

PRÉPARATION (ANALOGIQUE) À LA LOGIQUE SÉQUENTIELLE

ES102 / CM5



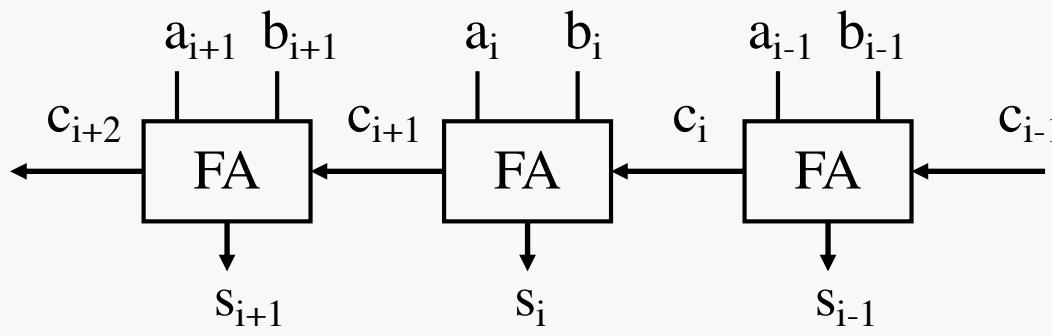
DU COMBINATOIRE AU SÉQUENTIEL

- Bloc combinatoire = investissement ;
comment maximiser sa productivité ?
 - en lui donnant si possible un nouveau calcul à faire aussitôt fini le précédent
 - délais combinatoires à mieux maîtriser
 - suppose une organisation générale favorisant l'utilisation comme nouvelles entrées de résultats juste produits
 - notamment de façon itérative (boucles)
 - besoin de mémoriser et transférer des données : usage de *bascules D*
 - besoin de coordination temporelle : logique séquentielle *synchrone*

délai → *analogique*

mémorisation → *analogique*

Le temps ne sera plus subi mais exploité méthodiquement



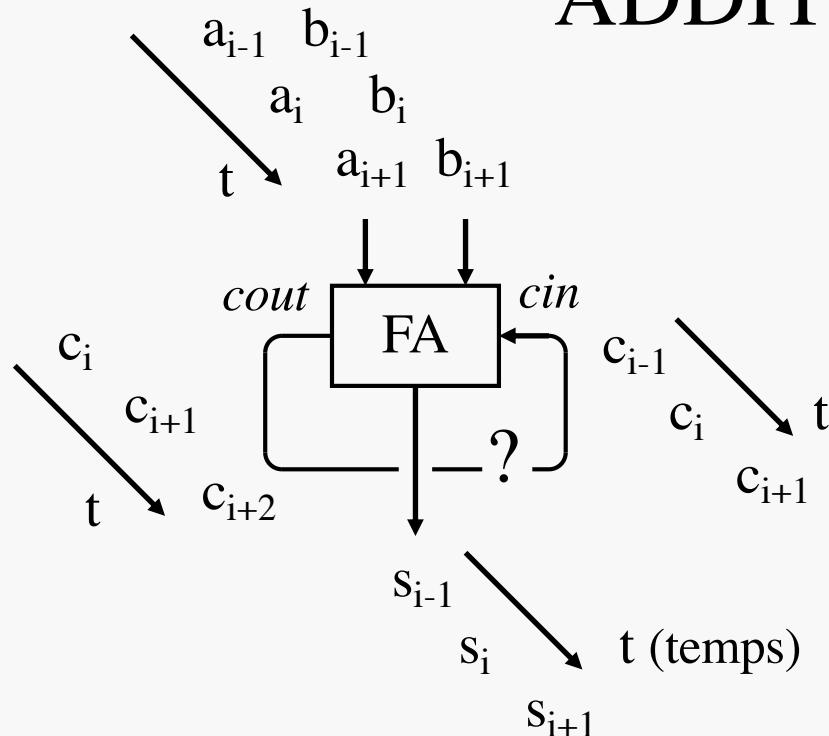
Additionneur à
retenue propagée

n FAs

(à $\sim 30t$ le FA)

UN SEUL FA POUR ADDITIONNER ?

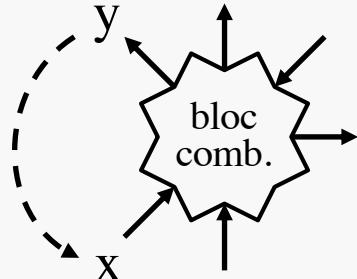
cas
d'école



Additionneur bit-série

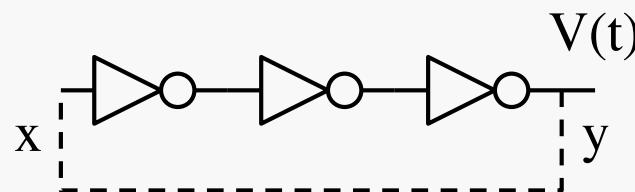
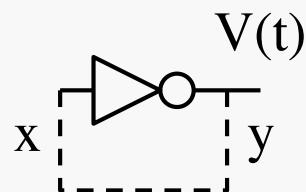
- à condition de savoir amener les (a_i, b_i) progressivement en entrée...
- bien moins coûteux (en FAs...), mais forcément moins rapide
- mais comment reboucler *cout* sur *cin* ?
 - cycle combinatoire interdit... ↵

les autres entrées fixées,
 $y(x) \in \mathcal{F}_1$

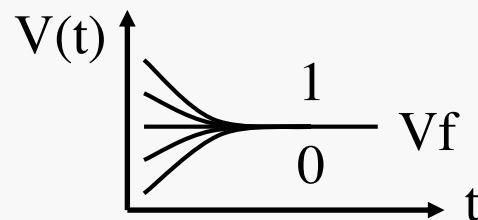
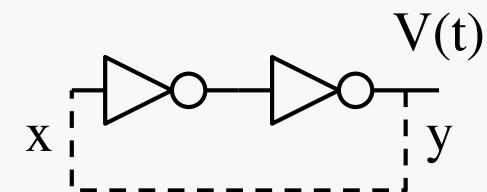


BOUCLE FERMÉE (CYCLE) COMBINATOIRE : DANGER !

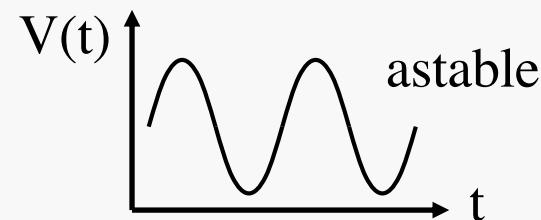
lorsque $y(x) = x'$



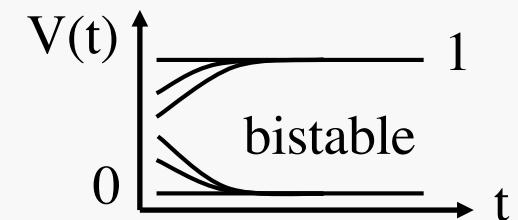
lorsque $y(x) = x$



Inverseur rebouclé
stable à $V=V_f$, frontière
entre 0 et 1 logiques

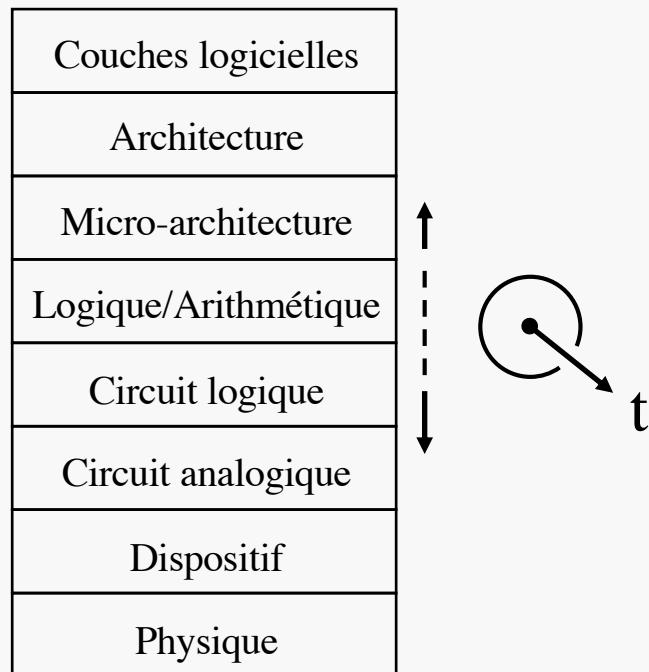


Oscillateur en anneau
3 intégrations en cascade
 \Rightarrow instable \Rightarrow oscille



Cellule mémoire
Incongru ici, mais
bientôt très utile

Comportements analogiques inadmissibles



LA BASCULE D BRIQUE SÉQUENTIELLE FONDAMENTALE

ES102 / CM5a

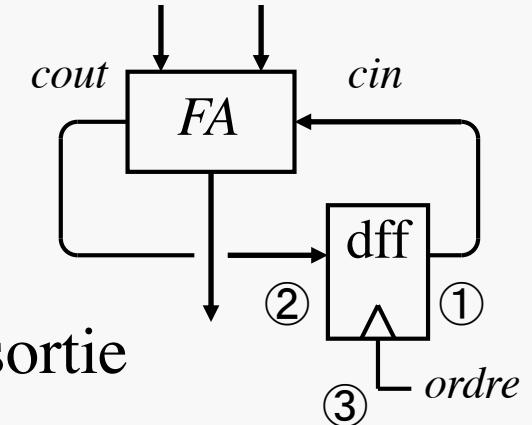
SOLUTION : LA BASCULE D

Pour reboucler un circuit combinatoire sur lui-même

(*t.q. la sortie cout d'un FA vers son entrée cin*)

il faut interposer un élément, appelé
 « *bascule D* » (*D flip-flop, dff*), capable :

- ① d'imposer une valeur (*ici c_i*) durablement sur sa sortie
 c-à-d en entrée du circuit combinatoire (*ici le FA*),
 assez longtemps pour qu'il puisse faire son calcul (*ici $\geq \tau_{FA}$*)
- ② de récupérer ensuite le résultat du calcul (*ici c_{i+1}*) sur son entrée
 et la mémoriser, tout en maintenant sa sortie inchangée
 pour ne pas perturber le circuit combinatoire pendant cette mémorisation
- ③ enfin de transférer cette valeur mémorisée vers sa sortie, sur ordre
 pour recommencer un nouveau cycle, à partir de ①

