

从 Protel 99SE 传递设计到 DXP

目录

摘要	1
生成项目	1
元件	2
库	2
连接和唯一标识	2
网络认定范围	3
PCB 导入向导	3
板外形	3
分割层	3
特殊规则转换	4
规则	4
仿真模型参数及配置	4
多通道设计	5
存储工作	5
注意	5

摘要

这篇文章简要概述了从 Protel 99SE 设计转化到 DXP 中的 PCB 项目的步骤。讨论了数据库与项目组间的相互关联，介绍了元件及库的创新，描述了设计继承中潜在的问题以及如何解决这些问题的信息。

生成项目

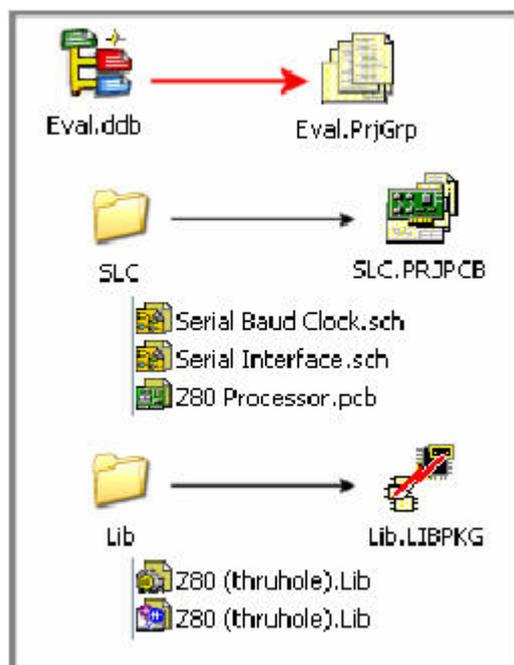
非常简单的在 DXP 菜单中打开数据库文件，就可以将一个 99SE 的数据库文件转换为 DXP 中的项目。确定这个操作启动一个两步的流程。首先，所有数据库文件中的内容被放到一个新的 Windows 文件夹中（有和数据库文件相同的名字且存放在同一路径下）。其次一些附近的文件在 DXP 中产生并且打开，也就是一个项目组文件，外径几个独立的项目文件。

当你的 99SE 数据库包含了多种类型的文档——在第一步所有的都被写到硬盘上——那么在出现第二步的自动项目构建中只考虑 PCB，原理图，库及网络表。会为每一个至少包含这几种文件类型其中之一文件夹创建一个独立的项目文件。项目的类型取决于各个文件夹的内容。

如果这个文件夹包含有原理图，网络表或 PCB 文件，那么将创建一个 PCB 项目，所有被承认的设计文档会包含在里面（也就是原理图，PCB 和库文件）。

如果这个文件夹中只有原理图库而没有原理图或 PCB，一个库包将被创建，这个文件夹中的原理图库文件将被添加到库包中。（PCB 库不会自动的被添加到库包中，你可以指定搜索路径以找到模型。）

DXP 通过文件的扩展名识别文档。当 DXP 项目生成时，数据库中任何不带有适当扩展名（或根本就没有扩展名）的文档会遗失。记住，然而这些文档会被写到硬盘上，可以通过以适当的扩展名重命名这些文档，将它们从 Windows 浏览器中拖到 DXP 项目面板来使项目包含它们。



元件

当你检查你的 DXP 中的设计文档时，你会发现一些变化—这些变化不会打断你的工作流程。一个变化是原理图元件和库的所有文本栏和部件栏会转化为参数。当 99SE 为每个元件提供 8 个文本栏和 16 的部件栏时，DXP 可以添加的无数量限制的可利用的参数。

一些 99SE 元件中特殊的栏中的内容会作为仿真数据被保存。当这些栏内容同样被用于仿真时，DXP 会将它们的值传送到模型栏而不是参数栏。象参数一样，一个单独的元件可以无限制的添加多个模型。与参数不同的是，在同一时间，给出的模型中只能有一个被使能。

库

DXP 对库的处理提供了一些创新，但是都不是强制性的。在你的 99SE 中的所有库都可以在 DXP 中直接工作；简单的将你所需要的库（原理图或 PCB）加载到 DXP 的库面板中，它们可以象以往一样工作。

那就是说，强烈建议你建议你适应于 DXP 的新的库资源。库面板中已加载库列表可以被重新排列，以便于你决定优先在哪个库中搜索匹配的封装。这里也允许你使用第三种类型的库：集成库。

集成库允许你将特定的模型分配到特定的元件，然后将它们绑定到一起。这样，当一个元件被放到原理图中时，同步的，一个 PCB 文档，在库中的，针对这个元件的你准备使用的封装模型出现。这里不会有“元件没有被找到”的错误，因为集成库文件事实上是一个包含所有在集成包中模型的数据库。利用它们所提供的安全性与便携性，你会最终将集成库包含到你的设计流程中。在此期间，你已经创建的库会继续很好的工作。

