

自作回路をSUBCKTにし 暗号化してリリースする手順

- 自作回路の
SUBCKT化
1. 自作回路をキャプチャーする
 2. 外部端子を付ける
 3. シンボル（箱）を作る
 4. シンボルを編集する
 5. 回路のNet-List を作る
 6. SUBCKT の形式に編集する
- 暗号化
7. SUBCKTファイルを暗号化する
 8. 暗号化されたファイルを編集する

— SANKYOSHA — FAE : Michio Shibuya — 2020/01/10... 渋谷道雄・和田謙太郎 —

自作回路のキャプチャー（簡単な例題）

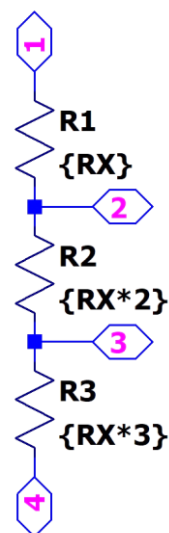
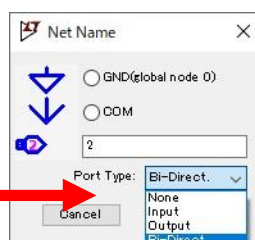
回路図のキャプチャー

抵抗器3本で分圧する回路を作る。

抵抗ラダーは<R>-<2R>-<3R>にならべ、基本となる抵抗値（右の回路図中のRX）はパラメータとして後から設定できるようにする。

外部端子を付ける

SUBCKTとして外部と接続するピンにはラベルを付け、ラベル属性として「Port Type」(Input, Output, Bi-Direct の3種類)のどれかを決めておかなければならない。



とりあえず、回路図名を「RX_tap.asc」として保存

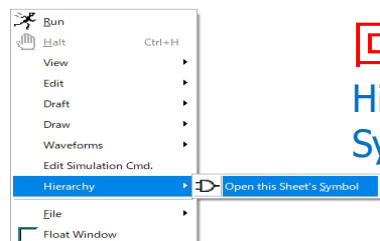
— SANKYOSHA — FAE : Michio Shibuya

Symbol (箱) を作る (1)



メニューバーから・・・

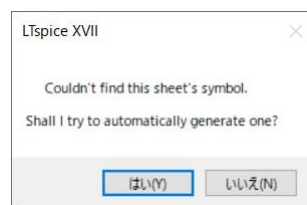
Hierarchy からプルダウンし、「Open this Sheet's Symbol」を選択。



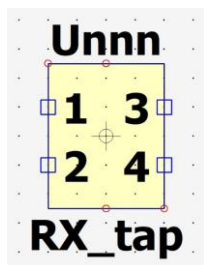
回路図中で、マウスの右クリック・・・

Hierarchy からプルダウンし、「Open this Sheet's Symbol」を選択。

初めてこの回路のシンボルを作るときは、右の図のようなメッセージが表示されるので「はい」をクリックし、新規に生成する。



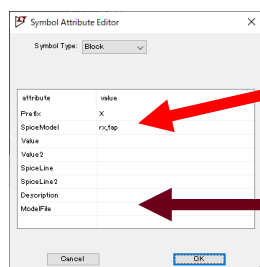
Symbol (箱) を作る (2)



箱が表示されたら・・・

(ファイル名は元の回路図名で、拡張子が「*.asy」になっている。)・・・このウインドウの中でマウスの右クリックをし、Attributes からEdit Attributes をクリックする。

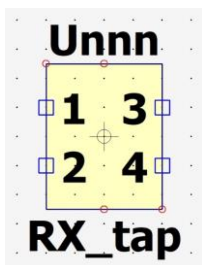
(ピンの順序は、原則として入力ピンが左側、出力ピンが右側になることが多いが、絶対的な決まりではない。)



Symbol Attribute Editor で、Prefixの項目に“X”、SpiceModelの項目に“RX_tap” (拡張子は付けない) を入力する。その他の項目はブランク (空白) にする。

操作手順によっては、ModelFileの欄に、Symbolを生成した時点のフォルダ名 <フルパス表示> +モデル名が入っているので、その行をダブルクリックをしてハイライトにし、消す (ブランクにする)。

Symbol (箱) を作る (3)