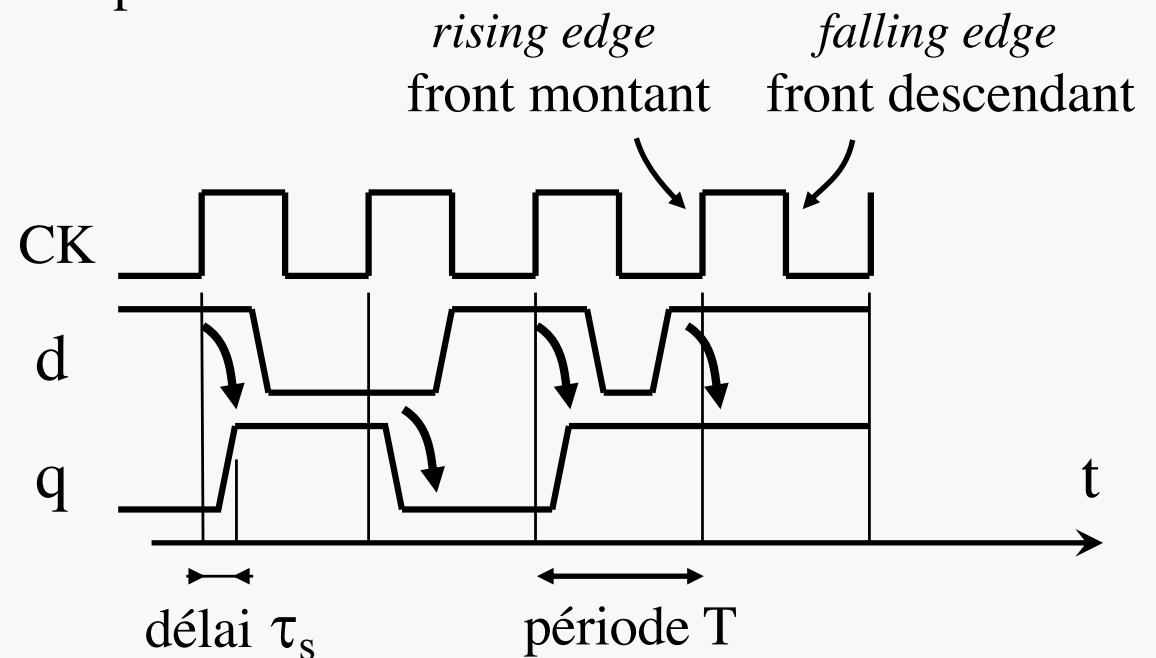
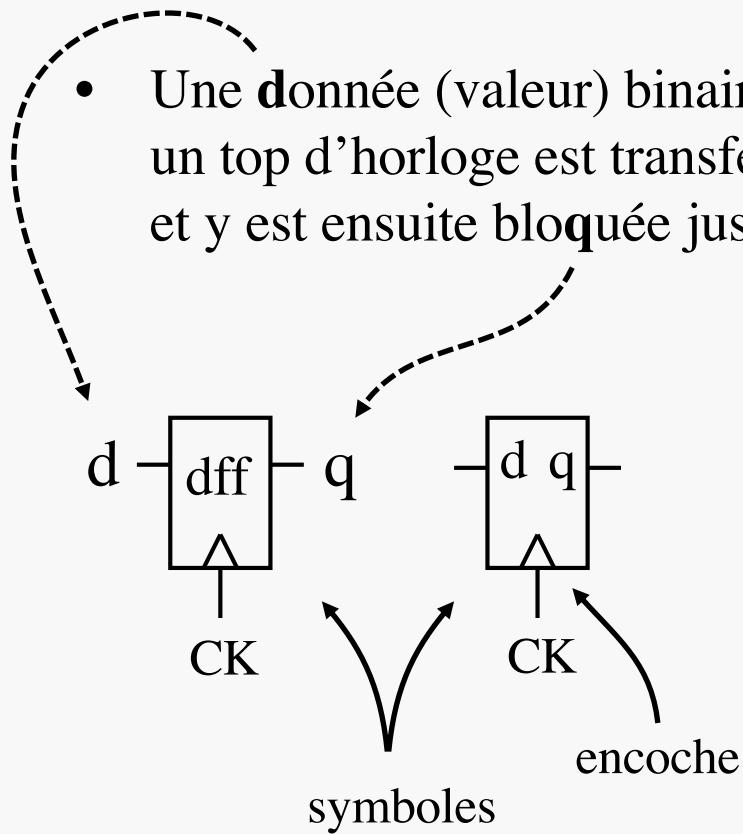


Donc 2 bits à gérer à la fois

Comment ? Réponse tout à l'heure...

BASCULE D : MODE D'EMPLOI (1)

- Ordres de transfert = *fronts montants*
(transitions de 0 à 1)
d'un signal binaire périodique appelé *horloge*
usuellement noté CK, pour ClocK
- aussi appelés *tops d'horloge (clock ticks)*
- Une donnée (valeur) binaire présente en entrée d'une bascule D juste avant un top d'horloge est transférée en sortie juste après (un certain délai τ_s) et y est ensuite bloquée jusqu'au top suivant



c'est une convention :
le choix des fronts descendants
aurait aussi été possible

BASCULE D : MODE D'EMPLOI (2)

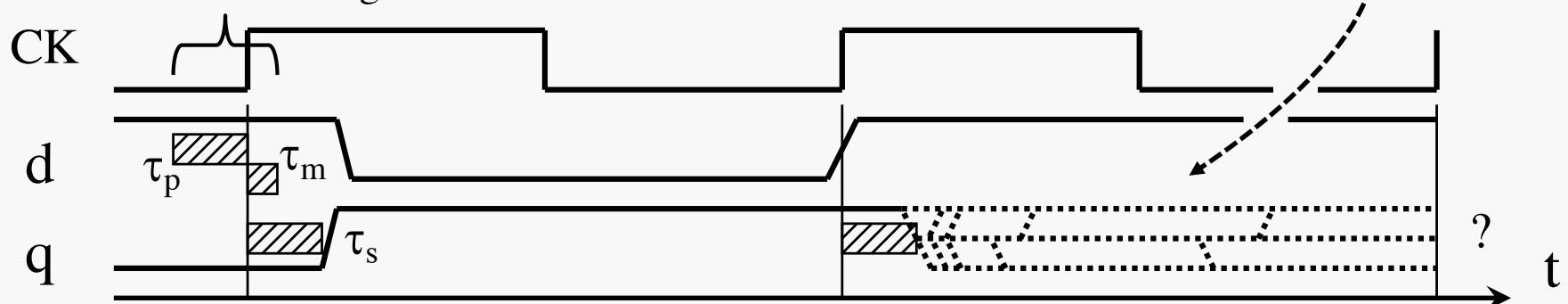
- La donnée transférée est dite *échantillonnée* au top d'horloge
 - L'entrée de la bascule D doit être stable (pas de variation) et binaire (!) sur un petit intervalle autour du top d'horloge :

The diagram illustrates the time interval $[top - \tau_p, top + \tau_m]$. Two curved arrows point from the labels "temps de prépositionnement" (*setup time*) and "temps de maintien" (*hold time*) to the endpoints of the interval, indicating the duration of these two phases relative to the central *top* position.

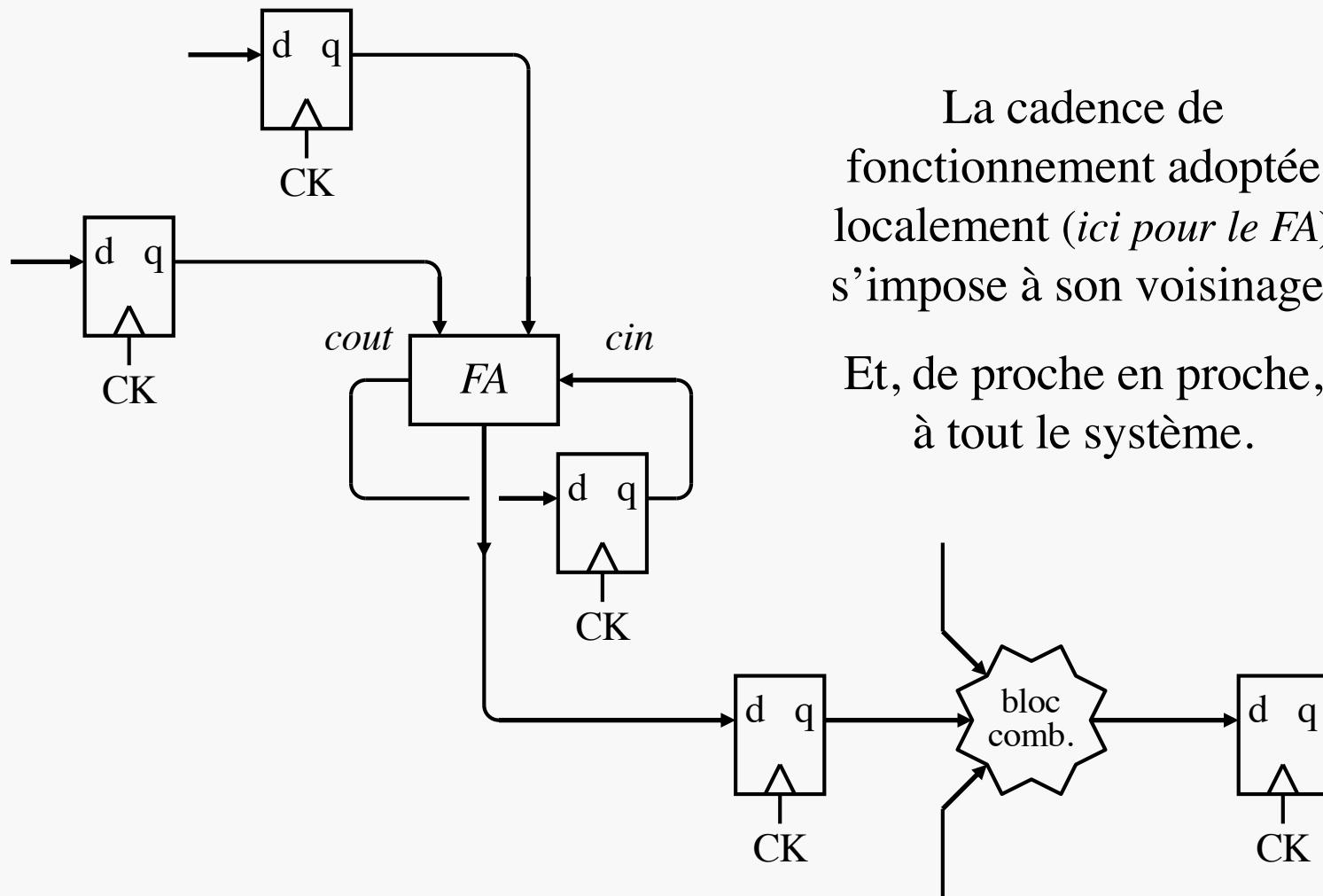
- Une entrée de bascule D respectant cette condition de stabilité, à chaque top de l'horloge CK, est dit **synchrone** (avec CK)

- Sinon, comportement aléatoire de la bascule D :
 - qui finira par présenter une valeur binaire mais imprévisible sur sa sortie
 - en un délai très souvent court, mais malheureusement non borné...

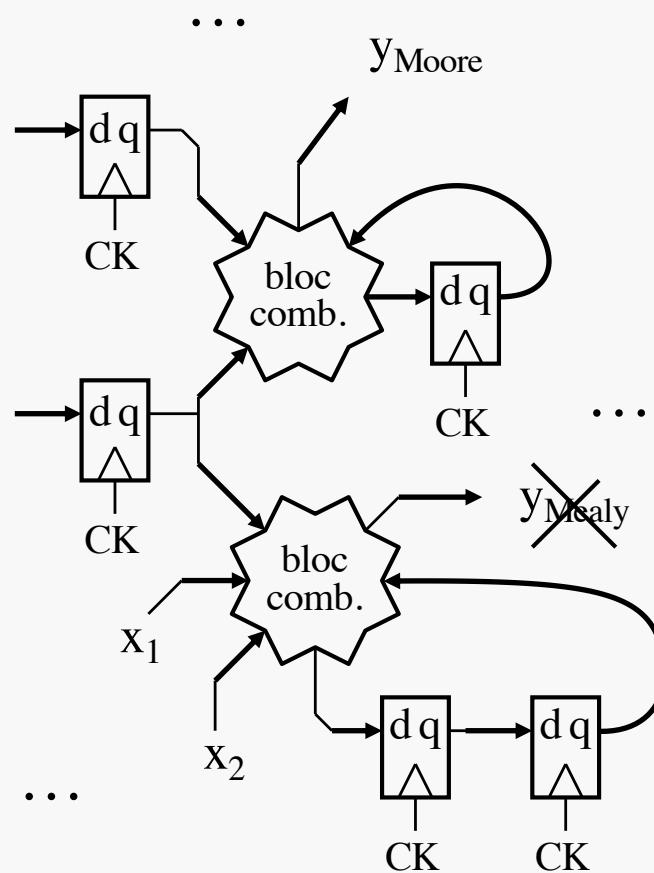
intervalle d'échantillonnage



CONTAGION SYNCHRONE...

