



Innoveda社が提唱するHDL2PCBのコンセプト

デジタル・ボード設計から 回路図が消える!?

Mark Hepburn

回路に含まれる部品のピン数が増えてくると、回路図エディタを使って回路情報を入力する必然性が薄れてくる。もはや、レイアウト設計に必要なネットリストを作成するためだけに回路図を作成しているという意見もある。では、今後のボード設計における回路情報の入力手法はどうなっていくのか。ここでは米国Innoveda社が提唱する「HDL2PCB」のコンセプトについて紹介する。同社はプリント基板設計の入力方式が回路図から記述言語に移行すると見ており、そのための設計環境を開発している。(編集部)

プリント基板(PCB: printed circuit board)設計は、ここ数年で劇的に変わりました。この急激な変化をもたらした原因の一つは、プリント基板があまりにも複雑になったことです。エンジニアにとって、数百の信号を持つ4層基板を設計することは、かつては難題でした。ところが、今や、数千の信号を持ち、1,000ピン以上のLSIを搭載する20層の基板を開発する例さえ出てきました。

システムの性能を引き上げる絶え間ない努力の結果、最先端の開発現場では、高速デジタル回路に関する各種の課題が出てきています。ICパッケージの小型化やクロック周波数の上昇、立ち上がり/立ち下がり時間の短縮が、設計手法に大きな影響を与えています。最近では、1ns未満のエッジ周期に合わせて動作する部品やGHzの周波数で動くマイクロプロセッサ、微小なパッケージなどの影響で、シグナル・インテグリティ(信号の品質)に関するトラブルが顕在化しています。

ボード設計者は、引き続き高機能・高性能なシステムを追求しながら、同時に効率的な設計を行わなければならないくなっています。製品開発サイクルの短縮、シグナル・インテグリティ問題の増加、機能検証の要求、そして設計の

いっそうの複雑化が、ボード設計者に製品開発手法の変革を迫っています。

●集積度の高いデジタルLSIが急増

設計時の生産性を改善できる領域の一つは、設計入力作業です。今日のボード設計者は、一般に、回路図ベースのグラフィックス・ツール(いわゆる回路図エディタ)を設計入力の道具として用いています。回路図は、部品とその間の接続関係をグラフィックスによって表現したものです。この表現はフラット(非階層的)であり、設計の複雑さを抽象化してくれます。

回路図を使用してシステムの構造と構成要素間の接続を表現する際、主として問題となるのが「情報の密度(単位面積当たりの情報量)」です。回路図は、部品がどのように相互接続されているのかを直感的に示してくれます。しかし、現在のグラフィックスによる手法では、視野スペースの使用効率が必ずしもよくありません。さらに、現在のデジタル回路設計では、デジタル機能の多くが、事実上、回路ブロックの中に収められてしまうため、グラフィックス表現によって伝達できる情報は限られてしまいます。

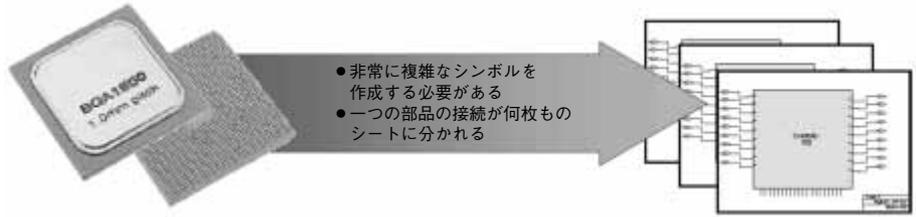
この問題は、集積度の高い部品が増えるにつれて、深刻になります。ボード上で高集積の機能を利用するためには、そのパッケージに膨大な数のI/Oを設けなければなりません。データのバンド幅が大きくなれば、200以上の接続を持つ複数の高速バスが必要になります。

こうした現実が、結果的にピン数のきわめて多いデバイスを生み、さらに、これらのピン数の多いデバイスを使用する機会が着実に増えています。3、4年前の大型デバイスは通常500ピン程度でした。現在では、1,200~1,600ピン

