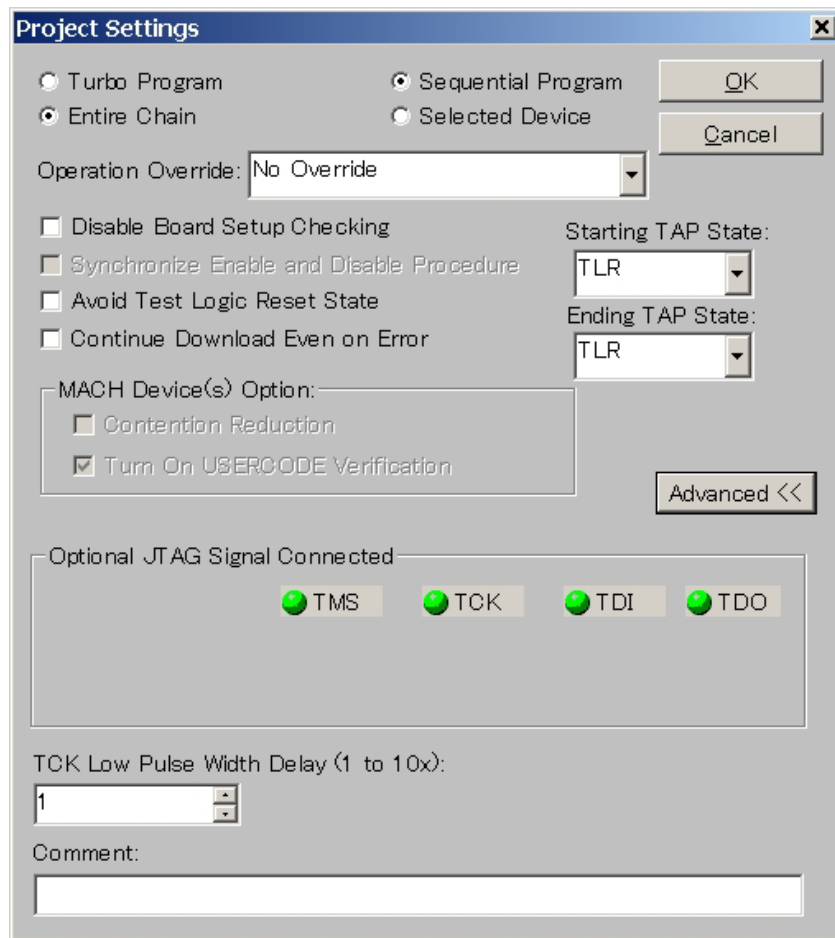


図 2-12 書き込みに関する設定画面(拡張設定)

ここでは、以下のような設定が可能です。

表 2-3 書き込みに関する設定内容(拡張設定)

項目	内容
TCK Low Pulse Width Delay	TCK のクロック幅を調整します。数字を大きくすれば、クロック幅が大きくなり、書き込み時間が増大します。
Comment	チェーンファイルにコメントを記入することができます。

## 2.6 ATE プログラミング・ベクタの生成

基板テスター(ATE : Automatic Test Equipment)用にプログラミング・ベクタを生成することができます。プログラミング・ベクタを使用すると、製造時に基板のテストとプログラミングを同時に行うことができます。

対応している ATE は次になります。

- ・ Gen Rad
- ・ HP 3070
- ・ HP 3065
- ・ Teradyne 1800
- ・ Teradyne L200/300

また、上記以外でも SVF(Serial Vector Format)に対応している ATE であれば、SVF ファイルを生成することで、ATE でのプログラミングが可能です。

SVF のスペックについては、下記 URL から参照できます。

<http://www.asset-intertech.com>

もしくは、:¥ispTOOLS¥ispvmsystem¥documents¥svf\_standard.pdf から参照可能です。

ATE プログラミング・ベクタ生成画面は、以下の何れかの方法で起動します。

- Project Generate ATE Vector を選択
- ATE アイコンをクリック

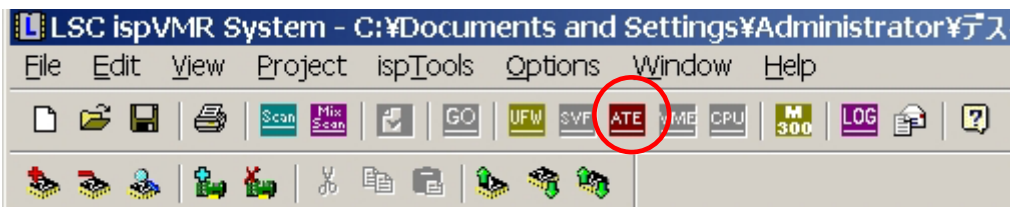


図 2-13 ATE プログラミング・ベクタ生成画面の起動

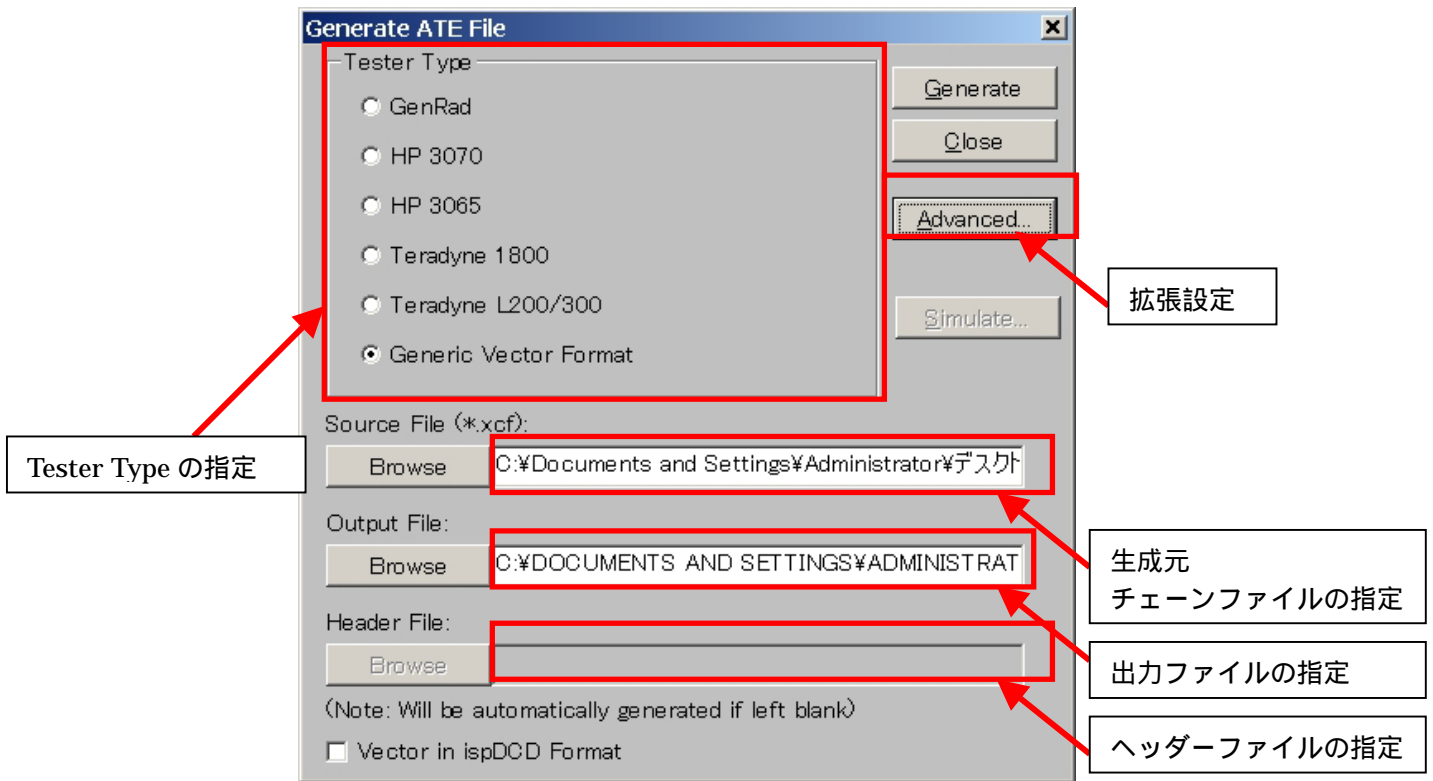


図 2-14 ATE 生成画面

図 2-14のような、ATE 生成画面が起動しましたら、まず Tester Type を指定します。続いて、生成基チェーンファイルの指定、出力ファイルの指定を行います。また、Header File には、各種ピン情報などを含む、ヘッダーファイルを指定します。ブランクとした場合は、自動的にヘッダーファイルは生成されます。

