

PAC-Designer

ユーザーマニュアル (*Platform Manager* 編)

目次

1	概要	4
1.1	PAC-Designer6.0.....	4
1.2	ライセンスの申請と設定.....	4
1.3	ブランクデザインから作成する方法.....	5
2	プラットフォーム・マネージャー ファミリの設計方法	6
2.1	FPGA ロジック 開発環境の選択	7
2.2	アナログ入力(Analog Inputs).....	8
2.3	CPLD デジタル入力(CPLD Inputs)	10
2.4	CPLD ロジック 出力(CPLD Logic Outputs)	11
2.6	高電圧出力(High Voltage Outputs)	12
2.8	クロック&タイマー	13
2.10	DC/DC コンバータ・ウィザード.....	15
2.12	マージン/トリム(Margin/Trim).....	23
2.14	トリム コンフィグレーションのオプション設定	26
2.15	トリム設定の一覧確認.....	27
2.17	I2C コンフィグレーション	28
2.19	CPLD Logic(シーケンス・コントローラ)	29
2.19.1	オプション設定	29
2.19.3	Pin の定義	31
2.19.5	インストラクションの設定方法.....	32
2.19.6	インストラクションの設定方法(Output).....	35
2.19.7	インストラクションの設定方法(Wait for Boolean Condition).....	37
2.19.8	インストラクションの設定方法(Wait for Timeout Value)	39
2.19.9	インストラクションの設定方法(Boolean Condition with Timeout)	40
2.19.10	インストラクションの設定方法(If/Then/Else)	41
2.19.11	インストラクションの設定方法(If / Then / Else with Timeout).....	42
2.19.12	インストラクションの設定方法(Goto).....	43
2.19.13	インストラクションの設定方法(Start Timer/Stop Timer).....	44
2.19.14	インストラクションの設定方法(例外処理).....	45
2.19.15	監視ロジック出力回路の生成(Supervisory Logic Equation)	46
2.20	FPGA Logic (FPGA ロジック)	47
2.20.1	PAC-Designer - LogiBuilder フロー	47
2.20.2	PAC-Designer - Text Editor / Schematic Editor フロー.....	48
2.20.3	PAC-Designer and DAIAMOND フロー	51
2.21	UES エディタ	52
3	コンパイル	53
3.1	コンパイル方法	53
3.1.1	PAC-Designer - LogiBuilder フローの場合	53
3.1.2	PAC-Designer - Text Editor / Schematic Editor フローの場合	53
3.1.1	PAC-Designer and DAIMOND フローの場合	54
3.2	コンパイル完了	55
3.2.1	エラー、ワーニングと対処方法.....	56
4	シミュレーション	57

4.1.1	CPLD Logic(シーケンス・コントローラ)のシミュレーション.....	57
4.1.2	クロックとシミュレーション時間の設定	58
4.1.3	入力信号波形の設定.....	58
4.1.4	シミュレーションの実行.....	59
4.1.5	汎用 HDL シミュレータを使用する方法	60
4.2	供給電源ランプレートのシミュレーション.....	61
4.3	FPGA Logic のシミュレーション	64
5	デバイスへの書き込み.....	65
5.1	PAC-Designer からの書き込み.....	65
5.1.1	ダウンロード・ケーブルの設定.....	65
5.1.2	デバイスへのダウンロード.....	66
5.2	ispVM System からの書き込み.....	67
6	I2C Utility.....	69
7	プラットフォーム・マネージャーを正しく使用するために(TN1223 より).....	71
7.1	デバイスの電源要求.....	71
7.2	I/O ピンのインタフェース	73
7.3	クロックピンの使用方法.....	74
	更新履歴.....	75

1 概要

プラットフォーム・マネージャ・ファミリの設計、シミュレーション、書き込みを行なうためには、PAC-Designer を使用します。PAC-Designer 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。

初めて、PAC-Designer を使用する場合は、弊社日本語 HP【<http://www.latticesemi.co.jp/>】で公開しています、日本語チュートリアルを合わせて参照することを推奨します。

1.1 PAC-Designer6.1

- ① DAIAMOND v1.3 をサポートしました。

1.2 ライセンスの申請と設定

PAC-Designer6.1 は無料ツールで、無償のライセンスも必要としません。
インストール直後からご利用いただけます。

1.3 ブランクデザインから作成する方法

新規デザイン作成時は、PAC-Designer を起動し、デバイスの選択を行いません。デバイスの選択は、PAC-Designer のメニューから File→New を選択しますと、図 1-1 のような画面が開きます。

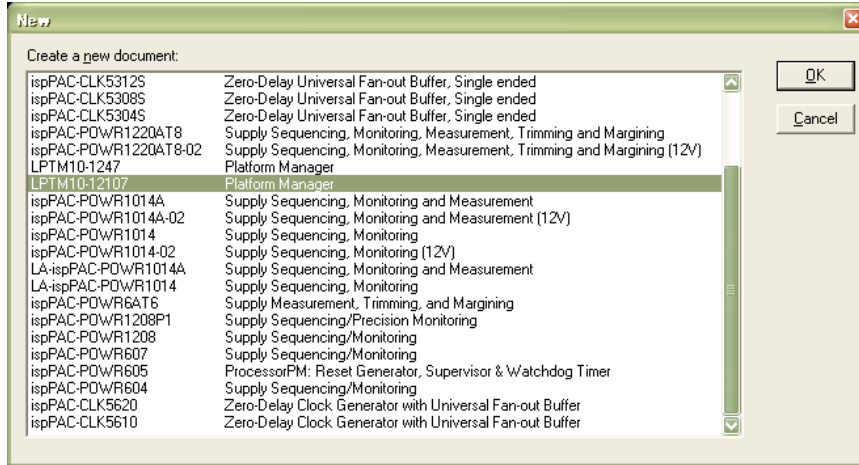


図 1-1 デバイスの選択

上記画面にて、設計するデバイスを選択し OK ボタンをクリックします。

リファレンス・デザインから作成する方法

弊社 Web 上で公開されているリファレンス・デザインを参照することができます。

Web 上から、Downloads => IP and Reference Design を選択し(図 1-2)、ページ右下の Platform Manager のコーナーからリファレンス・デザインをダウンロードします。

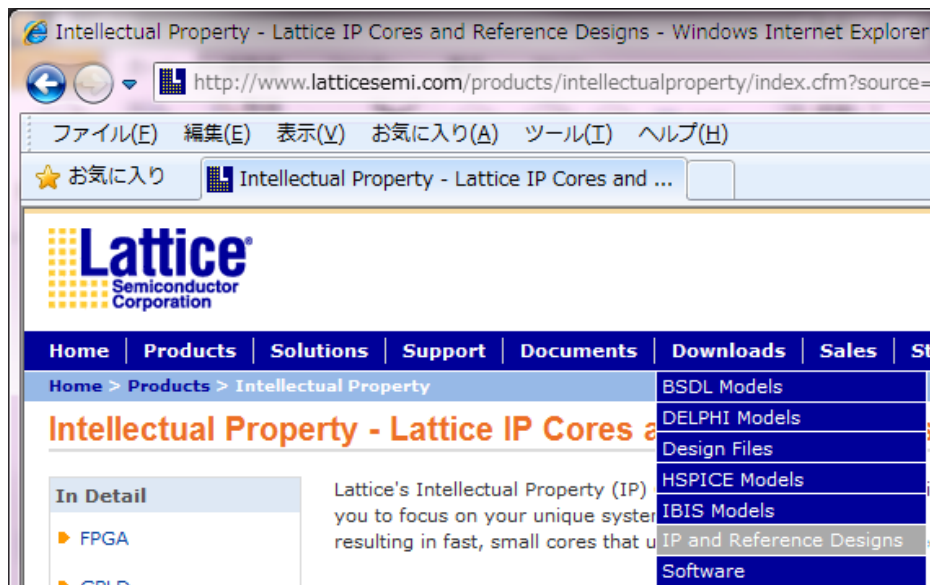


図 1-2 リファレンス・デザインの場合

2 プラットフォーム・マネージャー ファミリの設計方法

本節では、プラットフォーム・マネージャー ファミリ(LPTM10-1247, LPTM10-12107)の設計方法について、説明いたします。ここでは、代表して LPTM10-12107 を例に説明いたします。

デバイスの選択画面にて、LPTM10-12107 を選択しますと図 2-1 のような画面が開きます。

本画面にて、各ブロックをダブルクリックしますと、それぞれの設計画面が開きます。

各ブロックの詳細については、次ページ以降で説明いたします。

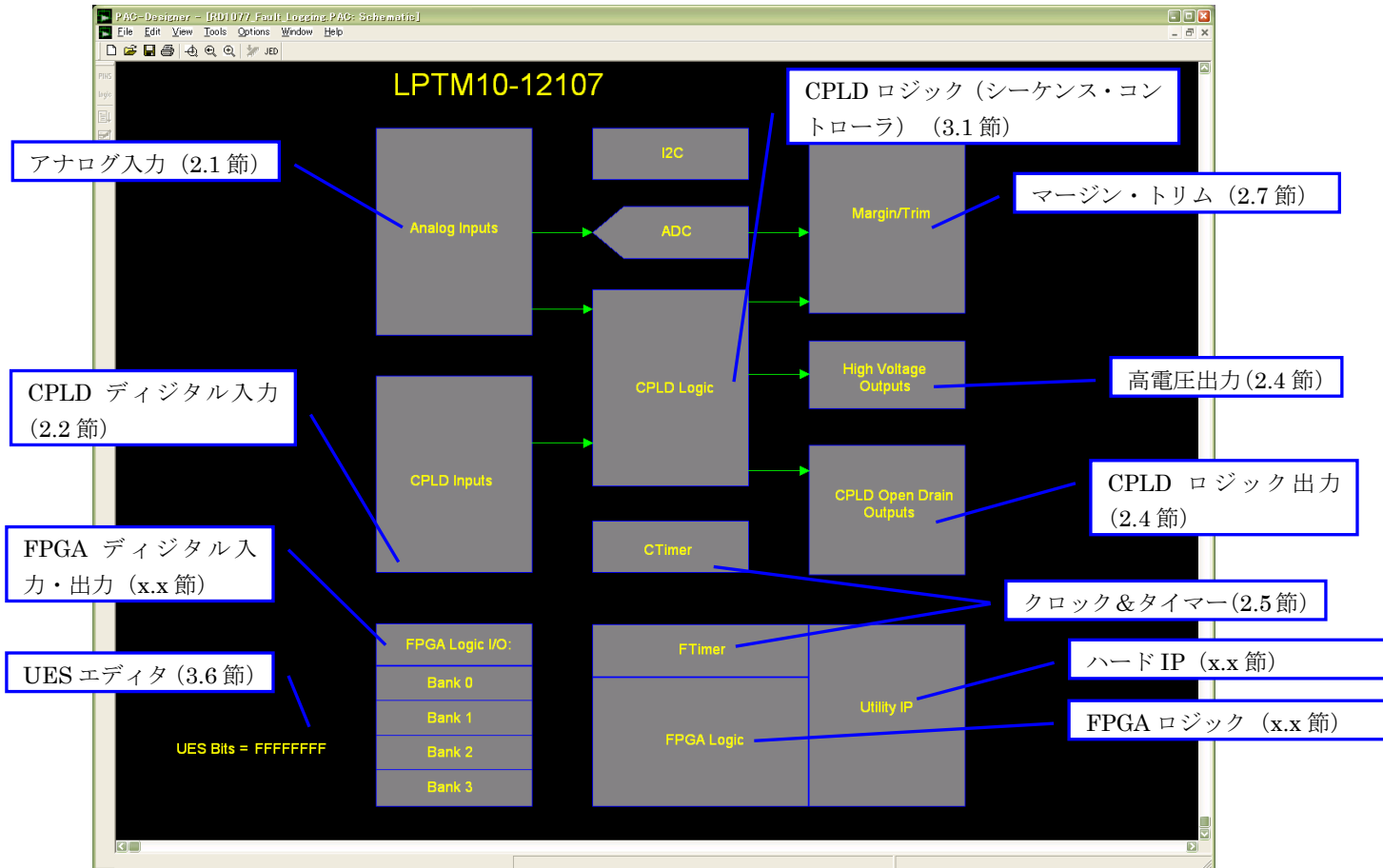




図 2-1 LPTM10-12107 メインウインドウ

基本操作方法

ツール画面上でクリック可能な状態になると、カーソルが以下の状態に変化し操作が可能となります。各種設定はモジュールの設定画面にて行います。

 もしくは  で、そのモジュールの設定画面へと移動します。

2.1 FPGA ロジック開発環境の選択

対象デバイス : LPTM10-1247, LPTM10-12107

FPGA ロジック部の開発環境を選択します。PAC-Designer のメニューより、Option => FPGA Environment をクリックし、環境設定画面を開きます(図 2-2)。

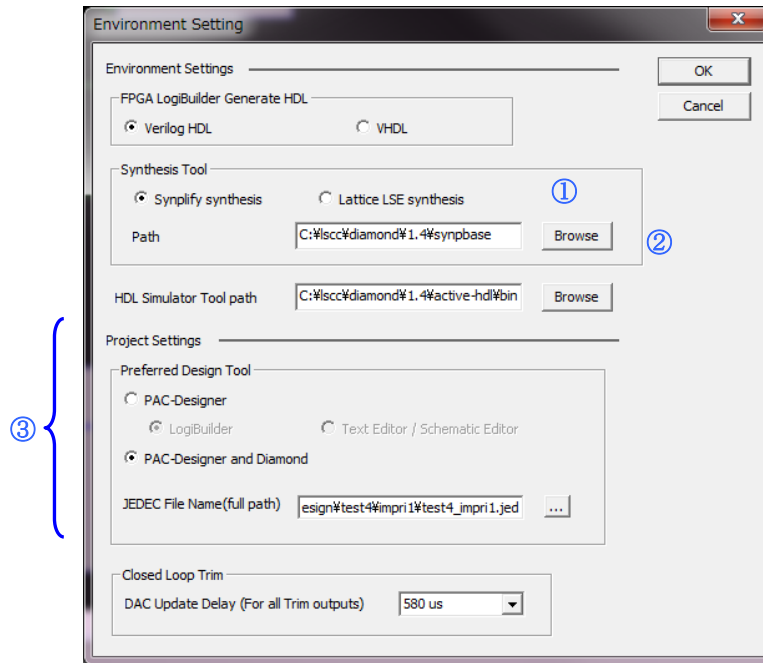


図 2-2 FPGA ロジック開発環境の選択画面

上記画面が開きましたら、下記 3 つの設定を行ないます。

- ① 合成ツールの選択をします。
- ② HDL シミュレータのインストール・パスを設定します (HDL シミュレータを使用しない場合は設定不要)。
- ③ 下記 3 種類の FPGA 開発環境から 1 つ選択します。チェックボックスは、プラットフォーム・マネージャーを選択したプロジェクトを開いている場合に有効となります。チェックボックスがグレーアウトしている場合、新規でプラットフォーム・マネージャーのプロジェクトを作成してから選択します。下記環境の(イ),(ウ)では、弊社の開発ツール DAIAMOND v1.3 以降が必要です。

(ア) PAC-Designer - LogiBuilder フロー

HDL や回路図入力を必要としません。FPGA 部の設計にも CPLD 部の設計と同様に、LogiBuilder を使用します。

(イ) PAC-Designer - Text Editor / Schematic Editor フロー

HDL や回路図によりデザインをエントリーします。設計時の UI は PAC-Designer のみを使用しますが、バックグラウンドで DAIAMOND を使用する為、DAIAMOND のインストールが必要です。

(ウ) PAC-Designer and Daiamond フロー

HDL や回路図によりデザインをエントリーします。FPGA 部の設計には、別途 DAIAOND を別途使用します。DAIAMOND のインストールが必要です。

2.2 アナログ入力(Analog Inputs)

対象デバイス : LPTM10-1247, LPTM10-12107

アナログ入力ブロックでカーソルが□に変わる箇所をダブルクリックすると、Logic I/O Assignment ウィンドウの Analog Inputs タブが開きます(図 2-4)。

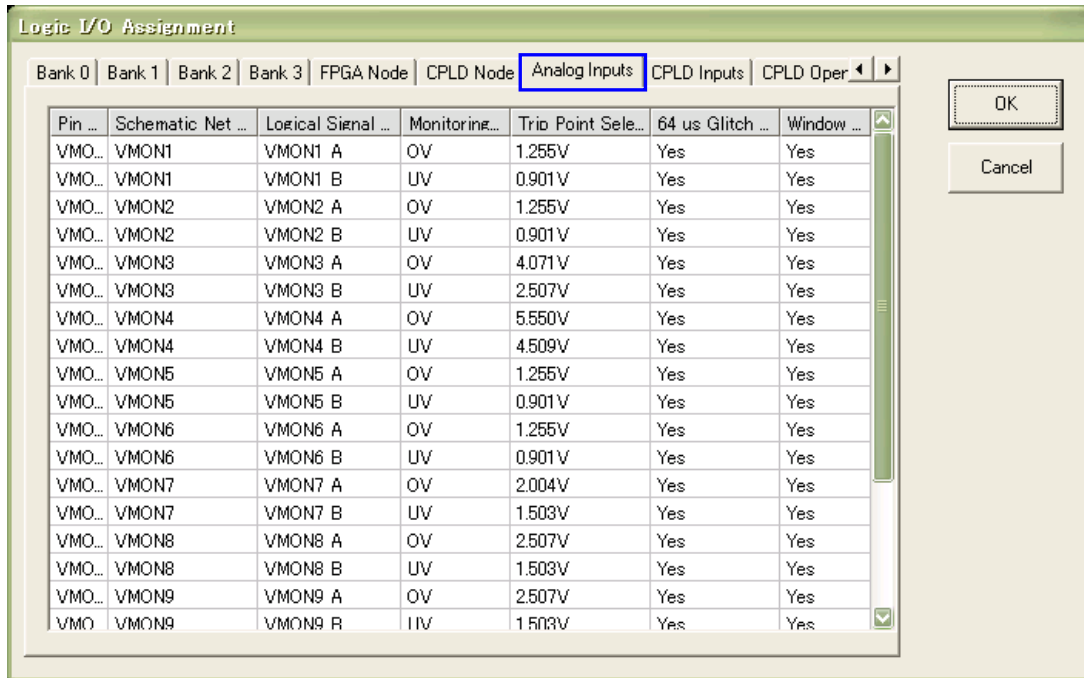



図 2-3 アナログ入力設定画面

図 2-3 の画面上にて、カーソルが  に変わる箇所をダブルクリックしますと、図 2-4 の(Analog Input Setting)画面が起動します。

なを、アナログ入力ピンは LPTM10-1247, LPTM10-12107 共に 12 系統の入力を持ちますが、LPTM10-1247 ではシングルエンド入力 5 系統 + 差動入力 7 系統で構成され、LPTM10-12107 では差動入力 12 系統で構成されています。

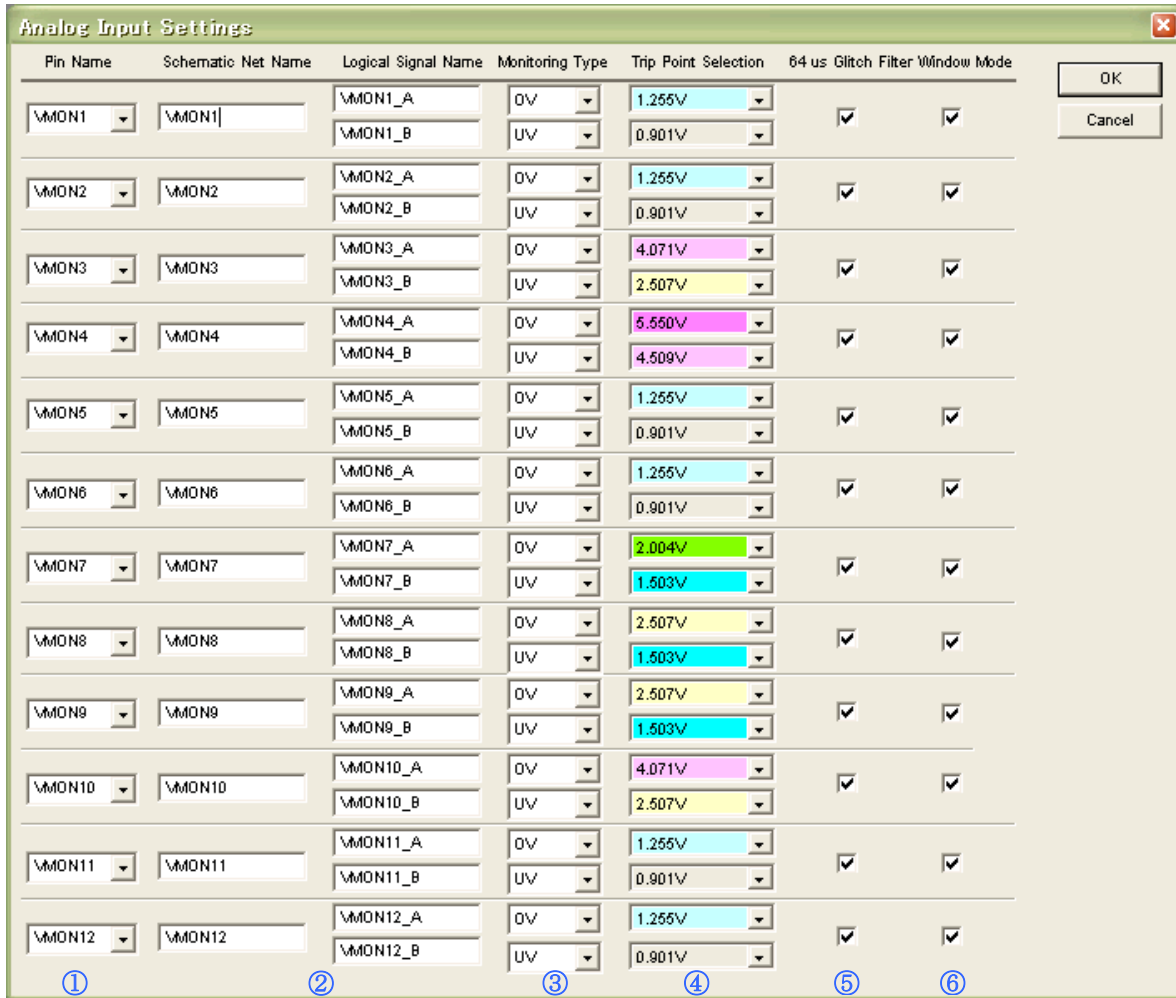


図 2-4 アナログ入力設定画面(LPTM10-12107)

上記画面が開きましたら、下記 6 つの設定を行ないます。

- ① 入力ピンを選択します(デフォルトのままでも構いません)。
- ② 各ピンの信号名を定義します。Schematic Net Name は回路図の入力信号名として、Logical Signal Name は回路図の出力信号名と後述のシーケンス・コントローラ内で使用されます(デフォルトのままでも構いません)。尚、プラットフォーム・マネージャーでは、1入力に対して閾値を二つ設定することができ、1 入力に対して二つの Logical Signal Name を任意に設定可能です。
- ③ 監視条件を、OV(Over Voltage)、UV(Under Voltage)から選択します。OV/UV いずれのモードでも、測定電圧が設定値を上回った場合に“1”、設定値を下回った場合に“0”が返り値となります。2 つのモードの相違点は、ヒステリシス特性の違いのみです。
- ④ プルダウンメニューより、コンパレータのトリップポイント電圧を設定します。(基準電圧に対して色分けされています)
- ⑤ 64us のグリッチフィルタを使用するか否かを設定します。チェックでグリッチフィルタを使用します。
- ⑥ ウィンドウ・コンパレータ・モードにするか否かを設定します。ウィンドウ・コンパレータ・モードに設定した場合、②で設定した 2 つの値の範囲内であるか否かを監視します。

各デバイスの、トリップポイント選択範囲は以下のようになります。

LPTM10-1247 : 0.075V, 0.668V~5.734V(OV), 0.075, 0.664V~5.703V(UV)

LPTM10-12107 : 0.075V, 0.668V~5.734V(OV), 0.075, 0.664V~5.703V(UV)

※0.075V(75mV)は、“電源が完全にオフした事の検出用”として用いる事が出来ます。

2.3 CPLD デジタル入力(CPLD Inputs)

対象デバイス : LPTM10-1247, LPTM10-12107

CPLD デジタル入力ブロック(カーソルが□に変わる箇所)をダブルクリックすると、図 2-5 の(CPLD Inputs)画面が起動します。

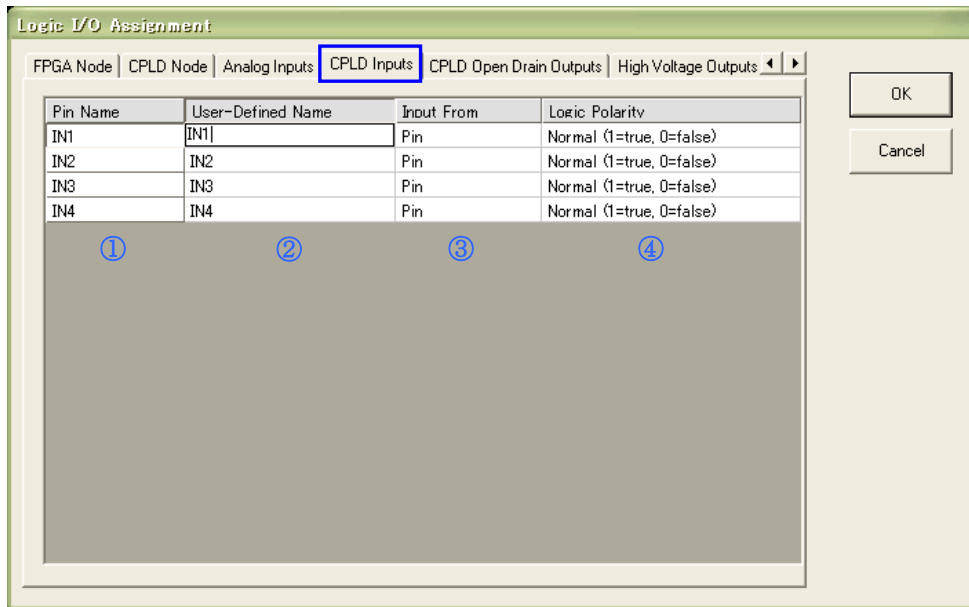


図 2-5 デジタル入力設定画面

ここでは、デジタル入力ピンの信号名をユーザで任意に設定することができます(デフォルトのままでも構いません)。

上記画面が開きましたら、下記 4 つの設定を行ないます。

- ① 入力ピンを選択します(デフォルトのままでも構いません)。
- ② 各ピンの信号名を定義します。User-Defined Name は、後述のシーケンス・コントローラ内で使用されます(デフォルトのままでも構いません)。
- ③ デジタル入力信号 1(デフォルト名 IN1)の制御を、pin の入力状態もしくは JTAG のユーザレジスタ設定の何れかから選択します。デジタル入力信号 2 から 4(デフォルト名 IN2 から IN4)の制御を、pin の入力状態もしくは IC2 レジスタ設定の何れかから選択します。
- ④ ロジックの極性を選択します。Normal(Active High)か Active Low が選択可能です。

