



8. Portes logiques

Principes de fonctionnement des ordinateurs

Jonas Lätt

Centre Universitaire d'Informatique



Trouvé une erreur sur un transparent? Envoyez-moi un message

- sur Twitter [@teachjl](#) ou
- par e-mail jonas.latt@unige.ch



Contenu du cours

Partie I: Introduction

1. Introduction

2. Histoire de l'informatique

3. Information digitale et codage de l'information

4. Codage des nombres entiers naturels

5. Codage des nombres entiers relatifs

6. Codage des nombres réels

7. Codage de contenu média

8. Portes logiques

9. Circuits logiques combinatoires et algèbre de Boole

10. Réalisation d'un circuit combinatoire

11. Circuits combinatoires importants

12. Principes de logique séquentielle

13. Réalisation de la bascule DFF

14. Architecture de von Neumann

15. Réalisation des composants

16. Code machine et langage assembleur

17. Réalisation d'un processeur

18. Performance et micro-architecture

19. Du processeur au système

Partie II: Codage de l'information

Partie III: Circuits logiques

Partie IV: Architecture des ordinateurs

Responsabilités d'un ordinateur



Représentation / stockage de données

Traitement / manipulation de données

Séquence d'états binaires

Circuits

- Composant élémentaire: **porte logique**
- Une porte logique traite une faible quantité d'information (quelques bits).



Déduction logique

Au restaurant, vais-je prendre un dessert? Ca depend

- S'il y a des profiterolles, je prends. Autrement
 - Si les desserts ne sont pas faits maison, je ne prends pas.
 - Si les desserts sont faits maison, je prends uniquement
 - Si je suis accompagné, ou
 - Il y a un dessert au chocolat

Motivation: logique mathématique



Propositions logiques (vrai / faux)

Symbole	Proposition
D:	Vous choisissez de prendre un dessert
M:	Les desserts sont faits maison
A:	Vous êtes accompagné(e)
C:	Au moins l'un des desserts est au chocolat
P:	Il y a des profiterolles

Motivation: logique mathématique



Symbole	Proposition
D:	Vous choisissez de prendre un dessert
M:	Les desserts sont faits maison
A:	Vous êtes accompagné(e)
C:	Au moins l'un des desserts est au chocolat
P:	Il y a des profiterolles

D est vrai si (M est vrai et A est vrai) ou (M est vrai et C est vrai) ou P est vrai

Algèbre de Boole



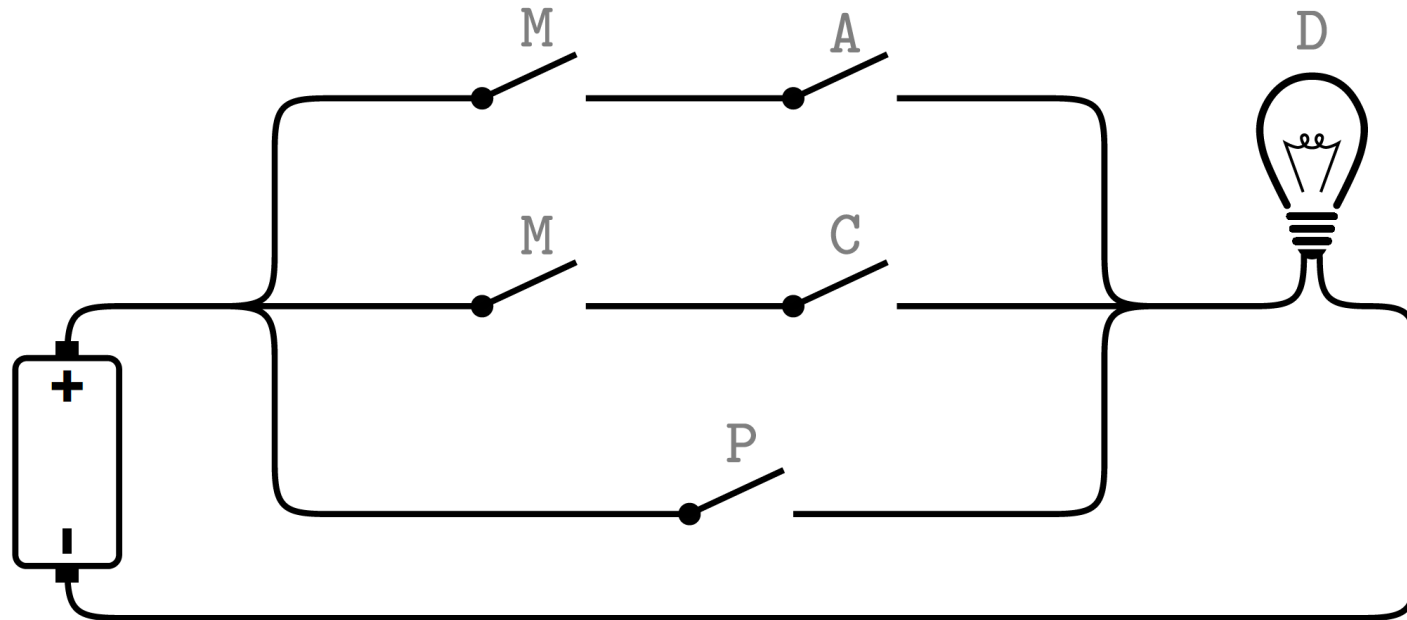
D est vrai si (M est vrai et A est vrai) ou
(M est vrai et C est vrai) ou P est vrai