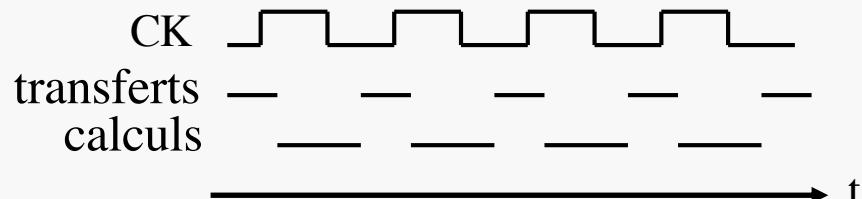


CIRCUIT SÉQUENTIEL SYNCHRONE



= ensemble de bascules D ⚡

- toutes commandées par la même horloge CK
- interconnectées par des blocs combinatoires
- avec des entrées (t.q. x₁) et sorties (t.q. y_{Moore})
- horloge = chef d'orchestre
- toutes les bascules transfèrent simultanément
- temps découpé en une alternance transferts/calculs

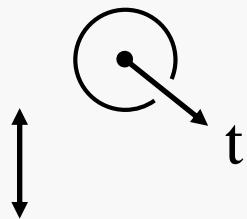


- les entrées doivent être des signaux synchrones
(sans oublier les délais combinatoires avant bascules D)

- sortie du circuit = probable entrée d'un autre circuit séquentiel cadencé par CK
 - certifiable synchrone si ne dépend combinatoirement que de sorties de bascules D
 - par précaution, on se limitera à de telles sorties, dites de type *Moore*
 - évitant celles dépendant combinatoirement d'une entrée, dites de type *Mealy*



Couches logicielles
Architecture
Micro-architecture
Logique/Arithmétique
Circuit logique
Circuit analogique
Dispositif
Physique



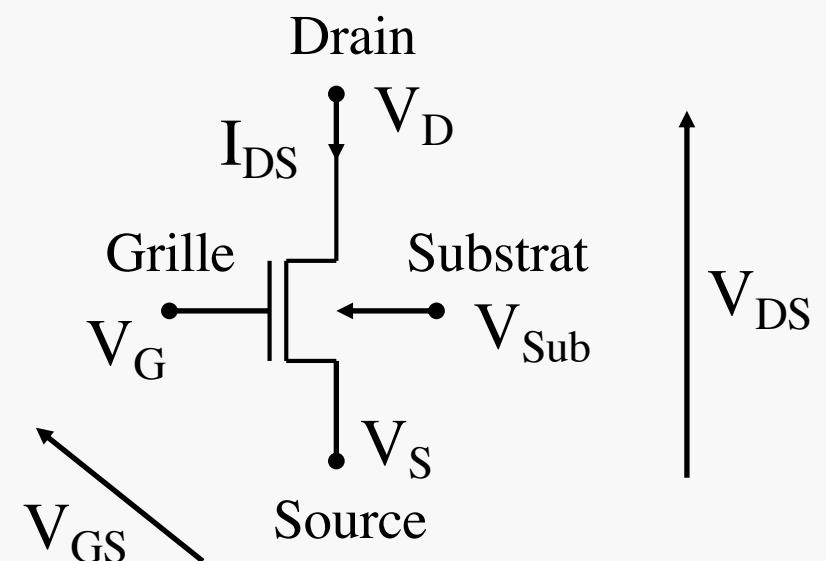
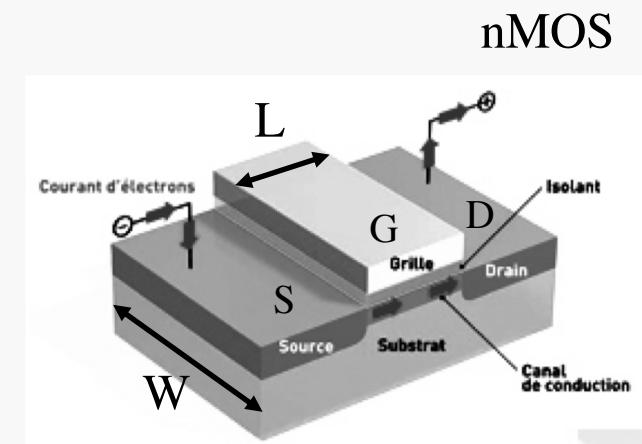
FONDEMENTS ANALOGIQUES ENTRE CONTINU ET DISCRET

ES102 / CM5b

LE TRANSISTOR MOS

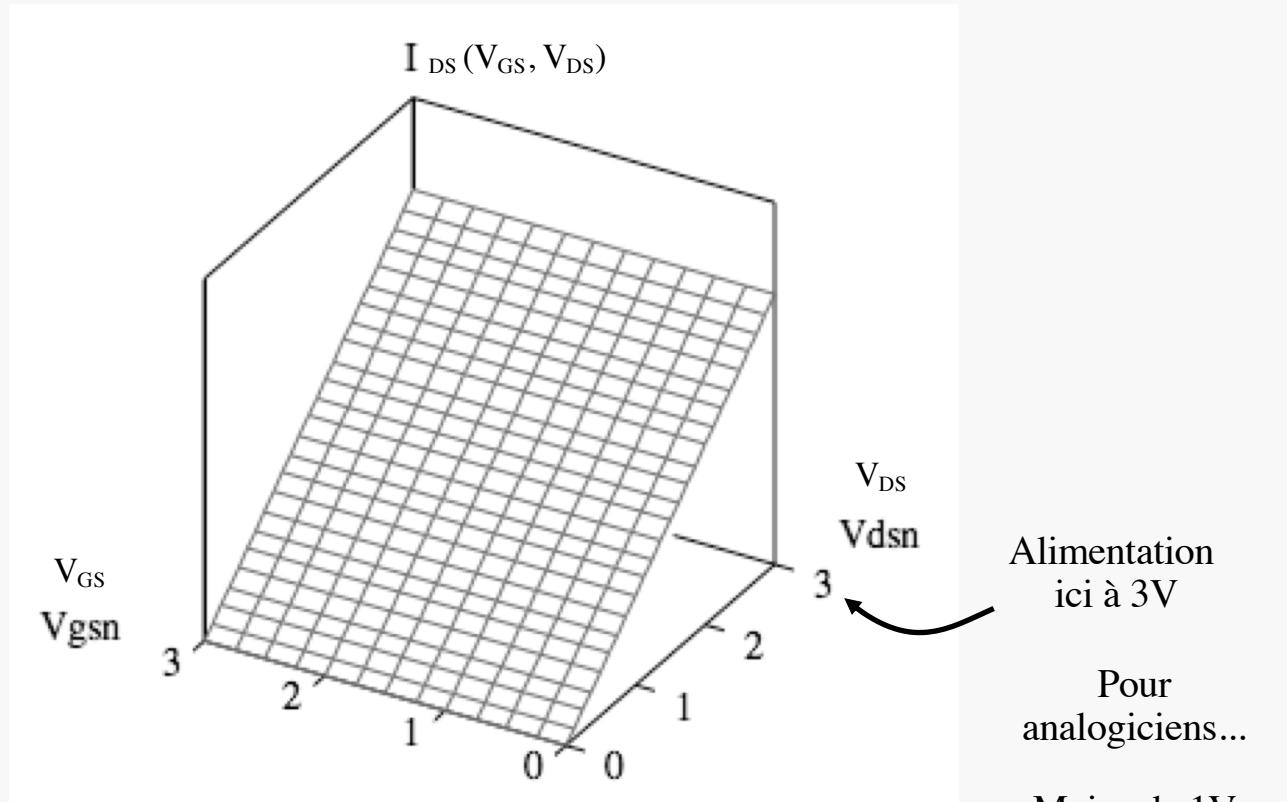
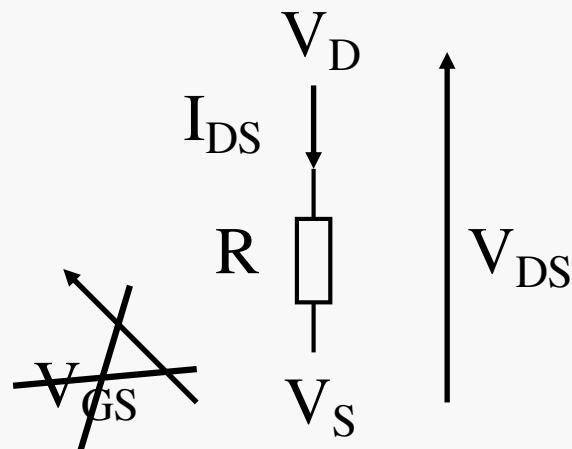
DEUX TENSIONS, UN COURANT

- 4 terminaisons : *Grille*
Source Substrat Drain
- Chacune à un certain potentiel :
 V_G, V_S, V_D et V_{Sub}
- V_{Sub} , à la masse pour un nMOS ②, disparaît
- Et presque toujours : ③ $V_S = V_{Sub}$
- Le courant drain-source I_{DS} ne dépend plus que de $V_{GS} = V_G - V_S$ et de $V_{DS} = V_D - V_S$
- ① $V_{GS} > V_{tn}$ & ④ $V_{DS} > 0 \Rightarrow I_{DS} > 0$
- mais comment I_{DS} se comporte-t-il ?
 - examinons la surface $I_{DS}(V_{GS}, V_{DS}) \dots \leftarrow$
 - proportionnel à W/L



SURFACE COURANT-TENSIONS : REPRÉSENTATION 3D

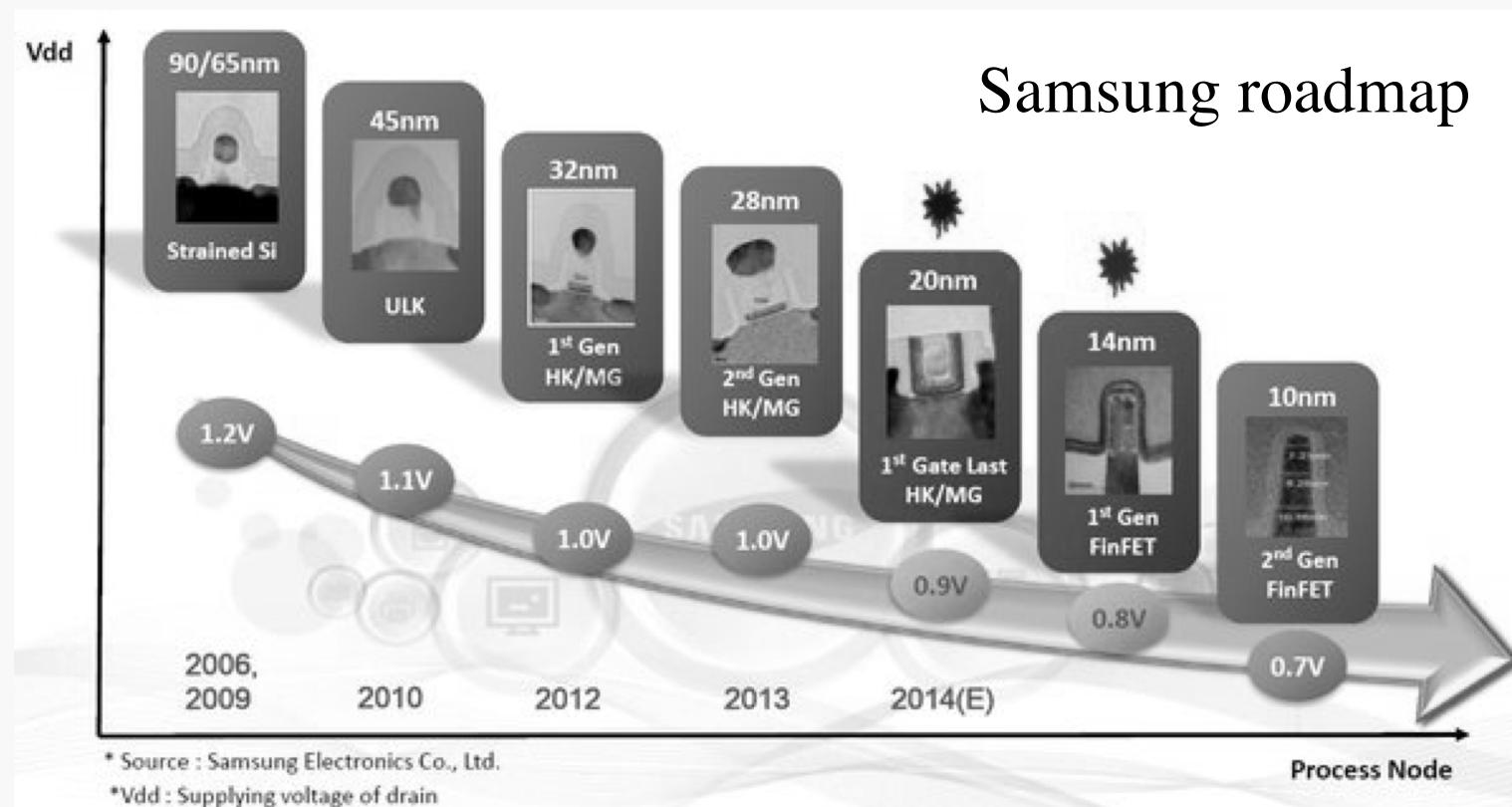
Si le nMOS était une simple résistance...