2.4 CPLD ロジック出力(CPLD Logic Outputs)

対象デバイス : LPTM10-1247, LPTM10-12107

CPLD ロジック出力ブロック(カーソルが□に変わる箇所)をダブルクリックすると、図 2-6 の(CPLD Open Drain Outputs)画面が起動します。

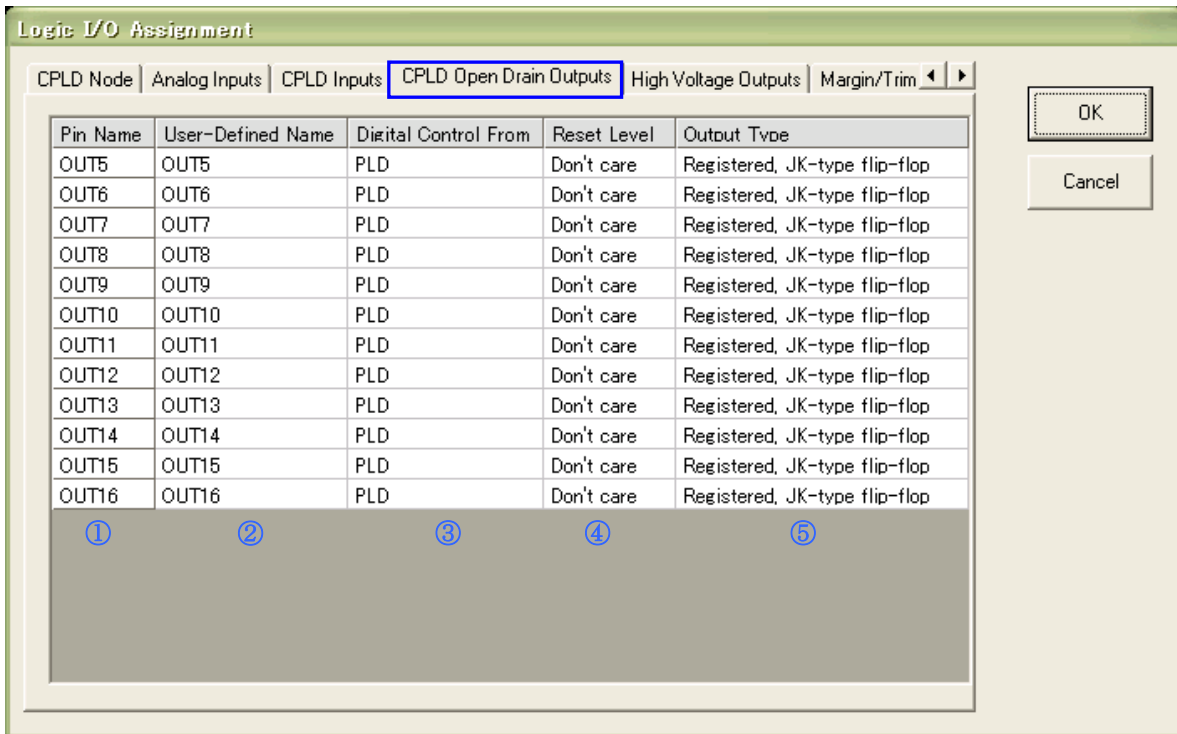


図 2-6 CPLD ロジック出力設定画面

上記画面が開きましたら、下記つの設定を行ないます。

- ① 出力ピンを選択します(デフォルトのままでも構いません)。
- ② 各ピンの信号名を定義します。User-Defined Name は、後述のシーケンス・コントローラ内で使用されます(デフォルトのままでも構いません)。
- ③ 出力状態を PLD で制御するか I2C バス経由で制御するかを設定します。
- ④ パワーオンリセット解除後の論理を、“Don't Care”, “Set low”, “set high”の何れかから選択します。
- ⑤ 出力タイプを、“JK-FF タイプ”, “D-FF タイプ”, “T-FF タイプ”, “組み合わせ論理タイプ”の何れかから選択します。

2.6 高電圧出力(High Voltage Outputs)

対象デバイス : LPTM10-1247, LPTM10-12107

プラットフォーム・マネージャーでは 4 つの出力ピンを高電圧出力(High Voltage Outputs)として設定することができます。また、高電圧出力ピンは、オープンドレイン出力とすることも可能です。高電圧出力は、電源シーケンスを制御するために、ボード上の N チャネル MOSFET のゲートをドライブするのに必要な、N チャネル MOSFET ゲートドライバとして使用することができます。

高電圧出力ブロック(カーソルが□に変わる箇所)をダブルクリックすると、図 2-7 の (High Voltage Outputs) 画面が開きます。

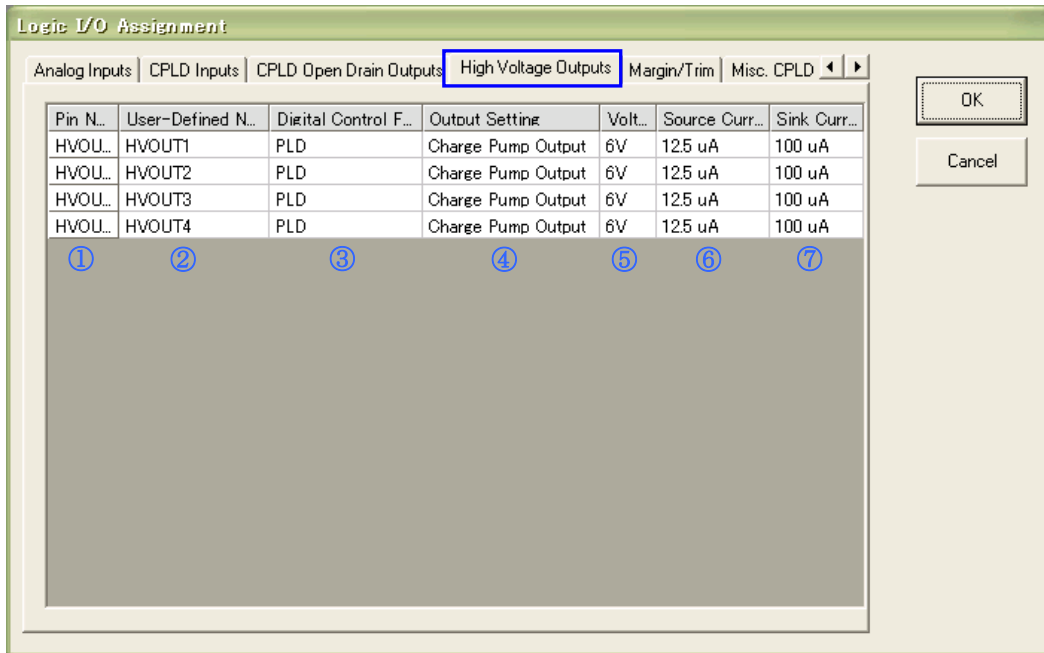


図 2-7 高電圧出力設定画面

上記画面が開きましたら、下記 7 つの設定を行ないます。

- ① 出力ピンを選択します(デフォルトのままでも構いません)。
- ② 各ピンの信号名を定義します。User-Defined Name は、後述のシーケンス・コントローラ内で使用されます(デフォルトのままでも構いません)。
- ③ 出力ピンの制御を PLD で行なうか I2C バス経由で行なうかを設定します。
- ④ 高電圧出力とする場合は、チャージポンプ出力(Chare Pump Output)にチェックし、出力電圧および電流値を設定します。オープンドレイン出力として使用する場合は、オープンドレイン出力(Open Drain Logic Output)にチェックをします。オープンドレイン出力の場合は、通常の出カレベルとなります。
- ⑤ チャージポンプ出力の出力電圧を、“6V”、“8V”、“10V”、“12V”から選択します。
- ⑥ チャージポンプ出力のソース電流を、“12.5uA”、“25uA”、“50uA”、“100uA”から選択します。
- ⑦ チャージポンプ出力のシンク電流を、“100uA”、“250uA”、“500uA”、“3000uA”から選択します

2.8 クロック&タイマー

対象デバイス : 対象デバイス : LPTM10-1247, LPTM10-12107

プラットフォーム・マネージャーには、CPLD 部で使用可能な CTimer と、FPGA 部で使用可能な FTimer があります。CTimer ブロック(カーソルが□に変わる箇所)をダブルクリックすると、図 2-8 の(Clock & Timers)画面が開きます。

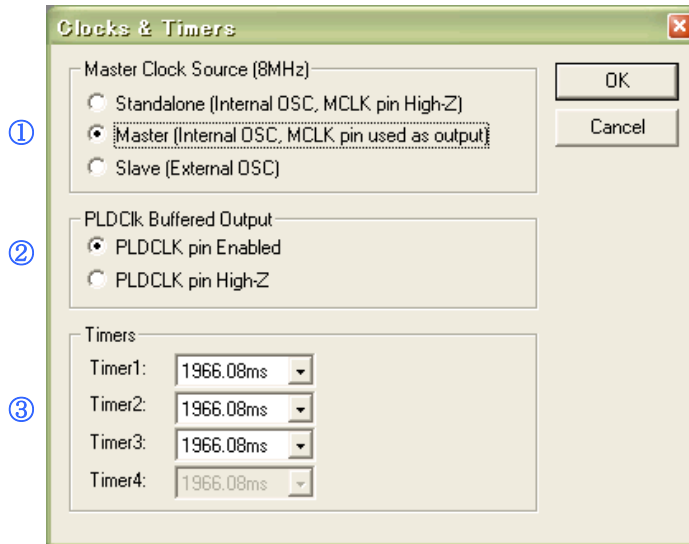


図 2-8 クロック&C タイマー設定画面

上記画面が開きましたら、下記 3 つの設定を行ないます。

- ① クロックソースを設定します。
 - Standalone : 内部 OSC を使用し、MCLK ピンは Hi-Z
 - Master : 内部 OSC を使用し、MCLK ピンにクロックを出力
 - Slave : 外部のクロックを使用し、MCLK ピンは入力ピン
- ② PLDCLK ピンの出力を設定します。
 - PLDCLK pin Enabled : PLDCLK ピンからクロックを出力
 - PLKCLK pin High-Z : PLDCLK ピン出力は Hi-Z 状態
- ③ 1~3 種類のタイマー値をプルダウンメニューから選択します。Timer4 については、FPGA 部で使用する FTimer の為にリザーブされます。

ここで、設定したタイマー値はシーケンス・コントローラ内部の、タイマー値を使用するインストラクションで使用することが出来ます。上記の通り、タイマー値は 3 つ個別に設定することが可能で、インストラクション毎に任意のタイマー値を用いることが出来ます。

FTimer ブロック(カーソルが□に変わる箇所)をダブルクリックすると、図 2-9 の (Clock & Timers) 画面が開きます。Add ボタンをクリックして、タイマー設定を追加します。一度追加したタイマーを削除するには、Remove ボタンをクリックします。

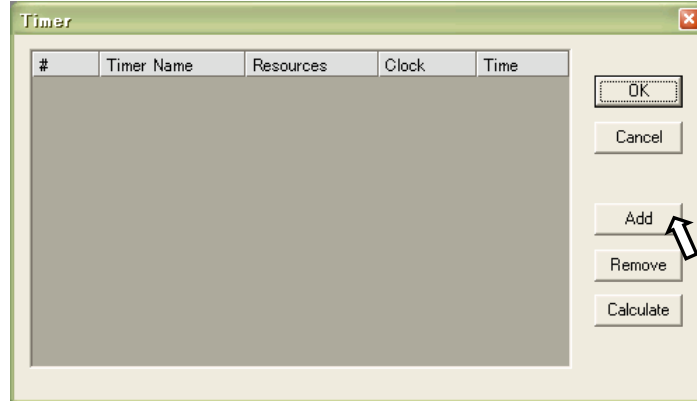


図 2-9 F タイマー設定画面

“Time”に希望の時間を入力して、Calculate ボタンをクリックすると“Resource”に必要なスライス数を表示します。FPGA 部のリソースに空きがある限り、タイマーの個数と上限時間が増やせます。

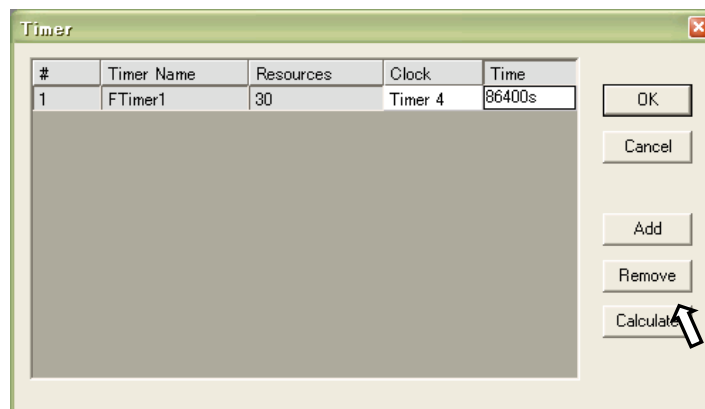


図 2-10 F タイマー設定画面

2.10 DC/DC コンバータ・ウィザード

対象デバイス : LPTM10-1247, LPTM10-12107

トリミング機能付 DC/DC コンバータを用いてトリミング機能を実現するためには、DC/DC コンバータの特性を設定する必要があります。

メニューバーの「DC-DC」アイコンをクリックすると、DC/DC コンバータ・ウィザードが起動します(アイコンが表示されない場合、メニューの View→Margin Toolbar にチェックをつけて表示させます)。

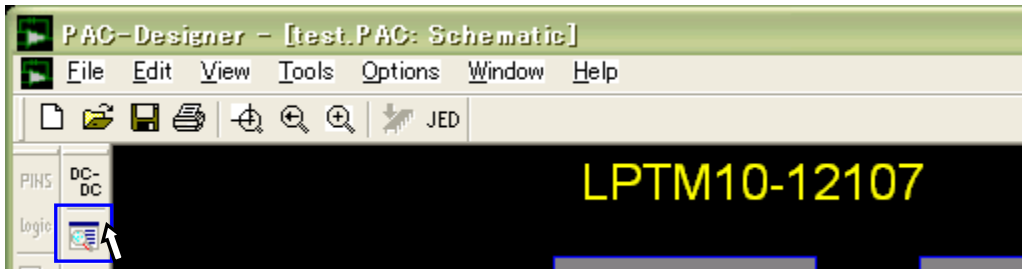


図 2-11 DC/DC コンバータ・ウィザードの起動

DC/DC コンバータ・ウィザードが起動したら、「New」ボタンをクリックし、ライブラリ名を入力し、新規ライブラリの生成を行ないます。名前を付けて OK をクリックしたら、次へ(N)をクリックします。

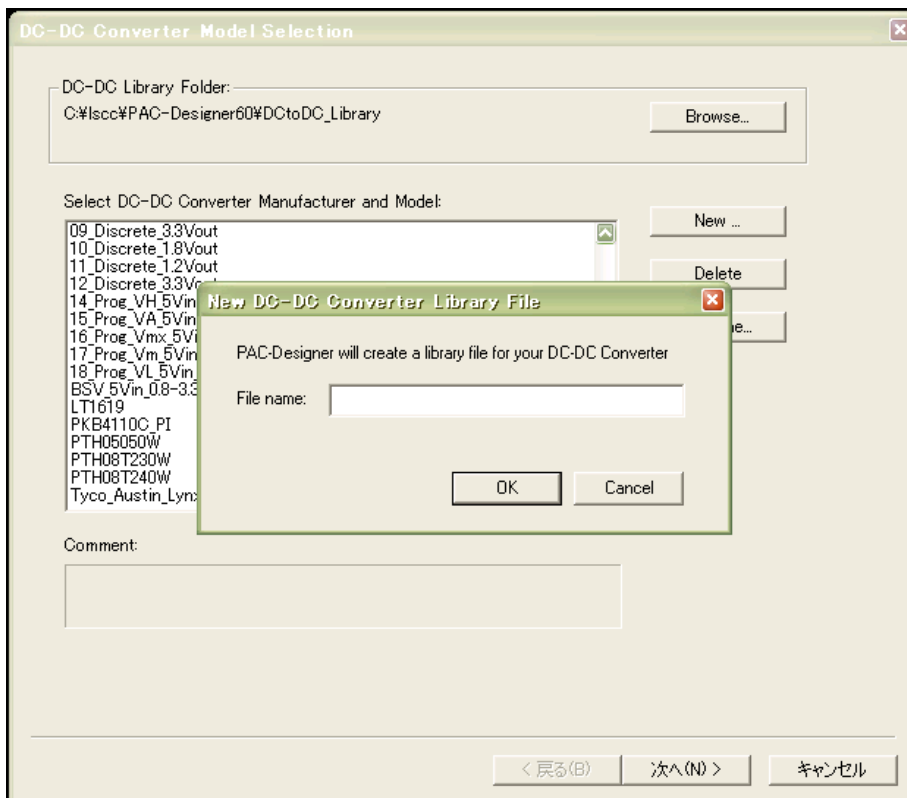


図 2-12 新規 DC/DC コンバータ・ライブラリーの生成

続いて、DC/DC コンバータのタイプを以下の 4 種から選択します。

1. トリムピンと V_{out} もしくは GND を抵抗で接続し、抵抗値により基準電圧に対して数%の出力電圧調整が可能なタイプ
2. ピンと GND を抵抗で接続することで、抵抗値により広いレンジで出力電圧を変更できるタイプ
3. トリムピンと V_{out} を抵抗で接続することで、抵抗値により広いレンジで出力電圧を変更できるタイプ
4. 外部のフィードバック抵抗により出力電圧を設定するタイプ

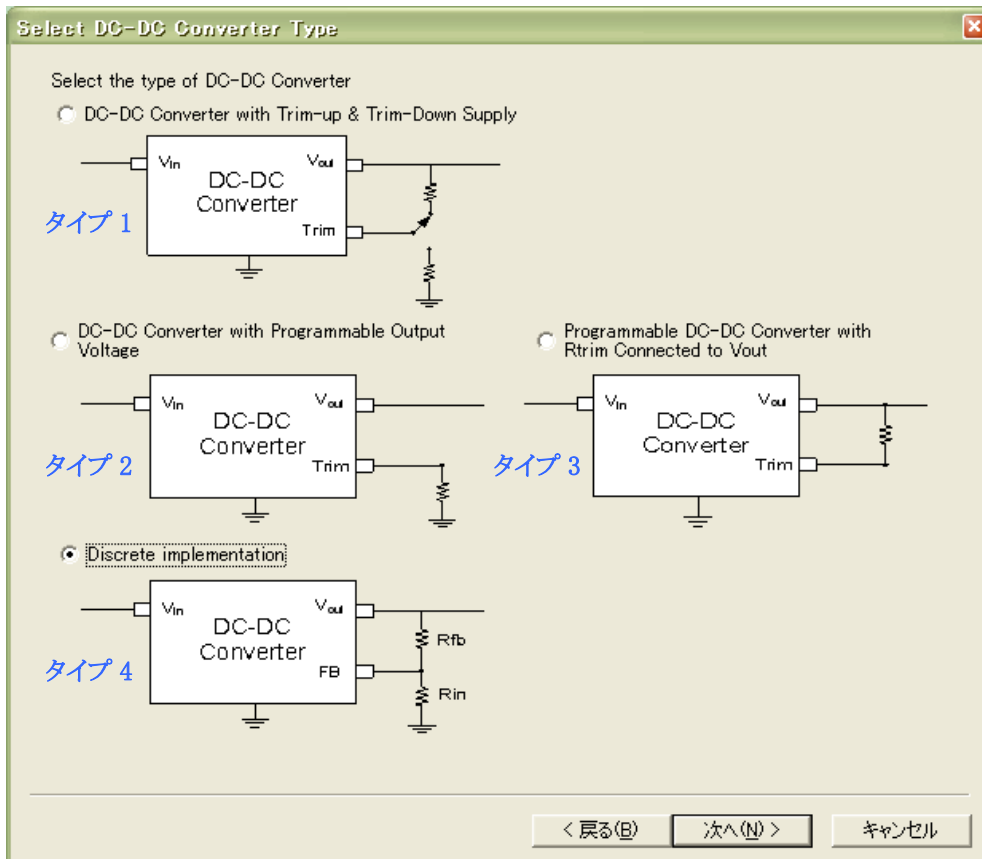


図 2-13 DC/DC コンバータのタイプ選択画面

タイプを選択したら、次へ(N)をクリックします。

続いて、選択したDC/DCコンバータのパラメータ設定を行ないます。使用するDC/DCコンバータのデータシートを参照し、パラメータを設定します。
お使いになります DC/DC コンバータのデータシートをお手元にご用意下さい。

タイプ1

トリムピンと Vout もしくは GND を抵抗で接続し、抵抗値により基準電圧に対して数%の出力電圧調整が可能なタイプ

下記の画面が起動しましたら、パラメータの入力を行ないます。入力が完了しましたら「Save」ボタンをクリックしてライブラリを保存し、「完了」ボタンをクリックします。

DC-DC Converter Datasheet Example Configurations

Nominal Output Voltage
With Trim Resistor Open: 3.3 V

Parameter	Example1 R to GND	Example2 R to GND	Example3 R to Vout
Resistor	33 V	33 V	33 V
Voltage with Resistor	3.63 V	3.15 V	2.7 V
Resistor Value to Set	5447 ohms	12333 ohms	10333 ohms

Comment: 3.3V DC/DC Converter

Save configurations to library file: Power Suplyer ****

Buttons: <戻る(B) | 完了 | キャンセル

データシートを参照し、抵抗値と電圧値の構成例を入力します。抵抗を介して GND と接続した場合の例を 2 つ。Vout と接続した場合の例を 1 つ入力します。
入力する値は、実際に使用する値である必要はございません。

データシートを参照し、トリム用レジスタ未使用時の通常電圧を入力

保存するライブラリ名を入力します。必要に応じコメントも記載しておくと便利です。

図 2-14 DC/DC コンバータのパラメータ設定画面(タイプ 1)

タイプ 2

トリムピンと GND を抵抗で接続することで、抵抗値により広いレンジで出力電圧を変更できるタイプ

下記の画面が起動しましたら、内部の基準電圧(Vref)もしくは内部の基準抵抗値(Rref)及び基準電流(Iref)の入力を行ないます。入力が完了しましたら「次へ」ボタンをクリックします。

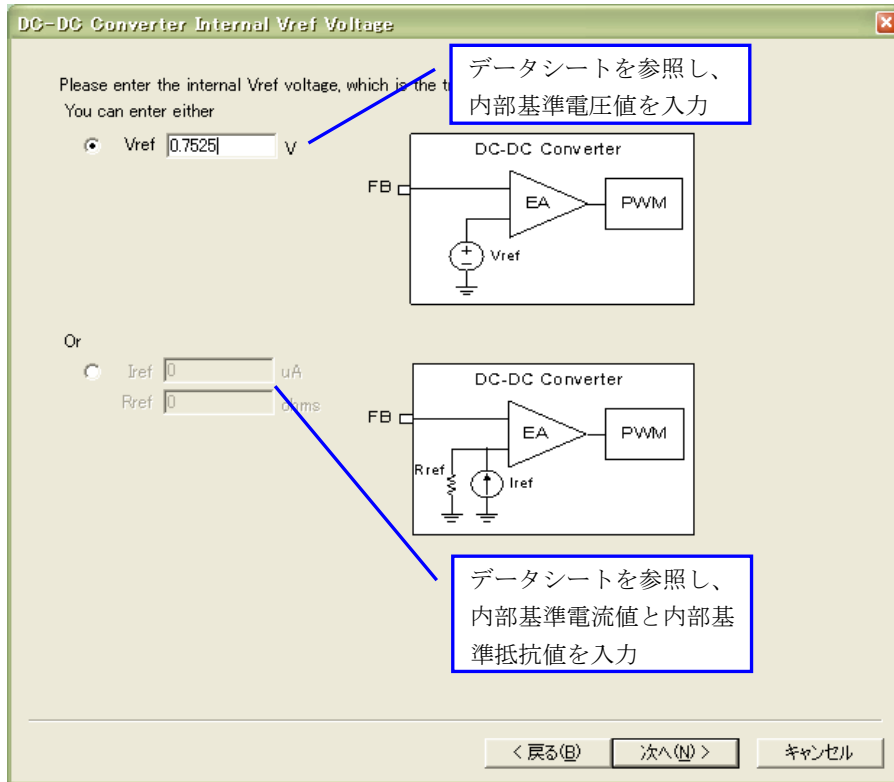


図 2-15 DC/DC コンバータのパラメータ設定画面(タイプ 2-1)

下記の画面が起動しましたら、パラメータの入力を行ないます。入力が完了しましたら「Save」ボタンをクリックしてライブラリを保存し、「完了」ボタンをクリックします。

DC-DC Converter Datasheet Example Configurations

Nominal Output Voltage With Trim Resistor Open: 0.7525 V

DC-DC Converter Datasheet Example Configuration Equations			
	Example1 R to GND	Example2 R to GND	Example3 R to Vout
Trim Resistor	0.7525 V	0.7525 V	0 V
Trimmed/Margined Output Voltage with Trim Resistor Connected	1.2 V	3.3 V	0 V
Trim Resistor in ohms Required to Set Output Voltage as Above	41973 ohms	3160 ohms	0 ohms

Comment: 2.5V DC/DC Converter

Save configurations to library file: Power Suplyer *****

Buttons: < 戻る(B), 完了, キャンセル

図 2-16 DC/DC コンバータのパラメータ設定画面(タイプ 2-2)

タイプ 3

トリムピンと V_{out} を抵抗で接続することで、抵抗値により広いレンジで出力電圧を変更できるタイプ

下記の画面が起動しましたら、内部の基準電圧(V_{ref})もしくは内部の基準抵抗値(R_{ref})及び基準電流(I_{ref})の入力を行ないます。入力が完了しましたら「次へ」ボタンをクリックします。

