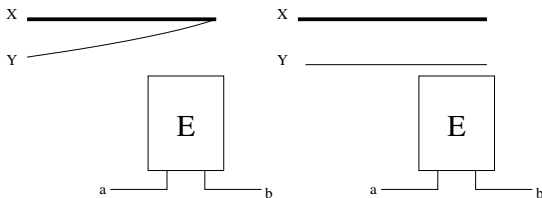


# Architecture des ordinateurs

Circuits pour calculer

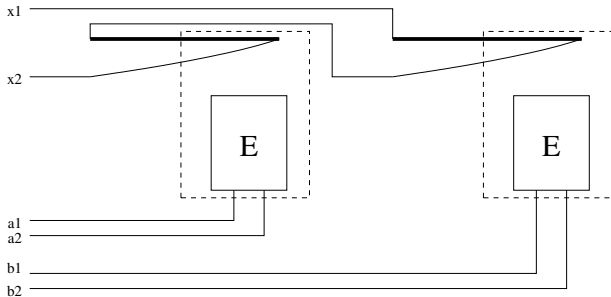
# Relais (1/2)

- ▶ Construit à base d'**électro-aimants**
- ▶ Utilisés pour construire des calculateurs entre 1920 et 1940
- ▶ Élément de base d'un circuit logique :  
courant entre  $a$  et  $b$  (V)  $\Rightarrow$  pas de courant entre  $X$  et  $Y$  (F)  
pas de courant entre  $a$  et  $b$  (F)  $\Rightarrow$  courant entre  $X$  et  $Y$  (V)



## Relais (2/2)

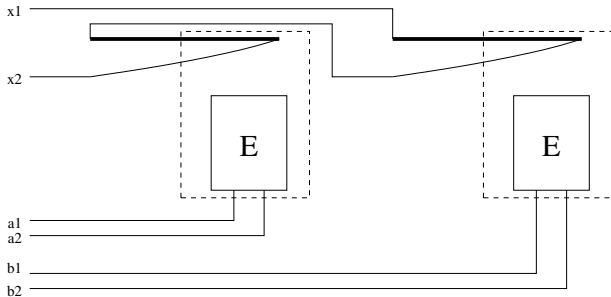
- Circuit qui code ?



- Années 1940 : utilisation de lampes électroniques (commutent plus rapidement)
- Milieu des années 1950 : découverte du transistor

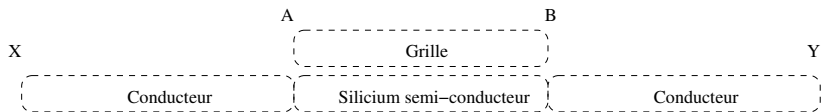
## Relais (2/2)

- Circuit qui code ?  $\neg a \wedge \neg b$



- Années 1940 : utilisation de lampes électroniques (commutent plus rapidement)
- Milieu des années 1950 : découverte du transistor

# Transistors



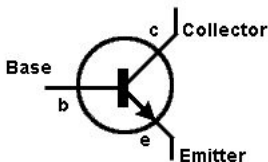
- ▶ tension entre *A* et *B* : silicium semi-conducteur dans l'état conducteur, le courant circule entre *X* et *Y*
- ▶ PNP : quand la grille est alimentée le courant passe
- ▶ NPN : quand la grille est alimentée le courant ne passe plus

# Transistors

- ▶ PNP : quand la grille est alimentée le courant passe
- ▶ NPN : quand la grille est alimentée le courant ne passe plus

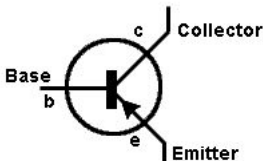
## Bipolar Transistor Circuit Symbols

NPN Transistor



**N** **N**ever  
**P** **P**oints  
**N** **i**N

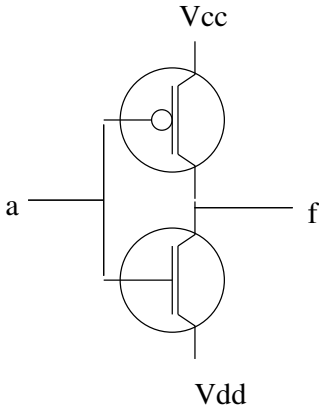
PNP Transistor



**P** **P**oints  
**N** **i**N  
**P** **P**ermanently

# Inverseur CMOS (circuit VLSI)

- ▶  $V_{cc}$  : tension haute,  $V_{dd}$  : tension basse
- ▶ a : entrée
- ▶ f : sortie (fonction NON)
- ▶ Deux transistors N et P pour éviter les courts circuits



