

Aspects techniques des circuits numériques

Introduction

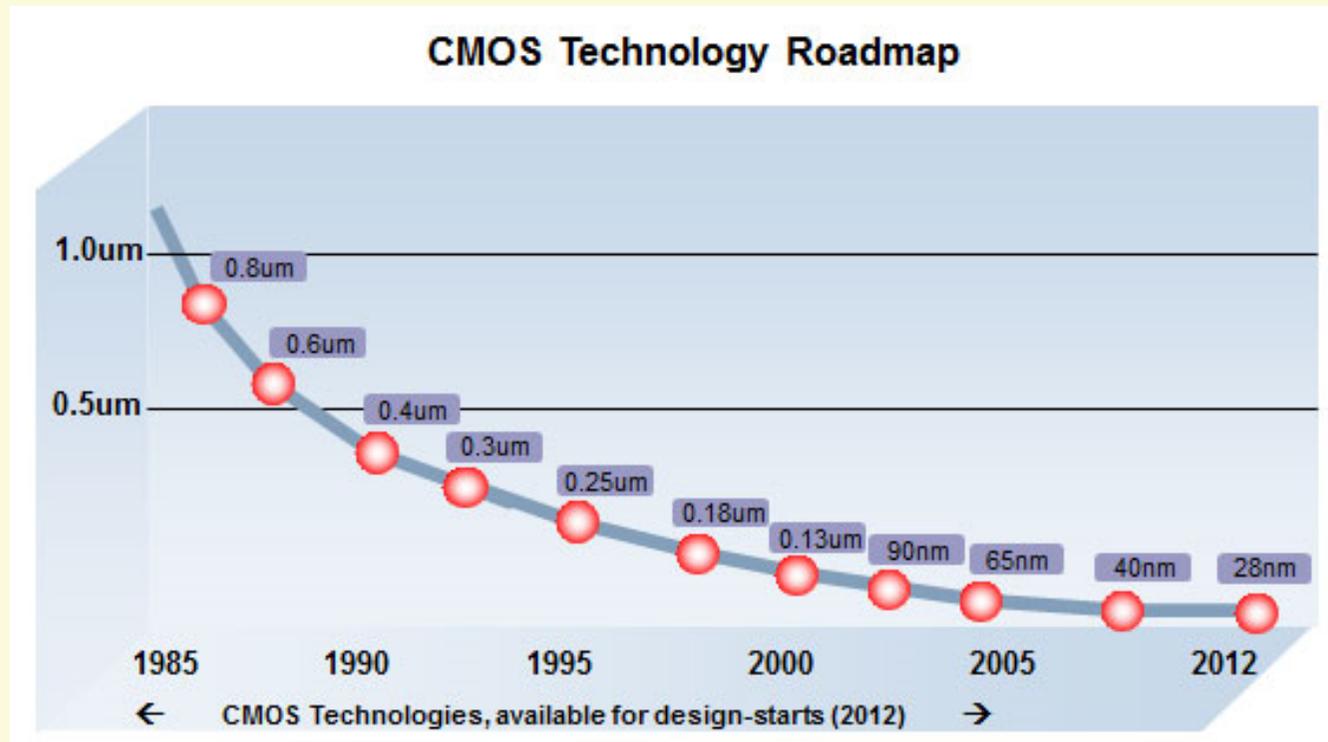
- Nous avons jusqu'à présent considéré les circuits logiques comme étant parfaits
- Nous allons voir quelques caractéristiques électriques des circuits logiques
 - Tension, niveaux logiques, courant, ...
 - Temps de propagation
 - Caractéristiques dépendent de la technologie

Technologies : Les familles

- La technologie influence les paramètres électriques
 - Tension, niveaux logiques, courant, vitesse,...
- Famille **TTL** – Transistor-Transistor Logic
 - Transistors bipolaires, obsolète, sauf très hautes fréquences
- Famille **CMOS** - Complementary metal oxide semi-conductor
 - Transistors MOS, **technologie utilisée actuellement**
- Famille **ECL** – Emitter Coupled Logic
 - Transistors bipolaires, très rapide mais forte consommation

Technologie: évolution CMOS

- Roadmap Toshiba,

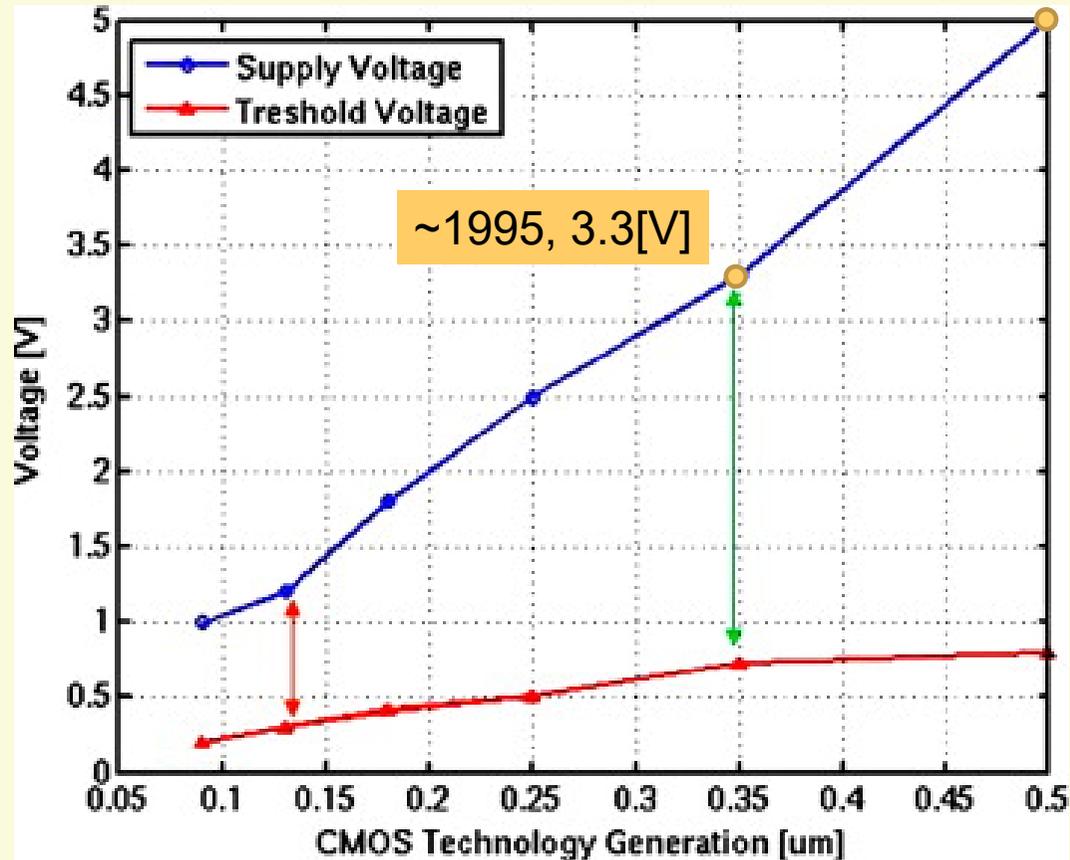


<http://www.toshiba-components.com/asic/Technology.html>

Technologies : évolution

- De 1960 jusqu'en 1995 : $>0.50 \mu\text{m}$
 - Tension fixe à 5 volt pour tous les circuits numériques
- Dès 1990-1995 :
 - évolution de la technologie $< 0.50 \mu\text{m} \Rightarrow$
diminution de la tension d'alimentation en dessous de 5 V
- En 1995 :
 - Pentium Pro X86, 150Mhz, 5,5 millions transistors, 16 bits, $0.50 \mu\text{m}$
 - tension alimentation du cœur: 3,3 V
- En 2014 FPGA Xilinx Ultrascale: 16 nm FinFET, 3D IC à 0.72 V
- En 2017 FPGA Altera Stratix 10: 14 nm Tri-Gate process, 0.8 V
- En 2019 FPGA Intel-Altera Flacon Mesa:10 nm FinFET
- En 2019 FPGA Xilinx Everest: 7 nm (MCM: Multi-Chip-Module)