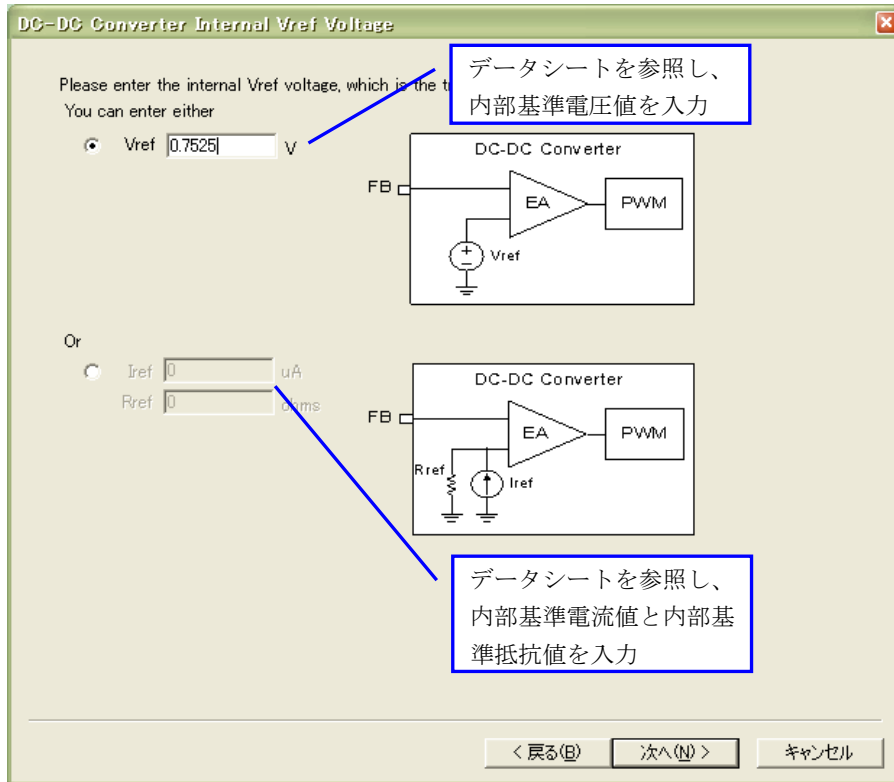


図 2-17 DC/DC コンバータのパラメータ設定画面(タイプ 3-1)

下記の画面が起動しましたら、パラメータの入力を行ないます。入力が完了しましたら「Save」ボタンをクリックしてライブラリを保存し、「完了」ボタンをクリックします。

データシートを参照し、トリムピン開放時のVout電圧を入力

データシートを参照し、抵抗値と電圧値の構成例を2つ入力します。入力する値は、実際に使用する値である必要はありません。

保存するライブラリ名を入力します。必要に応じコメントも記載しておくと便利です。

	Example1 R to Vout	Example2 R to Vout	Example3 R to Vout
Trim Resistor	0.7525 V	0.7525 V	0 V
Trimmed/Margined Output Voltage with Trim Resistor Connected	1.2 V	3.3 V	0 V
Trim Resistor in ohms Required to Set Output Voltage as Above	41973 ohms	3160 ohms	0 ohms

Comment: 25V DC/DC Converter

Save configurations to library file: Power Suplyer

< 戻る(B) 完了 キャンセル

図 2-18 DC/DC コンバータのパラメータ設定画面(タイプ 3-2)

タイプ 4

外部のフィードバック抵抗により出力電圧を設定するタイプ

下記の画面が起動しましたら、パラメータの入力を行ないます。入力が完了しましたら「Save」ボタンをクリックしてライブラリを保存し、「完了」ボタンをクリックします。

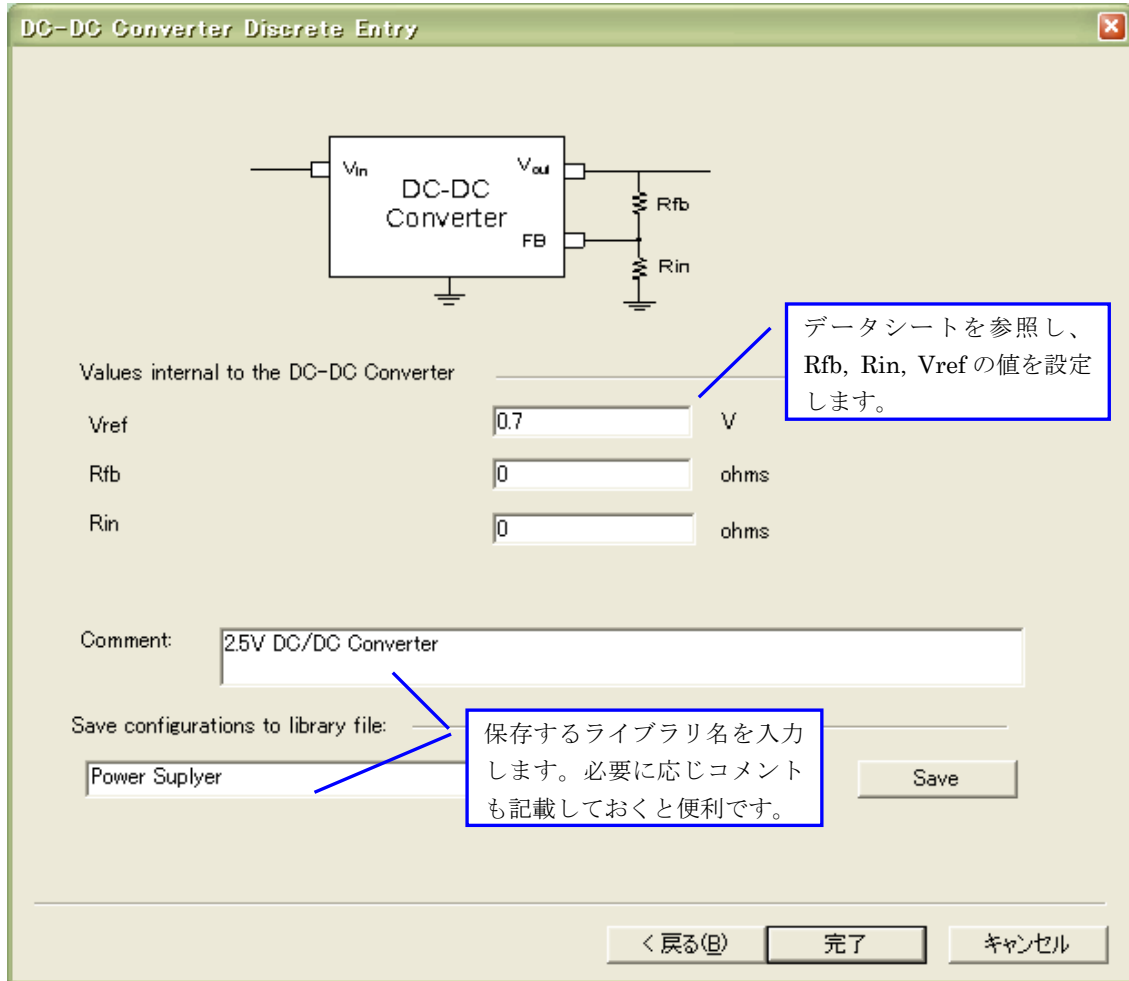


図 2-19 DC/DC コンバータのパラメータ設定画面(タイプ 4)

2.12 マージン/トリム(Margin/Trim)

対象デバイス : LPTM10-1247, LPTM10-12107

閉ループのトリマ調整時の、DAC のアップデート遅延時間を設定します。PAC-Designer のメニューより、Option => FPGA Environment をクリックし、環境設定画面を開きます(図 2-20)。閉ループのトリマ調整時の、DAC のアップデート遅延時間をプルダウンメニューより設定します。

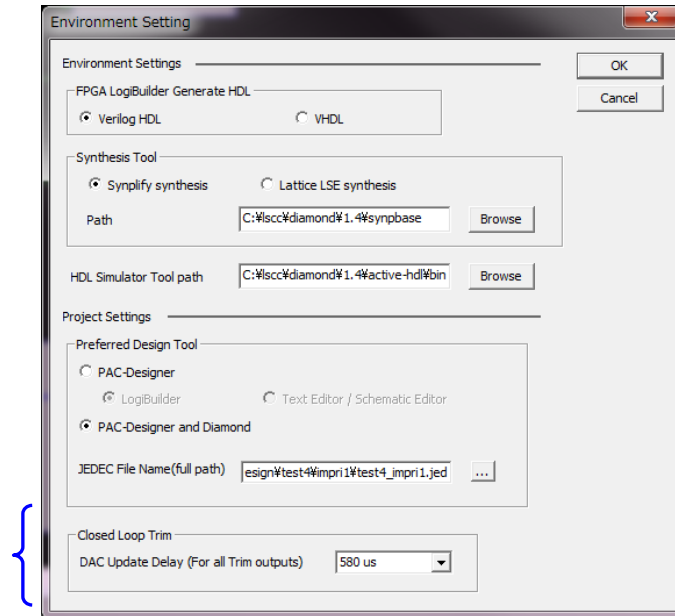


図 2-20 FPGA ロジック開発環境の選択画面

マージン/Trim ブロック(カーソルが下矢印に変わる箇所)をダブルクリックすると、図 2-21 の画面が開きます。この画面上の青点線で囲まれたエリアにて、設定したい Trim*をクリックしますと、図 2-22 (Trim * Configuration)の画面が起動します。

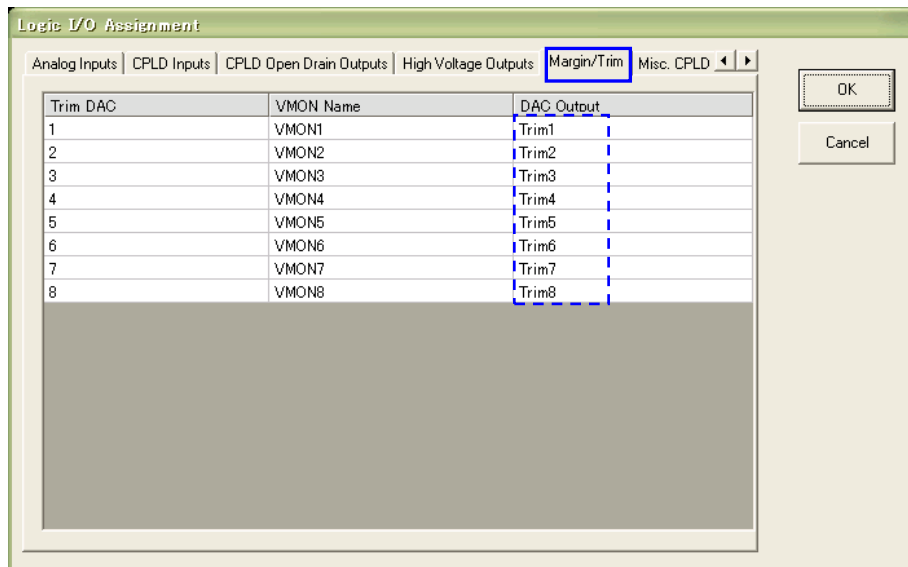


図 2-21 グローバルトリミング設定画面(LPTM10-12107 の場合)

続いて、トリマの設定を行ないます。

トリマは最大でそれぞれ 8 種類 ((LPTM10-12107)、6 種類 ((LPTM10-1247)使用できます。

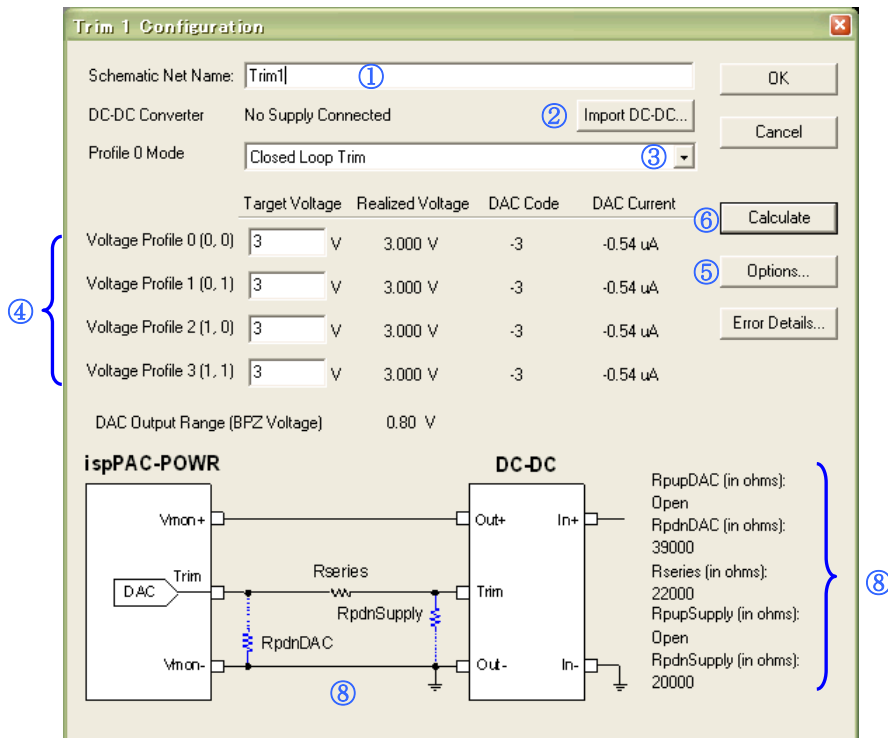


図 2-22 トリマ設定画面

上記画面が開きましたら、下記 8 つの設定を行ないます。

- ① 各ピンの信号名を定義します。Schematic Net Name 出力信号名で使用されます(デフォルトのままでも構いません)。
- ② 2.7 章にて設定した DC/DC コンバータのライブラリを読み出します。
- ③ Profile0 のモードを設定します。
- ④ 各プロファイルについて、ターゲット電圧を設定します。
- ⑤ 必要に応じて、オプションの設定を行ないます(詳細は 2.13 節を参照ください)。
- ⑥ 「Calculate」ボタンをクリックし、抵抗値を計算する。
- ⑦ 計算の結果、エラーがあった場合は、「Error Details」ボタンをクリックするとエラーの詳細を見ることが出来ます(次ページ 図 2-23 参照)。
- ⑧ 実ボード上に実装すべき抵抗値の値と接続イメージが表示されます。接続イメージは、図 2-13 で選択した DC-DC のタイプにより異なります。

図 2-23 に「Error Details」ボタンをクリックしたときに表示される、エラー内容の一例を示します。

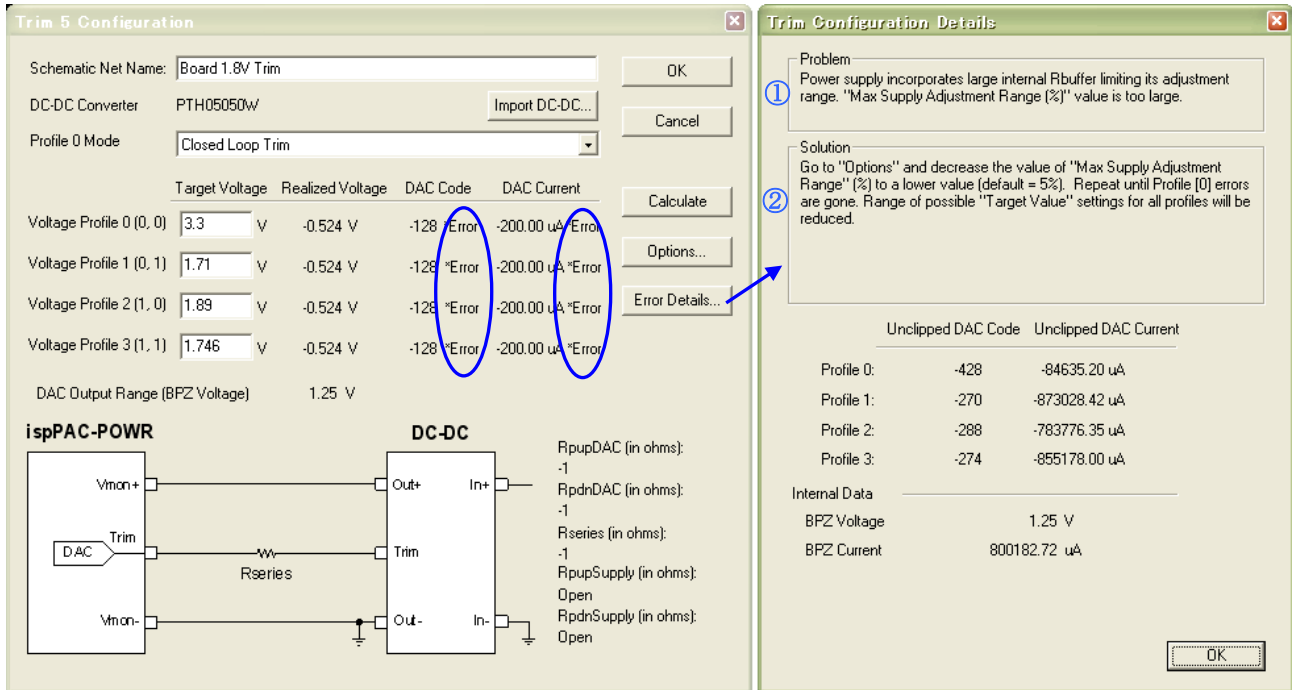


図 2-23 エラー内容の例

上図①に示されるように、エラー内容が「Problem」に表示され、その解決策が②「Solution」に表示されます。

2.14 トリム コンフィグレーションのオプション設定

対象デバイス : LPTM10-1247, LPTM10-12107

図 2-22 の⑤の、「オプション」ボタンをクリックしますと、図 2-24 のような画面が起動します。

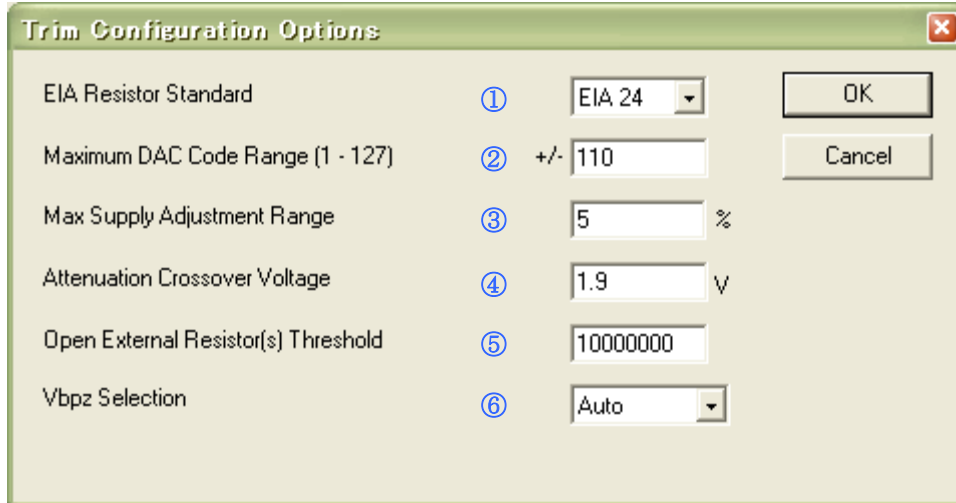


図 2-24 Trim コンフィグレーションのオプション設定画面

上記画面が開きましたら、必要に応じて下記 6 つの設定を行ないます。

- ① 抵抗網の系列を設定します。
- ② DAC コードの範囲を設定します。
1～127 の範囲で選択することが出来ます。デフォルト設定は 110 です。
数字を大きくすることにより、設定可能な電圧範囲は広がります。ただし、デジタル閉ループモード時の過渡応答特性に対するエラーを許容するための値となっているため、**デフォルト設定を推奨します。**
- ③ 出力電圧の範囲を設定します。
設定出力電圧の最大、最小の範囲を設定します。
設定した範囲により、DAC コードのステップサイズが決定されます。
範囲を広くすれば、設定可能な出力電圧の範囲は広がりますが、設定電圧の数値は粗くなります。範囲を狭くすれば、出力電圧の範囲は狭まりますが細かな設定が可能となります。尚、**出力電圧の誤差を±1%以内に抑えるためには、レンジを±5 以内に設定する必要があります。**
- ④ 内部アッテネータが有効となる電圧を設定します。
内蔵の DAC の前段にアッテネータがあり、設定した値を超えるとアッテネータが有効となります。基本的には、デフォルトの 1.9V で問題ありません。
但し、アッテネータが有効になりますとアッテネータの誤差が発生しますので、2.0V 付近の電圧を VMON に入力する場合には、設定を変更することにより精度を良くすることが出来ます。設定可能な範囲は、1.8V～2.1V です。尚、アッテネータによる誤差は最大±1%です。
- ⑤ 外部抵抗をオープンにする場合の閾値を設定します。
計算結果で、設定値以上の値となった場合 Open となります。
- ⑥ バイポーラ・ゼロ・電圧を設定します。Auto 設定の場合、ツールが自動的に最適な値に設定します。**デフォルト設定を推奨します。**

2.15 トリム設定の一覧確認

対象デバイス : LPTM10-1247, LPTM10-12107

トリムブロックの設定が完了しましたら、各トリム出力の設定値の一覧を表示することが出来ます。
トリムブロックの設定画面にて、View→Trim Output Summary を選択します(図 2-25)。

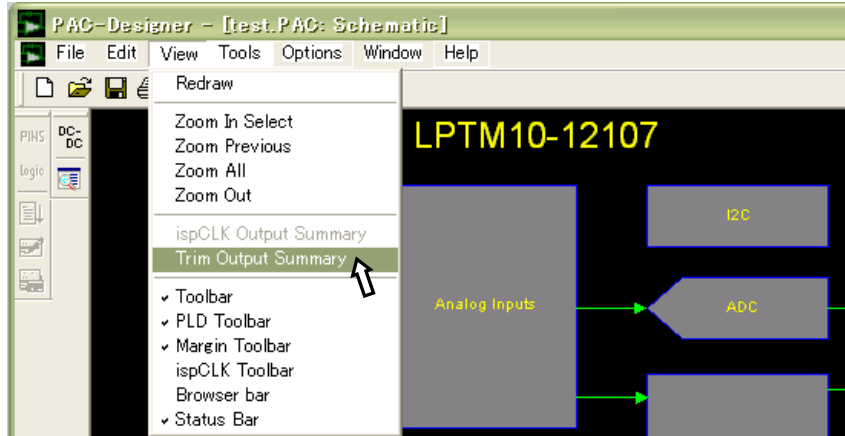


図 2-25 トリム設定の一覧画面起動方法

上記操作にて、図 2-26 の画面が起動し、設定内容を確認することが出来ます。
左右に表示しきれていない場合、下部のスクロールバーで閲覧可能です。

DAC Output	Voltage Profile ...	Profile 0 Voltage	Profile 1 Voltage	Profile 2 Voltage	Profile 3 Voltage	BPZ Voltage
Trim1	Closed Loop	3.000V	3.000V	3.000V	3.000V	1.25V
Trim2	Closed Loop	3.000V	3.000V	3.000V	3.000V	1.25V
Trim3	Closed Loop	3.000V	3.000V	3.000V	3.000V	1.25V
Trim4	Closed Loop	3.000V	3.000V	3.000V	3.000V	1.25V
Trim5	Closed Loop	3.000V	3.000V	3.000V	3.000V	1.25V
Trim6	Closed Loop	3.000V	3.000V	3.000V	3.000V	1.25V
Trim7	Closed Loop	3.000V	3.000V	3.000V	3.000V	1.25V
Trim8	Closed Loop	3.000V	3.000V	3.000V	3.000V	1.25V

図 2-26 トリム出力設定の一覧画面

2.17 I2C コンフィグレーション

対象デバイス : LPTM10-1247, LPTM10-12107

I2C コンフィグレーションブロック(カーソルが□に変わる箇所)をダブルクリックすると、図 2-27 のような画面が開きます。

ここでは、I2C インタフェースを使用するための設定を行ないます(I2C インタフェースを使用しない場合は設定不要です)。I2C インタフェースを使用することで、設定内容のリード・ライトや、ピンの入力や出力状態の制御を行なうことができます。

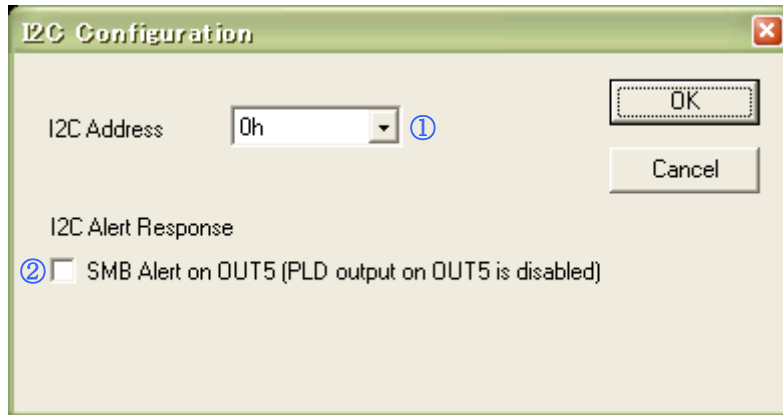


図 2-27 I2C コンフィグレーションウインドウ

上記画面が開きましたら、下記 3 つの設定を行ないます。

- ① I2C インタフェース用のデバイスアドレスをプルダウンから設定します。
デバイスアドレスは、0h 以外を使用してください。また同一 I2C バス上に複数のデバイスを接続する場合は、他のデバイスと違うアドレスに設定してください。
- ② I2C バスのアラート・レスポンス信号を使用する場合、OUT5ピンの出力を利用することができます。チェックボックスにチェックを入れた場合、OUT5 ピンの出力はアラート・レスポンス信号出力 (Active Low) となります。

