

第 5 章 モジュール / IP 生成

本章では Lattice FPGA ファミリ固有のモジュールや Lattice オリジナル IP のパラメータ設定と生成用ツールである、IPexpress および Clarity Designer の詳細について記述します。カウンタや、汎用的な機能モジュールを実装する場合、論理合成ツールに依存する推論 (Inferencing) ベースの手法は、ターゲットフリーという長所はありますが、IPexpress/Clarity Designer で生成したモジュールを RTL 記述で明示的にインスタンスした方法に比較して、エリアやスピード的に劣る場合が一般的です。

5.1 概要

Diamond では IPexpress と呼ぶツールが本機能のために用いられてきましたが、Ver.3.3 以降は ECP5 ファミリを最初の対応デバイスファミリとして、新規に Clarity Designer が導入されました。IPexpress と Clarity Designer の機能比較概要を以下に示します。ご覧の通り、Clarity Designer はモジュール / IP のパラメータ設定と生成に加えて、『ビルド』および『プランニング』機能が追加されています。将来的には新規導入ファミリを随時 Clarity Designer による対応に移行していきます。

特に ECP5 における DDR メモリ・インターフェイスの実装時は、PCB 設計の前に最適なポート配置を Clarity Designer のプランニングで決定することを強く推奨します。『パーティシャル VCCIO (GND)』の配置も推奨 (ツールが自動的に選択) されていますので、関連テクニカルノートを参照するとともにご注意ください。

表 5-1. IPexpress と Clarity Designer の機能比較

		IPexpress	Clarity Designer
パラメータ 設定、生成	Modules	Yes	Yes
	IP	Yes	Yes
	Download IP	Yes	Yes
ビルド	Rule Checking	No	Yes
	Generate Connectivity	No	Yes
	Connection Assistance	No	Yes
	Design Re-use	No	Yes
フロアプラ ニング (PCS、DDR メモリ / GDDR)	pre-Synthesis	No	Yes
	Placement Assistance	No	Yes
	Rule Checking	No	Yes
	Graphical Usage	No	Yes

Clarity Designer については 5.3 節で詳述しますが、IPexpress に対して以下のような違い・特長があります。

- ・ IPexpress が単一のモジュールや IP を個別に生成するのみ。その他の作業一切はユーザが行う
- ・ Clarity Designer はモジュールや IP の生成のみならず、“サブシステム”と呼ぶこうした複数のブロックの接続指定して上位階層の機能ブロックとして扱うことが可能
- ・ Clarity Designer はハードマクロを含む IP の配置 (Placement) の支援、一部ポート指定が可能
- ・ Clarity Designer のサブシステムにはユーザ記述 HDL モジュールも取り込み可能

© 2014 Lattice Semiconductor Corp. (註: 本 Lattice Diamond 日本語マニュアルは、日本語による理解のため一助として提供しています。その作成にあたっては各トピックについて、それぞれ可能な限り正確を期しておりますが、必ずしも網羅的ではなく、或いは最新でない可能性があります。また、意図せずオリジナル英語版オンラインヘルプやリリースノートなどと不一致がある場合もあり得ます。疑義が生じた場合は、ラティスセミコンダクター正規代理店の技術サポート担当にお問い合わせ頂くか、または極力最新の英語オリジナル・ソースドキュメントを併せて参照するようにお願い致します。)

本日本語ドキュメントは Diamond 3.3 用ですが、Clarity Designer に関しては 3.4 用の記述ですので、ご注意ください。Clarity Designer には現時点で ECP5 のみが対応します。ツールの成熟度等の観点から Diamond 3.4 (または 3.5) 以降を使用してください。

5.2 IPexpress の基本操作

5.2.1 ウィンドウの起動


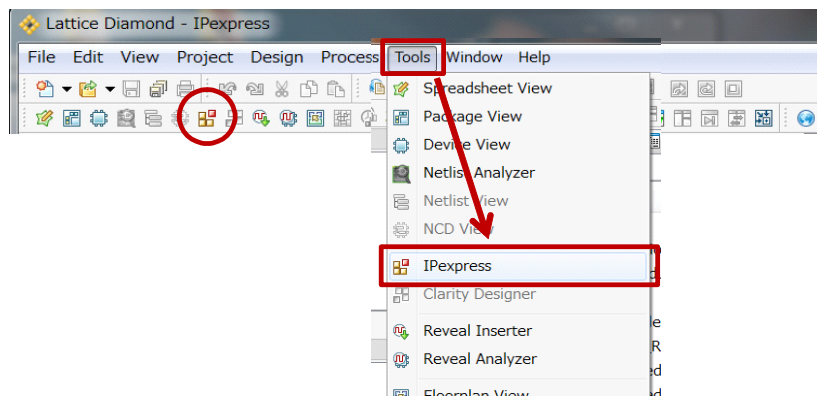
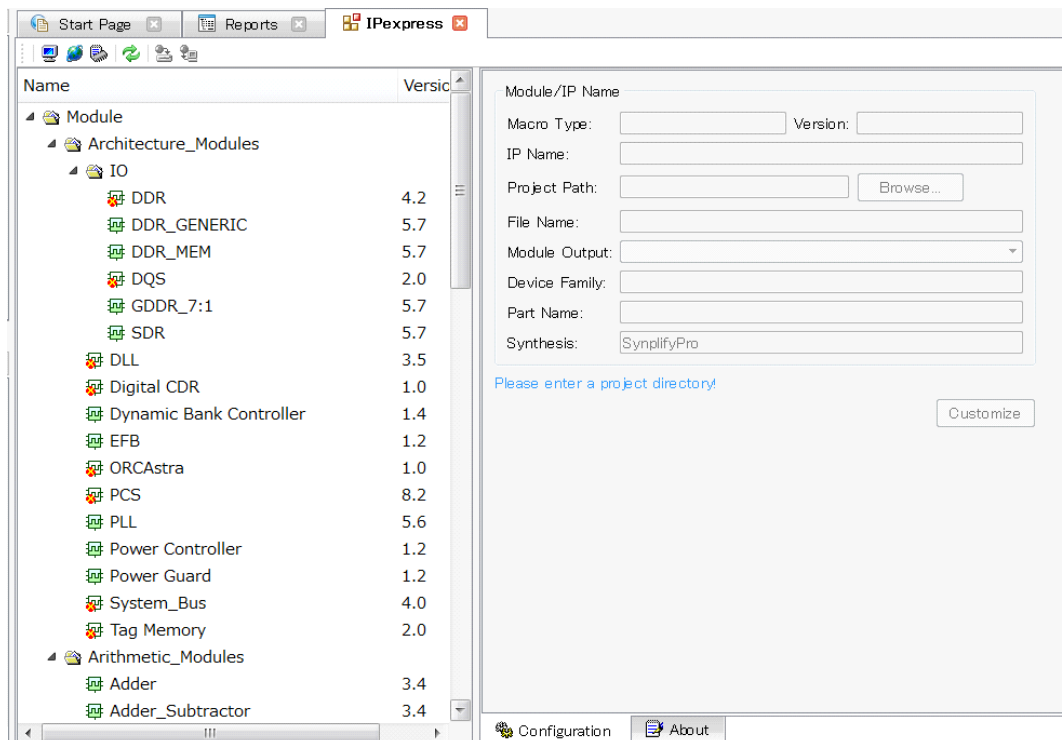
Clarity Designer 非対応デバイスでは IPexpress を起動します。メニューバーの Tools をクリックすると表示されるツール群の中から IPexpress を選択する (図 5-1、左) か、アイコンメニューの中から  をクリックします (同、右)。

図 5-1. IPexpress の起動 (左: アイコンから、右: メニューバーから)



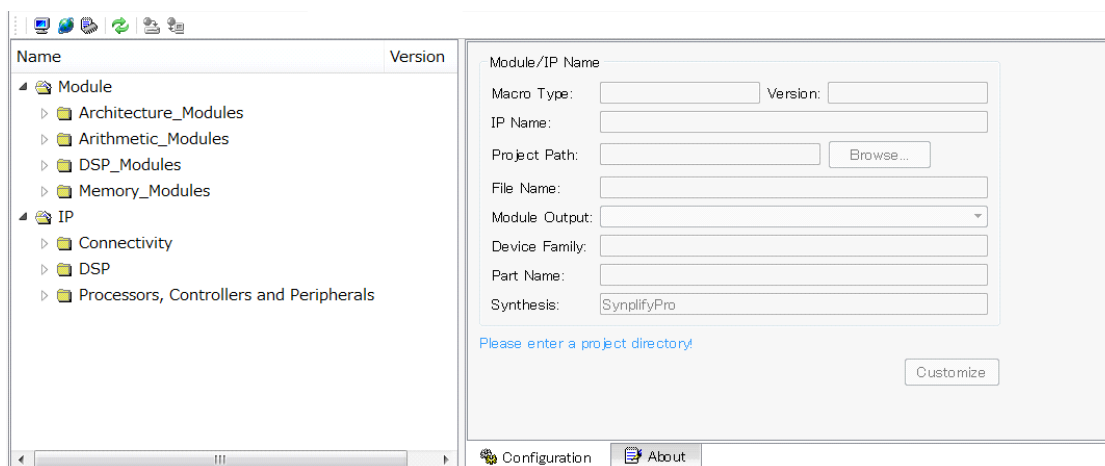
初期画面は図 5-2 のようになります。

図 5-2. IPexpress 初期画面



左枠の Name 部は Module セクションと IP セクションに分かれており、それぞれがサブセクションに分割されています。サブセクションの構成は図 5-3 のようになっています（意図的に詳細を隠した状態）。

図 5-3. IPexpress の Name 部の標準的なセクション（項目）構成



Architecture_Modules 下には図 5-2 のようなマクロが表示され、これは各デバイスファミリー固有の機能ブロックが分類されています。従って表示アイテムはファミリー毎に異なります。図 5-2 は MachXO3L の場合です。

Arithmetic_Modules と DSP_Modules、および Memory_Modules 下のマクロアイテムは図 5-4 のようになります。Arithmetic_Modules はカウンタや演算機能を LUT+FF で構成する場合の機能ブロックです。DSP_Modules は演算機能一式の機能ブロックで、パラメータを指定する際の個別詳細設定 GUI で DSP 用ハードマクロを用いるか、LUT+FF で実現するかを選択できます。Memory_Modules セクションはオンチップメモリの生成用で、Distributed_RAM 部は LUT+FF を用いた構成（分散メモリ）、EBR_Components はブロックメモリ（EBR）を用いた構成です。どちらにも属さない FIFO と FIFO_DC および RAM_Based_Shift_Register は個別詳細設定 GUI で EBR を用いるか、LUT+FF で実現するかを選択します。


なお、各名称行頭にあるアイコンのうち  印は、選択しているデバイスでは未サポートを意味します。デバイスの選択が正しいか（意図したものか）を含めて、留意が必要です。例えば MachXO3L には DSP マクロがありませんので、DSP_Modules 部にある各マクロは同印が表示されて選択できません。

図 5-4. Module 部サブモジュール毎のマクロ項目（MachXO3L の例）

Name	Version
Module	
Architecture_Modules	
Arithmetic_Modules	
Adder	3.4
Adder_Subtractor	3.4
Comparator	3.5
Complex_Multiplier	1.7
Convert	1.3
Counter	4.5
FFT_Butterfly	1.4
LFSR	3.6
Mult_Add_Sub	2.7
Mult_Add_Sub_Sum	2.7
Multiplier	4.8
Multiply_Accumulate	2.7
Sin-Cos_Table	1.5
Subtractor	3.4
DSP_Modules	
ADDER_TREE	1.4
BARREL_SHIFTER	1.3
MAC	4.2
MMAC	2.2
MULT	4.2
MULTADDSUB	4.2
MULTADDSUBSUM	4.2
SLICE	1.6
WIDE_MUX	1.4
Memory_Modules	
Distributed_RAM	
Distributed_DPRAM	3.8
Distributed_ROM	2.7
Distributed_SPRAM	3.8
EBR_Components	
RAM_DP	6.4
RAM_DP_TRUE	7.4
RAM_DQ	7.4
ROM	5.3
FIFO	5.0
FIFO_DC	5.7
RAM_Based_Shift_Register	5.1

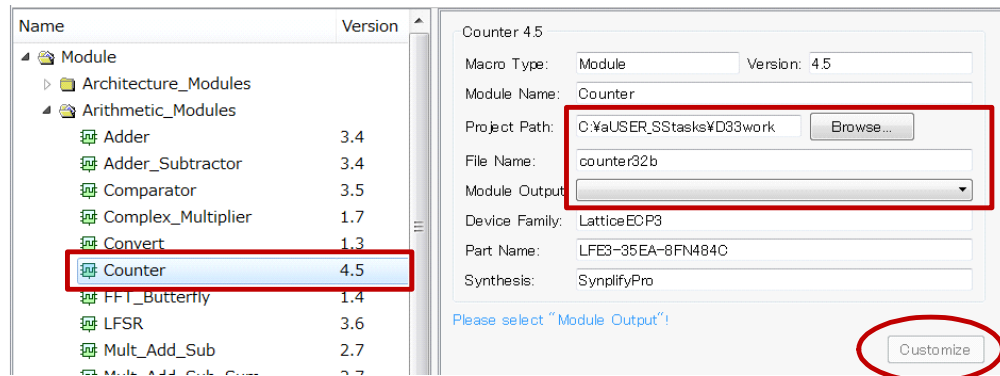
Lattice Diamond 日本語ユーザガイド

Name 枠の下部は IP セクションになっています。ここに表示されるアイテム・構成はデバイス毎に異なり、また後述のようにユーザがダウンロード、インストールする（インストール済み）IP に依存します。図 5-3 はあくまで一例です。

5.2.2 モジュール生成手順

カウンタモジュール生成の具体例を用いて、手順・方法の概要を記述します。

図 5-5. カウンタマクロの生成準備



まず該当するセクションから Counter を選択します。右枠の中に所定の情報が表示されます。ユーザが入力するのは赤枠長方形内、Project Path、File Name、Module Output です。

Project Path はデフォルトでインプリメンテーション・フォルダがロードされています。変更する場合には Browse ボタンをクリックして所望のフォルダを指定します。File Name はモジュールの名称（インスタンス名）です。Module Output は初期表示は図 5-5 のようにブランクですが、行の一部をクリックするとプルダウン形式で HDL 言語を選択できます（図 5-6）。