続いて、「Advanced」ボタンをクリックし、起動した画面の「More」ボタンをクリックすると、のような画面が起動します。

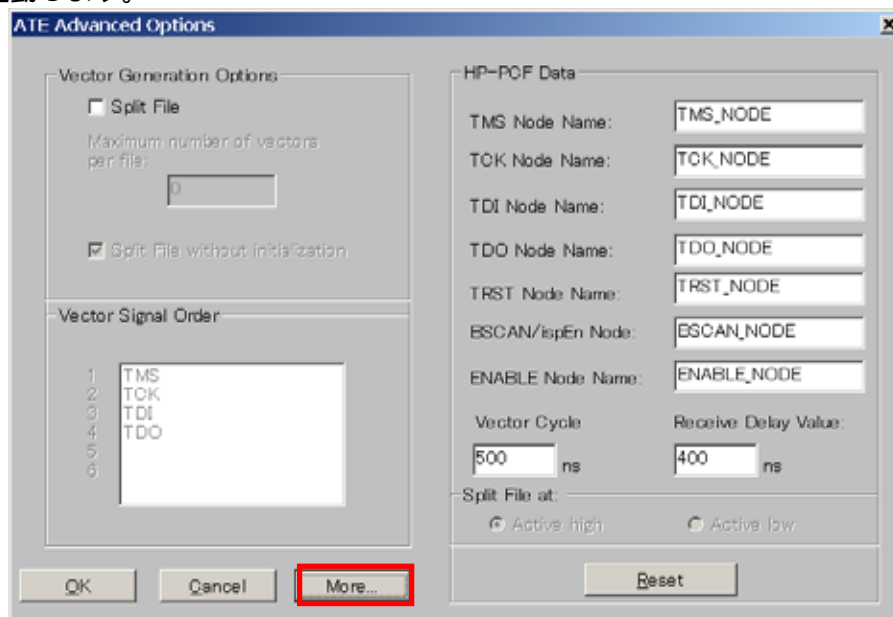


図 2-15 ATE Advanced Options 画面

ここでは、必要に応じて設定を行います。  
 各設定の内容をに示します。

表 2-4 書き込みに関する設定内容

項目	内容
Split File	ファイルサイズを削減するために、ベクタファイルを分割します。
Maximum number of vectors per file	各ベクタファイルの、許容するベクタの最大数を指定します。この値は、ATE ベンダーから得ることができます。
Split File without initialization	ファイルサイズを削減するために、ベクタファイルを分割します。このオプションを有効にするためには、ispEN,MODE および SCLK にプルダウン抵抗が必須です。
Program Vectors Only	ベリファイ動作をスキップします。
Erase Vectors Only	消去動作のみの ATE ファイルを生成します。これは、チェーンファイル上の個々のデバイスに対する設定を無視します。
Vector Signal Order	各ベクタの信号オーダを指定します。
HP-PFC Data	More ボタンを押した時のみ、この設定は可能です。プログラミングピンの名前、リードおよびライトのディレイ値を設定できます。

設定後、「OK」ボタンをクリックして ATE Advanced Options 画面を閉じ、ATE 生成画面の「Generate」ボタンをクリックすると、指定したフォルダに ATE プログラミング・ベクタファイルが生成されます。

## 2.7 SVF ファイルの生成

SVF ファイル生成画面は、以下の何れかの方法で起動します。

- Project Generate SVF File を選択
- SVF アイコンをクリック

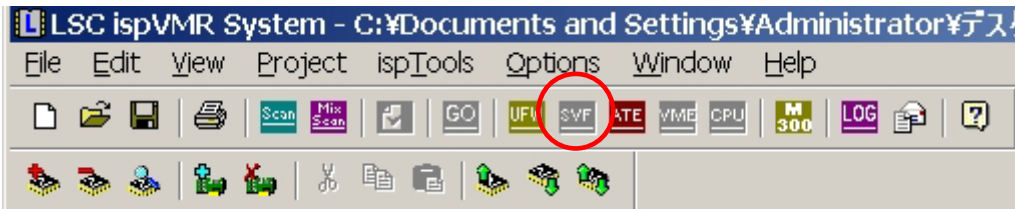


図 2-16 SVF ファイル生成画面の起動

### 2.7.1 デバイス単体の SVF ファイルの生成

デバイス単体の SVF ファイルを生成する場合は、図 2-17 のような SVF ファイル生成画面が起動しましたら、Options で「Build SVF File For Single Device」を選択します。続いて、出力ファイルの指定、生成元ファイルの設定を行います。チェーン上に複数のデバイスがある場合は、何番目のデバイスに対して生成を行うのかを入力してから、同様の設定を行います。設定が完了しましたら、「Generate」ボタンをクリックすることで、指定したフォルダに SVF ファイルが生成されます。

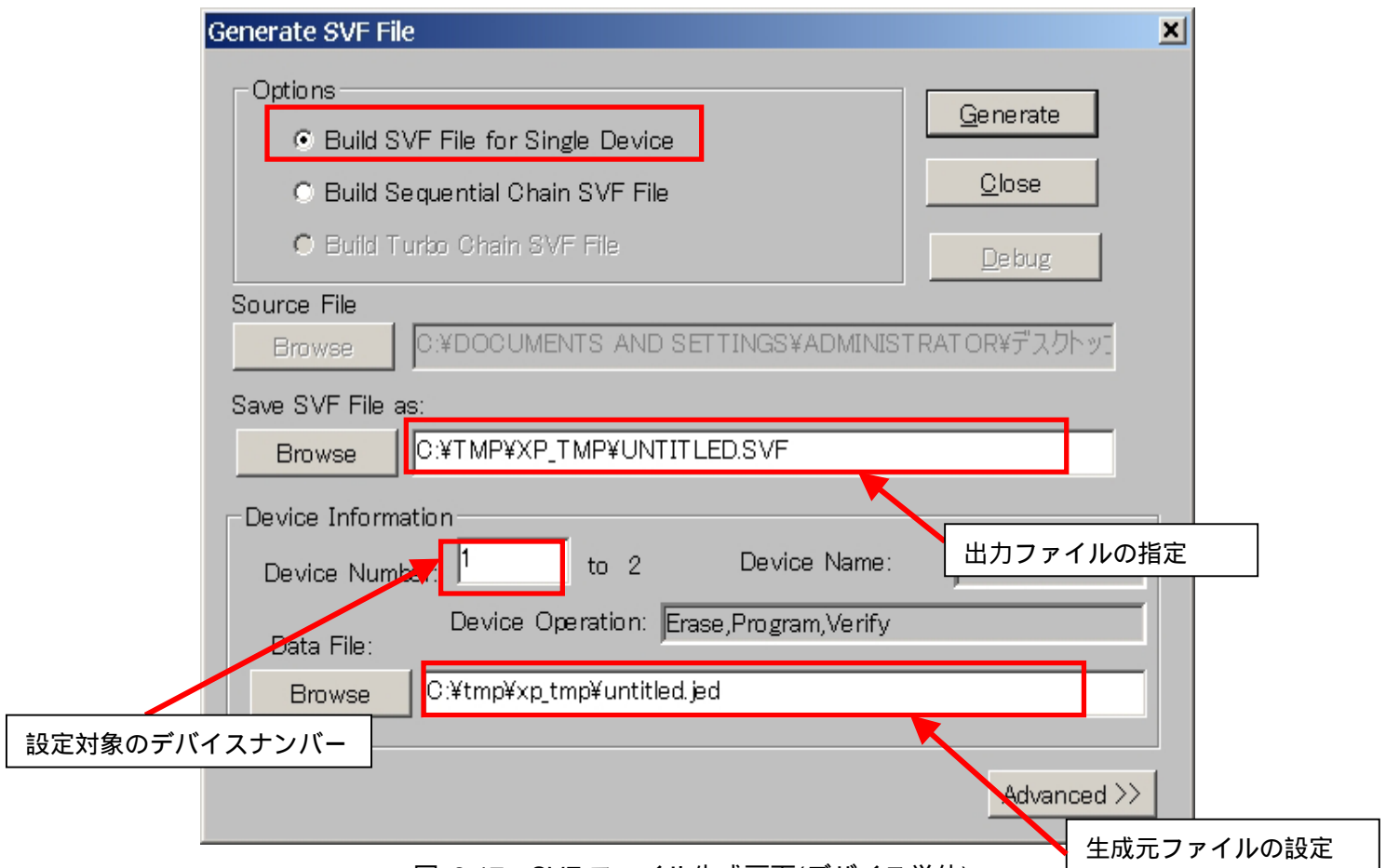


図 2-17 SVF ファイル生成画面(デバイス単体)

### 2.7.2 チェーン全体の SVF ファイルの生成

チェーン全体の SVF ファイルを生成する場合は、図 2-18のような SVF ファイル生成画面が起動しましたら、Options で「Build Sequential Chain SVF File」を選択します。続いて、出力ファイルの指定、生成元ファイルの設定を行います。設定が完了しましたら、「Generate」ボタンをクリックすることで、指定したフォルダに SVF ファイルが生成されます。

