

## 第6章 プリミティブ素子の作成法とネットリスト出力

本章では、NS-Draw におけるプリミティブ素子の作成法と、ネットリスト出力について説明します。NS-Draw においては、各回路図は、スキーマと呼ばれる回路図の実体と、上位階層で参照されるシンボル図の2つのデータをセットで持ちます。図1に1ビット全加算器の例を示します。図1(a)のスキーマで記述される回路ブロックに、図1(b)のようなシンボル図を作成することで、上位の回路から、モジュールとして呼び出すことが可能となり、多ビット加算回路のような階層構造をもった回路図を作成することができます。

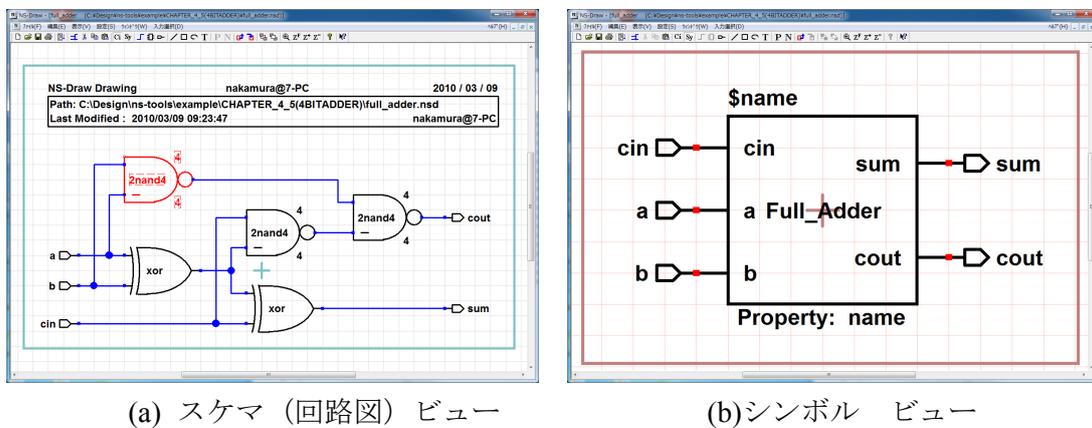


図1 回路図の例1 (1ビット全加算器)

さらに図1(a)のスキーマで、NAND ゲートを選択し、e キーにより階層を降りると、図2の2入力 NAND ゲートの回路図が開きます。この2入力 NAND ゲートも、やはりスキーマ図とシンボル図の組から構成されています。

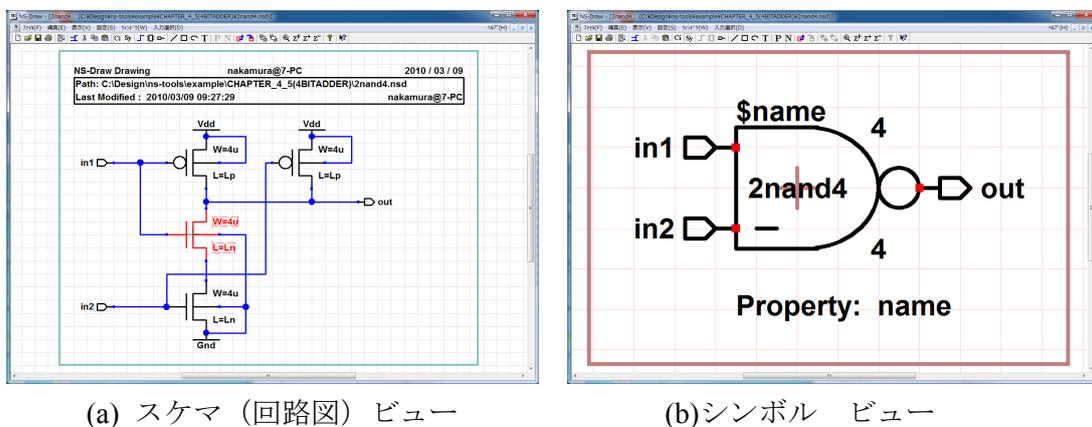


図2 回路図の例2 (2入力 NAND ゲート)