図 5-6. モジュール記述言語の指定



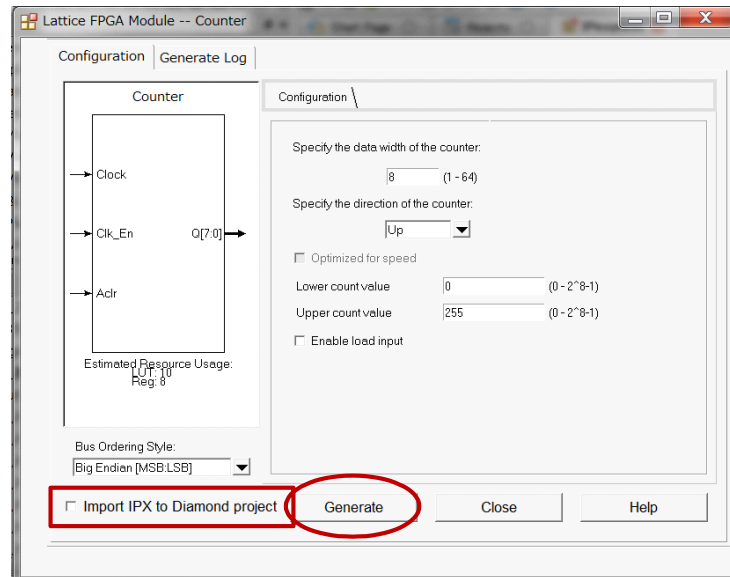
これらを全て入力・指定後、右下の赤丸内に示す『Customize』ボタンをクリックします。いずれかが入力されていないと（図 5-5 のように）グレーアウトのままで次に進めませんので留意します。その後に表示されるウィンドウでモジュールのパラメータ指定を行います。

カウンタの場合、図 5-7 がパラメータ設定 GUI になります。対象モジュールに拘わらず、数値入力セルとプルダウンによるオプションや値の選択のセルに分かれますが、基本的に全て所望の設定を行う必要があります。本例では上部に『Configuration』タブが一つあるのみですが、モジュールによっては複数のタブがありますので、全タブを確認します。

入力・指定完了後、下部の『Generate』ボタンをクリックすると、指定した Verilog もしくは VHDL 言語の RTL ファイルが生成されます（同時に拡張子が .ipc、.ipx、など、付随する複数のファイルも生成されます）。

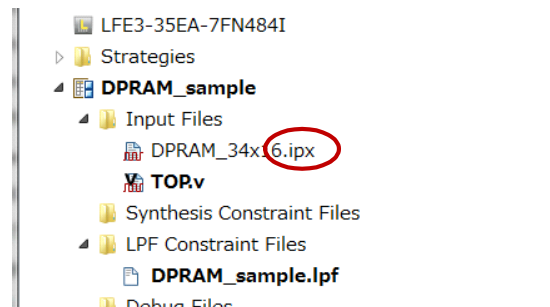
生成された RTL ファイル (<File Name>.v / .vhd) はマニュアルでプロジェクト（インプリメンテーション）に取りこむ必要があります（2.6.2 節を参照してください）。他方、自動で取りこむ方法があります。図 5-7 の下部左に赤枠で示す、[Import IPX to Diamond Project] のラジオボタンにチェックをすることで、<module_name>.ipx という拡張子のファイルが自動的にロードされます（図 5-8）。

図 5-7. Counter のパラメータ設定 (Configuration) GUI



.ipx ファイルはツールが必要なパラメータ情報を全て含む、Diamond 固有のテキストファイルです (ユーザは編集しないこと)。ipx ファイルを取りこむ利点は、パラメータの変更が容易な点があります。プロジェクト進捗に伴い、或いはその他要因で変更する可能性がある場合は、.v / .vhd ではなく、.ipx が良いでしょう。.ipx の当該行をダブルクリックすると、指定済みのパラメータをロードした状態で再び図 5-7 のようなパラメータ設定 GUI が表示されます。他方、インスタンスする場合やソース記述をチェックしたい場合は .ipx は不都合ですので .v や .vhd を参照する必要があります (勿論、論理シミュレーションに .ipx を用いることはできません)。

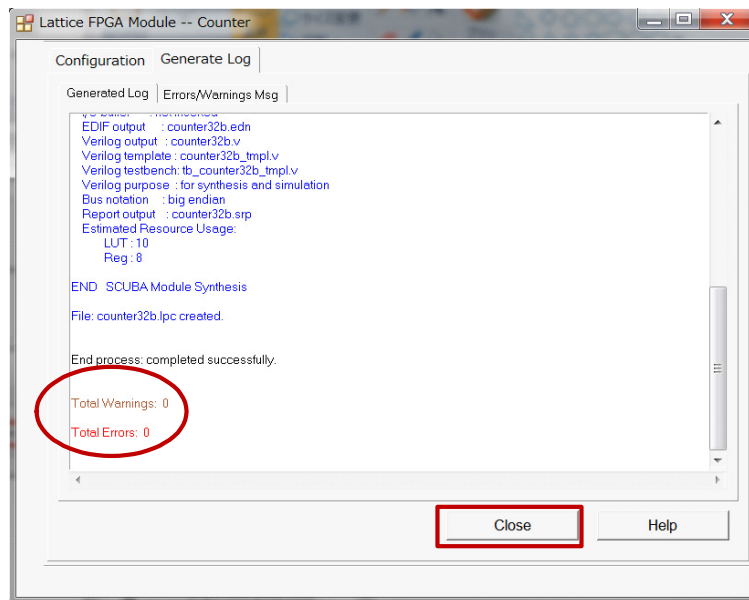
図 5-8. IPX チェック後生成して自動で取り込む例 (File List 枠内 Input Files 部)



プロジェクトには単一モジュールに対して .v / .vhd と .ipx 両方をインポートする事も可能ですが、必ず一方を右クリックして [Exclude from Implementation] 指定するようにします (グレーアウトされる)。

Generate ボタンをクリックした後は、図 5-9 のような GUI が表れます。必ず Total Warnings / Errors が 0 であることを確認後、Close ボタンで終了します。0 でない場合は何らかの問題があります。

図 5-9. モジュール Generate 後の GUI 表示例

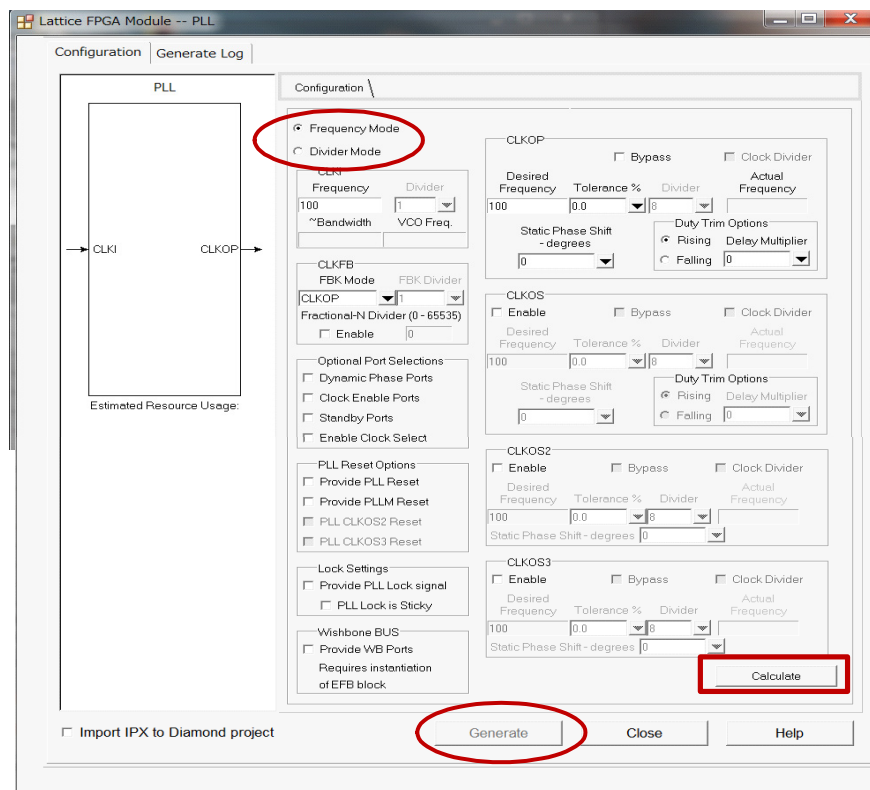


5.2.3 特定のモジュール生成時の留意事項

5.2.3.1 PLL モジュール

PLL モジュールの生成時のパラメータ設定 GUI 例を図 5-10 に示します。

図 5-10. PLL モジュールのパラメータ入力 GUI 例 (MachXO3L)



Configuration タブ表示の直下に二つのパラメータ指定モードがオプションとして選択できます。[Frequency Mode] は入出力クロックの指定セルに周波数を与える場合（デフォルト）、[Divider Mode] は入出力ポートに付随する分周器の値を直接与える場合に選択するものです。通常は前者を使用します。後者を使用する場合は分周値を求める際に十分な知識が求められます。詳細は各デバイスファミリのクロック / PLL 関連テクニカルノートを参照します。また各パラメータ入力セルの項目についての定義等も、同テクニカルノートをご参照ください。

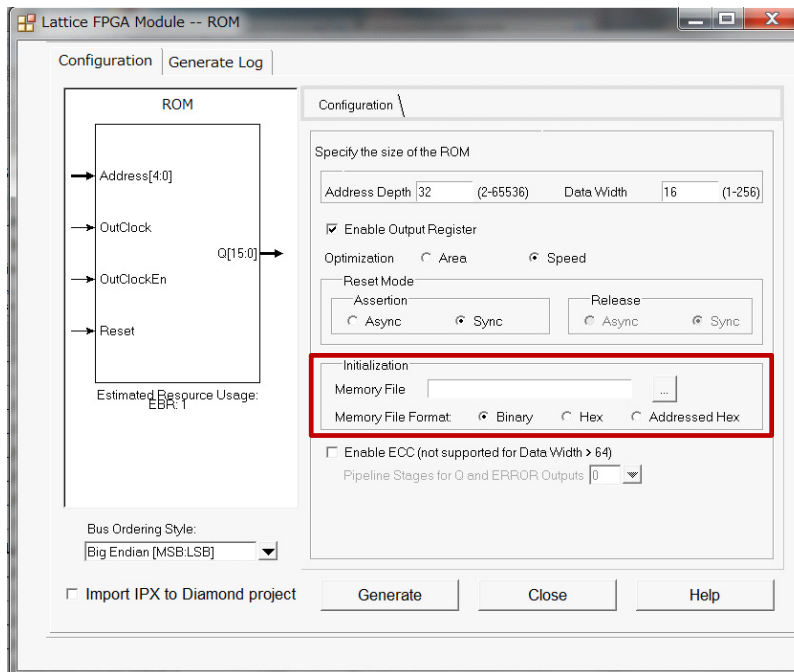
通常のモジュール生成では一通りパラメータを設定し終わると『Generate』ボタンがアクティブになります。PLL に限り、それだけでは不十分で、右下の赤枠で示す『Calculate』ボタンをクリックする必要があります。これによって DRC が実行され、果たして入力通りの周波数や分周値で PLL 規定内動作可能かどうかをツールが確認します。周波数値が所望にならなかつたり（[Tolerance] で指定）、最低・最高周波数範囲に収まらなかつたり、ということがないと検証された場合に限り、Generate ボタンが有効になりクリックできます。無効の場合は、設定内容を再吟味する必要があります。

なお、Calculate 後に左上部 CLKI セクションの下部にある [Bandwidth] と [VCO Freq.] セルに計算結果が表示されます。それぞれ閉ループ帯域幅と VCO 周波数です。

5.2.3.2 ROM モジュール

分散 (Distributed) ROM でも EBR ベースの ROM でも同様の留意事項です。ROM モジュールの生成時のパラメータ設定 GUI 例を図 5-11 に示します。

図 5-11. ROM モジュールのパラメータ入力 GUI 例 (MachXO3L)



ROM の場合はコンフィグレーション時にメモリ内のデータが初期化され、リードアクセスのみが許容されます。従って、一般のモジュールと異なり、ROM モジュール生成時に初期化データを与えてやる必要があります。図 5-11 の赤枠が該当する入力部です。

初期化ファイルは拡張子が "<任意>.mem" とするテキストファイル、という約束があります。ブラウザしてファイルを選択・指定します。ここでテキストファイルの初期値を記述するフォーマットに制限があります。図中にあるように [Binary] か [Hex]、または [Addressed Hex] のいずれかである必要があります。それぞれのフォーマットは、図 5-12 に示すような形式です。詳細は各デバイスファミリのメモリ関連テクニカル

ノートをご参照ください。

なお、前述のとおり、これらを全て入力後 Generate クリックして、モジュールを生成しますが、形式が一致、かつ初期化ファイルに記述されるデータ語長とデータ語数が、GUI 中の [Address Depth] と [Data Width] の値と一致しなければなりません。不一致があると、図 5-9 に示すような生成ステータス表示 GUI で結果がエラー 0 にならず、正常終了しません。そのまま抜けてもモジュールは生成されていません。再度パラメータと初期化ファイルの内容に一貫性があることを確認して再生成することになります。

図 5-12. 初期化 .mem ファイルの書式概要 (左 : Binary、中 : Hex、右 : Addressed Hex)

00100000010000000010000001000000	A001	-A0 : 03 F3 3E 4F
00000001000000001000000010000001	0B03	-B2 : 3B 9F
0000001000000001000000001000000010	1004	
0000001100000001100000001100000011	CE06	
0000010000000010000000100000001000	0007	
	040A	
	0017	
	02A4	

5.2.4 IP の生成準備

Diamond インストール直後はラティス提供の IP が使用できる状態にはなっていないので、初めに意図する IP を（個別に）ダウンロードしてインストール、その後パラメータ設定を行い、生成します。

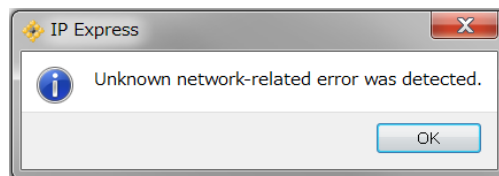
まず、ダウンロードするためにラティスの IP サーバにアクセスします。🌐 をクリックし（図 5-13、左）、次に表示される "IP (Click to get IP information)" をクリックします（同図、右）。

図 5-13. ラティス IP のダウンロード



なお、ここで PC 環境はネットワークにアクセスする必要があります（問題があるとメッセージが表示されます：図 5-14）。



図 5-14. ネットワーク接続に問題がある場合のメッセージ



IP サーバが応答後は図 5-15 左のように、ターゲットとしているデバイスに対応する IP のリストがバージョン情報と共に表示されます（グレー表示の（古い）バージョンは対象としないことを推奨します）。カテゴリーは 5 つに分類されています。"Connectivity" には PCI Express や SDI (SMPTE) などが、"DSP" には FIR Filter / Scaler / Viterbi / CFC Filter / CORDIC / CSC / FFT / NCO などが、"Processors, Cont..." には DDR2/3 Controller などが含まれています。

図 5-15. IP サーバー応答後の表示例 (左: 表示直後、右: カテゴリ理解のために操作後)