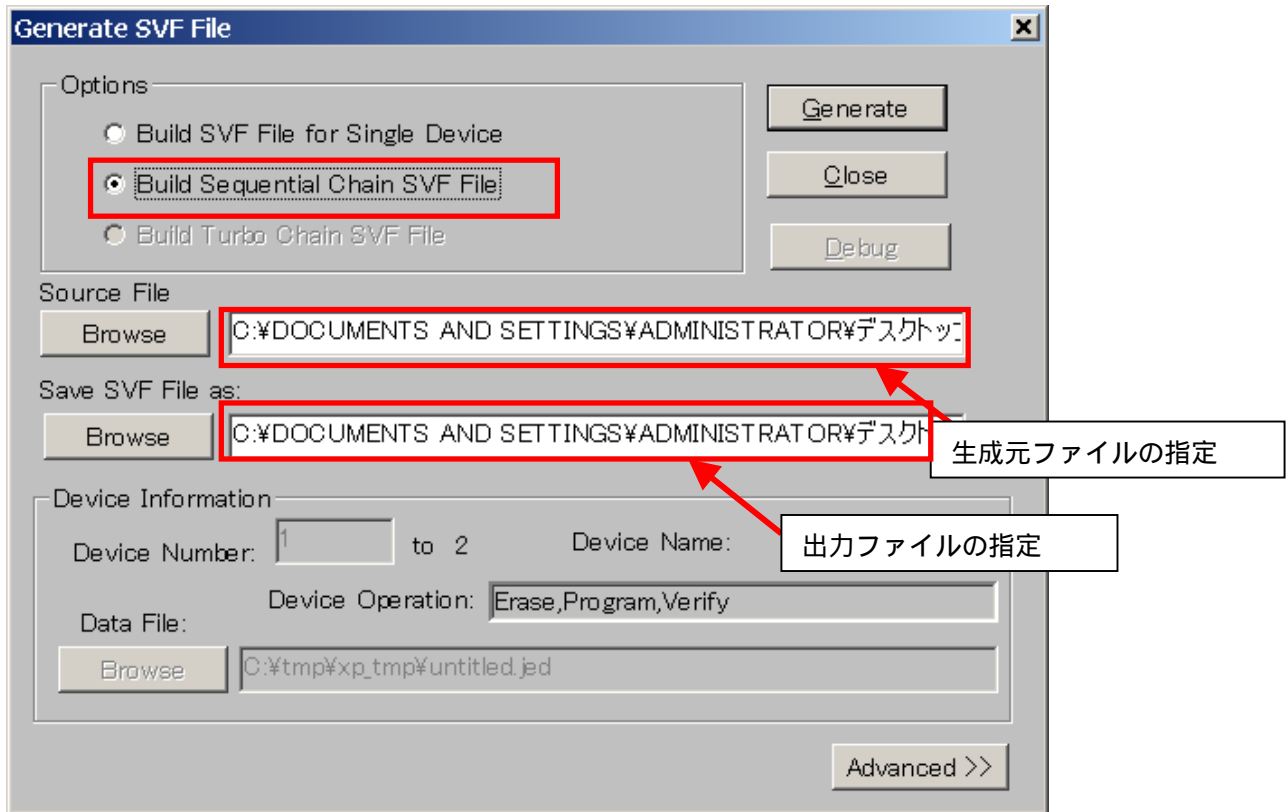


図 2-18 SVF ファイル生成画面(チェーン全体)

## 2.8 VME ファイルの生成

### 2.8.1 VME ファイルの生成

VME ファイルは、エンベデッドプログラム用に生成されるバイナリファイルになります。VME ファイルはチェーンおよびデータ情報をもったファイルになります。

VME ファイル生成画面は、以下の何れかの方法で起動します。

- Project Generate SVF File を選択
- VME アイコンをクリック

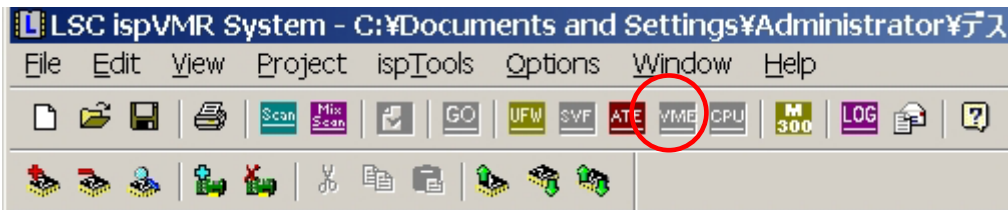


図 2-19 VME ファイル生成画面の起動

上記何れかの方法で、VME ファイル生成画面を起動しますと、以下のような画面が開きます。ここで、タブの VME12、VME11 等の VME\*\*の\*\*部はバージョンを示します。VME ファイルの生成には、最新のバージョンをご使用ください。

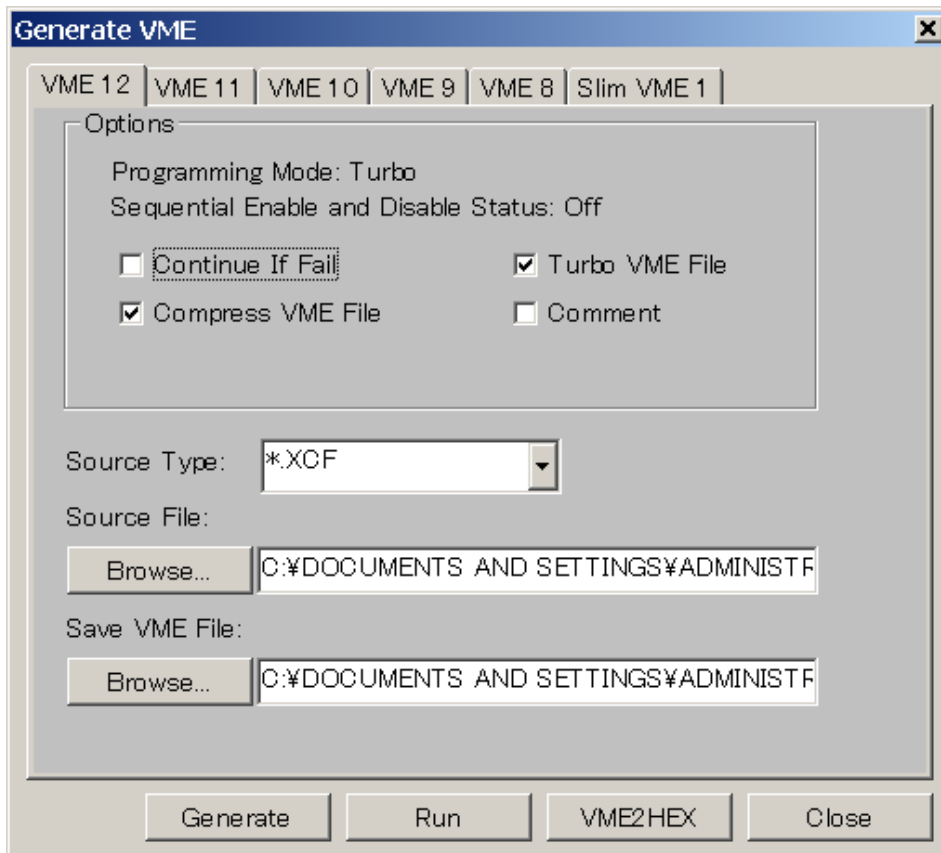


図 2-20 VME ファイル生成画面

生成元ファイル、出力ファイルを設定しましたら、オプションの設定を行います。オプションの設定内容を表 2-5に示します。

表 2-5 Programing Mode 設定内容

項目	内容
Continue If Fail	プログラムまたはベリファイ前に、デバイスのユーザーコードとVMEファイルのユーザーコードが一致するかどうかをチェックします。
Turbo VME File	Program Setting..で設定したモードを無視して、Turboモード設定となります。
Compress VME File	VME ファイルを圧縮します。
Loop VME File	ループによって、生成される VME ファイルサイズを削減します。
Commnet	コメント付の VME ファイルを生成します。

オプション設定完了後、「Generate」ボタンをクリックしますと、指定したフォルダに VME ファイルが生成されます。



## 2.8.2 エンベデッド用 C ソースファイルの生成

図 2-20 の VME ファイル生成画面にて、「VME2HEX」ボタンを押すと、図 2-21 のような VME ファイルを HEX ファイルに変換するための画面が起動します。

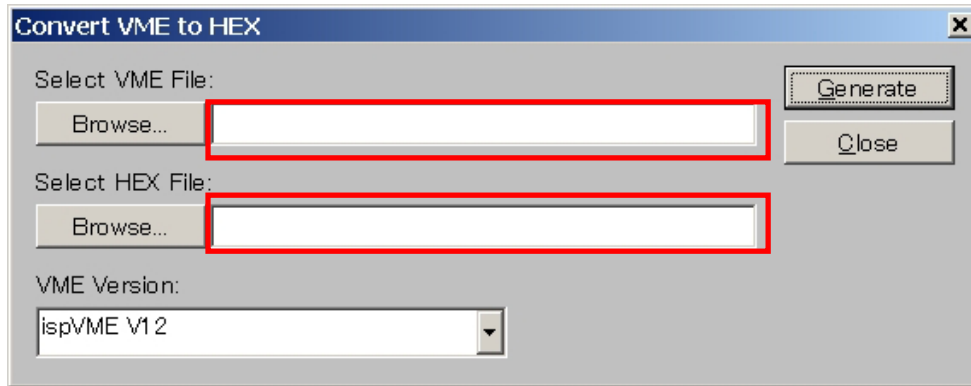


図 2-21 VME ファイルから HEX ファイルへの変換

ここで、変換元ファイルと、出力先ファイルを指定し、「Generate」ボタンをクリックしますと、図 2-22 のようなコマンドプロンプト画面が起動します。「続行するには何かキーを押して下さい」とメッセージがでますので、キーを何か押して下さい。キーを押しますと、C 言語ファイル(拡張子 c)とヘッダファイル(拡張子 h)が生成されます。C 言語ファイルは、デバイスの容量に応じていくつか分割して生成されます。