2.19 CPLD Logic(シーケンス・コントローラ)

各デバイスのメインウインドウにて、CPLD Logic(シーケンス・コントローラ)部をダブルクリックすると、図 2-28 のような画面が起動します。

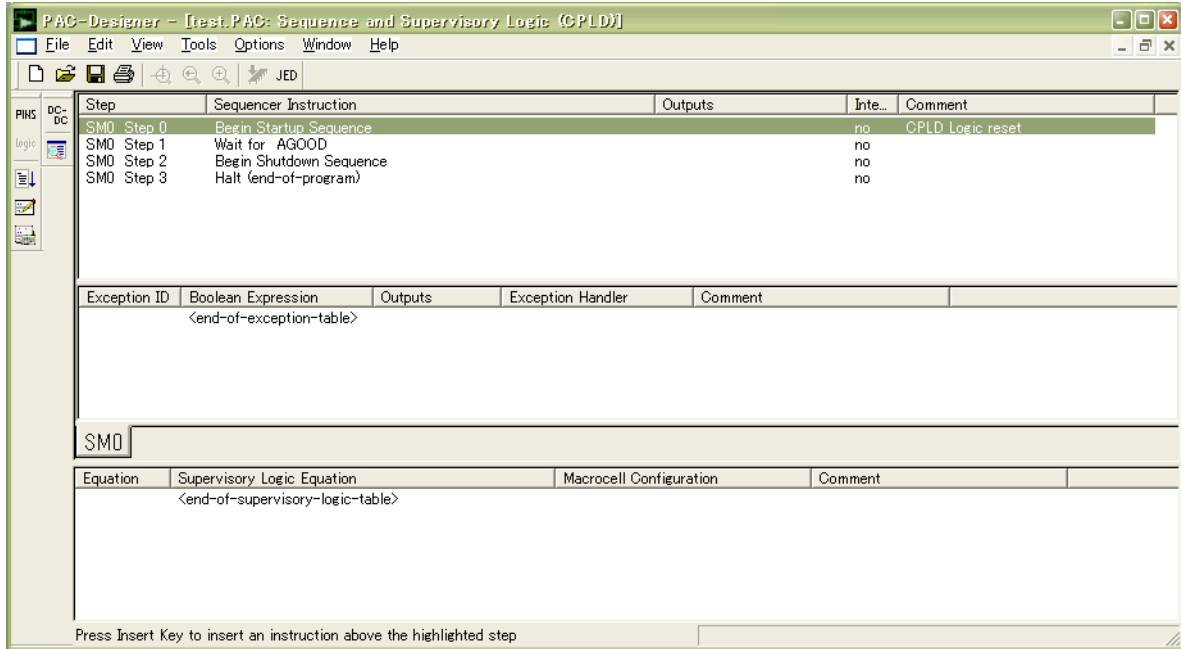


図 2-28 シーケンス・コントローラ画面

本画面にて、CPLD Logic(シーケンス・コントローラ)の設計を行ないます。
本画面にて設計したステップ順に、インストラクションが実行されます。

2.19.1 オプション設定

CPLDLogic(シーケンス・コントローラ)では、オプション設定により、より多くのロジックを PLD に実装することが可能となります。メニューから[Options]→[LogiBuilder Options]を選択しますと、図 2-30 のようなオプション設定画面が起動します。

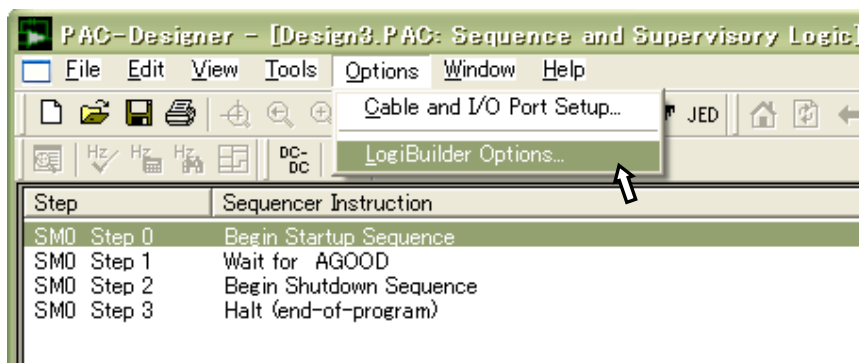


図 2-29 オプション設定画面の起動

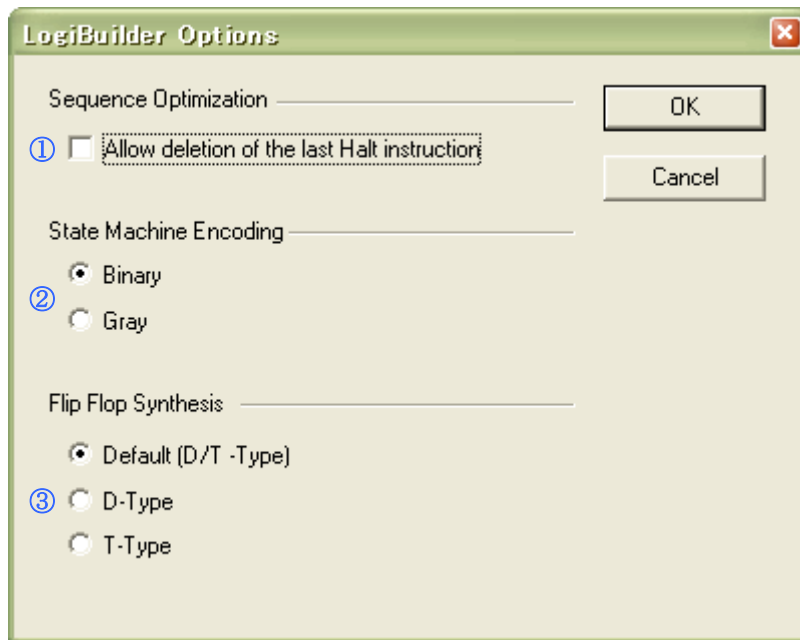


図 2-30 オプション設定画面

上記画面が起動しましたら、下記3つの設定を行ないます。

- ⑦ シーケンス最適化の設定をします。
デフォルトでは、最終行に「Halt」インストラクションが入っています。これは、プログラムが偶発的に最終行に到達した場合に、プログラムを終了させるために用いられています(次のステップが無い場合)。しかしながら、この「Halt」インストラクションは、1Step を使用するため、リソースを必要とします。このオプションをチェックすることにより、最後の「Halt」インストラクションの削除を可能にし、最終行を(end-of-program)タブに置き換えます。そしてどのインストラクション(一般に Go To あるいは if-then-else インストラクション)でも最終行に使用することが出来ます。
- ⑧ ステートマシンのエンコーディング方法を選択します。
デフォルト設定は、Binary エンコードです。デザインによっては、Gray エンコードを用いることで、使用するリソースを少なくすることが出来ます。
- ⑨ フリップフロップ合成設定
通常はデフォルト設定でご利用下さい。

2. 19. 3 Pin の定義

「PINS」のアイコンをクリックすると、ピン定義のリストが表示されます(図 2-31)。ここでも、入力と出力の状態について定義することが出来ます。

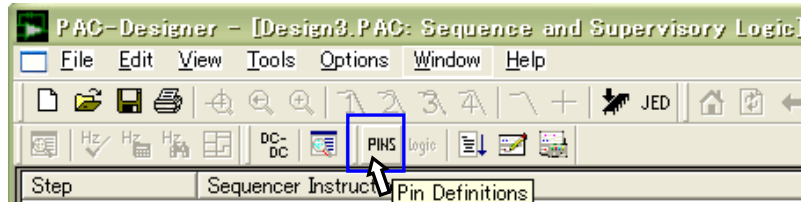


図 2-31 PIN 定義画面の起動

2.2, 2.3, 2.4, 2.5 節で行った入出力ピンの設定画面が開きます(図 2-32)、設定を行うタブを選び必要な設定を行います。

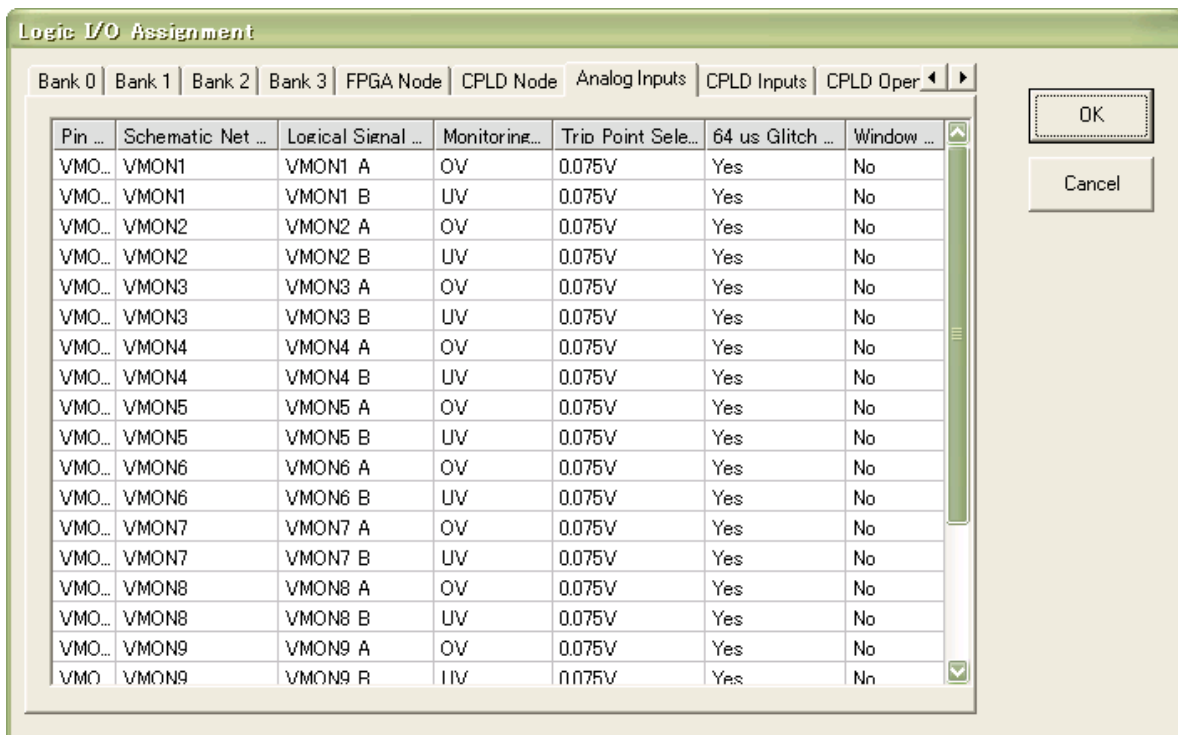


図 2-32 ピンの定義編集画面

2. 19. 5 インストラクションの設定方法

29 ページの、図 2-28 のような CPLD Logic(シーケンス・コントローラ)画面にてインストラクションの追加を行なうには、挿入前のシーケンスをハイライトし、メニューの Edit→Insert Instruction をクリック(またはキーボードの Insert キーを押します(図 2-33))。図 2-33 の例では、ハイライトされた Step2 と Step1 の間に選択したシーケンスが挿入されます。

インストラクションの削除を行なうには、削除したいシーケンスをハイライトし、メニューの Edit→Delete Instruction をクリックします。

ステートマシンを複数組実装させることが可能です。複数のステートマシンを実装させる場合は、メニューの Edit→Multiple State Machines をクリックし、State Machine Name の項で適当な名前を付けて保存します。

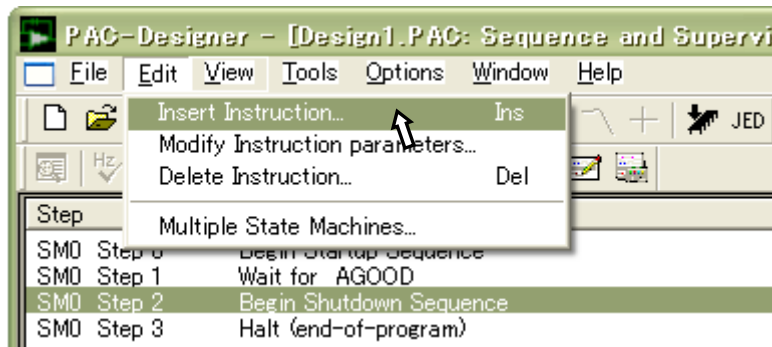


図 2-33 インストラクションの追加

その後、追加するインストラクションの選択画面が起動します(図 2-34)。

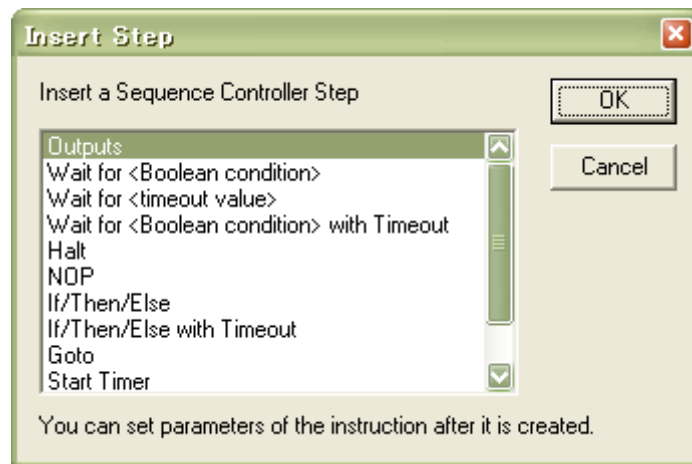


図 2-34 追加するインストラクションの選択画面

上記画面にて追加したいインストラクションをハイライトし、OK ボタンをクリックしますとインストラクションが追加されます。各インストラクションの内容について、次ページから説明いたします。また、各インストラクションの設定方法について 0~0 にて説明いたします。

Output: 出力をセットもしくはリセットします。
→ 電源へのイネーブル制御や、CPUリセット等に使用します。

Wait for <Boolean Condition>: 定義したブーリアン条件が成立するまで待機します。
→ 電源の安定や FPGA のコンフィギュレーションの完了まで待機する等に使用します。

Wait for AGOOD: デフォルトで “Wait for AGOOD” というインストラクションが入っています。この AGOOD 信号は、パワーオンリセット後に、VMON 入力のコンパレータの自動キャリブレーション、デジタル・グリッチフィルタの初期化が完了したことを示します。**VMON 入力を使用する全てのロジックは、AGOOD 信号がアクティブになるまで必ず待機する必要があります。**
誤って、“Wait for AGOOD” インストラクションを削除してしまった場合は、Wait for インストラクションの追加により復旧できます。

Wait for <Timeout Value>: タイマーが満了するまで、このステップで待機します。
→リセット信号の引き延ばしや、シーケンスの遅延制御等に使用します。

Wait for <Boolean Condition> With Timeout: 定義したブーリアン条件が成立するか、タイマー満了までこのステップで待機します。その後イベントタイプに応じて他のステップにジャンプします。
→ 規定時間内に監視電源の立ち上がりを待つ為や、パワーダウン中にプロセッサからのレスポンス待機等に使用します。

Halt: 無条件にこのステップで動作停止
→ 電源立ち上げ時のシーケンス完了後に、例外処理が発生するまでシーケンス制御を停止させる場合や、完全にシーケンス制御が完了して動作を停止させる場合に使用します。

Halt (end-of-program): プログラムの終了を示します。
→ プログラムが偶発的に最終行に到達した場合に、プログラムを終了させるために用いられています(次のステップが無い場合)。このインストラクションは、オプション設定により削除が可能です(2.19.1 参照)。

NOP: ノーオペレーション
→ タイマー動作を連続したステップで使用できないため、タイマー動作前に 1 ステップジャンプするために使用します。

If / Then / Else: ブーリアン条件をチェックし、結果により指定したステップにジャンプします。
→ 条件付きのプログラム・フロー制御指示

If / Then / Else With Timeout: ブーリアン条件のチェック結果とタイマーの満了に基づいて3つのジャンプを持つインストラクションです。このインストラクションを使用するためには、予め “Start Timer” インストラクションにてタイマーをスタートさせておく必要があります。
→ 規定時間内に、電源が立ち上がるまで待機する場合等に使用します。

Go To <step x>: 指定したステップにジャンプします。
→ 無条件でステップをジャンプする場合に使用されます。

Start Timer: 指定したタイマーをスタートします。✖
→ このインストラクションは、“If / Then / Else With Timeout” インストラクションと共に使用されます。

Stop Timer: 指定したタイマーをストップします。✖
→ 指定したタイマーを強制終了するために使用されます。

要確認!

✖ Stop Timer/Stop Timer インストラクションを使用する場合、Timer Gate のピンタイプを JK にする必要があります(設定方法は 3.1.2 節参照)。

上記インストラクションに加えて、シーケンス・コントローラには以下の2つの擬似インストラクションがデフォルトで追加されています。

Begin Startup Sequence: スタートアップ・シーケンスの開始を明示します。

→このシーケンスは削除可能で有り、削除することによりステップ数(リソース使用率)を削減することが可能です。ただし、**最初のインストラクションを Wait for Timeout Value にすることは出来ません**ので、その場合は **Wait for AGOOD** にするか、**Begin Startup Sequence** を残すかにしてください。

重要: Ver5.1 以降の PAC-Designer では本命例は割り込み不可 (Interruptible 項が “no”) としております。

Begin Shutdown Sequence: シャットダウン・シーケンスの開始を明示します。

→このインストラクション以降に追加させるインストラクションは、割り込み許可設定のデフォルト値がディセーブルとなります(このインストラクションは削除可能です)。

コラム : 電源投入直後の動作と “Begin Startup Sequence” “Begin Shutdown Sequence” について

シーケンス設計のために PAC-Designer を開始すると、“Begin Startup Sequence”、“Wait for AGOOD”、“Begin Shutdown Sequence”、“Halt”などの行がデフォルトで入っています。これは設計者が所望の処理のために命令を追加・記述編集する際に、どこに挿入していけば良いかが容易に判るためというのが元々の目的の一つです。

このうち必須な命令が “Wait for AGOOD”、殆どの場合必要なものが “Halt” ですが、“Begin Startup Sequence”と “Begin Shutdown Sequence”は本来なくても構いません(但し最初のインストラクションを Wait for Timeout Value にすることは出来ません)。AGOOD はパワーオンリセット後に、VMON 入力コンパレータの自動キャリブレーション、デジタル・グリッチフィルタの初期化が完了したことを示す信号です。VMON 入力を使用する全てのロジックは、AGOOD 信号がアクティブになるまで必ず待機する必要があります。従って、ユーザの全ての命令はこの後に挿入することを期待しています。

“Begin Startup Sequence” 行はコンパイル後には NOP として扱われますが、PAC-Designer のバージョンによって、割り込みを許容するかしないか (Interruptible 項が ver4.9.9a までは “yes”、ver5.1 以降は “no”) 異なっており、これを変更することはできません。Ver4.9.9a 以前のツールをご使用の場合は以下の注意が必要です。

仮に例外処理を開始するための条件として “監視対象の電源系統が所定の電圧値以下になった場合” ((NOT VMON1) OR (NOT VMON2) 等)を考えます。当然のことながら、ボードレベルで電源が投入された直後はこの条件が成立しますので、AGOOD のアサート前に、例外処理が開始されることとなります。即ちデバイス内部の初期化処理が完了しないにもかかわらず、例外処理を実行するという状況に陥ります。

例外処理の条件としてこのように記述・設計することは、必ずしも特異とは言えません。従ってそうせざるを得ないケースでは、“Begin Startup Sequence” の行を削除することがより妥当です。

2. 19. 6 インストラクションの設定方法(Output)

Output インストラクションを選択した場合、図 2-35 のように<no outputs specified>と表示されます。

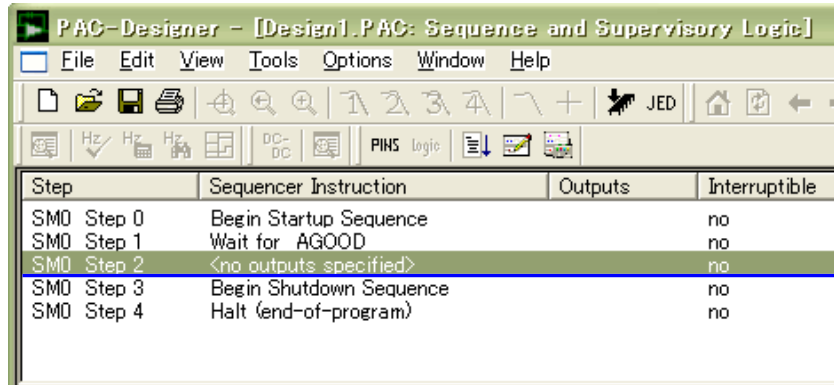


図 2-35 Output インストラクション追加後の画面

上記画面にて< no outputs specified >の行をダブルクリックしますと、図 2-36 のような出力信号の状態を編集する画面が起動します。

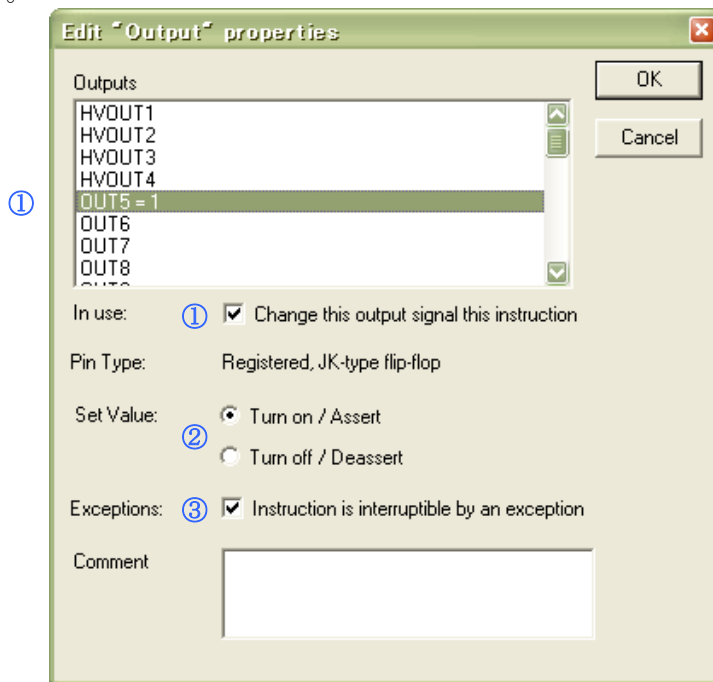


図 2-36 Output インストラクションの設定画面

上記画面が起動しましたら、下記3つの設定を行ないます。

- ① 出力状態を変更したい信号を選択し、「Change this output signal this Instruction」にチェックをつける。
- ② High レベルを出力するか、Low レベルを出力するか選択する。
- ③ 割り込み可能か否かを指定する(チェックを入れると割り込み可能)。

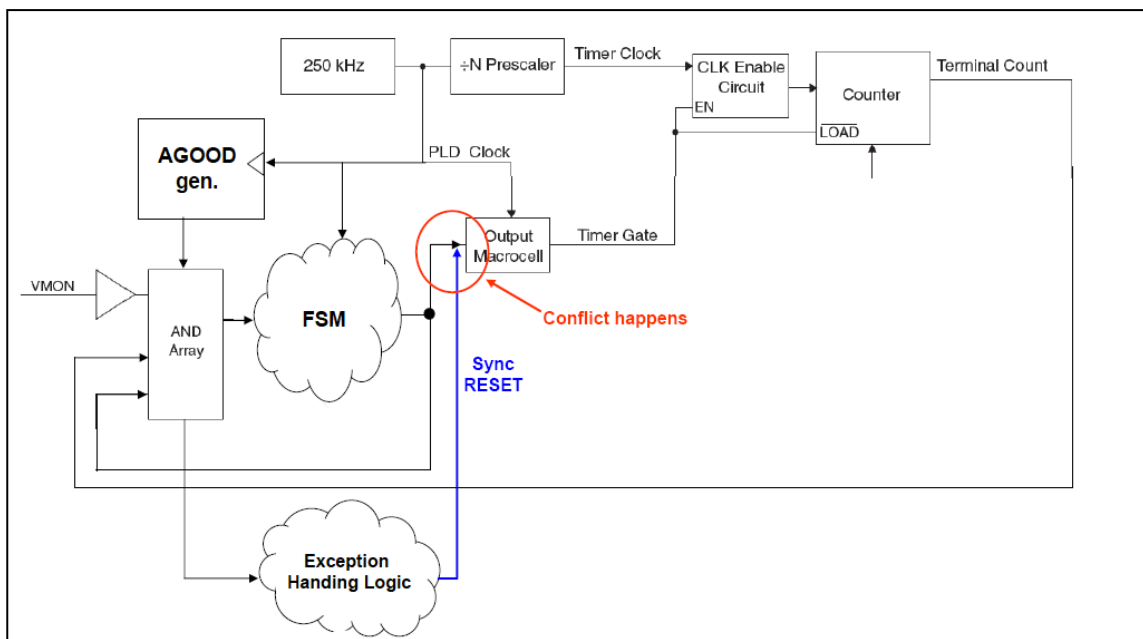
重要：図 2-36 の “Outputs” 候補としてタイマーを強制リセットする信号 `TIMER<n>_GATE` (<n>: 1~4) が表示されますが、これは使用しないで下さい。タイマーの開始・停止処理は “Start Timer” や “Stop Timer” で実現します。特に例外処理で本信号を操作すると予期しない動作をする可能性があります。

コラム： 例外処理と `TIMERn_GATE` 信号の操作について

`TIMERn_GATE` という信号は下図の様に、内蔵タイマーのクロック・イネーブルとタイマーカウンタ値の `LOADn` 制御を行うためのマクロセル出力です。

元来本出力は正規シーケンス内で操作し、シーケンサ本体のステートによってマクロセル出力を生成することを期待しています。仮に例外処理 (Exception) 内にて本信号の値を操作すると、シーケンサ本体からの制御と競合する (不一致になる、或いは意図しないタイミングでリセット/ロード・トリガされる) 場合が生じ得ます。そうするとクロック・イネーブルと値のロードが正常に (意図した通りに) 行われなくなり、従ってタイマー動作が予測不能となり、結果としてシーケンサ入力の “AND Array” で正しい状態遷移のための論理が構成できないこととなります。すなわち誤動作する (期待動作と異なる、或いは想定していない状態遷移を引き起こす) 可能性が大きくなります。

以上より特に例外処理内での `TIMERn_GATE` 信号の直接操作を行うべきではないと言えます。実際にタイマーを制御する場合、Start Timer / Stop Timer という命令が用意されていますので、その開始と停止はこれを用いることによって実現できます。例外処理内であれ、正規シーケンスであれ、基本的に `TIMERn_GATE` 信号は直接操作しないことを強く推奨します。



2. 19. 7 インストラクションの設定方法(Wait for Boolean Condition)

Wait for Boolean Condition インストラクションを選択した場合、図 2-37 のように Wait for <booleanExpr>と表示されます。