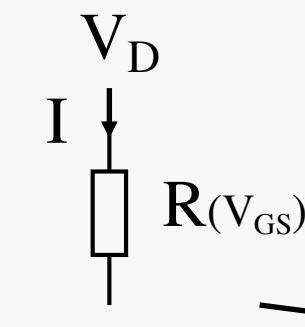
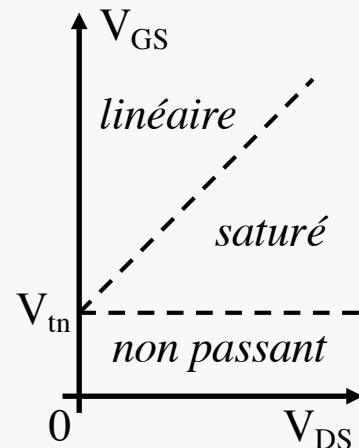
... la surface $I_{DSn}(V_{GSn}, V_{DSn})$ serait un plan,
incluant l'axe { $V_{DSn}=0$ & $I_{DSn}=0$ }

Mais effets de seuil et de saturation rendent le (n)MOS fortement non-linéaire (↔ ↔)

TENSIONS D'ALIMENTATION SUR LES TECHNOLOGIES SAMSUNG ...

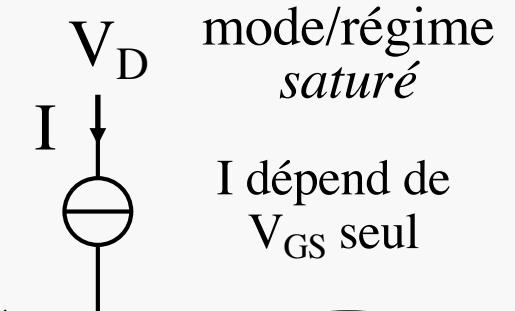
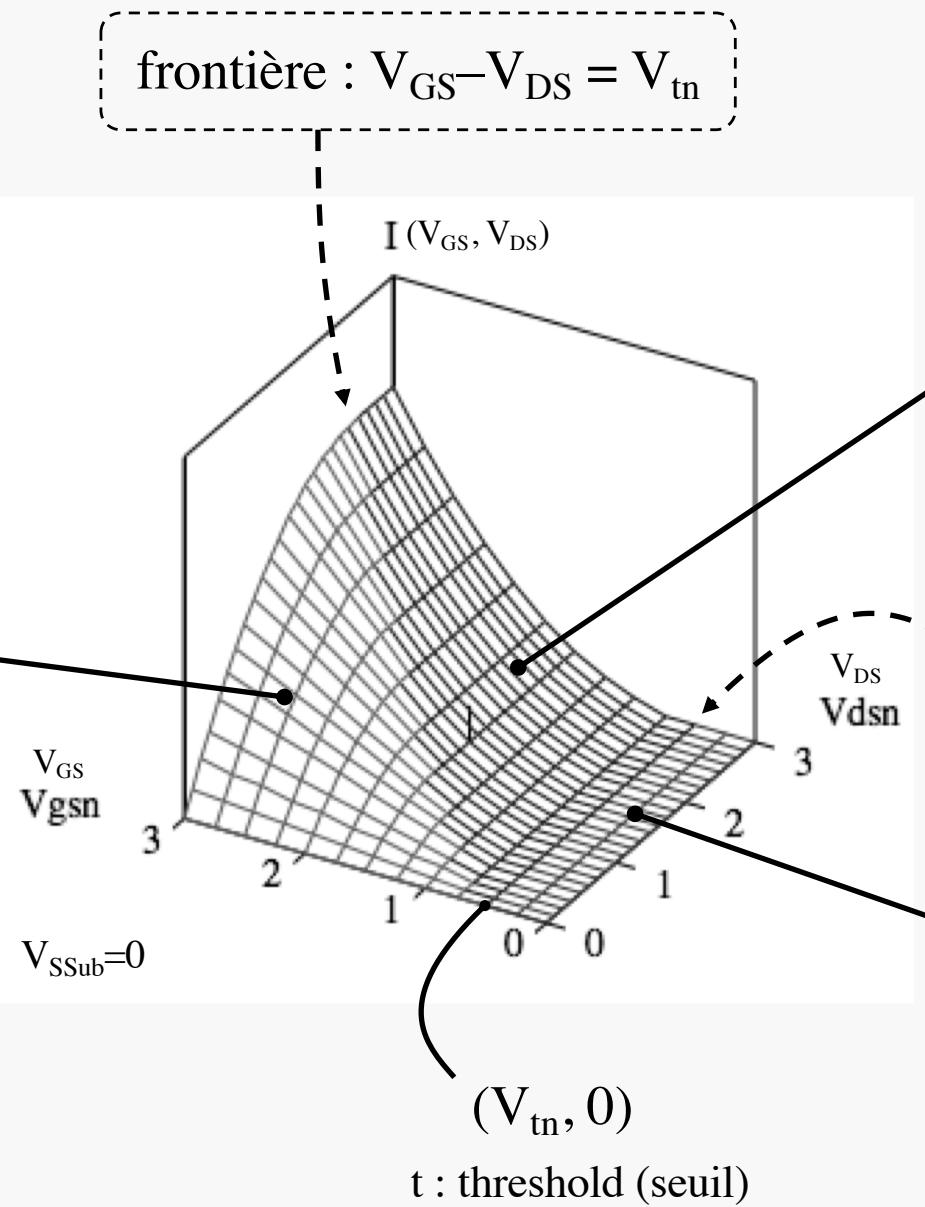


nMOS : $I_{DSn}(V_{GSn}, V_{DSn})$ en 3D



mode
linéaire
(ohmique)

canal
faiblement
« drainé »

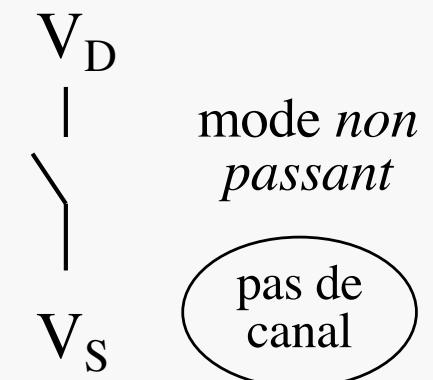


mode/régime
saturé

I dépend de
 V_{GS} seul

canal
totalement
« drainé »

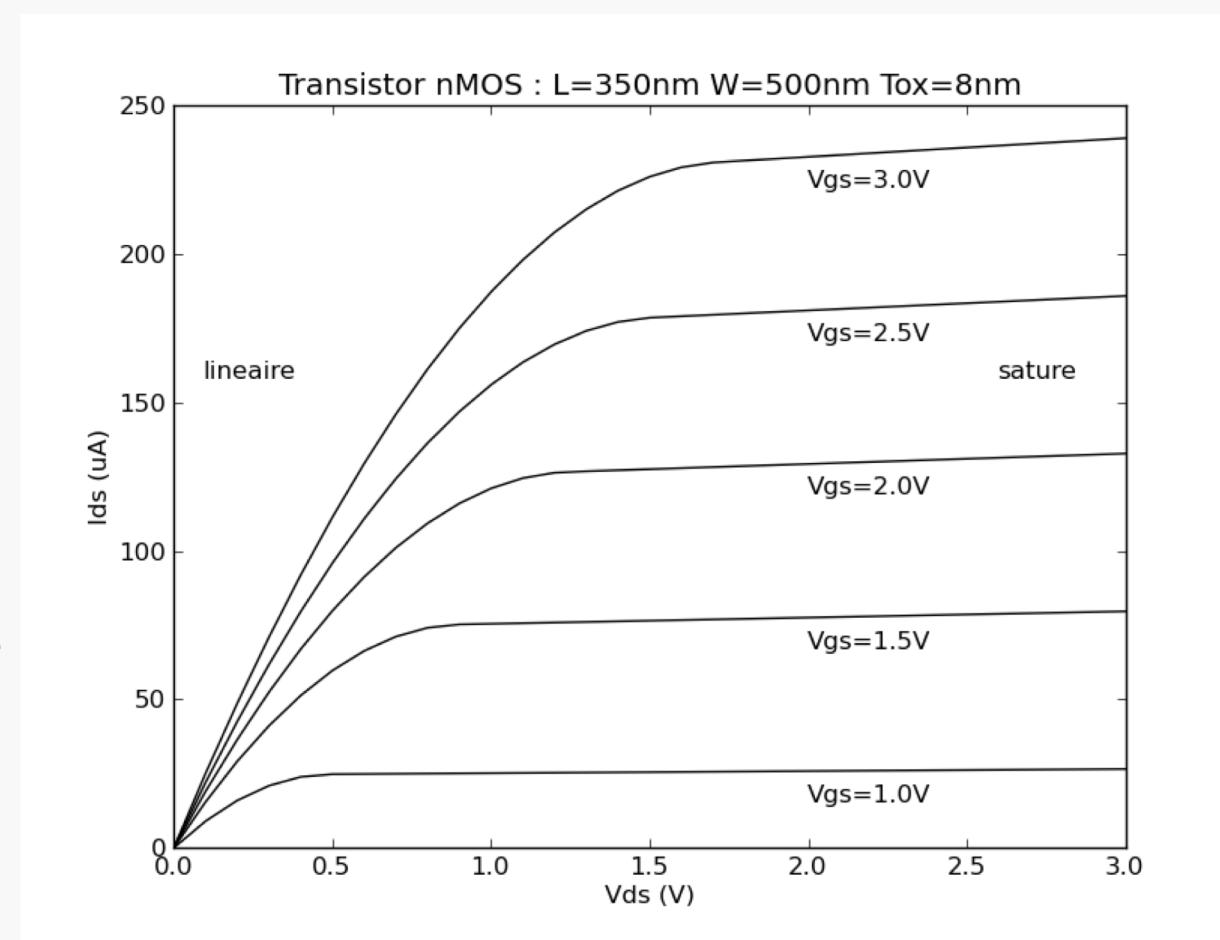
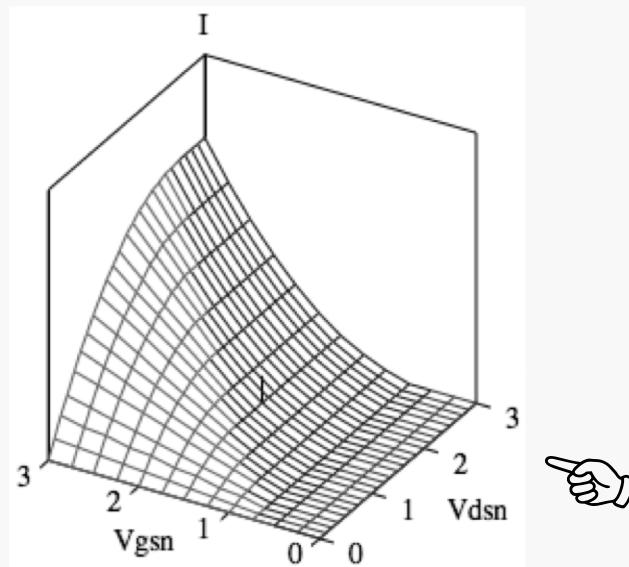
frontière
 $V_{GS} = V_{tn}$



pas de
canal

TRANSISTOR nMOS (EN 2D) :

CARACTÉRISTIQUES I_{DSn} | V_{GSn} (V_{DSn})



*Planche déplacée par rapport à 2019
(laissée par souci de cohérence avec la vidéo)*

MOS HÉMIPLÉGIQUE

- Essayons de véhiculer du 1 logique via un transistor nMOS (au lieu d'un pMOS)

– expérience : charge d'une capacité C_{out} initialement déchargée ($V_{out}=0$)

– situation inhabituelle :

- la source d'électrons est C_{out}
- tension variable sur la source : $V_{Sn}=V_{out}$
- tension fixe sur le drain : $V_{Dn}=Vdd$