Technologie: évolution tension d'alimentation



Jusqu'en ~1990: 5 [V]

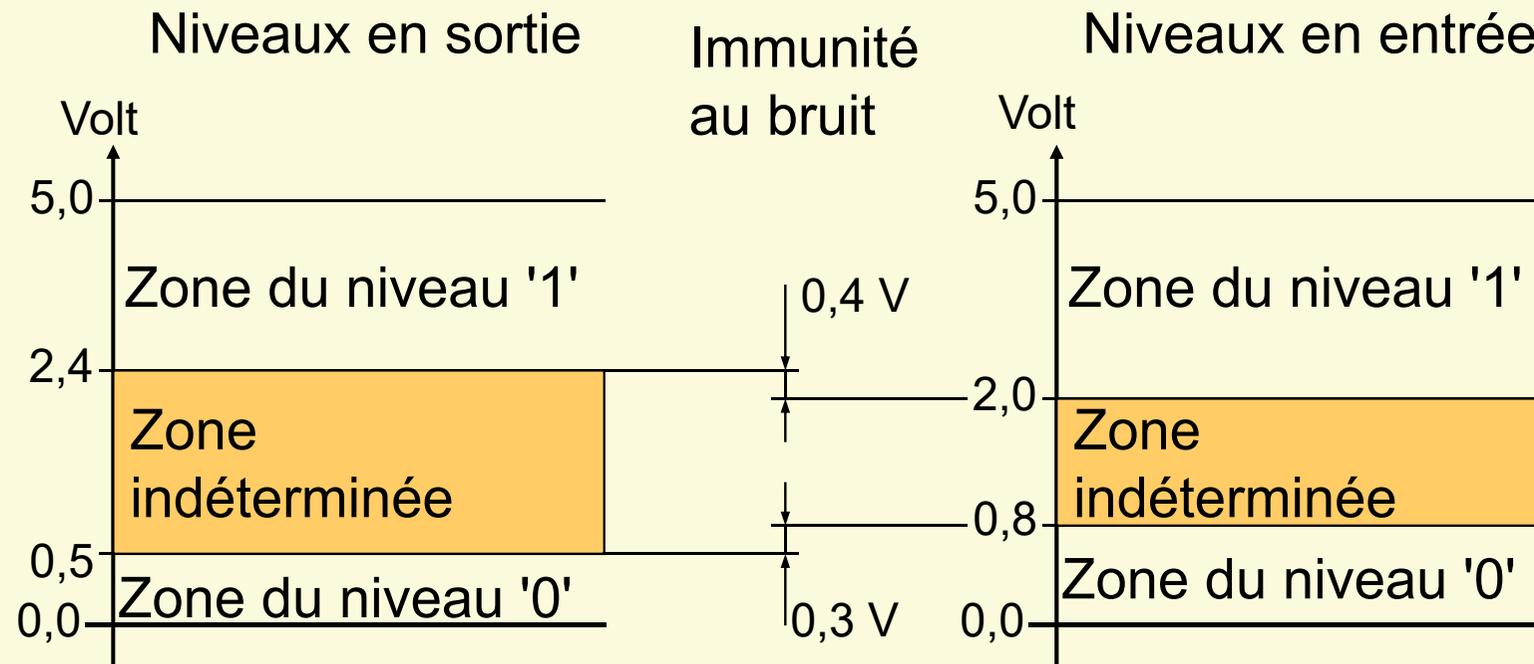
~1995, 3.3[V]

Threshold: tension de seuil
(tension min pour faire conduire
le transistor)

Source: <http://www.sciencedirect.com/science/article/pii/S0924424711002986>

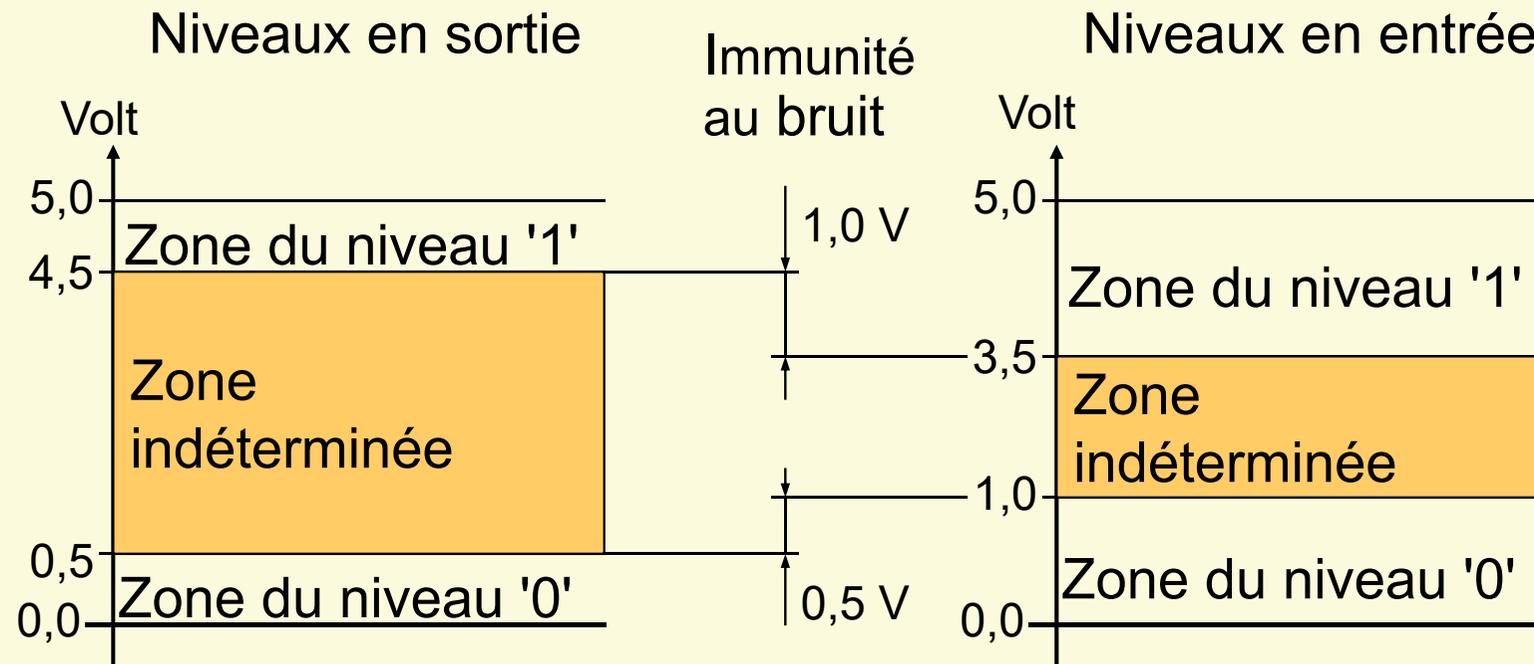
Niveaux logiques et immunité au bruit

- Exemple de la technologie TTL :

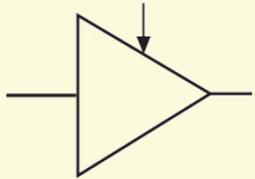


Niveaux logiques et immunité au bruit

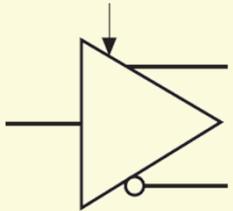
- Technologie CMOS :



Les familles d'entrées / sorties (IOs)



- **IOs simples (asymétriques, single-ended)**
 - LVCMOS, LVTTTL, HSTL, SSTL, AGP, GTL, PCI, ...