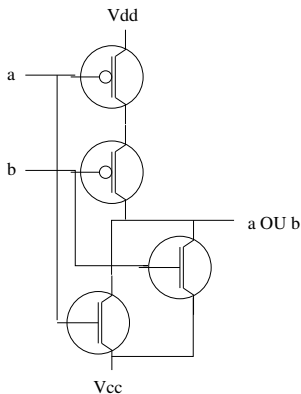
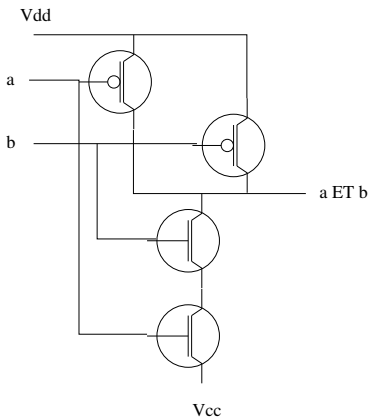
# Circuit CMOS

- ▶ Circuit qui réalise un ET ?
- ▶ Circuit qui réalise un OU ?



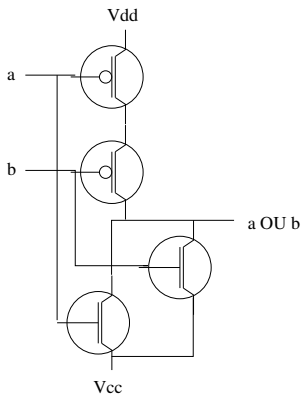
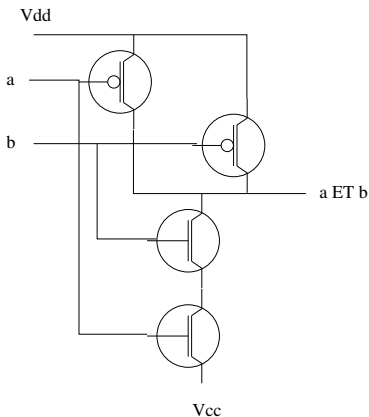
# Circuit CMOS

- ▶ Circuit qui réalise un ET ?
- ▶ Circuit qui réalise un OU ?



# Circuit CMOS

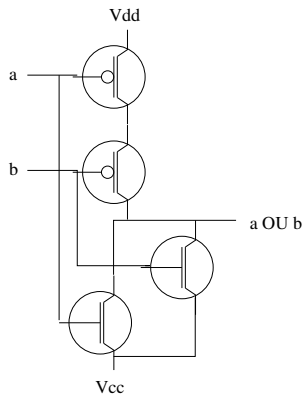
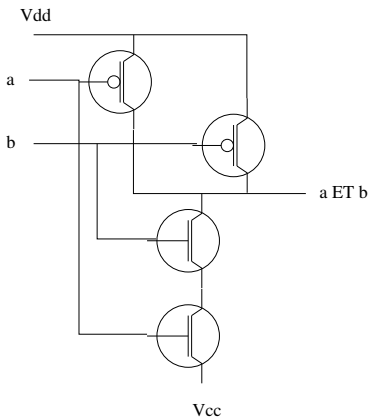
- ▶ Circuit qui réalise un ET ?
- ▶ Circuit qui réalise un OU ?



- ▶ Circuit qui réalise un NON-ET ?
- ▶ Circuit qui réalise un NON-OU ?

# Circuit CMOS

- ▶ Circuit qui réalise un ET ?
- ▶ Circuit qui réalise un OU ?

