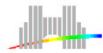


Méthode du *logical effort*

Arnaud Tisserand
Arénaire INRIA LIP

École thématique ARCHI05, 21–25 mars 2005, Autrans



Plan

- 1 Rappels sur les transistors MOS et les portes logiques CMOS
- 2 La méthode du *logical effort* pour une porte
- 3 Modèle de porte utilisé
- 4 Méthode pour plusieurs portes
- 5 Remarques et limites
- 6 Exemples

Introduction

Méthode simple permettant d'évaluer le délai de circuits (CMOS essentiellement) et de calculer une "bonne" taille des différents transistors.

Article de Ivan Sutherland et Robert Sproull dans *Advanced Research in VLSI 1991 : Logical effort : designing for speed on the back of an envelope.*

Logical Effort

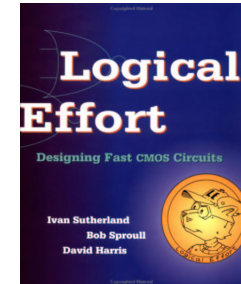
Designing Fast CMOS Circuits

I. Sutherland, B. Sproul and D. Harris

1999

Morgan Kaufmann Publishers

ISBN : 1-55860-557-6



Partie 1

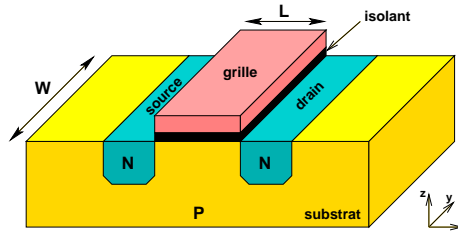
Rappels sur les transistors MOS et les portes logiques CMOS

Structure des transistors MOS

Il existe deux types de **transistors MOS** (*metal oxide semiconductor*) :
 ↪ les transistors **N** et les transistors **P**

Un transistor de type N est composé de :

- substrat (Si) dopé P
- drain et source dopés N
- couche isolante
- grille



Dans une zone dopée N, les porteurs de charge majoritaires sont des électrons (ce sont des trous dans le cas d'une zone dopée P). Pour un transistor de type P on inverse les dopages.

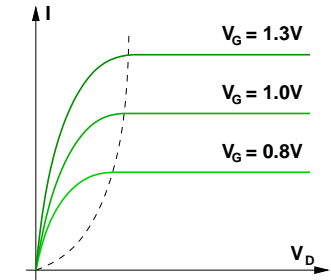
Modèles du transistor

Caractéristique de sortie d'un transistor :

$$I = \begin{cases} 0 & V_G < V_T \\ \beta \left((V_G - V_T)V_D - \frac{V_D^2}{2} \right) & 0 < V_D < V_G - V_T \\ \frac{\beta}{2} (V_G - V_T)^2 & 0 < V_G - V_T < V_D \end{cases}$$

où

$$\beta = C_{\text{techno}} \times \frac{W}{L}$$



Modèle logique :

	Transistor N	Transistor P
G à 0	bloquant	passant
G à 1	passant	bloquant

Un vrai modèle de transistor pour la simulation

Modèle Spice d'un transistor N en technologie 0.18 μm de chez TSMC (106 paramètres, source : www.mosis.org) :