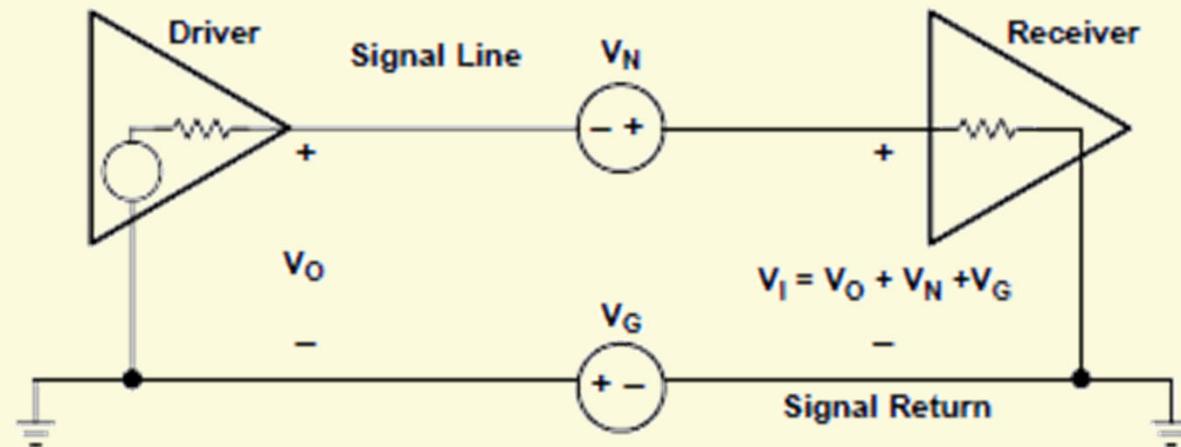


- **IOs différentielles (symétriques)**
 - LVDS, RSDS, TMDS, Differential HSTL and SSTL, ...

Les familles d'entrées / sorties (IOs)

- Standards pour spécifier les types de IOs
 - ANSI, American National Standards Institute
 - nombreux standards IOs défini par JEDEC
 - Joint Electronic Device Engineering Council
 - JEDEC est accrédité par ANSI
 - JEDEC et standards industriels IC ont défini des tensions d'alimentations pour les IOs:
 - $3.3V \pm 0.3V$, $2.5V \pm 0.2V$, $1.8 V \pm 0.15V$, ...

I/Os simples (asymétriques)



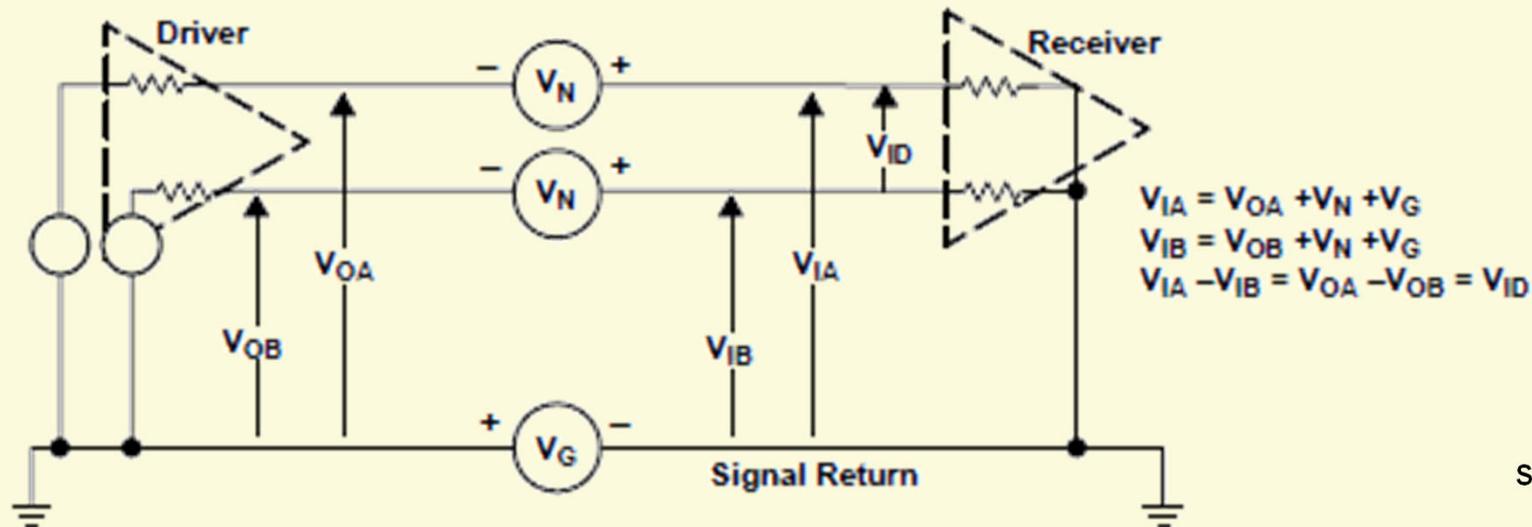
source TI

- Signal référencé par rapport à la masse (un fil)
- Etat logique (0/1) dépendant du niveau du signal
- Nécessite dans certains cas des terminaisons (résistances) séries ou parallèles.

LVTTL /LVCMOS (IOs simples)

- 3.3-V Low Voltage TTL ou Low voltage CMOS
 - Alimentation du circuit 3.3V
 - Plage du signal : $-0.5 \text{ V} \leq V_I \leq 3.8 \text{ V}$
 - Niveaux identiques pour TTL & CMOS !
 - Niveaux sortie : bas < 0.4V, haut > 2.4V
 - Niveaux entrée: bas < 0.8V, haut > 2V
 - Limitation en fréquence : 200 à 600 MHz
- Autres voltages (LVCMOS): 2.5V, 1.8V, 1.5V

IOs différentielles (symétriques)



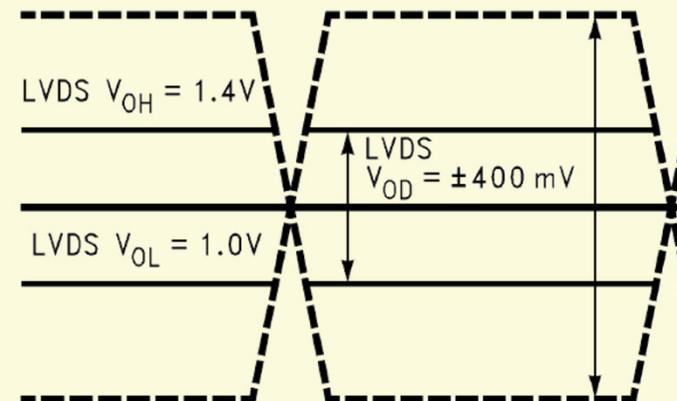
source TI

- Signal différentiel non référencé par rapport à la masse (2 fils)
=> excellente immunité au bruit et aux perturbations
- Etat logique (0/1) dépendant de la polarité du signal
- Nécessite une terminaison parallèle.

LVDS (CMOS)

- 3.3V, 2.5V Low Voltage Differential Signaling

- Alimentation du circuit
3.3V, 2.5 V ou inférieur



- Niveaux sortie : bas 1V, haut 1.4V
- Niveaux entrée: $\pm 400 mV$ (seuil $\pm 100mV$)
- Limitation en fréquence : $> 1 GHz$