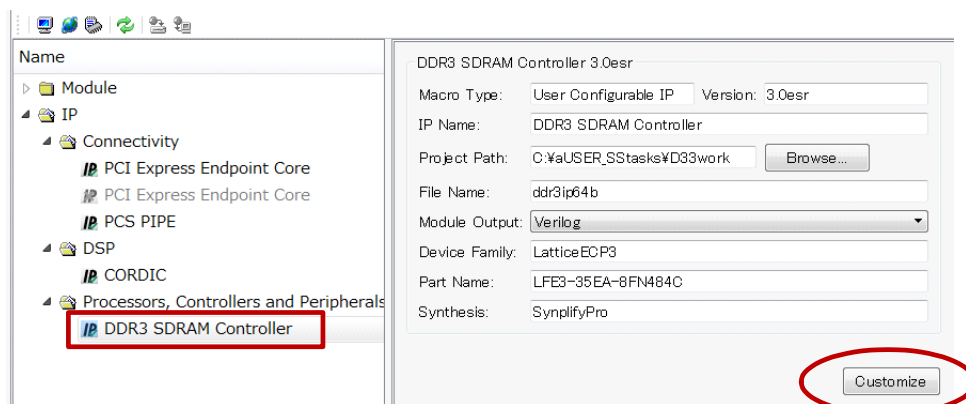
Name	Version	Name	Version
IP		IP	
Communications		Communications	
10Gb+ Ethernet MAC	4.3	Connectivity	
10Gb+ Ethernet MAC	4.4	DSP	
10Gb+ Ethernet MAC	4.5	Platform Manager Support	
2.5G Ethernet MAC	2.0	Processors, Controllers and Peripherals	
2.5G Ethernet MAC	2.1	DDR SDRAM Controller CP	1.0
2.5G Ethernet PCS	1.0	DDR SDRAM Controller CP	1.1
2.5G Ethernet PCS	1.1	DDR SDRAM Controller CP	1.2
CPRI	3.3	DDR SDRAM Controller	6.10
CPRI	3.4	DDR SDRAM Controller	6.11
SGMII Gb Ethernet PCS	3.3	DDR SDRAM Controller	6.8
SGMII Gb Ethernet PCS	3.4	DDR SDRAM Controller	6.9
SGMII Gb Ethernet PCS	3.5	DDR2 SDRAM Controller CP	1.0
SGMII Gb Ethernet PCS	3.6	DDR2 SDRAM Controller CP	1.1
SPI4	2.10	DDR2 SDRAM Controller CP	1.2
SPI4	2.7	DDR2 SDRAM Controller	7.0
SPI4	2.8	DDR2 SDRAM Controller	7.1
SPI4	2.9	DDR2 SDRAM Controller	7.2
Tri-Speed Ethernet MAC	3.4	DDR2 SDRAM Controller	7.3
Tri-Speed Ethernet MAC	3.5	DDR2 SDRAM Controller	8.0
Tri-Speed Ethernet MAC	3.6		

IP をダウンロードするには、所望の IP / バージョン行を選択して  アイコンをクリックするか、選択行を右クリック後表示される "Download <IP 名、バージョン番号>" を選択します。この場合は、ダウンロードしたパッケージを元に、手動でインストールする必要があります。ダウンロードとインストールを連続して自動で行うには  アイコンをクリックするか、選択行を右クリック後表示される "Install <IP 名、バージョン番号>" を選択します。ダウンロードのみを選択した場合は、最初にダウンロード先のフォルダ選択を促す表示が出ますので指定します。デフォルトの "C:\%LatticeCore" が適切 (推奨) です。ダウンロード・ファイルは実行形式 "<IP 名、バージョン番号>.exe" のようになっていますので、ダブルクリックしてインストーラを実行します。

5.2.5 IP のパラメータ設定と生成

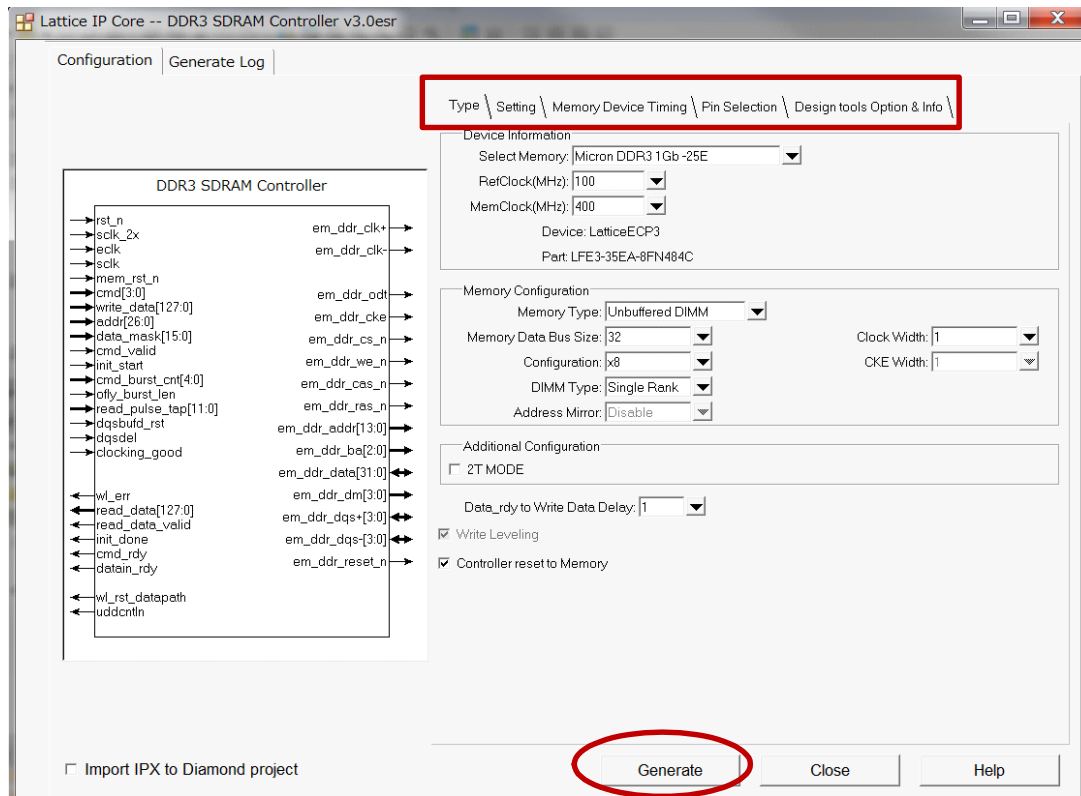
DDR3 メモリコントローラ IP の具体例を示すことで、手順・方法の概要を記述します。まず左枠 Name 内で当該 IP 行を選択し、右枠に図 5-5 と同様の要領で必要な入力 (Project Path、File Name、Module Output) を完了後、『Customize』ボタンをクリックします。

図 5-16. DDR3 メモリコントローラ IP の生成準備



次に表示されるのがパラメータ入力・設定用 GUI です。これは勿論 IP ごとに異なります。DDR3 メモリコントローラの場合は図 5-17 のようになります。

図 5-17. DDR3 メモリコントローラ IP のパラメータ設定 GUI



モジュール生成時と同様に、各セルに適宜入力した後、**Generate** ボタンをクリックします。GUI 上部の赤枠のように、IP によっては複数のタブがありますので、留意します。また、前出と同様に『Import IPX to Diamond Project』ボックスがありますので、チェックすることで .ipx ファイルを自動的にインポートできます。

IP の場合、**Generate** をクリック後、図 5-9 と同様に "Total Errors 0" が表示されますが、それまでに或る程度の処理時間を必要とします。処理途中で『Close』ボタンをクリックしないように注意します。誤ってクリックした場合は、再度手順を最初から繰り返す必要があります。時間が掛かるのは、バックグラウンドで論理合成ツールが呼び出されて実行されているためです。処理時間は IP やパラメータに依存して変わります。

なお、図 5-16 では右枠最下段の [Synthesis] 部表示が "Synplify Pro" になっています。プロジェクト生成時やインプリメンテーションの設定時に選択して有効になっているツールがデフォルトで表示されます。論理合成ツールの選択等については第 6 章を参照してください。

また、IP によってはパラメータ設定 GUI 表示の中に、オプション項目として使用する（可能性のある）論理合成ツールを指定するセクションやタブがあります。その場合は適宜指定します。

5.3 Clarity Designer

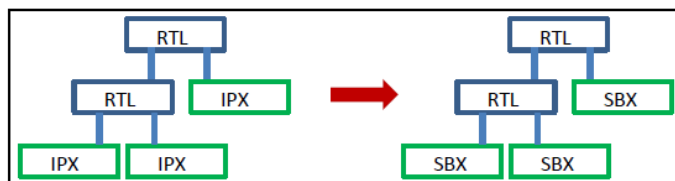
（本日本語ドキュメント自体は Diamond 3.2/3.3 用ですが、本節のみについては Diamond 3.4 の Clarity Designer を元に記述しています。ご注意ください。）

5.3.1 Clarity Designer と IPexpress の対比

前節で記述したように、IPexpress が生成するモジュールの出力ファイルは Diamond 固有のフォーマットである <module_name>.ipx と HDL (<module_name>.v / .vhd) 等です。これに対し、Clarity Designer では <sub_system_name>.sbx と HDL 等です。

Clarity Designer ではやや上位の概念から ”サブシステム” という用語を用います。単一の .sbx ファイルにはユーザ生成 HDL を含む複数のモジュール (又はコンポーネント) が存在することが許容されます。図 5-19 のように .sbx がトップモジュールとして構成できることが最終ゴールです。

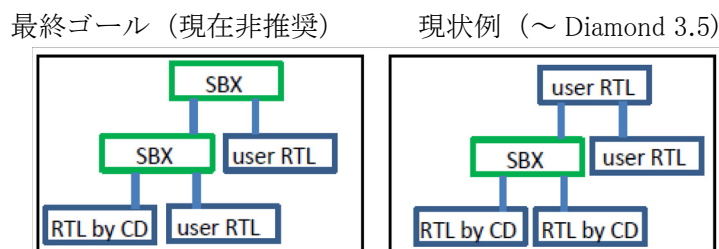
図 5-18. Clarity Design ベース設計の IPexpress モジュール置き換え構成例



IPexpress で生成したモジュールを用いた実装回路の典型的な構造例は図 5-18 の左の通りです。トップモジュール配下に .ipx や HDL 記述のユーザモジュールが階層を構成しています。図 5-18 の右のように Clarity Designer でも ipx (ファイル) の置き換え、或いは同様の構成をとることが可能です。

最終的には図 5-19 の左も可能にすることですが、Diamond 3.5 時点では右のようにすることが現実的です。

図 5-19. Clarity Design ベース設計の ”サブシステム” 的構成例 (図中 ”CD” は Clarity Designer を示す)



5.3.2 下位モジュール vs. トップモジュール

5.3.2.1 下位モジュールとしての使用

Clarity Designer 生成モジュール (.sbx) を下位モジュールとして使用する場合 (図 5-18、右) の留意点は以下の通りです。基本的に IPexpress と同様です。

- ・ 各 sbx モジュールはユーザ記述 HDL 内でインスタンスする
- ・ 複数の sbx を単一の HDL で、或いは複数の HDL でインスタンス可能
- ・ sbx ファイルには配置指定情報含むことも可能 (”Planning”)

5.3.2.1 トップモジュールとしての使用

Clarity Designer 生成サブシステム (.sbx) をトップモジュールとして使用する場合 (図 5-19) の留意点は以下の通りです。

- ・ ユーザ HDL は全てモジュールとしてインポートする
- ・ Clarity Designer サブシステム内で全ての IP、モジュールを sbx 内でインスタンスする
- ・ Clarity Designer サブシステム内で全てのモジュール間の接続を行う
- ・ トップの sbx、およびユーザ HDL 内は sbx を含むことはできない (sbx はトップのみ)

以上のように、設計の柔軟性やツールとしての成熟度の観点から、**.sbx をトップとして据える構成は Diamond 3.5 時点でも推奨しません**。従って本ユーザガイド（の本節）に記述はありません。バージョンアップで順次機能拡充がなされる予定ですので、それに応じて次版以降にて追記していきます。

5.3.3 Clarity Designer で可能な作業

Clarity Designer で可能なタスクは以下の通りです（詳細は後述します）。

- ・ モジュールや IP のパラメータ設定
- ・ IP のダウンロードとインストール
- ・ IPexpress ファイル .ipx をインポートして .sbx に変換
- ・ HDL をインポートしてモジュールとして使用
- ・ サブシステムとして下位モジュール相互を接続指定（Builder タブ）
- ・ PCS/SERDES や DDR エLEMENTの配置指定（Planner タブ）

他方、以下の作業はできませんので留意ください。

- ・ 新規 HDL のエントリー
- ・ .lpf（制約ファイル）の編集
- ・（PCS / DDR 以外の）汎用 IO の配置指定（従来通りスプレッドシート・ビューか lpf ファイルで）

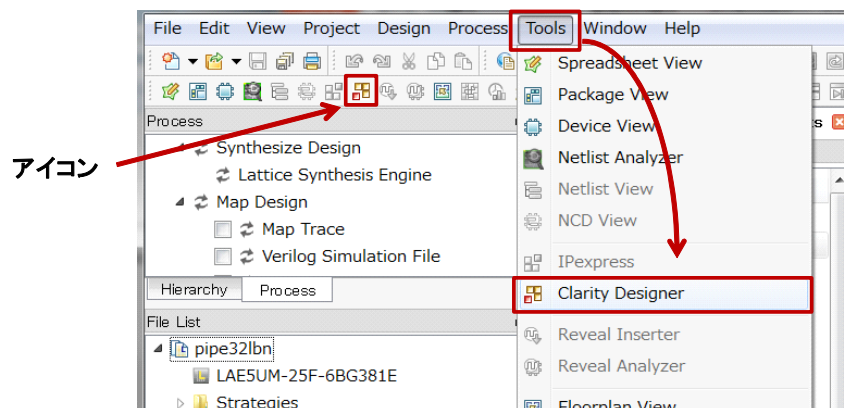
また、Clarity Designer 対応のプロジェクト / インプリメンテーションに関する留意点は以下です。

- ・ 単一 sbx ファイルには複数のモジュールが存在可能
- ・ 複数の PCS、複数の DDR モジュールはそれぞれ単一の sbx に含めるようにする（配置指定を正しく処理させるため）
- ・ File List 中に ipx ファイルは使用不可
- ・ sbx ファイルをインポートする前に必ず『Generate』する
- ・ sbx のパラメータ等を編集・変更した場合は必ず『再度 Generate』する

