



# コンポーネントのモデルとライブラリ の概念

## 概要

Article  
AR0104 (v1.1) July 28, 2004

このアーティクルでは、コンポーネントのモデルおよびライブラリの定義と、それらの関係について解説します。モデルと対応させるための検索順序は、この検索を特定のモデルに限定するオプションと同様の解釈ができます。

コンポーネントは電子製品の根本的な構成パーツですが、デザインキャプチャおよびインプリメント・プロセスでは、各々異なった方法でコンポーネントを表現する必要があります。つまり回路図では論理的なシンボル、PCB ではフットプリント(型)、シミュレーション用には SPICE 定義、シグナルインテグリティ解析用には適切なシグナルインテグリティの種類、あるいは完成した PCB の 3D 表示用には 3 次元表示で表現されます。

これらの表現全てが全コンポーネントに必要なとは限りません。しかし論理的なものが開始点です。全てのコンポーネントは最低限、回路図ライブラリで識別名と共に定義されています。それらには単一またはマルチパートの形式で、ピンおよびグラフィックのシンボルを含み、選択表示できるディスプレイ・オプションもあります。そのためすべての回路図デザインにおいてもコンポーネントが配置されています。しかしモデルがコンポーネントに追加されるまで、それは実用的な意味を持ちません。

## 定義

**コンポーネント:** デザインに配置可能なオブジェクトです。

**シンボル:** 回路図に配置されるコンポーネントのグラフィック表示。シンボルには、外形を定義する描画オブジェクトや電気的な接続点を定義するピンを含めることができます。

**パート:** 集合抵抗やリレーの様なコンポーネントは、回路図に個別に配置することができる一連の個別のパート (マルチパートコンポーネントと呼ばれます) として描くことができます。

**モデル:** 実際のドメインで有効なコンポーネントです。

**フットプリント/型:** PCB レイアウトのコンポーネントを表すモデルに使用された名称です。フットプリント/型は、ボードレイアウト上でコンポーネントを接続したり、配置する為に必要なスペースを定義するコンポーネントの形状や PCB のパッド一式がグループ化されたものです。

**ドメイン:** 種類、グループあるいは領域。DXP 2004 環境の有効なドメインには、PCB レイアウト、SPICE シミュレーション、シグナルインテグリティ解析、そして 3D が含まれます。

**ライブラリ:** コンポーネントやモデルを収納したファイル。

**モデルライブラリ:** コンポーネントモデルを収納したファイル。

**コンポーネントライブラリ:** 回路図コンポーネントを収納したファイル。

**統合ライブラリ:** 回路図コンポーネントおよびそれらに関連したモデルを収納したファイル。

## 基本

回路図の段階では、デザインは論理的に接続されたコンポーネントの集まりです。デザインをテストシミュレーションする為には、シミュレーション、PCB レイアウト、シグナルインテグリティ解析など、別のモデル化するドメインへそれらを移行させる必要があります。

各ドメインには、コンポーネントに関する情報と回路図シンボルのピンに情報をマップする手段が必要となります。これらのドメイン情報の標準的なフォーマットとしては、IBIS、MDL、CKTなどのモデルファイルの情報があります。いくつかの情報はモデルファイルには存在せず、例えば SPICE のピンマッピングやネットリスト生成のデータはシステムで保存、管理されています。