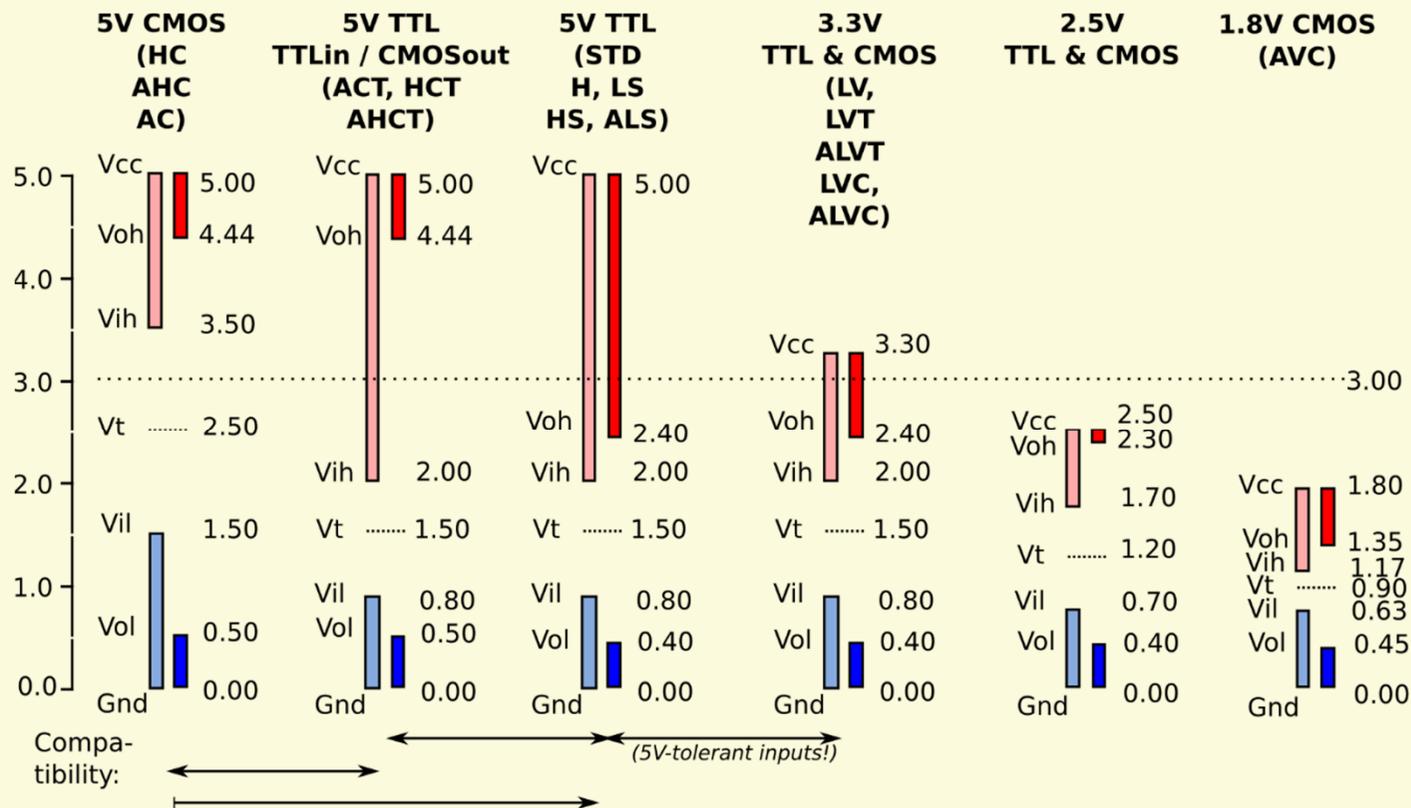
Quelques standards CMOS et ECL

- LVTTL/LVCMOS à 3.3V (Low voltage TTL)
 - simples (applications générales peu performantes)
- HSTL de 1.8 à 1.5V (High Speed Transceiver Logic)
 - simples ou différentielles (application : bus, mémoires ...)
- SSTL de 3.3V à 1.5V (Stub Series Terminated Logic)
 - simples (application : mémoires DDR ...)
- CML et PCML (Current mode logic et Pseudo CML)
 - Très hauts débits : > 5Gb/s
- 3.3V à 1.4V LVPECL version optimisé de PECL

- Evolution régulière pour s'adapter aux nouvelles technologies et débits de transfert

Niveaux logiques et immunité au bruit

- Comparaison des niveaux de tension de différents standards logiques



Data source: EETimes, A brief recap of popular logic standards (Mark Pearson, Maxim).

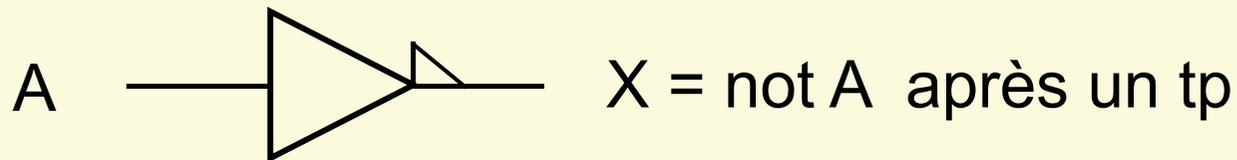
Aléas

- Etat transitoire ne répondant pas aux équations logiques du circuit
- Dû aux temps de propagation
- Se produit lors des commutations
- Sans conséquences dans un circuit purement combinatoire
- Peut provoquer un mauvais fonctionnement dans un circuit séquentiel

Temps de propagation ...

Un signal subit un retard lorsqu'il passe dans une porte logique.

Exemple avec une porte NON :



Le signal X a un retard (t_p) sur le signal A

.... temps de propagation

Violation des postulats de l'algèbre de Boole :

