

## TP N°1

### CELLULES LOGIQUES AU NIVEAU TRANSISTOR

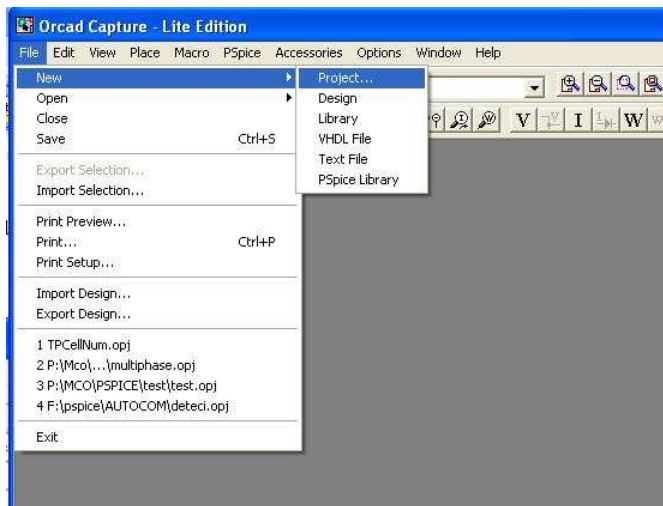
#### Objectif :

L'objectif de ce TP est de découvrir et d'analyser les performances de cellules logiques réelles développées dans une technologie submicronique. Par « cellules » on entend : des portes logiques combinatoires de base et la bascule D (élément important pour toute logique séquentielle). L'intégralité du TP se fera sous l'environnement Orcad-PSpice en utilisant les fichiers des modèles de MOS de la technologie AMS-CMOS 0.35  $\mu\text{m}$ .

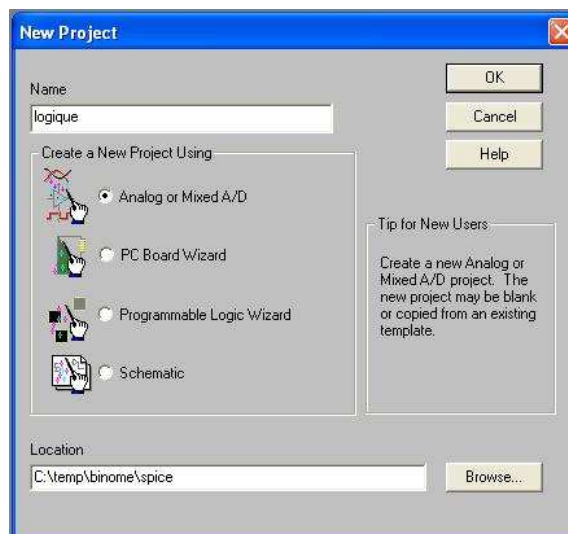
#### I – Analyse des performances dynamiques d'un inverseur CMOS

##### I-1 – Démarrage :

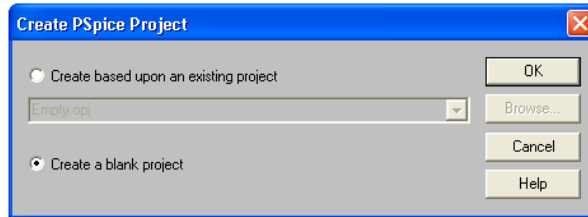
1- Tout d'abord, il vous faut créer un nouveau projet sous l'environnement Orcad-PSpice :



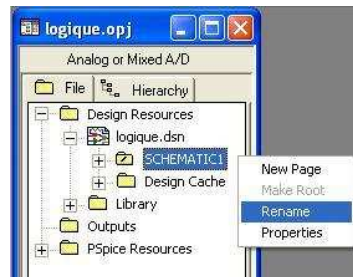
2- Choisir un bon répertoire, et sélectionner « Analog or Mixed A/D »