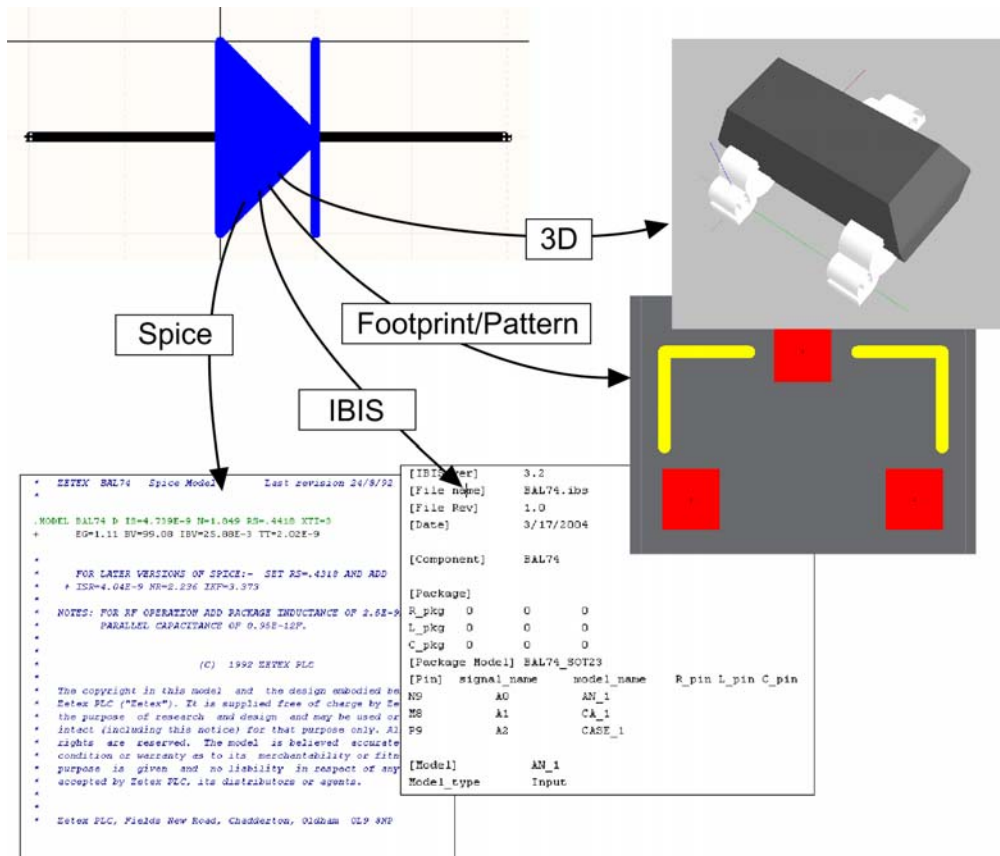


図1. 各ドメインにあるコンポーネントをどのように作成するか情報は、Model files に保存されています。

IBIS シグナルインテグリティモデルや VRML または IGES 3D モデルは、使用する前に DXP 2004 フォーマットモデルにインポートする必要がありますことに注意して下さい。IBIS モデルは、**Signal Integrity Model** ダイアログ (SI モデルをコンポーネントに追加する時に開きます) で直接インポートされます。VRML と IGES モデルは、それらが回路図コンポーネントに追加される前に Pcb3DLib にインポートする必要があります。



全ての必要なドメイン情報は、追加されている各モデルへ個別のインターフェースを格納する回路図コンポーネント内に含まれています。実際、完成したモデルは、コンポーネントに格納されたモデルマッピング情報やモデルライブラリに格納されたドメインモデル情報の組み合わせです。