– en début de charge, $V_{GSn}=V_{DSn}=Vdd=3V$

– le courant fait monter $V_{out} = Vdd - V_{DSn}$

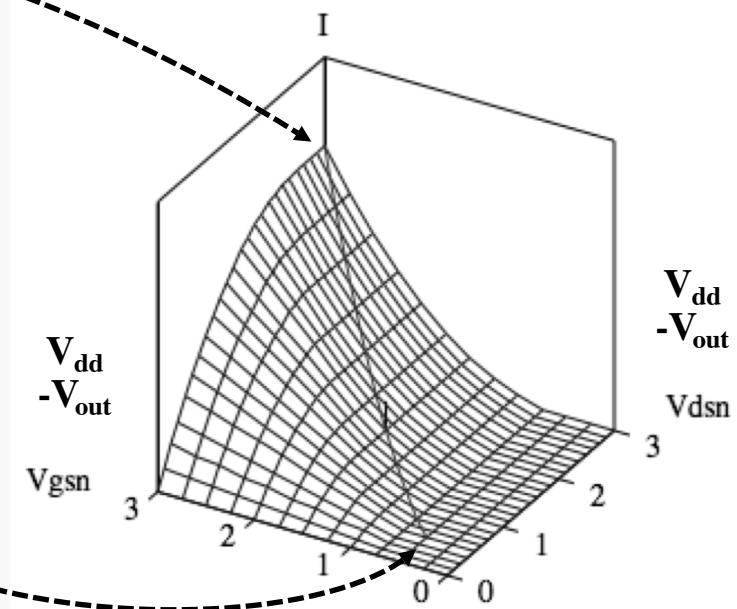
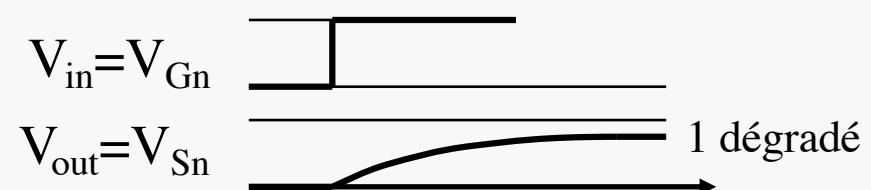
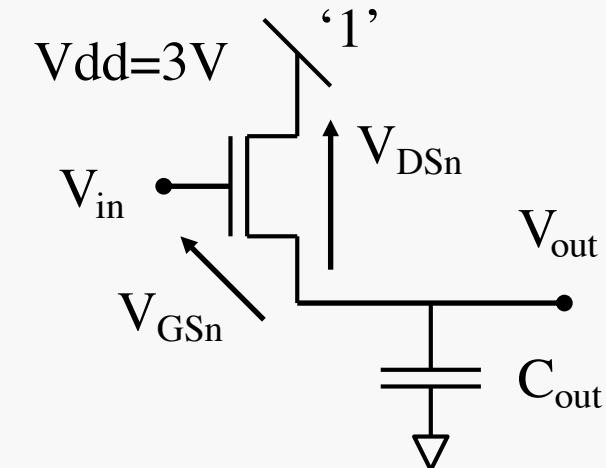
$\Rightarrow V_{GSn}$ et V_{DSn} , toujours égales, diminuent

- descente en diagonale

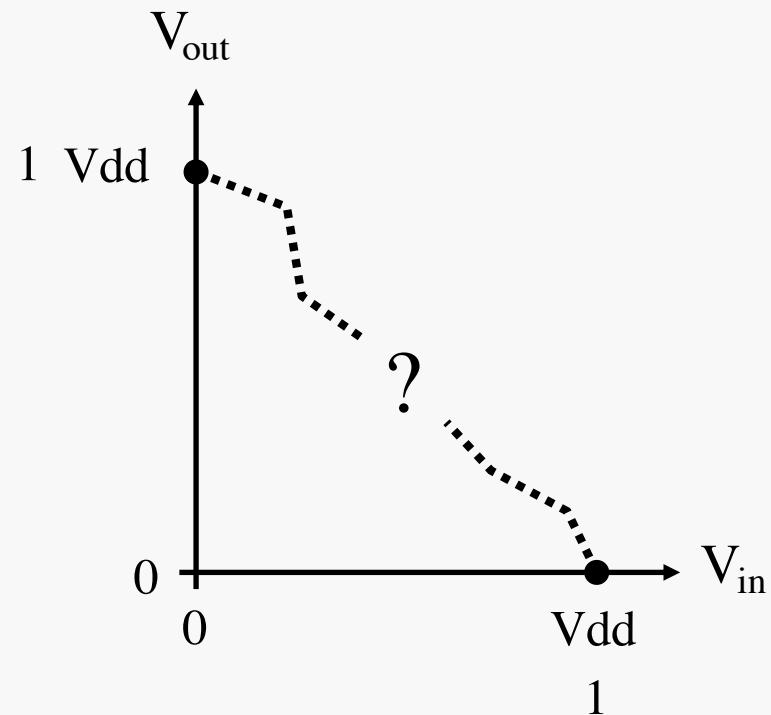
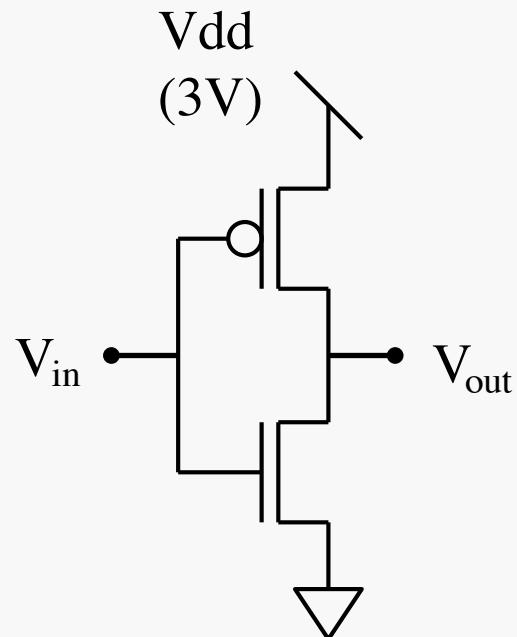
tant que le nMOS demeure passant, donc jusqu'à $V_{DSn}=V_{tn}$, d'où finalement $V_{out} = Vdd - V_{tn}$
c'est-à-dire un *1 logique dégradé* !

- dégradation pire en réalité, car $V_s \gg V_{sub}$ (règle ③ violée), ce qui écrase la surface

\Rightarrow *Il est effectivement impossible pour un nMOS de véhiculer du 1 !*
Ni pour un pMOS de véhiculer du 0 !



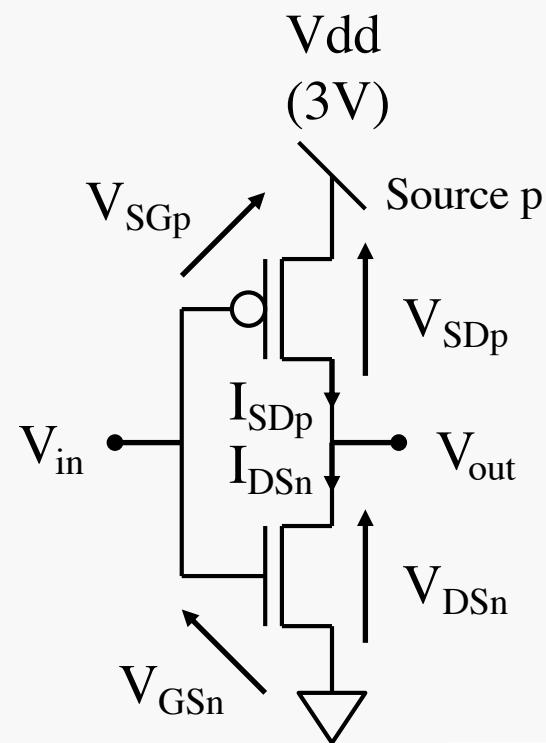
INVERSEUR CMOS : $V_{out}(V_{in})$



Que se passe-t-il entre 0 et 1 ?

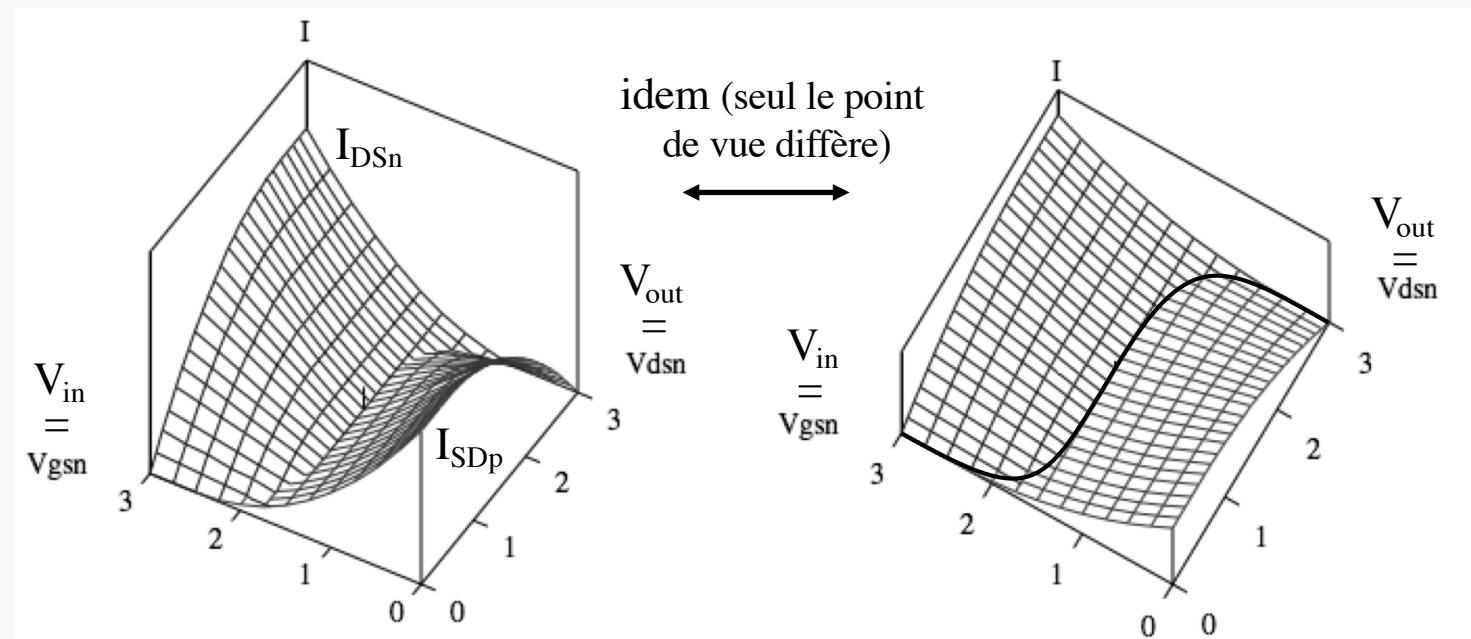
Des propriétés importantes en dépendent,
qui se généraliseront à toute porte CMOS