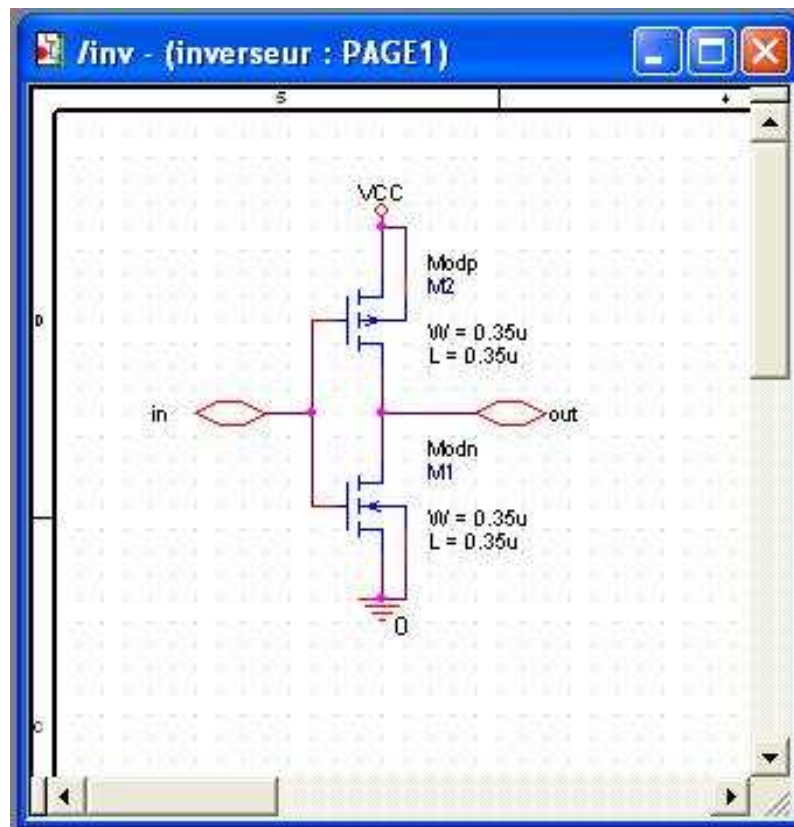
## 3- Créer un projet :



## 4- Renommer le schéma « Inverseur ».

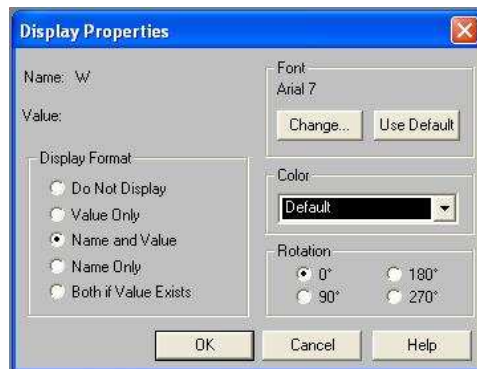


Et réaliser le schéma d'un inverseur CMOS.



## Conseils :

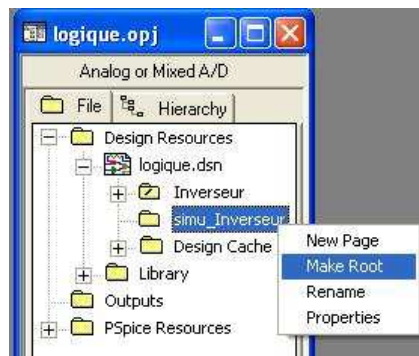
- Utiliser les symboles MBREAKN et MBREAKP de la librairie BREAKOUT,
- Donner les noms de modèles Modn et Modp aux 2 composants,
- Renseigner les champs W (largeur) et L (Longueur) correctement ( $0.35\mu$ ) en spécifiant « Display – Name and Value » pour chacun d'eux.



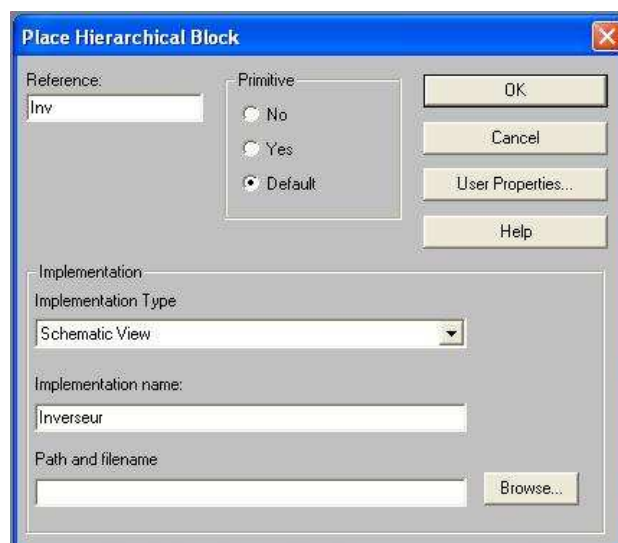
- La masse doit bien comporter la valeur « 0 »