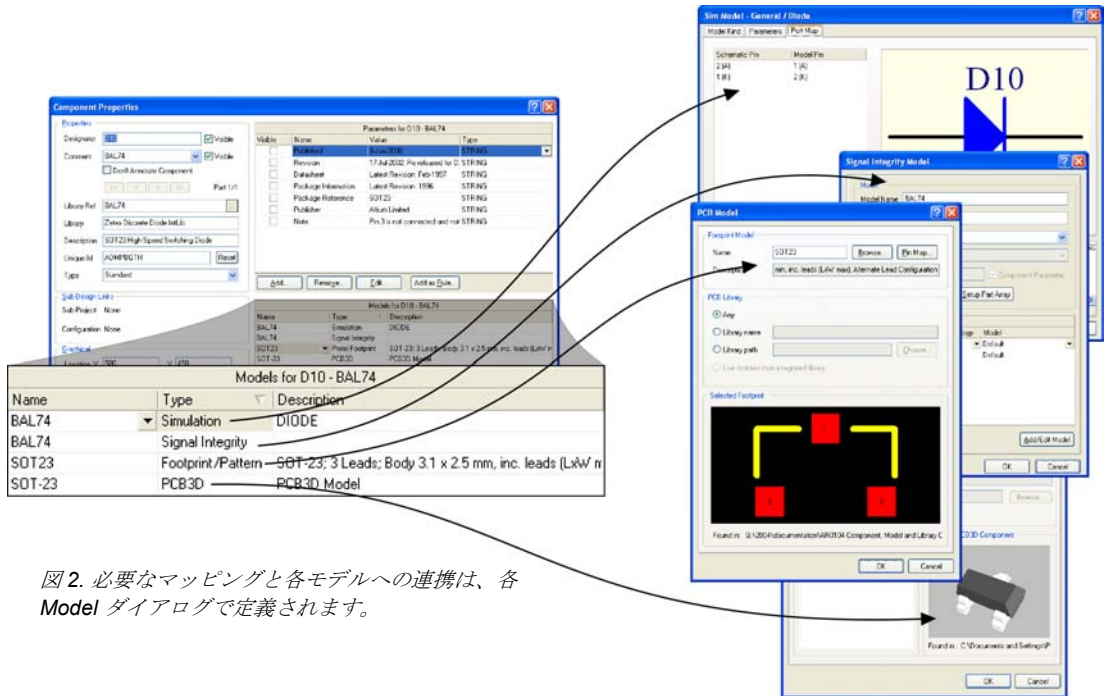


図2. 必要なマッピングと各モデルへの連携は、各 Model ダイアログで定義されます。

コンポーネントは、複数のドメイン用のモデルやドメインごとに複数のモデルを持つことができます。それらの一つが現在のモデルとして設定されます。

## ライブラリタイプ

DXP 2004 には、3つのタイプのライブラリがあります。

**モデルライブラリ** - 各ドメインのモデルは、通常モデルライブラリと呼ばれる "Model containers" に格納されます。いくつかのドメインでは、保存がファイルにつき 1 モデルである SPICE のような形式でもモデルファイル (\*.MDL, \*.CKT) として参照されます。別のドメインでは、モデルは通常、パッケージ形式のライブラリ (\*.PcbLib, \*.PcadLib) 内にグループ化された PCB フットプリント/型の様なユーザ指定の分類に従ってライブラリファイル内にグループ化されます。ファイルされます。

**回路図ライブラリ** - これらには回路図コンポーネントやモデルインタフェース定義が含まれています (\*.SchLib)。モデルインタフェース定義は、各々のモデルライブラリへのリンクを含んでいます。

**統合ライブラリ** - これらは回路図ライブラリの一式です。それらのリンクしたモデルライブラリと共に、1つのファイルへ 'コンパイル' されたのが統合ライブラリ (\*.IntLib) です。統合ライブラリへのコンパイルの利点は、全てのコンポーネント情報が単一の移動できるファイルで利用できる点です。統合ライブラリを編集する場合は、ソースを抽出してからそれを編集後に再度コンパイルすることが必要です。

### モデルの参照

コンポーネントにモデルを追加する時はいつでも、モデルを検索する場所をどのようにコントロールしたいかを設定するオプションを利用できます。あるモデルタイプから他のモデルタイプに変更でき、**Model editor** ダイアログには、一般的にこれらのオプションが含まれています。

**Any** – 一致するモデルについて、全てのライブラリを検索。

PCB3D Model

Name

PCB3D Library

Any

Library Name

Library Path  

From integrated library

**Library Name** – 一致するモデルについて、この名称のライブラリのみ検索。

PCB3D Library

Any

Library Name

Library Path  

From integrated library

**Library Path** – 一致するモデルについて、この保存場所のこの名称のライブラリのみ検索。

PCB3D Library

Any

Library Name

Library Path  

From integrated library

**Integrated library** – このコンポーネントを配置する為に使用した統合ライブラリから直接、モデルを引き抜きます。統合ライブラリは、有効な場所で利用することができます。

PCB3D Library

Any

Library Name

Library Path  

Use PCB3D model from integrated library Diodes.IntLib

## Model の場所 - 有効な検索場所

以前の項目で定義した **Reference criteria** によれば、モデルを要求する操作を実行するごとに、システムは有効な場所でモデルを検索します。例えば、回路シミュレーションを実行する時、各コンポーネントにリンクした **SPICE** モデルは、**XSPICE** シミュレーションエンジンによって検索され使用されます。他の例では、回路図キャプチャから **PCB** レイアウトにデザインを移行します。この過程の間は、各コンポーネントのフットプリント/型が検索され、**PCB** に配置されます。