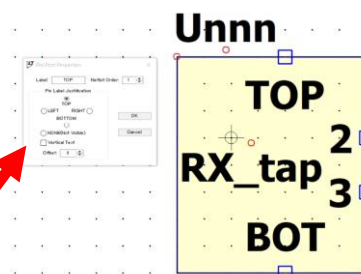
箱の編集・・・

Symbolは回路図中に取り込んで利用するとき、使い勝手がいいように、ピンの位置や、箱の形、文字の表示・・・など、カスタマイズしておくことよ。長方形以外の形にする場合には、Asyの編集画面の中で、Drawを駆使して新規の図形を作ることができる。ここでは長方形のまま編集する例を示す。

長方形の左上・右下の赤い「アンカー・マーク」はドラッグ（ホットキーF8）でつまむ（マウスで囲む）ことでドラッグできるようになり、適当な場所でクリックすることで固定できる。図形のアンカーポイントは、グリッドの1/16単位でスナップするので、ピンのあるラインはグリッド上に一致させるように注意する。ピン位置もドラッグできる。

ピン（正方形）の上でマウスの右クリックをすると、ピン位置に対する文字の位置を変更できる。

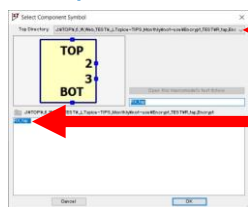
ピンの名称は、この窓の「Label:」の中の文字列を編集すればよい。



出来上がったSymbolを試してみる (1)

New Schematic で回路図画面を開き・・・

Component (ホットキーF2) で今作ったSUBCKT (の箱) をまず配置する。

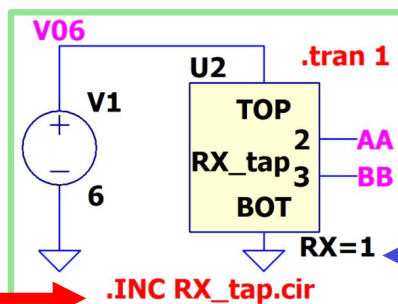


この選択窓で、先ほどSymbolを保存したフォルダーを選択する。

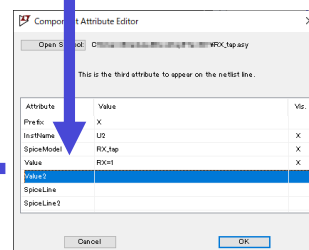


すると、先ほど保存した部品名が表示されるので、それをクリックして選択し、[OK]する。

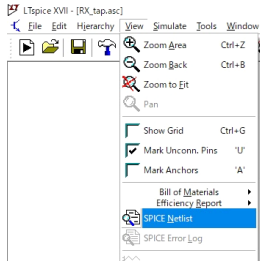
回路図を完成させ、先ほどのSymbolを保存したフォルダーと同じところに回路図も保存する。モデルファイルを「.INC」コマンドでインクルードする。この段階では、まだ、「RX_tap.cir」ファイルを用意していないので、このままではシミュレーションできない（次ページ以降で解説する）。



内部パラメータのRXを定義する。回路図上のシンボル（箱）の上でマウスの右クリックをし、「Component Attribute Editor」を開き、「Value」などの枠にパラメータを定義する。

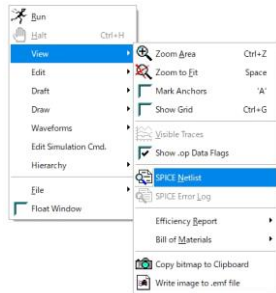


Net-List を作る



回路図表示にしてメニューバーから・・・
View からプルダウンし、「SPICE Netlist」を選択。

あるいは・・・



回路図中で、マウスの右クリック・・・
View からプルダウンし、「SPICE Netlist」を選択。

J:¥・・・回路図のあるPath・・・

```
R1 1 2 {RX}
R2 2 3 {RX*2}
R3 3 4 {RX*3}
.backanno
.end
```

← 出力された NetList

表示されたビューワの上で右クリックし、「Edit as Independent Netlist」を選択すると、「名前を付けて保存」ウィンドウがあらわれてネットリストを保存することができる。