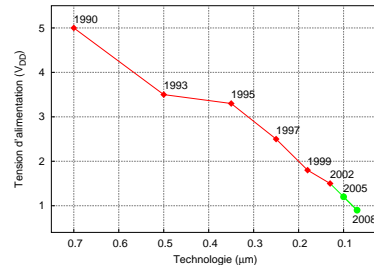
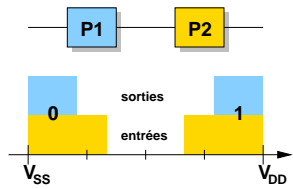
```
.MODEL CMOSN NMOS (
+VERSION = 3.1          TNOM = 27          TOX = 4E-9
+XJ = 1E-7             NCH = 2.3549E17    VTH0 = 0.3618568
+K1 = 0.5821674       K2 = 2.962352E-3    K3 = 1E-3
+K3B = 3.1746246     W0 = 1E-7          NLX = 1.784411E-7
+DVT0W = 0           DVT1W = 0          DVT2W = 0
+DVT0 = 1.0776375    DVT1 = 0.3574214    DVT2 = 0.0606977
+U0 = 257.825805     UA = -1.445098E-9   UB = 2.280431E-18
+UC = 5.132975E-11  VSAT = 1.002296E5   A0 = 1.9572227
+AGS = 0.4279783    B0 = 1.291312E-8   B1 = 6.025607E-7
+KETA = -0.0112723  A1 = 3.225587E-4   A2 = 0.8886833
+RDSW = 105         PRWG = 0.5         PRWB = -0.2
+WR = 1             WINT = 0          LINT = 1.345391E-8
+XL = 0             XV = -1E-8        DWG = -1.012269E-8
+DWB = 8.38965E-9  VOFF = -0.090305   NFACTOR = 2.2452365
+CIT = 0           CDSC = 2.4E-4       CDSCD = 0
+CDSCB = 0         ETA0 = 3.37666E-3  ETAB = 1.141951E-5
+DSUB = 0.017061   PCLM = 0.7636672   PDIBLC1 = 0.1793189
+PDIBLC2 = 2.914511E-3  PDIBLCB = -0.1     DROUT = 0.7552449
)
```

```
+PSCBE1 = 4.184752E10  PSCBE2 = 2.410517E-9  PVAG = 0.0261218
+DELTA = 0.01         RSH = 6.7            MOEMOD = 1
+PRT = 0             UTE = -1.5          KT1 = -0.11
+KT1L = 0            KT2 = 0.022         UA1 = 4.31E-9
+UB1 = -7.61E-18     UC1 = -5.6E-11     AT = 3.3E4
+WL = 0              WLN = 1             WW = 0
+WWN = 1             WWL = 0             LL = 0
+LLN = 1             LW = 0              LWN = 1
+LWL = 0             CAPMOD = 2          XPART = 0.5
+CGDO = 7.51E-10     CGSO = 7.51E-10    CGBO = 1E-12
+CJ = 9.520232E-4    PB = 0.8            MJ = 0.3763097
+CJSW = 2.543816E-10  PBSW = 0.8          MJSW = 0.1472251
+CJSWG = 3.3E-10     PBSWG = 0.8         MJSWG = 0.1472251
+CF = 0              PVTH0 = -6.376792E-4  PRDSW = -0.5939392
+PK2 = 1.01238E-3    WKETA = 4.251478E-3  LKETA = -7.831209E-3
+PU0 = 8.6592416     PUA = 5.50172E-12   PUB = 0
+PVSAT = 1.405109E3  PETA0 = 1.003159E-4  PKETA = 1.134176E-3
)
```

Valeurs logiques

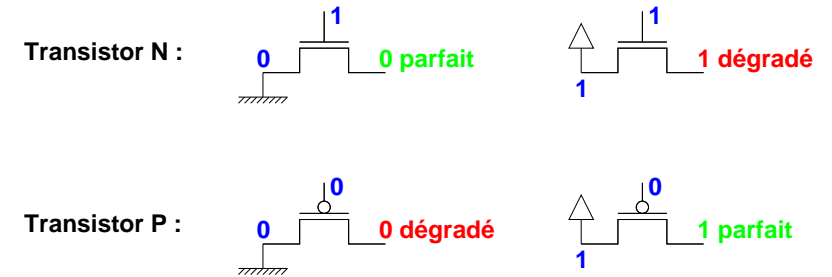
- Le **zéro** logique (0) est codé par la tension de référence (masse) notée V_{SS} ou \perp .
- Le **un** logique (1) est codé par la tension d'alimentation (positive) notée V_{DD} ou $\hat{\uparrow}$.

Il faut adopter un codage des tensions permettant un bon fonctionnement en présence de bruit modéré \rightarrow état = **plage** de tensions :



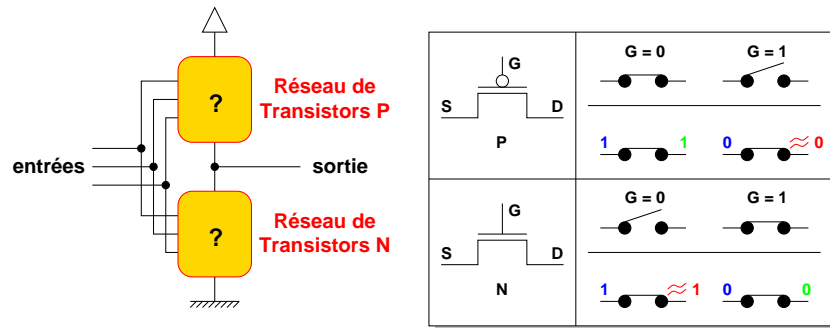
Problème de transmission de certaines valeurs

Du fait des tensions de seuil, les transistors ne laissent pas passer correctement toutes les valeurs :



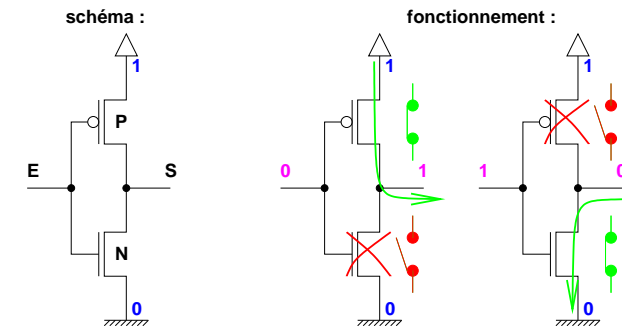
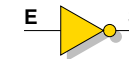
Portes logiques CMOS

Il existe de nombreuses solutions pour faire des portes logiques à partir des transistors. Une des plus utilisée aujourd'hui est la logique **CMOS** (*complementary MOS*). On utilise au mieux les deux types de transistors : N et P.

