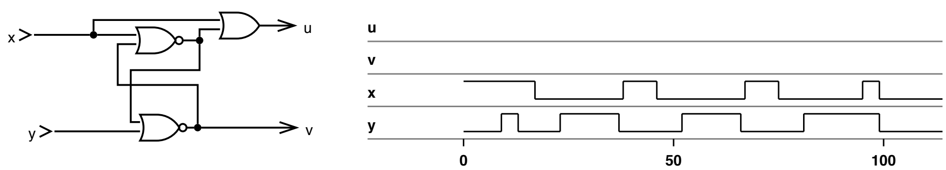
Licence Informatique S3
Architecture des ordinateurs
Aix Marseille Université 2016-2017
`http://pageperso.lif.univ-mrs.fr/~peter.niebert/archi2016.php`

TD numéro 3

Bascules

Exercice 1 [impulse] Vous avez vu dans le cours un circuit qui donne généralement 0 sauf sur un petit intervalle après un front montant de son entrée. Concevez un circuit qui fait de même, mais pour des fronts descendants.

Exercice 2 [simulation] Pour le circuit donné et pour le chronogramme des entrées, calculez le chronogramme des sorties. Indication : il convient de calculer également les signaux internes du circuit.



Exercice 3 Réalisez une bascule D complète comme dans TkGate telle que son comportement correspond à la spécification suivante (en ignorant la vitesse) :

- Elle dispose d'entrées `CK` (clock), `D` (donnée), `_EN` (entrée inversée enable), `_CLR` (entrée inversée clear), et une sortie `Q`.
- Lorsque `_CLR` est actif (0), `Q:=0`.
- Lorsque `_CLR` est non actif (1) :
  - Lorsque `_EN` est actif (0) et il y a un front montant sur `CK`, on affecte `Q:=D`.
  - Sinon (`_EN` non actif (1) ou hors front montant sur `CK`) `Q` ne change pas.

Réalisez cette bascule en utilisant une bascule RS, un circuit pour la détection de fronts et quelques portes logiques pour contrôler la bascule RS.

Circuits combinatoires et vitesse

On souhaite réaliser un comparateur pour des nombres de n bits avec les opérations $<$ (strictement plus petit) et $\leq$ (plus petit). Nous allons procéder de deux façons différentes et étudier chacune des deux solutions en terme de nombre de portes logiques utilisées et de temps de calcul.

Exercice 4 [Comparateur associatif]

Question 4.1 *Dessiner deux circuits $Comp_<$ et $Comp_\leq$ ayant tous les deux 2 entrées a et b et une sortie s valant 1 si respectivement $a < b$ et $a \leq b$.*

On utilise un comparateur par paire de bits, qui dispose de plus de deux entrées valant les résultats de comparaison pour les bits précédents dans les deux nombres. On retrouve à la figure 1 à gauche un des ces comparateurs associatifs et sur la droite, leur assemblage pour obtenir un comparateur 3 bits.

Chaque comparateur associatif utilise le principe suivant :

- $(a_n a_{n-1} \dots a_2 a_1)_2 < (b_n b_{n-1} \dots b_2 b_1)_2$ ssi
  $$a_n < b_n \quad \text{ou} \quad a_n = b_n \text{ et } (a_{n-1} \dots a_2 a_1)_2 < (b_{n-1} \dots b_2 b_1)$$
- $(a_n a_{n-1} \dots a_2 a_1)_2 \leq (b_n b_{n-1} \dots b_2 b_1)_2$ si et seulement si
  $$a_n < b_n \quad \text{ou} \quad a_n = b_n \text{ et } (a_{n-1} \dots a_2 a_1)_2 \leq (b_{n-1} \dots b_2 b_1)$$

Question 4.2 *Selon ce principe, définir un circuit COMB permettant de construire un comparateur associatif en combinant des entrées $<_{in}$ et $\leq_{in}$ et les résultats des circuits $Comp_<$ et $Comp_\leq$.*

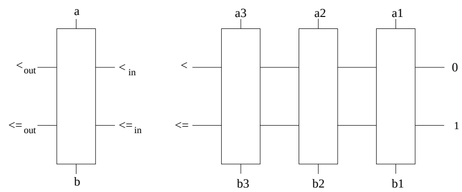


FIGURE 1 –

Question 4.3 *Pour le circuit obtenu, donner, en fonction de n, le nombre de portes logiques nécessaires à la réalisation d'un comparateur de nombres de n bits comme celui décrit figure 1.*

Question 4.4 *Pour le circuit obtenu, calculer, en fonction de n et de t le temps de passage d'une porte logique pour les signaux, au bout de combien de temps les résultats de la comparaison sont disponibles sur la sortie.*

Exercice 5 [Comparateur récursif]

Etudions maintenant une autre manière de réaliser un comparateur n bits ; nous supposons ici que n est une puissance de 2, *ie* $\exists d \in \mathbb{N}$, $n = 2^d$.

Question 5.1 *Montrer comment les deux relations d'ordre $<$ et $\leq$ entre $(a_n a_{n-1} \ldots a_{\frac{n}{2}+1})_2$ et $(b_n b_{n-1} \ldots b_{\frac{n}{2}+1})_2$ et entre $(a_{\frac{n}{2}} a_{\frac{n}{2}-1} \ldots a_1)_2$ et $(b_{\frac{n}{2}} b_{\frac{n}{2}-1} \ldots b_1)_2$ permettent de calculer les relations d'ordre entre $(a_n a_{n-1} \ldots a_2 a_1)_2$ et $(b_n b_{n-1} \ldots b_2 b_1)_2$.*

Question 5.2 *Réaliser un comparateur pour n bits à l'aide de 2 comparateurs pour $\frac{n}{2}$ bits (dont vous supposez disposer) et le circuit COMB obtenu précédemment.*

On peut de manière récursive appliquer le même principe et construire chacun des comparateurs pour $\frac{n}{2}$ bits avec 2 comparateurs pour $\frac{n}{4}$ bits qui eux-même peuvent se construire avec .....

Question 5.3 *Décrire comment selon ce principe, on peut obtenir un comparateur pour n bits en utilisant <u>uniquement</u> des circuits $Comp_<$, $Comp_\leq$ et COMB.*

Question 5.4 *Calculer le nombre de circuits $Comp_<$, $Comp_\leq$ et COMB et donc, le nombres de portes logiques utilisé(e)s dans votre comparateur récursif pour n bits.*

NB : *le terme général de la suite* $u_n = \begin{cases} 2u_{\frac{n}{2}} + c & si\ n > 1 \\ k & si\ n = 1 \end{cases}$ *est* $u_n = n * k + (n-1) * c.$

Question 5.5 *Calculer, en fonction de n et de t le temps de passage d'une porte logique , au bout de combien de temps les résultats de la comparaison pour n bits sont disponibles sur la sortie.*

Exercice 6 [addition*] Le délai d'un circuit combinatoire est la durée maximale entre un changement d'un ou de plusieurs entrées et la stabilisation de toutes les sorties sur les valeurs de la fonction booléenne.

Analysez le circuit d'addition en chaîne d'additionneurs complêts. Simlifiez en estimant le délai maximal de chaque porte à 1, puis calculez le délai du circuit entier. Trouvez également deux vecteurs d'entrées avec un seul bit changé qui engendre une séquence de changements à la sortie d'une durée maximale.