5.3.4 Clarity Designer の起動

FIFO_DC モジュール生成の例を用いて手順概要を記述します。

図 5-20. Clarity Designer の起動



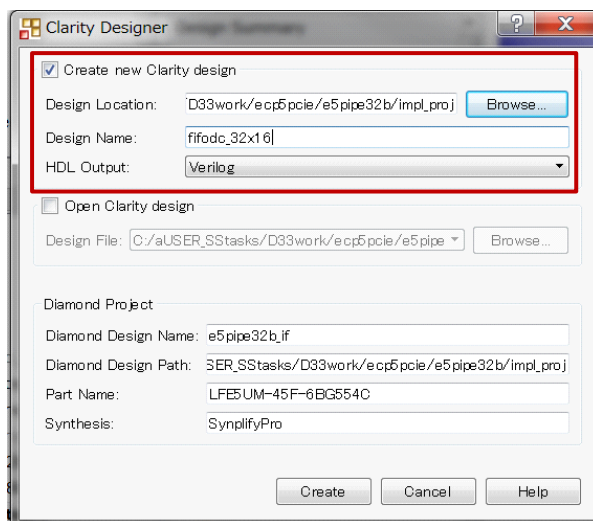
Clarity Designer の起動は IPexpress 同様です。アイコン（IPexpress の隣）をクリックするか、Tools->Clarity Designer と辿ります（Generate 後、File List に取り込まれる .sbx 行をダブルクリックしても立ち上がります）。

初めに立ち上がるのは IPexpress とやや異なり、図 5-21 のようなウィンドウです。Clarity Designer ではサブシステムと呼ぶ、モジュール（又はコンポーネント）より上位の概念を扱うことができます。本ウィンド

ここでは [Design Name] 欄に ”サブシステム名” のような意味での名称を入力します。次ステップ以降のモジュール生成を繰り返すことで、複数のモジュールをこのサブシステム配下を含めることが可能です。

新規の場合は赤枠内を選択・記述します。入力方法は基本的に IPexpress と同様です(下部の Diamond Project セクションが正しいことを確認します)。完了後、『Create』ボタンをクリックします。

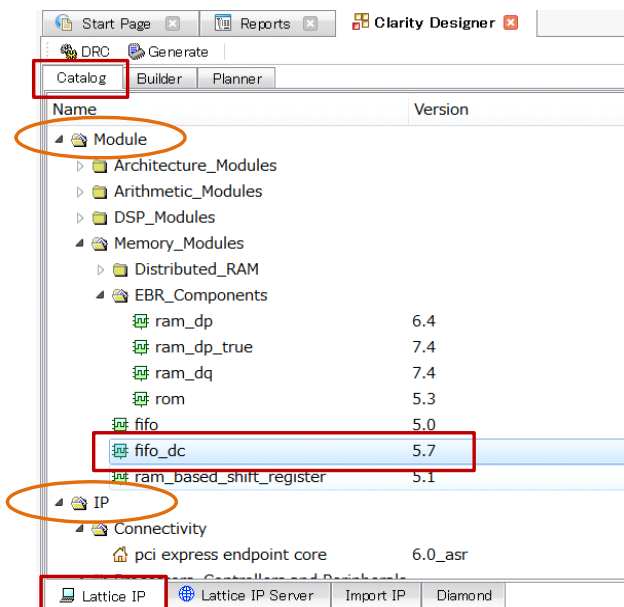
図 5-21. 初期画面：サブシステム名の入力



5.3.5 モジュールのパラメータ設定 (Configuration)

次に表示されるのが図 5-22 のようなウィンドウです。

図 5-22. カタログタブ (Module 部各セクションを畳んだ後の例)



IPexpress とほぼ同じ画面ですが、上部には三つのタブ、『Catalog』と『Builder』、および『Planner』があり、『Catalog』タブが選択された状態で立ち上がります。”Module”部は IPexpress と同様です。その下には IP 部があり、その時点でインストール済みの IP リストが表示されます。

ここで、”コンフィグレーション (Configuration)” という語は、パラメータ設定、或いはオプション設定をすることを意味します。本章では同じ意味で ”パラメータ設定” と既述します。『Catalog』タブでモジュールや IP のパラメータ設定を行います。これは IPexpress と同様です。

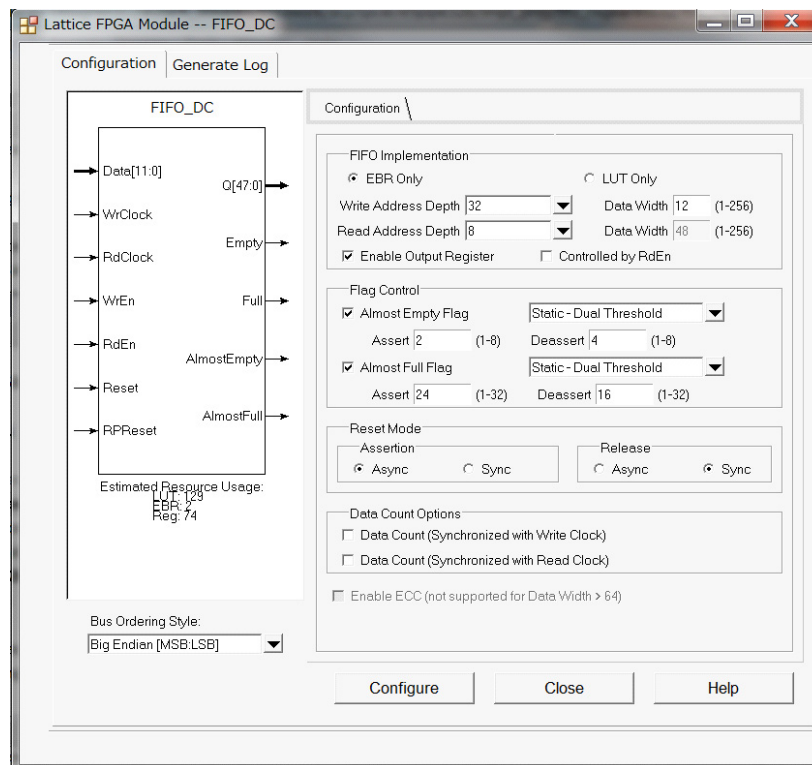
下部には四つのタブがあり、『Lattice IP』タブが選択された状態です。新たにラティスの IP サーバーから IP をダウンロード・インストールする場合は、IPexpress と全く同じ要領で『Lattice IP Server』タブを選択して作業します (図 5.2.4 節参照)。

図 5-23. インスタンス名入力ウィンドウ



例として ”fifo_dc” モジュール行をダブルクリックします。図 5-23 のようなウィンドウが表示されますので、インスタンス名を ”Instance Name” 欄に入力後、『Customize』ボタンをクリックします。図 5-24 のようなパラメータ設定ウィンドウが立ち上がります。

図 5-24. FIFO_DC モジュールの生成 GUI



IPexpress との第一の違いは GUI 下部にあるボタンで、『Generate』ではなく『Configure』です（Generate は後述するビルドステップで行います。）。第二の違いは IPexpress で GUI 左下にあった [Import IPX to Diamond Project] のようなオプションがない点です。Clarity Designer では『Generate』後に sbx ファイルが自動的に取り込まれるためです。

既述のように、Clarity Designer 特有の機能で、サブシステム配下に複数のモジュールが取り込めます。複数インスタンスする場合は、都度『Catalog』タブを選択して、同じ手順を繰り返します。

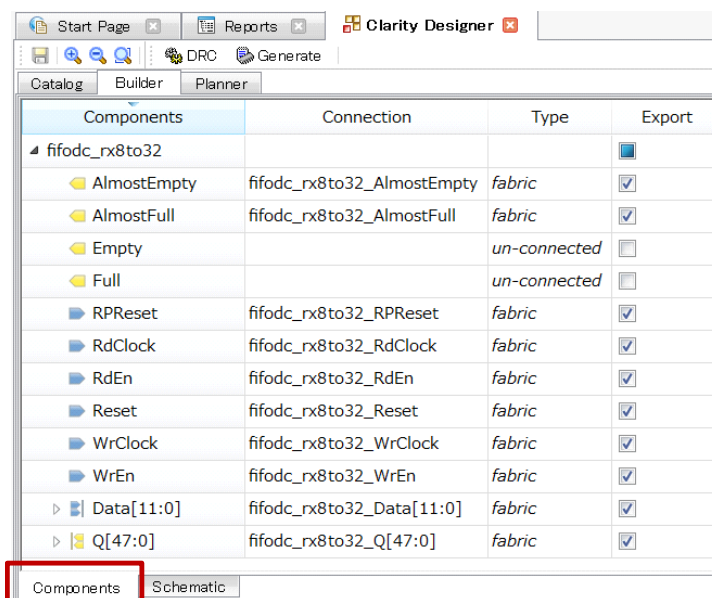
5.3.6 ビルド

『Builder』タブで行う作業の一つは、GUI 下部の『Components』タブを選択して、コンポーネントのポートをサブシステム（上位）に引き出すかどうか（、及びコンポーネント間の接続）です。IPexpress との整合性もあり、デフォルトではコンポーネントの全ポートは [Export] セルがチェックされていて、上位に引き出されています（Type = “fabric”）。その必要がない場合はチェックボックスをクリックして [Type] 欄を “un-connected” とします。

Type 欄にはこれ以外に “internal” と “pad-pin” があります。前者は『Planner』タブでの接続指定、後者はデバイスのパッケージ I/O との接続を意味します。なお、ポートを 0/1 固定レベルに接続することはできません。また、論理式（回路）と接続することもできません。[Connection] 列のセルはユーザが任意に名称を変更できます。

図 5-26 に示す例は、コンポーネントのポート同士を接続する場合です。

図 5-25. FIFO_DC モジュールの接続設定例



”Control” キーを押しながらクリックすることで複数の信号を選択し、そのいずれかの上でマウス右クリック後に [Connect] を選択します。同一のソース信号に接続する複数の入力信号は何本でも同時に選択できますが、出力信号は一本のみです。この場合、入力信号を一本のみ ”Export” として他を un-connected にしないと ”Connect” アクションが選択できません。一本を Export する場合、出力信号は必ずしも選択しなくて構いません。

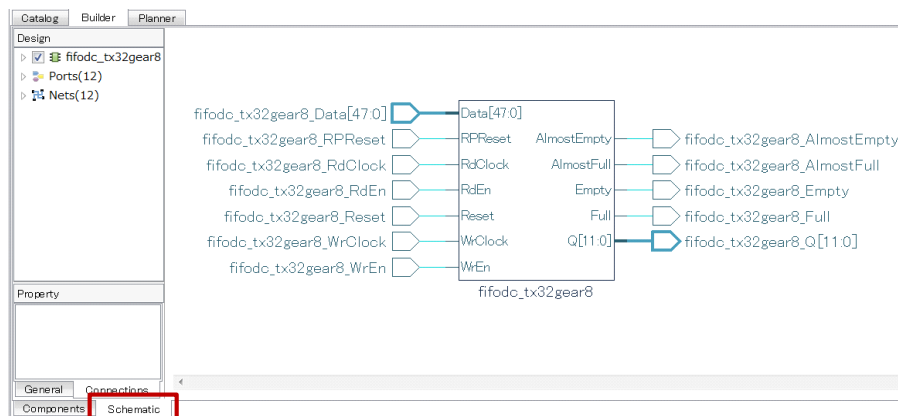
”Connect” されると、”un-connected” 指定した入力ポートも Type 表示が ”fabric” に変わります。接続を解除するには、再度右クリックして [Disconnect] を選択します。

図 5-26. 入力ポート間を接続する例

Components	Connection	Type	Export
AmmosEmpty	fifodc_tx32w_AmosEmpty	fabric	<input type="checkbox"/>
AlmostFull	fifodc_16x16_AlmostFull	fabric	<input checked="" type="checkbox"/>
Empty	fifodc_16x16_Empty	fabric	<input checked="" type="checkbox"/>
Full	fifodc_16x16_Full	fabric	<input checked="" type="checkbox"/>
RPRReset	fifodc_32x16w_RPRReset	fabric	<input type="checkbox"/>
RdClock	fifodc_32x16w_RdClock	fabric	<input type="checkbox"/>
RdEn	fifodc_32x16w_RdEn	un-connected	<input type="checkbox"/>
Reset	fifodc_32x16w_Reset	fabric	<input checked="" type="checkbox"/>
WrClock	fifodc_32x16w_WrClock	fabric	<input checked="" type="checkbox"/>
WrEn	fifodc_32x16w_WrEn	fabric	<input checked="" type="checkbox"/>
Data[15:0]	fifodc_16x16_Data[15:0]	fabric	<input checked="" type="checkbox"/>
Q[15:0]	fifodc_16x16_Q[15:0]	fabric	<input checked="" type="checkbox"/>
fifodc_32x16w			<input checked="" type="checkbox"/>
AlmostEmpty	fifodc_32x16w_AlmostEmpty	fabric	<input checked="" type="checkbox"/>
AlmostFull	fifodc_32x16w_AlmostFull	fabric	<input checked="" type="checkbox"/>
Empty	fifodc_32x16w_Empty	fabric	<input checked="" type="checkbox"/>
Full	fifodc_32x16w_Full	fabric	<input checked="" type="checkbox"/>
RPRReset	fifodc_32x16w_RPRReset	fabric	<input checked="" type="checkbox"/>
RdClock	fifodc_32x16w_RdClock	fabric	<input checked="" type="checkbox"/>
RdEn	fifodc_32x16w_RdEn	fabric	<input checked="" type="checkbox"/>
Reset	fifodc_32x16w_Reset	fabric	<input checked="" type="checkbox"/>
WrClock	fifodc_32x16w_WrClock	fabric	<input checked="" type="checkbox"/>
WrEn	fifodc_32x16w_WrEn	fabric	<input checked="" type="checkbox"/>

下部の『Schematic』タブをクリックすると、図 5-27 のようにコンポーネントレベルの I/O ポート図が表示されます。単一コンポーネントの場合は、このウィンドウは閲覧（確認）用のみです。

