

Chapter 13 Circuit Subdivision

Introduction

Sonnet では、大きな回路をより小さなサブブロックにいくらかでも分割することができ、その回路全体の応答を生成するために、ネストブロックの中でその結果を結びつけることができます。この方法により、正確な解を得ながらも、回路の解析に必要な処理時間と必要メモリを大幅に減らすことができます。

行列を解く時間は N^3 に比例するので、回路のサブセクション数は処理時間を決定する上で最も重要な要素の 1 つです。回路のサブディビジョンがどのように処理時間を減らすのかを説明するために、それぞれがサブブロックの 2 分の 1 のサブセクション数を持つ 2 つのサブブロックについて考察してみます。行列を解く全体の時間はここで 4 倍速くなります。

$$2(N/2)^3 = N^3/4$$

回路のサブディビジョンでは、回路をより少ないサブセクションの、より小さな部分に分け、解析の処理時間とメモリを少なくでき、これがこの技法で利点です。このトレードオフは、解析に誤差を招きます。しかし、適切に回路をサブディバイドすることによって、処理時間を減らせる一方で、誤差を最小限にすることができます。

回路はサブディビジョンラインに渡って電磁的結合がないところで分割されなくてはなりません。強く結合が生じている所は、サブプロジェクトの内側に含めなくてはなりません。このように、回路内で強く結合している部分すべてを考慮します。回路を注意してサブディバイドすれば、結果の精度は大変高くなります。

回路のサブディビジョンはすべての設計に適切であるというわけではありませんが、サブディビジョンが応用できる大きな回路（周波数あたり 5-10 分の処理時間がかかるもの）の場合は、処理効率がかなり上がります。

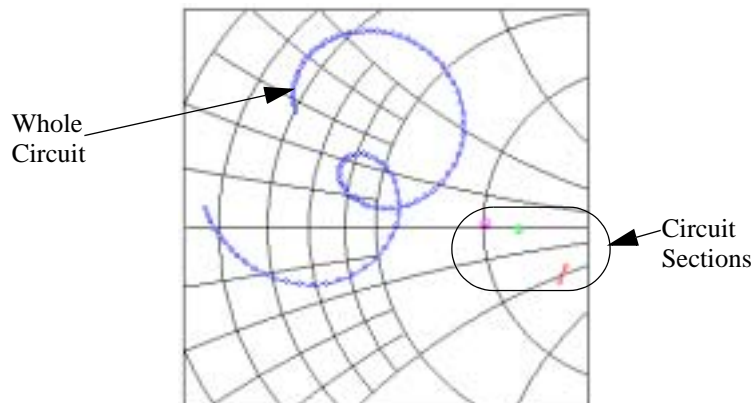
回路のサブディビジョンのもう 1 つの利点は、ネスト解析における周波数の補間です。回路全体をシミュレートするために、回路のサブプロジェクトの応答データを連結する時に、ネストを使います。応答データが周波数帯域に渡って、大きく変わらないようにサブプロジェクトを選べば、そのサブプロジェクトでは大変少ない周波数点を計算するだけですみます。従って、サイズが小さいので時間とメモリが少なくすむというだけでなく、より小さな回路をわずかな周波数点で解析することもできます。ネストファイルで補間すると、図形プロジェクトで周波数点のデータを計算するよりも、ずっと短い時間ですみます。

Chapter 13 Circuit Subdivision

以下はこの方法を用いた典型的な利点を示した例です。

Large Circuit		30 minutes/frequency X 25 frequencies = 750 minutes total
Netlist (Subdivided Large Circuit)		
Small Circuit	Small Circuit	Small Circuit
4 mins/freq X 5 freq 20 minutes	3 mins/freq X 5 freq 15 minutes	4 mins/freq X 5 freq 20 minutes
Netlist Total Analysis Time = 55 minutes - 14X faster		

