

# Création de Symboles pour gEDA/gaf

Ales V. Hvezda\*

27 septembre 2003

---

\*Ce document est relaché sous GFDL (<http://www.gnu.org/copyleft/fdl.html>) par ahvezda@geda.seul.org). Il a été traduit par Iznogood de la Iznogood-Factory (<http://www.iznogood-factory.org>)

# Contents

<b>1</b>	<b>Survol</b>	<b>3</b>
<b>2</b>	<b>Création de symbole de composant</b>	<b>3</b>
<b>3</b>	<b>Prérequis</b>	<b>4</b>
<b>4</b>	<b>Style</b>	<b>6</b>
4.1	Texte . . . . .	6
4.2	Attributs . . . . .	6
4.3	Graphiques . . . . .	6
4.4	Broches . . . . .	7
4.5	Électrique . . . . .	7
<b>5</b>	<b>Conventions de nommage d'empreintes</b>	<b>8</b>
5.1	Notes . . . . .	8
5.2	Boîtiers de circuits intégrés . . . . .	8
5.3	Boîtiers de circuits intégrés SMT . . . . .	9
5.4	Semiconducteurs basics . . . . .	10
5.5	Semiconducteurs SMT de base . . . . .	11
5.6	Composants passifs . . . . .	11
5.7	Composants SMT passifs . . . . .	12
<b>6</b>	<b>Trucs et astuces</b>	<b>12</b>
<b>7</b>	<b>Exemple</b>	<b>13</b>
<b>A</b>	<b>Historique de la Révision du Document</b>	<b>16</b>

# 1 Survol

Ces document décrit la création de symboles de composants, incluant les conventions de style et les trucs/astuces et les choses à respecter lors du dessin de symboles pour le système gEDA/gaf.

## 2 Création de symbole de composant

Les symboles de composants (ils sont connus ici comme « symboles ») sont dessinés en utilisant gschem de la même manière que l'on dessinerait une page de schéma. Vous avez ici les étapes de création d'un symbole dans le système gEDA/gaf :

1. Lancez gschem et trouvez une page blanche ou lancez : `gschem filename-1.sym`
2. Dessinez le symbole (voyez le guide du style ci-dessous pour les conventions).
3. Déplacez le symbole à l'origine en utilisant Edit/Symbol Translate...
  - Agrandissez au moins une fois.
  - Assurez-vous que snap est à ON (c'est critique).
  - Assurez-vous que grid snap size est initialisé à 100 (c'est critique).
  - Sélectionnez "Symbol Translate..." ou saisissez les raccourcis claviers équivalent.
  - Entrez 0 dans le chap de saisie et pressez OK.

Le déplacement du symbole vers l'origine est une étape nécessaire. Pour déplacer un symbole ailleurs, entrez un offset (en mils) qui est un multiple pair de 100. Assurez-vous que que toutes les broches sont sur un point de grille à 100 mil.

4. Sauvegardez le symbole en utilisant Save ou SaveAs... Vous avez ici quelques conventions de nommage de symbole :
  - Les symboles sont nommés : `symbolname-#.sym`
  - Les symboles se terminent avec une extension `.sym`.
  - Les symboles ont un `-#` où `#` est nombre. `#` est typiquement à 1 mais s'il existe plusieurs symboles pour un composant alors ce nombre s'incrémente simplement.
  - Les noms de symboles sont typiquement en minuscules mais les lettres qui font partie de la zone nombre sont en majuscules.

- La règle de casse ci-dessus peut ne pas être respectée si le nom de fichier semble incorrect ou mauvais.
5. Placez le symbole dans un des répertoires spécifiés par les lignes component-library dans le fichier system-commonrc. Une fois fait, le symbole devrait être visible immédiatement, peut être sélectionné et placé avec le menu « Add/Select Component... ».

### 3 Prérequis

Cette section décrit les différents prérequis qui doivent être mis en ordre pour créer un symbole valide permettant d'afficher et de netlister dans le système gEDA/gaf. La plupart des prérequis sont centrés autour des attributs attachés ou à l'intérieur du symbole.