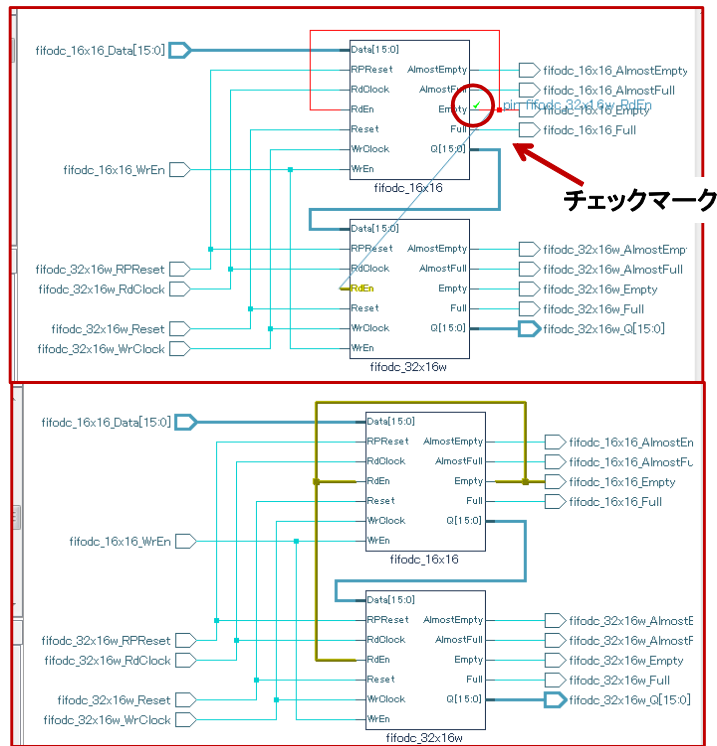
図 5-27. Builder => Schematic タブの表示例



複数コンポーネントがある場合には、Schematic ウィンドウ上でドラッグ & ドロップによって接続指定が可能です。

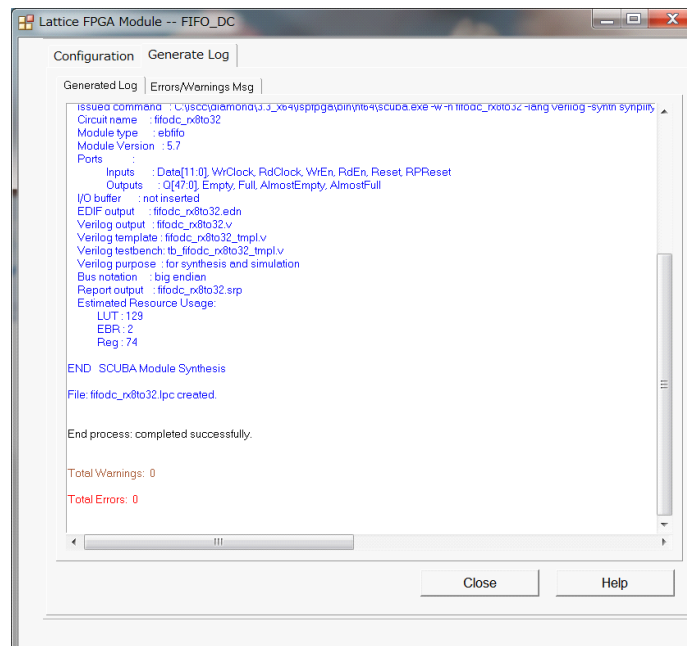
入力ポート同士や入力・出力ポート間を接続する場合、まず『Components』タブで一方を "un-connected" 指定します。次に『Schematic』タブを選択し、意図する一方のポートを選択し（クリックを離す）、接続先のポートにカーソルを移動します。許可される接続の場合は配線が赤色となり、同時に（わかりにくいですが）緑色（水色）のチェックマークがポート近辺に表示されます。良ければ赤色に変わった配線の上で左クリックして確定します。接続情報が描画され直して表示されます。左クリックだけでは接続指定のアクションが継続していますので、他とも接続する場合は繰り返します。終了する場合には、左クリックに続いてさらに右クリックします（図 5-28 下）。

図 5-28. Schematic ウィンドウで接続指定する例



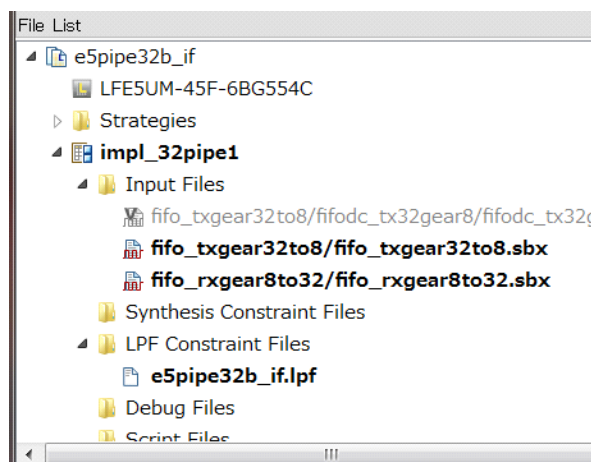
以上、ポート接続の設定を完了したら、Catalog / Builder / Planner タブ表示の上にある『Generate』ボタンをクリックしてモジュールを生成します。

図 5-29. モジュール生成後の GUI 例



IPexpress 同様にモジュール生成が正常に終了したことを確認してから Close します。正常に生成後は sbx ファイルが自動的に取りこまれ、File List 部に表示されます。

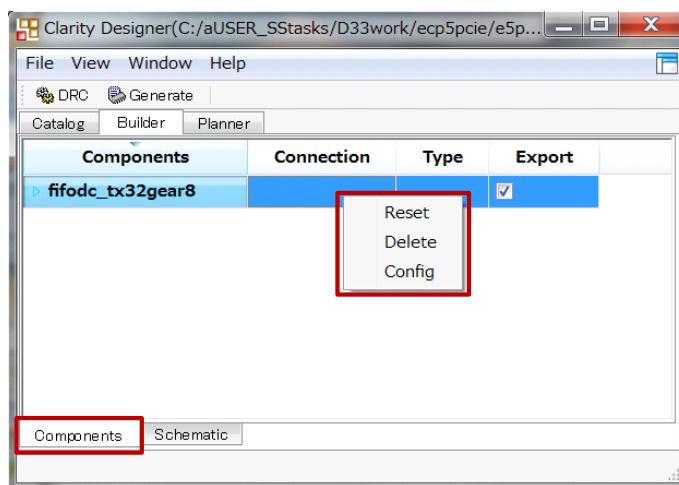
図 5-30. モジュール生成後の File List セクション例



一度生成した後に、sbx ファイルのパラメータを変更する場合は、まず Input Files 内の当該 sbx 行をダブルクリックします。図 5-31 のようなウィンドウが表示されます。

変更したい当該コンポーネント行を選択して (本図では一つしかない) 右クリックすると、三つのアクションが選択できます。Reset はパラメータを初期状態に戻し、Config を選択すると図 5-24 のような生成時にパラメータを設定した状態で Clarity Designer が立ち上がります。適宜変更・修正して再度 Generate します。Generate しないと変更は有効になりませんので留意して下さい。

図 5-31. sbx ダブルクリック後の GUI 表示例



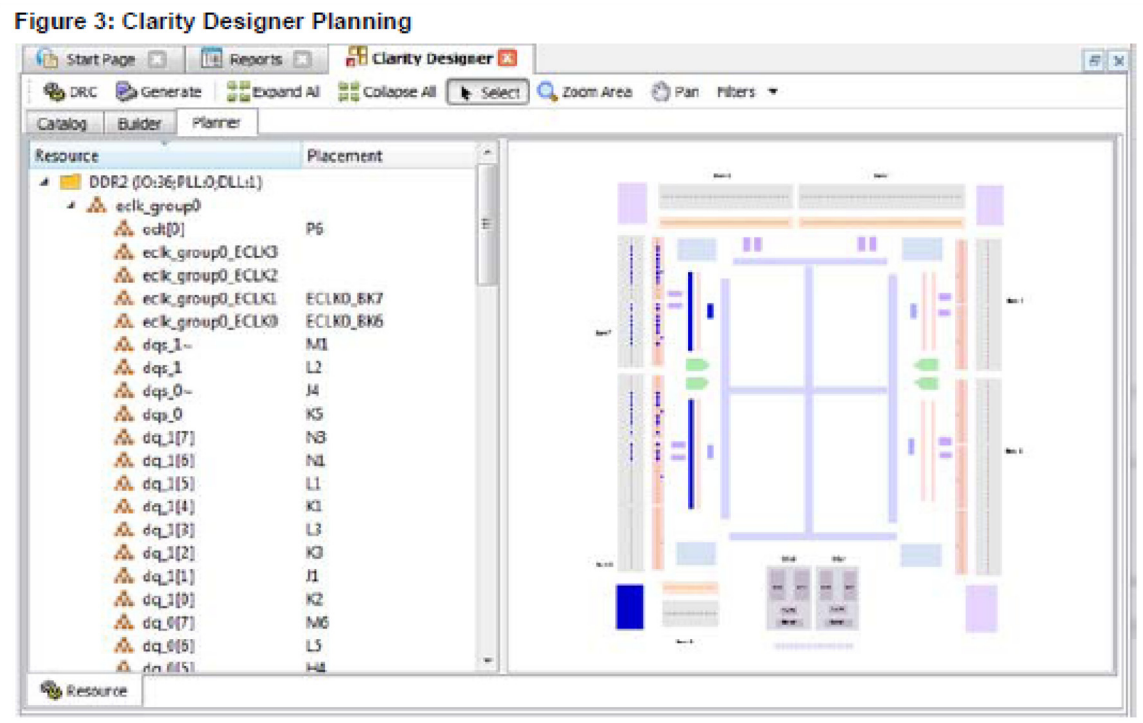
5.3.7 プランニング

プランニングタブでは、特定のデザインエレメントをドラッグ&ドロップ形式でターゲットとして選択しているチップのリソースに配置指定 (割り当て) することができます。Diamond 3.4 時点で (IP 関連は) ECP5 の PCS/SERDES (DUC) と Generic DDR および DDR3/LPDDR3 メモリインターフェイスのみが対応しています。

前節のように、デバイス・アーキテクチャ固有のマクロ (モジュール) 生成ではプランニングは必須ではなく、ビルド作業後に "Generate" して問題ありません。他方、ここに示す PCS や IP ではプランニングしないと Generate できませんので、ご留意ください。"DRC" をクリックして確認できます。

詳細フローはそれぞれの例を扱う 5.3.9 節、5.3.10 節、5.3.11 節で記述します。

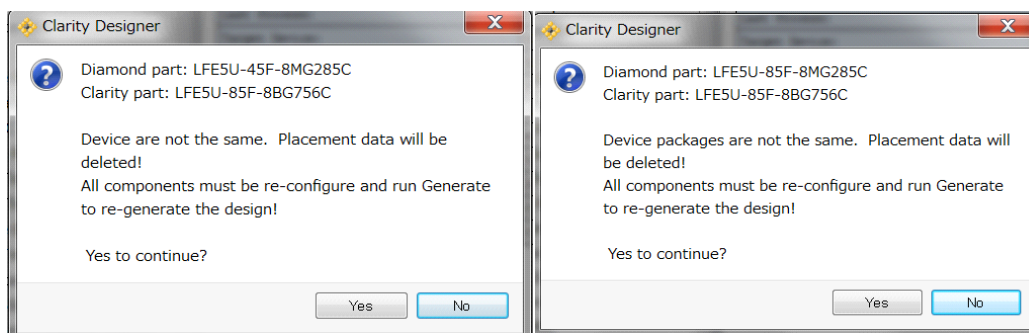
図 5-32. プランニング作業中の例 (DDR2 メモリコントローラ)




【重要】 Clarity Designer 生成ファイル sbx はターゲットとするデバイスサイズ (LUT 規模)、およびパッケージに依存する情報を含んでいます。サブシステム生成後、或いはプロジェクト (インプリメンテーション) を再利用したり、変更する場合、或いは既存 sbx ファイルをインポートした場合などに、デバイスやパッケージが異なると、図 5-33 のようなメッセージが表示されます (配置 (placement) 指定してなくても同様)。再度 "Generate" し直す必要がありますので、ご注意ください。

同様にインプリメンテーション (プロジェクト) で論理合成ツールの指定を変更した場合も、全てのサブシステム .sbx を再度 Generate する必要がありますので、ご注意ください。

図 5-33. デバイスサイズ (左) やパッケージ (右) を変えた場合のウォーニング例



5.3.8 ビルド / プランニング後のリソース情報

ビルド / プランニングを終え Generate すると、図 5-34 のように『Resource』タブのあるウィンドウ枠内にハードウェア・リソースの使用・割り当て情報が表示されます。右上のアイコン  をクリックすると

ウィンドウがデタッチできますので (図 5-35)、拡大表示して確認します。図 5-35 は ”DDR_Generic” サブシステムの例です。

図 5-34. リソース表示ウィンドウ枠

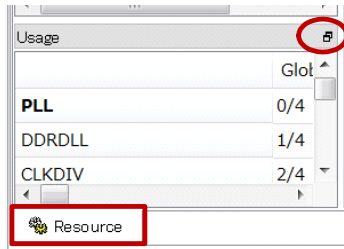


図 5-35. デタッチ状態のリソース表示ウィンドウ

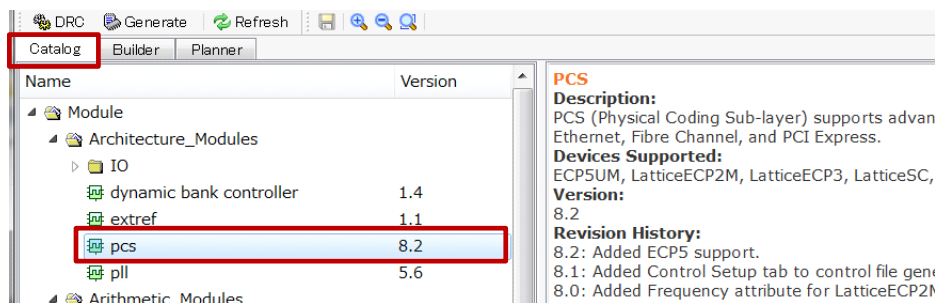
Usage	Global	Bank0	Bank1	Bank2	Bank3	Bank6	Bank7	Bank8
PLL	0/4							
DDRDLL	1/4							
CLKDIV	2/4							
ECLKBRIDGE	0/2							
Edge Clocks				1/2	0/2	0/2	1/2	
Number of IOs	20/245	0/32	0/40	10/32	0/48	0/48	10/32	0/13
Number of Differential Pins		0/16	0/20	5/16	0/24	0/24	5/16	0/6
TRUE LVDS Outputs				10/16	0/24	0/24	6/16	
VCCIO of the Bank				2.5			1.2,1.35,1.5,1.8,2.5,3.3	
DQS Groups				0/2	0/3	0/3	0/2	
PCLK Pins		0/2	0/2	0/2	0/2	0/2	1/2	
Dedicated PLL Input Pins		0/3	0/3	0/3	0/3	0/3	1/3	
Virtual VCCIO Pins		0/4	0/6	0/4	0/6	0/6	1/4	

5.3.9 プランニング例 1 ~ PCS (SERDES)

本節では SERDES サブシステムのプランニング (配置指定) を例に手順を既述します。サブシステムには二つのモジュール、PCS と exref (外部基準クロック入力マクロ) を含むものとします (一般的な構成)。