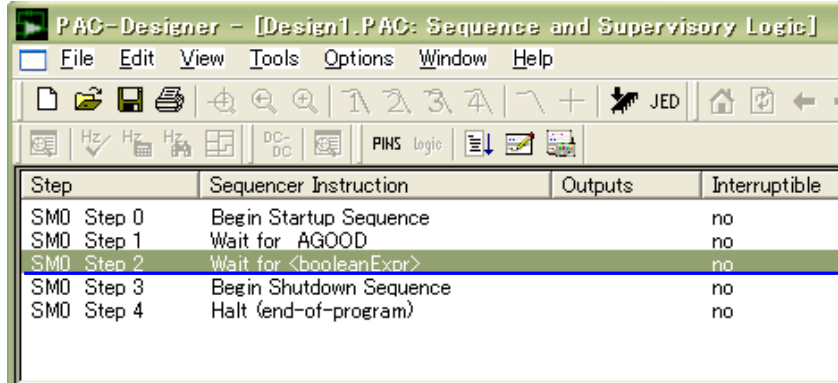


図 2-37 Wait for Boolean Condition インストラクション追加後の画面

上記画面にて Wait for <booleanExpr>の行をダブルクリックしますと、図 2-39 のような画面が起動します。

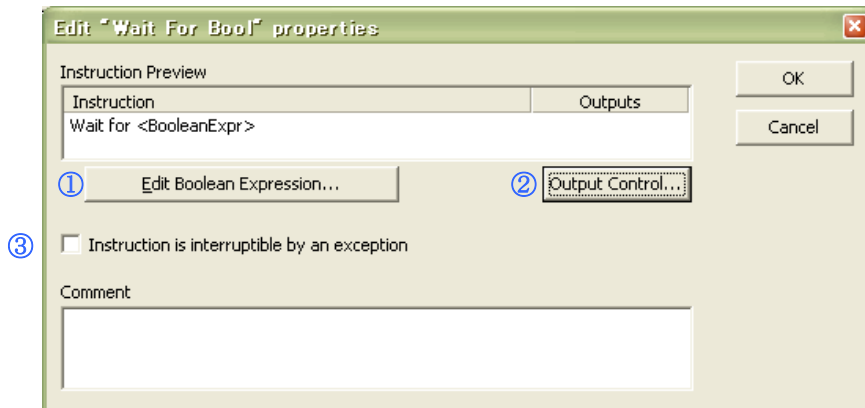


図 2-38 Wait for Boolean Condition インストラクションの設定画面 1

上記画面が起動したら、下記3つの設定を行ないます。

- ① “Edit Boolean Expression”をクリックすると、次ページの図 3-16 画面が起動し、論理式の編集ができます(詳細は次ページ)。
- ② 必要に応じ、“Output Control”をクリックすると、前ページの図 3-13 画面が起動し、①で編集する論理式に連動した出力が設定可能です。
- ③ 割り込みを可能にする場合は、「Instruction is interruptible by an exception」をチェックします。

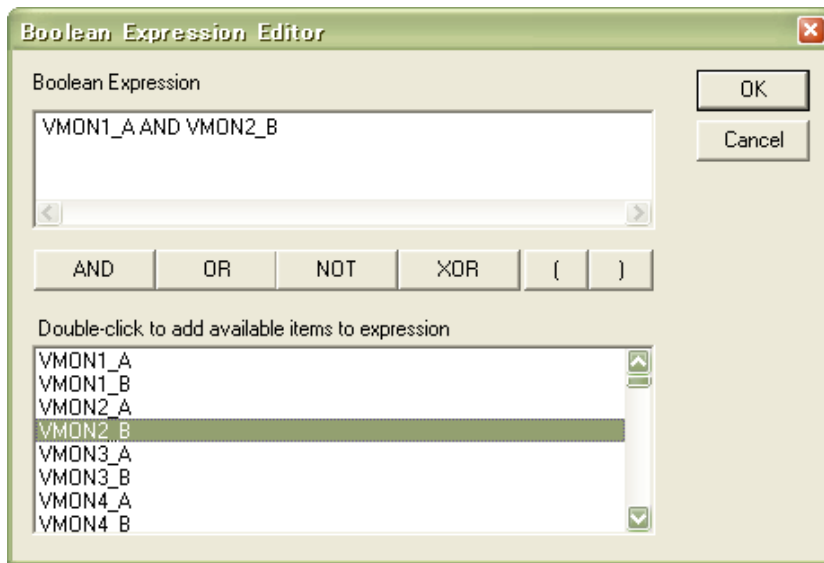


図 2-39 Wait for Boolean Condition インストラクションの設定画面 2

上記画面にて信号名や AND、OR と言った論理演算子をダブルクリックして追加し、論理式を編集します。例えば "VMON1_A" をダブルクリックし、"AND" をクリックした後、"VMON2_B" をダブルクリックすると、Boolean Expression = VMON1_A AND VMON1_B となります。編集後 OK をクリックし、図 3-38 画面へ戻ります。

2. 19. 8 インストラクションの設定方法(Wait for Timeout Value)

Wait for Timeout Value インストラクションを選択した場合、図 2-40 のように Timer1 を使用した場合のインストラクションが追加されます。

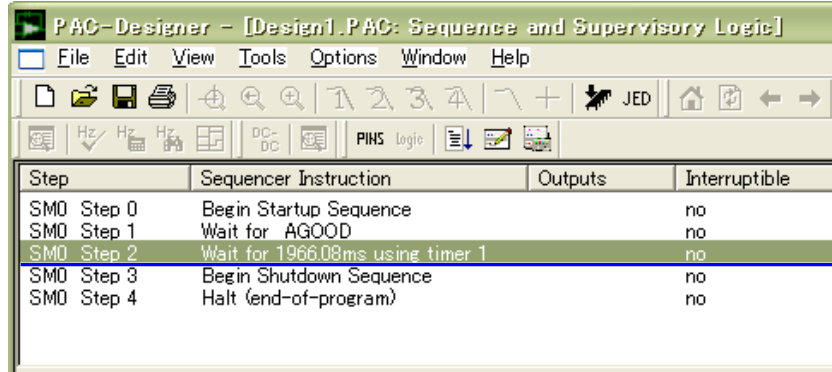


図 2-40 Wait for Timeout Value インストラクション追加後の画面

上記画面にて Wait for ***ms using timer 1 の行をダブルクリックしますと、図 2-41 のような画面が起動します。

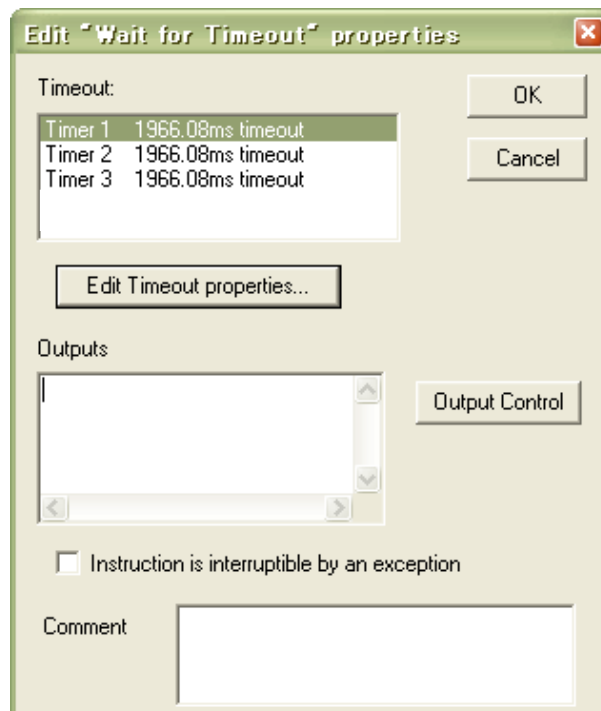


図 2-41 Wait for Timeout Value インストラクションの設定画面

上記画面にて使用するタイマーを選択します。また、「Edit Timeout properties」をクリックしますと、タイマーの設定画面が起動します。タイマーの設定方法については、2.6 節タイマー&クロックを参照ください。

割り込みを可能にする場合は、「Instruction is interruptible by an exception」をチェックします。

2. 19. 9 インストラクションの設定方法(Boolean Condition with Timeout)

Boolean Condition with Timeout インストラクションを選択した場合、図 2-40 のように Timer1 を使用した場合のインストラクションが追加されます。

Step	Sequencer Instruction	Outputs	Interruptible
SMD Step 0	Begin Startup Sequence		no
SMD Step 1	Wait for AGOOD		no
SMD Step 2	Wait for <booleanExpr> or 1966.08... If Timeout Then Goto 0		no
SMD Step 3	Begin Shutdown Sequence		no
SMD Step 4	Halt (end-of-program)		no

図 2-42 Boolean Condition with Timeout インストラクション追加後の画面

上記画面にて Wait for <BooleanExpr> or ***ms using timer 1 の行をダブルクリックしますと、図 2-43 のような画面が起動します。

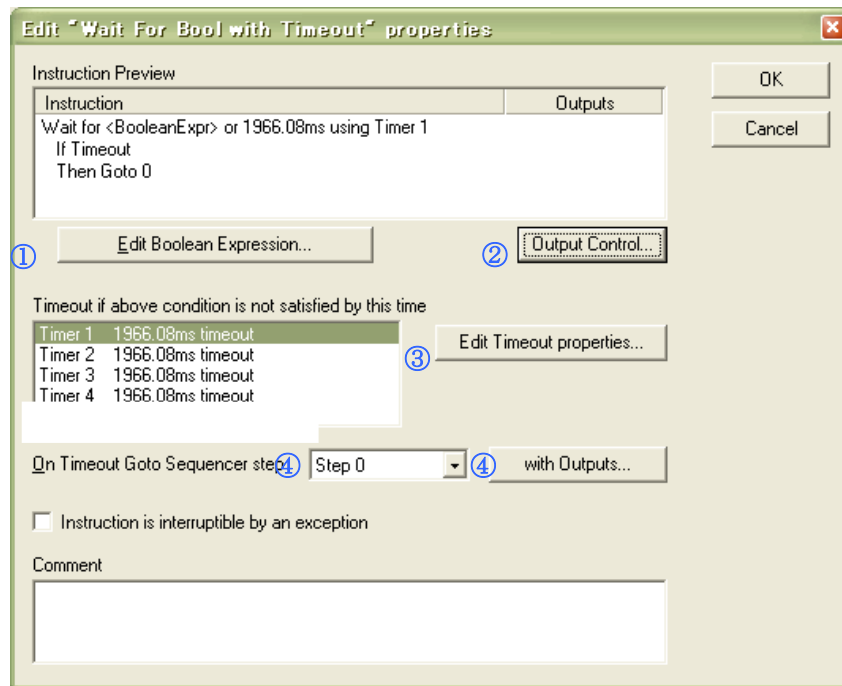


図 2-43 Boolean Condition with Timeout インストラクションの設定画面

上記画面が開きましたら、下記 5 つの設定を行ないます。

- ① Edit ボタンを押し、ブーリアン形式の論理式の設定を行なう(設定方法は、0 節と同様)。
- ② 出力設定の選択を行なう(設定方法は 3.1.4 節と同様)。
- ③ タイマーの選択を行なう(設定方法は、0 節と同様)。
- ④ タイムアウトした場合の、どのステップに移行するかを設定する(同時に出力設定も可能)。
- ⑤ 割り込み可能か否かを指定する。

2. 19. 10 インストラクションの設定方法(If/Then/Else)

If / Then / Else インストラクションを選択した場合、図 2-44 のように If <booleanExpr> Then Goto 0 Else Goto 0 と表示されます。

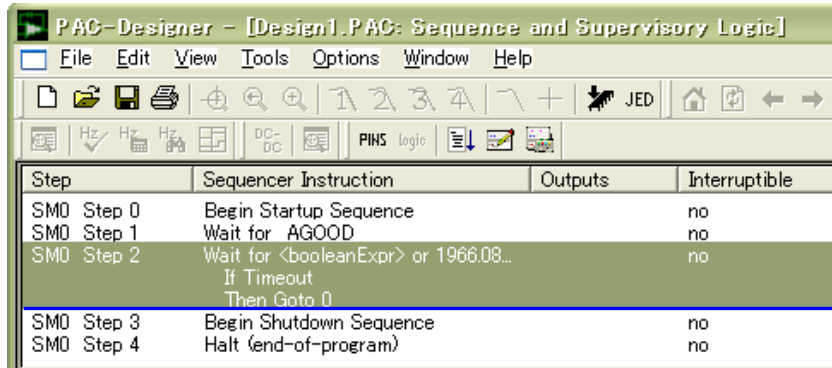


図 2-44 If/Then/Else インストラクション追加後の画面

上記画面にて If <booleanExpr> Then Goto 0 Else Goto 0 の行をダブルクリックしますと、図 2-45 のような画面が起動します。

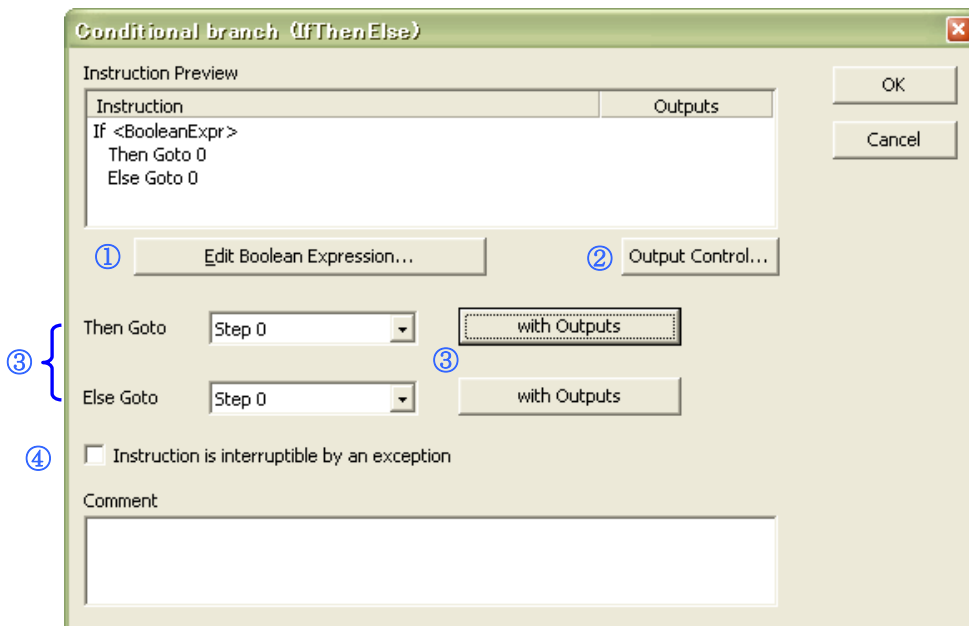


図 2-45 If/Then/Else インストラクションの設定画面

上記画面が開きましたら、下記 4 つの設定を行ないます。

- ① Edit ボタンを押し、ブーリアン形式の論理式の設定を行なう(設定方法は、0 節と同様)。
- ② 出力設定の選択を行なう(設定方法は 3.1.4 節と同様)。
- ③ ①の条件が成立した場合(Then)と成立しない場合(Else)の、ステップの移動先を設定する(同時に出力設定も可能)。
- ④ 割り込み可能か否かを指定する。

2. 19. 11 インストラクションの設定方法(If / Then / Else with Timeout)

If / Then / Else with Timeout インストラクションを選択した場合、図 2-46 のように If <booleanExpr> Then Goto 0 Else If Timer1 Then Goto 0 と表示されます。

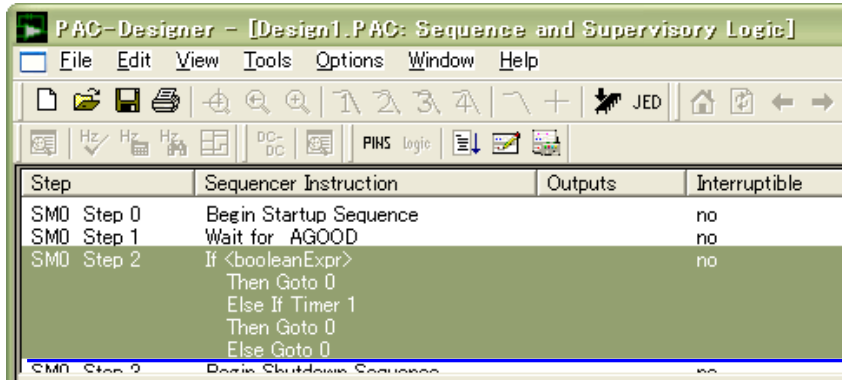


図 2-46 If / Then / Else with Timeout インストラクション追加後の画面

上記画面にて If <booleanExpr> Then Goto 0 Else If Timer1 Then Goto 0 の行をダブルクリックしますと、図 2-47 のような画面が起動します。①～⑤については同時に出力設定も可能です。

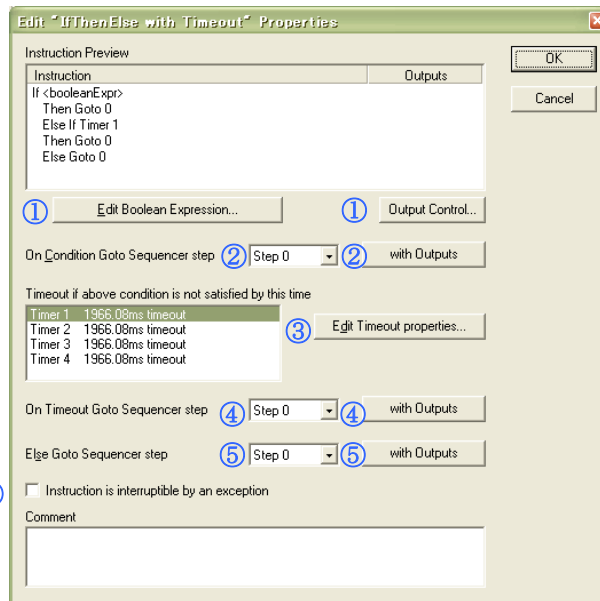


図 2-47 If/Then/Else with Timeout インストラクションの設定画面

上記画面が開きましたら、下記 6 つの設定を行ないます。

- ① Edit ボタンを押し、ブーリアン形式の論理式を設定する(設定方法は、0 節と同様)。
- ② ①の条件が成立した場合(Then)の、ステップの移動先を設定する。
- ③ タイマーの選択を行なう(設定方法は、0 節と同様)。
- ④ 選択したタイマーが満了している場合の、ステップの移動先を設定する。
- ⑤ 選択したタイマーが満了していない場合の、ステップの移動先を設定する。
- ⑥ 割り込み可能か否かを指定する。

・本インストラクションでは、タイマーは開始しませんので Start Timer インストラクションにて Timer をスタートさせておく必要があります。

2. 19. 12 インストラクションの設定方法(Goto)

Goto インストラクションを選択した場合、Go to Step0 と表示されますので、Go to Step0 の行をダブルクリックしますと、図 2-48 のような画面が起動します(下記は Outputs で 2 系統の出力制御を設定済みの画面例です)。

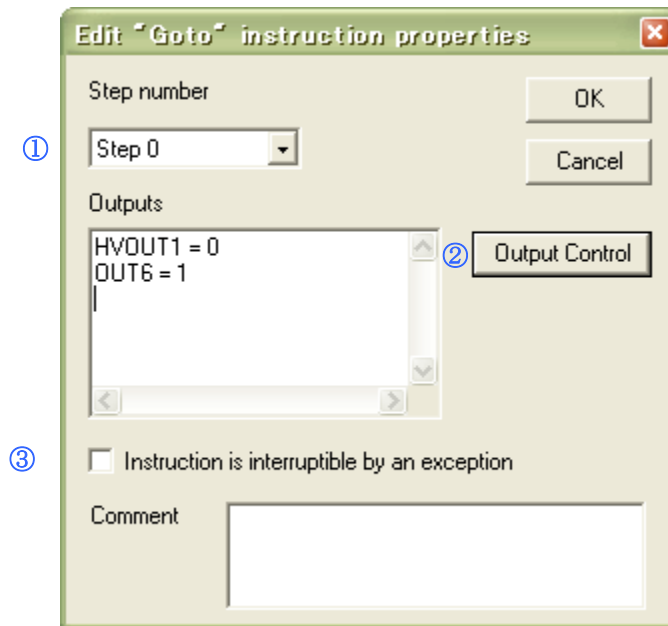


図 2-48 Goto インストラクションの設定画面

上記画面が開きましたら、下記 3 つの設定を行ないます。

- ① 移動先のステップを選択します。
- ② 必要に応じて、出力設定を行なう。
- ③ 割り込み可能か否かを指定する。

2. 19. 13 インストラクションの設定方法(Start Timer/Stop Timer)

Start Timer もしくは Stop Timer インストラクションを選択した場合、Start Timer 1(**ms)もしくは、Stop/Reset Timer1 と表示されますので、この行をダブルクリックしますと、図 2-49 のような画面が起動します。

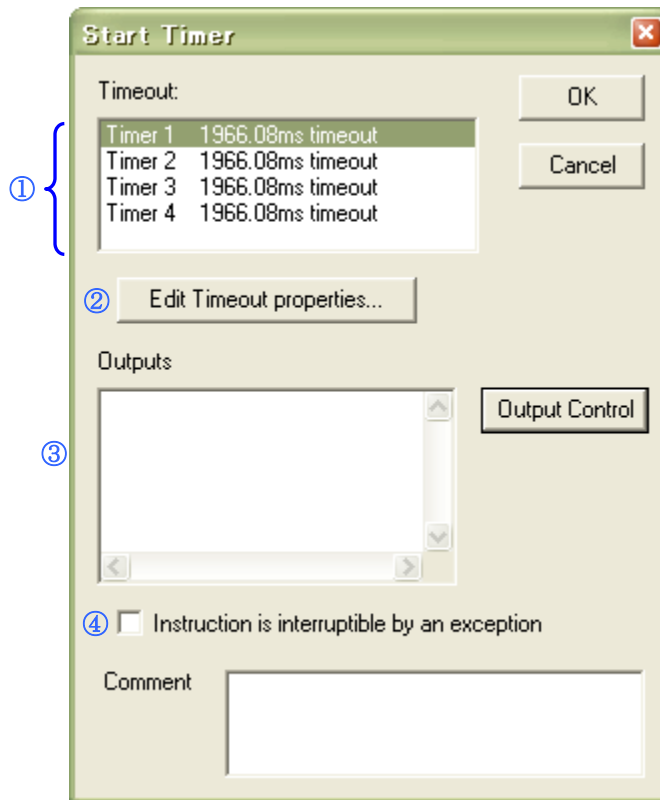


図 2-49 Start Timer/Stop Timer インストラクションの設定画面

上記画面が開きましたら、下記 4 つの設定を行ないます。

- ① 使用するタイマーを選択する。
- ② この画面からも必要に応じてタイマーの設定を変更可能。図 2-8 の右側と同様の画面が開き、0 節に記載の方法と同様にタイマーの設定を行なうことが可能。
- ③ 必要に応じて、出力設定を行なう。
- ④ 割り込み可能か否かを指定する。

Start Timer および Stop Timer は、If / Then / Else with Timeout インストラクションや Wait for Boolean インストラクションと組み合わせて使用します。タイマーがリセット時および満了前は内部の Timer<n>_TC 信号が”0”となり、タイマーが満了すると”1”となります。

•Start Timer / Stop Timer インストラクションを使用する場合、Timer Gate のピンタイプを JK にする必要があります。

2. 19. 14 インストラクションの設定方法(例外処理)

例外処理を追加するには、シーケンスの下の枠に表示されている<end-of-exception-table>の行をダブルクリックします。

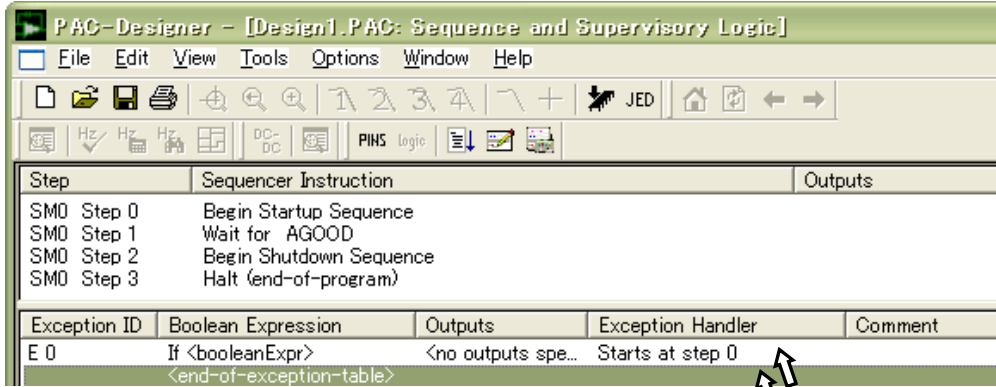


図 2-50 例外処理の設定の起動

<end-of-exception-table>の行をダブルクリックしますと、図 2-50 のように if <booleanExpr> という 1 行が追加されますので、この行をダブルクリックして、図 2-51 のような設定画面を起動します。

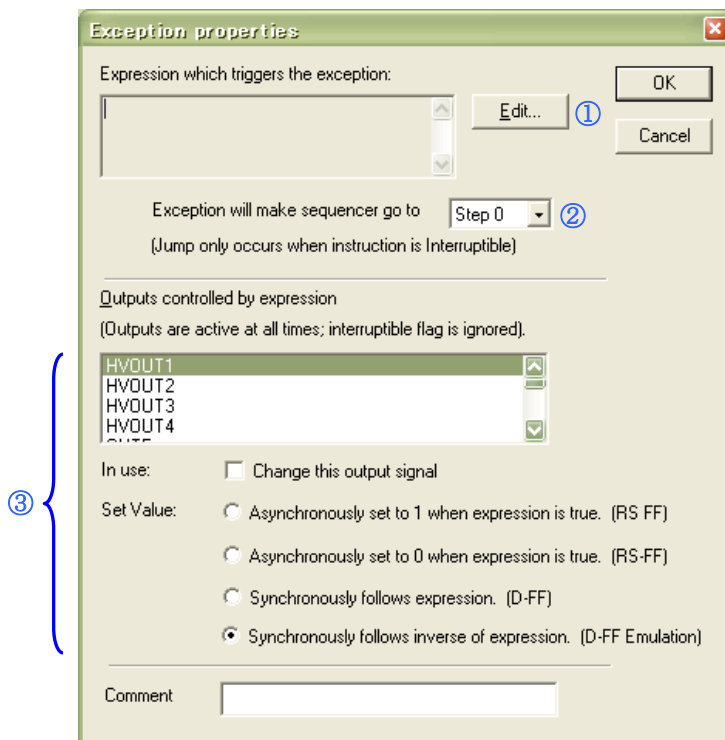


図 2-51 例外処理の設定画面

上記画面が開きましたら、下記3つの設定を行ないます。

- ① Edit ボタンを押し、ブーリアン形式の論理式の設定を行なう(設定方法は、0 節と同様)
- ② ①の条件が成立した場合(Then)の、ステップの移動先を設定する。
- ③ 出力信号の状態を変化させる場合は、出力信号を選択し、変化後の出力状態を設定します。