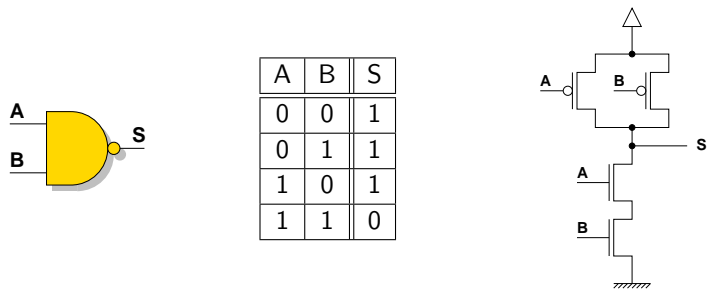


Inverseur

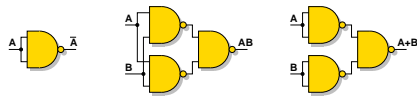
C'est la porte la plus simple : juste deux transistors (1 N et 1 P).



Porte NAND (non-et)

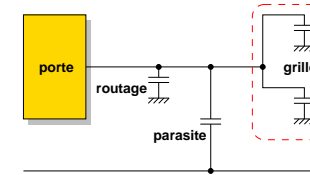


La porte NAND est universelle. On peut faire toutes les autres portes logiques avec des combinaisons de portes NAND (mais c'est pas efficace).



Charge et décharge des nœuds du circuit

La structure du circuit se comporte comme une multitude de capacités qu'il faut charger et décharger. Ces capacités sont partout : grilles des transistors, structures de routage, structures parasites. . .

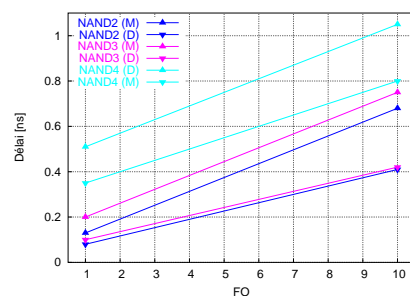
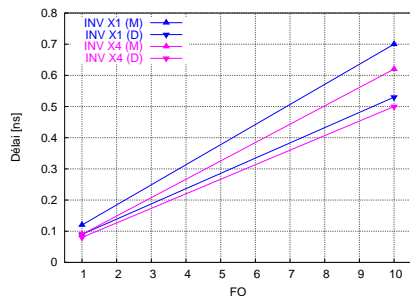
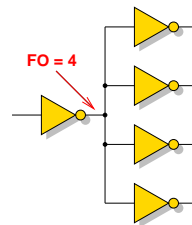


Solutions pour faire des circuits rapides :

- faire des petits circuits (petites capacités)
- utiliser un fort courant pour charger plus vite les capacités

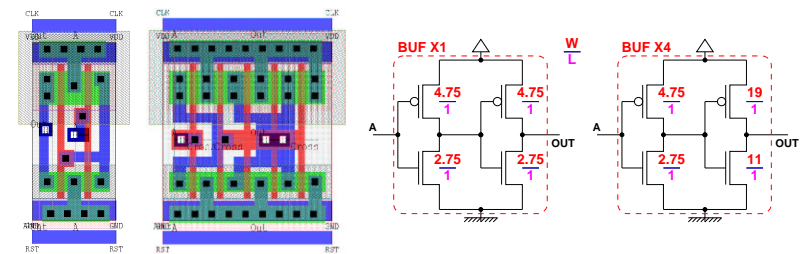
Sortance

Le délai d'une porte (temps pour changer d'état) dépend de sa charge en sortie. La **sortance** (**fan-out**) donne une indication de cette charge. On la mesure comme le nombre d'entrées de portes reliées à la sortie (normalisé en nombre d'entrées d'un inverseur de base en général).



Régénération du signal par *buffer*

La fonction du *buffer* est juste de régénérer le signal sans rien changer à sa valeur ($f(x) = x$). On le réalise en mettant en série deux inverseurs.



caractéristique	BUF X1	BUF X4
taille (h×l) [λ]	53 × 25	53 × 50
capacité A [fF]	5.89	5.89
$T_{0 \rightarrow 1}$	11 + 439 × C_{out}	17 + 132 × C_{out}
$T_{1 \rightarrow 0}$	12 + 318 × C_{out}	21 + 137 × C_{out}

Dimensionner des transistors

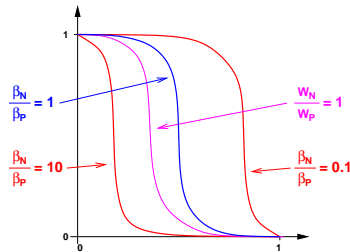
Le gain d'un transistor est composé d'un facteur technologique constant C et d'un facteur géométrique :

$$\beta = C \times \frac{W}{L}$$

La mobilité des électrons et des trous n'est pas la même. Le facteur technologique est différent pour les transistors N et les transistors P (rapport entre 2 et 3 environ).