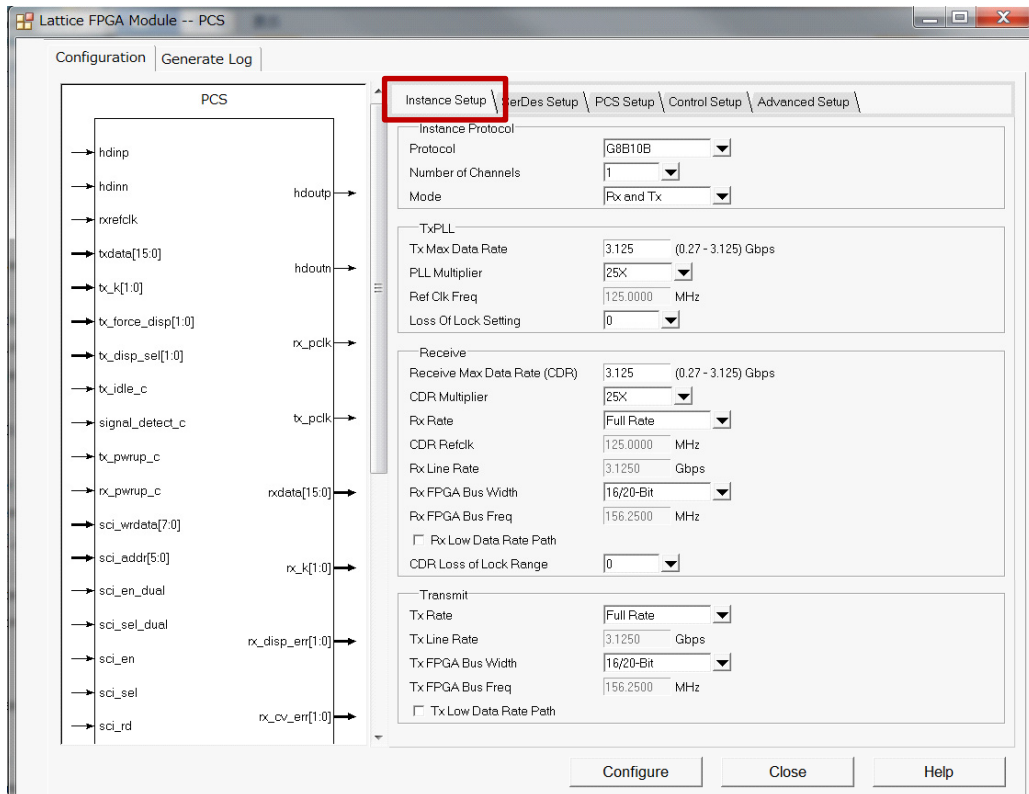
まず Clarity Designer を起動し、サブシステム名等の初期設定を行います (図 5-21 参照)。次に PCS モジュールを生成します。図 5-36 に示すように [Catalog] タブで [pcs] を選択します (ウィンドウ下部のタブは [Lattice IP])。

図 5-36. PCS モジュールの生成開始



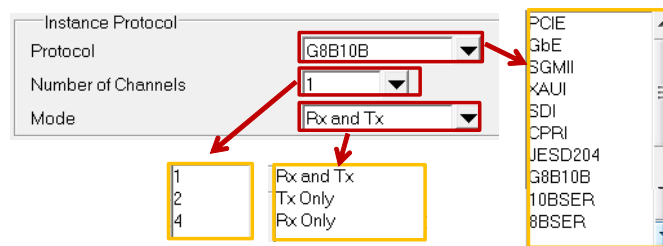
ダブルクリックすると図 5-37 が立ち上がります。上部にタブが五つありますが、該当する所は全て設定します。その中でも基本設定は [Instance Setup] タブの内容で、その上部を抜き出したものが図 5-38 です。

図 5-37. PCS コンフィグレーション ~ Instance Setup タブ



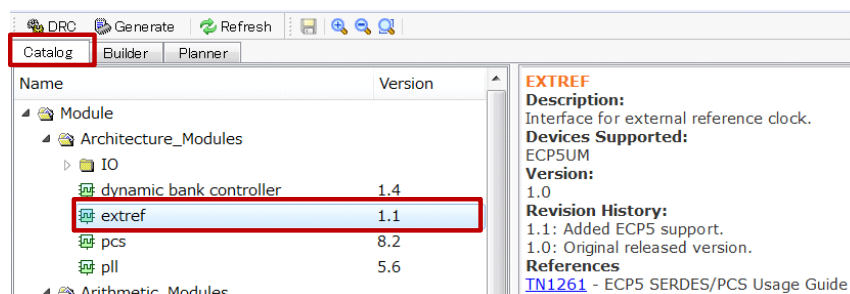
ここで対応プロトコルとチャネル数、送受信対応等を指定します。選択するプロトコルに従って、その他タブの表示や選択できるパラメータ等が変わります。各パラメータの詳細説明は ECP5 のテクニカルノート TN1261 をご参照ください。

図 5-38. PCS の基本設定 (Instance Setup の上部)



次に同様にして extref モジュールのコンフィグレーションを行います。図 5-39 のように Catalog タブで extref 行をダブルクリックします。

図 5-39. extref モジュールの生成開始



本モジュールはパラメータが少ないため、詳細説明は割愛します。両モジュール（コンポーネント）をコンフィグレーション後の [Builder] タブは図 5-40 のようになります（モジュールのポート名表示行を展開した後）。

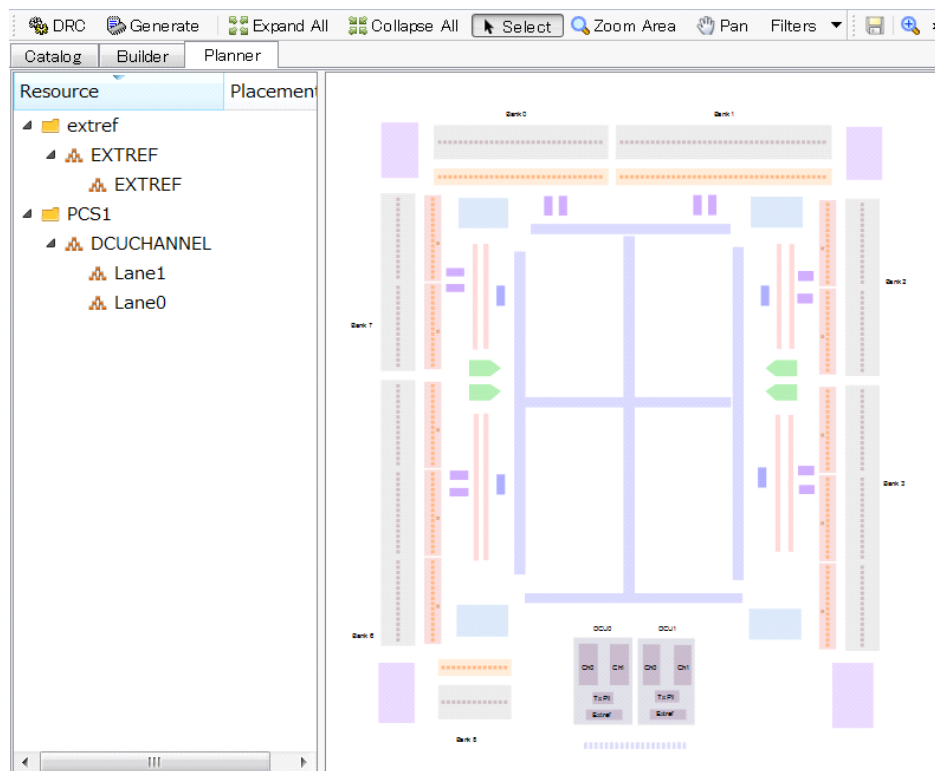
図 5-40. PCS と extref をコンフィグレーション後のサブシステム (Builder タブ)

Components	Connection	Type	Export
extref			<input checked="" type="checkbox"/>
refclkn	extref_refclkn	pad-pin	<input checked="" type="checkbox"/>
refclko	extref_refclko	fabric	<input checked="" type="checkbox"/>
refclkp	extref_refclkp	pad-pin	<input checked="" type="checkbox"/>
PCS1			<input checked="" type="checkbox"/>
cyawstn_dual0		un-connected	<input type="checkbox"/>
lsm_status_ch0_s		un-connected	<input type="checkbox"/>
lsm_status_ch1_s		un-connected	<input type="checkbox"/>
pll_lol	PCS1_pll_lol	fabric	<input checked="" type="checkbox"/>
pll_refclki		un-connected	<input type="checkbox"/>
rsl_disable	PCS1_rsl_disable	fabric	<input checked="" type="checkbox"/>
rsl_rst	PCS1_rsl_rst	fabric	<input checked="" type="checkbox"/>
rst_dual_c	PCS1_rst_dual_c	fabric	<input checked="" type="checkbox"/>

5.3.6 節で既述したとおり、各ポートをファブリックや外部ピンに引き出すかどうかの指定や、モジュール（コンポーネント）間の接続指定などを行います。

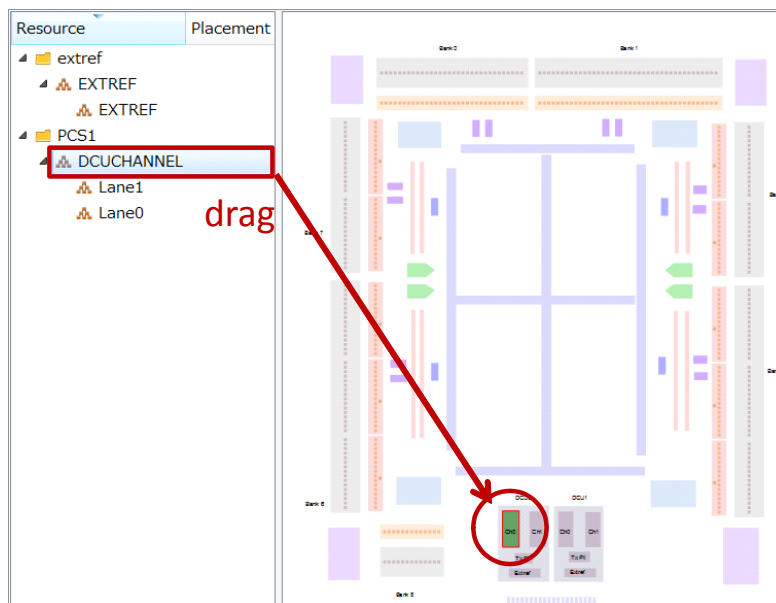
[Planner] タブをクリックすると、初期表示は図 5-41 のようになります（各リソース名の行を展開して表示した後）。[Placement] 欄はまだ空白です。

図 5-41. PCS と extref をコンフィグレーション後のサブシステム (Planner タブ)



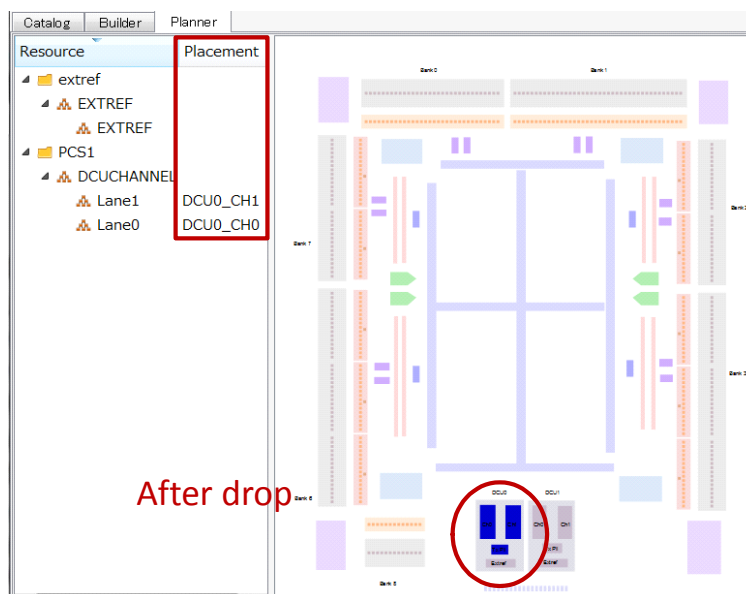
PCS の配置指定は、コンポーネントマークのある [DCUCHANNEL] を選択してドラッグ&ドロップして行います。図 5-42 はドラッグ中のスクリーンダンプです。ドラッグ先で ” 禁止マーク ” が出ず、リソース表示色が図のように変化すれば問題ありません。

図 5-42. PCS (DCU) のドラッグによる配置指定



ドロップ後は図 5-43 のように物理リソースの色が青色になります。また、Placement 欄にリソースの物理位置情報が表示されます。

図 5-43. PCS (DCU) のドロップ後の表示例



次に同様にして extref モジュールもドラッグ&ドロップします。基本的に extref は同一 DCU 内とします。

特にマルチプロトコル対応のサブシステムの場合は、ここで DCU のクロック入力設定を (確認) します。のように配置した DCU をダブルクリックして ”DCU Settings” ウィンドウを表示させます。プルダウン形式で TX PLL とレシーバの入力ソースを選択します。

図 5-44. extref をドロップ後の表示例

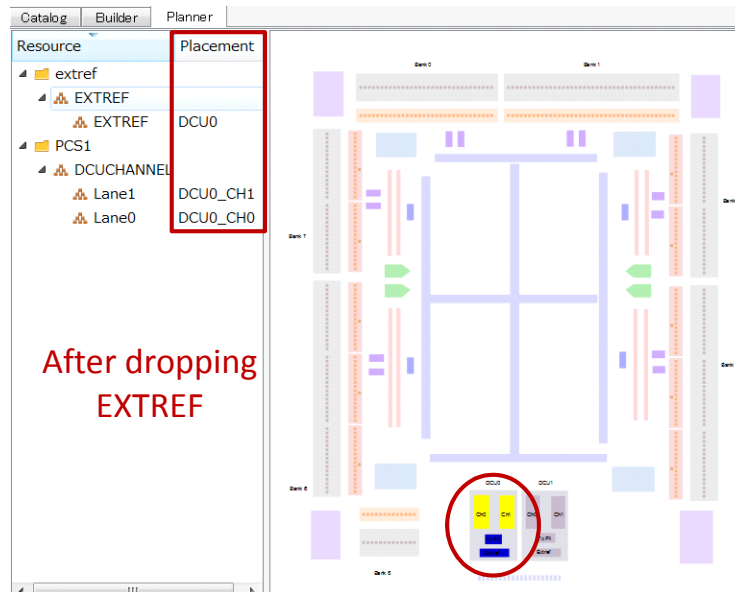


図 5-45 は本例が単一チャネルのみですので選択肢は二つしか表れません。図 5-46 は別の例で DCU を二つ使用する場合で、外部クロック入力の候補が増えているのがわかります。

図 5-45. DCU クロックソースの設定

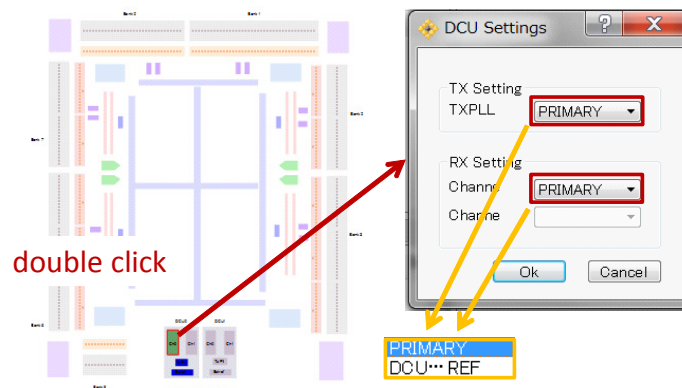
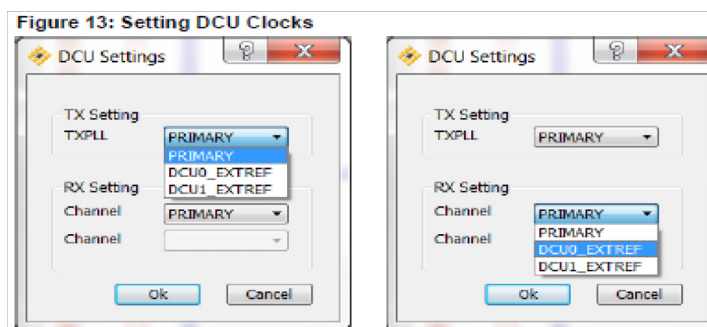
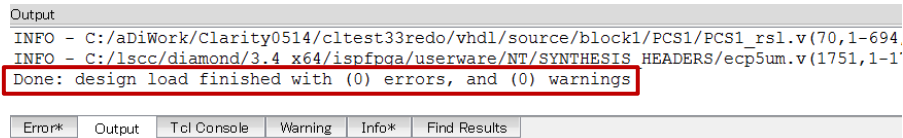