ここで、図 2 (a) のスキーマ図において、MOSFET を選択して、さらに階層を降りると MOSFET 素子の回路図に到達します。図 3 に NMOSFET の NS-Draw 回路図を示します。この NMOS 素子の回路図はシンボル図のみを持ち、スキーマ図はありません。このようなシンボル図のみで構成される回路は、回路図階層の最下層になり、これより下の階層には降りることができません。これを NS-Draw ではプリミティブ素子と呼んでいます。ネットリスト出力時には、各階層の回路図は階層モジュール (SPICE では、サブサーキット) として、順次階層的に出力されますが、このプリミティブ素子では、このプリミティブ素子内のネットリスト指定子で指定された記述が、ネットリストに出力されます。図 3 の例では、

“NetList: spice Format: M\$name? \$d \$g \$s \$ss \$model W=\$W L=\$L AD=\$AD AS=\$AS PD=\$PS PS=\$PS NRD=\$NRD NRS=\$NRS” という部分がこれに相当します。

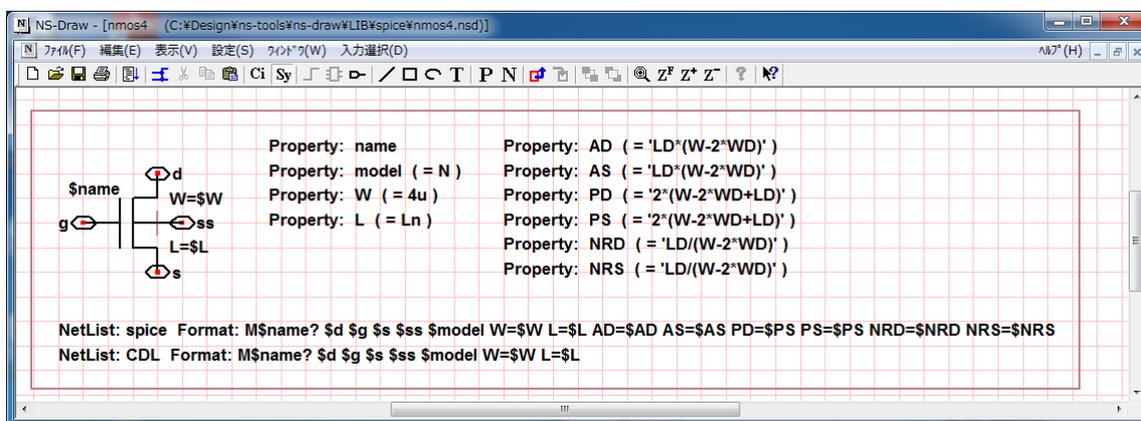


図 3 回路図の例 3 (NMOS プリミティブ素子: シンボル図のみ)

本章では以下、プリミティブ素子を作成する方法として、新たに MOSFET 素子を作成例を示します。

まず、新規作成で、新しい回路図を開き、k キーにより、図面をシンボル表示に切り替えてください。背景のグリッドが緑系から赤系に変わります。まず、図 4 のように、ライン描画(L キー)を使って、MOSFET のシンボル図形を作成します。今回は、Source と Drain 端子の区別がわかるように、また、エンハンスメント型を示すよう図 3 のシンボルとは形状を変更しています。グリッドに乗らない線を描く場合には、シフトキーを押しながら線の端点をドラッグすることで、グリッド外へ配置することができます。