Net-List の編集・・・SUBCKTにする

```
.SUBCKT RX_tap 1 2 3 4
R1 1 2 {RX}
R2 2 3 {RX*2}
R3 3 4 {RX*3}
.ends
```

End SUBCKT の意味

1行目に「.SUBCKT」ではじめ、そのあとに1スペースをおき、そのあとに<サブサーキット名>を書く。さらに、スペースで区切りながら、外部に取り出すピン番号を並べる。

終わりから2行目の「.backanno」の行は消し、最後の「.end」を「.ends」に書き換える。
正式には「.ends RX_tap」のようにサブ・サーキット名を添える。(サブサーキット名は省略可能)

このファイルを「モデルファイル名」を付けて保存する。

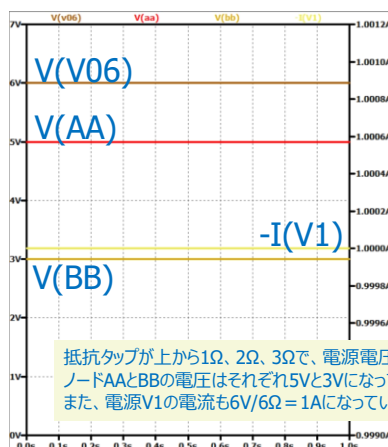
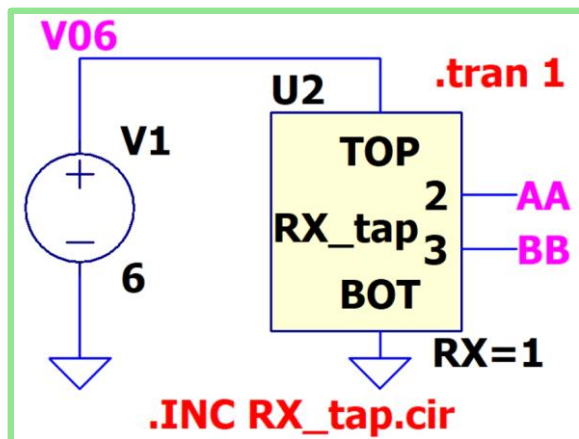
ファイル名の拡張子には「*.sub」や「*.cir」など、サブ・サーキット(モデル)ファイルと分かるようにする。ここでは「RX_tap.cir」として保存した。

最終的に暗号化すると、オリジナル・ファイルは上書きされてしまうので、バックアップを残しておくこと。

出来上がったSymbolを試してみる（2）

いよいよ、すべての準備が整ったので……

シミュレーションの実行



暗号化手順

Windowsの「コマンド プロンプト」を開く……

モデルファイルを保存したディレクトリーに移動し……

Windowsの環境変数が設定されていない場合は以下のコマンドでPATHを設定

```
>path c:\Program Files\LTC\LTspiceXVII
```

次に以下を入力すると、

```
J:>XVIIx64.exe -encrypt RX_tap.cir
```



…と表示され、暗号化を行う。
回路規模にもよるが、数分から数十分か
かることもある。完了すると……



[OK]すると、もとの「*.cir」ファイルは暗号化されたファイルで置き換えられる。

暗号化されたファイルを編集する

最初の9行のコメント
部分は編集しない

この間にコメント行(「*」で始
まる)によって、説明を加える

*Begin: 以下暗号化された
部分にも手を加えない

```
* LTspice Encrypted File
*
* This encrypted file has been supplied by a 3rd
* party vendor that does not wish to publicize
* the technology used to implement this library.
*
* Permission is granted to use this file for
* simulations but not to reverse engineer its
* contents.
*
* =====
* SUBCKT call name : RX_tap ← SUBCKTを呼び出す名称は必須
* Parameter : RX=** is needed; ← 内部パラメータがある場合
* tap TOP-1RX-2RX-3RX-BOT ← にはその名称を書く
* =====
* Begin:
E4 A1 5B 05 58 19 7F E4 24 D4 E8 BC 92 29 3C 36
E1 A0 28 4C 03 7A 49 B2 2F 22 00 31 47 E9 6F D3
.....以下、暗号化されたコードが続く
```

暗号化されたModelを試してみる

先ほどの確認回路のままで.....

シミュレーションの実行...まったく同じ結果になっていることが確認できる。
内部パラメータ「RX=...」も編集できることも確認できる。