2. 19. 15 監視ロジック出力回路の生成(Supervisory Logic Equation)

監視ロジック出力回路(ステップ実行とは関係なく常時監視)を追加するには、シーケンス・コントローラの下の子に表示されている<end-of-supervisory-logic-table>の行をダブルクリックします。

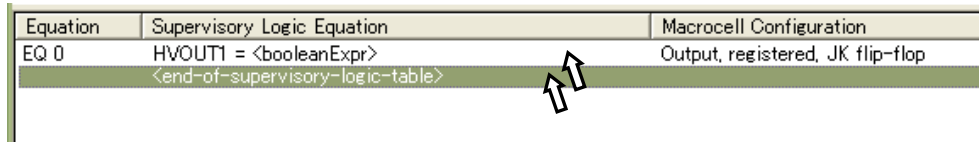


図 2-52 監視ロジック出力回路設定画面の起動

<end-of-supervisory-logic-table>の行をダブルクリックしますと、図 2-53 のように HVOUT= <booleanExpr> という 1 行が追加されますので、この行をダブルクリックして、図 2-54 のような設定画面を起動します。

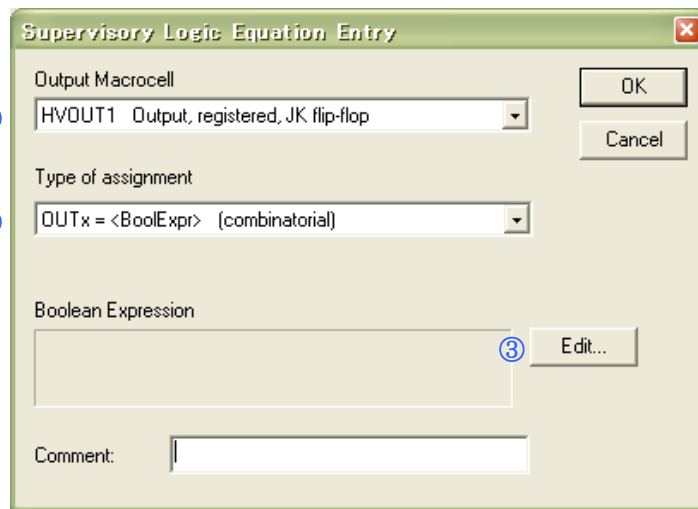


図 2-53 監視ロジック出力回路の設定

上記画面が開きましたら、下記3つの設定を行ないます。

- ① 組み合わせ論理を出力するピンを指定します。
- ② アサインのタイプを選択します。
- ③ Edit ボタンを押し、ブーリアン形式の論理式の設定を行なう(設定方法は、0 節と同様)。

出力ピンのタイプと、アサインタイプの組み合わせには制限があります。表 2-1 にて可となっている組み合わせのみ可能です。必要に応じて、ピンタイプの定義を変更してください。

表 2-1 ピン定義とアサインタイプ

アサインタイプ	ピン定義			その他注意点
	Pin=JK FF	Pin=D-type FF	Pin=Combinatorial	
Combinatorial	不可	不可	可	-
D	不可	可	不可	-
.ap (async set to 1)	可	可	不可	Active High
.ar (async reset to 0)	可	可	不可	Active High

2.20 FPGA Logic (FPGA ロジック)

FPGA ロジック部の設計方法は、2.1 節で選択した設計手法により異なります。3

2.20.1 PAC-Designer - LogiBuilder フロー

この手法では、HDL や回路図入力を必要としません。FPGA 部の設計も、2.2 節で CPLD Logic 部の設計と同様に、LogiBuilder を使用します。

各デバイスのメインウインドウにて、FPGA Logic (FPGA ロジック) 部をダブルクリックすると、図 2-54 の LogiBuilder 画面が起動します。LogiBuilder を用いた設計手法については、2.2 節を参照下さい。

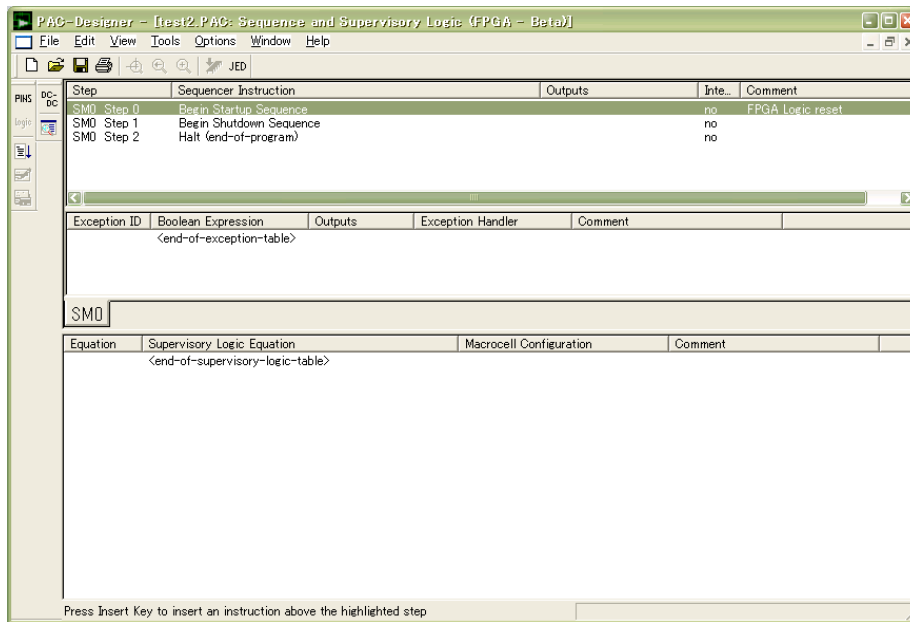


図 2-54 LogiBuilder 画面

2. 20. 2 PAC-Designer - Text Editor / Schematic Editor フロー

HDLや回路図によりデザインをエントリします。設計時のUIはPAC-Designerを使用しますが、バックグラウンドで論理合成ツールや回路図エディタを使用します。その為、PAC-Designer の他に Lattice 社開発ツール DAIAMOND のインストールが必要です。DAIAMOND のインストール、及び詳しい操作方法については、下記の DAIAMOND 日本語マニュアルをご参照下さい。

弊社 HP より、ドキュメント=>日本語ドキュメント=>ソフトウェアマニュアルより、DAIAMOND の項もしくは、直接下記のアドレスを参照下さい。

<http://www.latticesemi.co.jp/docs/toolmanuals/diamond/index.cfm>

各デバイスのメインウィンドウにて、FPGA Logic (FPGA ロジック) 部をダブルクリックすると、図 2-54 の File List Manager 画面が起動します。

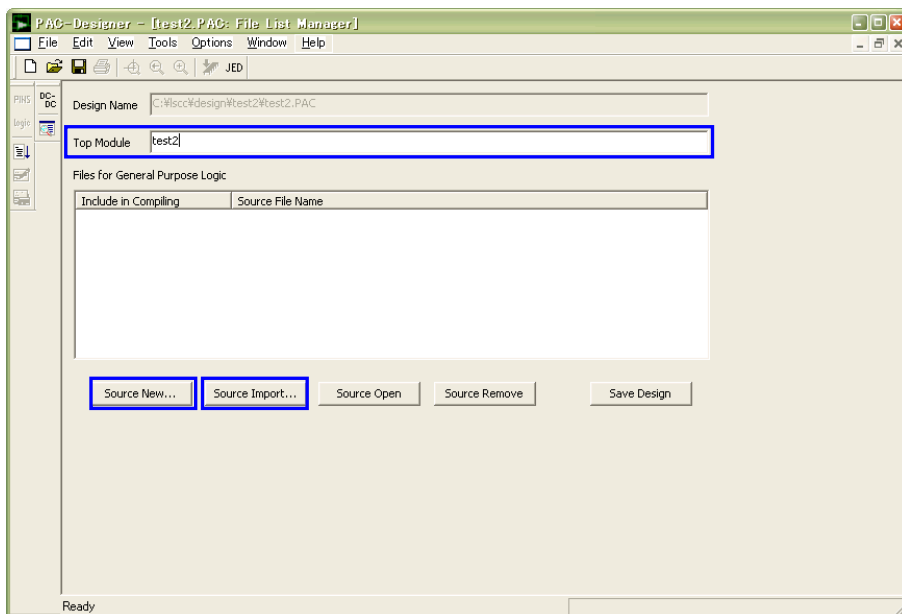


図 2-55 File List Manager 画面

新規にデザインを作成する場合は“Source New”を選択し、エントリの種類を選択しデザインを作成します。

別途作成済の HDL や、DAIAMOND で作成済の回路図(拡張子 sch)を取り込む場合は、“Source Import”を選択し、エントリの種類を選択しデザインを取り込みます。Top Module には最上位階層となるモジュール名を設定します。

このフローの場合、下記何れかの方法で FPGA Logic 部のピンアサインを設定します。

方法1. 制約ファイル“lpf”での設定方法

制約ファイル“lpf”を直接編集してピンアサインを設定します。“lpf”はプロジェクトフォルダ => project_name_FPGAFiles => “Top Module Name(図 2-55 で指定)”.lpf としてテキストファイルで準備されています。lpf には下記フォーマットでピン配置を行います。

```
LOCATE COMP “ピン名” SITE “ピン番号”;
```

例

```
=====
# ポート”PORT_A”を BGA パッケージの”A10”ピンにアサインする場合
LOCATE COMP “PORT_A” SITE “A10” ;
# ポート”PORT_B”を QFP パッケージの”34”ピンにアサインする場合
LOCATE COMP “PORT_B” SITE “34” ;
```

方法 2. HDL ソースでの設定方法

HDL ソース内でもピンアサインの設定を行う場合の記述は言語および論理合成ツールによって異なります。なお、HDL ソース内での設定と制約ファイル内の設定が異なる場合、上記制約ファイル“lpf”の設定が優先されます。

VHDL での設定方法

```
attribute LOC : string ;
attribute LOC of ポート名 : signal is “ピン番号” ;
```

例

```
=====
entity LOCEXAMPLE is
port(
RST   : in std_logic;
CLK   : in std_logic;
.
DTO   : out std_logic_vector(2 downto 0)
);
attribute LOC           : string;
attribute LOC of RST    : signal is “31” ; --// QFP パッケージの場合の記述例
attribute LOC of CLK    : signal is “A10” ; --// BGA パッケージの場合の記述例
attribute LOC of DTO    : signal is “A1, A2, A5” ; --// バスの記述例
```

--// “,”で区切ってピン番号を羅列。
--// “,”の後に空白を入れるとエラー

Verilog HDL での設定方法

```
input/output 宣言 // synthesis LOC = “ピン番号”
または
input/output 宣言 /* synthesis LOC = “ピン番号” */
```

例

```
=====
module LOCTEST ( RST, CLK, ..., DTO);
input RST ; // synthesis LOC = “A10”
input CLK ; /* synthesis LOC = “31” */
```

2. 20. 3 PAC-Designer and DAIAMOND フロー

HDL や回路図によりデザインをエントリします。FPGA 部の設計には、別途 DAIAMOND を別途使用します。Lattice 社開発ツール DAIAMOND のインストールが必要です。DAIAMOND のインストール、及び詳しい操作方法については、下記の DAIAMOND 日本語マニュアルをご参照下さい。

弊社 HP より、ドキュメント=>日本語ドキュメント=>ソフトウェアマニュアルより、DAIAMOND の項もしくは、直接下記のアドレスを参照下さい。

<http://www.latticesemi.co.jp/docs/toolmanuals/diamond/index.cfm>

DAIAMOND ではプロジェクトを新規作成し、Jedec 生成までを含めた設計を完了させておきます。DAIAMOND でのプロジェクト作成時デバイス選択は、Platform Manager よりデバイスを選択します。

FPGA 部のデザイン完了後、PAC-Designer にてそのデザインを取り込みます。2.1 節の FPGA ロジック開発環境の選択画面にて、PAC-Designer and DAIAMOND を選択し、プロジェクト名とプロジェクト・ディレクトリを設定します(図 2-56)。

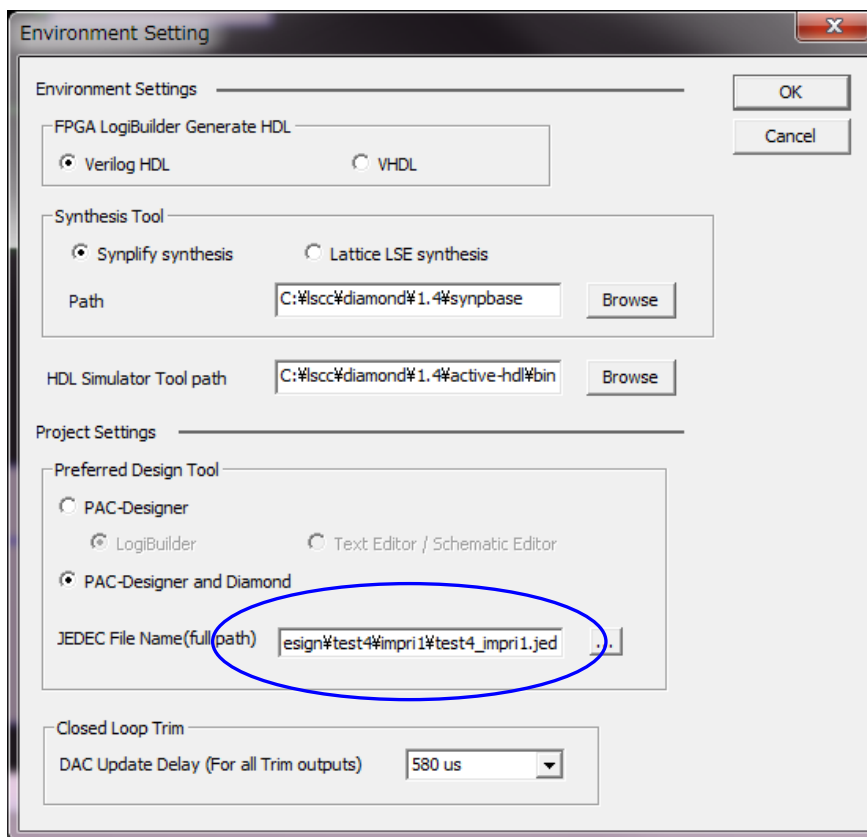


図 2-56 File List Manager 画面

2.21 UES エディタ

各デバイスの UES エディタ部をダブルクリックすると図 2-57 のような画面が起動します。ここで、設定したい bit を選択した状態で Toggle ボタンをクリックするか、各ビットをダブルクリックすると値が”0”, ”1”交互に変化しますので、各ビット値を設定します。

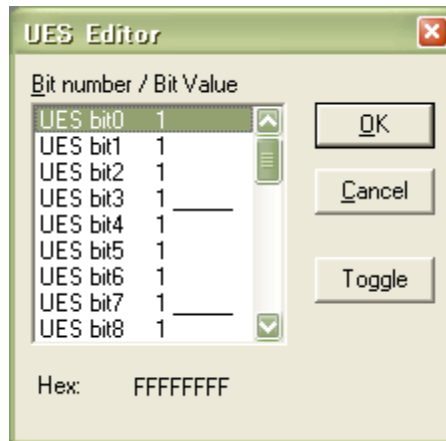


図 2-57 UES エディタ

設定が完了しましたら、OK ボタンを押して終了します。

3 コンパイル

3.1 コンパイル方法

CPLD Logic と FPGA Logic の設計が完了しましたら、フロー毎にいずれかの方法でコンパイルを実行します。

3.1.1 PAC-Designer - LogiBuilder フローの場合

PAC-Designer の CPLD Logic もしくは、FPGA Logic の画面で、2つの方法によりコンパイルを実行することができます(図 3-1)。

- ① メニューから[Tools]→[Compile LogiBuilder Design]を選択する。
- ② Compile Design アイコンをクリックする。

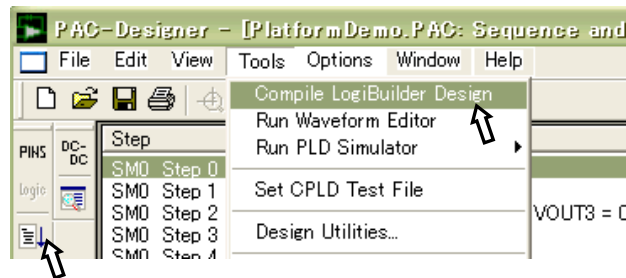


図 3-1 コンパイルの実行

3.1.2 PAC-Designer - Text Editor / Schematic Editor フローの場合

PAC-Designer の CPLD Logic の画面で、2つの方法によりコンパイルを実行することができます(図 3-1)。

- ① メニューから[Tools]→[Compile LogiBuilder Design]を選択する。
- ② Compile Design アイコンをクリックする。

PAC-Designer の FPGA Logic の画面も、2つの方法によりコンパイルを実行することができます(図 3-2)。

- ① メニューから[Tools]→[Compile HDL Design]を選択する。
- ② Compile Design アイコンをクリックする。

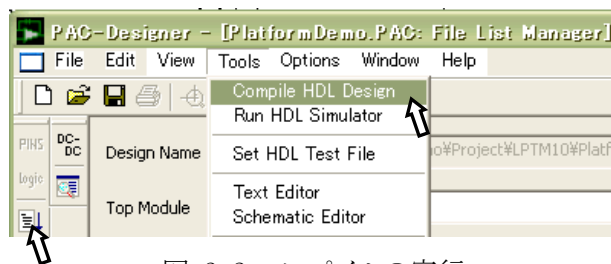


図 3-2 コンパイルの実行

3. 1. 1 PAC-Designer and DAIMOND フローの場合

PAC-Designer の CPLD Logic の画面で、2つの方法によりコンパイルを実行することができます(図 3-3)。

- ① メニューから[Tools]→[Compile LogiBuilder Design]を選択する。
- ② Compile Design アイコンをクリックする。

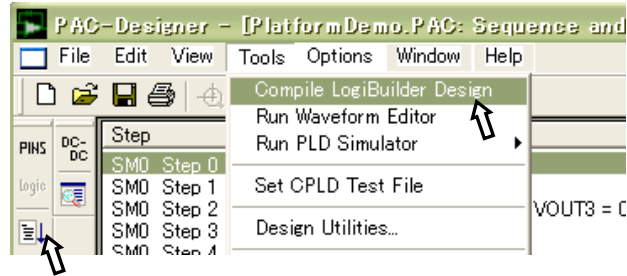


図 3-3 コンパイルの実行

3.2 コンパイル完了

コンパイルを開始するとMS-DOS画面が起動し、正常に完了すると図 3-4 のような画面が表示されますので、OKボタンを押してコンパイルを終了します。コンパイルの詳細レポートを見るには、Click here to see fitter report をクリックします。

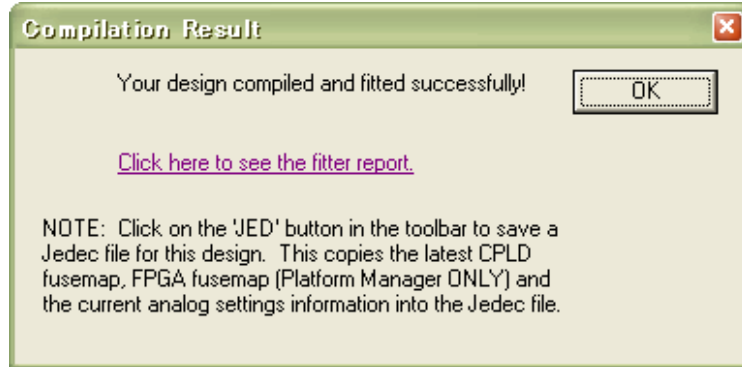


図 3-4 コンパイル成功画面

CPLD Logic に問題がある場合は、図 3-6 のような画面が表示され、OK をクリックするとエラー内容が表示されますので、エラーを修正し、再度コンパイルを実施します。

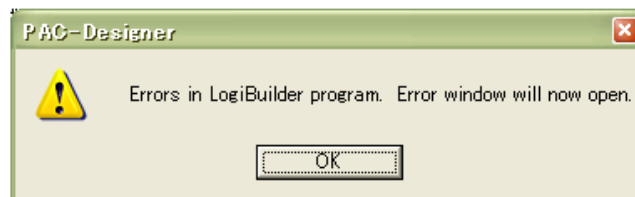


図 3-5 コンパイル失敗画面

FPGA Logic に問題がある場合は、図 3-6 のような画面が表示され、Click here for more details! をクリックするとエラー内容が表示されます。OK をクリックしこのウィンドウを閉じます。エラーを修正し、再度コンパイルを実施します。

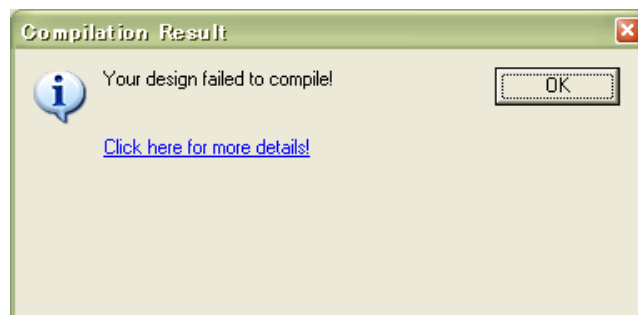


図 3-6 コンパイル成功画面

3.2.1 エラー、ワーニングと対処方法

本節ではコンパイル時に CPLD Logic 部で良く出るエラーメッセージについて説明します。

=====
Error 3: Instructions that start a timer may not follow one another. This includes WaitFor_Timer or Start_Timer instructions.
=====

WaitFor_Timer や Start_Timer 等のタイマーをスタートさせるインストラクションを連続させた場合に、このエラーが発生します。ソフトウェアの制限によりタイマーを連続して使用することはできません。タイマーインストラクションの間に NOP(No Operation)インストラクションを追加することで対処可能です。

=====
Error 8: At least one OUTPUT instruction is required, with at least one write.
Reason: The ABEL language used to implement the PLD requires at least one output.
=====

シーケンス内に OUTPUT インストラクションが一つも無い場合に、このエラーが発生します。少なくとも1つ以上の OUTPUT インストラクションを用いる必要があります。

=====
Error 10: Exception has empty Boolean Expression.
=====

例外処理(Exception)の追加だけされており、Boolean Expression 設定が空の場合表示されます。設定を完了させるか、使用しない場合は該当行を削除します。

=====
Error 14: Supervisory Logic equation has empty Boolean Expression.
=====

Supervisory Logic equation の追加だけされており、Boolean Expression 設定が空の場合表示されます。設定を完了させるか、使用しない場合は該当行を削除します。

=====
Error 22: StartTimer requires Timer to be in JK-mode.
=====

Start Timer インストラクション使用時に、使用するタイマーの Timer Gate 信号のタイプが D タイプになっている場合に、このエラーが発生します。ピンの定義画面にて、該当信号を JK タイプに変更してください(設定方法は、3.1.2 節を参照)。

上記以外のエラーメッセージが出力された場合は Lattice か代理店の FAE に問い合わせてください。

4 シミュレーション

4.1.1 CPLD Logic(シーケンス・コントローラ)のシミュレーション

PAC-Designer では、作成した CPLD Logic(シーケンス・コントローラ)のシミュレーションを行なうことが出来ます。但し、シーケンスをコントロールするデジタル部分のみシミュレーション可能で、センス電源管理部のアナログシミュレーションは含みません。PAC-Designer の CPLD Logic の画面にて、以下2つの何れかの方法によりシミュレーション波形の入力画面を起動します。