INVERSEUR CMOS (EN 3D)



toutes
grandeur
positives

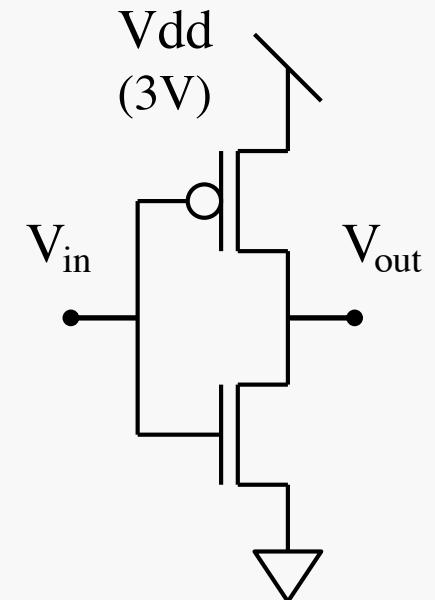
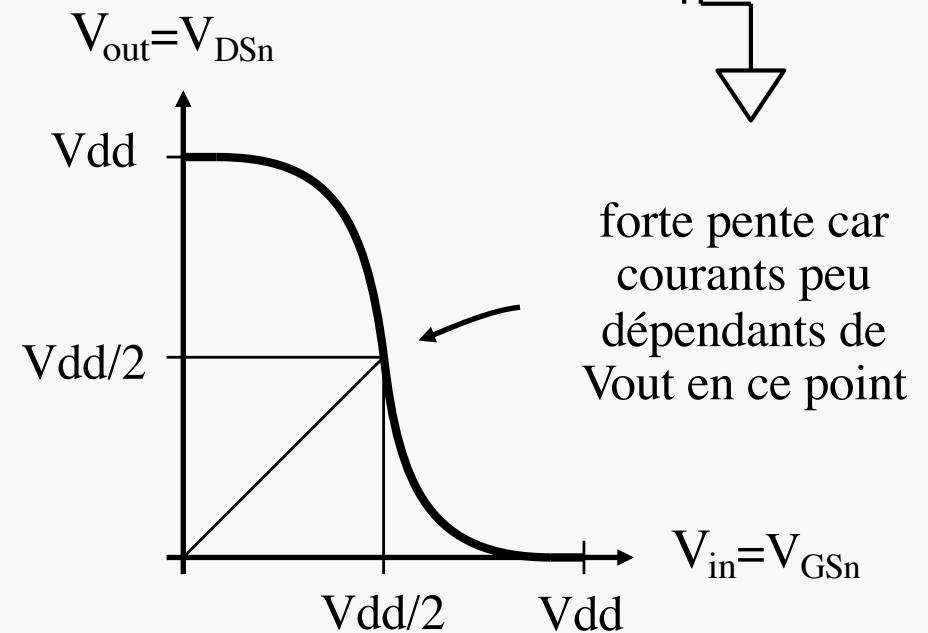
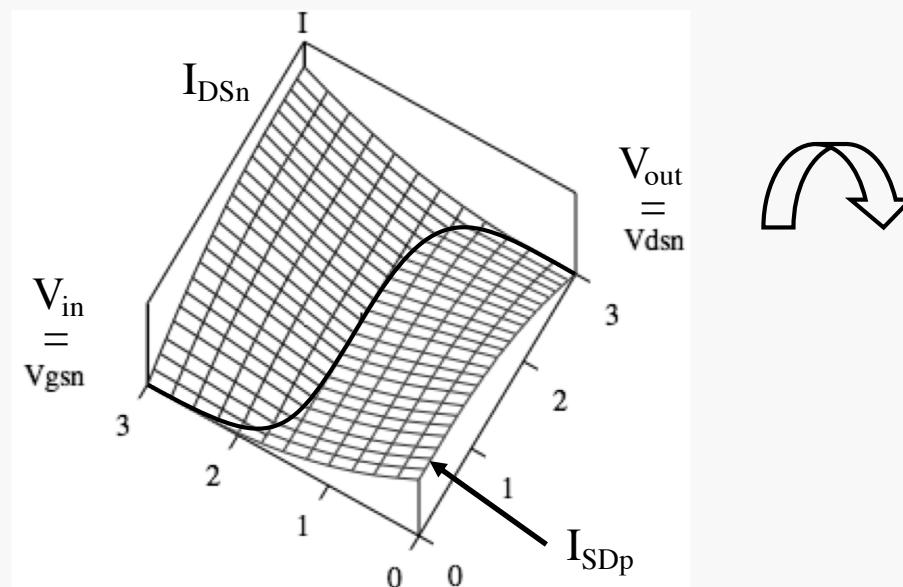
- V_{in} et V_{out} (considérés indépendants) déterminent I_{SDn} et I_{SDp}
 - nMOS : $V_{GSn} = V_{in}$ et $V_{DSn} = V_{out}$ \rightarrow surface 1
 - pMOS : $V_{SGp} = (V_{dd} - V_{in})$ et $V_{SDp} = (V_{dd} - V_{out})$
 $\Rightarrow I_{SDp}$ maximal pour $V_{in} = V_{out} = 0$ \rightarrow surface 2
- mais $I_{SDp} = I_{DSn}$
 - car courant nul sur la sortie en régime permanent
 $\Rightarrow V_{out}$ s'asservit sur V_{in} pour rendre les courants égaux en se positionnant à l'intersection des 2 surfaces



INVERSEUR CMOS (DU 3D AU 2D) : $V_{\text{out}}(V_{\text{in}})$

L'intersection des 2 surfaces $I_{DSn}(V_{\text{in}}, V_{\text{out}})$ et $I_{SDp}(V_{\text{in}}, V_{\text{out}})$ est une courbe 3D, dont la projection verticale fournit la caractéristique $V_{\text{out}}(V_{\text{in}})$

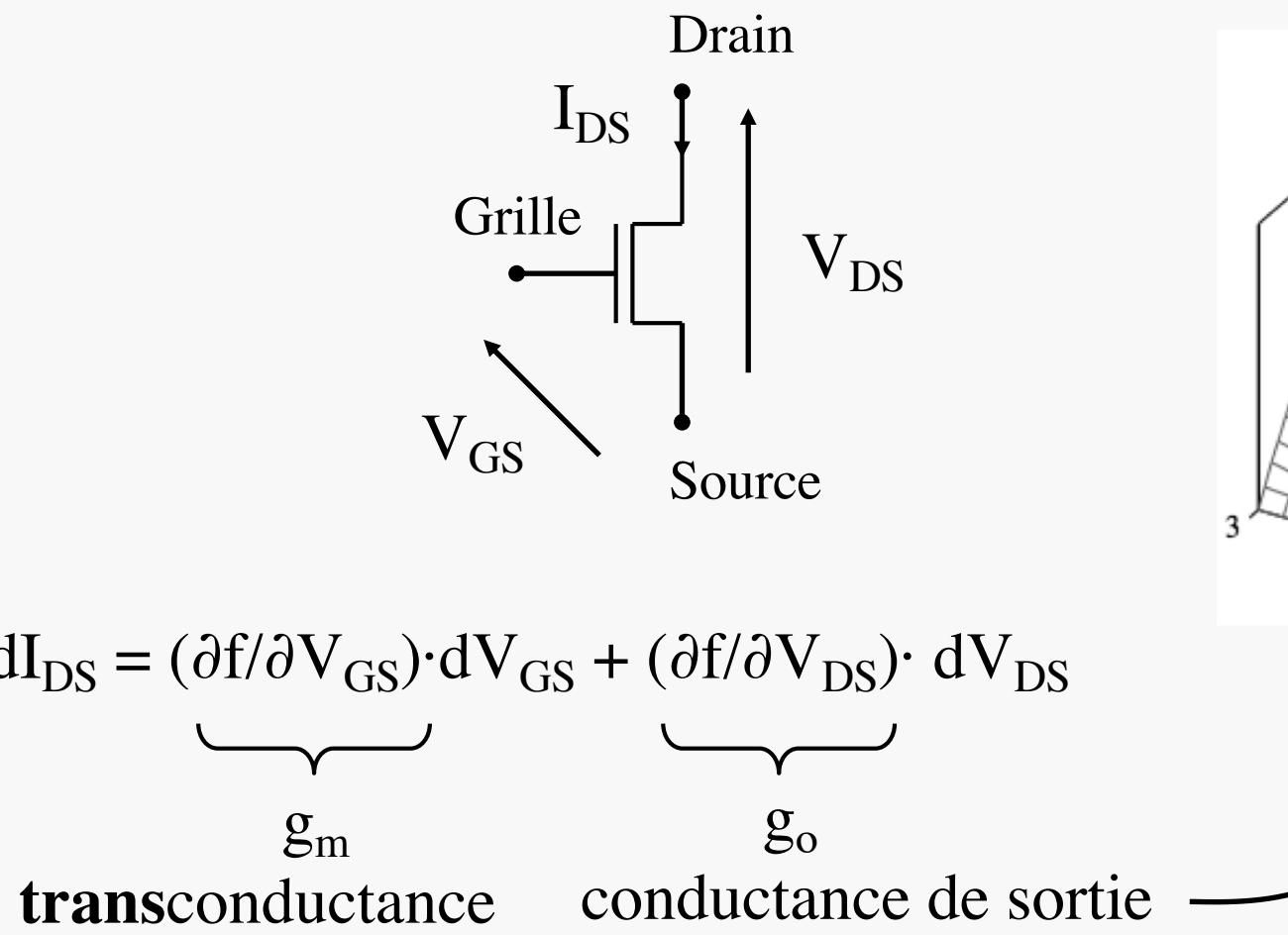
Si pMOS et nMOS de même « force » ($\downarrow \downarrow$),
surfaces ci-dessous ~ symétriques et
 $(V_{dd}/2, V_{dd}/2)$ centre de symétrie