Lancer `gsymcheck` contrôlera que tous ces prérequis sont atteints. `gsymcheck` émettra des erreurs fatales qui sont sérieuses et doivent être corrigées. `gsymcheck` émettra aussi des alertes sur des éléments qui doivent être corrigés mais ne sont pas fatals.

Pour plus d'information sur les attributs présentés ici, veuillez aller voir le Attributs de Symboles pour gEDA/gaf (<http://www.iznogood-factory.org/pub/gEDA/attributes.fr.pdf>).

- **device=DEVICENAME** doit être placé quelquepart dans le symbole et rendu invisible. **device=** est le nom du composant et est nécessaire. Le nom de composant est typiquement en majuscules (lettres capitales). Cet attribut ne doit pas être utilisé comme label. Utilisez un objet texte séparé pour le label. Si l'objet est graphique alors **device=** doit être initialisé à none (**device=none**). Il n'est plus nécessaire d'attacher cet attribut ; il faut simplement qu'il existe comme **device=DEVICENAME** et c'est suffisant.
- **graphical=1** doit exister quelquepart dans un symbole qui est purement graphique (tel qu'un titre block ou un symbole decon). Les symboles qui possèdent cet attribut n'ont pas de signification électrique ou d'importance pour le circuit. N'oubliez pas d'initialiser **device=none**.
- **description=text** doit exister quelquepart dans le symbole. Cet attribut fournit une description, sur une ligne, du symbole.
- Toutes les broches doivent avoir une paire d'attribut qui leur sont attachés : **pinseq=#** et **pinnumber=#**. Le premier attribut, **pinseq=#** est seulement un séquence de nombres qui incrémente séquentiellement en débutant à 1. Le second attribut **pinnumber=#** est le numéro de la broche. Lorsqu'un symbole est netlisté, les numéros de broche sont sortis dans l'ordre de leur séquence. Le numéro de broche peut être alphanumérique (i.e. comme E ou C).

- Toutes les broches doivent aussi avoir un **pinlabel**=value d'attaché. Cet attribut est le nom ou le label de la broche (vs le numéro de broche). Cet attribut est aussi utilisé lorsqu'un symbole est utilisé dans un schéma hiérarchique. Veuillez rendre cet attribut vert (au lieu de l'attribut jaune par défaut).
- Toutes les broches doivent aussi avoir un **pin**type=value de rattachés. Cet attribut décrit le type de broche. Les valeurs possibles sont : in, out, io, oc, oe, pas, tp, tri, clk, pwr. Veuillez aller voir le Attributs de Symboles pour plus d'informations.
- Si un composant possède des slots multiples dans un boîtier (tel que un 7400 (NAND) qui possède 4 NAND par boîtier) alors vous avez besoin d'un attribut **numslots**=#. Le # est le nombre de slots que possède le composant. numslots= doit exister dans le symbole et être rendu invisible. Un slot additionnel nécessite les attributs décrits ci-dessus.
- Si un composant possède des slots multiples dans un boîtier physique, vous avez alors besoin d'inclure un **slotdef**=#:#,#,#... pour chaque slot. Le premier # correspond au numéro de slot. Si un composant possède 4 slots, il doit y avoir des attributs **slotdef**=1:..., **slotdef**=2:..., **slotdef**=3:..., et **slotdef**=4:..., existant quelquepart dans le symbole et rendez-le invisible. Les # consécutifs ont une correspondance de un-à-un avec les attributs **pinseq**=# et spécifier quel doit être le **pinnumber**=# correspondant lorsque le slot est initialisé. Voyez les 7400-1.sym attaché pour un exemple sur comment tout doit fonctionner.
- Il est recommandé que tous les symboles qui ont des slots ont un attribut **slot**=1 dans le symbole.
- **footprint**=PACKAGENAME devrait exister quelquepart dans le symbole qui peut être utilisé avec le netlister de PCB. PACKAGENAME est l'empreinte (footprint) PCB ou le type de paquet comme DIP14 ou DIP40. Veuillez aller voir le chapitre sur les **Conventions de nommage d'empreintes** pour de plus amples détails. Voyez aussi la documentation PCB et les gnetlist/docs/README.pcb pour plus d'informations sur cet attribut.
- Vous devriez mettre un attribut **refdes**=U? dans le symbole. Rendez la valeur visible et elle sera affichée (attachée à l'extérieur du symbole, de telle manière qu'elle soit éditée) lorsque le symbole est placé dans un schéma.
- L'attribut label= ne devrait pas être attaché quelquepart dans le symbole. Il est obsolète.
- L'attribut name= ne doit pas être attaché quelquepart dans le symbole.
- L'attribut netname= ne doit pas être attaché quelquepart dans le symbole. Il est seulement utilisé dans les schémas.