- ① メニューから[Tools]→[Run Waveform Editor]を選択する。
- ② Waveform Editor アイコンをクリックする。

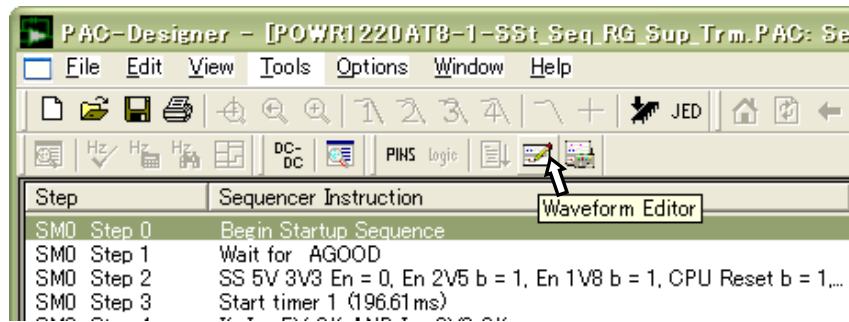


図 4-1 シミュレーション波形入力画面の起動

上記の方法により起動しますと、図 4-2 のような画面が表示されます。

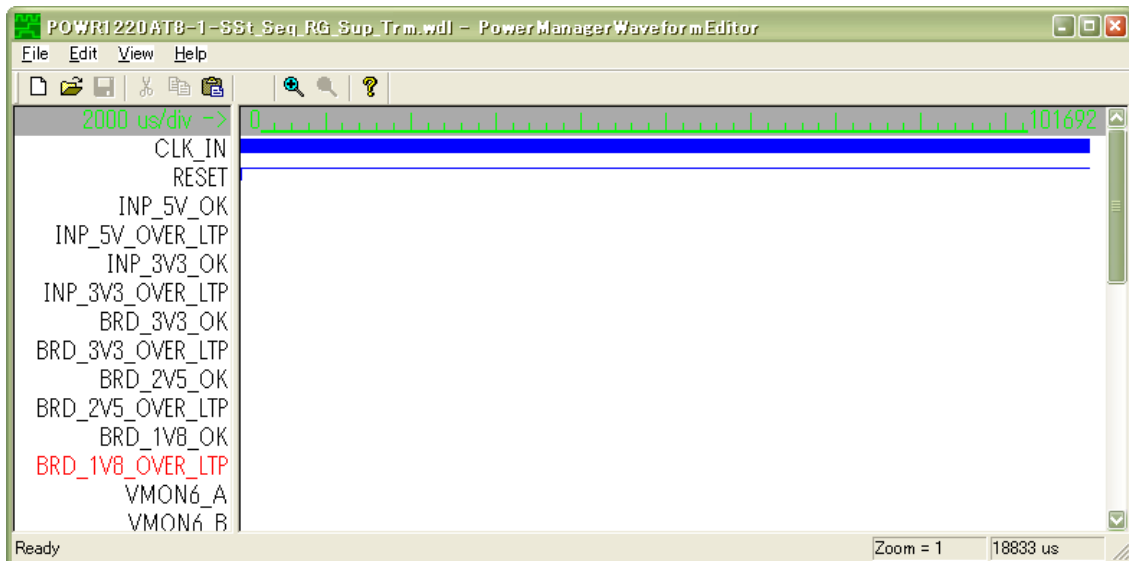


図 4-2 シミュレーション波形入力画面

上記画面起動時、入出力信号および CLK_IN(内部クロック)と RESET(アクティブ Low の外部リセット信号)が、デフォルトでリストアップされます。

4.1.2 クロックとシミュレーション時間の設定

まず、クロック周波数とシミュレーション時間の設定を行ないます。

図 4-2 の波形入力画面にて CLK_IN をダブルクリックしますと、図 4-3 のような画面が起動します。

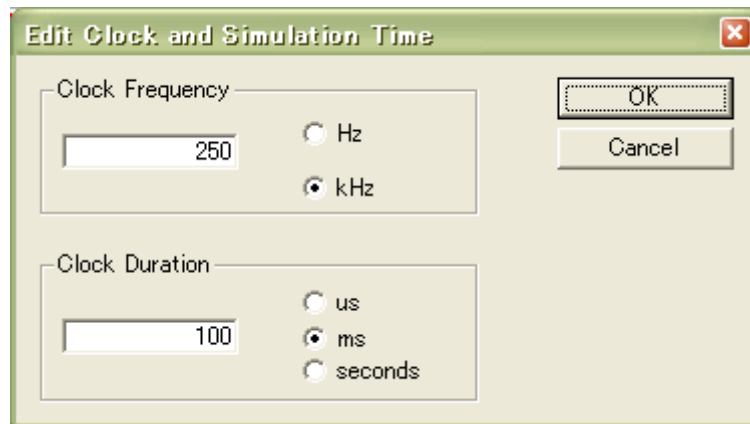


図 4-3 クロックとシミュレーション時間の設定画面

上記画面が起動しましたら、クロック周波数とシミュレーション実行時間を入力し、OKボタンをクリックします。

4.1.3 入力信号波形の設定

続いて、入力信号の設定を行ないます。

図 4-2 の波形入力画面にて入力信号をダブルクリックしますと、図 4-4 のような画面が起動します。

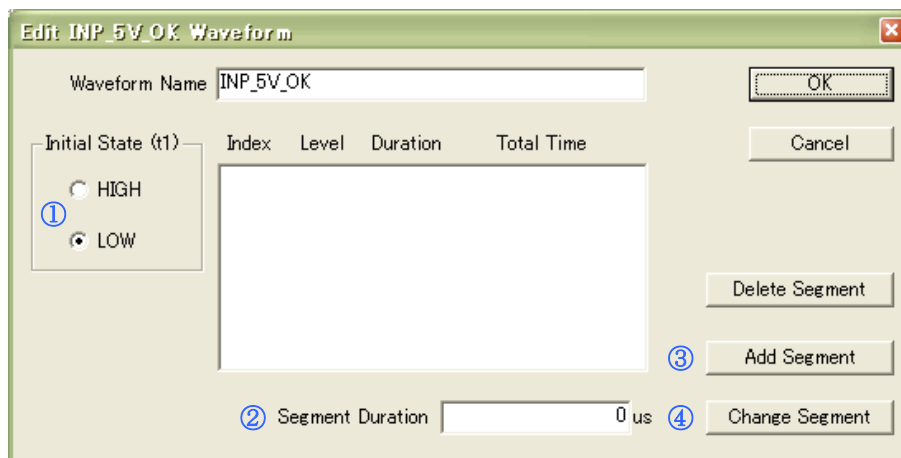


図 4-4 入力信号波形の設定

上記画面が開きましたら、下記 4 つの設定を行ないます。

- ① Initial State で HIGH もしくは LOW を選択する。
- ② Segment Duration に①で選択した値の保持時間を入力します。
- ③ 続いて値を変化させるには、Add Segment ボタンをクリックします。
- ④ Change Segment ボタンをクリックし、保持時間を設定します。

4.1.4 シミュレーションの実行

シミュレーション波形の入力が完了しましたら、波形を保存しシミュレーションを実行します。
シミュレーションは、以下2つの何れかの方法により実行します。

- ① メニューから[Tools]→[Run PLD Simulator]を選択する。
- ② PLD Simulator アイコンをクリックする。
 - ・ シーケンス・コントローラ画面表示時のみ実行可能です。

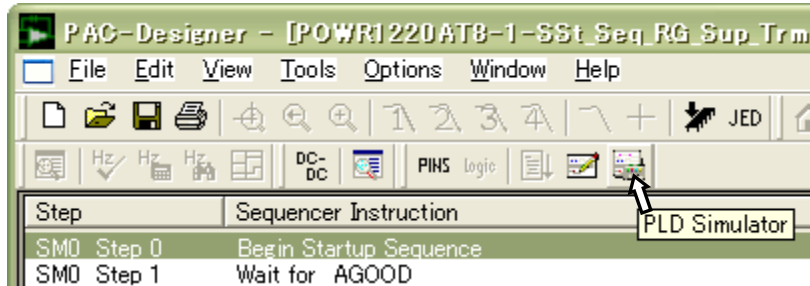


図 4-5 シミュレーション波形入力画面の起動

コンパイルを開始すると MS-DOS 画面が起動し、図 4-6 のような画面でシミュレーション結果が表示されます。

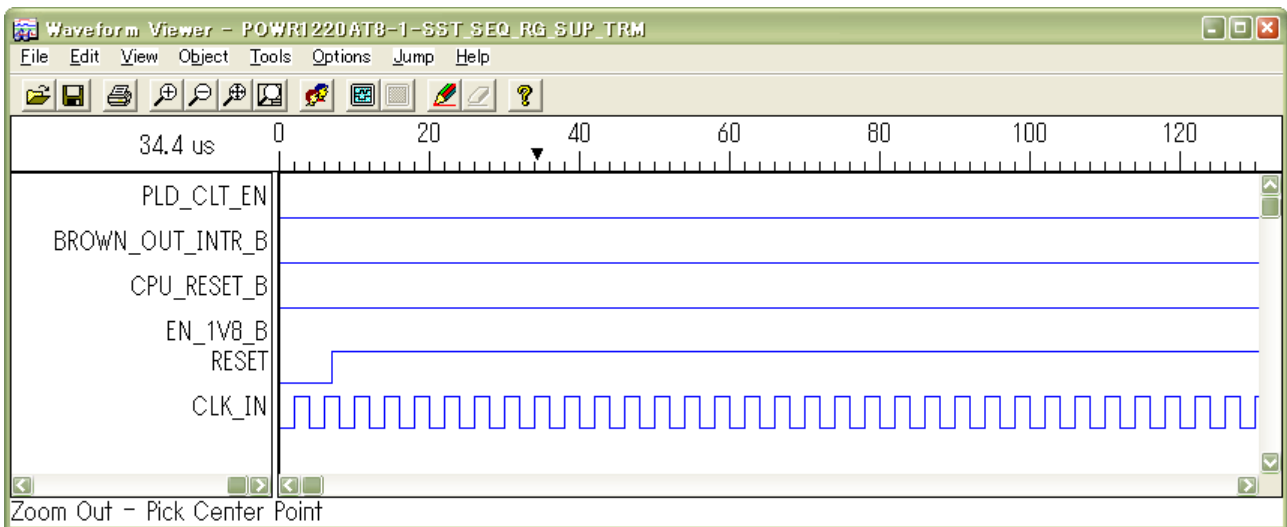


図 4-6 シミュレーション結果例

4.1.5 汎用 HDL シミュレータを使用する方法

シミュレーション・ツールとして汎用の HDL シミュレータを使用することも可能です。ファイルメニューから [File] → [Export] を選択します(図 4-7)。



図 4-7 ファイルのエクスポート

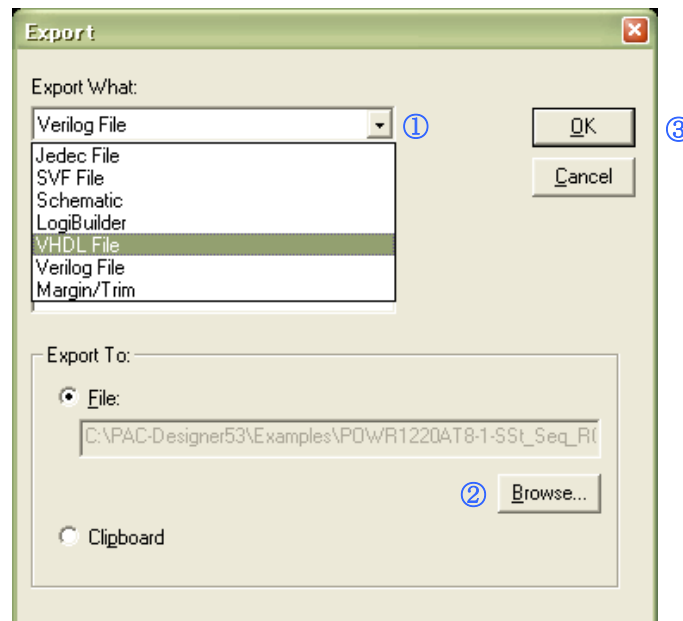


図 4-8 フォーマットの選択

図 4-8 の画面が開きましたら、下記 3 つの設定を行ないます。

- ① Export What で VHDL File か Verilog File を選択。
- ② Browse ボタンをクリックし、Export To の File 欄にファイルの生成場所とファイル名を指定。
- ③ OK ボタンを押し、ファイルを生成する。

尚、汎用 HDL シミュレータ用の POWR デバイスライブラリは、PAC-Designer をデフォルトでインストールした場合、下記に保存されています。

C:\PAC-Designer**¥cae_library¥simulation¥verilog¥powr¥src
 C:\PAC-Designer**¥cae_library¥simulation¥vhdl¥powr¥src

4.2 供給電源ランプレートのシミュレーション

PAC-Designerでは、HVOUT出力で制御する供給電源のランプレートをシミュレーションすることが可能です。まずメニューの「Tools」→「Design Utilities…」を選択しますと、図 4-9 のような画面が起動します。

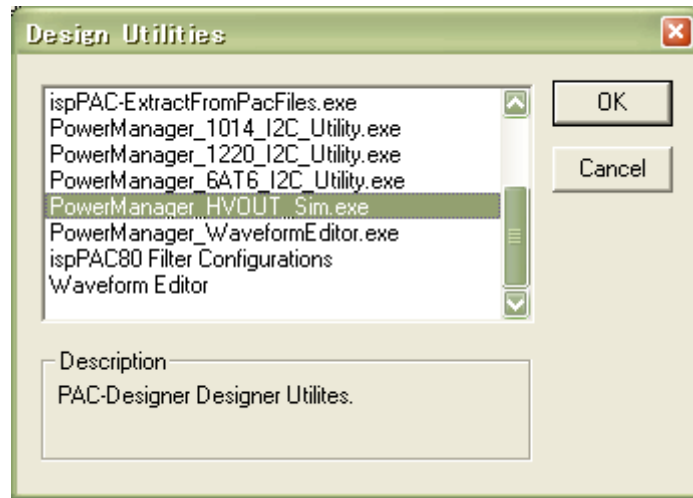


図 4-9 デザイン・ユーティリティ選択画面

上記画面より「PowerManagerHVOUT_Sim.exe」を選択しますと、図 4-10 のような画面が起動します。

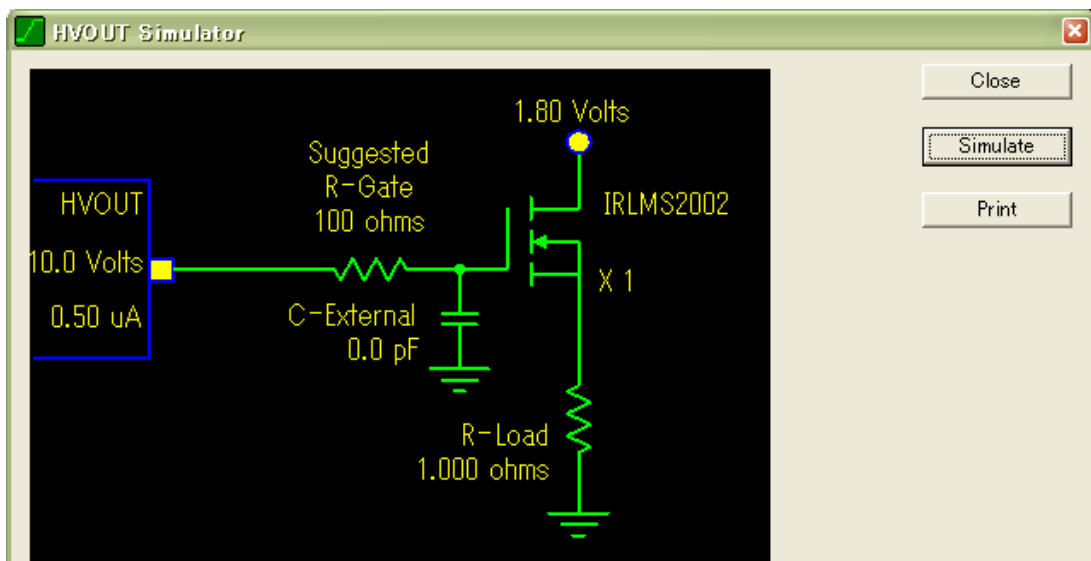


図 4-10 高電圧出力 シミュレータ

高電圧シミュレータが起動したら、HVOUTからの出力ピンに接続されるFETおよび周辺回路についてのパラメータを調整します。

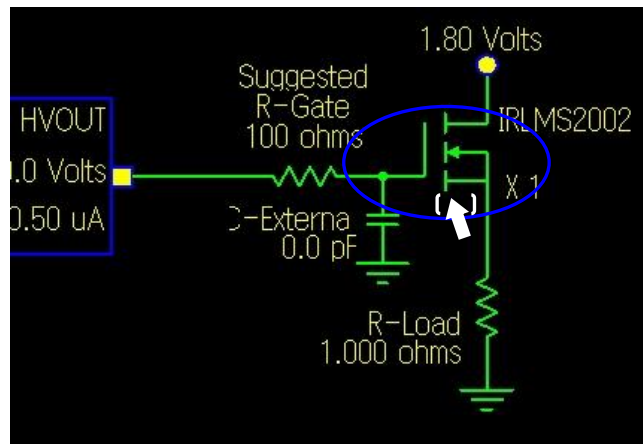


図 4-11 MOSFET の設定

高電圧シミュレータ画面にて、MOSFET 付近をダブルクリックしますと、図 4-12 のような画面が起動します。

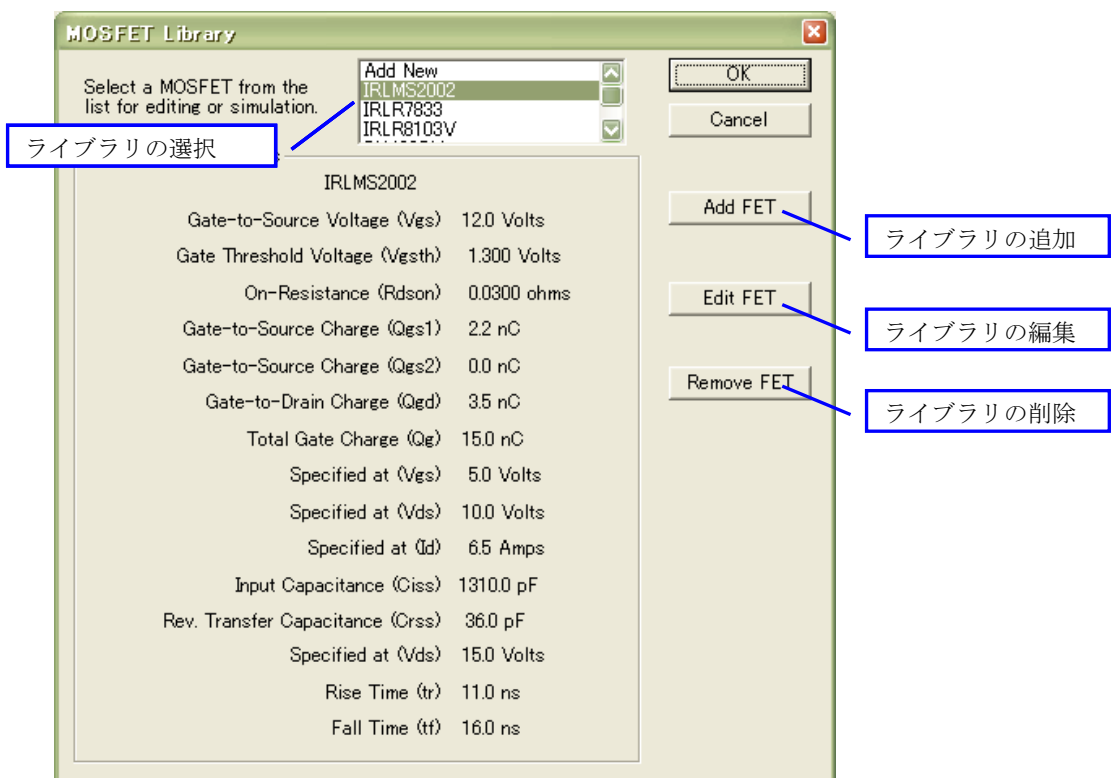



図 4-12 MOSFET ライブラリ

上図の画面が起動しましたら、一覧から MOSFET を選択します。
また、Add FET ボタンをクリックすることで、新規にライブラリを追加することが可能です。
登録されているライブラリを編集するには、Edit FET ボタンをクリックします。

MOS-FET のライブラリを選択しましたら、その他のパラメータを編集します。
各パラメータ上(マウスアイコンがに変化する場所)をダブルクリックしますと、編集画面が起動します。

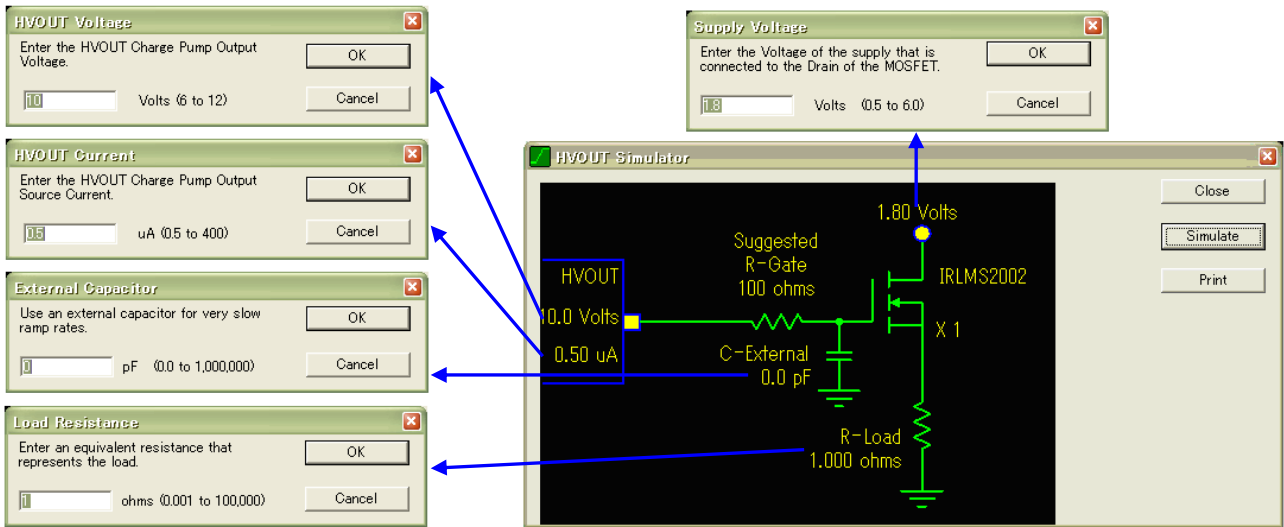


図 4-13 パラメータの編集

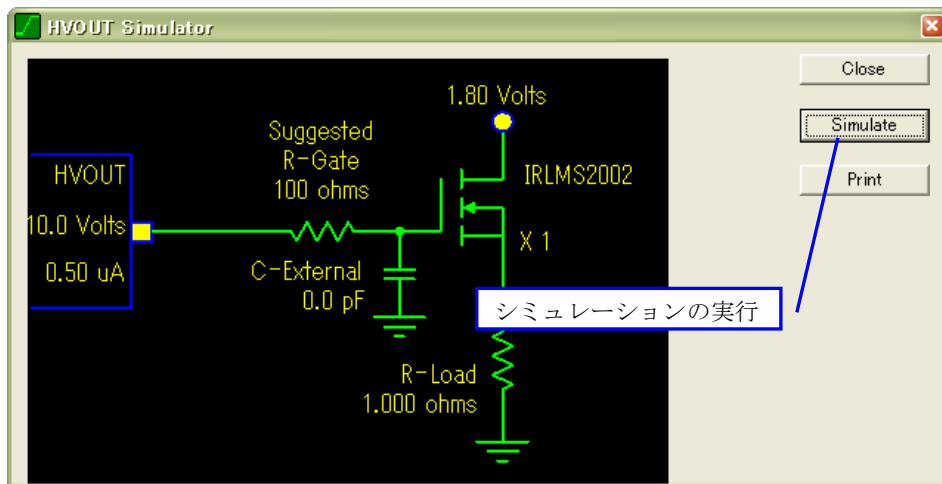


図 4-14 シミュレーションの実行

各パラメータの編集が完了しましたら、Simulate ボタンをクリックしシミュレーションを実行します。

シミュレーションを実行しますと、図 4-15 のような画面が起動します。

本画面では、「ディヴィジョンの変更」、「カーソルの表示」、「カーソル位置に対する電圧」、「時間の表示」、「2点間の傾きの表示」、「印刷」、「CSV ファイルへの出力」を行なうことができます。尚、カーソル 1 は左クリックを押しながら、カーソル2は右クリックを押しながら移動することが可能です。

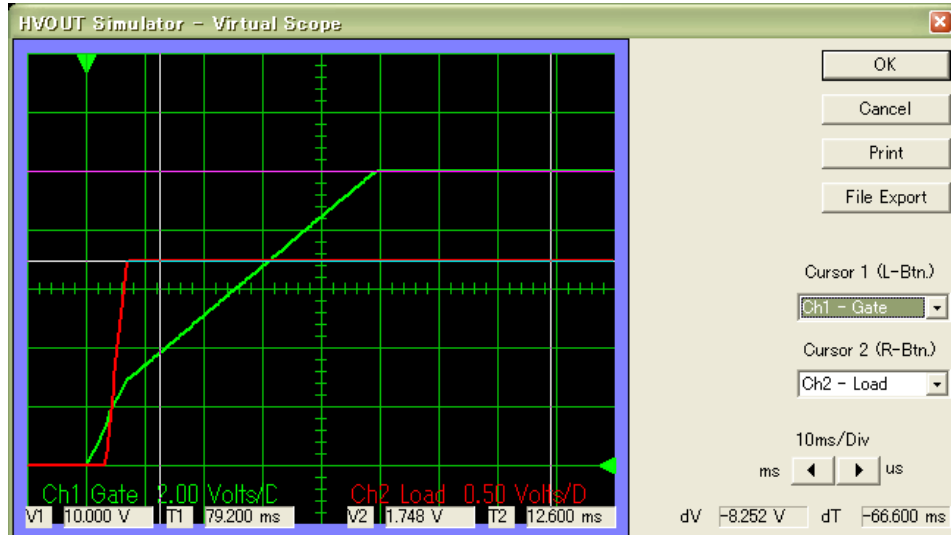


図 4-15 シミュレーション画面

4.3 FPGA Logic のシミュレーション

作成した FPGA Logic のシミュレーションは、DAIAMOND に付属の HDL シミュレータ、ACTIVE HDL を使用します。

5 デバイスへの書き込み

デバイスへの書き込み方法は、PAC-Designer から書き込む方法と、ispVM System を使用する方法の 2 種類があります。

5.1 PAC-Designer からの書き込み

5.1.1 ダウンロード・ケーブルの設定

デバイスへのダウンロードを行なうケーブルとして、パラレルポートタイプと USB タイプを使用することが出来ます。PAC-Designer からダウンロードを行なうためには、どちらのタイプのダウンロード・ケーブルを使用するかを設定する必要があります。

ダウンロード・ケーブルの設定を行なうには、メニューから[Options]→[Cable and I/O Port Setup]を選択します。

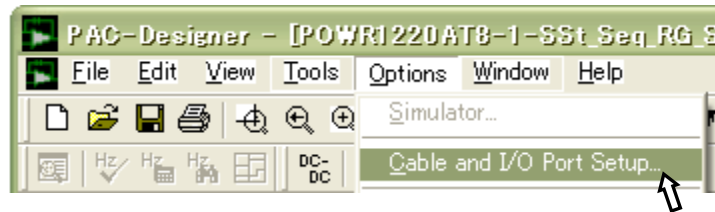


図 5-1 ダウンロード・ケーブルの設定画面の起動

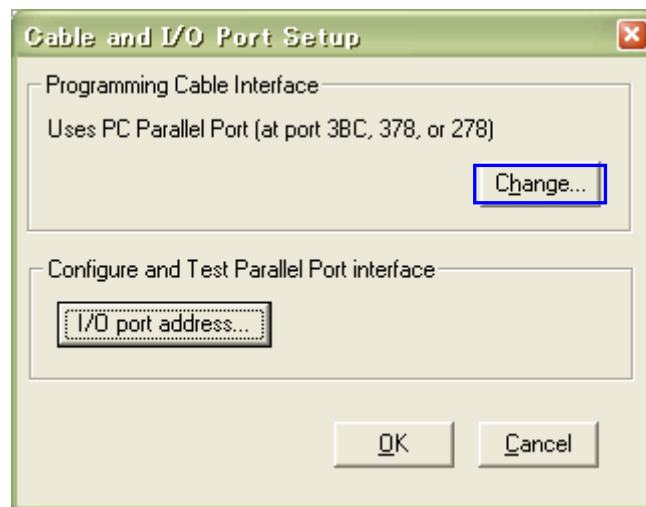


図 5-2 ダウンロード・ケーブルの設定

図 5-2 のような画面が起動しましたら、Change ボタンをクリックします。

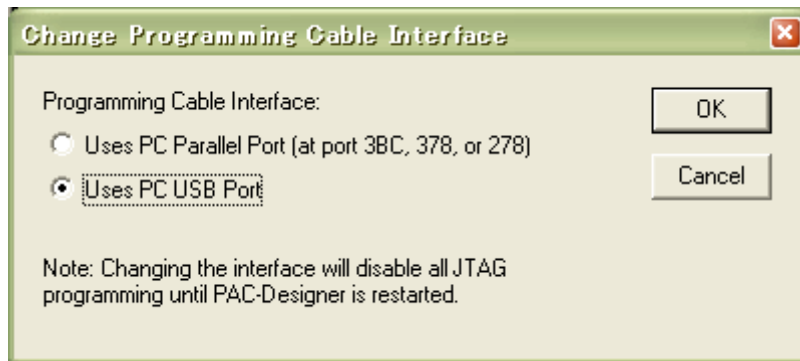


図 5-3 ダウンロード・ケーブルの選択

図 5-3 のような画面が起動しましたら、Parallel タイプか USB タイプかを選択し、OK ボタンをクリックします。

5.1.2 デバイスへのダウンロード

PAC-Designer からデバイスへの書き込みを行なうには、以下2つの方法により実行することができます。

- ① メニューから[Tools]→[Downloads]を選択する。
- ② Downloads アイコンをクリックする。

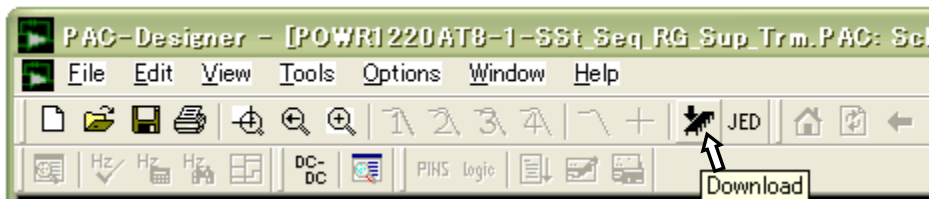


図 5-4 デバイスへの書き込み

5.2 ispVM System からの書き込み

PAC-Designer 上で JEDEC ファイルを生成することで、CPLD や FPGA 同様に ispVM System を使用して書き込みを行なうことが可能です。