検索できるモデルの有効な場所は、以下を含みます。

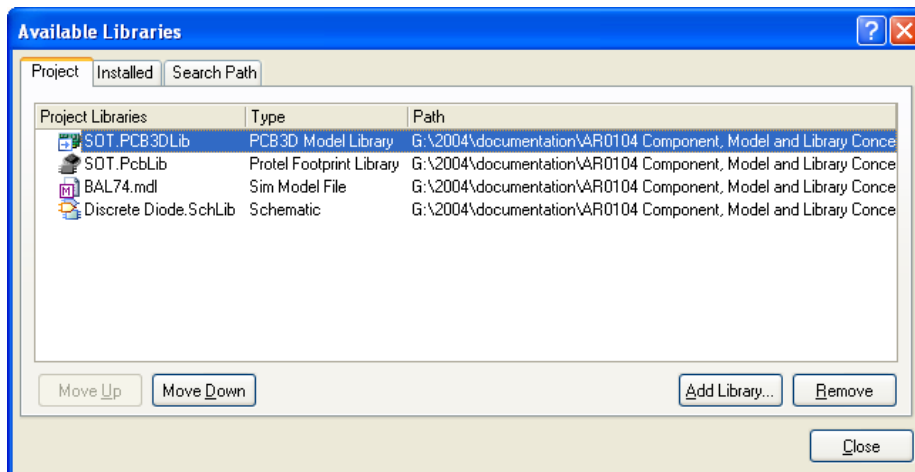
**Project Model/Library** – モデルとモデルライブラリは、プロジェクトに追加することができます。全てのプロジェクトファイルの様に、モデル/ライブラリファイルはプロジェクトにリンクされます。従って、多くのプロジェクトにモデル/ライブラリをリンクすることができます。この方法の利点は、プロジェクトを開く時はいつでもモデルを利用できることです。不利な点は、もし、プロジェクトファイルのある PC から他の PC へ移動した場合、**Project** フォルダ構造に保存されていないモデル/ライブラリを忘れてしまうことです。プロジェクトモデル/ライブラリは、現在のプロジェクトに属するドキュメントを編集している時に利用できます。

**Installed Library** – これらのライブラリは、**DXP 2004** 環境と関連しています。**Installed libraries** のコンポーネントは、全ての開いているプロジェクトに利用できます。

**Project Search Path** – モデル/ライブラリは、**Options for Project** ダイアログの検索パスを指定して、プロジェクトで利用できます。各検索パスではフォルダを定義し、もし、**Recursive** オプションが有効ならばサブフォルダを含めることができます。**Search Path** で見つかった全てのモデルとライブラリファイルは、有効になります。**Search Path** フォルダに多くのファイルがある場合、検索パスを使用してモデルを検索することは、遅くなる傾向があることに注意して下さい。

## 利用できるモデル/ライブラリの確認

**Available Libraries** ダイアログで、現在のプロジェクトで利用できるモデル/ライブラリを再確認することができます。**Libraries** パネルで **Libraries** ボタンをクリックするか、または **Design » Add/Remove Library** を選択してダイアログを開きます。



## コンポーネントのモデルとライブラリ概念

図3. *Available Libraries* ダイアログでは、アクティブなプロジェクトに利用できる全てのモデル/ライブラリを表示します。

### 一般的な検索順序

検索のルールは、一致するモデルが見つかりとすぐにモデルの検索を停止することです。統合ライブラリに関連しない全てのモデルについては、以下の順番で検索されます。

- Project Libraries
- Installed libraries
- Search Paths

この順番が、*Available Libraries* ダイアログで左から右へ続いていることに気づくでしょう。実際に、利用できるライブラリがこのダイアログ内で上から下へ順番に並んでいるので、全体の検索順序を設定するのが容易です。