〔図1〕

ピン数の多い部品が増えると回路図シートの枚数が増える

ボード設計では、よりいっそう大きく、ピン数の多いデバイスを使用するようになった。回路図ベースの手法を利用して設計入力を行うと、複雑なシンボルとその接続を複数のシートにまたがって配置させなければならない。結果として、回路の中のデータの追跡と管理がきわめて難しくなる。



のデバイスさえ登場しています。

●ピン数の増加がボード設計にもたらした二つの問題

このような大型デバイスの登場は、現在の回路図を中心とするボード設計の世界に、二つの問題を引き起こしています(図1)。

第1に、必要なI/Oを表現するための回路図シンボル(以下、シンボル)をどのように作成するかという問題です。このような部品を回路図上に配置するには、複数のシンボルに分割して表示するよりほかに方法がありません。一つのデバイスが10個以上のシンボルに分けて示されることさえあります。この場合、I/Oの意味が直感的にわかるように適切にシンボルを分割しなければならず、注意が必要です。

例えば、回路設計者は入力の段階では、V_{cc}(電源)とGND(グラウンド)の接続がデータバスと混在することを望みません。このことは、企業レベルのライブラリ管理(構築)の作業に多くの問題を提起しています。ライブラリ管理と回路設計は別工程であるため、設計者が望んでいるとおりの形状のシンボルができ上がってくることはまずありません。ライブラリ管理の工程と回路設計工程の間に連続性がないため、結局、設計者自身がシンボルの作成や検診といった骨の折れるしごとを引き受けざるをえなくなります。その結果、ライブラリの一貫性の確保と管理に膨大な手間がかかることになるのです。

第2の問題は、回路図を何十枚ものシートに分割しなければならないことです。いったんシンボルができ上がると、設計者は部品間の接続を定義するために、グラフィックス環境(回路図エディタ)を利用します。設計者は複数のさまざまなシンボルの間を接続する必要がありますが、一つのシンボルだけでも大きくて複雑である場合が多く、そのシンボルだけで回路図シートの大部分を占めてしまうことさえあります。例えば、1枚のシートに一つのシンボルしか書かれていないという場合もよくあります。

このため、設計者はオフシート・コネクタと呼ばれる記号でシート間の接続を定義したり、回路を何十枚ものシートに分割せざるをえません。このようなオフシート・コネクタの利用は構造的には正しいものの、各コネクタに関する情報が複数のシートにまたがってしまうため、ユーザは接続情報を視覚的に一括して確認することができません。

●ネットリスト生成のためだけに回路図をかいている

部品間の接続の定義が終わると、設計者はプリント基板のレイアウト工程で必要となる属性情報や設計制約などを注釈として回路図に付加します。注釈が付けられると、プリント基板用のネットリストが生成され、次の工程に渡されます。

ボード上の一部または全体について機能シミュレーションを行う必要がある場合、設計者はその作業のためだけに、貴重な時間を費やして回路図をVHDLまたはVerilog HDLなどのHDL(hardware description language)に変換します。今日の回路図エディタはこうした変換機能を搭載していますが、シミュレーションのために回路図を人手で修正しなければならない箇所もあります。通常、こうした工程では、設計者がアナログ部品を取り除いたり、テストベンチに接続したり、シミュレーション・モデルの関連付けを行います。

これが終わると、設計者は変換されたHDLが正しく当初の意図を反映しているかどうかを目視で確認します。これは単調で非常にエラーが発生しやすい作業です。そして、検証作業中にエラーを修正するための変更を加え、次に、その変更をもとの回路に反映させなければなりません。多くの場合、シミュレーションを導入すると、設計フローが複雑化してしまい、その結果、作業が二度手間になるケースがあります。

一歩下がって設計工程全体を眺めてみると、大規模なデジタル部品で構成される設計の場合、回路図を使用する