连接和唯一身份

在 99SE 与 DXP 中相同的，唯一身份使原理图和 PCB 中的对象保持联系，甚至当它们其中之一的基本参数被修改时也保持联系。然而需要注意的是，当你从一个 99SE 设计产生一个新的 DXP 项目时，所有的原理图和 PCB 元件之间的联系都会消失。

重新建立连接关系很简单。首先，重置原理图部分的所有唯一身份，使其较短一些以便于管理，然后给其他对象新的值，如原理图符号。那么在 PCB 部分，成对的带有封装的元件出现在元件联系对话框中。完全同步 99SE 的数据库需要做到两个程序：首先，根据标识符匹配添加联系关系（默认的联系），然后执行更新。

DXP 与 99SE 根本的不同点是，建立连接并非同步的先决条件。举例来说，如果你跳过前面描述的先后次序，只是简单直接的尝试在从 99SE 带到 DXP 的 PCB 设计中运行更新/导入命令，你会得到以下的信息，根据唯一名称的同步失败，你仍然可以根据标识符来继续匹配。这样做不会影响到你设计中的唯一身份栏内容，这意味着当你重复这个过程时，同样的失败错误会报告。给原理图元件和 PCB 封装分配同样的唯一身份是保持它们间牢靠连接的唯一方法。

网络认定范围

在你的原理图对象中应该适当的分配特定的网络定义。默认的，项目选项对话框中的设置会自动分配（基于内容）。这意味着如果你的项目包含了原理图符号和图纸入口在内，范围会被设置为分等级的。如果你的项目内有端口而没有图纸入口，范围会被设置为平行的。如果你的项目既没有图纸入口也没有端口，网络标签会成为全局的。

如果你不希望使用自动探测，你可以简单的分配独立的范围应用于项目而不管它所包含的内容。建议在 99SE 项目使用全局范围的端口和网络标签，因为这些不会通过 DXP 的自动探测找到。

PCB 导入向导

当你第一次在 DXP 中打开一个继承下来的板子，一个导入向导会帮助你分配板形，层分割以及规则转换。

板形

所有 DXP 中的 PCB 设计都需要板形。这是 DXP 中的新特性，所以在你从以前的版本导入板子时必须加上板形。导入向导提供两个选项：一个矩形的框包含你所有的设计对象，或者一个根据你的设计产生的更精确板外框。如果你选择“latter”选项，你的布线框和机械层会被分析用于为你的设计生成板边界的板形。无论你选择什么选项，一个预览框会显示出预想的板形。

板形定义板的物理外框，同样提供内层阻隔线外框。因为位面是底片图形，阻隔线在板层和板边缘间创建一个无铜的区域，阻止其与加工板边缘距离过窄。这些线在面层上是不能被直接编辑，但是在 DXP 中可以在任何时候重新定义板形，因此阻隔线可以被重新安排。层堆栈管理器允许你改变你在导入向导中设置的内层阻隔距离。

分割层

DXP 改变了分割已定义层的方法。以前，分割层被摆放到内层的顶部，每一个都在它自己的“bubble”范围内。大不相同的是，DXP 分割层根据无论哪儿出现阻隔对象来创建与层上其他部分的隔离推断出分割层。这些阻隔区域不属于一个或另一个分割层；DXP 设计不再需要重叠或并列的线靠在临近的分割层。

这里有一个例外：导入向导允许你在继承分割层模式操作。如果你在在你的设计中导入层遇到问题或者如果你的 PCB 包含分割层需要在早期的版本中作进一步的编辑时，我们才建议你在继承分割层操作。稍后，你会将你的设计转换

到 DXP 层模式；与此同时，新的分割层会放到内层上，而不是通过阻隔来推断。

当你转换你的设计到新的方法时，你能降低你的分割层精确度。你不是必须这么作，因为你的继承分割层仍然可以工作在 DXP 中，但是它们可能包含了多余的线以至使你的板子更复杂及超过它应有的计算深度。在 DXP 中更新 99SE 分割层定义最简单的方法是，先添加一个新的面层，然后描绘已出现的区域到新的层。一旦做了这个，选择旧层上的所有对象然后删除它们。当这个层从所分配的网络被断掉后，在层堆栈中可以删除这个层。最后，检查每一个分割区域的网络是否被正确分配（使用 PCB 浏览器面板的分割层编辑器是最简单的方法）。

特殊规则的转换

在一些 Protel 的早期版本中不允许焊盘根据扩展规则设置到阻焊层，也就是说在一些早期的设计中有可能有焊接层或阻焊层扩展规则只定义到某个单一焊盘上。导入向导会检测出你的设计中类似的规则，并提出转换它们到焊盘设置，因此简化你设计规则的设置。从另一方面来说，导入向导会提出创建一个新的规则分离各个层，因为一些旧的 Protel 版本不允许层间的连接。

规则

另一个在 DXP 中的不同是设计规则的范围，这个范围通过在列表面板中使用 query language 定义到高亮的对象。所有你的已存在的规则将被正确的导入，除了范围会被显示为一个简单的 query，如 InNet (GND)，这个范围是通过先前一系列的对话框标签及下拉选框来建立的。要在整个板子上引用规则，默认的范围应该被保留。