## 4 Style

Cette section décrit le style dans lequel est utilisé la bibliothèque de symbole standard gEDA/gaf.

### 4.1 Texte

- Tous les labels de Text devraient avoir 10 pt en taille.
- Les Text (les labels, pas les attributs) devraient être de couleur numéro 9 (texte | vert).

### 4.2 Attributs

- Les numéros de broches (qui sont des attributs) devraient être tous en taille de 8 pt.
- Les attributs attachés doivent être jaunes. La couleur est initialisée automatiquement au jaune si l'objet texte est attaché.
- La seule exception à ceci est l'attribut pinlabel=, qui doit avoir la couleur numéro 9 (texte | vert). Si chaque objet texte dans un symbole est jaune, le symbole semble aussi jaune.
- Les attributs peuvent être attachés à une partie du symbole. Les attributs toplevel (comme les attributs device= ou net=) utilisés nécessitent d'être attachés à quelque chose pour être des attributs mais maintenant, ils ont simplement besoin d'exister dans le fichier symbole comme name=value.
- Étendons nous un peu sur la dernière phrase, aussi longtemps que l'objet texte possède le format name=value, il est considéré comme un attribut. Les attributs dans un symbole n'ont pas à être attachés à quoi que ce soit. De manière à voir les attributs cachés dans gschem, sélectionnez Edit/Show/Hide Inv Text.

### 4.3 Graphiques

- Lines, boxes, arcs et tous les autres graphiques devraient être de la couleur numéro 3 (graphique | vert).
- Les bulles de polarité devraient être de couleur numéro 6 (logic bubble | cyan)

## 4.4 Broches

- Les broches (pins) devraient toutes avoir une longueur de 300 mils (3 espaces de grille).
- Les broches doivent avoir la couleur numéro 1 (broches | blanc).
- Laissez 400 mils (4 espaces de grille) entre les broches (verticales), à moins que vous ne soyez en cours de dessin d'un symbole spécial, tentez alors simplement de lui donner un bon aspect.
- Les attributs de numéro de broches doivent être à 50 mils au-dessus (ou au-dessous ; ce qui est le plus sensé) de la broche à laquelle est doit être attachée.
- Les broches d'entrée sont à gauche et les broches de sortie sont à droite du symbole.
- Veuillez ne pas mélanger les entrées et les sorties sur le même côté du symbole, à moins que cela ne soit absolument nécessaire.
- Vous pouvez avoir des broches au-dessus ou au-dessous du symbole.
- L'ordre des lignes des broches (bus) doit aller de LSB (least significant bit) à MSB (most significant bit). Lors du dessin des broches qui forment une partie du bus, assurez-vous que le LSB du bus est en haut (ou pour les broches en haut/bas d'un symbole, laissez le reste des autres broches). Regardez le 74/74181-1.sym pour avoir un exemple correct de cet ordre (A0 en haut jusqu'à A3 et B0 en haut jusqu'à B3). Violer cette règle rendra le connexion des bus beaucoup plus difficile.

## 4.5 Électrique

- Ne dessinez pas les broches d'alimentation et de masse. Cette information sera transmise en utilisant les attributs (veuillez aller voir le document `netattrib`).
- La règle ci-dessus peut être cassée si nécessaire mais gardez en mémoire que la plupart des bibliothèques standard ne montrent pas les broches d'alimentation.
- Gardez en mémoire que les symboles sont supposés être symboliques, ils ne représentent pas le boîtier physique qui vient avec le composant.
- Il y a quelques désagréments sur ce qui est écrit au-dessus, il est aussi possible d'arranger les broches sur le symbole logiquement de telle manière qu'ils permettent un schéma moins encombré. Notez qu'il y a fréquemment un arrangement différent des broches comme celui du boîtier physique.