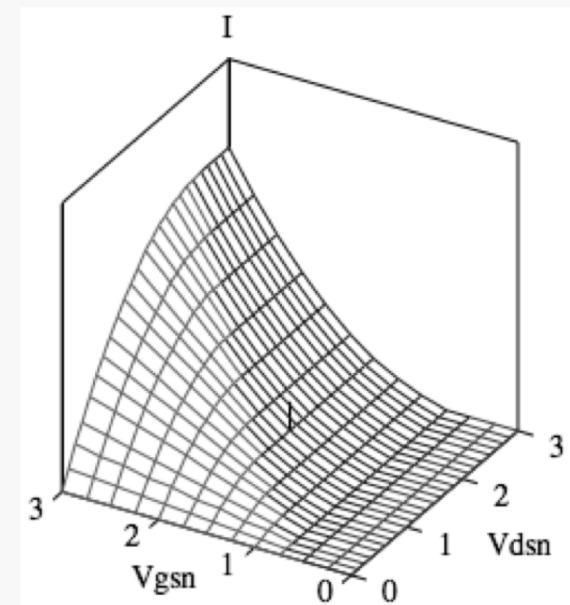
ANALYSE DIFFÉRENTIELLE

alias *analyse en petits signaux*

→ PC5/Exo1



$$I_{DS} = f(V_{GS}, V_{DS})$$

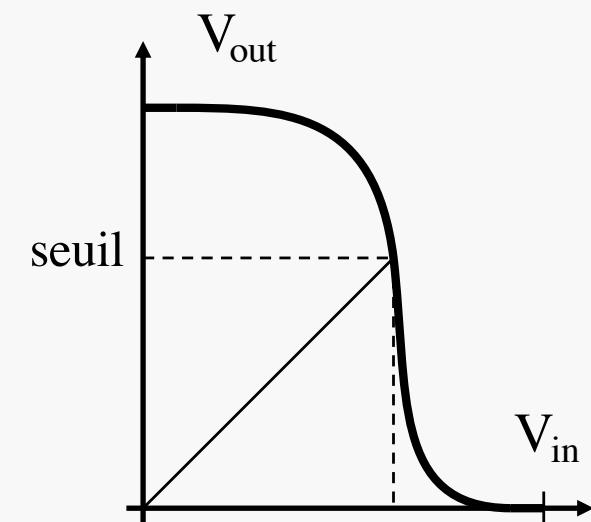
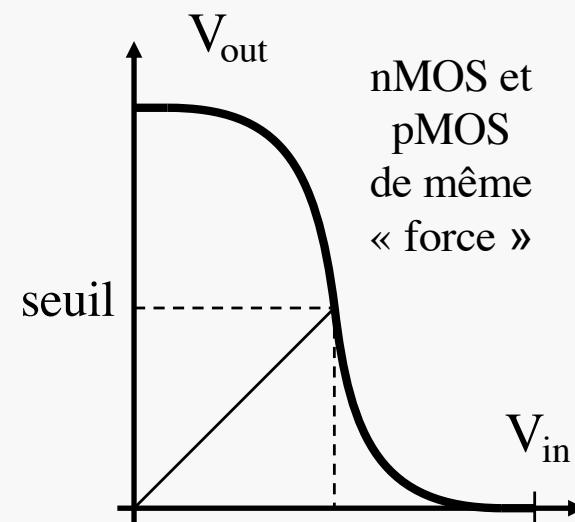
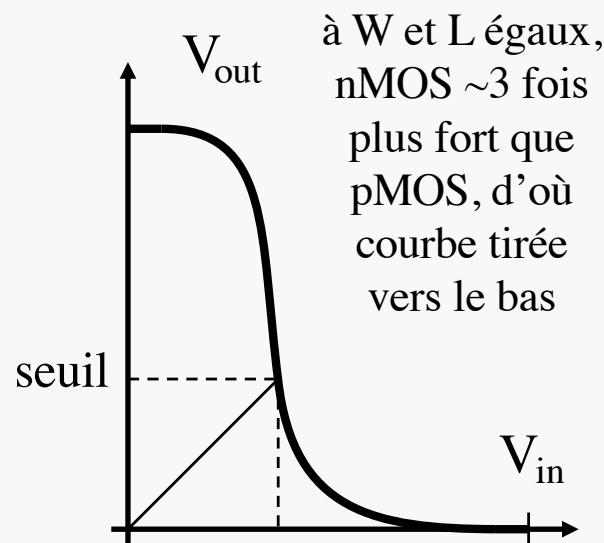
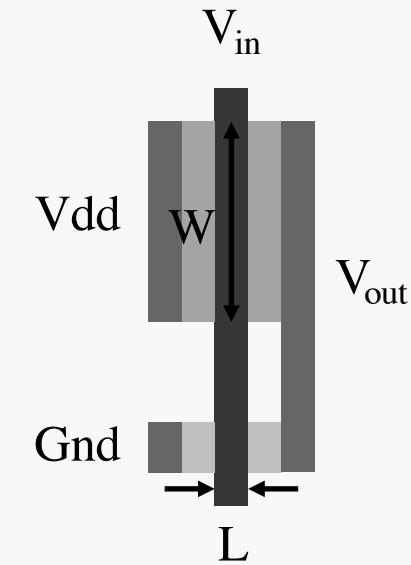
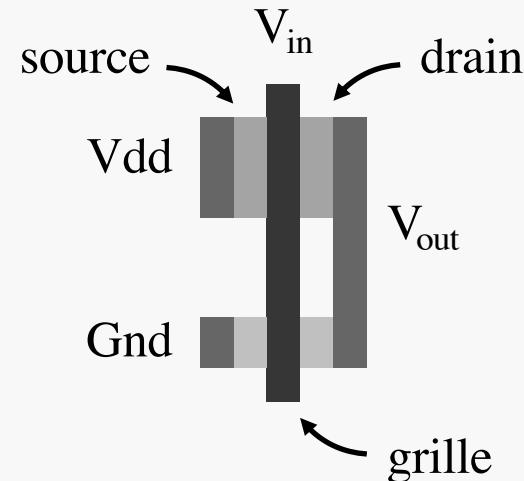
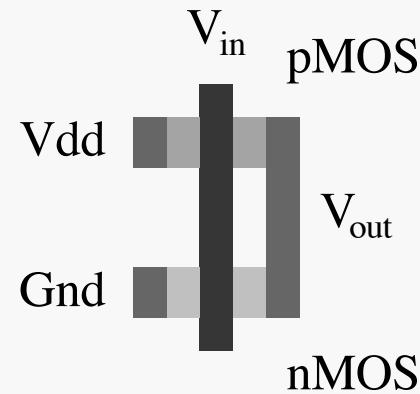


$$\frac{1}{g_o} = R_o$$

résistance
de sortie

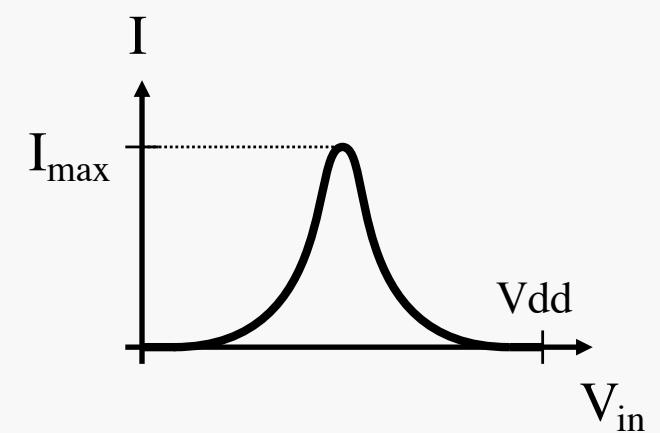
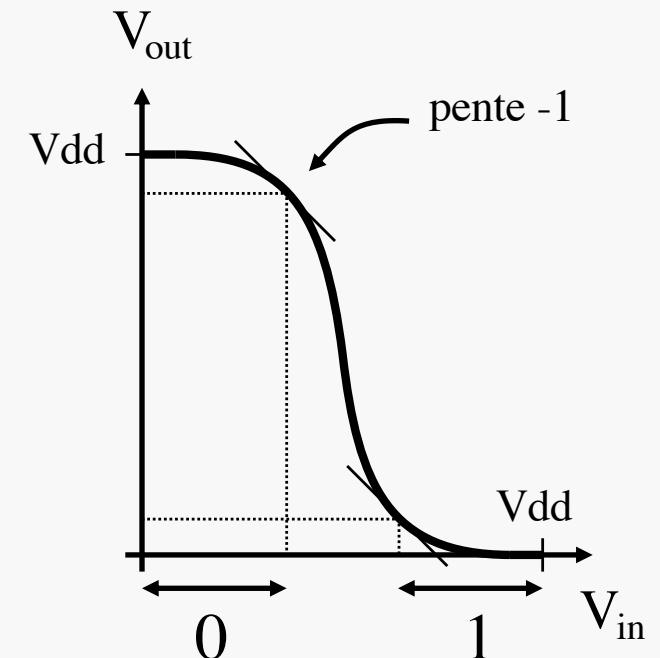
INVERSEUR CMOS : ÉQUILIBRAGE

Transistors représentés graphiquement en vue de dessus



INVERSEUR CMOS : VERTUS

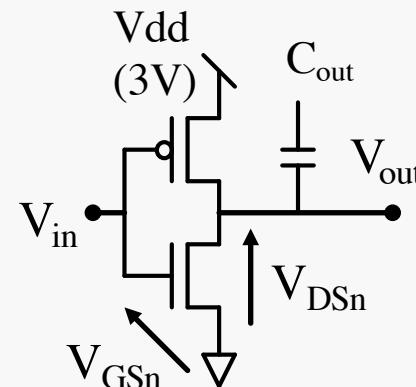
- Robustesse
 - l'agitation interne d'un circuit perturbe les niveaux logiques
 - par couplages capacitifs & inductifs
 - l'inverseur CMOS supporte ces perturbations et les « filtre »
→ niveaux logiques « régénérés »
- Sobriété
 - si niveaux logiques assez « propres » en entrée, pas de courant consommé
⇒ consommation lors des commutations seules
- Vertus dont héritent toutes les portes CMOS
 - un circuit CMOS dont les entrées sont statiques et « propres » ne consomme pas de courant !
 - quoique, courant par effet Tunnel à travers l'isolant de grille vu sa minceur désormais extrême (quelques couches atomiques)
 - remèdes *ad hoc* dans technologies récentes



INVERSEUR CMOS EN COMMUTATION → DÉLAI COMBINATOIRE

$V_{in} : 0 \rightarrow 1 \Rightarrow$

- $V_{GSn} : 0V \rightarrow 3V$
- pMOS → non passant
- nMOS → passant
 $\Rightarrow V_{out} = V_{DSn} : 3V \searrow \dots$
- source de courant pour commencer
- puis résistance une fois $V_{out} = V_{DSn}$ assez diminué

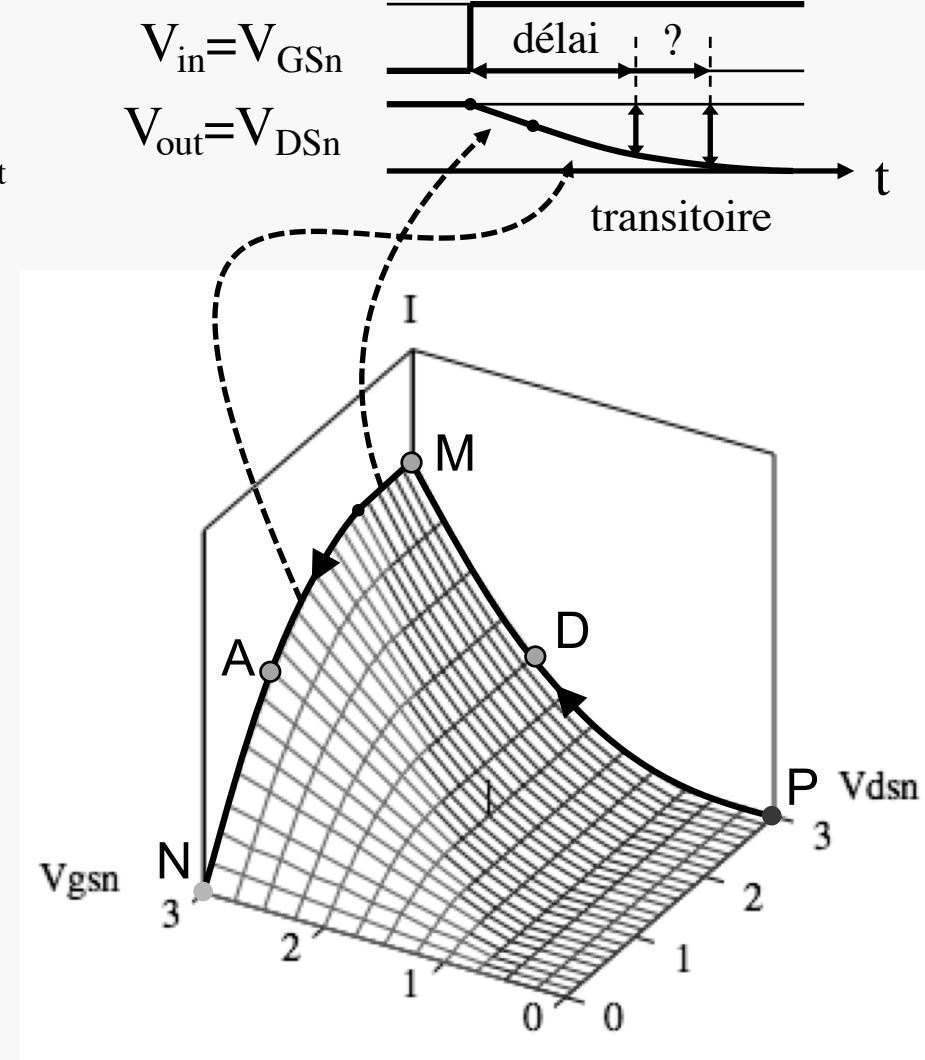


Fin de descente lente : $e^{-t/RC_{out}}$

Quand arrive-t-on à 0 (en N) ?