Dimensionner un inverseur :

- $\frac{\beta_N}{\beta_P} = 1$ ($T_{0 \rightarrow 1} \approx T_{1 \rightarrow 0}$)
- L minimal
- $W = f(FO)$



Mais les choses ne sont pas simples. . .

Sur le chemin critique on va utiliser des transistors permettant de charger rapidement les noeuds en sortie (donc avec un courant plus grand).

Mais ces « gros » transistors ont des grandes grilles, donc des grandes capacités. . .

Question : comment dimensionner l'ensemble des transistors dans une chaîne de portes pour obtenir la plus grande vitesse de fonctionnement ?

La méthode du *logical effort* offre une solution simple (mais approchée) au problème de dimensionner les transistors.

Généralités

Paramètres principaux à prendre en compte pour réaliser une portion de circuit qui respecte des contraintes de vitesse :

- structure en portes
 - ▶ nombre d'étages
 - ▶ type de portes (réécritures logiques)
- taille des transistors

Paramètres qui influencent la vitesse d'une porte :

- charge en sortie
- structure de la porte elle-même
 - ▶ partie utile (taille, nombre et disposition des transistors)
 - ▶ capacités parasites
- technologie (totalement fixée au niveau concepteur)

Partie 2

La méthode du *logical effort* pour une porte

Unités de mesure arbitraires

Afin de simplifier les calculs, la plupart des valeurs seront exprimées avec des unités arbitraires. Par exemple, dans le cas du délai, on a :

$$d = d_{abs} \times \frac{1}{\tau}$$

où

- d est le délai arbitraire [nombre sans unité]
- d_{abs} est le délai absolu [s]
- τ est le délai unitaire [s]

Dans la suite, les délais seront exprimés relativement au délai unitaire d'un inverseur τ_{inv} chargé en sortie par un autre inverseur identique et sans capacité parasite.

Exemple : $\tau_{inv} = 50$ ps pour une technologie $0.6 \mu\text{m}$.

Effort logique g

L'effort logique g donne une indication sur combien de fois une porte produit moins de courant en sortie qu'un inverseur de base (pour la même capacité en entrée).

portes	nombre d'entrées					
	1	2	3	4	5	n
INV	1					
NAND		4/3	5/3	6/3	7/3	$(n+2)/3$
NOR		5/3	7/3	9/3	11/3	$(2n+1)/3$
MUX		2	2	2	2	2
XOR		4	12	32		

Valeurs obtenues pour un facteur $\gamma = 2$ (ratio de la largeur des transistors P sur la largeur des transistors N, $\gamma = W_P/W_N$).

Remarque : n est le nombre d'entrées d'une porte.

Délai d'une porte d

$$d = f + p$$

où

- f est le délai dû à l'effort de la porte (ou étage)
- p est le délai parasite (indépendant de la taille de porte)

avec

$$f = g \times h$$

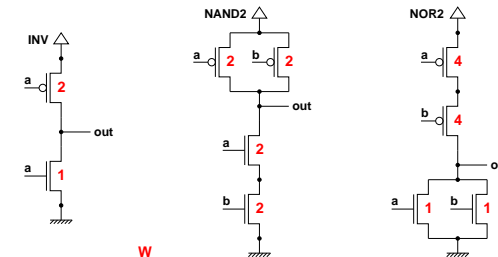
où

- g est l'effort logique de la porte (indépendant de la taille des transistors)
- h est l'effort électrique de la porte (dépendant que des charges en entrée et en sortie)

Le paramètre f "mesure" l'aptitude d'une porte à fournir un certain courant en sortie pour charger la capacité de sortie.

Effort logique g (suite)

Hypothèses : $\gamma = W_P/W_N = 2$ et $\forall t$ un transistor $L_t = L_{min-techno}$.



On souhaite avoir des courants dans les réseaux N et P égaux (temps de montée et de descente égaux). Le courant dans un transistor est proportionnel à W/L .

On a donc pour la NAND2 $g = \frac{4}{3}$ et pour la NOR2 $g = \frac{5}{3}$.

Effort électrique h

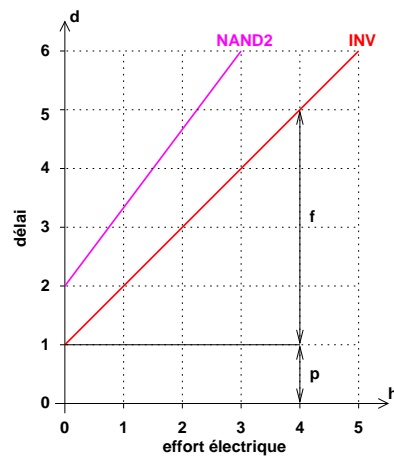
$$h = \frac{C_{out}}{C_{in}}$$

où

- C_{out} est la capacité de la charge en sortie de la porte (sans les capacités parasites internes). Terme au numérateur car plus la charge en sortie est grande plus le délai pour la charger est grand (hypothèse : charge d'un condensateur par un courant constant).
- C_{in} est la capacité d'entrée de la porte (fonction de la taille des transistors reliés aux entrées). Terme au dénominateur car, en supposant les transistors de longueur minimale, le courant dans le transistor dépend linéairement de sa largeur (donc sa capacité pour L fixe).