当一个先前版本的 Protel PCB 设计在 DXP 中被打开时，这个规则范围 (rule-scope) 转换会自动出现，以区分规则的优先次序（为了解决当规则范围发生重叠时的问题）。关键在于在 DXP 中定义规则范围的新方法，连同重新规定潜在冲突规则优先级的能力，提供了在 PCB 设计规则检查中空前的自由和控制制度。

仿真模型参数和配置

在 99SE 中，所有的仿真模型都包含在安装时提供的 SimulationModels.ddb 中。另一方面，在 DXP 中，提供了几个你可以得到这些模型的位置。（参考 Components、Models & Library Concepts 一文以获取更多信息。）

因为所有的 99SE 文档利用定义模型路径来连接原理图元件与仿真模型，最简单的在 DXP 中保持你的 99SE 中仿真工作的方法是从 99SE 的仿真模型数据库中导出所有的文件夹和模型到 Altium\Library\Sim 路径文件夹中。

DXP 支持使用一个完整路径来引用一个模型。当导入一个带有仿真元件的 99SE 原理图时，仿真模型的连接自动传输到 DXP 的完整路径模型位置栏。DXP 包含一个内部检查，当搜索一个完整的模型路径时始终包括了 \Altium\Library 文件夹，一旦仿真模型放在了它们的新位置，确信你的 99SE 中的设计将会仿真。

在 99SE 里，在分析设置对话框中的设置会被存储在数据库内的配置文件中。当在 DXP 中第一次仿真这个设计时，如果没有配置特别的仿真设置参数，DXP 会查找并使用这个.cfg 文件。当你存储你新的 DXP 项目时，仿真设置会被写入项目文件中，而这个旧的.cfg 文件会成为多余的。

多通道设计

可能还需要更多的关注你的那些多通道的 PCB 项目。在 99SE 中，多通道设计的真实情况是做很多那些子图的拷贝，在单独的图纸符号中再次指定和引用。现在，在 DXP 中真正让你只是重复引用同一个子图，你首先需要修改你的原理图。首先，从你的项目中只留下一个子图的拷贝，删除其他所有的子图。然后，使用不同的名字但是都引用这个剩下的子图，来更新相应的原理图符号。

然而，一个聪明的策略是每一个通道只留下一个原理图符号，其他所有的都删除，然后用一个适当的重复命令替换它的名字栏。这个方法使我们只用简单的改变这一个名字栏，通道的数量就可以改变无数次。（重复命令也可以用于网络；参考“Multi-Channel Mixer.PrjPcb”例子以得到更多信息。）

当你将你的项目改变导入到 PCB，新的元件级会被创建（每一级对应每一个通道）。然而，通道级必须在“Classes”对话框中手动添加。如果你的每一个通道都有一个 PCB 布局空间，你也许希望更新你的板设计以取代部件空间的拷贝格式命令。可以只留下一个通道的布线，其他的全部撤销，然后将这个通道的元件布局和布线拷贝到剩下的通道。

存储工作

在 DXP 中 PCB 和原理图文档的格式已经被完全的重新构造。由于这个原因，被传输的文件，尤其是 PCB 文档，在你的板子被以新的二进制格式存储前，会对正常的工作目的（如网络分析）反应缓慢。

另一个节省时间的提示是重新命名你的项目文件夹或者将文件夹从产生它的数据库所在处移走。另外，任何你对传输文档所作的修改都会有被覆盖的危险，这种情况会发生于你不经意的在 DXP 中再次打开数据库文件时。因为数据库文档和子文件夹被导出在你硬盘上的同一位置，所以移动这些文件夹不会对项目的完整性造成危害，项目文件使用相关联的路径到所有被添加的文件除了那些在不同驱动器上的文件。因此，所有引用的文档都处于或低于项目文件级别意味着一个包含所有文件的文件夹可以被重新定位而不会丢失它到子文档途径。

注意

可以通过添加一个输出工作文件到你的 PCB 项目来同时配置和产生输出。这些文件和你的当前设计是交互式的，允许一键产生所有的输出工作（类似 CAM 和打印）。99SE 的 CAM Manager (.cam) 和 Power Print Configuration (.ppc) 文件不能被 DXP 认出，所以导入设计的输出需要重新配置。

最后注意，一些 99SE 用户选择 Windows 文件系统而不是数据库存储系统。一个 .ddb 文件仍然会被创建，但是它包含的是关联信息而不是文档（与 DXP 中提供的项目模式很类似）。但是因为在这个 .ddb 文件中事实上没有文档，DXP 打开它时什么也不会做。要从这一类数据库创建项目，只要在正确的位置创建一个新的项目文件，然后右击它选择将它添加到项目。你可以在选择文档添加到项目对话框中选中多个文件，只执行一个步骤就将所有的设计文件带入新的 DXP 项目。