ネットリストの解析が実行されると、*em* はサブプロジェクトで設定されていない周波数でのデータをシミュレーションするために補間します。各サブプロジェクトは、解析全体と同じ最小と最高周波数で解析されるべきです。また、これらの値の間にあたる周波数で、妥当なデータの補間ができるようにその間にある十分な点で解析されるべきです。以下の図からわかるように、回路全体に妥当な応答データを得るためには多くの周波数が必要ですが、その回路全体の小さな部分の正確なデータを得るためには、応答データにそれほど変化のないかなり少ない周波数で十分です。



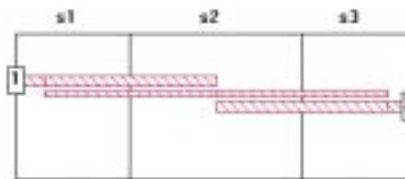
しかし、ある状況では周波数帯域を同じ周波数刻みですべての部分により高精度で解析する必要があるかもしれません。補間は、周波数帯域に渡って、サブプロシエクトの解析時間がかなりかかるとわかっている時に使うのが最良です。

Circuit Subdivision in Sonnet

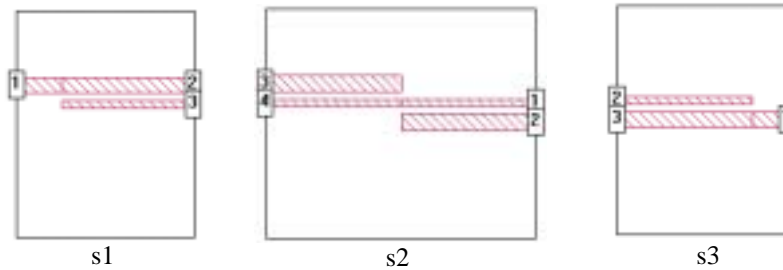
Sonnet では、回路のサブディビジョンによって project editor の中でシオトリ (図形) にサブディビジョンの行を挿入することができます。これらのサブディビジョンの行は subdivide コマンドが図形サブプロシエクトをつくる部分を成します。subdivide コマンドを選択すると、このソフトウェアはメインのネットリストファイルと

Chapter 13 Circuit Subdivision

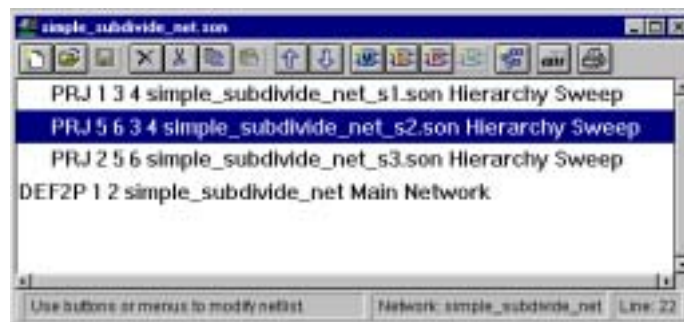
図形サブプロジェクトを作成します。メインのネットリストはサブプロジェクトを連結し、これによりこのネットリストの応答データは、サブプロジェクトの応答データの代わりとして使えます。



Source Circuit with Subdivision Lines Added



Generated Subprojects



Generated Main Netlist

メインの回路にパラメータまたは寸法が含まれていると、それらはサブディビジョンの過程で除去されてしまうことも知っておく必要があります。サブプロジェクトが作成された後に、図形の何かのパラメータを入力することができます。実際には、サブプロジェクトの1つの中でパラメータを使って、メインのネットリストプロジェクト上でパラメータ化を実行することは可能です。