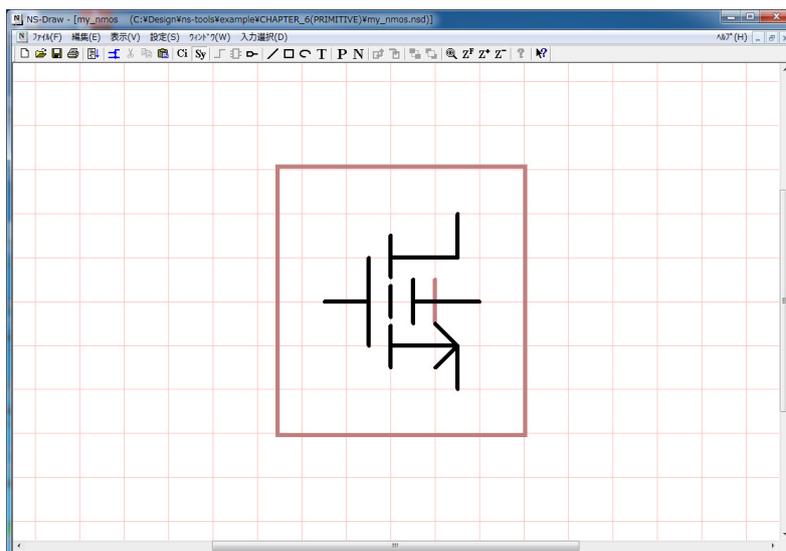


図4 MOSFETのシンボル図形の作成

次に端子を付けます。ここでは、ゲートと基板 (Sub) を入力端子、Source と Drain を I/O 端子、にして、それぞれ、Gate, Sub, Source, Drain と名前をつけてください。(SPICE ネットリストでは、端子種類の区別がないので、全てを I/O 端子にしても結構です。) 端子の位置については、他の回路との結線を容易にするために、グリッドに載せるようにしてください。図5のように、端子を4つの付けたら、my\_nmos.nsd という名前で保存してください。

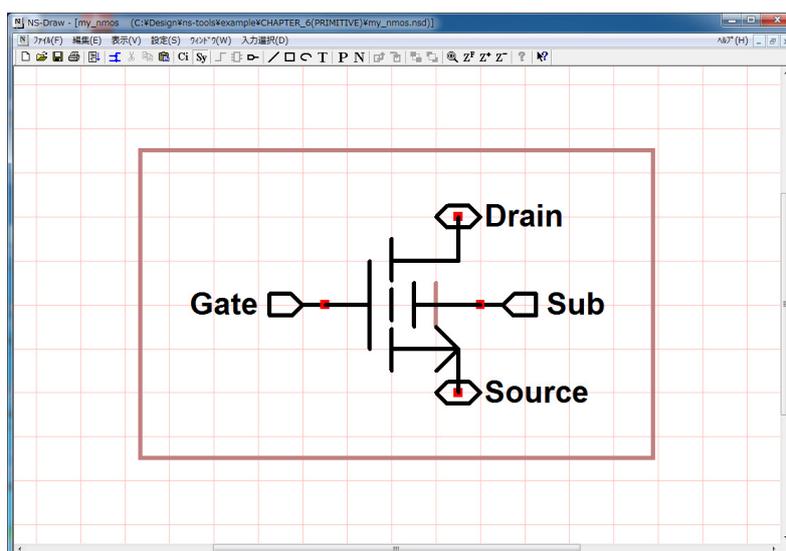


図5 シンボルへの端子の付加

次に、この作成した my\_nmos.nsd を用いて、図6のような回路図を作成します。抵抗には、名前として R0、値として 1K を設定しています。

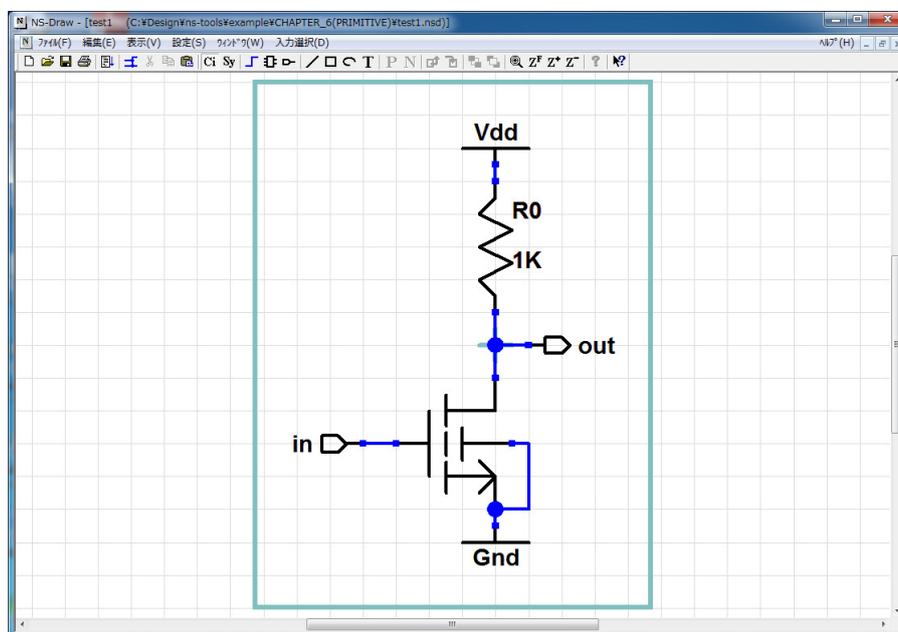


図6 新シンボルを利用した回路図作成

図6の回路を test1.nsd という名前で保存して、ネットリスト出力をした結果をリスト1に示します。今回作成した my\_nmos.nsd に相当する部分は赤字で示す部分で、不完全なサブサーキット記述として出力されています。