On parle aussi de *fanout* (mais en terme de capacité et pas de nombre de portes).

Interprétation graphique pour une porte



$$d = g h + p$$

$$f = g h$$

INV $p = 1, g = 1$
 NAND2 $p = 2, g = 4/3$

Délai parasite p

Hypothèse : valeur fixe totalement indépendante de la taille des transistors (faux en toute rigueur mais acceptable dans bon nombre de cas).

Valeurs de p couramment utilisées pour quelques portes de base :

portes	INV	NAND- n	NOR- n	MUX- n	XOR, XNOR
p	p_{inv}	np_{inv}	np_{inv}	$2np_{inv}$	$4p_{inv}$

Dans la suite on va utiliser $p_{inv} = 1$.

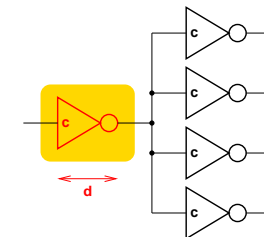
Valeurs de p_{inv} pour différentes technologies¹ :

techno. [μm]	2.0	1.2	1.2	0.8	0.8	0.6	0.6	0.35	0.35
V_{dd} [V]	5.0	5.0	3.3	5.0	3.3	3.3	2.5	3.3	2.5
p_{inv}	0.94	0.91	0.95	0.98	0.95	1.08	1.07	1.06	1.16

¹Source : "Logical Effort : Designing Fast CMOS Circuits", I. Sutherland, B. Sproul and D. Harris, 1999.

Exemple simple

Calculer le délai d de l'inverseur en rouge chargé par 4 inverseurs identiques en sortie :

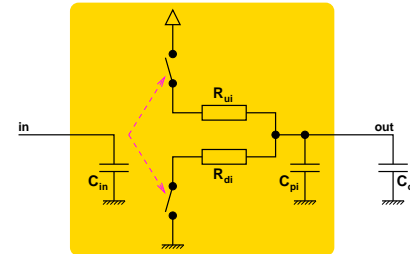


$$C_{out} = 4C_{in} \Rightarrow h = 4$$

$$d = g \times h + p = 1 \times 4 + 1 = 5$$

Modèle commun à toutes les portes

La méthode du *logical effort* est basée sur un modèle où les transistors sont des **résistances** qui chargent et déchargent des **condensateurs**. On suppose toutes les portes basées sur un même modèle à un facteur d'échelle α près.



$$C_{in} = \alpha C_t$$

$$R_i = R_{ui} = R_{di} = \frac{R_t}{\alpha}$$

$$C_{pi} = \alpha C_{pt}$$

Partie 3

Modèle de porte utilisé

« Patrons » pour les résistances et les condensateurs

Les paramètres C_t , R_t et C_{pt} sont fortement dépendant de la technologie :

$$C_t = \kappa_1(W_P L_P + W_N L_N)$$

$$\frac{1}{R_t} = \kappa_2 \frac{\mu_N W_N}{L_N} = \kappa_2 \frac{\mu_P W_P}{L_P}$$

où κ_1 et κ_2 sont des paramètres technologiques et μ la mobilité des porteurs de charge majoritaires.

Modèle final

On a donc

$$\begin{aligned} d_{abs} &= \kappa R_i (C_{out} + C_{pi}) \\ &= \kappa \frac{R_t}{\alpha} C_{in} \frac{C_{out}}{C_{in}} + \kappa \frac{R_t}{\alpha} \alpha C_{pt} \\ &= \kappa R_t C_t \times \frac{C_{out}}{C_{in}} + \kappa R_t C_{pt} \\ &= \tau (gh + p) \end{aligned}$$

où

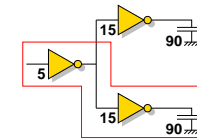
$$\tau = \kappa R_{inv} C_{inv}, \quad g = \frac{R_t C_t}{R_{inv} C_{inv}}, \quad h = \frac{C_{out}}{C_{in}}, \quad p = \frac{R_t C_{pt}}{R_{inv} C_{inv}}$$

Effort de branchement

L'**effort de branchement** b d'une porte est introduit pour tenir compte du fanout interne d'une porte qui n'est pas uniquement sur le chemin traité (non relié à la sortie).

$$b = \frac{C_{sur-chemin} + C_{hors-chemin}}{C_{sur-chemin}} = \frac{C_{total}}{C_{utile}}$$

Pour tout un chemin l'effort de branchement total est : $B = \prod b_i$



$$\begin{aligned} G &= g_1 g_2 = 1 \\ H &= 90/5 = 18 \\ GH &= 18 \\ h_1 &= (15 + 15)/5 = 6 \\ h_2 &= 90/15 = 6 \\ F &= g_1 g_2 h_1 h_2 = 36 \neq GH \end{aligned}$$

Partie 4

Méthode pour plusieurs portes

Composants de l'effort d'un chemin

Dans le cas d'un circuit composé de plusieurs portes, on va procéder pour chaque **chemin** (suite de portes entre une entrée et une sortie).