$$D = (M \cdot A) + (M \cdot C) + P$$

Circuit qui résout un problème logique



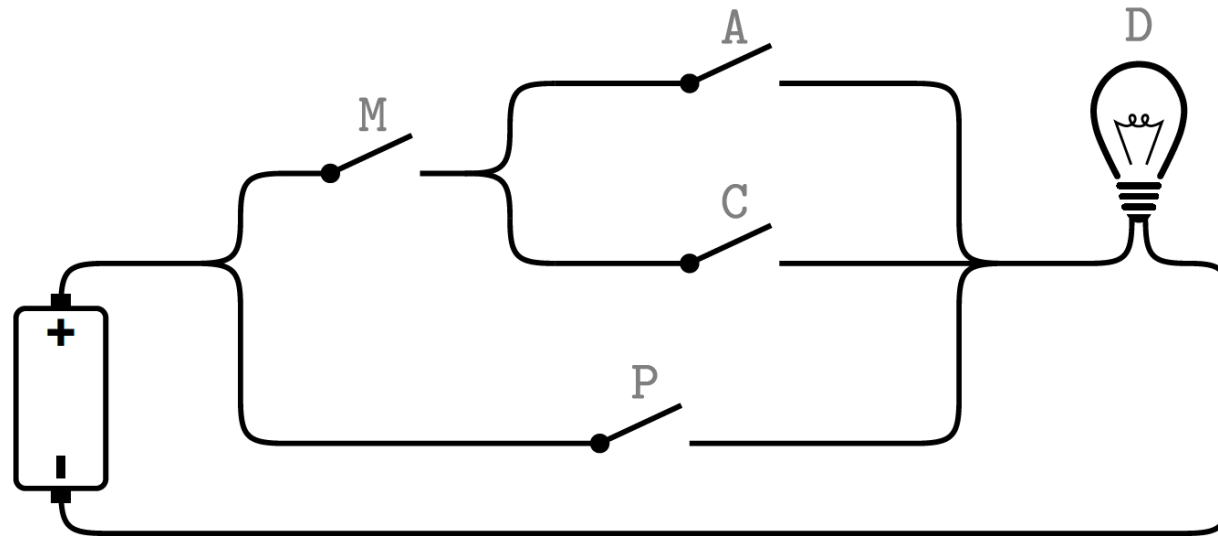
$$D = (M \cdot A) + (M \cdot C) + P$$



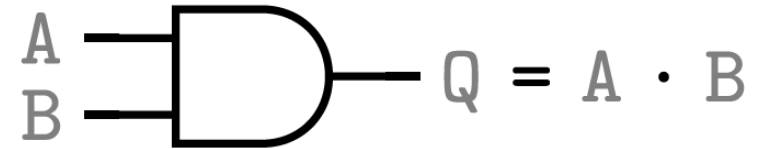
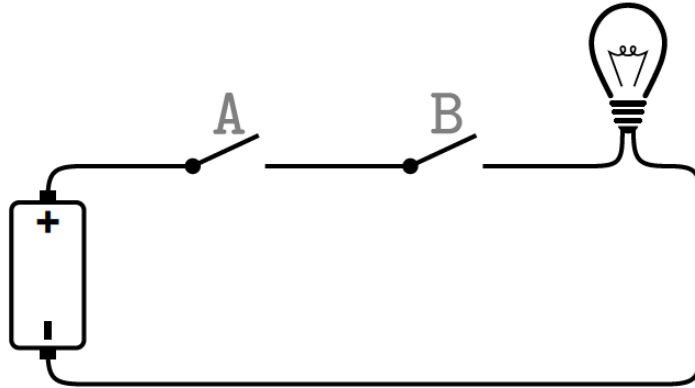
Algèbre de Boole



$$D = (M \cdot A) + (M \cdot C) + P = M \cdot (A + C) + P$$

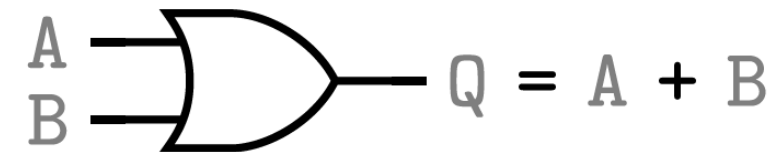
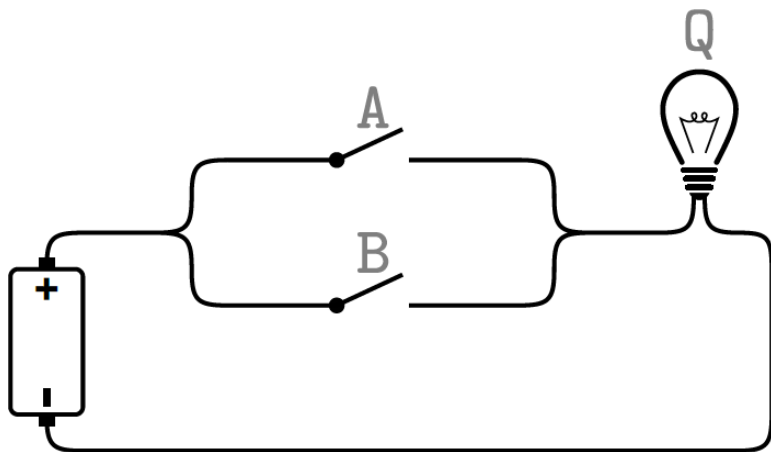
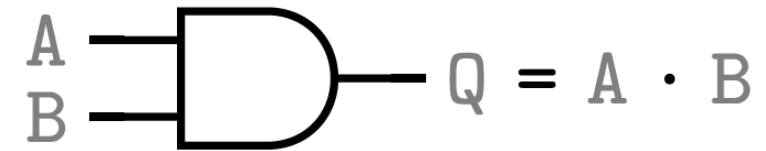
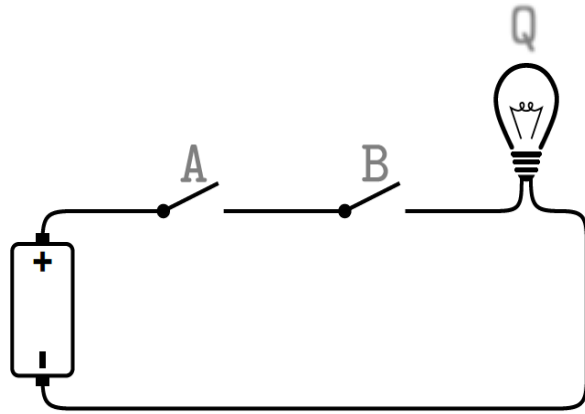


Portes logiques



- Porte logique: symbolise l'exécution d'une opération logique.
- Prend des valeurs binaires en entrée (exemple: A, B).
- Au moins une valeur binaire en sortie (exemple: Q) qui dépend de rien d'autre que les valeurs en entrée.