#### リスト1 ネットリスト出力結果

```
*****
* Top Level: test1
*+ (C:¥Design¥ns-tools¥example¥CHAPTER_6 (PRIMITIVE) ¥test1.nsd)
*****
X$name Gnd out in Gnd my_nmos
R0 out Vdd 1K
*****
* SubCircuit: my_nmos
*+ (C:¥Design¥ns-tools¥example¥CHAPTER_6 (PRIMITIVE) ¥my_nmos.nsd)
*****
*.SUBCKT my_nmos Source Drain Gate Sub
*.ENDS
.GLOBAL Vdd Gnd
.END
```

次に、正しくネットリスト出力させるための修正を行っていきます。まずこのMOSFETに設定できるプロパティの追加を行います。my\_nmos.nsdのシンボル図を開き、メニューバーの「P」ボタン「プロパティ定義行の挿入(pキー)」を押し、プロパティ名：

name, デフォルト値 : 空欄で OK ボタンを押します。

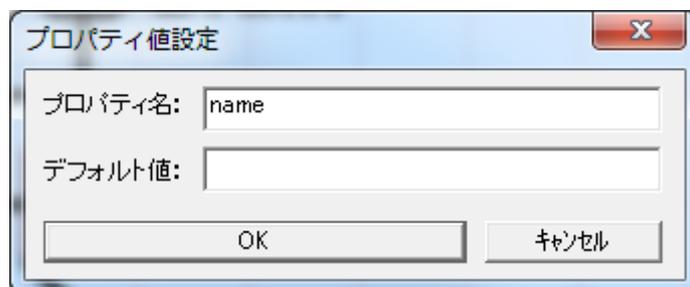


図7 プロパティ設定ダイアログ

図8を参考にシンボル図上の適当な位置に、Property行を配置します。さらに、以下の3つのプロパティを追加してください。(プロパティ名:L, デフォルト値:Ln)、(プロパティ名:W, デフォルト値:1u)、(プロパティ名:model, デフォルト値:nmos)

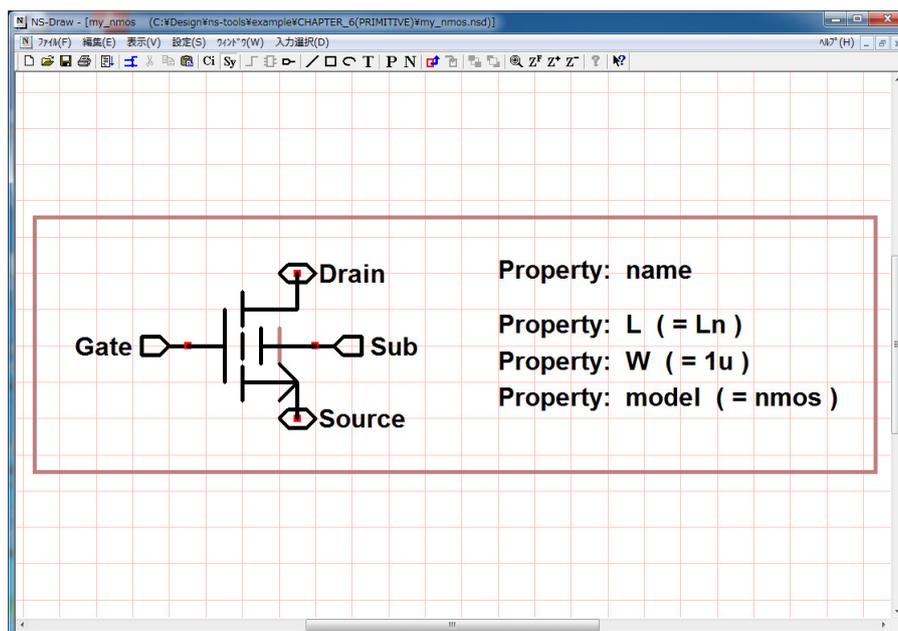


図8 プロパティの設定

さらに、メニューバーの「N」ボタン「ネットリスト定義行の挿入(f キー)」を押して、ネットリスト出力の内容を定義します。ここに記述した行の内容が、SPICE ネットリスト作成時に、このシンボルを表現する記述として出力されます。