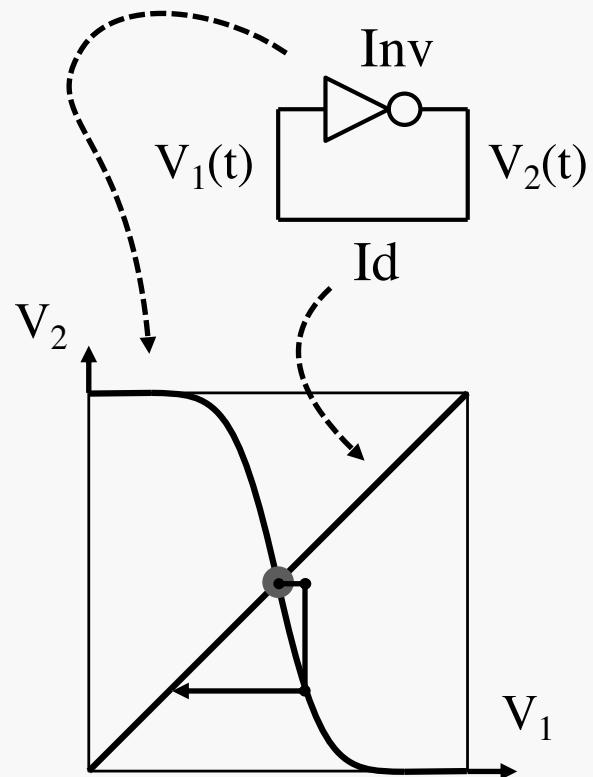
Pb mal posé, car même lenteur pour V_{in} à atteindre 1 auparavant, forcément

- transition initiale non instantanée
 $\Rightarrow M$ non atteint
- trajectoire ci-contre simpliste : simulation ou mesure (oscillateur en anneau) nécessaires
- approximation grossière en PC, mais basée sur une trajectoire mieux fondée : de D en A (iso-commutations partielles en entrée et en sortie)



→ raisonnement similaire pour autre porte CMOS

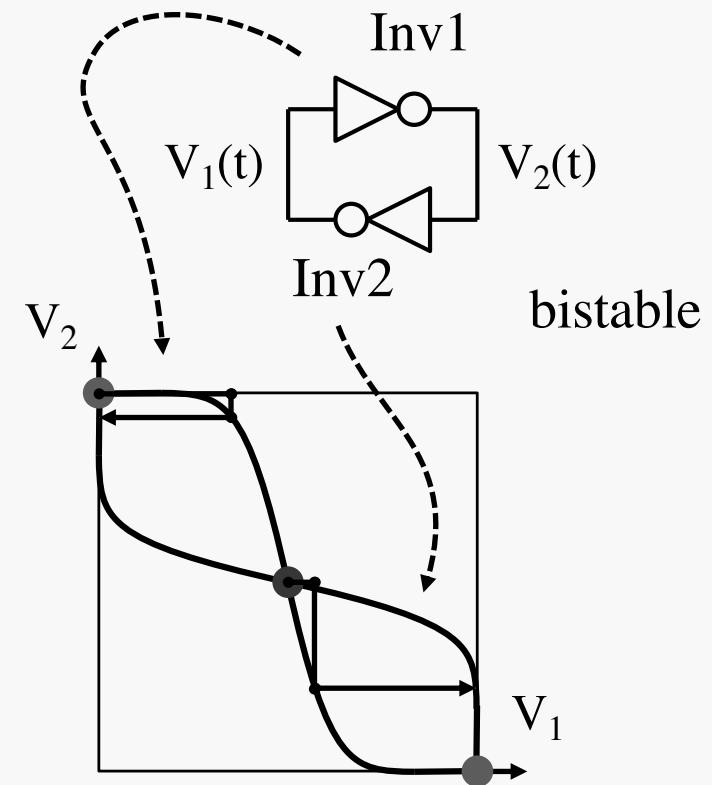
INVERSEUR CMOS ET BOUCLE FERMÉE



Un état d'équilibre stable, mais non binaire
(à mi-chemin entre 0 et 1) : montage prohibé

Pour 3 inverseurs en série, le même état d'équilibre devient instable, provoquant une oscillation.

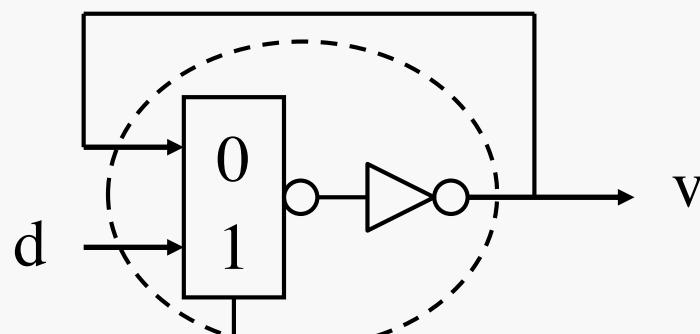
Equilibres d'autant plus (in)stables que les inversions sont raides...



2 états d'équilibre stables, binaires, permettant la *mémorisation d'un bit* ;
+ le même qu'à gauche, devenu instable : dit *métastable*, pour souligner son danger

VERROU (*LATCH*) POUVOIR ÉCRIRE SUR UN BISTABLE

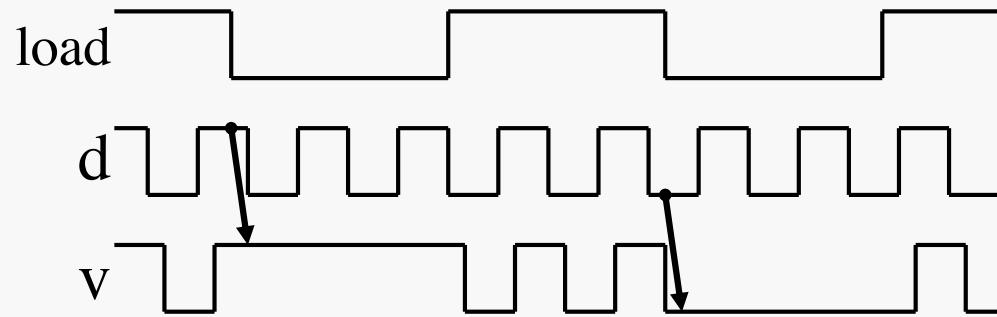
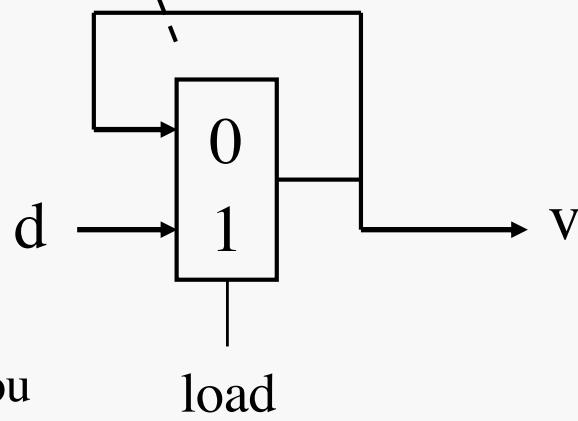
MUX introduit
dans la boucle
du bistable



identiques en
logique com-
plémentaire
CMOS

12 transistors,
complémentation
de *load* incluse

→ notre verrou
désormais

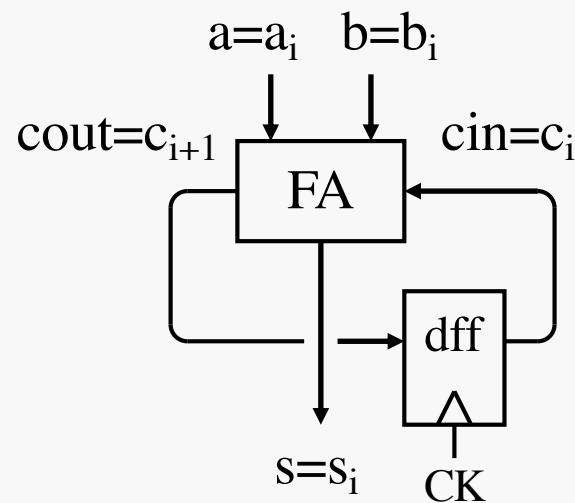


`load=1` : verrou ouvert et chargement de donnée
`load=0` : verrou fermé et blocage de donnée
→ donnée mémorisée sur le montage
bistable obtenu par boucle fermée

La donnée bloquée/mémorisée est celle présente sur d juste pendant le front descendant de *load*
→ *échantillonnage*

Elle doit être stable autour de cet instant sinon sortie binaire aléatoire

BASCULE D : 2 BITS À MÉMORISER

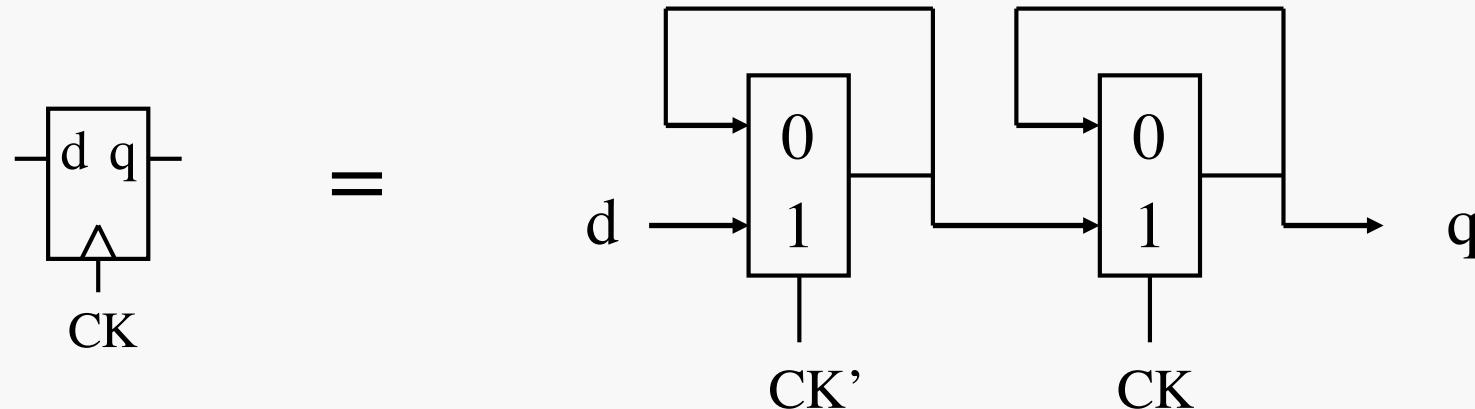


Rappels :

- Juste avant le prochain top d'horloge, la bascule D doit continuer d'imposer c_i sur sa sortie, tout en chargeant c_{i+1}
- besoin de 2 dispositifs de mémorisation binaire indépendants au sein d'une bascule D

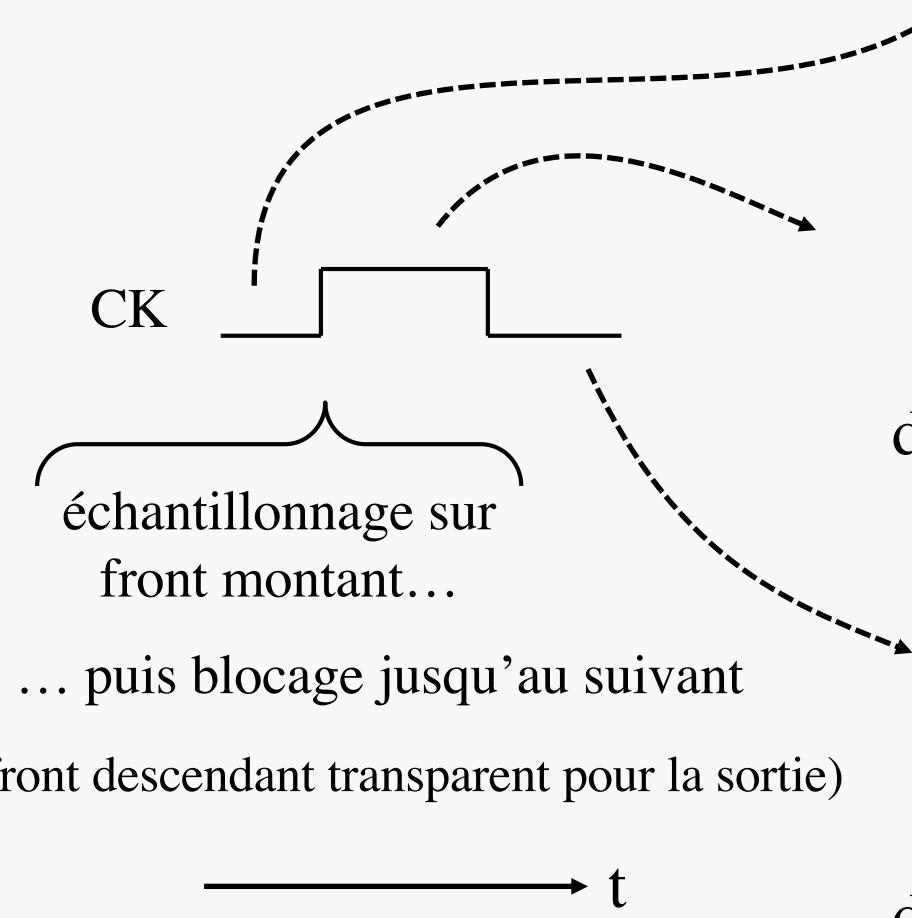
BASCULE D = 2 VERROUS CASCADÉS ET EN OPPOSITION DE PHASE

- l'un passant, l'autre bloquant (mémorisant), alternativement
 - l'un « maître », l'autre « esclave »
- Bascule D sur front montant (*rising edge D flip-flop*) :



BASCULE D :

FONCTIONNEMENT



Jamais de liaison directe entre d et q :
une bascule D réalise un *sas*