解析の1つの方法としての回路のサデ化シヨンは、一般的に次のように実行されます。

- 1 サデ化シヨンの前にできるだけ多くの回路のプパティ (属性) を入力します。誘電体層、誘電体ブリックと金属タイプ、グリッドの大きさ、Top までの高さ等が、作成された図形サブプロジェクトによって継承されます。
- 2 回路のどこをサデ化シヨンするかを決めます。サデ化シヨンラインに渡って電磁的結合がある接合部で、回路を分割しないようにするためには、このステップはしばしば専門知識と経験が必要とされます。
- 3 Project editor でサデ化シヨンラインを作成します。これらのラインはサブプロジェクトを作成するために使用します。図形プロジェクトは回路の各線路部分について作成されます。これらの図形プロジェクトは、少ないメモリを使ってより速く解析ができる、かなり小さい図形になっています。
- 4 ネットリストの補間機能を利用したいと考えている場合には、サデ化シヨンする前にプロジェクトの所望する周波数帯域全体を粗い周波数刻みとして、解析周波数コントロールを設定します。ネットリストと図形サブプロジェクトはすべてこれらの周波数の指定を継承します。このステップで Analysis Setup ダイアログボックスの中にこれらの周波数コントロールを入力しておく、これらを個々のサブプロジェクトに入力する手間が省けます。
- 5 Project editor の中で回路をサデ化シヨンし、サブプロジェクトとネットリストプロジェクトを作成します。ネットリストは回路全体に相当するネットワークの中で個々のサブプロジェクトを連結し、より大きな回路につなげる必要があれば、ポートと参照面がサブプロジェクトへ追加されます。
- 6 必要であれば、図形をきれいな形に整えるためにサブプロジェクトを編集します。ここで調整が可能なのは、2 の n 乗の切数のボックスにすること、グリッドサイズの調整、ブリックの z パーティションの設定、周波数スweep の指定の変更そしてパラメタの追加です。



TIP

ネットリストの中でサブプロジェクトにパラメタを追加する場合は、パラメタはそのネットリストの中では自動的に表示されません。従って、メインネットリストを保存し、それを再オープンして、パラメタを表示し、それらを編集できるようにする必要があります。

- 7 サブプロジェクトが既に粗い周波数スイープを解析するように設定されている場合は、ネットリストの中で解析コントロールを設定し、所望の解析周波数の完全なセットを使います。マスターのネットリストプロジェクト上で、解析が実行されると、*em* はサブプロジェクトの中の周波数点の間を補間し、処理時間を削減します。

Hierarchy Sweep を使用することも可能です。これはマスターのネットリストの中で設定された周波数帯域をすべてのサブプロジェクトの解析上に設定するものです。これは project editor で Analysis Setup ダイアログボックスの中の Hierarchy Sweep オプションを設定することによって実行されます。

- 8 ネットリストプロジェクトを解析します。ネットリストプロジェクトのデータ応答は回路全体に使われる解析結果を提供します。
- 9 解析がその結果のサブプロジェクト上で終了した後に、応答データをチェックし、正確に補間されたデータを提供するために十分な周波数点で計算されたことを検証することはよい考えです。

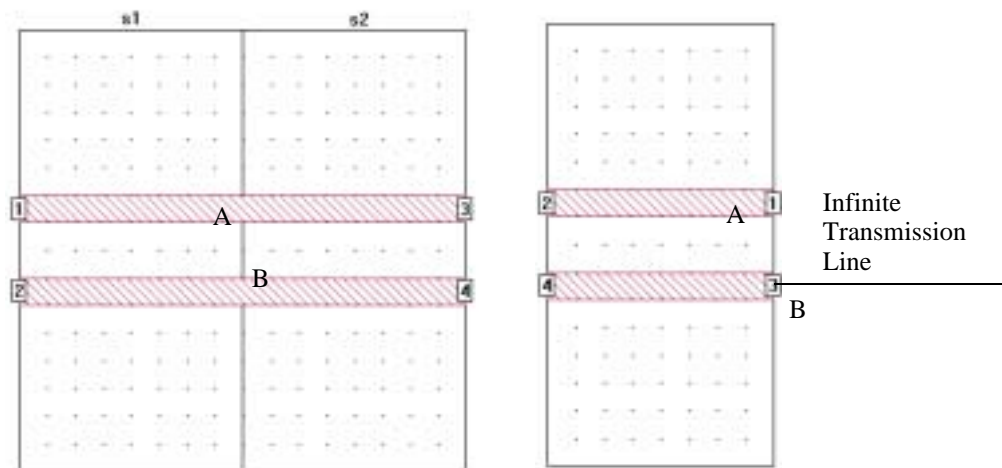
ネットリストプロジェクトにネットリストサブプロジェクトを含むことは可能なので、“二重に”サブディビジョンを使用することが可能です。一番はじめの回路をサブディバイドした後に、その結果の1つの図形サブプロジェクトの上でサブディビジョンを使用できます。この場合は、適切な PRJ 行上にある名前を古い図形サブプロジェクトから新しいネットリストのサブプロジェクトへ変える必要があるでしょう。