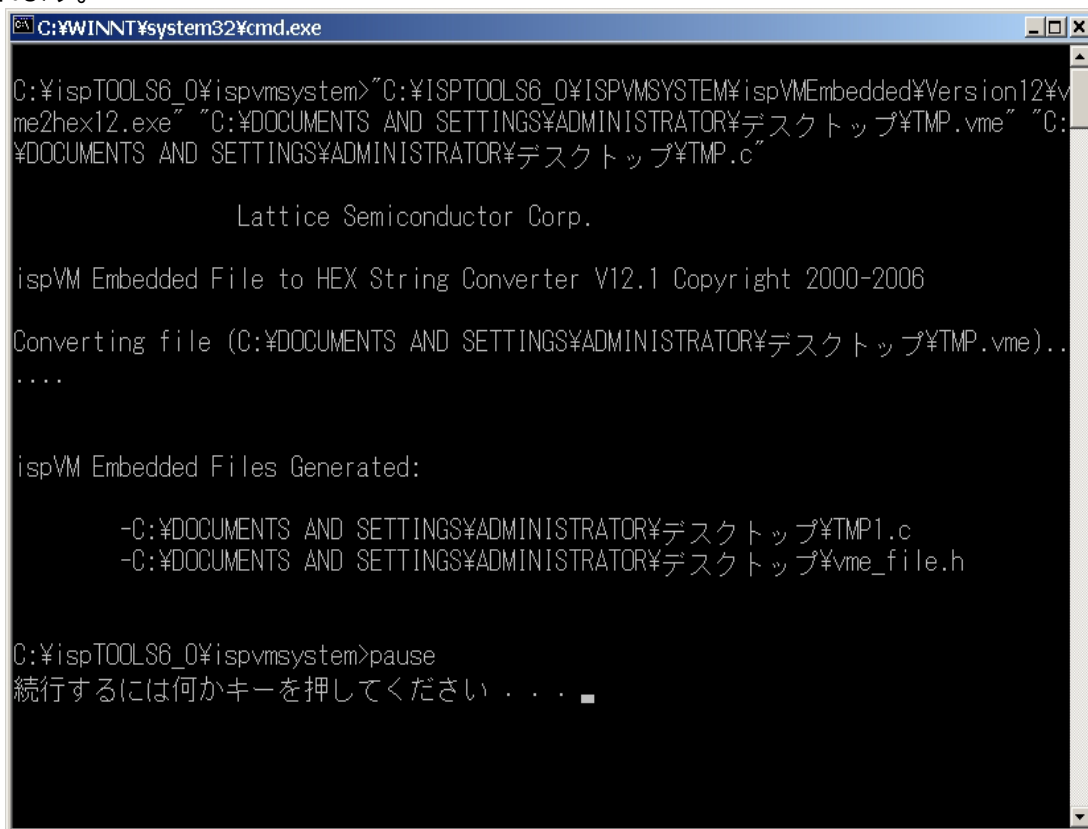


図 2-22 C 言語ファイル生成中画面

ispVM にインストールされている C 言語の公開ソースをもとに、各プラットフォーム用にカスタマイズを行い、CPU からのエンベデッドコマンドを完成させます。

公開ソースは下記フォルダに保存されています。

インストールフォルダ¥ispvmsystem¥ispVMEmbedded¥SoureCode

詳細説明は、readme.txt を参照願います。

## 2.9 CPU エンベデッド ビットストリームの生成

複数または単体の FPGA チェーン用に CPU エンベデッド ビットストリームを生成します。CPU エンベデッド ビットストリーム生成画面は、以下の何れかの方法で起動します。

Project Generate SVF File を選択  
CPU アイコンをクリック

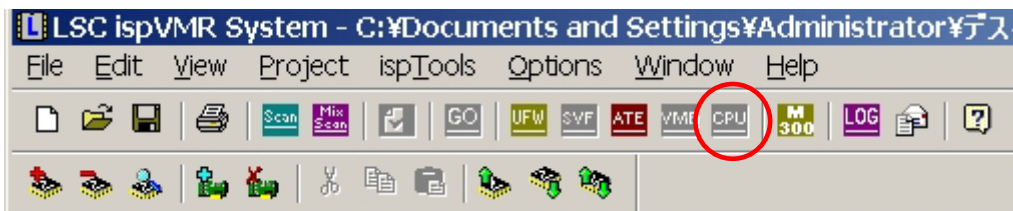


図 2-23 CPU エンベデッド ビットストリーム生成画面の起動

上記何れかの方法で、CPU エンベデッド・ビットストリーム生成画面を起動しますと、以下のような画面が開きます。

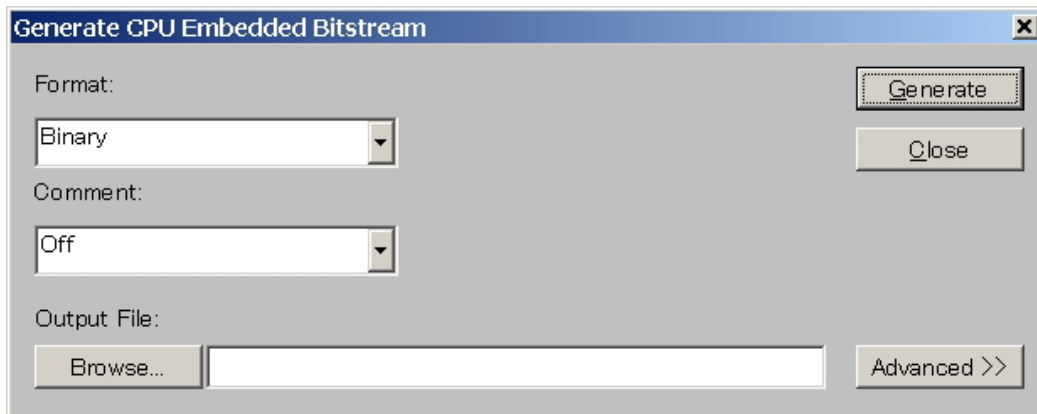


図 2-24 CPU エンベデッド ビットストリーム生成画面

続いて、フォーマット等の設定を行います。  
設定項目と内容について、表 2-6に示します。

表 2-6 オプション設定内容

項目	内容
Format	出力ファイルのフォーマットを設定します。 Binary : バイナリフォーマット C-CODE : Cコードのファイル Intel-HEX : インテル HEX フォーマット Text : テキストフォーマット(デバック用)
Comment	コメントを入れるか否かを設定します。 コメントを入れる場合は、ON に設定します。
Output File	出力ファイルを設定します。Browse をクリックし、出力フォルダ、ファイル名を設定します。
Advanced	
Configuration Mode	各 CFG ピンのコンフィグレーションを有効にします。
Byte Wide Bit Mirror	インテル HEX ファイルの MSB と LSB を反転します。

オプション設定完了後、「Generate」ボタンをクリックしますと、指定したフォルダに CPU エンベデッドビットストリームが生成されます。

## 2.10 コマンドラインモード

ispVM System は、コマンドラインで起動することが可能です。  
製造段階で、コマンドをバッチファイル化し、マウスオペレーションを排除することで、書き込みに要する時間を短縮することが可能となります。

### ・コマンドのシンタックス

コマンドの基本文型は以下のようになります。

ドライブ:¥ispVM ディレクトリ¥ispVM.exe -infile"<入力ファイルパス>" /-processtype<プロセスタイプ>/-cabletype<ケーブルタイプ>/-portaddress<ポートアドレス>/-processmode<プロセスモード>

表 2-7 コマンドラインの設定内容

項目	内容
入力ファイルパス(必須)	拡張子が xcf もしくは dld のチェーンファイル名を指定します。パスにスペースが含まれる場合には" "で囲ってください。
プロセスタイプ(オプション)	ターボモード(turbo)もしくはシーケンシャルモード(sequential)を指定します。何も指定しない場合は、シーケンシャルモードになります。
ケーブルタイプ(オプション)	ラティス製パラレルケーブル(lattice)、バンティス製ケーブル(vantis)、ラティス製 USB ケーブル(usb)を指定します。何も指定しない場合は、ラティス製パラレルケーブルになります。
ポートアドレス(オプション) パラレルケーブルの場合	LPT1(0x0378)、LPT2(0x0278)、LPT3(0x03BC)、カスタム(0xXXXX)を指定します。何も指定しない場合は、LPT1(0x0378)になります。
ポートアドレス(オプション) USB ケーブルの場合	0~15 の間でポートアドレスを指定します。15 の場合 Ezusb-15 と指定します。何も指定しない場合は、Ezusb-0 となります。
プロセスモード(オプション)	サイレントモード(-h)、メッセージモード(-o)、ウインドウモード(-w)を指定します。何も指定しない場合、サイレントモードとなります。

以下に、コマンド例を示します。

```
C:¥ispTOOLS¥ispvmsystem¥ispVM.exe -infile "C:¥demo¥demo.xcf" -processtype turbo -cabletype lattice -portaddress 0x0278
```