図 5-46. マルチ DCU 使用時の "DCU Settings" 例



以上で各種構成指定・設定は終了です。最後に [Generate] ボタンをクリックしてサブシステムを生成後、エラーがなければ終了します。エラーのある場合は解決して先に進みます。

図 5-47. Generate 後、コンソールのメッセージ例



なお、Clarity Designer で生成したサブシステム（モジュール、或いは ” コンポーネント ”）に関わる設計制約は、IPexpress の場合 LPF 制約ファイルにマニュアルで記載（転記）する必要がありました。Clarity Designer ではその必要は無く、物理制約ファイル PRF (<implementation>.prf) に反映される形式になっています（スプレッドシート・ビューで確認してもわかりません）。配置指定するプランニング情報とともに配置配線エンジンにパラメータを直接渡すためです。タイミングに関わる制約については、レポートファイルで確認することができます。

また、IPexpress では <PCS モジュール名 >.txt という名称で PCS/SERDES 設定用の『自動コンフィギュレーションファイル』(Auto Configuration File) がテキスト形式で RTL と共に出力されていました。Clarity Designer ではこうしたファイルは存在しません。従来ユーザが編集してパラメータ変更等をすることも可能でしたが、Clarity Designer 対象デバイスではできなくなりました。

5.3.10 プランニング例 2 ～ DDR Generic インターフェイス

ここではモジュール DDR Generic のプランニング例を示します。サブシステム名 (<name>.sbx) 設定は省略します (図 5-21 参照)。『Catalog』タブで [DDR Generic] を選択して (図 5-48) ダブルクリックすると、図 5-49 が立ち上がります。

図 5-48. DDR Generic モジュールの生成開始

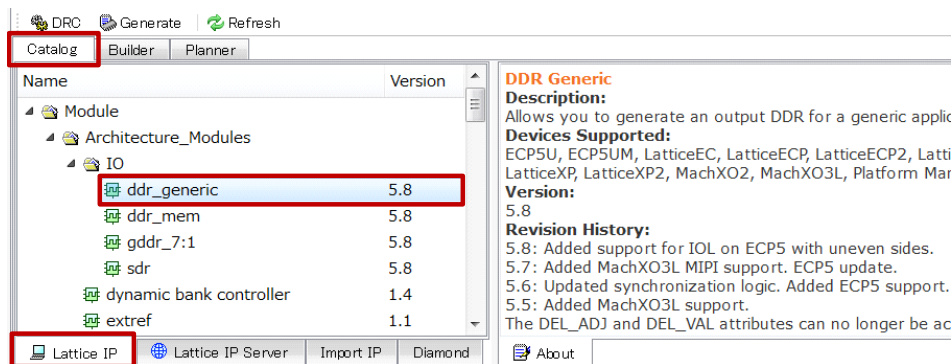
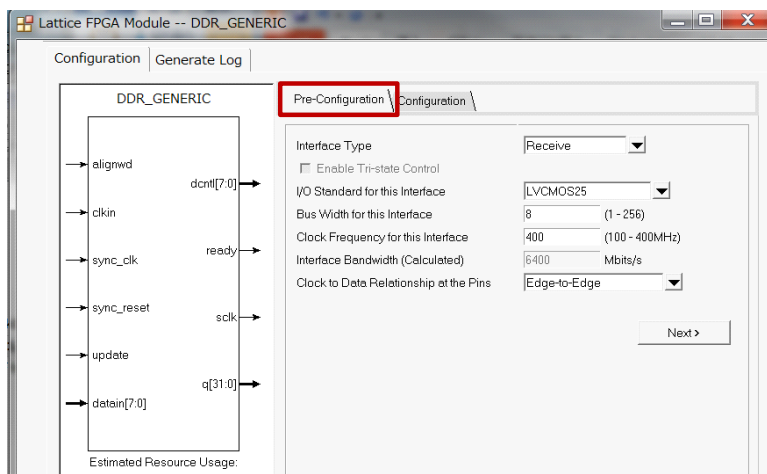


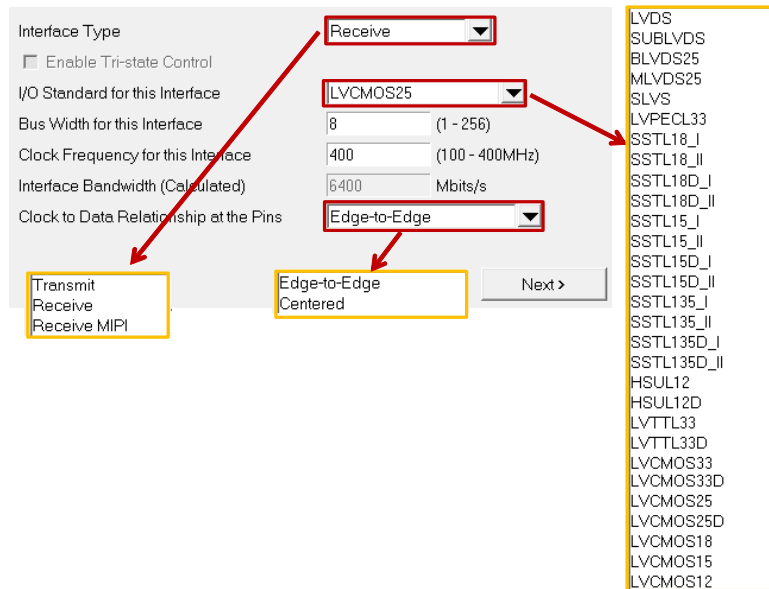
図 5-49. DDR_Generic のコンフィギュレーション ～ Pre-Configuration タブ



パラメータ設定 (Configuration) 用に、ウィンドウ上部に二つのタブがあります。通常は『Pre-configuration』での設定で完了します。データバス遅延が固定値 (デフォルト) 以外の必要があるなど、特定のケースでなければ『Configuration』タブでの設定は不要です。詳細は ECP5 のテクニカルノート TN1265 をご参照ください。

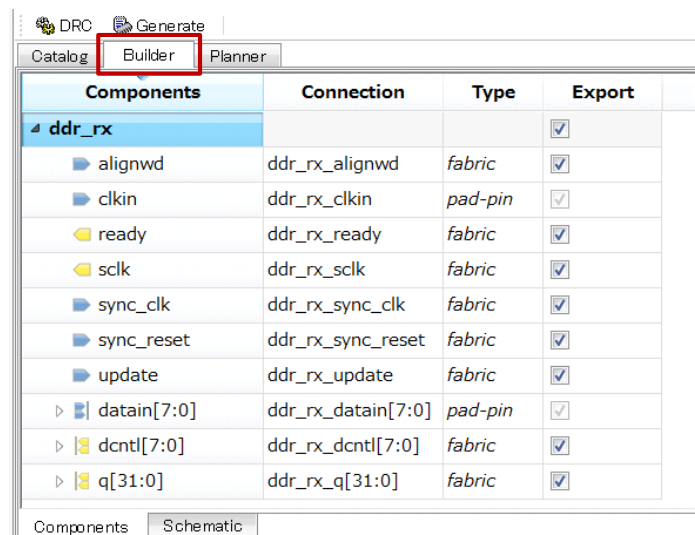
Pre-configuration の基本設定としては図 5-50 のように、Tx か Rx か (註: 基本的に [Receive MIP] は選択しないこと)、I/O インターフェイスタイプ、データレーン数、データレート、そしてクロックとデータの位相関係などがあります。

図 5-50. DDR Generic 基本パラメータ



コンフィグレーション完了後、ウィンドウ左下の『Configure』ボタンをクリックします (本例でのコンポーネント名は "ddr_rx" としている)、『Builder』タブをクリックしてコンポーネント名表示の行を展開すると、図 5-51 のようになります (ウィンドウ下部は『Component』タブ)。

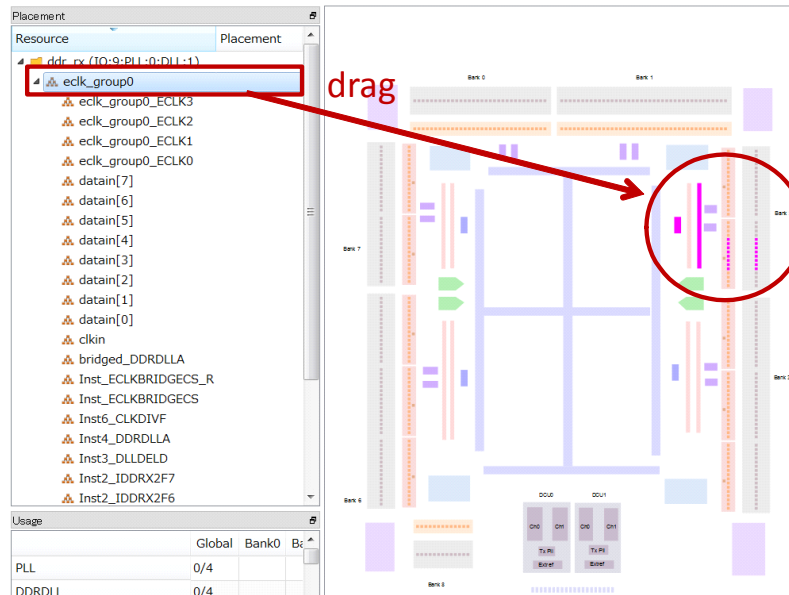
図 5-51. DDR Generic コンフィグレーション後の ddr_rx コンポーネント (Builder タブ)



次に『Planner』タブをクリックします。Resource 欄最上行の [ddr_rx] を展開し、eclk_group0 をドラッグ & ドロップして配置指定します。図 5-52 はバンク 2 にドラッグ中の様子です。ここでは同バンク下部の領域にカーソルを移動していますが、該当ピンとエッジクロックネット (ECLK) とクロック分周器 (CLKDIV)

などの所用リソースが同じくピンク色に変わり、配置可能なことを示します。ピンの選択は自動で判断されます。

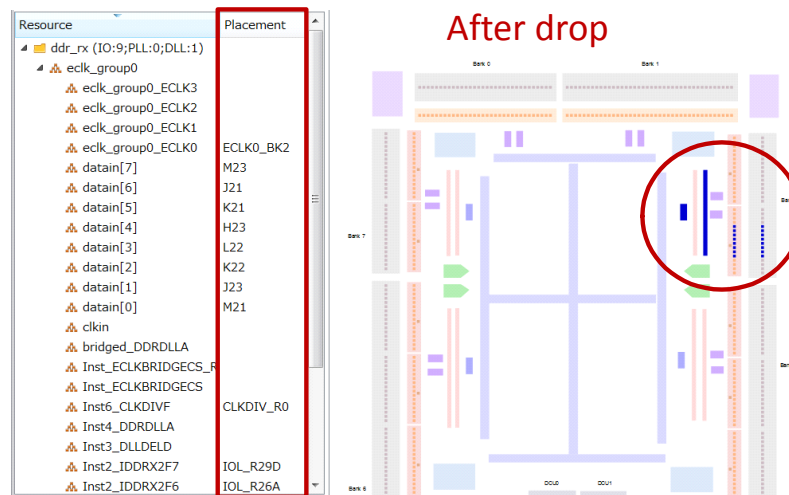
図 5-52. DDR_Generic (eclk_group) のドラッグによるバンク指定 (Planner タブ)



この状態でドロップする（マウスを離す）と、色が青に変わり、指定が受け付けられたことを示します。同時に、左枠 [Placement] 欄に割り当て結果としての位置情報が表示されます。

なお、コンポーネントやドラッグ先によって、付随して使用されるリソース位置が小おなるのは勿論ですが、場合によっては、例えば 1 本で済むべきクロックネットを 2 本使用せざるを得なくなるなどの結果になりますので、ドロップ先はよく注意するようにします。

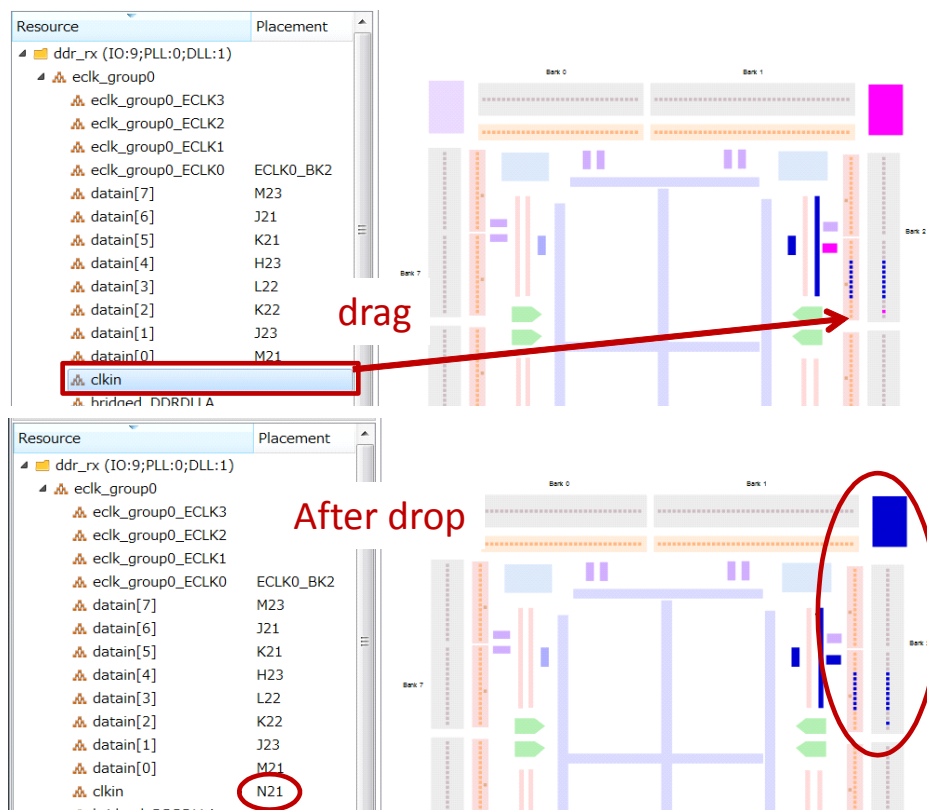
図 5-53. DDR_Generic (eclk_group) のドロップ後の表示例