Choosing Subdivision Line Placement

前述したように、回路のサブディビジョンを使う上で難しい点はその回路を分割するためにどこにサブディビジョンラインを置くかということです。サブディビジョンラインは、電磁的結合を無視してもよい図形と図形の上に置くべきです。強い結合や電流が急峻に流れている回路では、それらは個々のサブプロジェクトの内側に置くべきです。

Sonnet におけるポートの不連続部分のデインパッティングは、基本的にはポートのあるところで無限に長い伝送線路を接続することによって行われます。デインパッティングを行うと、精度を損なうことなく伝送線路をサブdivideすることができます。これにはマイクロストリップ線路、ストリップ線路、コプレーナを含む結合線路が含まれます。これを以下に図示します。

左下の回路は、結合伝送線路から構成されています。これは、大変シンプルな回路なので、サブdiviジョンはできませんが、この原則を説明するのに大変便利です。回路はサブdivideされる時は、2つのサブプロジェクトに分割されます。このサブプロジェクトは両方とも右に示した回路に似ています。ポートがデインパットされると、ポートの不連続部は、無限の伝送線路として接続されるので、A点とB点間の結合が問題となります。

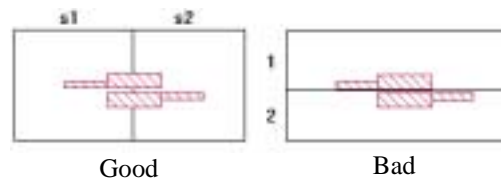


サブdiviジョンラインにわたって結合がある場所を避けることが大切です。サブdiviジョンラインは斜めの図形の縁を分割することはできません。以下に、サブdiviジョンラインのよい置き方と悪い置き方を説明します。

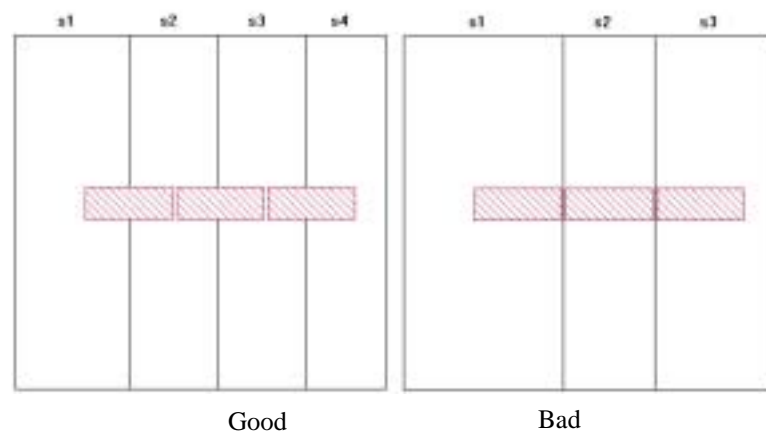
Good and Bad Placements of Subdivision Lines

この節では、回路にサブディビジョンラインを置く上での良い置き方と悪い（そしていくつかの間違った）置き方を説明します。1本または1本以上の伝送線路に垂直にサブディビジョンラインを置くのがラインの置き方としては一般的により方法です。

1つめの例は、一对の結合線路です。前述のとおり、左図のように結合線路を分割すると、結果の精度がほとんど損なわれません。しかし、右図では軸に沿って結合している部分を分割しており、ここではかなり相互作用を生じています。サブプロジェクトではこの結合を考慮に入れる方法がないので悪いデータを生じてしまいます。