## 5 Conventions de nommage d'empreintes

Cette section décrit les conventions de nommage des empreintes (footprints) utilisées dans gEDA/gaf.

Le but de la convention de nommage est d'établir un standard pour maintenir la même convention de nommage au travers des différentes phases de la chaîne de CAO. Ceci permet de s'assurer que l'effort collaboratif de gEDA/gaf n'est pas perdu.

### 5.1 Notes

- A moins que cela ne soit spécifié, les noms de broches numériques seront utilisés, en débutant par 1.
- $n$  est le numéro de broche.
- $m$  est l'espacement des broches en mils.
- $x$  est pour la dimension  $x$  du boîtier (en excluant les broches). En particulier,  $c$  est utilisé par la famille de boîtier QFP.
- SMT signifie surface mount (composant de montage en surface), les autres composants sont pour des placements avec trous.

### 5.2 Boîtiers de circuits intégrés

- Les boîtiers Dual in line avec jusqu'à 22 broches avec un espacement de broches à 100 mil et de lignes à 300 mil sont appelés DIP $n$ .
- Les boîtiers Dual in line avec 24 broches et plus avec un espacement de broches à 100 mil et de lignes à 300 mil sont appelés DIP $n$ N.
- Les boîtiers Dual in line avec un espacement de broches à 100 mil et de lignes à 400 mil sont appelés DIP $n$ H.
- Les boîtiers Dual in line avec 24 broches et plus avec un espacement de broches à 100 mil et de lignes à 600 mil sont appelés DIP $n$ .
- Les boîtiers Dual in line Shrink avec jusqu'à 24 broches avec un espacement de broches à 70 mil et de lignes à 300 mil sont appelés SDIP $n$ .
- Les boîtiers Dual in line Shrink avec 24 broches et plus avec un espacement de broches à 70 mil et de lignes à 400 mil sont appelés SDIP $n$ .
- Les boîtiers Single in line avec un espacement des broches à 100 mil sont appelés SIP $n$ N. Voyez aussi JUMPER, ci-dessous.
- Les boîtiers in-line Zig-zag sont appelés ZIP $n$ .
- Plastic leadless chip carrier with pin socket are called PLCC $n$ X.



### 5.3 Boîtiers de circuits intégrés SMT

- Les boîtiers SMT small outline avec jusqu'à 16 broches avec un espacement de broches à 50 mil et de lignes à 150 mil sont appelés *S0n*.
- Les boîtiers SMT small outline avec 16 broches et plus avec un espacement de broches à 50 mil et de lignes à 150 mil sont appelés *S0nN*.
- Les boîtiers SMT small outline avec un espacement de broches à 50 mil et de lignes à 200 mil au total sont appelées *S0nM*.
- Les boîtiers SMT small outline avec jusqu'à 20 broches avec un espacement de broches à 50 mil et de lignes à 300 mil sont appelés *S0nW*.
- Les boîtiers SMT small outline avec 20 broches et plus avec un espacement de broches à 50 mil et de lignes à 300 mil sont appelés *S0n*.
- Les boîtiers SMT small outline avec 44 broches et plus avec un espacement de broches à 50 mil et de lignes à 525 mil sont appelés *S0n*.
- Les boîtiers SMT small outline metric shrink des espacement de broches à 0.65 mm et de lignes à 323 mil sont appelés *MSSOPn*. *NOTE: A confirmer.*
- Les boîtiers SMT small outline metric shrink avec jusqu'à 44 broches avec un espacement de broches à 0.65 mm et de lignes à 420 mil sont appelés *MSSOPnW*.
- Les boîtiers SMT small outline metric shrink avec 44 broches et plus avec un espacement de broches à 0.65 mm et de lignes à 545 mil sont appelés *MSSOPnW*.
- Les boîtiers SMT small outline Shrink avec un espacement de broches à 0.25 mil et de lignes à 420 mil au total sont appelées *SSOPnW*.
- Les boîtiers SMT small outline Quarter size avec un espacement de broches à 0.25 mil et de lignes à 244 mil au total sont appelées *SSOPn*.
- Les boîtiers SMT small outline Thin avec un espacement de broches à 0.2165 mil et de lignes à 535 mil au total sont appelées *TSOPn*.
- Les boîtiers SMT small outline Thin avec un espacement de broches à 0.20 mil et de lignes à 795 mil au total sont appelées *TSOPnA*.
- Les boîtiers SMT small outline Thin avec un espacement de broches à 0.20 mil et de lignes à 559 mil au total sont appelées *TSOPnB*.
- Les boîtiers SMT small outline Thin shrink avec jusqu'à 28 broches avec un espacement de broches à 0.26 mil et de lignes à 260 mil sont appelés *TSSOPn*.

