

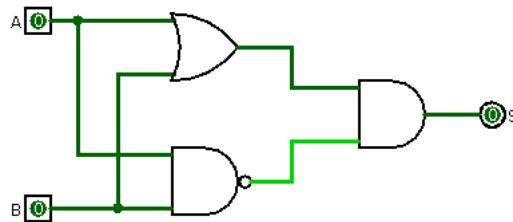
## Feuille de travaux pratiques n° 2

### Circuits combinatoires

Le logiciel logisim permet de simuler et d'analyser des circuits logiques combinatoires et séquentiels. On se reporterà à sa documentation accessible à partir de la page du cours sur *madoc*.

#### Simulation d'un circuit

- Construire le circuit ci-dessous :



- Insérer les portes OR et AND ainsi que les éléments d'entrée/sortie à partir de la barre d'outils ;
- Insérer la porte NAND à partir du sous-menu *Gates* de l'explorateur ;
- Sélectionner chaque porte logique du circuit et modifier ses caractéristiques dans la fenêtre de paramètres de façon à ne faire apparaître que deux entrées et une sortie pour chaque porte ;
- Relier les différents éléments du circuit et nommer les entrées et la sortie (cliquer sur l'icône A de la barre d'outils, puis sélectionner un élément d'E/S pour lui ajouter une étiquette).

- Cliquer sur le menu *Simulate*→*Logging...* et ajouter les entrées A, B et la sortie S dans la liste des E/S à surveiller. Choisir l'icône représentant une main dans la barre d'outils (passage en mode « simulation ») et cliquer alternativement sur chaque entrée; chaque clic modifie la valeur de l'entrée correspondante. L'onglet *Table* de la fenêtre affichée par le menu *Simulate*→*Logging...* construit la table de vérité du circuit au fur et à mesure.

- À quelle porte logique ce circuit est-il équivalent ?

On va désormais tirer parti des capacités de logisim à synthétiser un circuit à partir d'une expression logique pour refaire le travail précédent :

- Relancer logisim avec une page de travail vierge ;
- Ouvrir le menu *Window*→*Combinational Analysis*; ajouter deux entrées A et B et une sortie S ;
- Dans l'onglet *Expression*, écrire l'expression logique correspondant au circuit vu précédemment (les opérateurs à utiliser sont ceux du langage C/C++) puis cliquer sur *Enter* ; Logisim crée alors la table de vérité de l'expression (visible dans l'onglet *Table*) ainsi qu'une version minimisée grâce à un diagramme de Karnaugh (onglet *Minimized*) ;
- Cliquer sur *Build Circuit*.

#### Conception d'un circuit

On cherche maintenant à concevoir un circuit qui réalise l'addition de deux nombres binaires exprimés sur 4 bits.

1. Écrire la table de vérité de la fonction add1 réalisant l'addition de deux bits. La fonction add1 a deux entrées (les bits à ajouter) et deux sorties (le bit résultat et la retenue sortante  $R_s$ );
2. Réaliser dans logisim le circuit add1 ; tester le circuit pour vérifier que ses sorties correspondent bien à ce qui est attendu;
3. Écrire la table de vérité de la fonction add1r effectuant l'addition de deux bits et d'une retenue entrante  $R_e$  ; la fonction add1r a donc trois entrées (les deux bits à ajouter et la retenue  $R_e$ ) et deux sorties (le bit résultat et la retenue sortante  $R_s$ );
4. Réaliser le circuit correspondant à add1r en utilisant le nombre nécessaire d'instances du sous-circuit add1. Tester le circuit;
5. Réaliser le circuit add4 permettant l'addition de deux nombres binaires sur 4 bits en utilisant le nombre nécessaire d'instances du circuit add1r. Tester le circuit.