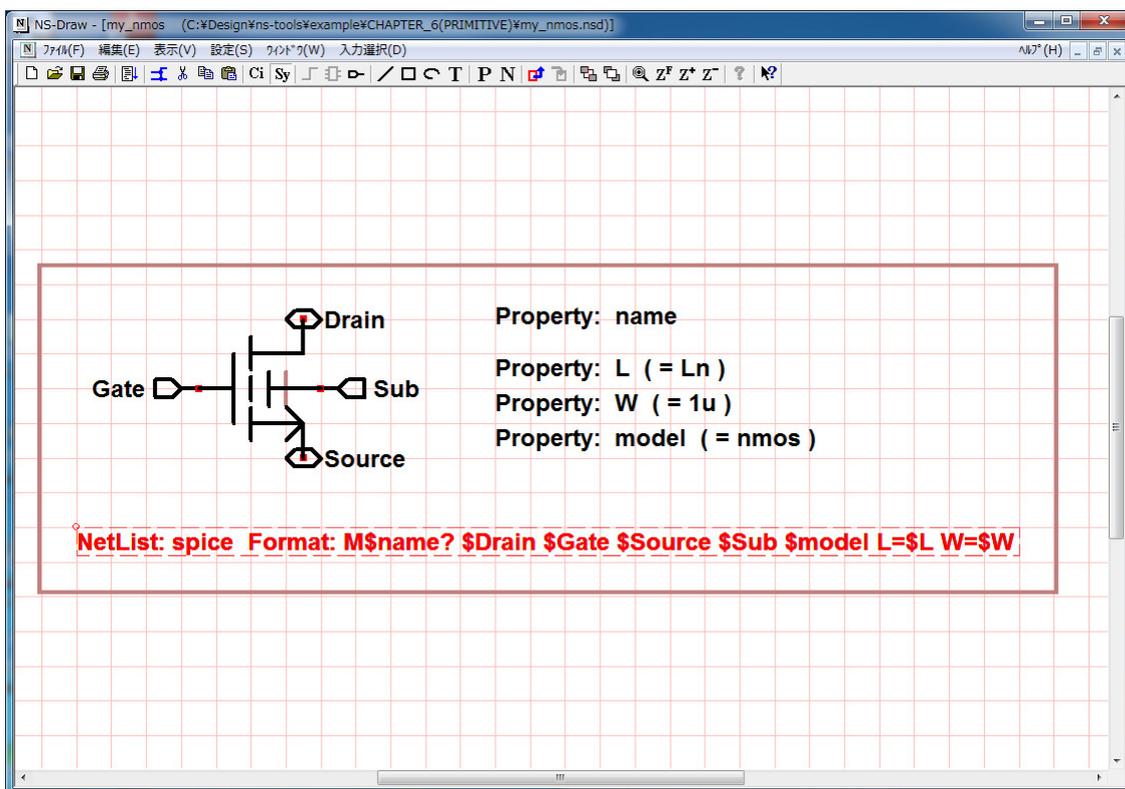


図9 ネットリスト指定行の挿入

ここでは、「ネットリスト指定行」として、以下の行を定義しています。

```
M$name? $Drain $Gate $Source $Sub $model L=$L W=$W
```

この行中にある \$name, \$model, \$L, \$W は、このシンボル図内で定義したプロパティ name, model, L, W それぞれの持つ値を参照します。ネットリスト指定行の所定の位置に \$name や \$L (先頭に \$ を付加) と記述しておくことで、そのプロパティの値をその部分に置き換えてネットリストを出力します。また、\$name の後につく ? の意味は、このシンボルが回路図上で複数置かれるときに、?の部分重複しない数字に置き換え、自動的に異なる素子名とするためのものです。自分でプリミティブを作成する場合、素子名の部分は上記のように O\$name? (O は SPICE ネットリスト上で素子を識別するアルファベット 1 文字 : MOSFET では M) という形にしてください。

また、\$Drain、\$Gate、\$Source、\$Sub のように、\$+端子名は、このシンボルを上位の回路図に配置したときに、その上位回路図接続端子名と置き換わります。もし \$ をつけず、単に Drain、Gate、等と記述した場合、ネットリスト上では、そのまま出力されてしまいますので、注意してください。しかし、例えば、「Gnd」などグローバル端子について、その名称のままネットリスト出力させたい場合は、\$ は不要です。