On a :

- L'effort logique du chemin est :

$$G = \prod g_i$$

- L'effort électrique du chemin est :

$$H = \frac{C_{out}}{C_{in}}$$

L'effort sur tout le chemin est :

$$F = GBH$$

Délai sur le chemin

Le délai sur le chemin est :

$$D = \sum d_i = D_F + P$$

où

- effort sur le chemin

$$D_F = \sum g_i h_i$$

- parasites sur le chemin

$$P = \sum p_i$$

Vers le délai minimal pour N étages

Principe de la méthode : le délai sur tout le chemin est minimal quand tous les étages supportent le même effort \hat{f} .

$$\hat{f} = g_i h_i = F^{1/N}$$

On a alors pour délai minimal

$$\hat{D} = NF^{1/N} + P$$

Notation : un $\hat{}$ pour les paramètres obtenus pour le délai minimal.

Dimensionner les différents étages

On peut alors dimensionner les étages :

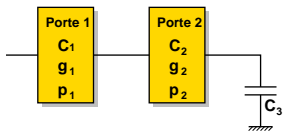
$$\hat{h}_i = \frac{\hat{f}}{g_i}$$

soit en procédant de la fin vers le début du chemin

$$C_{in,i} = \frac{g_i C_{out,i}}{\hat{f}}$$

Justification pour $N = 2$

Comment calculer le délai minimal ?



$$D = (g_1 h_1 + p_1) + (g_2 h_2 + p_2)$$

$$h_1 = \frac{C_2}{C_1}, \quad h_2 = \frac{C_3}{C_2}$$

$$H = h_1 h_2 = \frac{C_3}{C_1}$$

avec $h_2 = H/h_1$, on a :

$$D = (g_1 h_1 + p_1) + (g_2 \frac{H}{h_1} + p_2)$$

minimisation de D :

$$\frac{\partial D}{\partial h_1} = g_1 - \frac{g_2 H}{h_1^2} = 0$$

donc

$$g_1 h_1 = g_2 h_2 = \hat{f} = F^{1/2}$$

Donc tous les étages ont le même effort (pas toujours le même délai).

Nombre d'étages optimal \hat{N}

F	\hat{N}	\hat{D}	\hat{f}
0		1.0	
	1		0...5.8
5.83		6.8	
	2		2.4...4.7
22.3		11.4	
	3		2.8...4.4
82.2		16.0	
	4		3.0...4.2
300		20.7	
	5		3.1...4.1
1090		25.3	
	6		3.2...4.0
3920		29.8	

Pour $p_{inv} = 1$

Exemples de paramètres (calibration)

CMOS 0.6 μm ($V_{dd} = 3.3\text{V}$)², $\tau = 43\text{ps}$:

portes	INV	NAND2	NAND3	NAND4	NOR2	NOR3	NOR4
g	1.00	1.18	1.40	1.66	1.58	2.18	2.81
p	1.08	1.36	2.12	2.39	1.98	3.02	3.95

CMOS 0.11 μm de Fujitsu ($V_{dd} = 1.2\text{V}$)³, $\tau = 6.2\text{ps}$:

portes	INV	NAND2	NOR2	AOI	OAI	XOR2
g	1.00	1.25	1.63	1.78	1.66	2.29
p	1.42	1.64	3.41	3.48	3.56	4.37

²Source : "Logical Effort : Designing Fast CMOS Circuits", I. Sutherland, B. Sproul and D. Harris, 1999.

³Source : "Energy Optimization of High-Performance Circuits", Hoang Q. Dao, Bart R. Zeydel and Vojin G. Oklobdzija, PATMOS 2003.