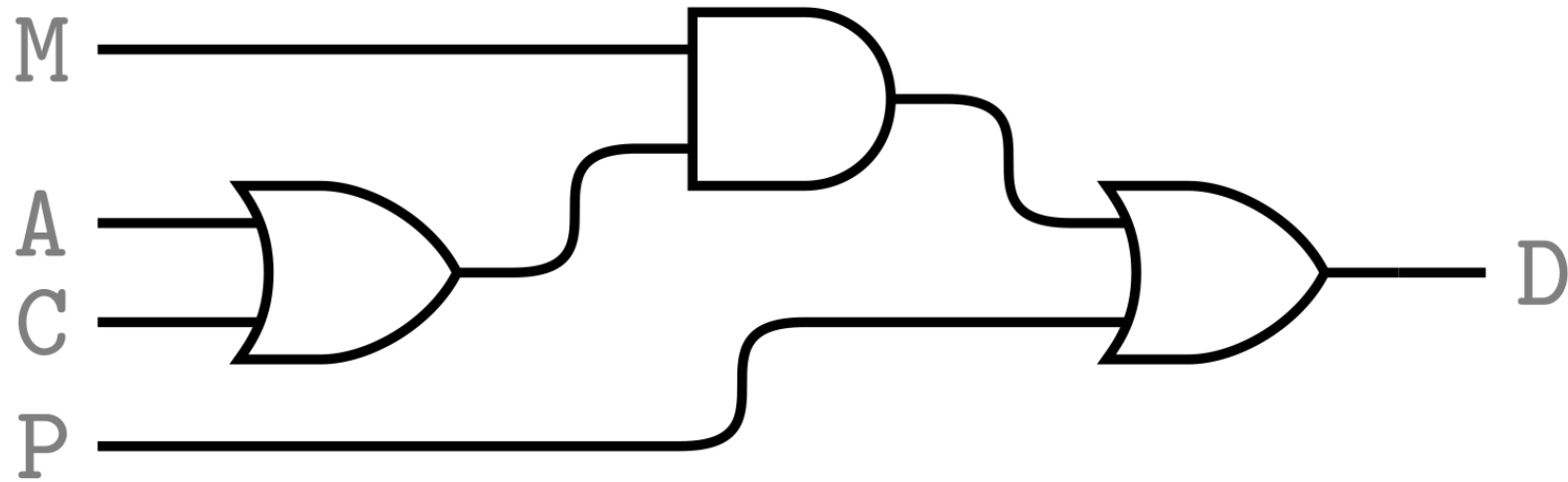
Portes Logiques



Combinaison de portes -> Diagramme de circuit



$$D = M \cdot (A + C) + P$$



Les portes logiques: table de vérité



- Une **table de vérité** énumère la sortie pour toute combinaison de valeurs en entrée.
- Elle décrit une porte de manière unique.

Entrée: l'état binaire
A vaut 0 ou 1

Entrée: l'état binaire B
vaut 0 ou 1

Sortie: l'état
binaire Q
dépend des
valeurs A et B

Exemple: table de vérité
pour la porte du «ET»:
Q vaut 1 si et seulement A
vaut 1 ET B vaut 1.

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

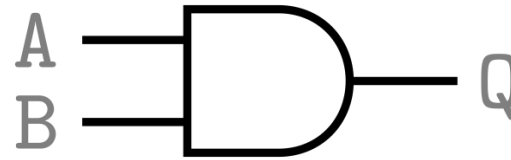
Les portes logiques



Définition d'une porte:
Table de vérité

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

Symboles d'utilisation pour une porte:



$$Q = A \cdot B$$

Symbole de
diagramme logique

Expression logique /
Expression booléenne

10. Portes logiques



Quelques portes logiques importantes

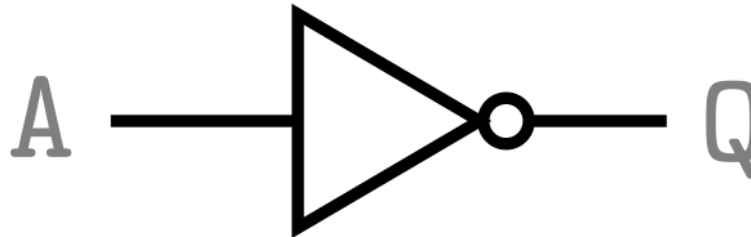
La porte de négation (le NON ou NOT)



Expression Booléenne

Diagramme logique

Table de vérité



A	Q

ou

$$Q = \neg A$$

ou

$$Q = \text{NOT } A$$

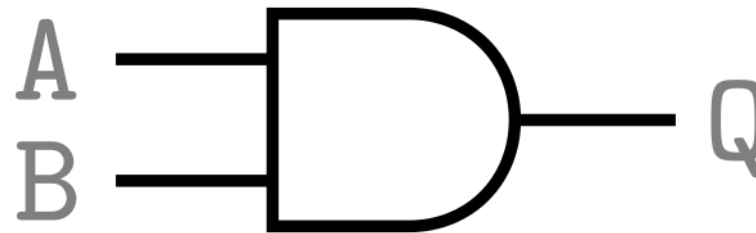
La porte de conjonction (le ET ou AND)