この状態ではまだクロック関連の指定が行われていません。Clarity Designer の考え方として、クロック入力には必ずマニュアルで別途指定することが必要です。そこで [clkln] ポートを選択し、先ほどと同じバンク 2 の下部ピン近辺にドラッグします。配置可能なピンがあれば、付随して使用される DDRDLL や遅延素子 DLLDEL などのクロック関連リソースと共にピンク色に変わります (図 5-53) ので、ドロップします。配置されたリソースが青色になり (図 5-54)、[Placement] 欄が配置後のリソース情報で満たされます。

次に Generate する前に、『Catalog』タブ表記の上部にある『DRC』をクリックしてルールチェックを実行します。問題がなければプランニングは終了です。例えば上記 [clkin] の指定をする前に DRC を実行すると、指定が必要であることを示すメッセージがコンソールに表示されます。

図 5-54. クロック入力ポート (clkin) の配置指定



最後に『Generate』すると、“ddr_rx.sbx” が File List 内に取りこまれていることが分かります。その後は Diamond フローに戻ります。

5.3.11 プランニング例 3 ～ DDR3 メモリインターフェイス

ここではラティスの DDR3 メモリインターフェイス IP を使用したプランニング例を示します。DDR3 IP のコンフィグレーションは終了しているものとします。ラティスの IP の場合、二つの上位リソース “eclk_group0” および “sclk_group1” が必要な構成になっています (自动生成)。プランニングはそれぞれについて行います。

まず “eclk_group0” をドラッグして配置先を探します。図 5-55 のように、有効なロケーションがあるとパッドや DLL など所用リソースの色がピンク色に変わります。ドラッグ先 (のポート位置) を変えるとそれらも変化しますので、所望の所でドロップします。この例はメモリとのデータバス幅が 64 ビットの場合のため、かなりのリソースを消費します。データポートは反時計回りに順にポートが割り当てられます。従って配置先バンクの版時計回りで先頭のポートにドラッグ先を移動すると最もバンクの使用効率的には最良になります。

ドロップ指定するリソースの順序や何らかの理由で、すでに指定済みの位置を取り消す場合、図 5-56 のようにリソース名を右クリックして “Reset” を選択します。

図 5-55. "eclk_group0" のドラッグによる配置先を探す

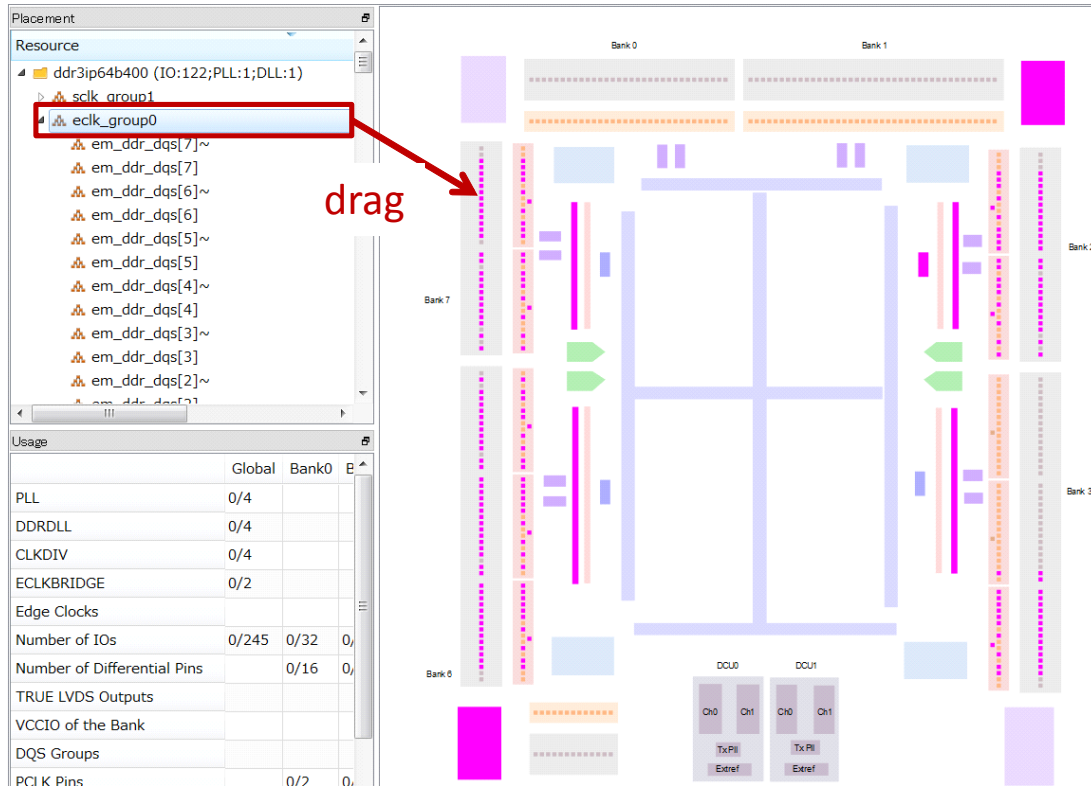
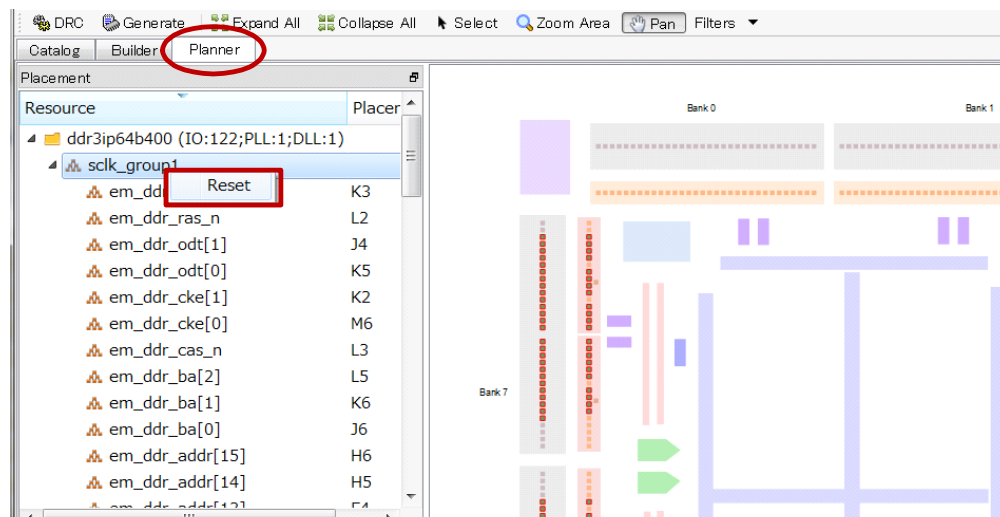


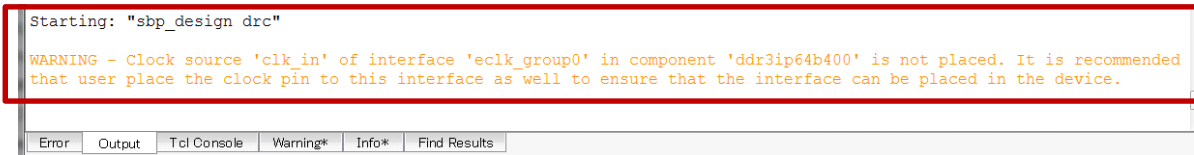
図 5-56. 配置指定済みのリソースを取り消し



もう一つのリソース "sclk_group1" についても同様にドラッグ&ドロップ指定します。

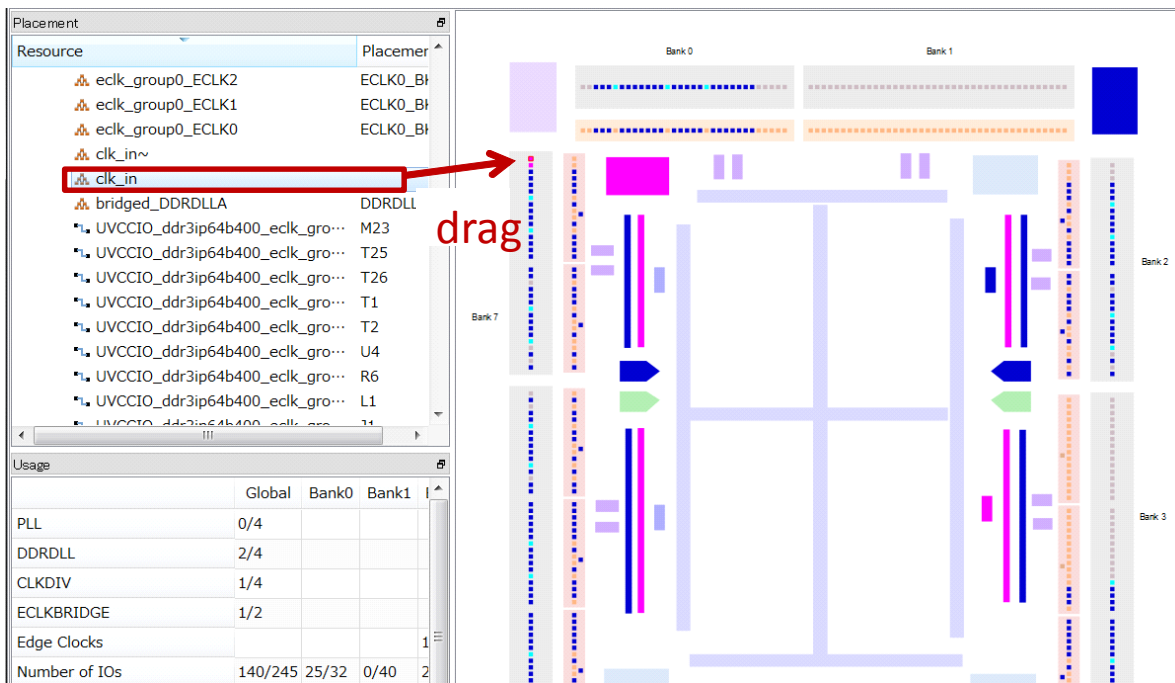
次に、二つの上位リソース "eclk_group0" および "sclk_group1" をプランニングしたので『DRC』をクリックしてみます。その結果 (図 5-57)、"clkin" ポートが未指定である旨のメッセージが表示されました (コンポーネント "ddr3ip64b400" 内インターフェイス "eclk_group0" のリソース "clk_in" が配置指定されていない)。

図 5-57. 未指定リソースがある場合のメッセージ例



前節の例で記述したように、Clarity Designer の考え方として、クロック入力は必ずマニュアルで別途指定することが必要です。リソース欄で”eclk_group0”を展開し、リソース”clk_in”をドラッグします。ドロップ可能なパッドにドラッグすると、図 5-58 のように関連リソースがピンク色になります。所望の位置でドロップします。なお、”clk_in”は差動入力ですが反転入力”~clk_in”をドラッグせず、必ず非反転側信号を扱います。

図 5-58. ”clk_in”の候補先ポートを探す



以上で DRC もパスしますので、『Generate』して Diamond フローに戻ります。シグナルインテグリティ向上のための”仮想接地 (或いはバーチャル VCCIO / GND)”は Clarity Designer が自動的に抽出・指定して、マッパーに渡します。ユーザが IPexpress で手修正も可能ですが、推奨ではありません。

バーチャル VCCIO/GND

本節冒頭に言及したように、Clarity Designer フローでは DDR3 メモリインターフェイスでの SSO 特性を改善する一助として、自動的に各 DQS グループ内のピンを仮想接地する機能がサポートされています。配置配線プロセスが完了後に、プロジェクトナビゲータ Design Summary で”Process Reports”=>”Signal/Pad”項を選択します。レポートウィンドウ内”Pinout by Port Name”セクションを確認すると、の例のようなリストが含まれています。PCB 上ではこれらを VCCIO に接続します。

図 5-59. バーチャル GND ピンのリスト例

Design Summary	Pin Name	Pin	IO Type	IO Mode	IO Config
Project	na1np[5]	A15/51	H51	HUKAFU_D1CH1	
Process Reports	hdoutn[0]	AD8/50	HS0	HD1XN0_DOCH0	
Synplify Pro	hdoutn[1]	AD11/50	HS0	HD1XN0_DOCH1	
Map	hdoutn[2]	AD17/51	HS0	HD1XN0_D1CH0	
Place & Route	hdoutn[3]	AD20/51	HS0	HD1XN0_D1CH1	
Signal/Pad	hdoutp[0]	AD7/50	HS0	HD1XFO_DOCH0	
JEDEC	hdoutp[1]	AD10/50	HS0	HD1XFO_DOCH1	
Analysis Reports	hdoutp[2]	AD16/51	HS0	HD1XFO_D1CH0	
	hdoutp[3]	AD19/51	HS0	HD1XFO_D1CH1	
	reset_n	AD3/8	LVCMS025_IN	PB18A	PULL:DOWN CLAMP:ON HYSTERESIS:ON
	u1_ddr3ip64b/UVCCIO_ddr3ip64b400_sclk_g*	C9/0	UVCCIO_OUT	PT24B	DRIVE:16mA CLAMP:ON SLEW:FAST
	u1_ddr3ip64b/UVCCIO_ddr3ip64b400_sclk_g*	A9/0	UVCCIO_OUT	PT18B	DRIVE:16mA CLAMP:ON SLEW:FAST
	u1_ddr3ip64b/UVCCIO_ddr3ip64b400_sclk_g*	C5/0	UVCCIO_OUT	PT9B	DRIVE:16mA CLAMP:ON SLEW:FAST
	u1_ddr3ip64b/UVCCIO_ddr3ip64b400_sclk_g*	U23/3	UVCCIO_OUT	PR53D	DRIVE:16mA CLAMP:ON SLEW:FAST
	xau1_pll_lol	AA2/8	LVCMS025_OUT	PB15A	DRIVE:8mA CLAMP:ON SLEW:SLOW
	xau1_pll_refclk1	E13/1	LVCMS025_IN	PT42A	PULL:DOWN CLAMP:ON HYSTERESIS:ON

FPGA の実装工程に先立ち PCB デザインをしてポート配置を決めることは推奨しません。各ポートの配置と共に、望ましい仮想接地も踏まえて FPGA デザインの後にポート配置を確定するようにご注意ください。

5.4 改訂履歴

Ver.	Date	page	内容
3.3v1.0	Mar.2015	-	初版 (Clarity Designer 関連既述は Ver.3.4 相当)
3.3.1 (3.3, rev1.1)	June 2015	1	Clarity Designer での DDR メモリ I/F 有り PCB デザインフローの注記を追加
		11	図 5-18、図 5-19 差し替え。関連記述修正・追加
		25	5.3.9 節 PCS プランニング後に、Clarity Designer 使用に関する注記を移動&追記
		28 ~ 30	5.3.11 節 DDR3 メモリインターフェイスのプランニング追記。仮想接地の記述追加
		-	(全般) タイポ、マイナーな記述修正等

--- *** ---