上位回路の test1.nsd に戻り、NMOSFET のシンボルをダブルクリックしてみると、図 10 のダイアログが開き、各プロパティ値を設定できます。値を設定していなければデ

フォルト値が表示されます。ここでは、name プロパティを MN1 に変更しました。



図 1 0 プロパティ値の設定

ネットリスト出力を行うと、リスト 2 の結果が得られます。新しく作成した MOSFET に相当する部分を赤字で示しましたが、回路の接続や、プロパティ値の出力が正しく行われ、MOSFET の SPICE 記述として出力されていることが確認できます。

#### リスト 2 ネットリスト出力結果

```
*****
* Top Level: test1
*+ (C:¥Design¥ns-tools¥example¥CHAPTER_6 (PRIMITIVE) ¥test1.nsd)
*****
MN1 out in Gnd Gnd nmos L=Ln W=1u
R0 out Vdd 1K
.GLOBAL Vdd Gnd
.END
```

次に、上位の回路図においても、プロパティ値を直接見やすいように、シンボルの一部としてプロパティ値を表示させる方法について説明します。図 1 1 に示すように、シンボル図において、メニューバーの T ボタン「テキスト入力 (t キー)」で、\$W/\$L と指定し、この文字列を図 1 2 のように、適当な位置に配置します。