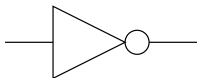


- ▶ Circuit qui réalise un NON-ET ?
- ▶ Circuit qui réalise un NON-OU ? on inverse V<sub>cc</sub> et V<sub>dd</sub>

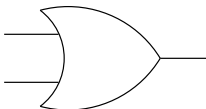
# Exercice

- ▶ Dessiner un circuit CMOS qui réalise un OU exclusif

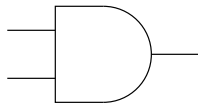
# Portes logiques ET, OU, NON



NON



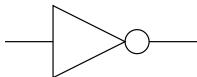
OU



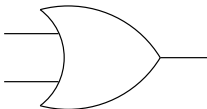
ET

Circuit équivalent à  $f(a, b, c) = \bar{a}.\bar{b}.\bar{c} + \bar{a}.b.c + a.b.\bar{c}$  ?

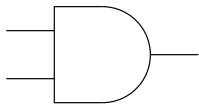
# Portes logiques ET, OU, NON



NON

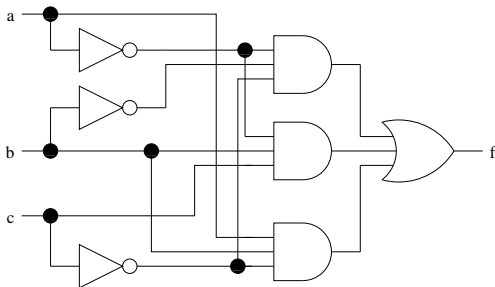


OU



ET

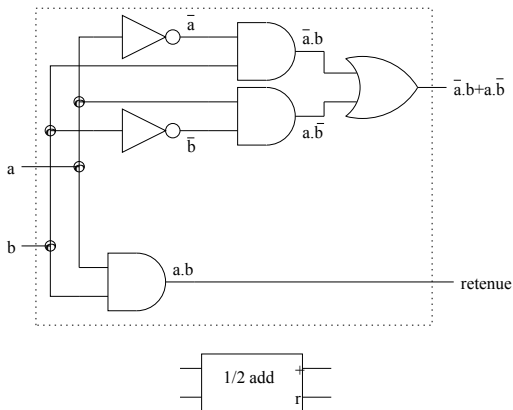
Circuit équivalent à  $f(a, b, c) = \bar{a}.\bar{b}.\bar{c} + \bar{a}.b.c + a.b.\bar{c}$ ?



# Demi-additionneur

<i>a</i>	<i>b</i>	somme <i>a + b</i>	<i>r</i>	<i>s</i>
0	0	0	0	0
0	1	1	0	1
1	0	1	0	1
1	1	10	1	0

$$r = a.b \text{ et } s = a.\bar{b} + \bar{a}.b$$



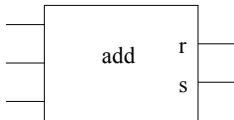
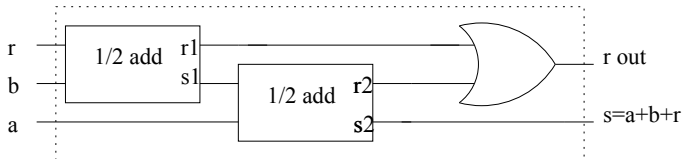
# Additionneur 1 bit

$a$	$b$	$r$	$r_1$	$s_1$	$r_2$	$s_2$	$a + b + r$	$r\ out$	$s$
0	0	0	0	0	0	0	0	0	0
0	1	0	1	1	1	1	1	0	1
1	0	0	0	1	1	1	1	0	1
1	1	0	1	1	0	10	10	1	0
0	0	1	1	1	1	1	1	0	1
0	1	1	1	0	0	10	10	1	0
1	0	1	1	1	0	10	10	1	0
1	1	1	1	0	1	11	11	1	1