Expression Booléenne

Diagramme logique

Table de vérité



A	B	Q

ou

$$Q = A \wedge B$$

ou

$$Q = A \text{ AND } B$$

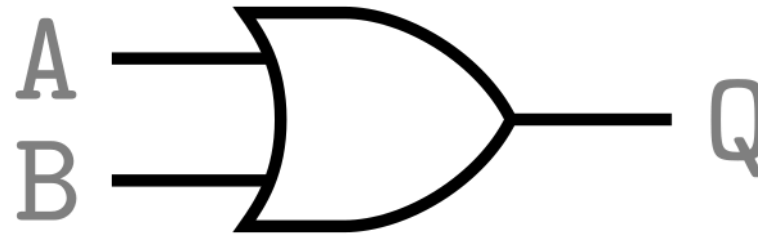
La porte de disjonction (le OU ou OR)



Expression Booléenne

Diagramme logique

Table de vérité



A	B	Q

ou

$$Q = A \vee B$$

ou

$$Q = A \text{ OR } B$$

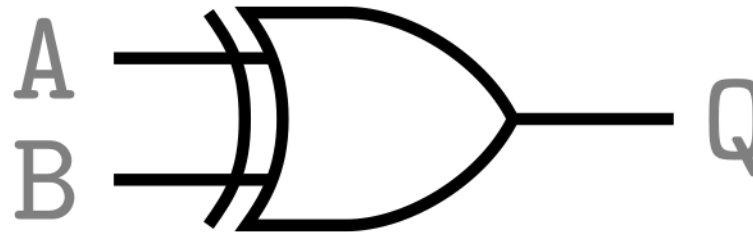
La porte du “OU exclusif” (le XOR)



Expression Booléenne

Diagramme logique

Table de vérité



A	B	Q

ou

$$Q = A \text{ XOR } B$$

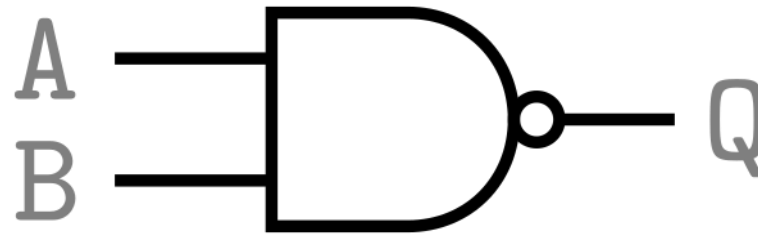
La porte du “ET inversé” (le NAND)



Expression Booléenne

Diagramme logique

Table de vérité



A	B	Q

ou

$$Q = \neg (A \wedge B)$$

ou

$$Q = \text{NAND}(A, B)$$

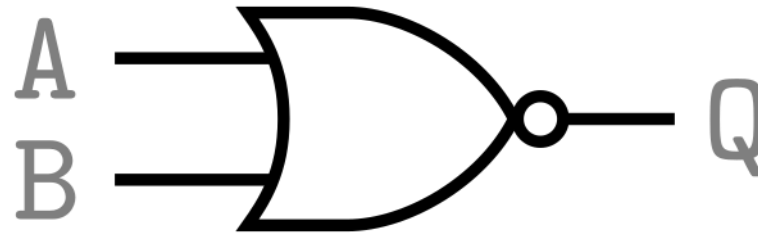
La porte du “OU inversé” (le NOR)



Expression Booléenne

Diagramme logique

Table de vérité



$$Q = \neg (A \vee B)$$

ou

$$Q = \text{NOR}(A, B)$$

A	B	Q

Exercice



Combien de portes logiques à 2 entrées et 1 sortie différentes existe-t-il?

votamatic.unige.ch

VRRN



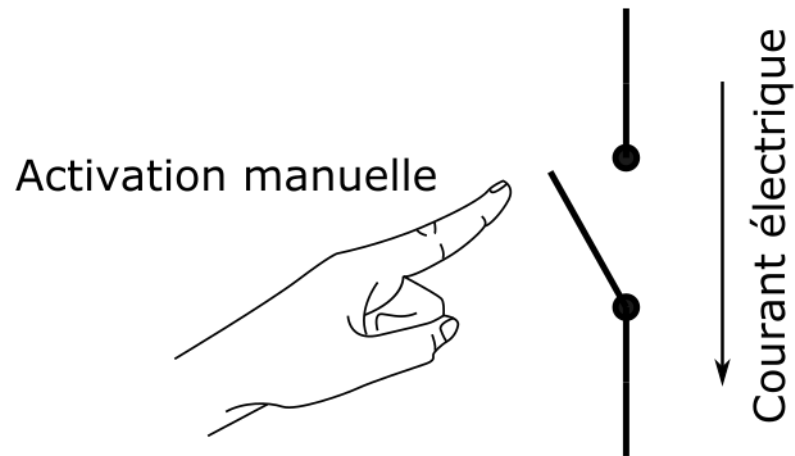


Principes électroniques des portes logiques

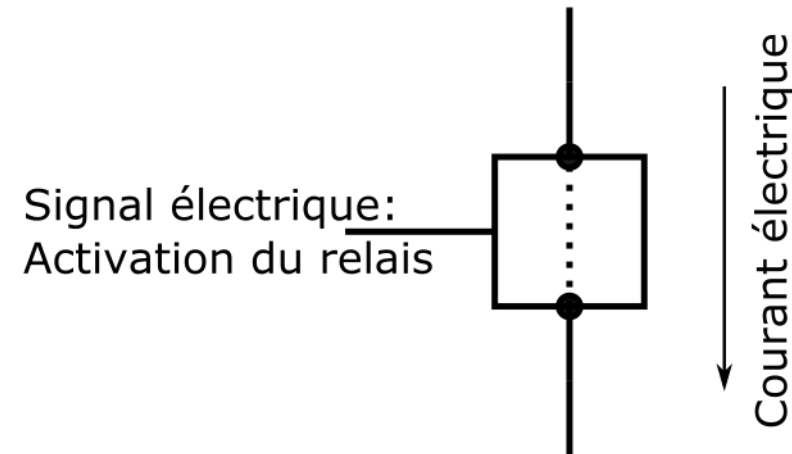
Idée



Interrupteur



Relais électronique: un interrupteur activé de manière automatique



En pratique: les relais électroniques modernes sont réalisés à l'aide de transistors.