- Les boîtiers SMT small outline Thin shrink avec 28 broches et plus avec un espacement de broches à 0.20 mil et de lignes à 319 mil sont appelés TSSOP $n$ .
- Les boîtiers Ultra Super Mini SMT avec jusqu'à 16 broches avec un espacement de broches à 0.5 mil sont appelés US $n$ .
- Les Plastic leadless chip carrier SMT sont appelés PLCC $n$ .
- Les Square quad-side flat pack SMT sont appelés QFP $n\_x$ .
- Les Rectangular quad-side flat pack SMT sont appelés QFP $n\_R$ .
- Les Square low profile quad-side flat pack SMT sont appelés LQFP $n\_x$ .
- Les Square thin quad-side flat pack SMT sont appelés TQFP $n\_x$ .
- Les Square Quad-side flat no-lead SMT without exposed paddle (back side contact) sont appelés QFN $n\_x$ . Le nombre de broches est  $n$  et la taille du boîtier est  $x$  mm.
- Les Square Quad-side flat no-lead SMT with exposed paddle (back side contact) sont appelés QFN $n\_x\_EP$ . Le nombre de broches est  $n$  et la taille du boîtier est  $x$  mm.
- Les Thin profile square Quad-side flat no-lead SMT without exposed paddle (back side contact) sont appelés TQFN $n\_x$ . Le nombre de broches est  $n$  et la taille du boîtier est  $x$  mm.
- Les Thin profile square Quad-side flat no-lead SMT with exposed paddle (back side contact) sont appelés TQFN $n\_x\_EP$ . Le nombre de broches est  $n$  et la taille du boîtier est  $x$  mm.
- Les oscillateurs de style dual in line sont OSC8 et OSC14.
- Les boîtiers à 5 broches SOT SMT sont des SOT25 et des SOT325.
- Les boîtiers à 6 broches SOT SMT sont des SOT26 et SOT326.

#### 5.4 Semiconducteurs basics

- Les diodes axiales sont appelées ALF $m$ . La broche 1 est la cathode.
- Les LED conventionnelles sont LED3 et LED5 pour 3 et 5 mm respectivement. La broche 1 est le plus. *NOTE: Doit être probablement changé pour être en ligne avec les convention de diode.*
- Les transistors TO sont T05, T092, T0126, T0220 etc. Les suffixes peuvent s'appliquer, i.e. T0126W est pour large (wide), T0126S est pour debout (standing), T0126SW est pour debout, large.

## 5.5 Semiconducteurs SMT de base

- Les boîtiers SMT de diode SOD utilisent leur nom de boîtier standard, i.e. SOD80, SOD87, SOD106A, SOD110. Il existe aussi les SOD123, SOD323 avec des broches étroites.
- Les boîtiers SMT de transistors SOT utilisent leur nom de boîtier standard, i.e. SOT23, SOT323. Il existe aussi SC90.
- Les boîtiers SMT transistor SOT avec une numérotation comme pour les diodes ( la broche 1 est la cathode, la broche 2 est l'anode) sont SOT23D, SOT323D.
- Les boîtiers SMT 4 broches SOT sont SOT89, SOT143, SOT223.