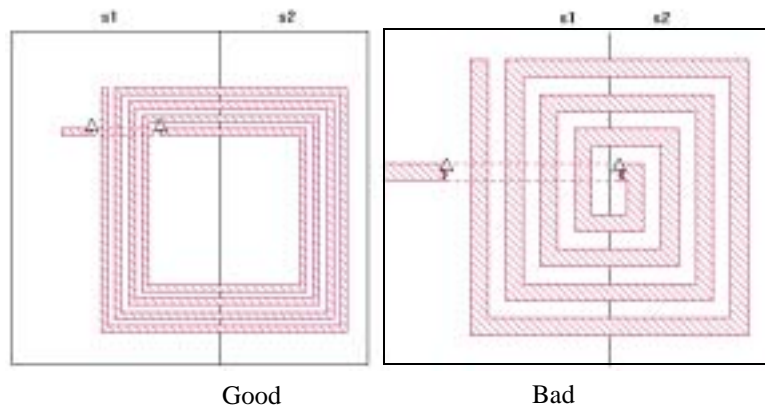
2 つめの例は、一連の共振器の分割のしかたです。この種類の構造には隣接する共振器の間のギャップ部分に強い結合があります。左図では、サグデーションラインは共振器間の結合を妨げないのでよい例となります。右図では結果のサブプロジェクトに共振器間の結合を含まないのでよくない例となります。



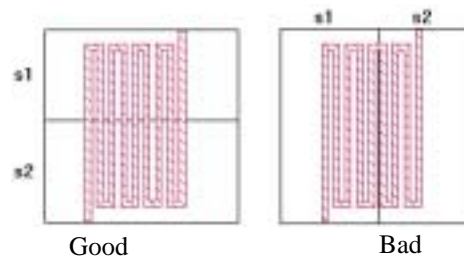
3 つめの例は四角いスパイルです。左図ではスパイルが分割されている位置が基本的には1つのまとまった結合伝送線路であり、サグデーションラインがこの伝送線路に垂直になっているのでよい例です。ここでは、

Chapter 13 Circuit Subdivision

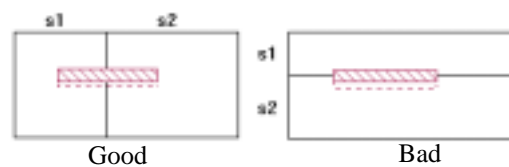
ｽﾌﾟｲｰﾙの左側と右側は十分に離れているので、結合が無視できます。右図ではｽﾌﾟｲｰﾙの左側にある線路が、右側の線路と強く結合しているので悪い例です。



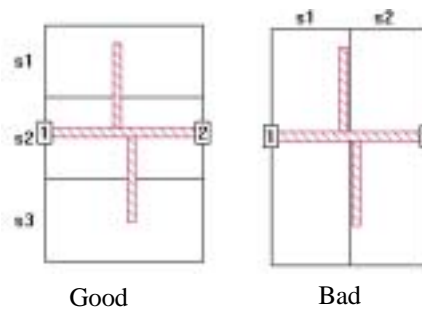
左図のｽﾀﾝﾀﾞｰﾄﾞ線路では、上部の曲がり部が下部の曲がり部から十分に離れており、その間の結合は無視でき、Sonnet では正確な答えを出せるように分割されています。右図では2つの接近した伝送線路の間の結合がサブリミナルによって削除されてしまうので、結果が不正確になってしまいます。



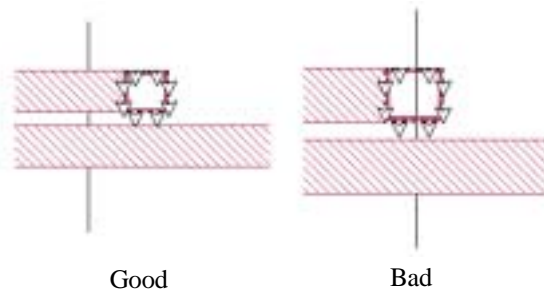
以下の回路は、異なる2つの層の上にある結合伝送線路です。同じことを言うことになってますが、伝送線路に垂直にサグデ化シヨライを置くのは正しく、平行に置くのは正しい方法ではありません。サグデ化シヨンは、サグデイルト部に渡って結合が無視できるものである限り、複数層の構造に有効です。