Partie 5

Remarques et limites

Bilan

Une méthode simple :

- 1 Déterminer l'effort sur le chemin $F = GBH$
- 2 Estimer le nombre minimum d'étages \hat{N}
- 3 Déterminer l'effort de tous les étages $\hat{f} = F^{1/\hat{N}}$
- 4 Estimer le délai minimal $\hat{D} = \hat{N}F^{1/\hat{N}} + P$
- 5 Dimensionner chaque étage en partant de la fin $C_{in,i} = \frac{g_i C_{out,i}}{\hat{f}}$

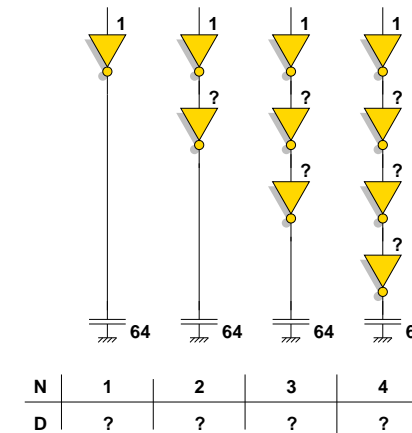
Limitations

- Solution approchée
- Problème de poule et d'oeuf : il faut la structure pour calculer G
- Modèle de délai simpliste :
 - ▶ pas de pente sur les signaux
 - ▶ pas de saturation (courants constants)
 - ▶ pas de délai dans les fils
- Optimisation en vitesse seulement
 - ▶ contrainte produit vitesse \times surface
 - ▶ contrainte de consommation

Partie 6

Exemples

Un classique



Un classique (calculs)

- Solution avec 1 étage

$$f = gh = 1 \times 64 = 64$$

$$d = f + p = 64 + 1 = 65$$

- Solution avec 2 étages

$$F = BGH = 1 \times 1 \times 64 = 64$$

$$F^{1/N} = 64^{1/2} = 8$$

$$D = NF^{1/N} + P = 2 \times 8 + 2 = 18$$

$$\text{Dim} : \frac{64}{8} = 8$$

- Solution avec 3 étages

$$F = BGH = 1 \times 1 \times 64 = 64$$

$$F^{1/N} = 64^{1/3} = 4$$

$$D = NF^{1/N} + P = 3 \times 4 + 3 = 15$$

$$\text{Dim} : \frac{64}{4} = 16, \frac{16}{4} = 4$$

- Solution avec 4 étages

$$F = BGH = 1 \times 1 \times 64 = 64$$

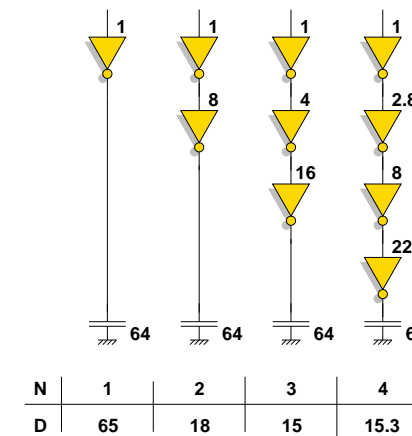
$$F^{1/N} = 64^{1/4} = 2.82$$

$$D = NF^{1/N} + P = 4 \times 2.82 + 4 = 15.31$$

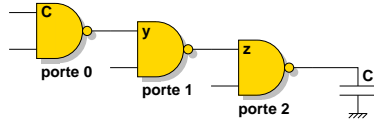
$$\text{Dim} : \frac{64}{2.82} = 22.6, \frac{22.6}{2.82} = 8,$$

$$\frac{8}{2.82} = 2.82$$

Un classique (solution)



Une petite chaîne de 3 portes NAND



Questions : délai minimal ? tailles des transistors pour les portes 1 et 2 ?

$$G = g_0 g_1 g_2 = (4/3)^3 = 2.37, \quad B = 1, \quad H = C/C = 1$$

$$\implies F = 2.37$$

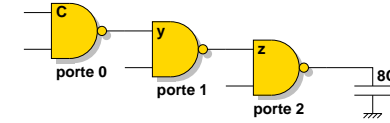
$$\implies \hat{D} = 3 \times 2.37^{1/3} + 3 \times 2 = 10.0$$

$$\hat{f} = 2.37^{1/3} = 4/3$$

$$z = \frac{C \times 4/3}{4/3} = C$$

$$y = \frac{C \times 4/3}{4/3} = C$$

Une petite chaîne de 3 portes NAND, mais. . .



Questions : délai minimal ? tailles des transistors pour les portes 1 et 2 ?

$$G = g_0 g_1 g_2 = (4/3)^3 = 2.37, \quad B = 1, \quad H = 8C/C = 8$$

$$\implies F = 18.96$$

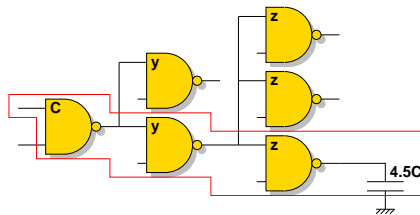
$$\implies \hat{D} = 3 \times 18.96^{1/3} + 3 \times 2 = 14.0$$