DXP 2004 環境では、モデルの場所を柔軟にコントロールできますが、各モデルタイプの正しいファイル拡張子を使用することが必要です。例えば、フットプリントは\*.lib または\*.pcblib の拡張子のファイルに無ければ見つけることはできません。同様に、SPICE .SUBCKT は\*.ckt ファイルに、または、SPICE .MODEL は\*.mdl ファイルに無ければ見つけることはできません。一致するモデルを検索できない時は、*Messages* パネルにエラーが表示されます。

## 統合ライブラリ

統合ライブラリは、ライブラリの特定のクラスです（回路図シンボルライブラリと全ての参照されたモデルは、一つのファイルにコンパイルされます）。統合ライブラリの利点は、携帯性と安全性があることです。全てのモデルは、統合ライブラリ内に含まれるので、モデルファイルはプロジェクトで利用できる必要があります。または、プロジェクトが別の場所へ移動された場合、モデルファイルも一緒に移動する必要があります。統合ライブラリ内のコンポーネントとモデルは、ライブラリを再コンパイル（*IntLib* ファイルを開いて *Sources* を抽出）しない限り、編集することはできません。

統合ライブラリは、以下によって構築されます。

- 統合ライブラリパッケージを作成
- 回路図シンボルライブラリを追加
- 各コンポーネントシンボルから必要なモデルを参照
- コンパイルした出力の保存場所を含む、統合ライブラリ・プロジェクトオプションを設定
- 統合ライブラリ(*Name.IntLib*)を作成する為に、統合ライブラリをコンパイル

もし、コンポーネントシンボルを統合ライブラリから配置する場合、その統合ライブラリから参照されたモデルがシステムで検索されるだけです。統合ライブラリは、システムでモデルが検索されようとしている時に、有効な検索場所で利用できる必要があることに注意して下さい。



統合ライブラリの詳細な情報については、[統合ライブラリを使用した高度なライブラリ管理](#) のアーティクルを参照して下さい。





コンポーネントライブラリの作成、コンポーネントシンボルの作成、モデルの参照の詳細については、[ライブラリコンポーネントの作成](#) のチュートリアルを参照して下さい。



統合ライブラリの作成の詳細については、[統合ライブラリの作成](#) のチュートリアルを参照して下さい。

## コンポーネントの属性

コンポーネントの基礎は、シンボルです。各コンポーネントシンボルを、固有のコンポーネントとして表すのではなく、システムではコンポーネントを構築する為に異なる方法をサポートしています。

### 同じグラフィックで異なるコンポーネント

各コンポーネントのコンポーネントシンボルが一つの場合、コンポーネントシンボル自体が実際の物理的なコンポーネントを表す集積回路の様なコンポーネントにとって理想的です。コンポーネントには、PCB フットプリント/型、シグナルインテグリティモデルや 3D モデルのようなモデルが含まれています。

等価コンポーネントでは、絵柄としては一つのシンボルになります。これは論理的に同じコンポーネントで、コンポーネントの仕様が異なる場合に役立ちます。例えば、様々なロジックファミリー (74ACT32 や 74HC32) で利用できるロジックゲートです。この場合、コンポーネントシンボルが一度、作成されれば、コンポーネントエリアスが各同じコンポーネントについて定義されます。コンポーネントエリアスは、**Schematic Library** エディタパネルで追加されます。コンポーネントエリアスは、一つのコンポーネントを複数の名称で表すことができます。

一種類のコンポーネントのシンボルは、抵抗の様な個別のコンポーネントに役に立ちます。コンポーネントには、複数の PCB フットプリント/型をリンクさせることができます。コンポーネントの値は、ライブラリ内では無く、コンポーネントが回路図に配置される時に定義されます。