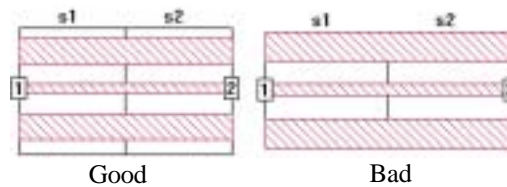
左図のダブルスタグ回路では、サグデ化シヨライは電流の流れる方向に垂直であり、不連続部からは離れている図形を分割しています。しかし、右図の回路では結合しているかもしれない2つのスタグの根元部分を分割しているサグデ化シヨライを示しています。



右図の回路のサテ化シヨナインは、この回路が層と層の間の via のまん中で分割されているので、間違った置き方です。一般的にはサテ化シヨナインは via のような不連続部の上には決して置かないでください。左図のサテ化シヨナインは正しい置き方です。

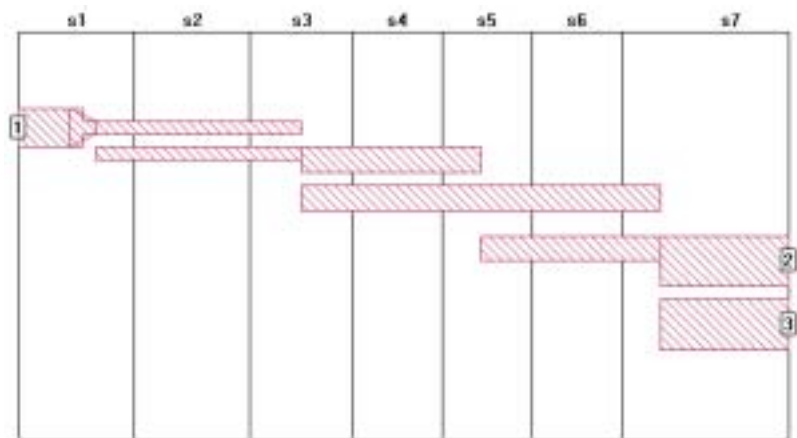


右図の回路のサテ化シヨナインは、ボックス壁で図形を分割しており、これはサテ化シヨナインでは間違った置き方です。ボックス壁にくっついていて図形は伝送線路のようなはたらきをしないのでサテ化シヨナインするのは間違っています。また、サテ化シヨナイン中に新しく追加されたポートは、ボックス壁に対してショート（短絡）してしまいます。左図の回路は、上部と下部の図形の間は、上部と下部のボックス壁に接触していないので正しい例です。

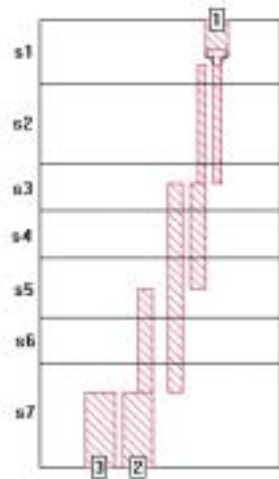


Subdivision Line Orientation

Subdivider は水平軸上でも垂直軸上でも回路を分割することができますが、方向を混合しないでください。回路を分割する方向の選択は、回路の構造によって異なります。以下に垂直方向を使う典型的な回路と水平方向を使う例を示します。



Example of Vertical Subdividers



Example of Horizontal Subdividers

前述したサブディビジョンを二重に使える、両方向を使うことができます。最初にメインの回路をサブディバイドする時に、サブディビジョンラインの1つの方向を選びます。次にその結果できた図形サブプロジェクトの上で、回路のサブディビジョンを実行しますが、この時には最初のサブディビジョンラインと反対方向を使います。