5- Faire un nouveau schéma nommé « simu\_Inverseur » dans lequel on va placer le symbole hiérarchique du composant inverseur.

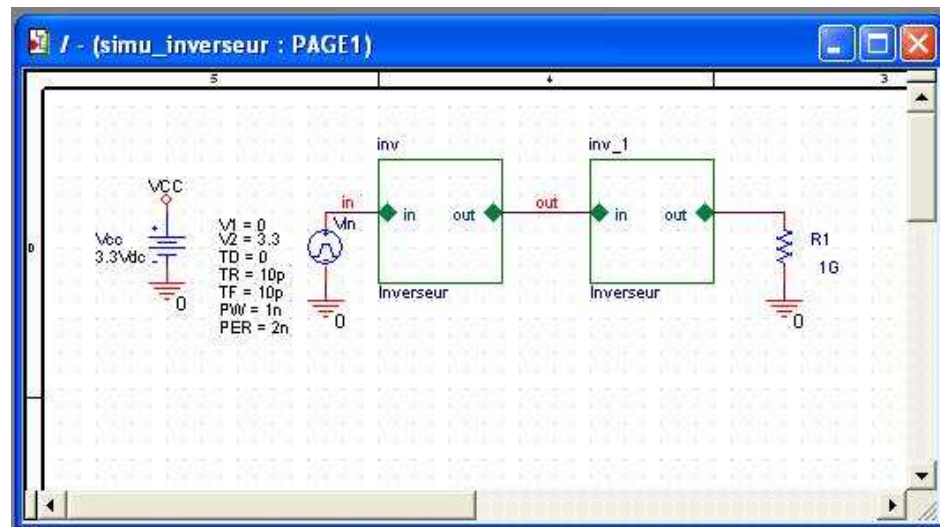
6- Placer ce schéma en tête du projet



Ensuite, créer dans le schéma un bloc hiérarchique pour implémenter un inverseur (par « Place » puis « Hierarchical Block »).



Enfin, réaliser le schéma suivant :



### I-2 Simulation des performances d'un inverseur CMOS élémentaire.

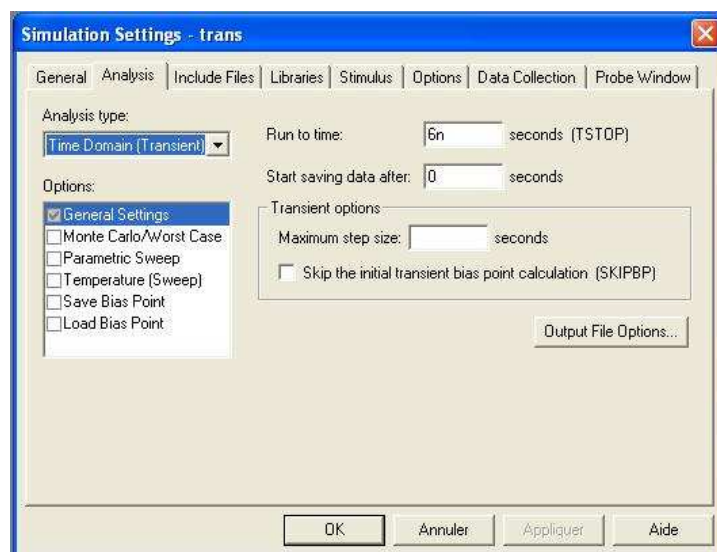
On charge l'inverseur à étudier par l'entrée d'un autre inverseur similaire afin de caractériser correctement les performances de la cellule lors de commutations.

On veut faire une simulation temporelle pour estimer les temps de montée et de descente de l'inverseur.