### 同じコンポーネントで異なるグラフィック

システムでは、同じコンポーネントで複数のグラフィックもサポートしています。例えば、標準形状のコンポーネントを使用して描かれたコンポーネントシンボルや、あるいは、特定の形状に従って描かれたコンポーネントシンボルが必要かもしれません。一つのコンポーネントについてグラフィック表示を追加するには、ライブラリエディタの **Tools** メニューか **Mode** ツールバーを使用して、新規のコンポーネントの **Mode** を追加します。

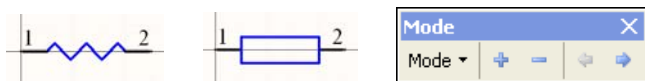


図 4. 一つのコンポーネントについて、複数のグラフィック表示を定義する為の **Mode** 機能を使用

### 標準でないコンポーネント形式


全てのコンポーネントが PCB に実装されたり、部品表で必要であるとは限りません。また、PCB に実装される全ての情報が回路図上で表される必要があるとは限りません。システムでは、(ライブラリまたは回路図での) **Component Properties** ダイアログの **Component Type** 属性でこのような設定を行うことができます。

## コンポーネントのモデルとライブラリの概念

例えば、PCB を接続するようなシャーシに組み込む部品を回路図に含めることで回路図がわかり易くなります。このコンポーネントが PCB BOM で必要無ければ、コンポーネントタイプを **Graphical** に設定することができます。グラフィックコンポーネントは、回路図の電氣的な検証や BOM には含まれず、回路図を PCB に同期させる間もチェックされません。この場合、**Component Type** を **Graphical** に設定します。

コンポーネントの別の特別な種類としては、テストポイントがあります。このコンポーネントは、回路図と PCB の両方で必要となりデザインの同期中にチェックされますが、BOM には含まれません。この場合、**Component Type** を **Standard (no BOM)** に設定します。

特別なコンポーネントの種類別の例として、ヒートシンクがあります。一般的に、それは回路図には表示せず（表示することは可能です）、回路図の電氣的な検証でチェックする必要はありませんが、BOM に含める必要があります。この場合、コンポーネントの種類を **Mechanical** に設定します。

様々なコンポーネントの種類の詳細については、**What's This help**  を使用するか、または **Component Properties** ダイアログを開いて F1 を押して下さい。

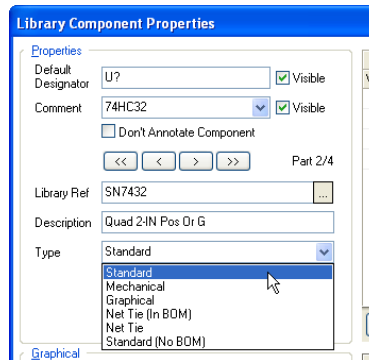



図 5. 特別なコンポーネントについてコンポーネントの種類を設定


## コンポーネントパラメータ

一般的に、コンポーネントには様々なテキスト情報が含まれています。これには、コンポーネントの電氣的な詳細情報（例えば、ワット数または許容差）、コンポーネントの購入先、在庫状況、設計者による注記、コンポーネントのデータシートの参照情報などを含めることができます。この情報は、ライブラリエディタでコンポーネントを作成中、または一度コンポーネントが回路図に配置された場合のいずれかの状態でコンポーネントにパラメータを追加することによって含められます。

Parameters for SN7432			
Visible	Name	Value	Type
<input type="checkbox"/>	Part Number	U001680	STRING
<input type="checkbox"/>	Manufacturer	Texas Instruments	STRING
<input type="checkbox"/>	Manufacturer P/N	SN74HC32N	STRING
<input type="checkbox"/>	Voltage-Rated	5V	STRING

図 6. 必要なコンポーネントの詳細を含めるには、コンポーネントパラメータを追加

 会社のデータベースからコンポーネントにリンクさせる為の詳細な情報については、[データベースからコンポーネントへのリンク](#) のチュートリアルを参照して下さい。