## 2.11 Universal File Writer

Universal File Writer(以下 UFW)は、データフォーマットの変換や、ファイルの圧縮、分割等をするためのツールです。

UFW は、以下の何れかの方法で起動します。

- ispTOOLS Universal File Writer を選択
- UFW アイコンをクリック

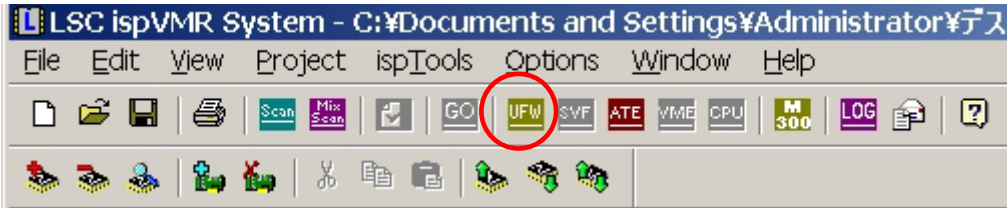


図 2-25 VME ファイル生成画面の起動

上記何れかの方法で、UFW を起動しますと、以下のような画面が開きます。

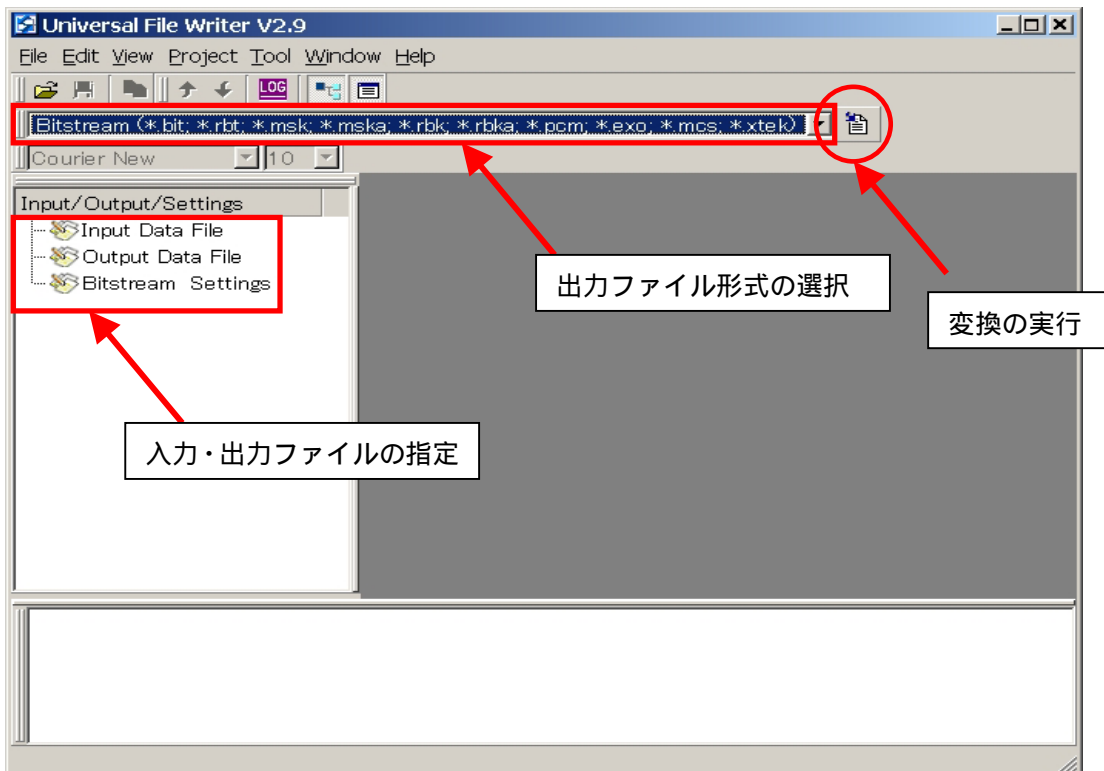


図 2-26 Universal File Writer(UFW)起動時の画面

UFW の画面が開きましたら、以下の手順でファイルのフォーマット変換を実施します。

- 出力ファイルの形式を選択します。
- 入力ファイルおよび出力ファイルを指定します。
- アイコンをクリックし、変換の実行

## 2.12 データの圧縮

EC/ECP/ECP2 ファミリでは、UFW を用いてコンフィグレーション・ビットストリームの圧縮を行うことができます。

データ圧縮の手順は以下のとおりです。

UFW を起動する。

出力ファイル形式を Bitstream にする。

入力ファイル・出力ファイルを設定する。

Bitstream Settings で[Compression]を ON に設定する。

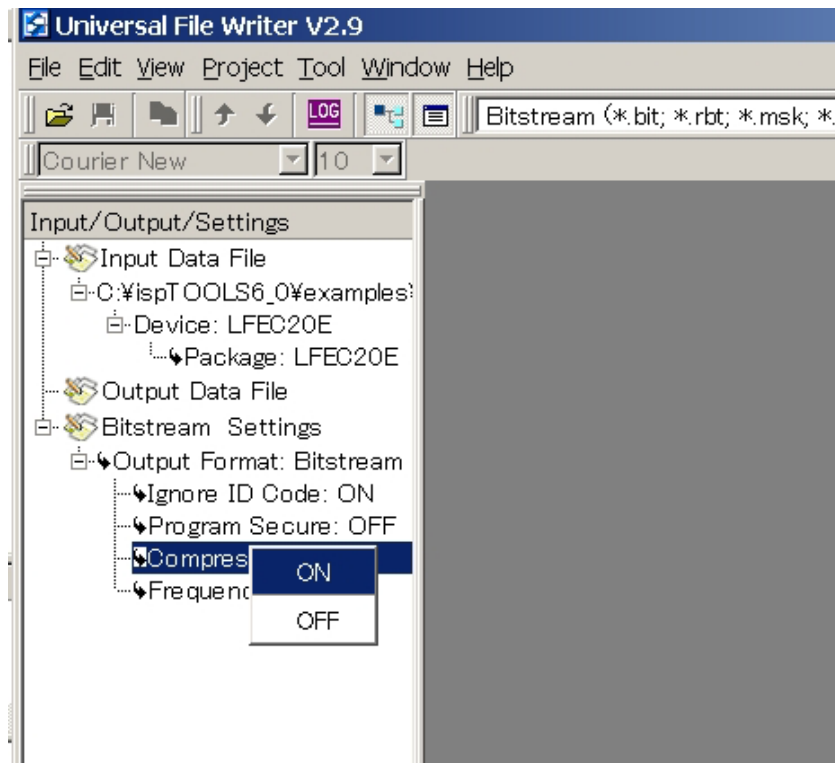


図 2-27 ビットストリームデータ圧縮オプションの設定

上記手順により、出力されるビットストリームファイルは圧縮されます。

## 2.13 Application Specific BSDL File Generator

Application Specific BSDL File Generator は、ユーザーデザインの信号名を使用して、アプリケーション特定の BSDL(Boundary Scan Description Language)ファイル生成するためのツールです。

VREF または LVDS ペアをサポートし、配置されている I/O に対して、変換された BSDL ファイルは VREF または LVDS ペアの振る舞いを正確に反映しています。このファイル生成するには、汎用の BSDL と、JEDEC または ISC ファイルが入力データとして必要となります。汎用の BSDL ファイルは、ラティスの WEB サイトから入手可能です。

Application Specific BSDL File Generator は ispVM Systems のメニュー画面から ispTOOLS Application Specific BSDL File Generator を選択することで、起動できます。

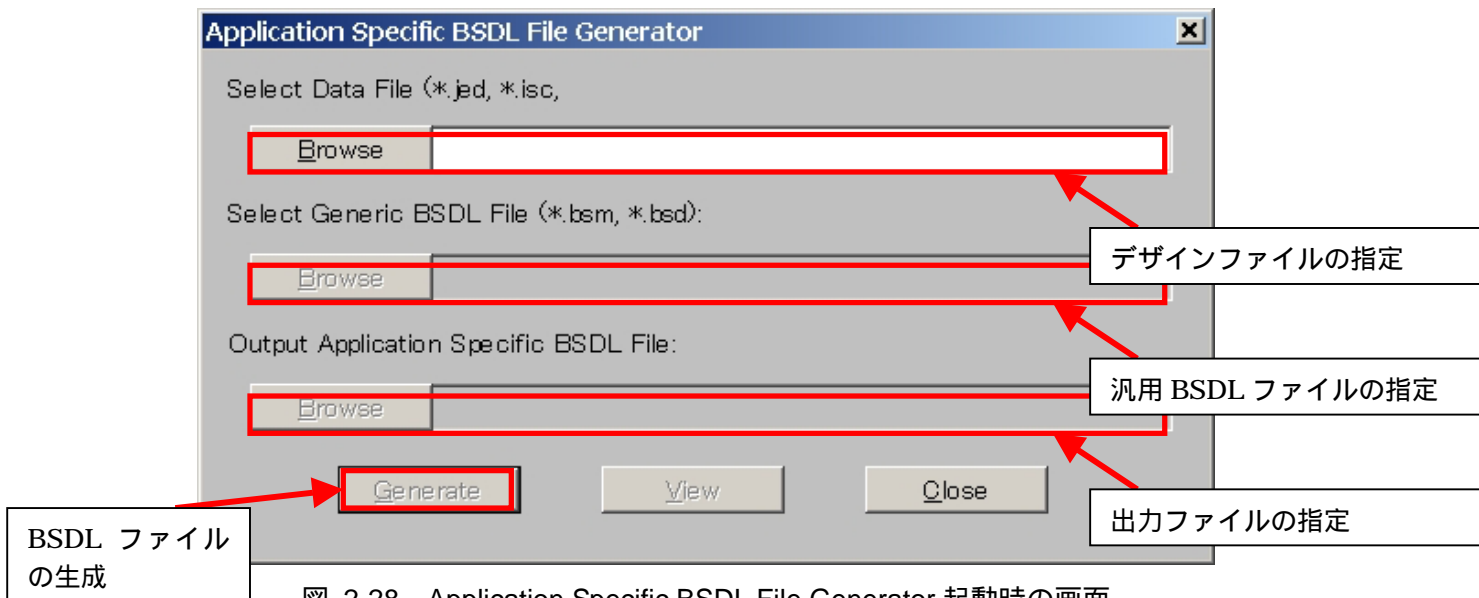


図 2-28 Application Specific BSDL File Generator 起動時の画面

起動しましたら、下記手順により BSDL ファイルの生成を行います。

- 特定のデザインファイルを指定
- 汎用 BSDL ファイルを指定
- 出力ファイルを指定
- Generate ボタンをクリック

## 2.14 Model300 の使用方法

Model300 は、Lattice 社でリリースしている簡易的な ISP/JTAG プログラマです。Model300 を使ってプログラミングを行うには、本体の他にデバイス毎、パッケージ毎に専用のソケットが必要になります。

Model300 では、Vcc のレベルが、1.8V、2.5V、3.3V および 5.0V のデバイスをサポートしており、ispVM ソフトウェアで書き込みを行うことができます。

Model300 には、本体、AC アダプタおよびダウンロードケーブルが付属しています。デバイス毎のソケットは別途購入の必要があります。ソケットの型式はラティス社の Web ページから確認することができます。



図 2-29 Model300 外観図



Model300 は以下の手順にて使用します。

Model300 の Programmer に電源を入れ、ダウンロードケーブルを接続します。  
ispVM System を起動し、メニューから ispTools Model300 Programmer を選択します(または M300 アイコンをクリックします)。