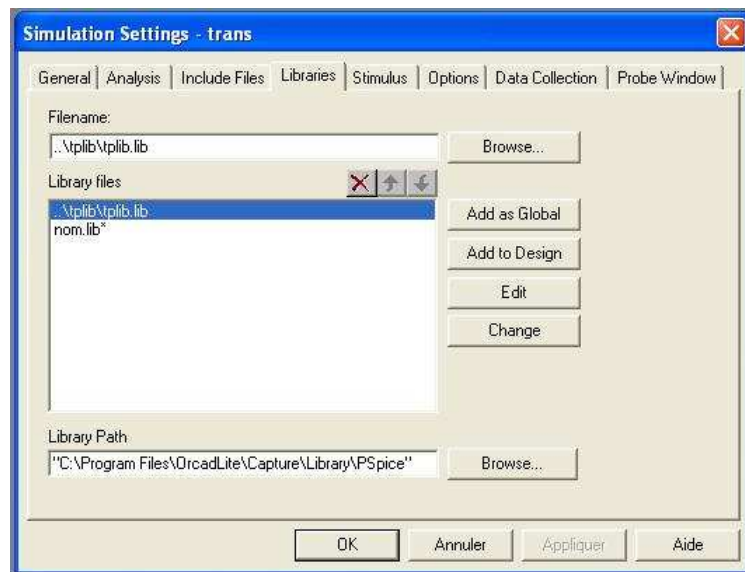
Pour cela : Faire un nouveau profil de simulation



Choisir une simulation « Transient » avec une analyse de 6ns et un pas de 1ps :



Enfin, ne pas oublier d'inclure la librairie « tplib.lib » qui contient les modèles des transistors MOS dans votre « Design » :



Lancer une simulation, et observer les signaux  $V(in)$   $V(out)$ . En déduire les temps de montée et de descente de l'inverseur (10% et 90% d'un échelon):

✓ Donner «  $T_r$  », et «  $T_f$  » ainsi que la courbe de simulation.

Observer maintenant le courant consommé par la porte lors des commutations :  $IS(inv.M2)$ .

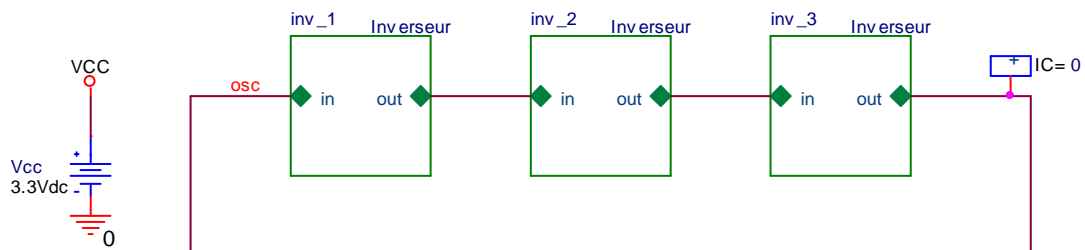
✓ Donner la valeur maximale «  $I_{max}$  » de ce courant. En utilisant la fonction intégrale :  $S(IS(inv.M2))$ , déterminez la charge  $Q$  utilisée.

Si un million de portes commutent en même temps, quel courant Max obtenez-vous :

✓ Donner «  $I_{Max_{1million}}$  » ainsi que la charge «  $Q_{1million}$  » équivalente. Proposer une valeur de capacité, placée en parallèle avec l'alimentation, pour remédier à ce problème de surconsommation en courant.

### I-3 Evaluation du temps de propagation moyen d'un inverseur CMOS.

Faites un nouveau schéma (que vous placerez à la racine du projet) nommé « simu\_oscillateur » :



En plaçant dans une boucle un nombre impair d'inverseurs, on obtient un oscillateur, qui oscille à une fréquence fonction du temps moyen de propagation «  $t_d$  » de la cellule élémentaire (avec  $N = 3$ ) :

$$f_{osc} = \frac{1}{2.N.td}$$

Ne pas oublier de placer le composant « IC1 » qui permet de spécifier les Conditions Initiales d'un nœud pour une simulation temporelle.