en ❶

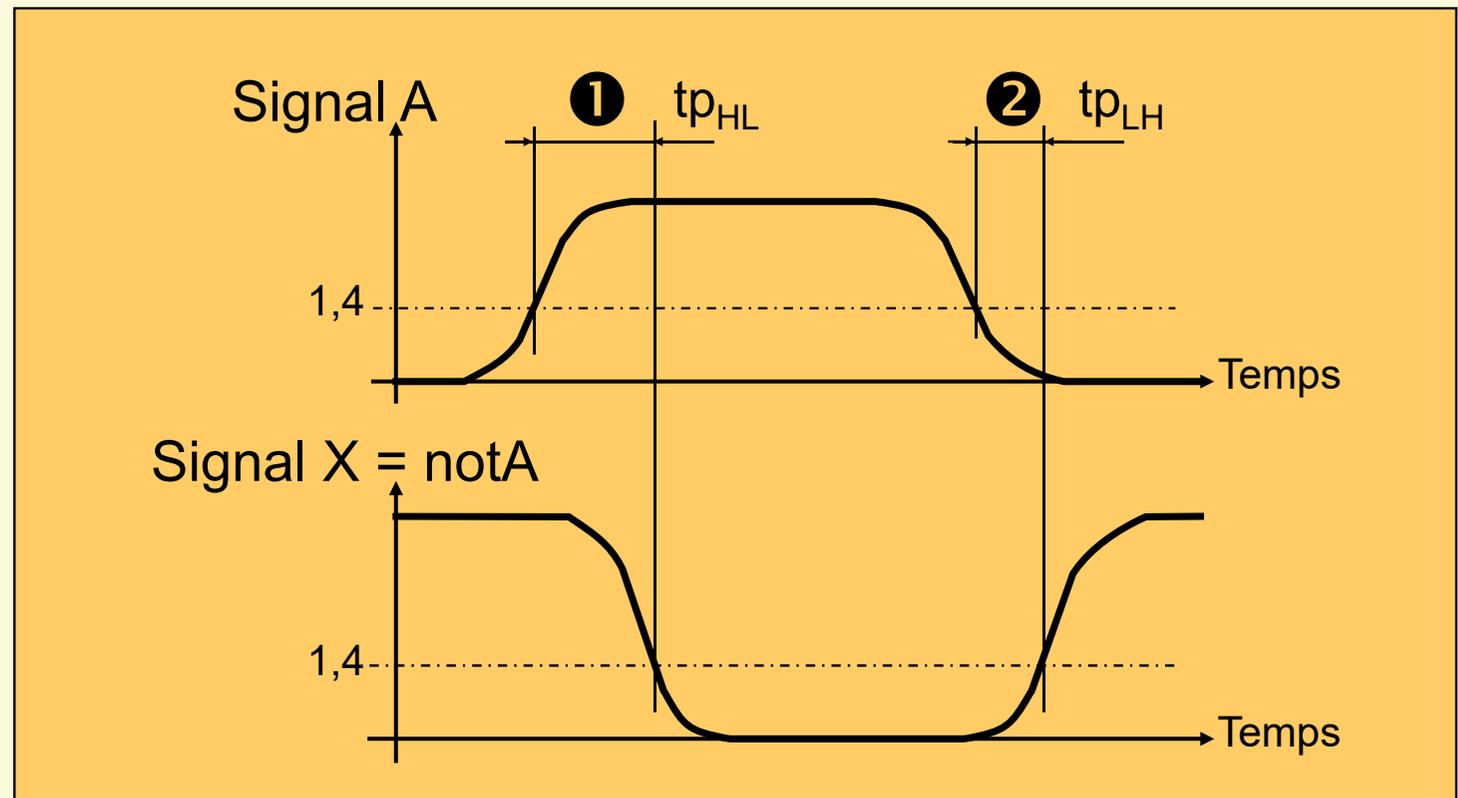
$$A \cdot \bar{A} = '0'$$

pas respecté

en ❷

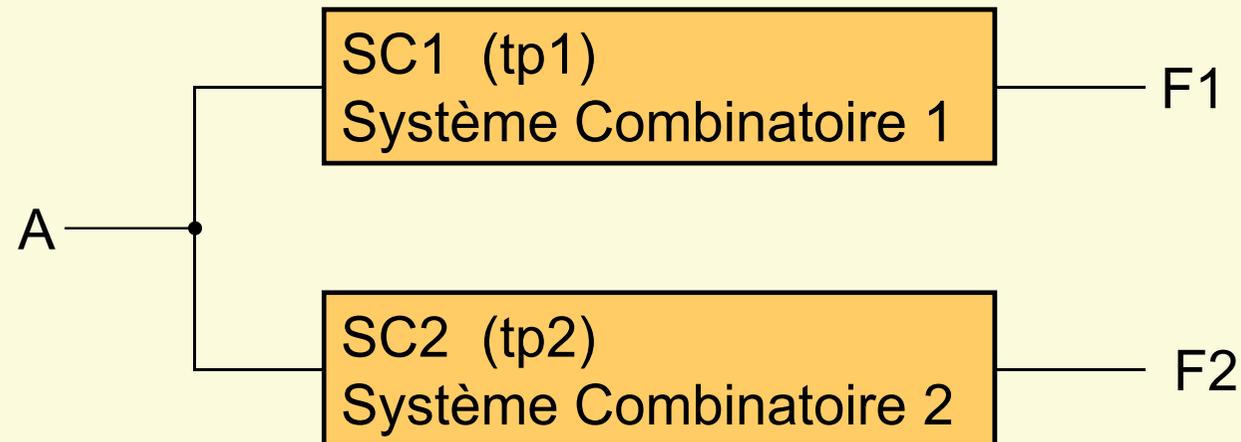
$$A + \bar{A} = '1'$$

pas respecté



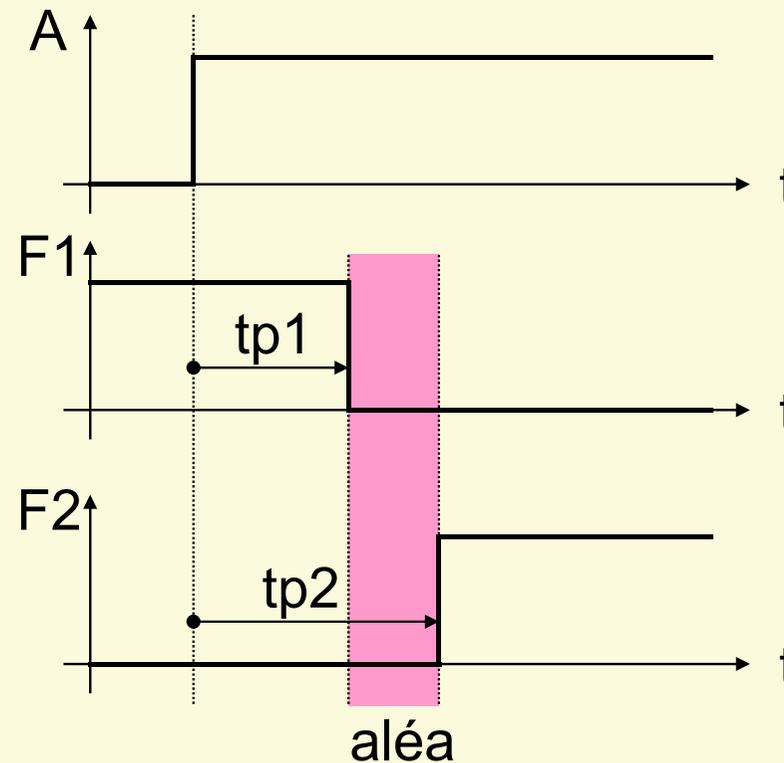
Aléas de propagation ...

- 2 fonctions dépendent d'un même signal
- Signal A change => les fonctions changent, mais pas simultanément !



... aléas de propagation

- Si $A = '0'$:
F1, F2 = "1 0"
- Si $A = '1'$:
F1, F2 = "0 1"
- Transition de A
aléas F1, F2 = "0 0"



Exercices

- Série " Aspects techniques des circuits logiques"
 - Exercices n° 58 et 59

Porte à collecteur/drain ouvert

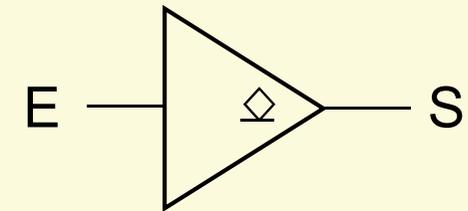
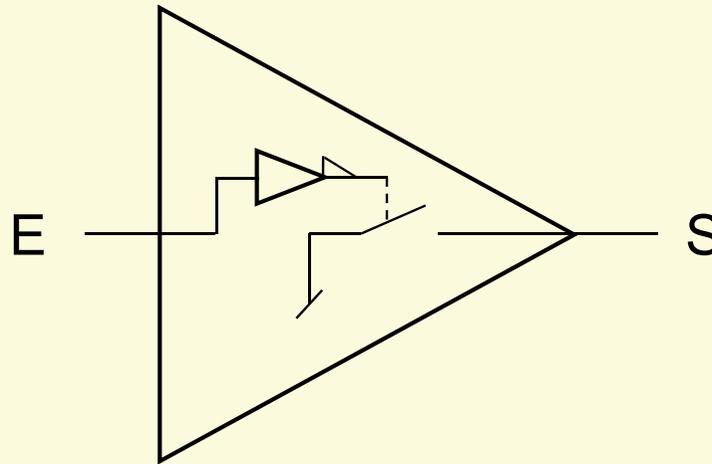
- But : avoir une sortie présentant un état dit « à haute impédance » (Z)
- Ces sorties peuvent être connectées ensemble sans risque de court-circuit
- Les sorties à collecteur ouvert permettent de réaliser un **ET câblé**.
En logique négative, il s'agit d'un **OU câblé**.

Porte à collecteur/drain ouvert

- Une sortie à collecteur/drain ouvert fournit deux états de sortie : '0' ou 'Z'

Table de vérité

E	S
'0'	'0'
'1'	'Z'



Symbole IEEE/CEI

Etat '1' et résistance de polarisation

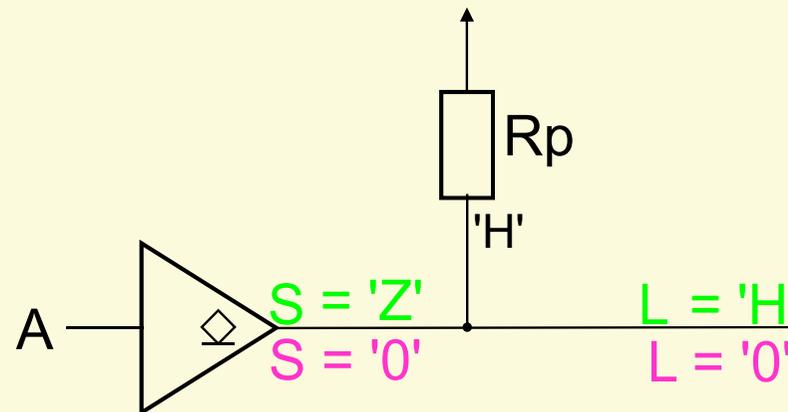
- L'état '1' n'est pas imposé par la sortie à collecteur ouvert
- Il requiert une résistance de polarisation (Pull-Up)