## 5.6 Composants passifs

- Les composants non-polaires axiaux (typiquement les résistances, les condensateurs) sont appelés *ACY $m$* .
- Les composants non-polaires à broches sur le dessous (radiaux), typiquement les condensateurs sont appelés *RCY $m$* .
- Les composants rectangulaires non-polaires à broches sur le dessous (typiquement les condensateurs) sont appelés *BRE $m$* .
- Un oscillateur standard est HC49 ou d'autres désignations HC sont nécessaires.
- Les jumpers avec un espacement de broche de 100 mil sur une ligne unique sont appelés *JUMPER $n$* . La différence principale comparée à un boîtier à ligne unique est la taille des trous.
- Les connecteurs à espacement de 100 mil double ligne avec une numérotation de broche DIP sont les *HEADER $n$ \_1*. Notez que  $n$  est un nombre pair.
- Les connecteurs à espacement de 100 mil double ligne avec une numérotation de câble ribbon sont les *HEADER $n$ \_2*. Notez que  $n$  est un nombre pair.
- Les connecteurs à plat avec verrous (angled full header connectors with latches) sont appelés *DIN41651\_ $n$* .
- Les connecteurs debout avec verrous (standing full header connectors with latches) sont appelés *DIN41651\_ $n$ S*.
- Les connecteurs femelle DSUB sont appelés *DB $n$ F*.
- Les connecteurs mâles DSUB sont appelés *DB $n$ M*.

- Les connecteurs femelles DIN card-to-card sont des DIN41612CnF. Ajoutez le suffixe S pour debout (standing).
- Les connecteurs mâles DIN card-to-card sont des DIN41612CnM. Ajoutez le suffixe S pour debout (standing).
- Les connecteurs AMP modulaires RJ avec écran sont appelés RJ11, RJ12 et RJ45.

### 5.7 Composants SMT passifs

- Les résistances, inductances, condensateurs, etc. SMT standard sont appelés 0201, 0402, 0603, 0805, 1206, 1210, 1806, 1812, 1825, 2020, 2706.
- Les condensateurs SMT au tantale sont appelés EIA3216, EIA3528, EIA6032 et EIA7343. La broche 1est le plus.
- Les électrolytiques SMT sont conçus pour un diamètre de boîtier de 1/10 mm: SME33, SME43, SME53, SME66, SME84, SME104.

## 6 Trucs et astuces

Cette section décrit quelques trucs et astuces qui rendra votre expérience de création de symboles plus facile.

- Évitez de dessiner des éléments en dehors de la grille. Si vous le faites, vous ne pourrez pas déplacer l'objet en utilisant la commande move (si la grille est à on) car l'objet sera sorti de la grille. [ceci était un ancien bogue, qui, je pense, doit être réparé mais évitez de la faire de toute manière]. Utilisez la commande de traduction de symbole (ou déplacez l'objet avec grid snap off)
- Si vous avez besoin d'une grille plus fine, utilisez alors Options/Snap Grid Spacing... pour le faire. Rappelez-vous simplement de le ramener à 100 une fois que vous êtes prêts à déplacer le symbolé vers l'origine.
- Si vous voulez déplacer un symbole depuis l'origine vers un autre endroit, utilisez alors la commande "Symbol translate" et entrez une valeur non-nulle. Assurez-vous que ce nombre est un multiple de 100 (i.e. 1000 ou 1100).
- Les broches **DOIVENT** être placées sur une grille espacée de 100 (au moins les broches qui ont des liaisons de connectées).
- Les broches **DOIVENT** être placées sur une grille espacée de 100 (au moins les broches qui ont des liaisons de connectées). Oui, cette ligne est dupliquée. Je n'ai pas suffisamment appuyé sur ce point.

- Rappelez-vous que les broches sont des objets spéciaux ; si vous voulez ajouter une broche, assurez-vous que c'est une broche et pas une ligne ou une liaison. Utilisez la commande Add/Pin pour placer une broche.
- N'incluez pas les liaisons ou les bus dans les symboles. Ce n'est pas supporté et cela n'a pas de sens de toute manière.