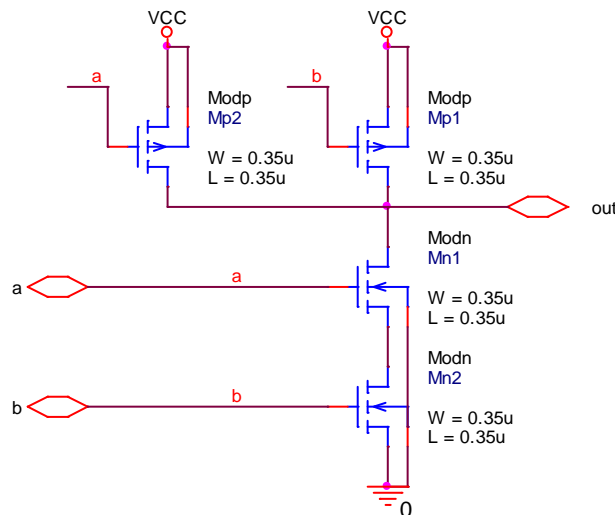
Par une analyse temporelle, estimer la valeur de la période  $T_{osc}$  d'oscillation, et en déduire «  $f_{osc}$  » et «  $td$  » (temps moyen de propagation).

✓ Donner «  $f_{osc}$  », et «  $td$  » ainsi que la courbe de simulation.

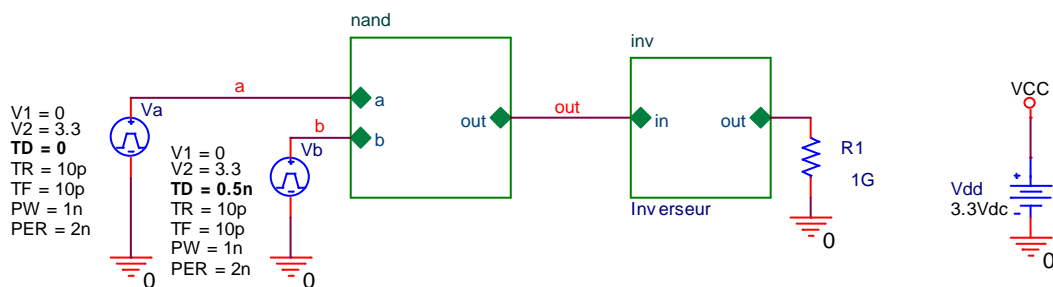
## II – Portes NAND et NOR

II-1 On se propose maintenant d'étudier le comportement d'une porte NAND réalisée avec des transistors CMOS.

Réaliser le schéma de la figure suivante dans un schéma nommé « nand », puis faire un schéma « simu\_nand » pour étudier son comportement.



*Schéma d'une porte NAND*

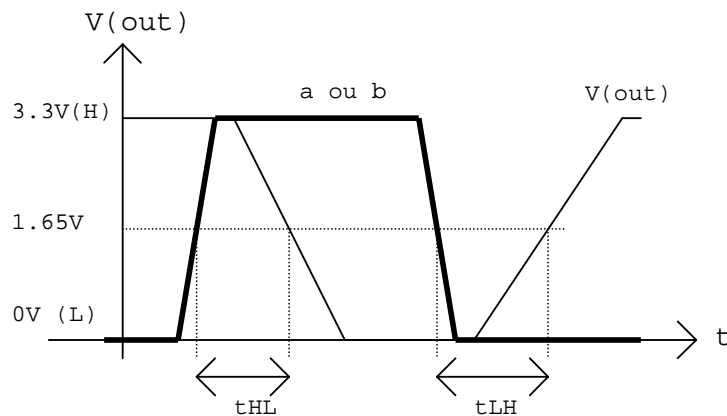


*Schéma de simulation de la porte NAND*

On fera une analyse « transient » de 6 ns, et on observera la sortie en fonction des entrées. Déterminez enfin les temps de propagation  $t_{LH}$  et  $t_{HL}$  de cette porte.

Rappel : Les délais de commutation d'une porte inverseuse sont définis de la façon suivante :

- $t_{HL}$  : temps de commutation du niveau Haut au niveau Bas de la sortie,
- $t_{LH}$  : temps de commutation du niveau Bas au niveau Haut de la sortie,



✓ Donner «  $t_{LH\_NAND}$  » et «  $t_{HL\_NAND}$  ».