Aperçu: technologies de logique électronique



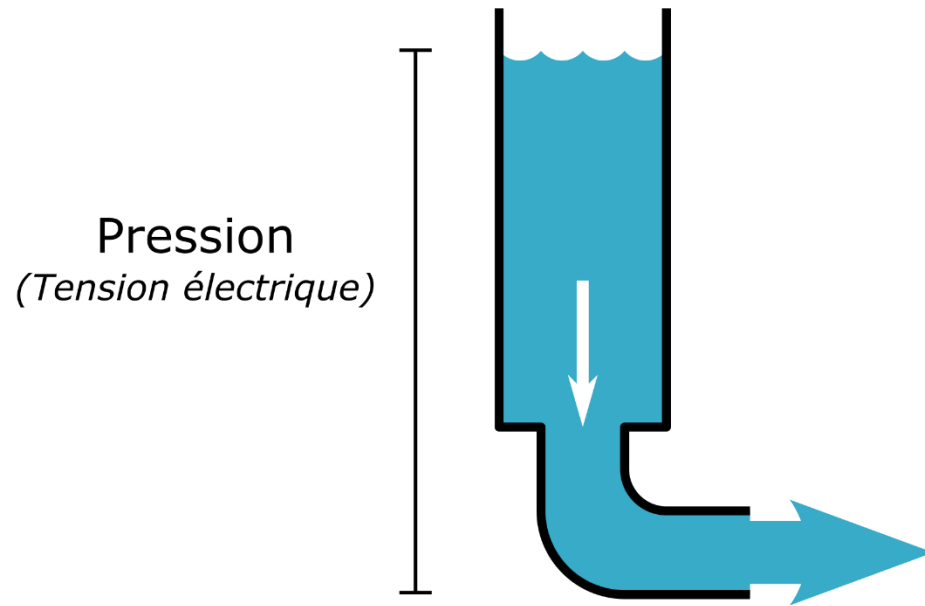
Approches traditionnelles:

- Diodes / Résistances
("Diode-Resistor Logic", DRL)
- Diodes / Transistors
("Diode-Transistor Logic", DTL)
- Transistor / Transistor
("Transistor-Transistor Logic", TTL)
- Résistance / Transistor
("Resistor-Transistor Logic", RTL)

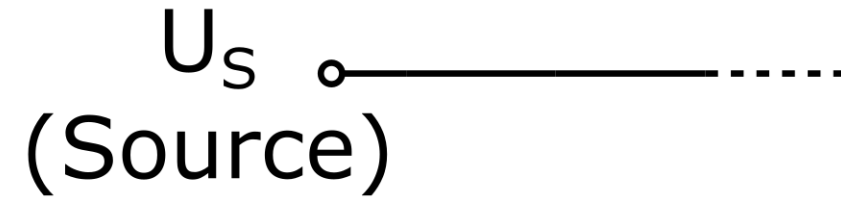
Approche moderne:

- Transistors à effet de champ
("Field-Effect Transistor", FET)
- Réalisation d'une logique TTL
 - Plus efficace et moins gourmand en énergie

Analogie hydraulique de l'électricité

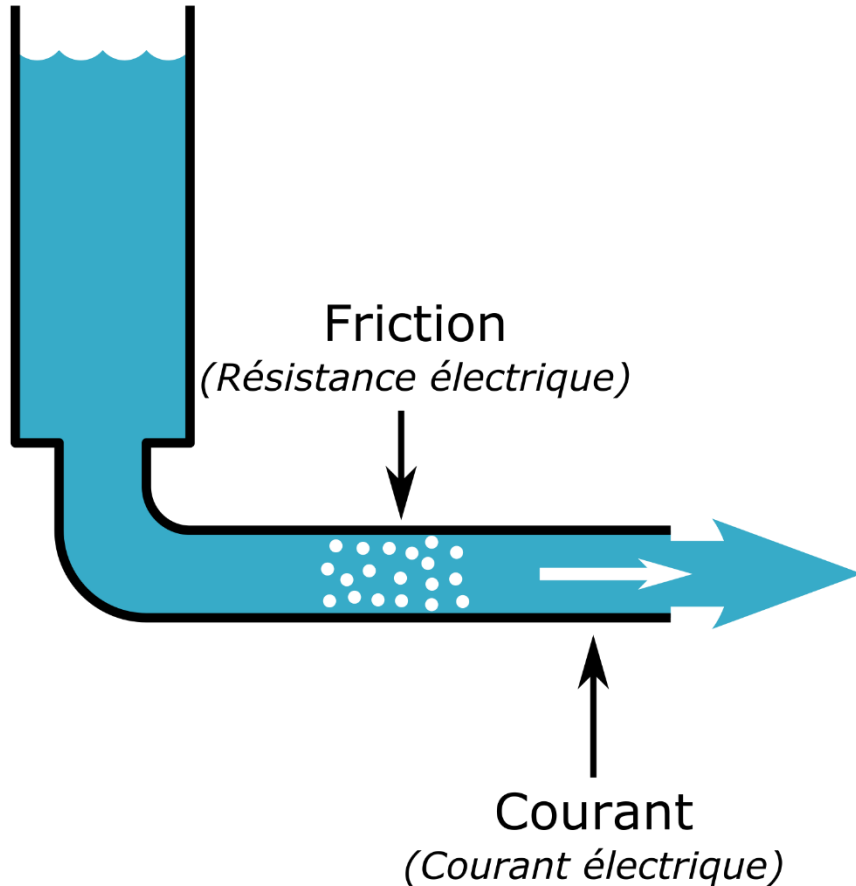


Une pression générée par un réservoir d'eau surélevé.