$$\hat{f} = 18.96^{1/3} = 8/3$$

$$z = \frac{8C \times 4/3}{8/3} = 4C$$

$$y = \frac{4C \times 4/3}{8/3} = 2C$$

Encore des NAND avec des branchements



$$G = g_0 g_1 g_2 = (4/3)^3 = 2.37, \quad H = 4.5C/C = 4.5$$

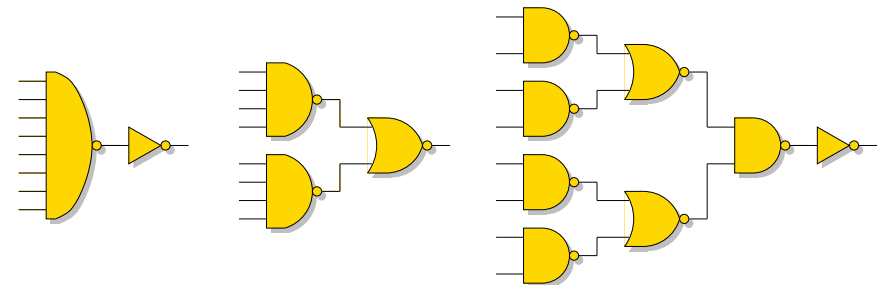
$$b_1 = \frac{y+y}{y} = 2, \quad b_2 = \frac{z+z+z}{z} = 3, \quad B = 6$$

$$\implies F = 64, \quad \hat{D} = 3 \times 64^{1/3} + 3 \times 2 = 18.0$$

$$\hat{f} = 64^{1/3} = 4$$

$$z = \frac{4.5C \times 4/3}{4} = 1.5C, \quad y = \frac{3z \times 4/3}{4} = 1.5C$$

Une porte ET à 8 entrées : différentes solutions



$g=10/3$	$g=1$	$g=2$	$g=5/3$	$g=4/3$	$g=5/3$	$g=4/3$	$g=1$
$p=8$	$p=1$	$p=4$	$p=2$	$p=2$	$p=2$	$p=2$	$p=1$

Une porte ET à 8 entrées : choix de la solution

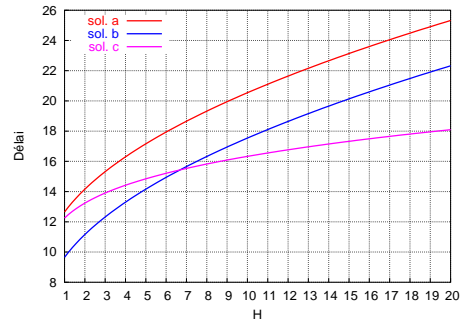
Solution a : $\hat{D} = 2(3.33H)^{1/2} + 9$

Solution b : $\hat{D} = 2(3.33H)^{1/2} + 6$

Solution c : $\hat{D} = 4(2.96H)^{1/4} + 7$

Valeurs de \hat{D} pour deux valeurs de H

	sol. a	sol. b	sol. c
$H = 1$	12.65	9.65	12.25
$H = 12$	21.64	18.64	16.77



Lecture

Bon livre sur les circuits intégrés numériques en général (avec une introduction au *logical effort*).

CMOS VLSI Design

A Circuits and Systems Perspective

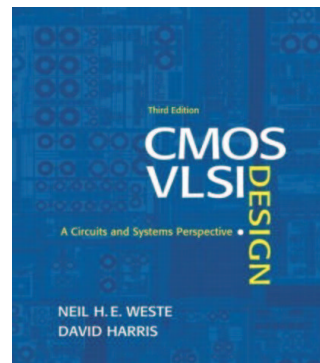
N. Weste and D. Harris

3rd Edition

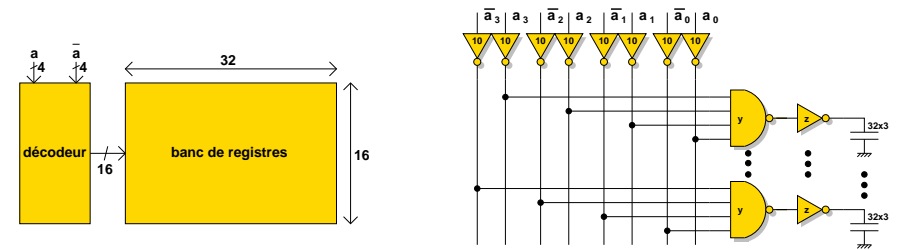
2004

Addison Wesley

ISBN : 0-321-14901-7



Un décodeur 4→16



solution	N	G	P	D
NAND4, INV	2	2	5	29.8
INV, NAND4, INV	3	2	6	22.1
INV, NAND4, INV, INV	4	2	7	21.1
NAND2, INV, NAND2, INV	4	16/9	6	19.7
INV, NAND2, INV, NAND2, INV	5	16/9	7	20.4
INV, NAND2, INV, NAND2, INV, INV	6	16/9	8	21.6
INV, NAND2, INV, NAND2, INV, INV, INV	7	16/9	9	23.1

Fin

Questions ?

Pour me contacter :

- arnaud.tisserand@ens-lyon.fr
- <http://perso.ens-lyon.fr/arnaud.tisserand/>
- Laboratoire LIP. ENS Lyon. 46 allée d'Italie. F-69364 Lyon cedex 07.

Merci.