Table de vérité

A	S	L
'0'	'0'	'0'
'1'	'Z'	'H'

S : état de la sortie si elle n'est pas connectée à la ligne

L : état de la ligne avec la sortie connectée à la ligne

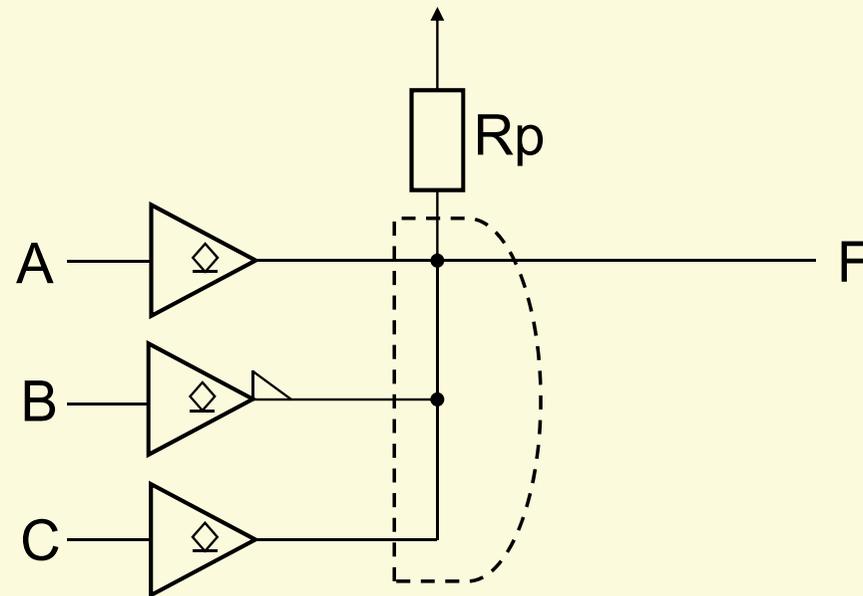


L'état 'H' est un état '1' faible

Réalisation d'une fonction câblée

Equation logique :

$$F = A \cdot \overline{B} \cdot C$$



La fonction réalisée est un ET câblé

Réalisation d'une fonction câblée

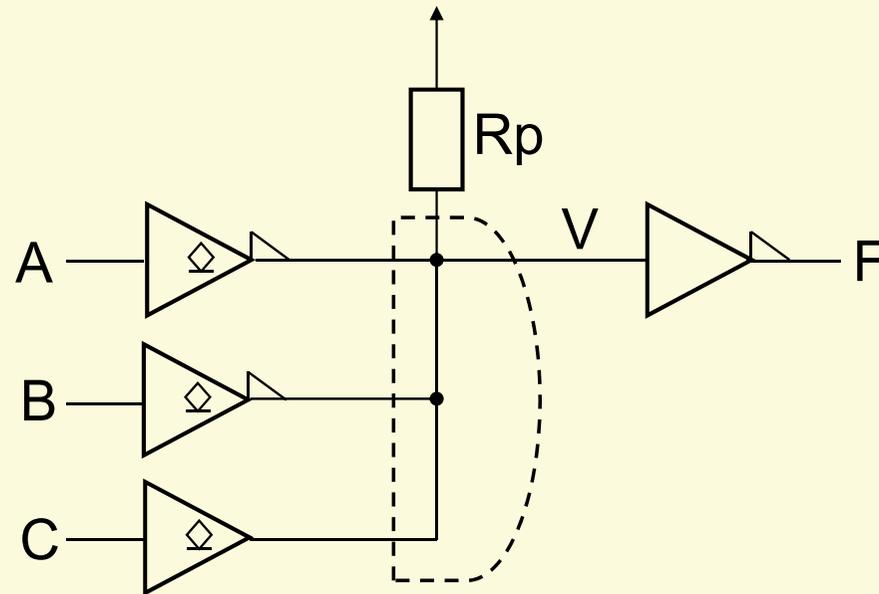
Equation logique :

$$F = A + B + /C$$

$$V = /A \cdot /B \cdot C$$

$$F = /V = / (/A \cdot /B \cdot C)$$

$$F = A + B + /C$$



La fonction réalisée est un OU câblé par l'inversion appliquée en sortie selon le théorème de De Morgan

Réalisation d'une fonction câblée

Equations logiques :

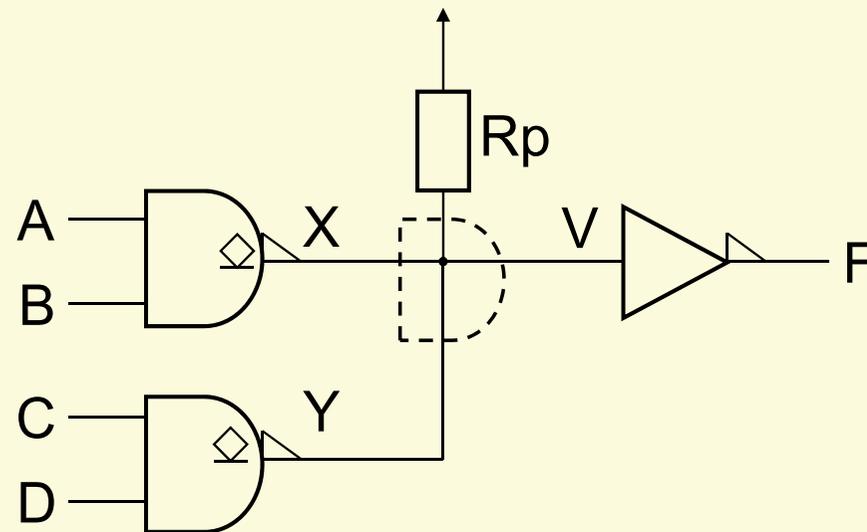
$$X = \overline{B \cdot A}$$

$$Y = \overline{D \cdot C}$$

$$V = X \cdot Y$$

$$F = \overline{V} = \overline{X} + \overline{Y}$$

$$F = B \cdot A + D \cdot C$$



La fonction réalisée est un **OU** câblé car le signal V représente F en logique négative

Exercice collecteur/drain -ouvert

- Réaliser les fonctions suivantes avec uniquement des buffers à collecteur ouvert inverseuses ou non-inverseuses.
 - $F_a = A \cdot B \cdot \overline{C}$
 - $F_b = A + \overline{B} + C$

Porte à trois états

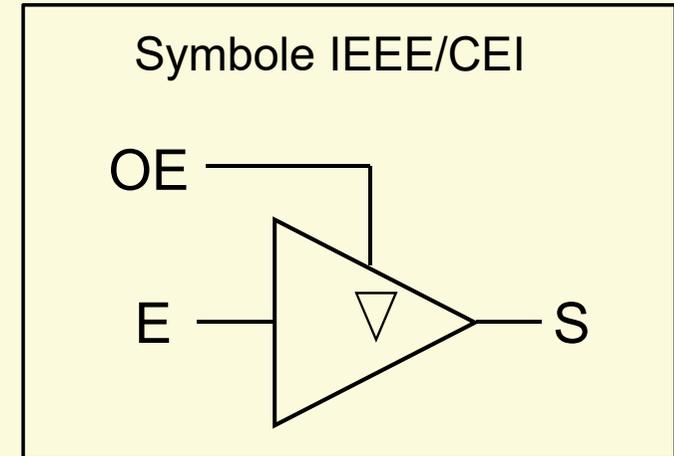
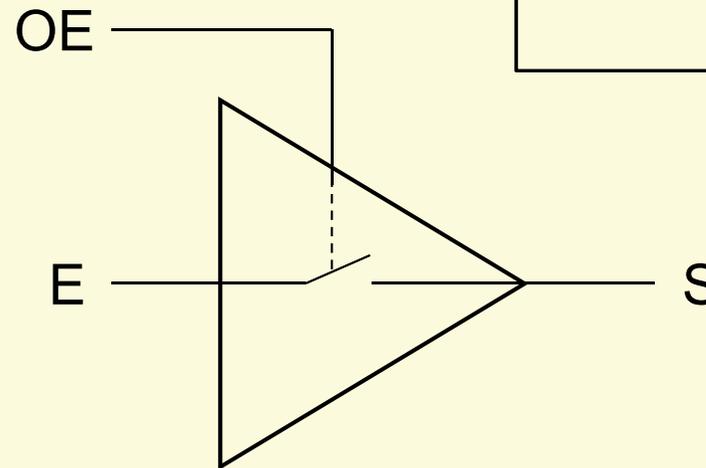
- But : avoir une sortie présentant un état dit « à haute impédance » ('Z')
- Les sorties des portes à trois états sont connectables ensemble pour autant qu'une seule soit active simultanément
- Les portes à 3 états sont utilisées pour créer des structures de BUS
On en trouve dans tous les ordinateurs