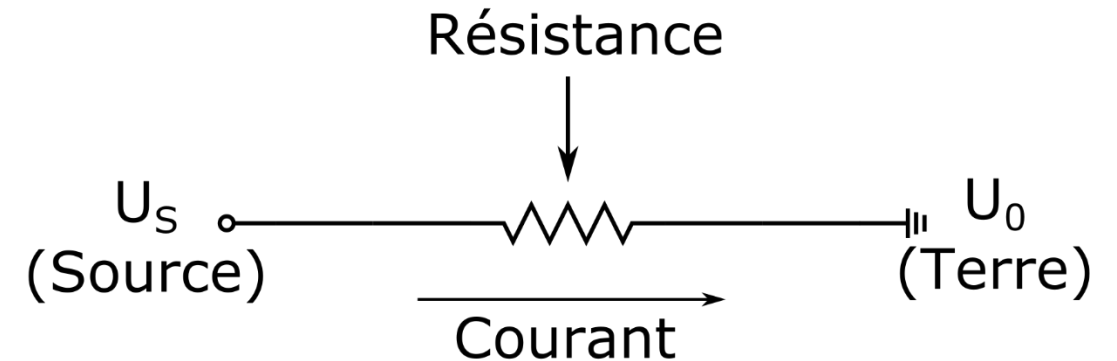


Une tension générée par une source, par exemple une pile.

Analogie hydraulique de l'électricité

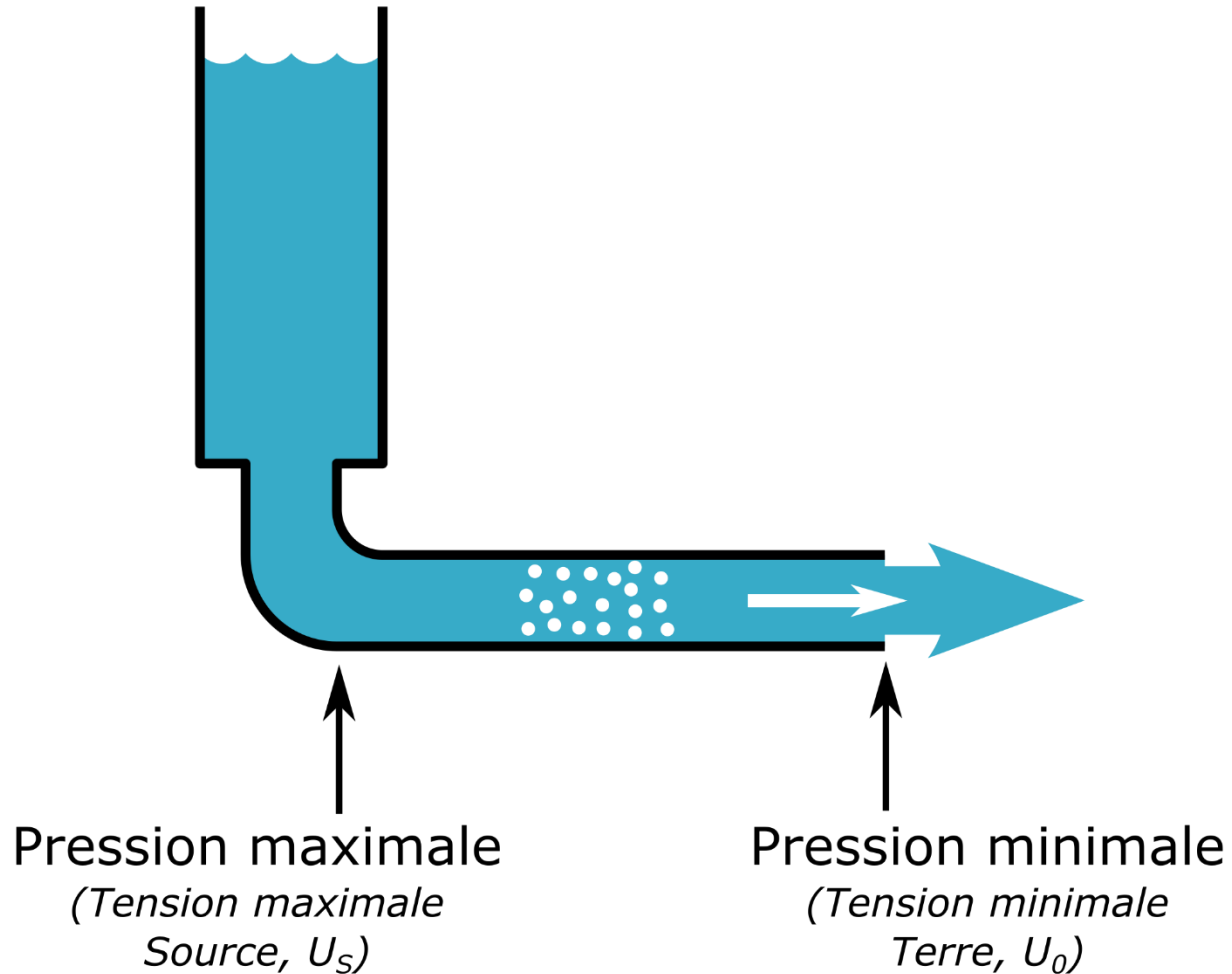


La pression mène à un courant d'eau dans le tuyau.