$$s = s_2$$

et

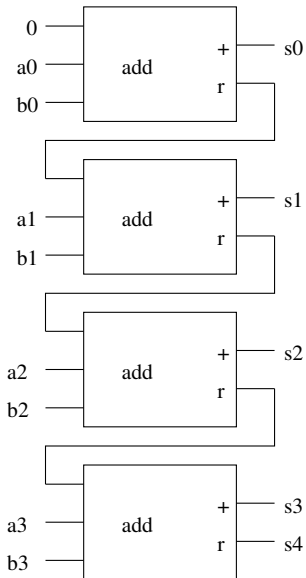
$$r_3 = r_1 \vee r_2$$





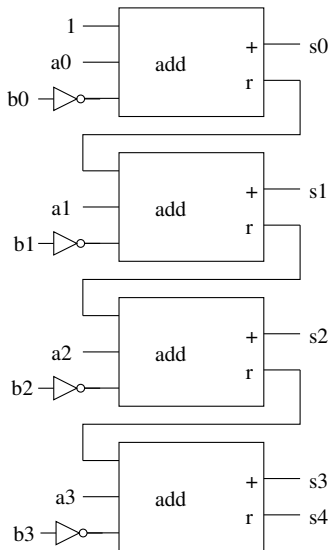
# Additionneur 4 bit

- ▶ addition de  $a_0a_1a_2a_3$   
avec  $b_0b_1b_2b_3$
- ▶ retenue initiale à 0



# Soustracteur 4 bit

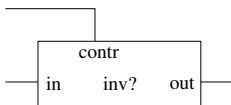
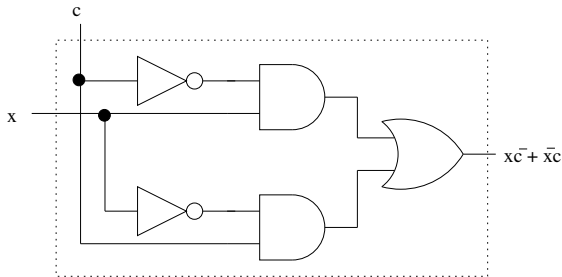
- ▶  $c_1(x)$  complément à 1 d'un nombre  $x$
- ▶ complément à 2 :  $c_1(x) + 1$
- ▶  $a - b = a + c_1(b) + 1$
- ▶ inversion de  $b$
- ▶ retenue initiale à 1



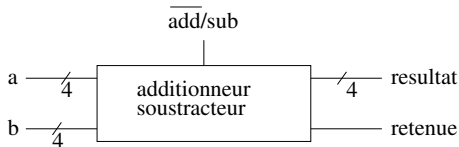
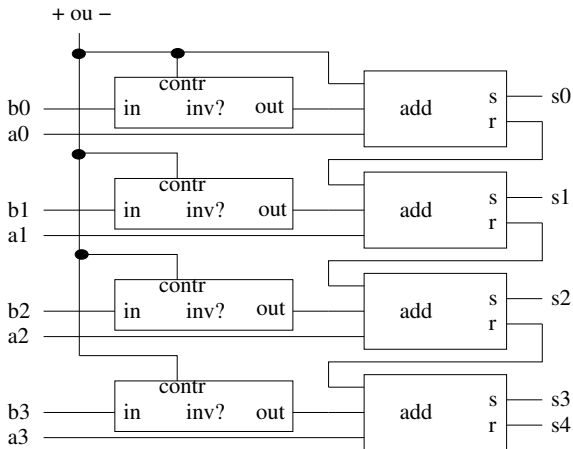
# Inverseur

- ▶ Faire un seul circuit qui additionne ou soustrait
- ▶ Besoin d'un bout de circuit pour basculer  $b$  en fonction d'un signal de contrôle
- ▶ On a déjà ce circuit : c'est un morceau du demi-additionneur !

$c$	$x$	$f$
0	0	0
0	1	1
1	0	1
1	1	0



# Additionneur/Soustracteur (2/2)



# Exercice

- ▶ Installer le logiciel logisim  
(<http://www.cburch.com/logisim/>) (ou smartsim si logisim ne marche pas sur votre machine (<http://smartsim.org.uk/>)).  
Ces logiciels sont disponibles au bocal.
- ▶ Lisez la documentation pour prendre le logiciel en main
- ▶ Créer les circuits (composants) 1/2add, add 1bit, inverseur, add/sub 4bit