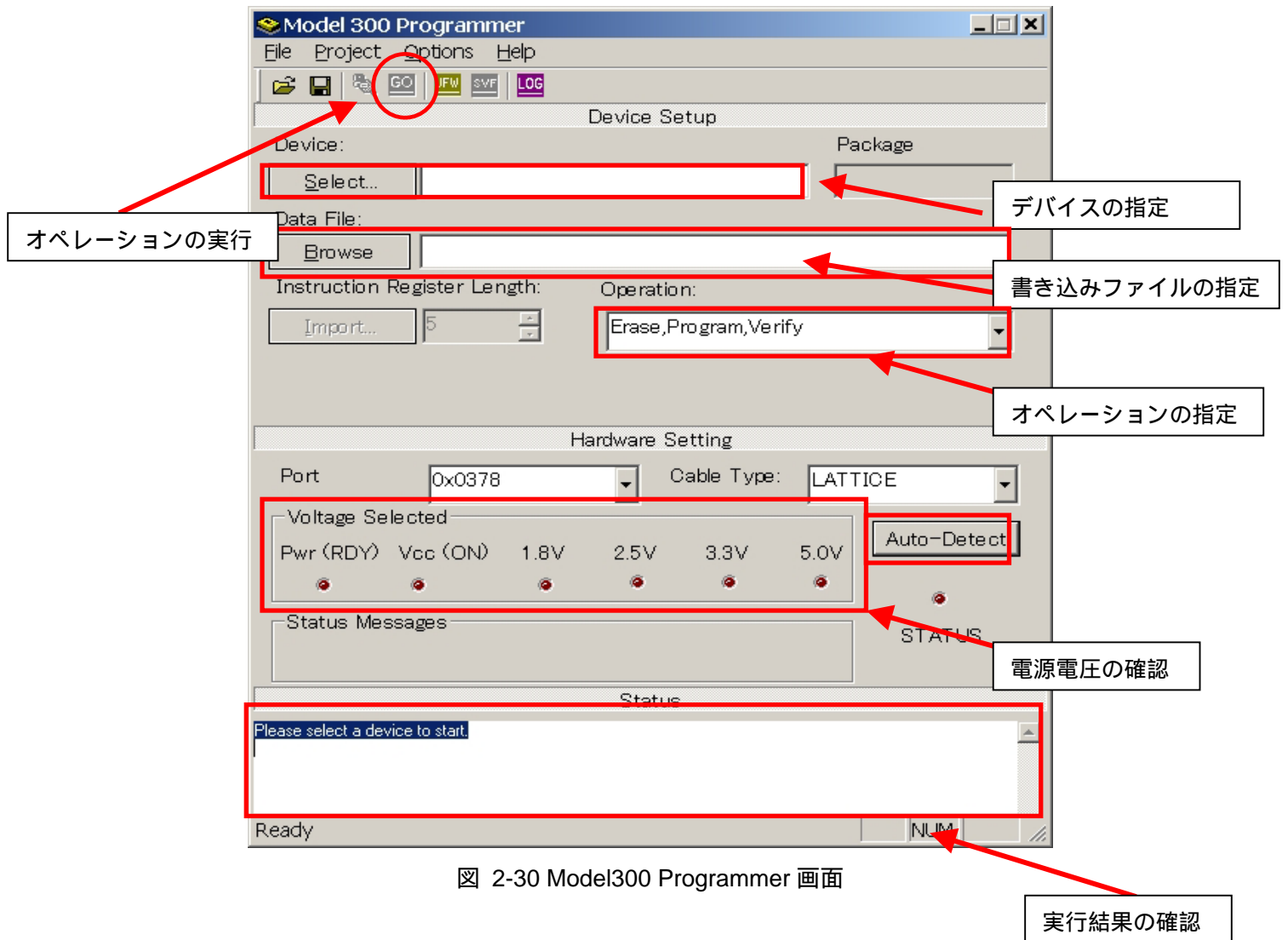


図 2-30 Model300 Programmer 画面

Model300 Programmer 画面から、デバイスを選択します。  
書き込みファイルを選択します。  
オペレーションを指定します。  
Auto-Detect を実行し、ケーブルを検出します(個別に指定することも可能です)。  
デバイスに対応する電源電圧の LED が点灯していることを確認します(ソフト上および Model300 上)。  
Go ボタンをクリックし、オペレーションを実行します。  
実施結果は、Status 欄に表示されます。

以上で ispVM System(応用編)は終了です。

### 3 トラブルシューティング(プログラミング関連)

本節では、プログラミングに関するトラブルシューティング法について説明いたします。

#### 3.1 ボード診断機能

ボード上での JTAG 信号のオープン/ショートの可能性のある個所の暗示、デバイスの IDCODE のチェックなどを行うことができます。

問題が発生した場合、JTAG ケーブルを接続し、ボードへ電源を供給した状態で ispVM System のメニューより ispTools Board Diagnostics を選択することで、ボード診断機能を実行することができます。図 3-1の図のように問題が無い場合は、デバイス型名、ID 等が正常に表示されますが、問題ある場合は、期待値と違う値が検出されます。

また、JTAG 信号のオープン/ショートの可能性のある個所は ? が表示されます。

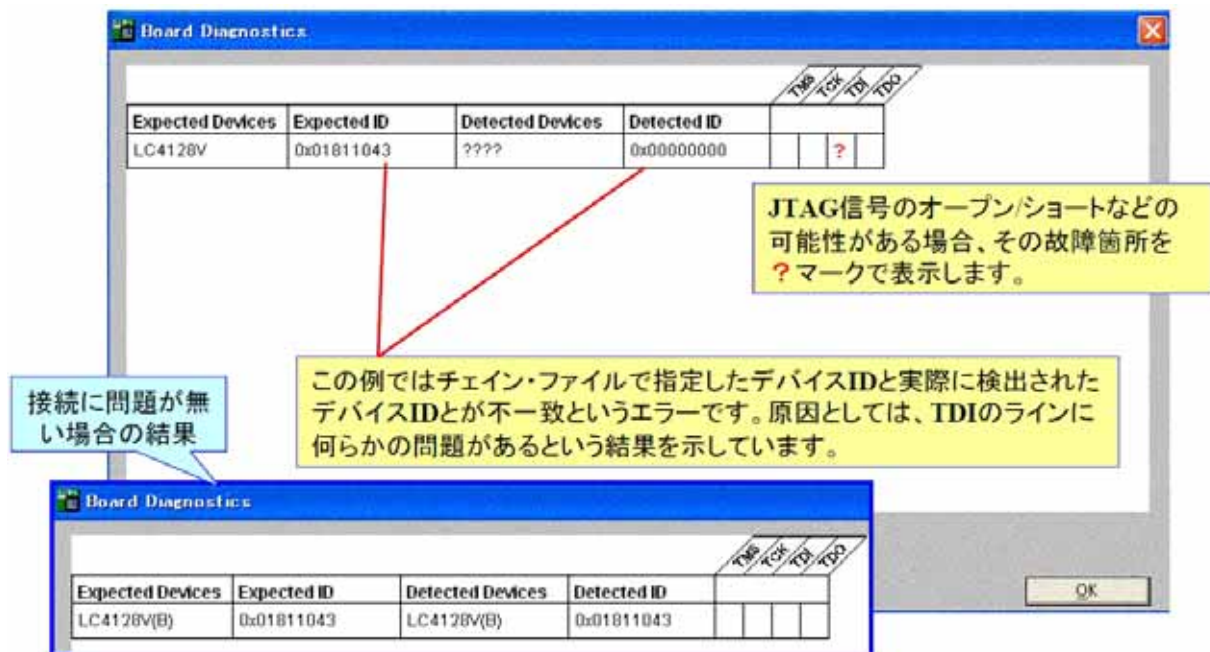


図 3-1 ボード診断機能

IDCODE に異常が発生している場合、以下のような事が考えられます。

検出した IDCODE がオール 1 もしくはオール 0 の場合、JTAG 信号の接続性に問題がある場合があります。また、非ラティスデバイスが混在している場合は、そのインストラクション・レジスタ長をチェックする必要があります。

検出した IDCODE が 1 と 0 の組み合わせで、かつ再現性が悪い場合は、TCK と TMS 信号の終端やバッファリングに問題がある可能性があります。

### 3.2 反復ダウンロード

反復ダウンロード機能を用いることで、ケーブルの接続性のチェックを行うことができます。

反復ダウンロード機能は ispVM System のメニューから ispTools Repetitive Download を選択することで起動します。



図 3-2 反復ダウンロード機能

上記のような画面が起動しましたら、プログラミングを繰り返す回数(Number of)と反復ダウンロード・プロセスを終了するまでのエラー回数(Stop on Error Number)を設定し、OK ボタンをクリックすることで、反復ダウンロードを開始します。

### 3.3 ケーブルデバッグ機能

ケーブルデバッグ機能は、テスト信号を出力しケーブルの接続性をチェックする機能です。

ケーブルデバッグ機能は、ispVM System のメニューから Options Cable and I/O Port Setup を選択して、Cable and I/O Setting 画面を起動し、Debug Mode ボタンをクリックすることで起動します。