Porte trois états, principe

Une porte trois états fournit trois états de sortie : '0', '1' ou 'Z'

Table de vérité

OE	E	S
'0'	'0'	'Z'
'0'	'1'	'Z'
'1'	'0'	'0'
'1'	'1'	'1'



Porte trois états : entity

- entity : les entrées/sorties du module

```
library IEEE;
use IEEE.Std_Logic_1164.all;

entity Trois_Etats is
  port( OE_i : in Std_Logic;
        -- Active la porte 3 etats
        E_i  : in Std_Logic;
        -- Donnee d'entree
        S_o  : out Std_Logic
        -- Sortie porte 3 etats
        );
end Trois_Etats;
```

Porte trois états : architecture

- Description de la fonctionnalité du bloc

```
architecture Flot_Don of Trois_Etats is
begin

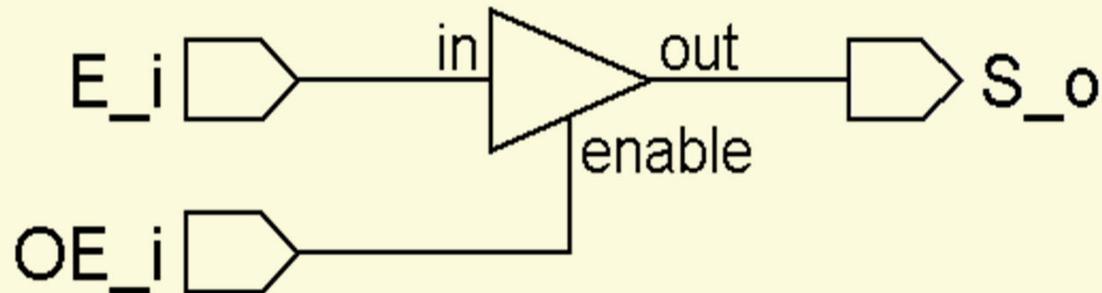
    S_o <= E_i when OE_i = '1' else
        'Z' when OE_i = '0' else
        'X'; -- pour la simulation

end Flot_Don;
```

Recommandé : utilisation d'instructions concurrentes

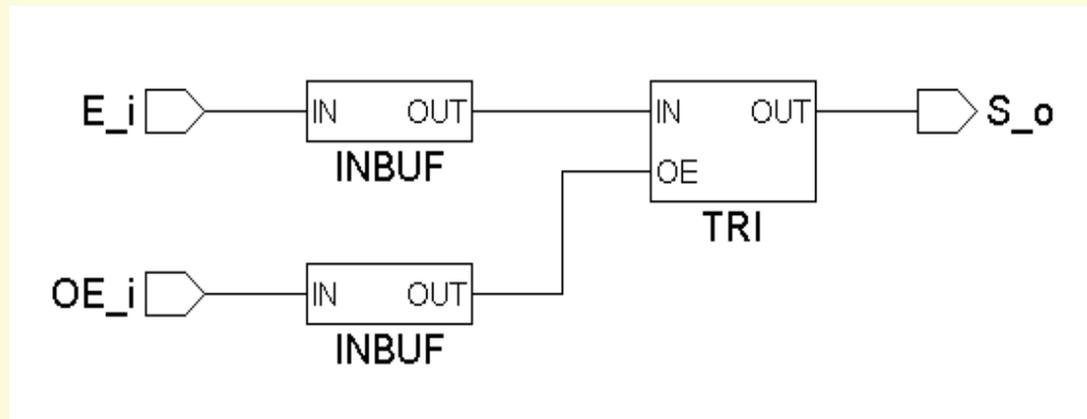
Porte trois états, vue RTL

- Interprétation «fonctionnelle» du synthétiseur



Porte trois états, vue Technology

- Traduction en «logique» du synthétiseur



Porte trois états et BUS

- Les portes trois états admettent l'interconnexion des sorties
- Comment assurer aucuns conflits sur bus:
 - impératif: une seule porte 3 états activée simultanément
 - système à processeur: résolu par le plan d'adressage
- Dans le cas du VHDL:
 - Comment se règle l'identification de la valeur du signal?
 - Comment se comporte le type Std_Logic dans ce cas ?

Type Std_uLogic

- Le type de base std_ulogic comporte 9 états

```
type std_ulogic is ('U',--état non initialisé  
                    'X',--état inconnu fort  
                    '0',--état logique 0 fort  
                    '1',--état logique 1 fort  
                    'Z',--état haute impédance  
                    'W',--état inconnu faible  
                    'L',--état logique 0 faible  
                    'H',--état logique 1 faible  
                    '-' --état indifférent  
                    );
```

Type Std_Logic

- std_logic est un sous-type de std_ulogic :

```
subtype std_logic is resolved std_ulogic;
```

- resolved est une fonction retournant le résultat de l'interconnection de plusieurs signaux de type std_Logic

```
function resolved (s: std_ulogic) return std_ulogic;
```