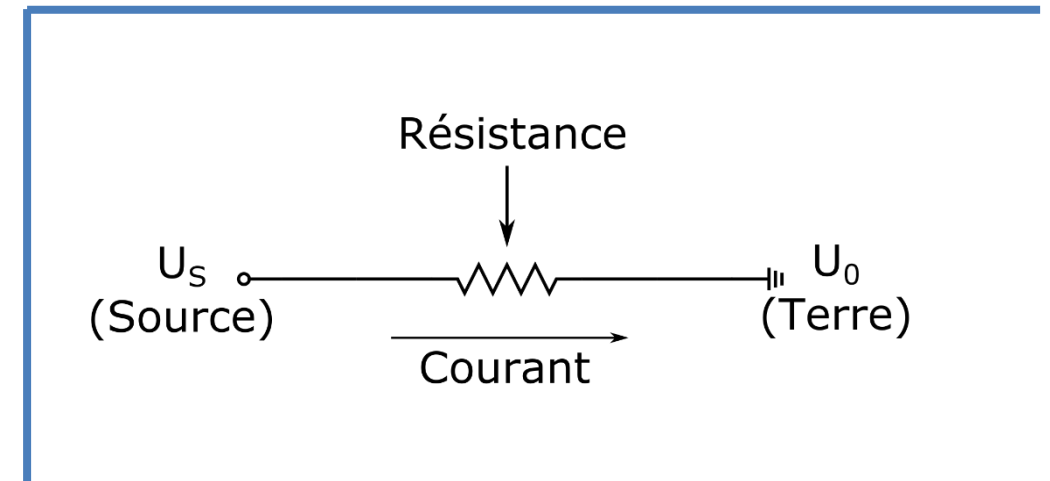


- Différence de tension entre source et terre mène à un courant électrique.
- Valeur du courant dépend de la résistance dans le fil.

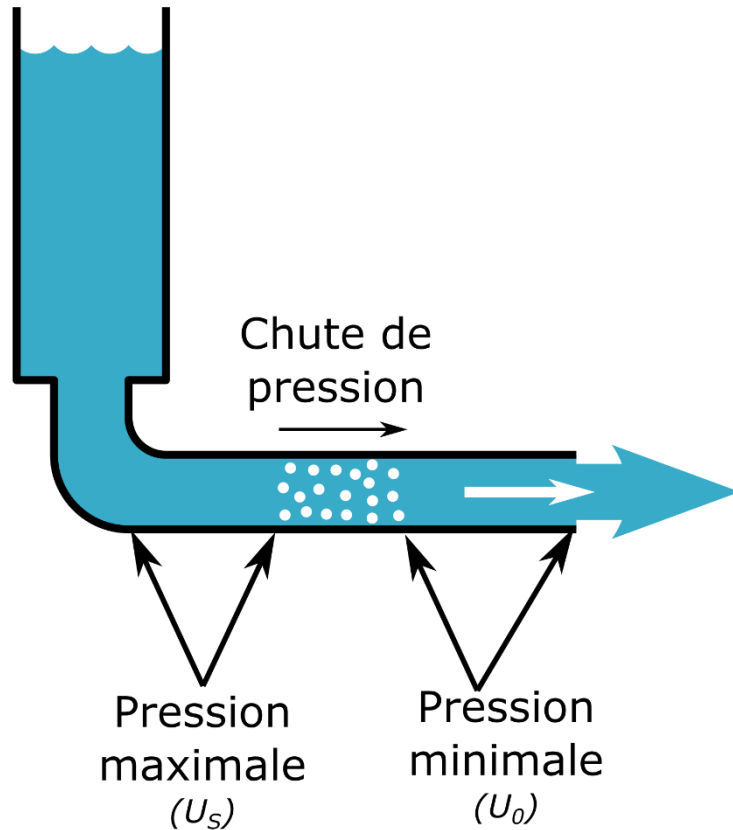
Analogie hydraulique de l'électricité



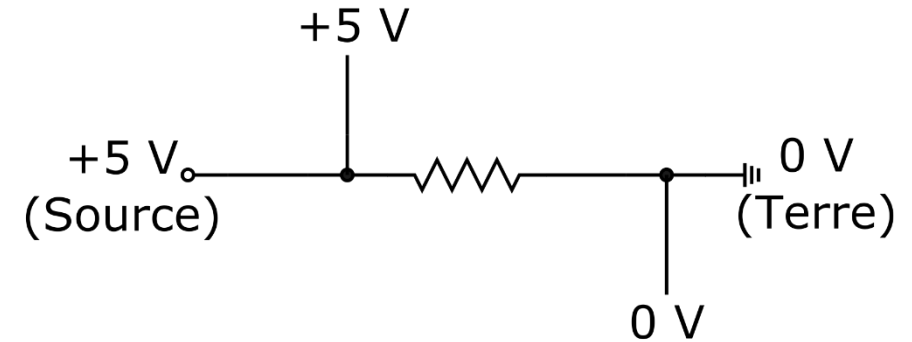
La pression est maximale à l'entrée du tuyau, puis elle chute le long du tuyau pour atteindre sa valeur minimale (la pression atmosphérique) à la sortie.



Analogie hydraulique de l'électricité

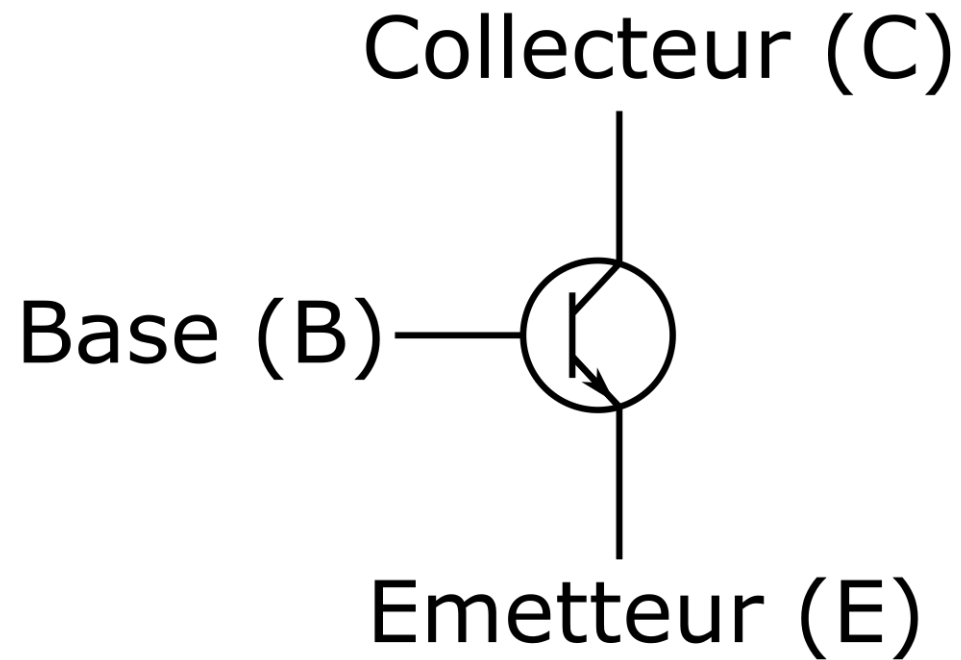


- Modèle: on néglige la friction des paroi. La seule friction est produite par le tas de cailloux au milieu.
- Pression constante à gauche et à droite du tas de cailloux.