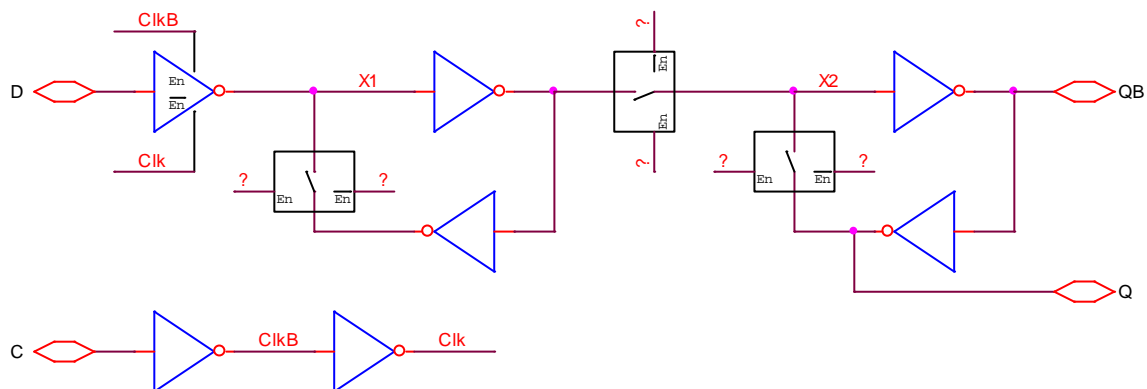
II-2 En vous inspirant du schéma de la porte précédente, dans un schéma nommé « nor » réalisez une porte NOR.

✓ Donnez le schéma de la porte NOR.

### III – Réalisation d'une Bascule D :

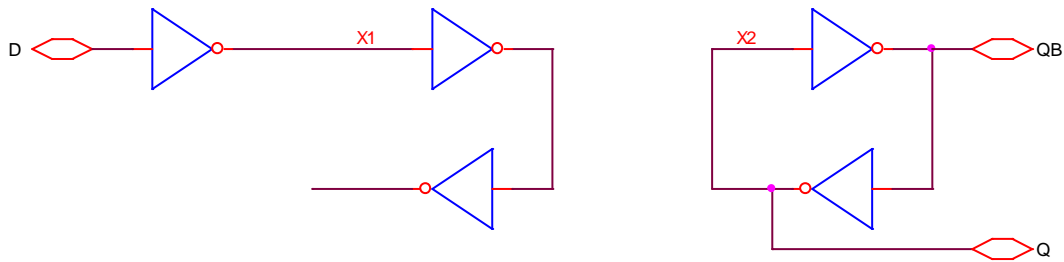
Plusieurs réalisations d'une bascule D sont possibles. On se propose ici d'étudier une architecture très couramment utilisée qui présente 2 avantages : un temps de commutation rapide et un nombre réduit de transistors.

Son schéma de principe est illustré dans la figure suivante :



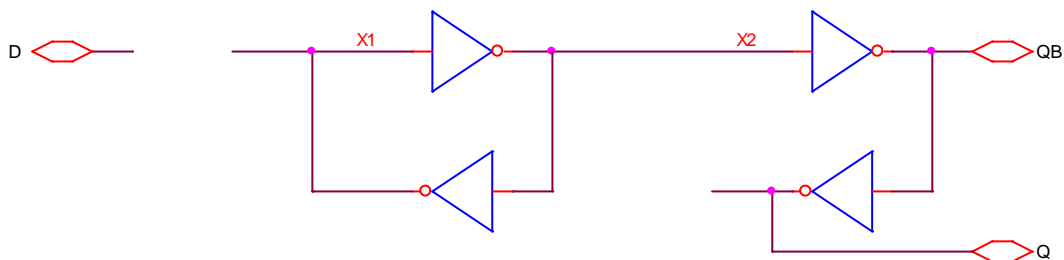
Il s'agit d'une bascule D réagissant sur front montant d'une horloge (appliquée sur l'entrée « C »). Il faut mettre les bonnes commandes sur les interrupteurs. On peut comprendre le principe de fonctionnement de cette bascule en observant son état en fonction du niveau de l'horloge (voir page suivante).