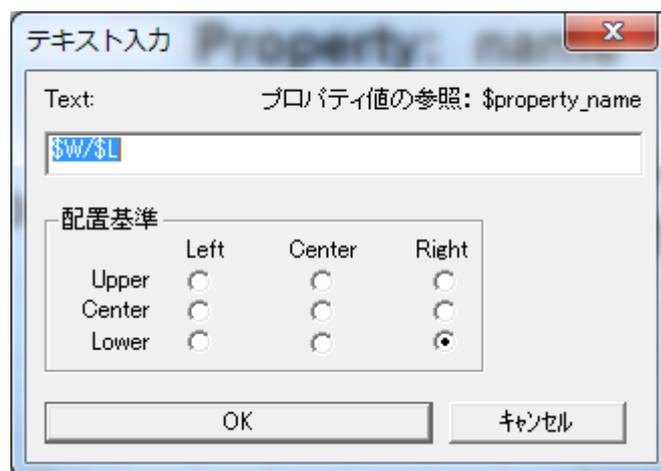


図 1 1 プロパティ値のテキスト表示

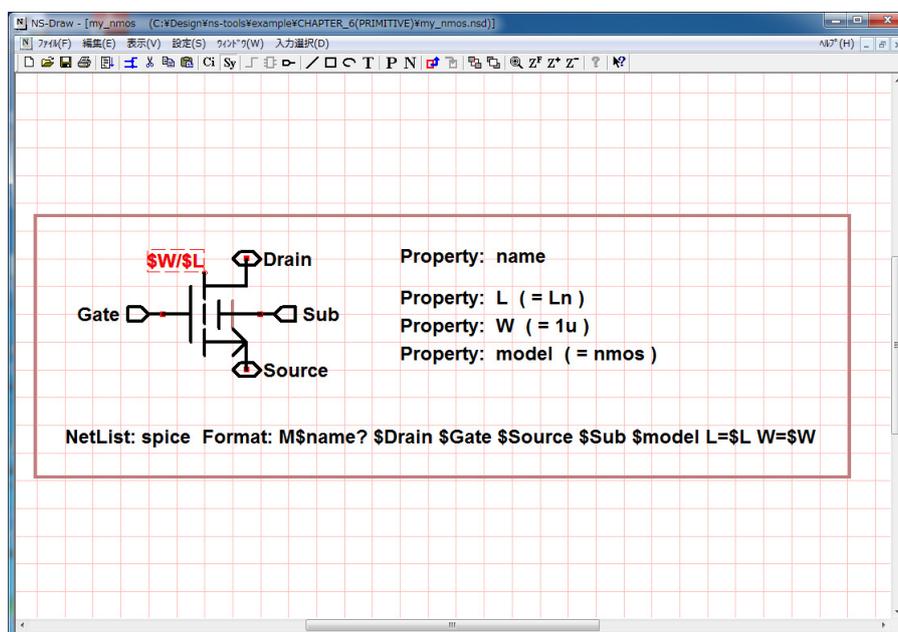


図 1 2 シンボル上へのプロパティ値表示の追加

この状態で、test1.nsdに戻れば、図 1 3のように、回路図上に、NMOSFETのW値とL値が表示されるようになります。

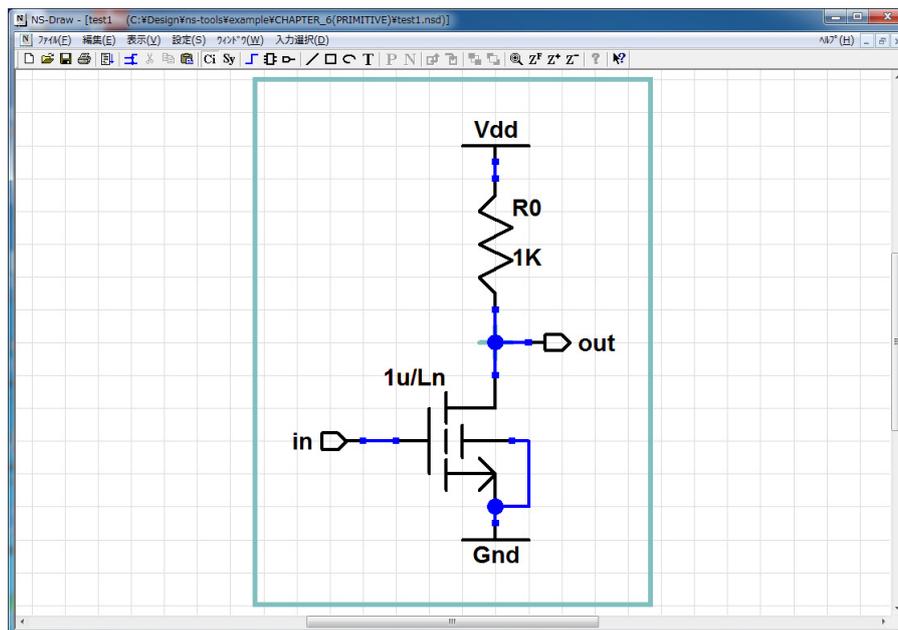


図 1 3 完成したシンボルを用いた回路図

最後に、実際に NS-Draw のライブラリ中に登録されている `nmos4m.nsd` を図 1 4 に示します。この章で作成した `my_nmos.nsd` に、さらに、プロパティとして、`M`, `AD`, `AS`, `PD`, `PS`, `NRD`, `NRS` 等が追加され、ネットリスト出力指定として、`spice` 以外に `CDL` 出力時の指定も追加されています。`AD` 等の定義で用いられるように、プロパティ値としては、演算式( ' ' で囲う)を登録することもできます。(この式は、`ns-spice` 等の `spice` シミュレータにおいて展開されることとなります)

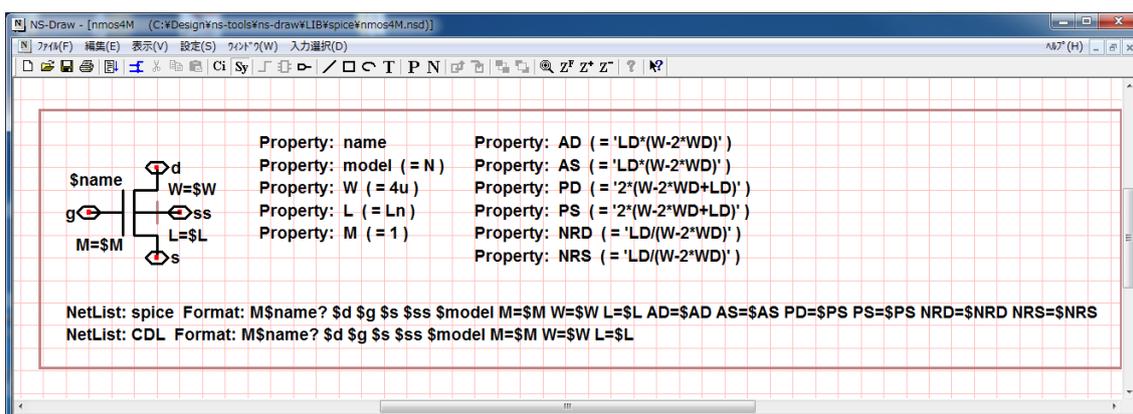


図 1 4 nMOSFET の定義(nmos4m.nsd)