## 7 Exemple

Cette section fournit un exemple simple qui tente de suivre toutes les règles ci-dessus. Ce symbole est une porte 7400 (porte NAND).

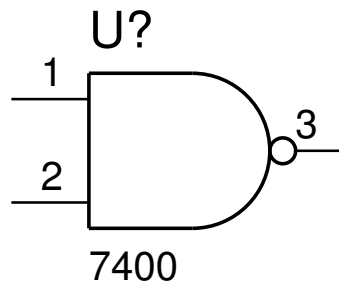
```
v 20020825
L 300 200 300 800 3 0 0 0 -1 -1
T 300 0 9 8 1 0 0 0
7400
L 300 800 700 800 3 0 0 0 -1 -1
T 500 900 5 10 0 0 0 0
device=7400
T 500 1100 5 10 0 0 0 0
slot=1
T 500 1300 5 10 0 0 0 0
numslots=4
T 500 1500 5 10 0 0 0 0
slotdef=1:1,2,3
T 500 1700 5 10 0 0 0 0
slotdef=2:4,5,6
T 500 1900 5 10 0 0 0 0
slotdef=3:9,10,8
T 500 2100 5 10 0 0 0 0
slotdef=4:12,13,11
L 300 200 700 200 3 0 0 0 -1 -1
A 700 500 300 270 180 3 0 0 0 -1 -1
V 1050 500 50 6 0 0 0 -1 -1 0 -1 -1 -1 -1 -1
P 1100 500 1300 500 1
{
T 1100 550 5 8 1 1 0 0
pinnumber=3
T 1100 450 5 8 0 1 0 2
pinseq=3
T 950 500 9 8 0 1 0 6
pinlabel=Y
T 950 500 5 8 0 1 0 8
pintype=out
```

```

}
P 300 300 0 300 1
{
T 200 350 5 8 1 1 0 6
pinnumber=2
T 200 250 5 8 0 1 0 8
pinseq=2
T 350 300 9 8 0 1 0 0
pinlabel=B
T 350 300 5 8 0 1 0 2
pintype=in
}
P 300 700 0 700 1
{
T 200 750 5 8 1 1 0 6
pinnumber=1
T 200 650 5 8 0 1 0 8
pinseq=1
T 350 700 9 8 0 1 0 0
pinlabel=A
T 350 700 5 8 0 1 0 2
pintype=in
}
T 300 900 8 10 1 1 0 0
refdes=U?
T 500 2250 5 10 0 0 0 0
footprint=DIP14
T 500 2450 5 10 0 0 0 0
description=4 NAND gates with 2 inputs
T 500 2650 5 10 0 0 0 0
documentation=http://www-s.ti.com/sc/ds/sn74ls00.pdf
T 500 2850 5 10 0 0 0 0
net=Vcc:14
T 500 3050 5 10 0 0 0 0
net=GND:7

```

Cet exemple produit ce qui suit (en utilisant gschem):

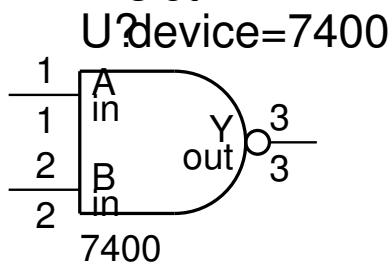


C'est le même symbole avec tous les textes cachés rendus visibles (via Edit/Show/Hide Inv Text):

```

net=GND:7
net=Vcc:14
documentation=http://www-s.ti.com/sc/ds/sn74ls00.pdf
description=4 NAND gates with 2 inputs
footprint=DIP14
slotdef=4:12,13,11
slotdef=3:9,10,8
slotdef=2:4,5,6
slotdef=1:1,2,3
numslots=4
slot=1

```



## A Historique de la Révision du Document

14 septembre 2002	Création de symbol.tex depuis symbols.html
31 octobre 2002	Fixé un mauvais exemple de symbole
11 février 2003	Ajout des convetions de nommage d'empreintes (footprint)
27 septembre 2003	Ajout du patch de Dan McMahon sur les QFP et QFN