**ETAT 1 : l'horloge 'C' est à l'état bas** (on est en attente d'un front montant). On veut le schéma équivalent suivant :



- Le nœud X1 donne la valeur complémentée de l'entrée D en permanence.
- Le nœud X2 garde une valeur constante (sortie Q constante égale à la valeur de D lors d'un front montant d'horloge antérieur).

**ETAT 2 : l'horloge 'C' est à l'état haut** (on vient de passer un front montant). On veut le schéma équivalent suivant :



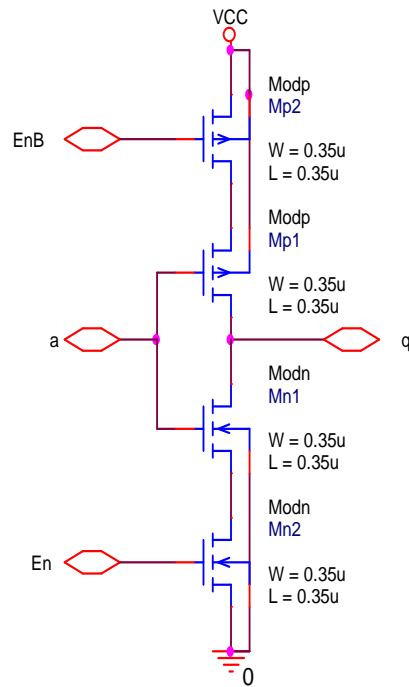
- Le nœud X1 stocke la valeur de l'entrée D complémentée lors du front montant d'horloge et garde cette valeur invariante.
- Le nœud X2 donne la valeur complémentée du nœud X1 en permanence.

On peut remarquer que l'inverseur d'entrée est un inverseur particulier puisqu'il possède un Enable.

- avec Enable = '1', il réalise correctement sa fonction d'inverseur.
- avec Enable = '0', sa sortie est Haute-Impédance, quelle que soit son entrée D.



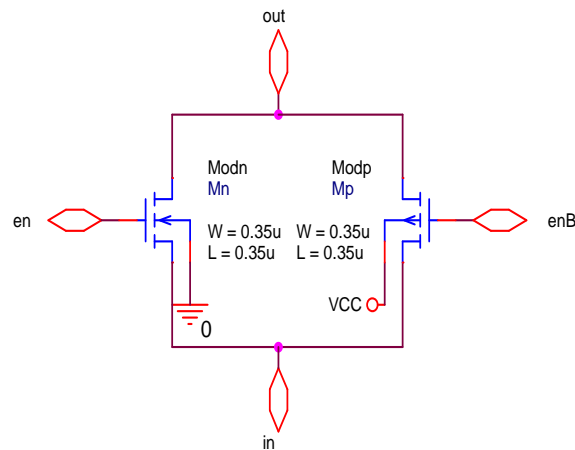
On réalise cette fonction de la façon suivante (avec entrées Enable 'En' et Enable complémenté 'EnB') :



*Inverseur avec commande Enable*

Pour piloter cette fonction logique, il faut bien sûr lui appliquer un signal En et son complémentaire EnB.

Une dernière cellule reste à réaliser pour pouvoir faire la bascule D : un interrupteur CMOS. On réalise cet interrupteur de la façon suivante :



*Interrupteur CMOS*

Son schéma est relativement simple. Son principe est le suivant :

Pour  $en = '1' = 3.3V$  ( $enB = '0' = 0V$ ), les deux transistors sont commandés à la fermeture.