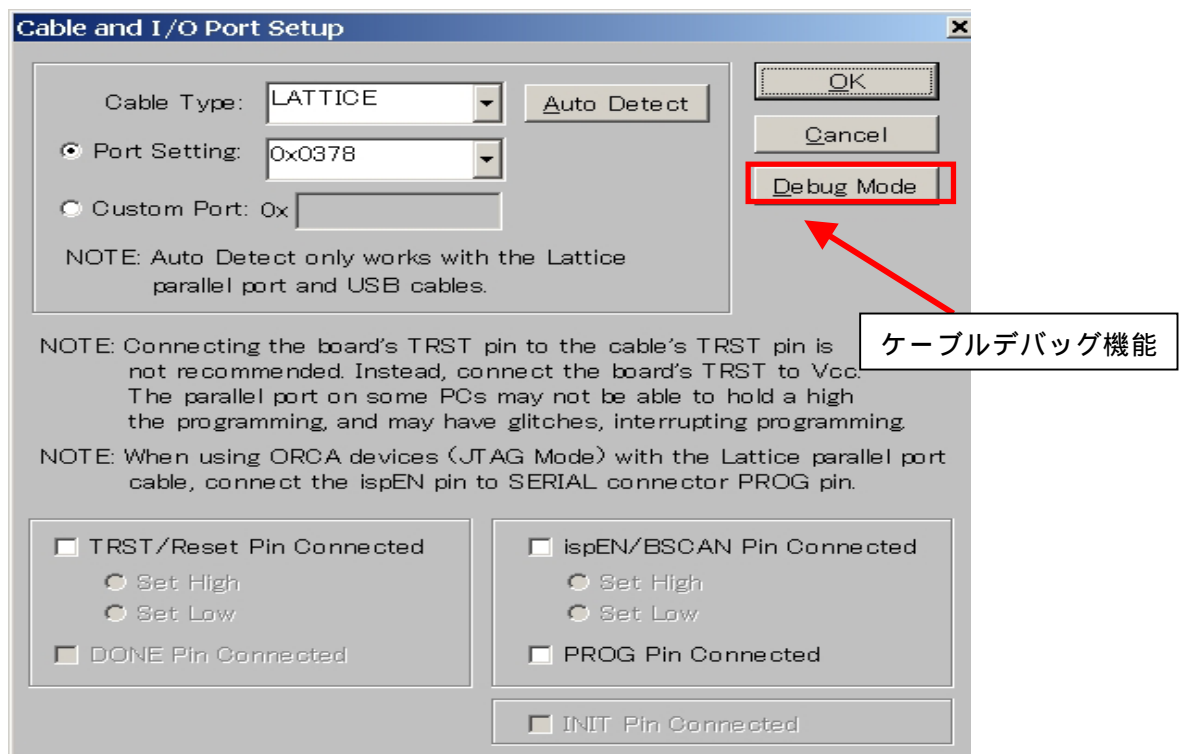


図 3-3 ケーブルデバッグ機能の起動

上記手順にて、ケーブルデバッグ機能を起動するとこのような画面が起動します。

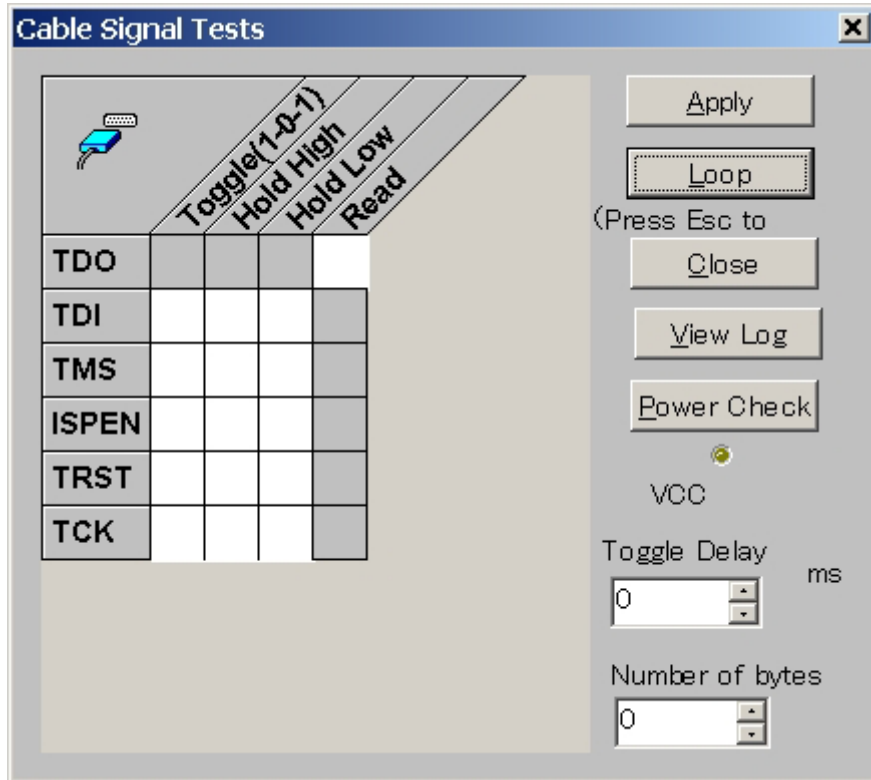


図 3-4 ケーブルデバッグ機能起動時の画面

上記画面にて、出力する信号、出力パターンにチェックを入れ、Loop で繰り返し出力し、ESC で終了となります。Power Check を実行するとケーブルに、電源が供給されているかを確認することができます。

各設定内容の詳細は、以下のとおりです。

#### Test Setting

Toggle : 1-0-1 を交互に出力

Hold High : 1 を出力

Hold Low : 0 を出力

Read(TDO Only) TDO からデータをリードバックします。

#### Apply

選択したピンに対して、設定内容を有効にします。特にトグル設定を行っているピンに対しては、トグル出力を開始します。

#### Loop

連続テストを実行します。

#### View Log

Cable Signal Test operations の結果を要約したテキストファイルを開きます。

#### Power Check

ケーブルとパワーコネクタのチェックを行います。ケーブルに正しく電源が供給されている場合は、緑色に点滅します。

### 3.4 SVF デバッガ

SVF デバッガは、ダウンロードケーブルに繋がれた実際のデバイスを動作させながら、SVF ファイルをデバッグする為のものです。

何らかの理由で、ISP/JTAG プログラミングが失敗する場合、いったん JEDEC もしくはチェーンファイルを SVF ファイルに変換し、それをデバッグすることで原因を究明できる場合があります。

SVF デバッガは、主な SVF シンタックスをサポートしていますが、あくまで CPLD プログラミング用の SVF のデバッグを意図しており、16 ビット・メモリ・サイズのみをサポートとなります。

SVF デバッガを使用するには、まず SVF インタープリタを起動します。

SVF インタープリタは、ispVM System のメニューから ispTools SVF Interpreter を選択することで起動できます。

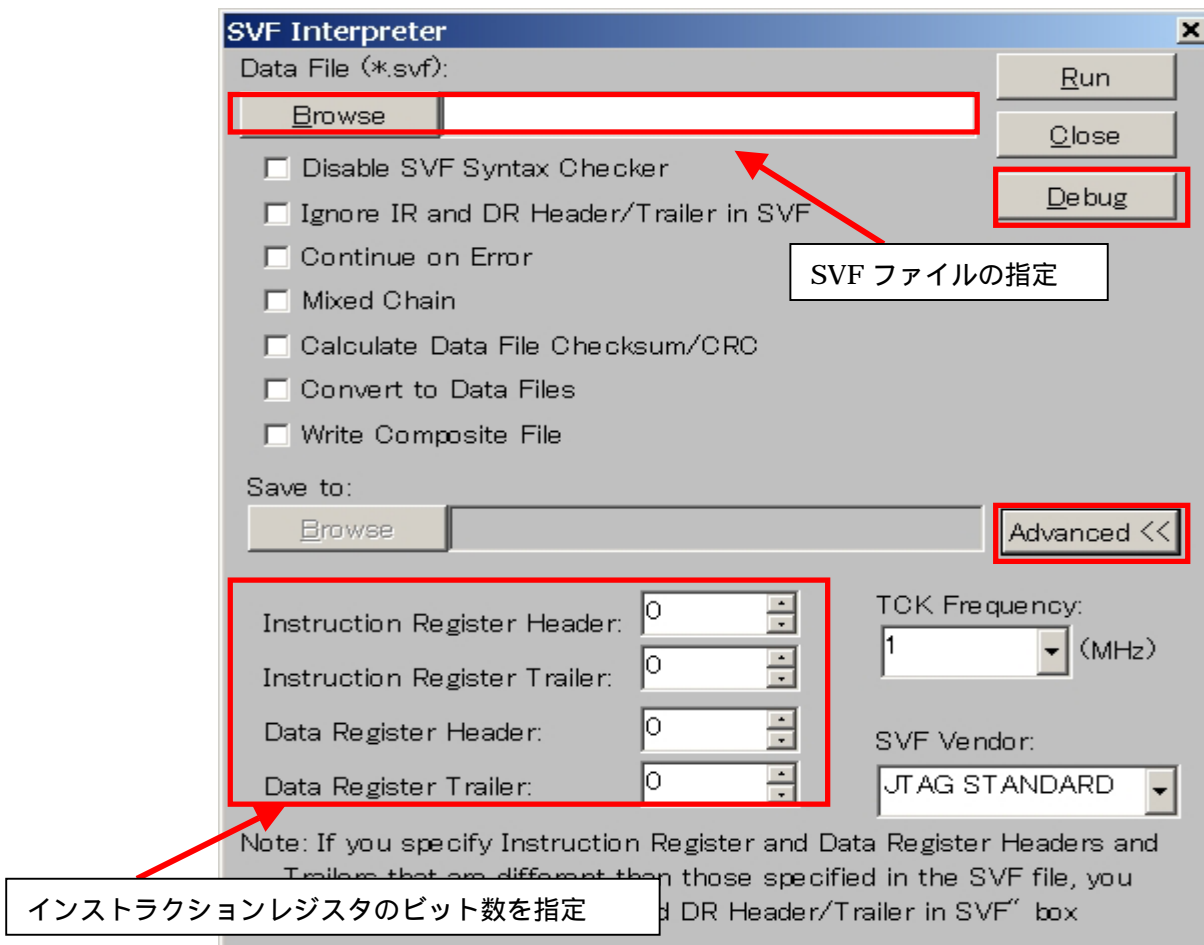


図 3-5 SVF インタープリタ

上記画面が開きましたら、次の手順で SVF デバッガを起動します。

生成した SVF ファイルを指定する。

Advanced ボタンをクリックする。

必要に応じて、インストラクションレジスタのビット数を設定します。

デバイス毎に異なります。

Debug ボタンをクリックし、SVF デバッガを起動します。

SVF デバッガを起動しますと、図 3-6のような画面が開きます。  
ここで、メニューの Configuration SVF Options を選択します(または SVF Options アイコンをクリックします)。

