

# TP7: Routage manuel "overcell"

1. 1 Objectif
2. 2 Spécification fonctionnelle
3. 3 Saisie du schéma
4. 4 Placement / Routage
5. 5 Validation du routage

## 1 Objectif

Nous avons vu dans le TP6 comment dessiner une cellule précaractérisée. Le but de ce TP7 est de réaliser le dessin des masques d'un bloc combinatoire résultant de l'interconnexion d'une quinzaine de cellules de la bibliothèque **SXLIB**.

On va donc effectuer *à la main* (c'est à dire en utilisant l'éditeur interactif **graal**), le placement et le routage de cet opérateur, pour mieux comprendre les problèmes que doivent résoudre les outils de placement/routage automatiques.

## 2 Spécification fonctionnelle

On souhaite réaliser le décodeur 3 vers 4 tel que décrit ci-dessous.



La sortie  $Z_i$  prend la valeur 1 quand le nombre de bits d'entrée ayant la valeur 1 est égal à  $i$ . Les 3 autres sorties prennent la valeur 0.

Les équations sont les suivantes:

- $Z_0 = \overline{A} \cdot \overline{B} \cdot \overline{C}$
- $Z_1 = (\overline{A} \cdot \overline{B} \cdot C) + (\overline{A} \cdot B \cdot \overline{C}) + (A \cdot \overline{B} \cdot \overline{C})$
- $Z_2 = (A \cdot \overline{B} \cdot C) + (\overline{A} \cdot B \cdot C) + (\overline{A} \cdot \overline{B} \cdot C)$
- $Z_3 = A \cdot B \cdot C$

De ces équations, on déduit le schéma en portes logiques inverseuses

Le bloc à réaliser contient donc 10 portes NAND3 et 5 inverseurs. Regardons plus précisément les caractéristiques de chaque cellule :

- La cellule NAND3 a une largeur de 5 pitches.

Les 3 signaux d'entrée  $E_1$ ,  $E_2$  et  $E_3$  sont accessibles sur 6 pistes de routage. Le signal de sortie  $S$  est accessible sur 7 pistes de routage.

- La cellule INVERSEUR a une largeur de 3 pitches.

Le signal d'entrée  $E$  est accessible sur 6 pistes de routage. Le signal de sortie  $S$  est accessible sur 7 pistes de routage.



On propose le placement ci-dessous pour cet opérateur, mais vous pouvez choisir un autre placement si vous le souhaitez :



Le dessin ci-dessous illustre un exemple de routage "overcell" pour la génération d'un signal de sortie

## 3 Saisie du schéma

Utiliser le langage **STRATUS** pour décrire le schéma proposé ci dessus, et générer le fichier 'decodeur'.vst correspondant à la description structurelle VHDL de cet opérateur. Valider ce schéma en écrivant quelques stimuli, et en simulant sous *asimut*.

## 4 Placement / Routage

Saisir sous **graal** le dessin du bloc *decoder* en instanciant les 5 portes *inv\_x1* et les 10 portes *na3\_x1*. Dessiner les fils de routage sous **graal**, puis vérifier l'absence de violation des règles de dessin. le routage "overcell" en lançant le vérificateur de règles de dessin **druc** sous **graal**.

Pour que cette vérification soit significative, il faut préalablement "mettre à plat" le bloc *decoder*, en utilisant la commande *real flat*.

## 5 Validation du routage

Extraire la netlist du bloc *decoder* au format .al avec l'outil **COUGAR**, mais sans descendre au niveau des transistors : On veut obtenir une *net-list* de cellules, et non une *net-list* de transistors.

Vérifier que les deux net-lists définies par les fichiers *decoder.vst* et *decoder.al* sont isomorphes en utilisant l'outil **lvx**.