 コンポーネントからデータシートのようなドキュメントを参照するには、パラメータ **HelpURL** を入れて下さい。パラメータ **value** はドキュメントを示し、PDF のページ番号を含めることができます。例えば、**HelpURL value** に **CR0118 FPGA Generic Library Guide.pdf#page=93** と記述されていれば、カーソルがコンポーネント上にある時に F1 ボタンを押すと、参照された PDF ファイルの 93 ページが表示されます。ローカルドキュメント（もし、パスが **Operating system** で定義されていないならば、パスを含めます）、または、インターネット URL を参照することができます。



もし、コンポーネントに **HelpURL** パラメータが含まれていない場合は、デフォルトのコンポーネント・ヘルプ・トピックが表示されます。

## マルチパートコンポーネント

複数のシンボルを使用して一つのコンポーネントを表す方が適切な場合があります（例えば、抵抗アレイの各抵抗、またはリレーのコイルや接点）。

ライブラリエディタの **Tools** メニューのコマンドを使用して、パートを追加、削除することができます。各パートが個々に表示されますので、そこでピンを追加します。

## パワーピン

コンポーネントのパワーピンは、他のコンポーネントピンの様に表示することができます。または、非表示にすることができます。非表示のピンは、**Pin Properties** ダイアログの **Connect To** 属性で定義されたネットに自動で接続されます。

コンポーネントデザイン中の非表示のパワー・ピンを表示するには、ライブラリエディタで **View** メニューの **Show Hidden Pins** オプションを有効にしてください。

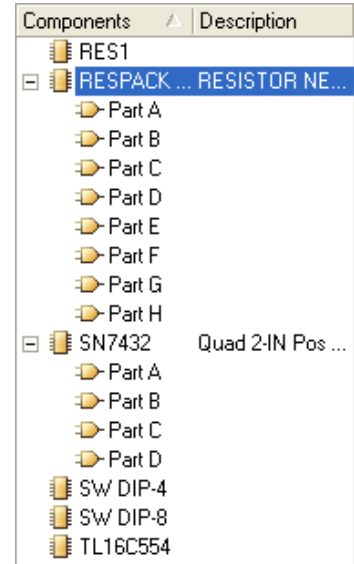


図7. コンポーネントは、一つ、または複数のパートを持つことができます。

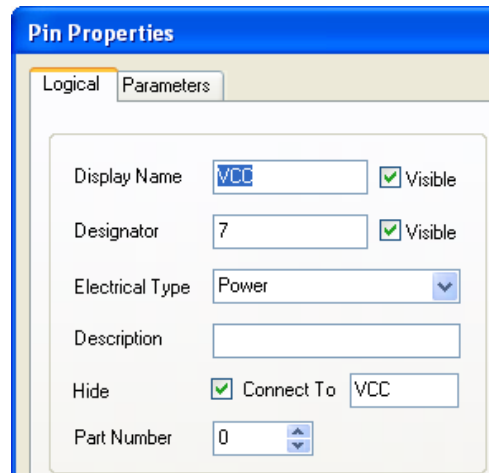


図8. 非表示のパワー・ピンについては、このピンが接続されるネットを **Connect To** 属性に定義します。

## 更新履歴

---

Date	Version No.	Revision
9-Dec-2003	1.0	New product release
28-Jul-04	1.1	Added images and extra detail to text

Software, documentation and related materials:

Copyright © 2004 Altium Limited.

Copyright © 2005 Altium Japan.

All rights reserved. Unauthorized duplication, in whole or part, of this document by any means, mechanical or electronic, including translation into another language, except for brief excerpts in published reviews, is prohibited without the express written permission of Altium Limited. Unauthorized duplication of this work may also be prohibited by local statute. Violators may be subject to both criminal and civil penalties, including fines and/or imprisonment. Altium, CAMtastic, CircuitStudio, Design Explorer, DXP, LiveDesign, NanoBoard, NanoTalk, Nexar, nVisage, P-CAD, Protel, Situs, TASKING, and Topological Autorouting and their respective logos are trademarks or registered trademarks of Altium Limited. All other registered or unregistered trademarks referenced herein are the property of their respective owners, and no trademark rights to the same are claimed.