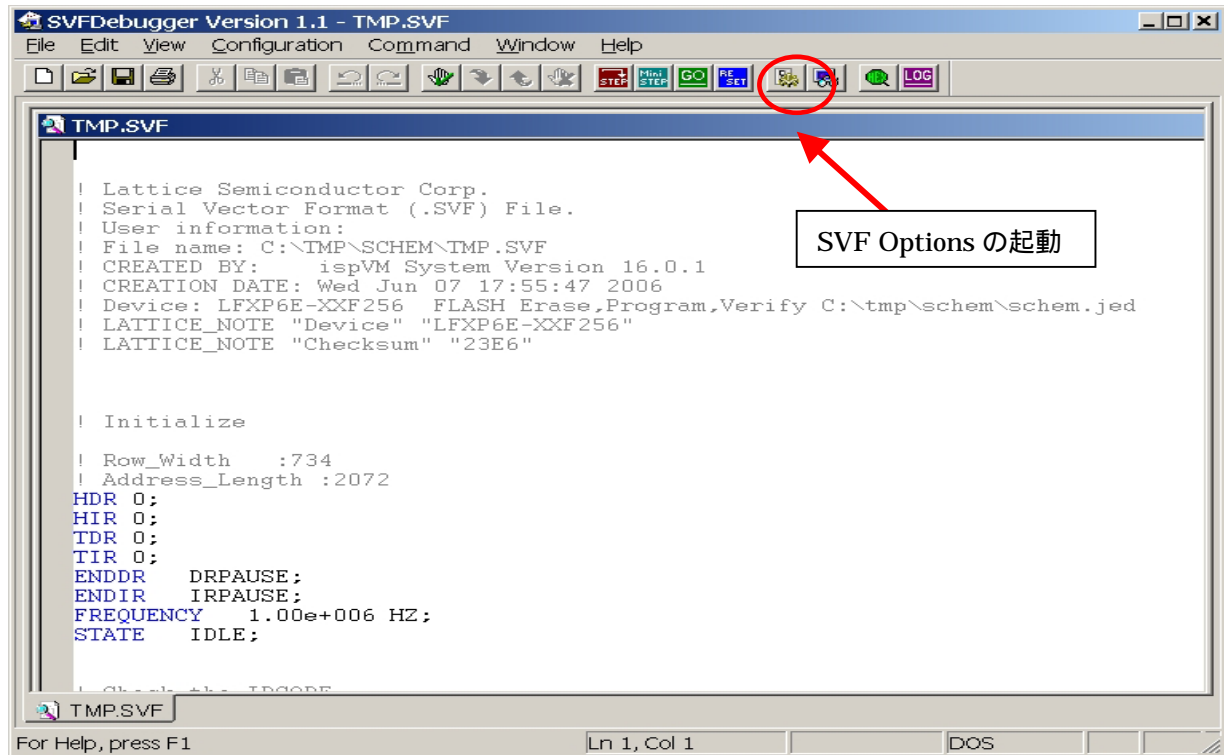


図 3-6 SVF デバッガ起動時の画面

SVF Options を起動しますと、図 3-7のような画面が起動します。  
ここで、JTAG チェーン上の単一デバイスをデバッグする場合、チェーン上にある前後のデバイスの命令レジスタ長を考慮した、ヘッダ長・トレイラ長を指定します。

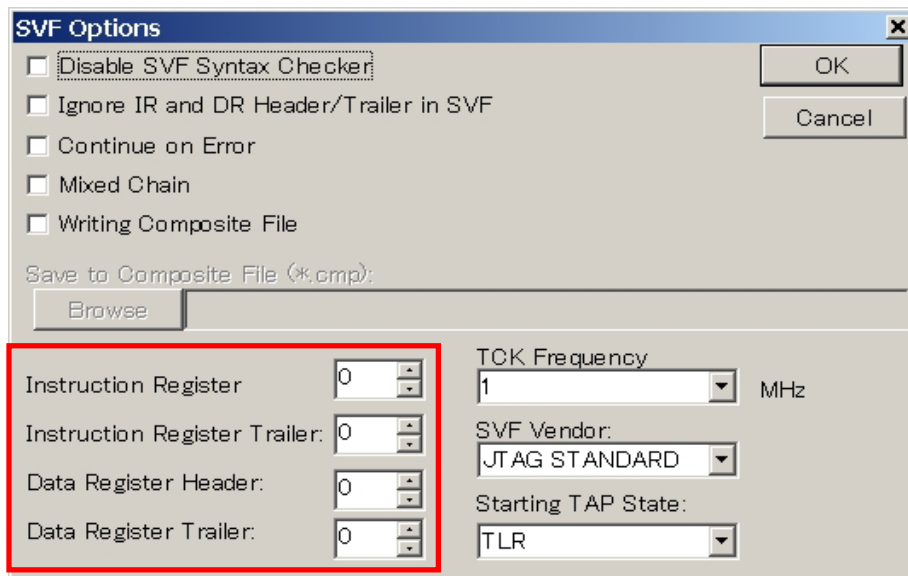


図 3-7 SVF Options 画面

図 3-8にヘッダ長・トレイラ長の設定例を示します。

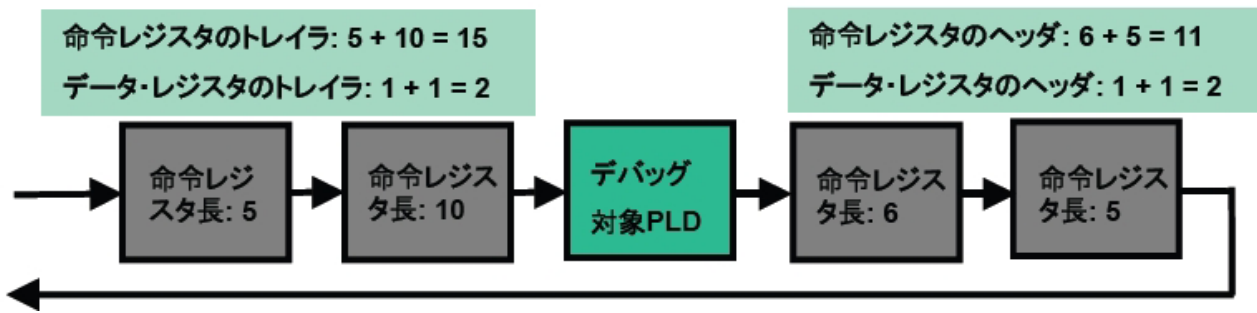


図 3-8 ヘッダ長・トレイラ長の設定例

続いて、メニューから Configuration SVF Output を選択しますと、のような画面が開きます。この画面では、デバッグしながら JTAG ピンや TAP コントローラの状態を確認することができます。

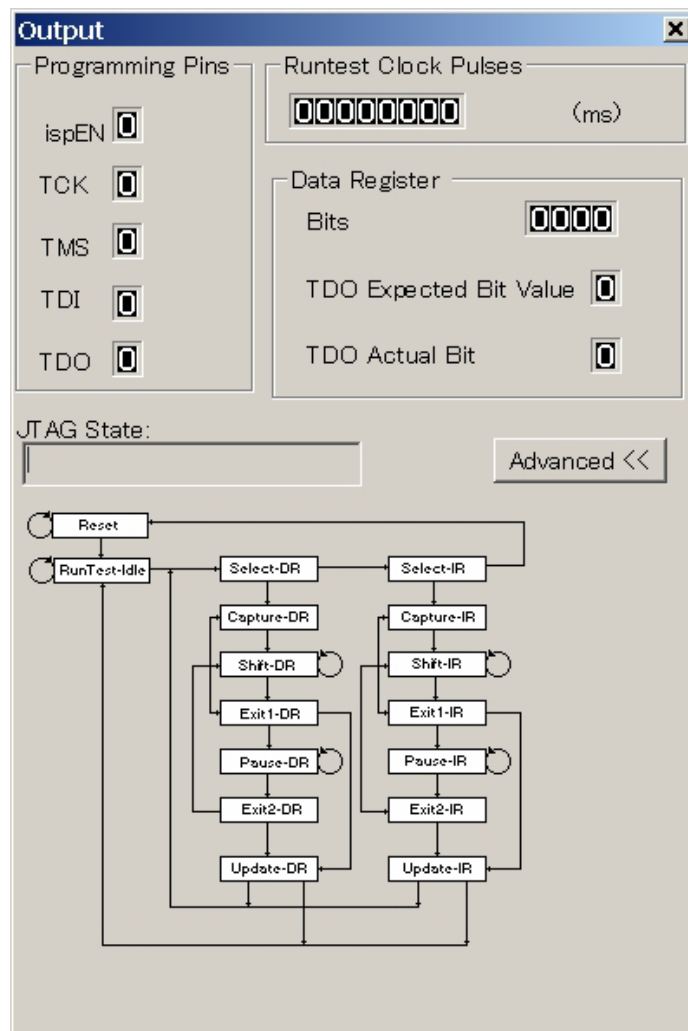


図 3-9 SVF Output 画面

以上でトラブルシューティング(プログラミング関連)は終了です。