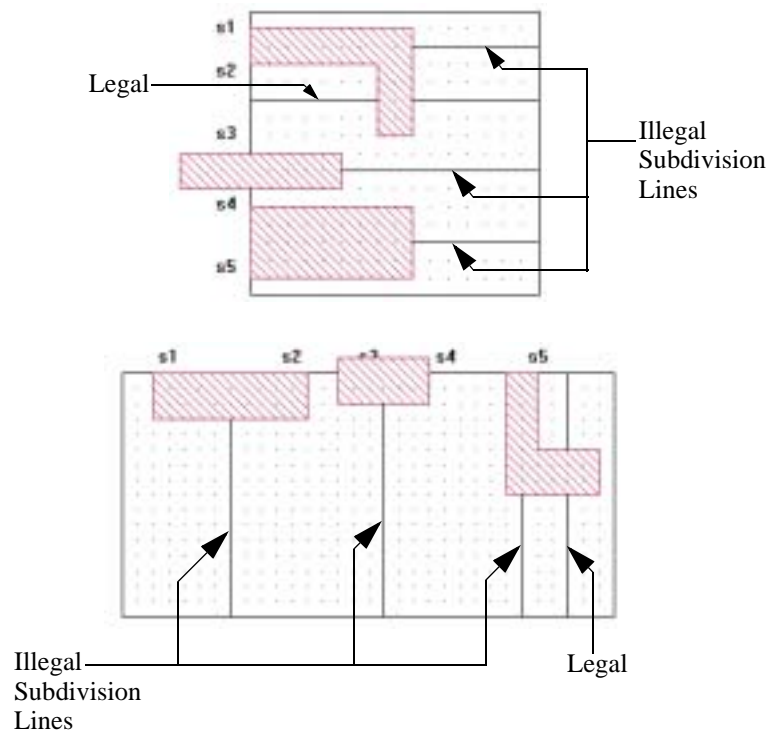
図形プロジェクトに subdivider を追加する前に、必ず回路の仕様が完了していることを確認してください。サブディビジョンを実行する時に作成されたサブプロジェクトは、ソースプロジェクトからそのプロパティを継承します。セルサイズ、金属タイプ、誘電体層のプロパティ、誘電体ブリック、金属パッドなどのプロパティがすべて、その後できるサブプロジェクトで使用されます。

回路に subdivider を置くと、ユーザがクリックした点を通る水平面または垂直面に subdivider を表わす1本の線が現れます。その結果できた回路のセクションに自動的にパッドがつけられます。サブディビジョンセクションは、方向によって左から右、または上から下へとパッドがつけられます。これらのパッドは常に連続した数字であり、修正はできません。

いったん subdivider が回路に追加されると、図形の中で他のオブジェクトを編集するように、subdivider を編集することができます。Subdivider をクリックし、移動できます。Object Visibility ダイアログボックスや Selection Filter ダイアログボックスの中で、サブディビジョンラインとパッドの表示や選択をコントロールすることができます。

以下は、サブディビジョンラインの条件です。

- グリッドから離れてはいけません。
- サブディビジョンラインにかかる結合のある場所においてははいけません。
- 図形の縁と同一線上に並んではいけない。
- 斜めの図形の縁を分割してはいけません。
- ポートを分割してはいけません。
- Symmetry (対称) 指定の下側に置いてはいけません。
- ボックス壁の所で図形を分割してはいけません。次の図を参照。



回路に所望のすべての subdivider を追加し終わったら、サブディビジョンを実行する前にプロジェクトファイルを保存する必要があります。

Setting Up Circuit Properties

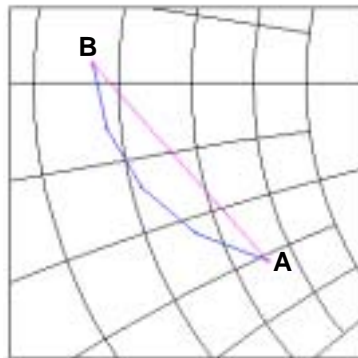
サブディバイドによって作成された図形サブプロジェクトは、ソースプロジェクトからそのプロパティを継承するので、サブディバイドの実行前に回路に必要なすべての属性の入力を終了している必要があります。これには、誘電体層（ボックスの上部の高さも含みます）、上部と下部のボックスの金属、金属と誘電体ブリックの材質、サイズ、ボックスのサイズの定義が含まれます。ここで入力しておくと、各サブプロジェクトにおいて、これらの値を入力する手間が省けます。