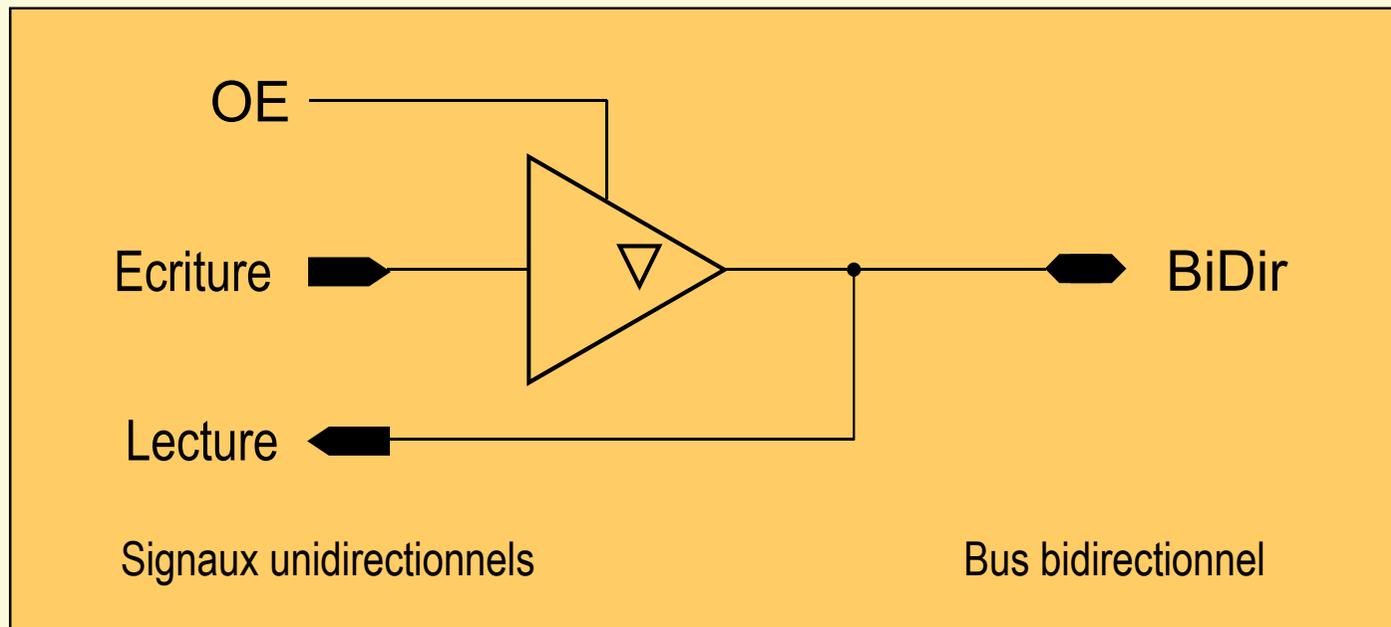
Table de résolution

- La valeur affectée au point d'interconnexion de deux sorties est régie par la table ci-après :

```
constant resolution_table : stdlogic_table := (
-- -----
-- |  U   X   0   1   Z   W   L   H   -   |  |
-- -----
( 'U', 'U', 'U', 'U', 'U', 'U', 'U', 'U', 'U' ), -- | U |
( 'U', 'X', 'X', 'X', 'X', 'X', 'X', 'X', 'X' ), -- | X |
( 'U', 'X', '0', 'X', '0', '0', '0', '0', 'X' ), -- | 0 |
( 'U', 'X', 'X', '1', '1', '1', '1', '1', 'X' ), -- | 1 |
( 'U', 'X', '0', '1', 'Z', 'W', 'L', 'H', 'X' ), -- | Z |
( 'U', 'X', '0', '1', 'W', 'W', 'W', 'W', 'X' ), -- | W |
( 'U', 'X', '0', '1', 'L', 'W', 'L', 'W', 'X' ), -- | L |
( 'U', 'X', '0', '1', 'H', 'W', 'W', 'H', 'X' ), -- | H |
( 'U', 'X', 'X', 'X', 'X', 'X', 'X', 'X', 'X' ) -- | - |
);
```

Bus bidirectionnel et porte trois états

- L'utilisation d'un *bus bidirectionnel* implique d'avoir accès à une ligne (un signal) en entrée et en sortie



Bus bidirectionnel : entity

- entity : les entrées/sorties

```
library IEEE;
use IEEE.Std_Logic_1164.all;

entity Bus_BiDir is
  port( OE_i      : in  Std_Logic;
        -- Active la porte 3 etats
        Ecriture_i : in  Std_Logic;
        -- Donnee a envoyer sur le bus
        Lecture_o  : out Std_Logic;
        -- Donnee lue sur le bus
        BiDir_io  : inout Std_Logic );
        -- Bus bidirectionnel
end Bus_BiDir;
```

Bus bidirectionnel : architecture

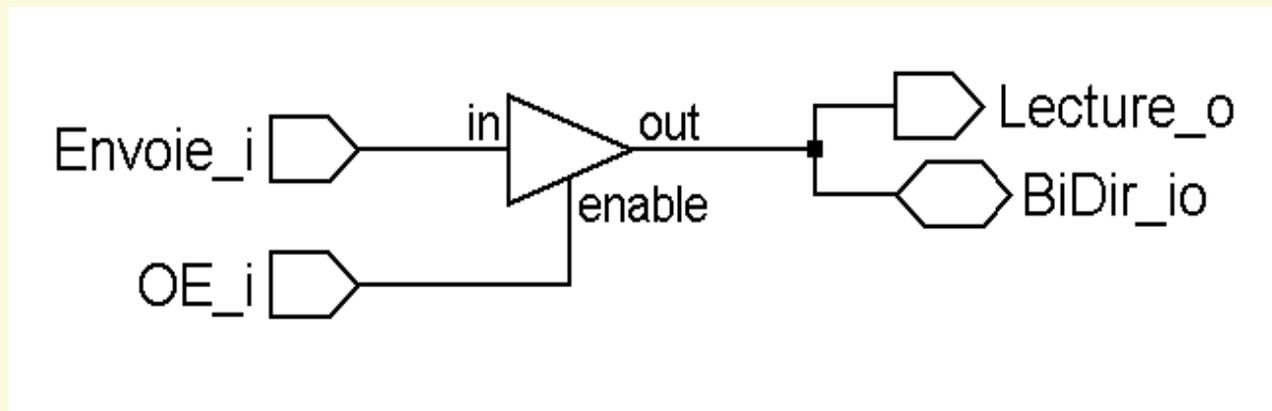
- Description de la fonctionnalité

```
architecture Flot_Don of Bus_BiDir is  
begin  
  
    BiDir_io  <= Ecriture_i when OE_i = '1' else  
                'Z' when OE_i = '0' else  
                'X'; --pour la simulation  
  
    Lecture_o <= to_x01(BiDir_io);  
  
end Flot_Don;
```

Recommandé : utilisation d'instructions concurrentes

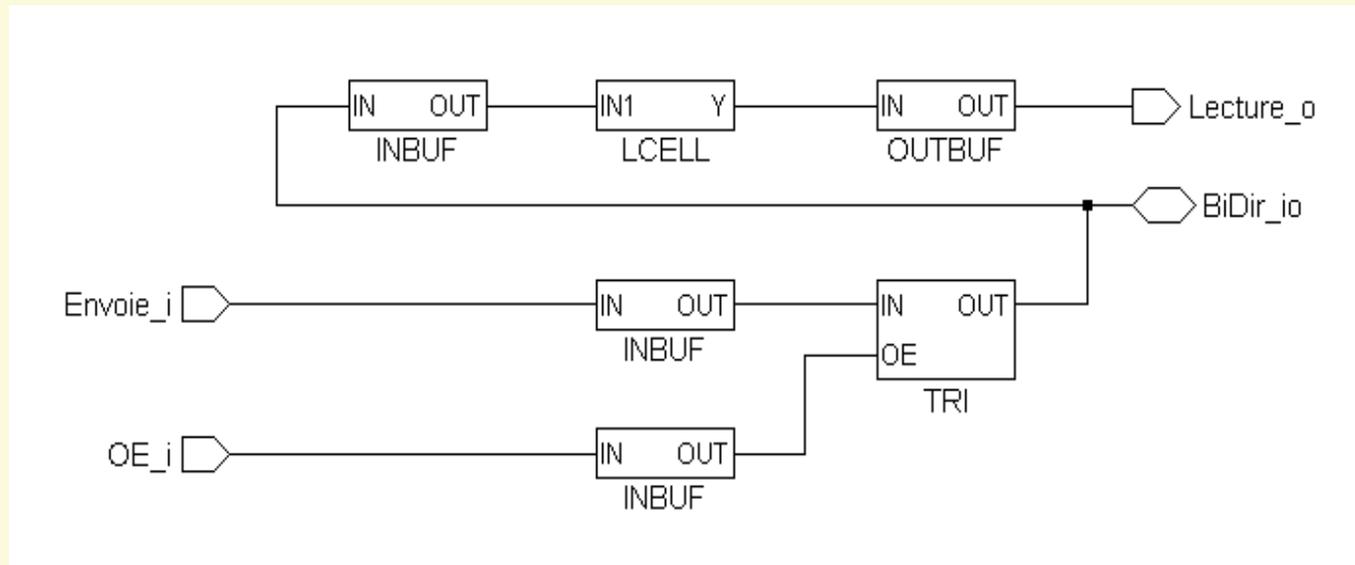
Bus bidirectionnel, vue RTL

- La vue « logique » du synthétiseur



Bus bidirectionnel, vue Technology

- Traduction en «matériel» du synthétiseur



Exercices

1. Réaliser une porte à collecteur ouvert avec une porte trois états.
Donner le schéma de la porte à collecteur ouvert.
2. Décrire en VHDL synthétisable une porte à collecteur ouvert
3. Réaliser un multiplexeur 4 à 1 avec un décodeur 2 à 4 et des portes trois états.
Donner le schéma du multiplexeur.

- Loi de MOORE:
 - http://en.wikipedia.org/wiki/Moore%27s_law
- Histoire technologie CMOS:
 - <http://www.icknowledge.com/history/>
 - http://en.wikipedia.org/wiki/Semiconductor_device_fabrication
 - <http://en.wikipedia.org/wiki/Semiconductor>