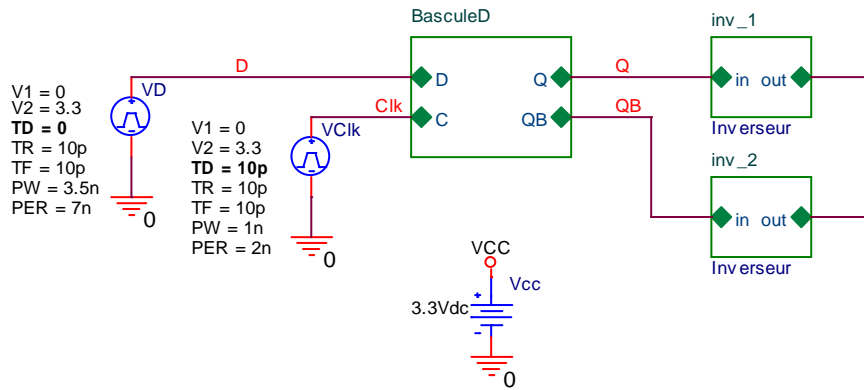
- si le signal d'entrée est proche de 3,3V le transistor PMOS restera fermé.
- si le signal d'entrée est proche de 0V le transistor NMOS restera fermé.

Pour  $en = '0' = 0V$  ( $enB = '1' = 3,3V$ ), les deux transistors resteront ouverts quel que soit le signal d'entrée.

\_ réaliser un schéma pour chaque nouveau composant : inverseur avec Enable et Interrupteur CMOS

\_ dans un schéma nommé « BasculeD » réaliser le schéma complet d'une bascule D en appliquant correctement les signaux 'Clk' et 'ClkB' aux entrées 'En' et 'EnB' des différents composants.

\_ enfin, faire un schéma de simulation « simu\_basculerD » avec les vecteurs d'entrée suivants :



*Schéma de simulation de la bascule D*

\_ Observer les signaux 'D', 'Clk', 'X1' et 'X2' (internes), 'Q' et 'QB'

\_ **✓ Rendre le schéma de la bascule D ainsi que les résultats de simulation.**

Au temps 14ns un problème intervient :

**✓ Expliquer ce phénomène.**

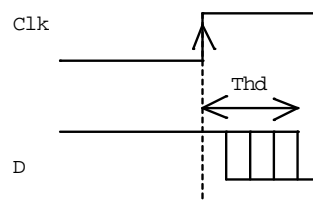
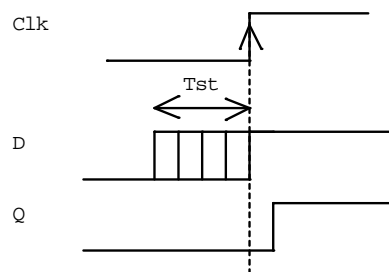
Mesurer le temps de propagation de la bascule (d'un front montant à un changement de la sortie Q) chargée avec un inverseur sur chaque sortie :

**✓ Donner le temps de propagation de la bascule «  $T_{\text{propagation}}$  ».**

**✓ Observer les temps de Setup et de Hold.**

*Temps de setup*

*Temps de hold*



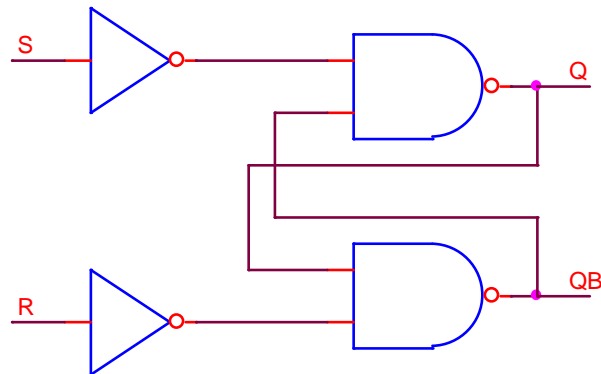
Rappels :

- Temps de setup ( $T_{\text{st}}$ ) : temps nécessaire pour garantir la transition de Q.
- Temps de hold ( $T_{\text{hd}}$ ) : temps nécessaire pour garantir la conservation de l'état de Q.

**IV – Réalisation d'une Bascule RS (travail optionnel):**

On veut maintenant réaliser une bascule RS (Reset - Set)

Voici le schéma d'une telle bascule :



*Bascule RS*

\_ Réaliser un schéma de simulation pour observer les sorties Q et QB en fonction des signaux d'entrée.

\_ ✓ Fournir le tableau suivant :

S	R	Q	QB
0	0		
0	1		
1	0		
1	1	?	?

\_ Mesurer les temps de commutation d'une telle bascule et

✓ donner «  $t_{LH\_RS}$  » et «  $t_{HL\_RS}$  ».