Setting Up the Coarse Step Size Frequency Sweep

マスターのネットリストプロジェクトを解析する時に、応答データを得るために補間機能を使おうとする場合には、その回路をサブdivideする前に、図形サブdivideを解析したいと思っている粗い周波数スイープで入力する必要があります。これらの周波数は、回路全体の解析周波数と同じ周波数の範囲を使いますが、より粗いステップで行います。サブdivideは、解析全体と同じ最小と最大周波数で解析されるべきで、また応答の妥当な補間がなされるようにその間を十分な周波数点で解析するべきです。

サブdivideより先に粗い周波数スイープを入力することによって subdivide コマンドによって作成されたマスターのネットリストと図形サブdivideは、すべて解析の設定を継承します。サブdivideした後に、これを解析する前にマスターのネットリストの中に所望するより細かい周波数スイープを入力する必要があります。更に、Hierarchy Sweepを終了する必要があります。

以下の図は5点の周波数点で解析した回路のミスマッチと、その横に2点のみで解析した同じ回路のミスマッチを示しています。ご覧のように、2つのデータ点のみを使用したものは、5点を使用したものよりも補間の誤差が多くなってしまいます。2つのデータ点が受け入れられるか否かはその点と点の接近具合によります。AとBが大変接近していたら、この2つのデータ点で十分です。もしAとBが離れていれば、5つまたはそれ以上のデータ点を使うべきです。



サブdivideの応答データのミスマッチを見て、十分な周波数点を選んだか、そしてそれによって補間されたどのデータも妥当な精度でデータが計算されているかを確認するのはよい考えです。

Subdividing Your Circuit

実際に回路を別々の図形サブプロジェクトやマスターのネットリストプロジェクトへサブディバイドするのは、ソフトウェアによって実行されます。ユーザはマスターのネットリストと図形サブプロジェクトの、所望の名前を入力します。また、無損失の金属の給電線をサブプロジェクトの中で生成されたポートに自動的に追加することもできます。

ボックス壁と不連続部が相互作用することを避けるために、もとの回路の部分に含まれた不連続部をボックス壁から離れた所に移動する必要がある時には、給電線を追加すべきです。給電線の使用はオプションです。給電線を追加することを選択する場合は、ソフトウェアによって算出された推奨の長さを入力するか、またはご自分で入力することができます。デフォルトでは、ソフトウェアは推奨の長さを使用して給電線を作成します。

サブディバイドが実行されると、Sonnet はユーザが subdivider を置いた回路の各セクションに 1 つの図形サブプロジェクトを作成します。Sonnet はまた、各図形サブプロジェクトを結合したマスターのネットリストを作成して、ユーザがサブディバイドしたもとの図形プロジェクトと同じ回路を作ります。

各図形サブプロジェクトは、もとの回路のプロパティを使用します。セルサイズ、誘電体層、誘電体および金属の材質、解析の設定などです。従って、すべての図形サブプロジェクトは、指定された解析周波数と同じ解析の設定を含んでいます。

Analyzing Your Subdivided Circuit

所望の応答データを得るために、マスターのネットリストの解析の設定を編集して、すべての所望の解析周波数が指定されるようにします。各図形サブプロジェクトは、より粗い周波数刻みの解析周波数で設定されます。ネットリストが解析される時には、*em* はまず図形プロジェクトの解析を実行し、ネットワークの各部分のための応答データを生成します。次にネットワーク全体の解析が行われます。*Em* は図形サブプロジェクトの解析によって得られる周波数の間を補間し、周波数点のデータを生成します。

Chapter 13 Circuit Subdivision

正しくサブディバイドされると、ネットリストの解析の結果は、より少ないリソースで回路を扱うという難しい問題に正確な解決法を提供してくれます。回路のサブディビジョンの使い方は、**Sonnet Supplemental Tutorials** の第 3 章の “Circuit Subdivision Tutorial ” をご覧ください。