JEDEC ファイルを生成するには、以下 2 つの方法により実行することができます。

- (ア) メニューから[File]→[Export]を選択する(下記図 5-5 の設定画面)。
- (イ) Export JEDEC File アイコンをクリックする(保存場所を示すダイアログ画面と共に、Jedec ファイルが出力直接出力されます)。

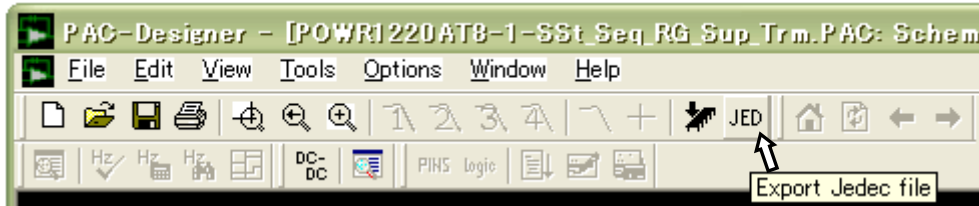


図 5-5 JEDEC ファイルの生成

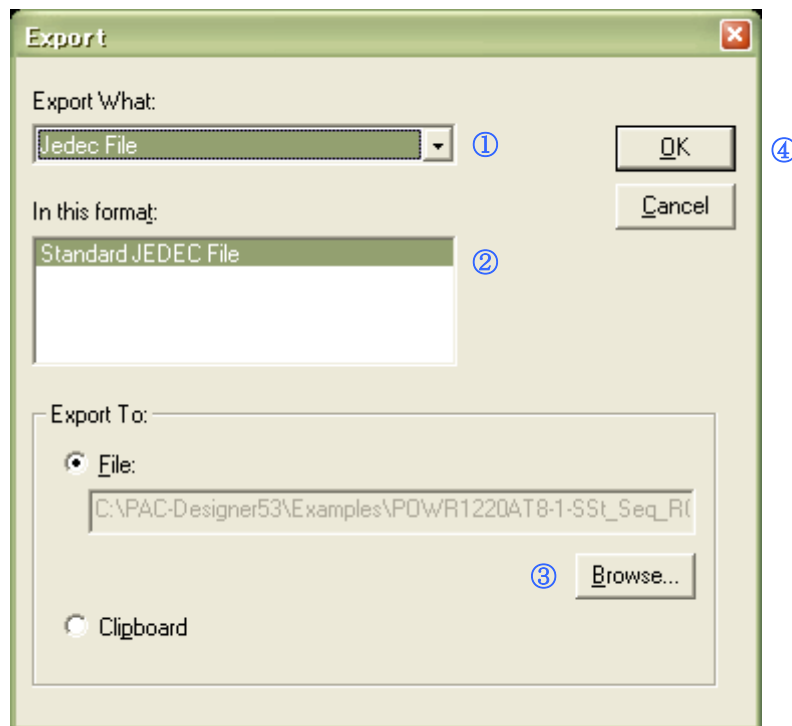


図 5-6 JEDEC ファイルの生成

図 5-6 の画面が開きましたら、下記4つの設定を行ないます。

- ① Export What で Jedec File を選択。
- ② In this format で Standard JEDEC File を選択。
- ③ Browse ボタンをクリックし、Export To の File 欄に JEDEC ファイルの生成場所とファイル名を指定
- ④ OK ボタンを押し、ファイルを生成する。

JEDEC ファイル生成後の、書き込み方法は CPLD/FPGA と同様です。
詳細は、ispVM System のユーザーマニュアルを参照ください。

JEDEC ファイルを生成しますと、ヘッダー部に Author(設計者)が入ります。
この欄が、日本語(2 バイト文字)の場合デバイスへの書き込みが正常に行なえませんが
Author を空欄もしくは、半角文字に変更する必要があります。

Author の変更は、メニューの[File]→[Summary Information]を選択し、図 5-7 のような画面にて変更します。

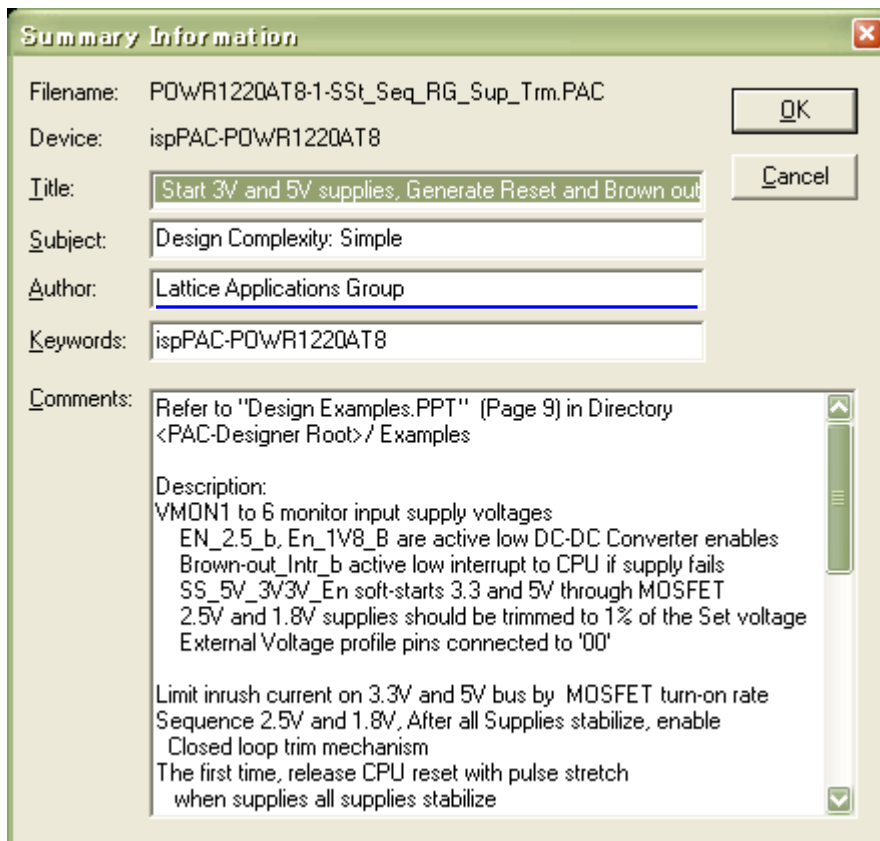


図 5-7 Summary Information 画面

6 I2C Utility

対象デバイス : LPTM10-12107

デバイスへの書き込み後、USB・パラレルのダウンロード・ケーブルを使用し、デバイス内部レジスタへのリード・ライトが可能です。ダウンロード・ケーブルの接続先は、デバイスの JTAG 端子ではなく、I2C 端子となります。

あらかじめ、デザインで I2C インタフェース用のデバイスアドレスを指定します(2.11 節)。

メニューの「Tools」→「Design Utilities…」を選択しますと、図 6-1 のような画面が起動します。

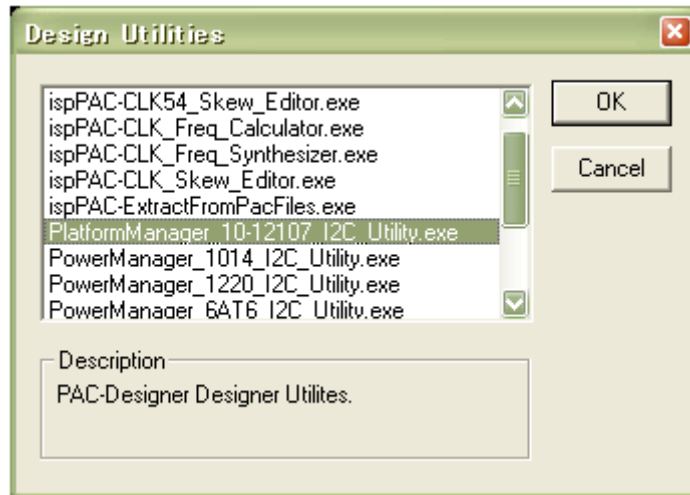


図 6-1 デザイン・ユーティリティ選択画面

使用するデバイスに合わせ、ユーティリティツールを選択し、OK を押すと図 6-2 の様な I2C ユーティリティが起動します。

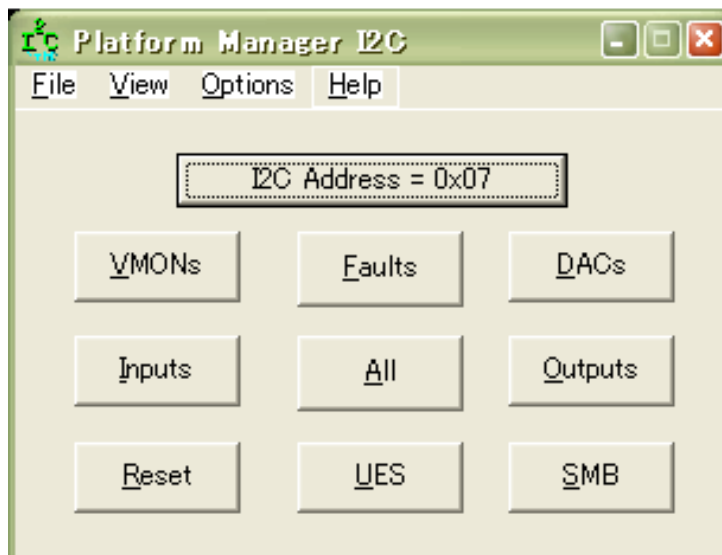


図 6-2 I2C ユーティリティ

Option の I2C Interface を起動し (図 6-3)、図 6-4 の画面にてダウンロード・ケーブルの種類を設定します。



図 6-3 I/O セットアップの起動

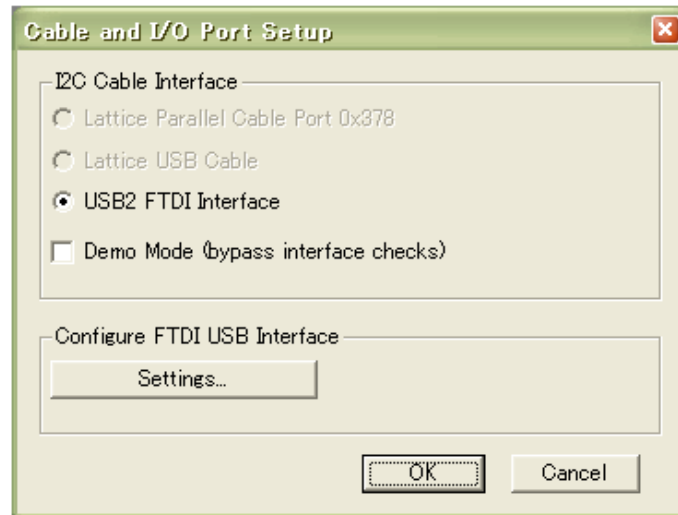


図 6-4 I/O セットアップ画面

I2C ユーティリティの I2C アドレスを、あらかじめデザインで設定している値に合わせます(図 6-5)。設定後は各メニューボタンにてレジスタ・アクセスが可能です。“Faults”メニューに関しては I2C アドレスの設定に関わらず動作します。

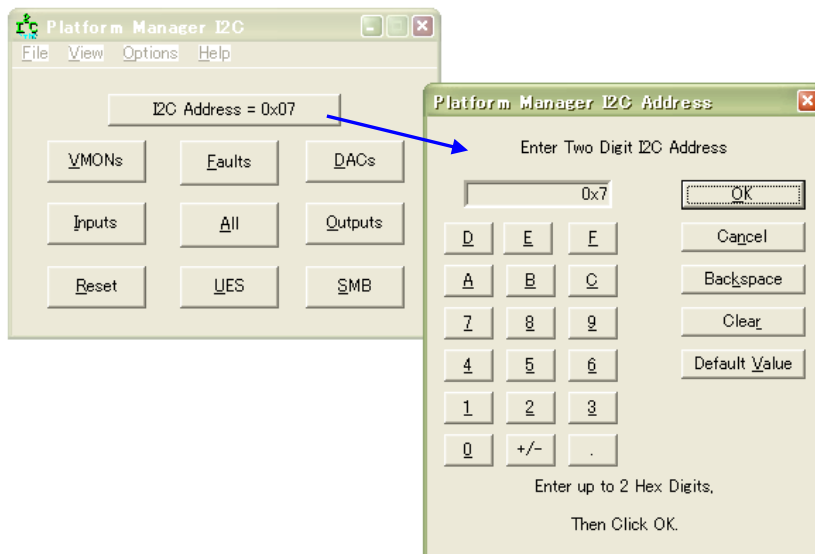


図 6-5 I2C アドレスの設定

7 プラットフォーム・マネージャーを正しく使用するために (TN1223 より)

プラットフォーム・マネージャーを正しくご使用いただく為、ボード設計前に本節も必ずご確認ください。

7.1 デバイスの電源要求

プラットフォーム・マネージャーは複数の電源システムを持ち、それぞれ適切に扱う必要があります。図 7-1 には基本的な電源周辺と JTAG 接続を示します。この図では以下に案内している、必要とされるすべてのコンデンサは描かれておりません。多くのアプリケーションでは、APS を除くすべての電源ピンは共通の 3.3V に接続します。VCCA はフェライト・ビーズで分離します。また、すべてのグラウンドピン (GND, GNDIO, PGND, PGND, PGND) は、共通の GND に接続します。

VCC

FPGA 部の主電源で推奨電圧範囲は 3.135V~3.465V です。すべての VCC ピンには可能な限り近くに、それぞれ 0.1 μ F のデカップリング・コンデンサを配置します。さら推奨としては、この 3.3V 電源システムには、1-10 μ F のタンタル・コンデンサを 1 個配置します。

VCCAUX

FPGA 部の補助電源で推奨電圧範囲は 3.3V \pm 5%です。VCCAUX ピンは可能な限り近くに、0.1 μ F のデカップリング・コンデンサを 1 個配置します。VCCAUX は上記の VCC と同一の電源システムに接続します。

VCCIO0, VCCIO1, VCCIO3

FPGA 部の各 I/O バンク電源です。これらの I/O バンク (0, 1, 3) 電源の推奨電圧範囲は 1.14V~3.464V です。各ピンには少なくとも 0.1 μ F のデカップリング・コンデンサを 1 個配置します。また、例えば複数の I/O バンクが 3.3V を使用する場合は、すべて共通の 3.3V に接続します。未使用の I/O バンクは VCCIO を未接続 (フローティング) にしておきます。

VCCIO2

FPGA 部の I/O バンク 2 と、FPGA 部の JTAG ポートの電源です。この電源の推奨電圧範囲は 2.25V~3.6V です (他の I/O バンクと異なるのでご注意ください)。0.1 μ F のデカップリング・コンデンサを 1 個配置します。

PVCCA

CPLD 部のアナログ回路電源です。0.1 μ F のデカップリング・コンデンサを 1 個配置します。また他の VCC レ系統から分離する為、フェライト・ビーズを 1 個配置します。推奨電圧範囲は 2.8V~3.96V です。

PVCCD

CPLD 部の主電源です。0.1 μ F のデカップリング・コンデンサを 1 個配置します。推奨電圧範囲は 2.8V~3.96V です。

PVCCINP

CPLD 部のデジタル入力ピン (N1~N4) 電源です。推奨電圧範囲は 2.25V~5.5V です。

PVCCJ

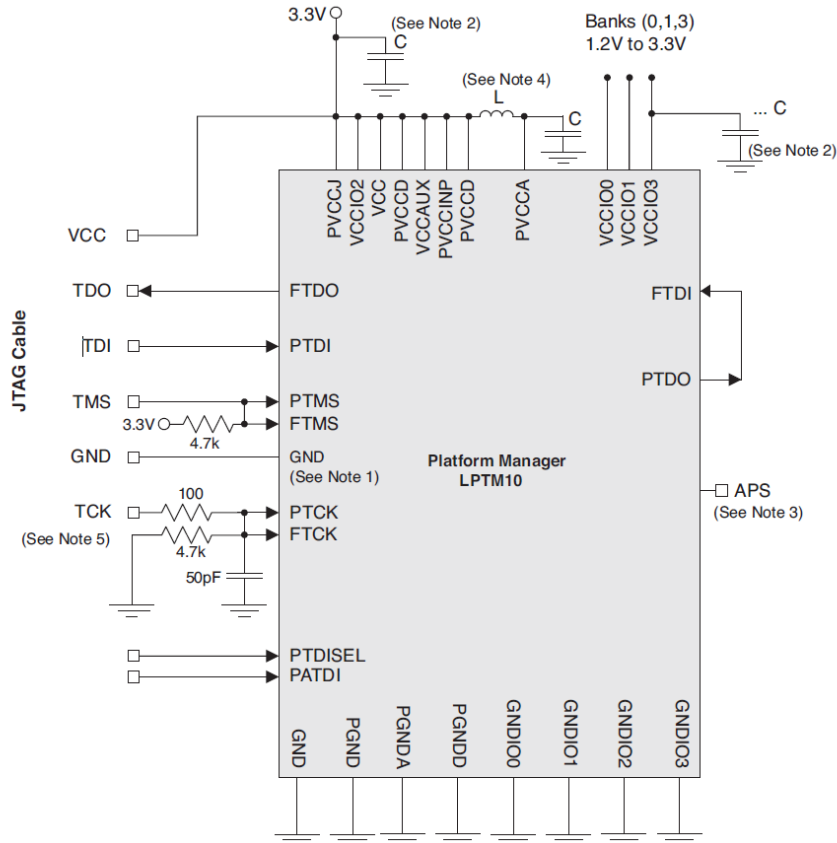
CPLD 部の JTAG ポート電源です。この電源の推奨電圧範囲は 2.25V~3.6V です。0.1 μ F のデカップリング・コンデンサを 1 個配置します。このピンは FPGA 部の JTAGA ポート電源である、VCCIO2 部への供給電源と同一の電源に接続します。

JTAGプログラミング

プラットフォーム・マネージャーは2つのJTAGポートを持ち、外部で接続が必要です。図7-1を参照し適切な接続を行ってください。

APS

デバイス内部E²CMOSへのもう一系統の電源供給ピンです。VCCDとVCCAが電源に接続されている、通常的环境下では、このピンにはなにも接続してはいけません。フローティングとします。



1. Connect all GND pins to Ground (GND, PGND, PGND A, PGND D, GNDIO0, GNDIO1, GNDIO2, GNDIO3).
2. Use at least one 0.1uF cap per VCC pin and at least one bulk Tantalum cap.
3. APS not connected.
4. Ferrite Bead, Steward Part# H10603P600R-10.
5. Optional R/C filter for noisy JTAG environment.

図 7-1 電源及び JTAG 接続

7.2 I/O ピンのインタフェース

CPLD 部入力ピン

CPLD 部には4つの入力ピンがあり、PVCCINPにより電源が供給されます。未使用の入力ピンはGNDに接続します。

CPLD 部入出力ピン

OUT5～OUT16はオープンドレイン出力ピンです。これらのピンは外部プルアップ抵抗で処理します。未使用の出力ピンはフローティングにします。プルアップ電圧は5Vまで対応可能です。

VMON ピン

VMONピンは電圧のモニタに使用します。負荷にできるだけ近い場所で測定対象に接続します。VMONGSピンはGNDに接続します。差動で精度の高い測定を行うために、VMON測定箇所の近くに接続します。未使用のVMONピンはフローティングかGND接続にします。

HVOUT ピン

VMONピンはNチャンネルMOS-FETの制御に使用します。またこのピンはオープンドレイン出力とすることもできます。MOS-FETドライバとして使用する場合、10～100Ωの直列抵抗を使用し、抵抗は可能な限りFETのゲート近くに配置します。オープンドレイン出力で使用する場合、外部プルアップ抵抗が必要です。プルアップ電圧は13Vまで対応可能です。未使用のHVOUTピンはフローティングかGND接続にします。

TRIM ピン

TRIMピンはDACのTRIM出力です。デバイス内部のトリム機能と外部抵抗と共に用い、外部DC-DCの出力電圧設定を行います。トリム用のレジスタは可能な限りDC-DCのトリム端子の近くに配置します。TRIMピンとVMONピンは関連付けがあり、例えばTRIM2はVMON2に関連つけられます。未使用のTRIMピンはフローティングにします。

FPGA I/O ピン

FPGA部のI/Oピンは、差動も含め様々な種類のIFをサポートします。I/Oをサポートするバンクは4つあり、各バンクは独立したVCCIOとGNDを持ちます。未使用のピンは内部プルアップ付の入力ピンとして設定されます。未使用ピンは全てオープンにします。

SLEEPN ピン

SLEEPNピンがLに設定されている間、FPGA部はスリープモードとなります。4.7K～10KΩのプルアップ抵抗でVCCに接続します。

I²C Control Signals

プラットフォーム・マネージャーはI²Cバスを持ち、ADCの値のリードやTRIM用のDAC制御を行います。I²Cバスには、標準的なオープンドレインのピンで接続します。SDAとSCLピンのプルアップ抵抗には2.2K～3.3KΩを使用します。バスは100kHz～400kHzで動作します。

7.3 クロックピンの使用方法

プラットフォーム・マネージャーは複数のクロックシステムを持ち、ボード上での接続とソフトウェアでの設定が必要です。また、FTIMER の為のクロック接続も重要です。以下に BGA パッケージと TQFP パッケージでのボード上での接続を示します。

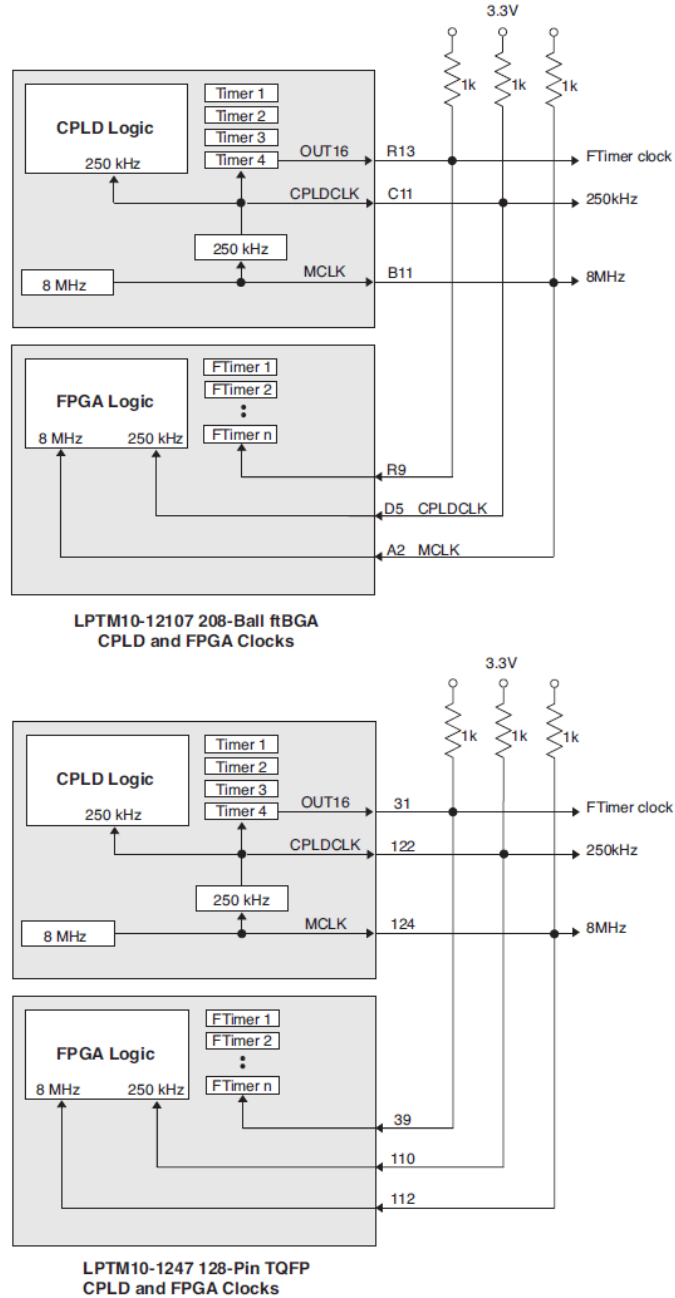


図 7-2 クロック接続

更新履歴

表 8-1 更新履歴

バージョン	リリース日	更新内容
Ver1.00	2010/12	初版リリース(PAC-Designer6.0 対応) K.Sakashiri
Ver1.10	2011/7	9 ページ、OV/UV 設定について補足追加。PAC-Designer6.1 の DAIAMOND 対応
Ver1.11	2012/2	誤植修正、PAC-Designer6.2 対応