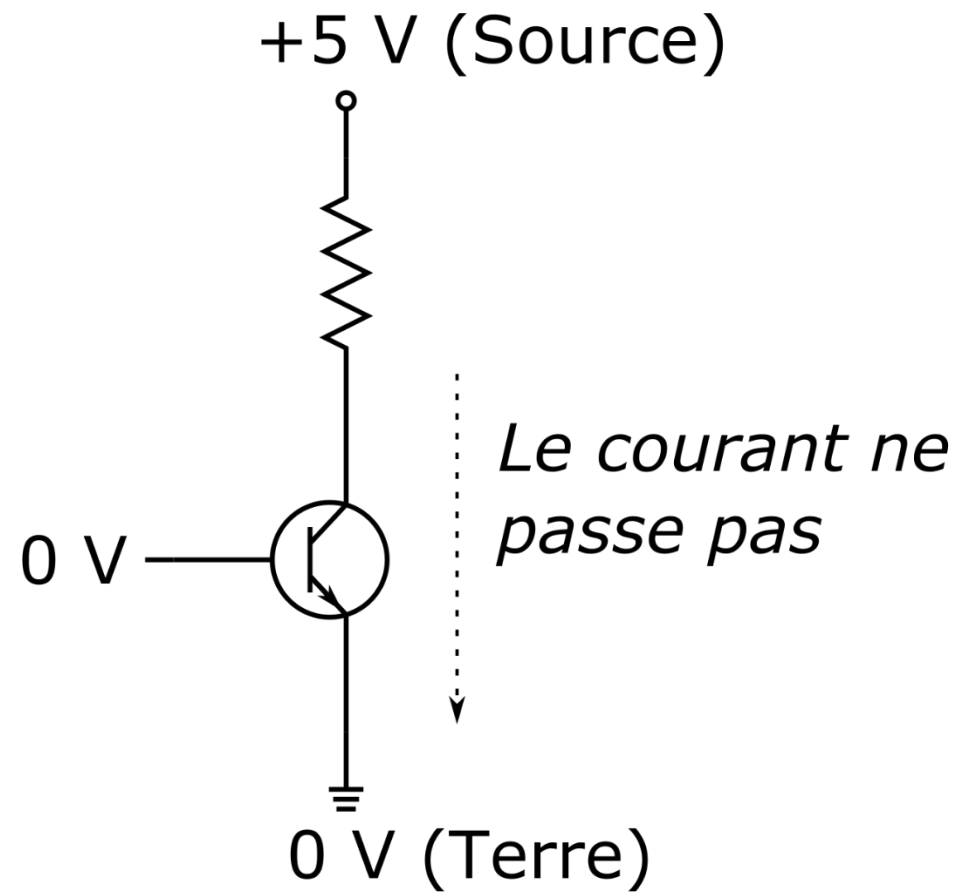
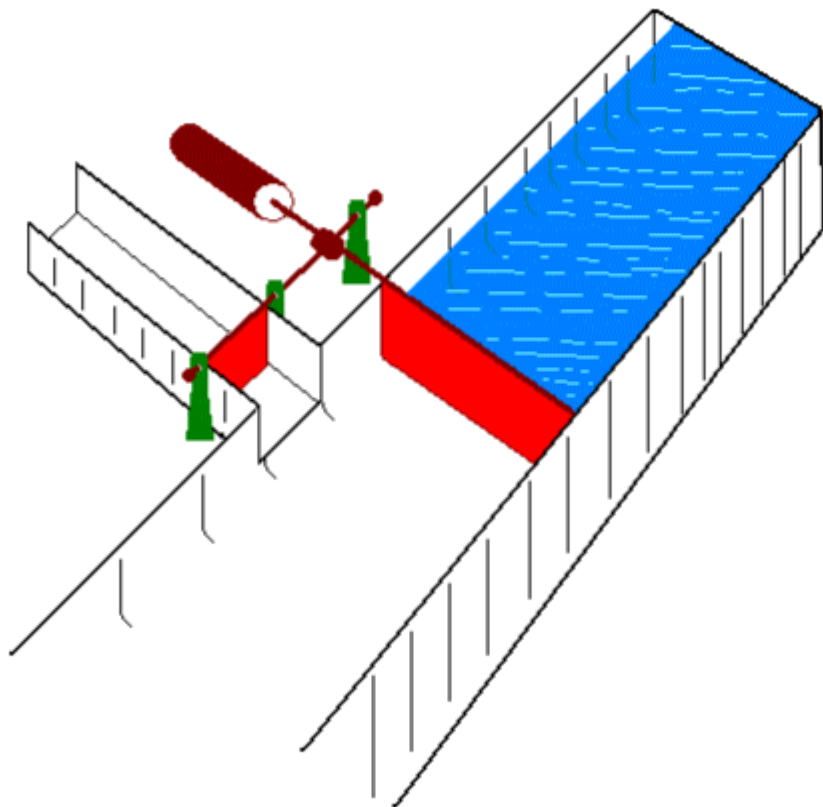
- Modèle: le fil conducteur possède une résistance nulle. La tension n'est modifiée que par la résistance.
- La tension vaut 5 Volts partout à gauche de la résistance.
- La tension vaut 0 Volts partout à droite de la résistance.

Le transistor

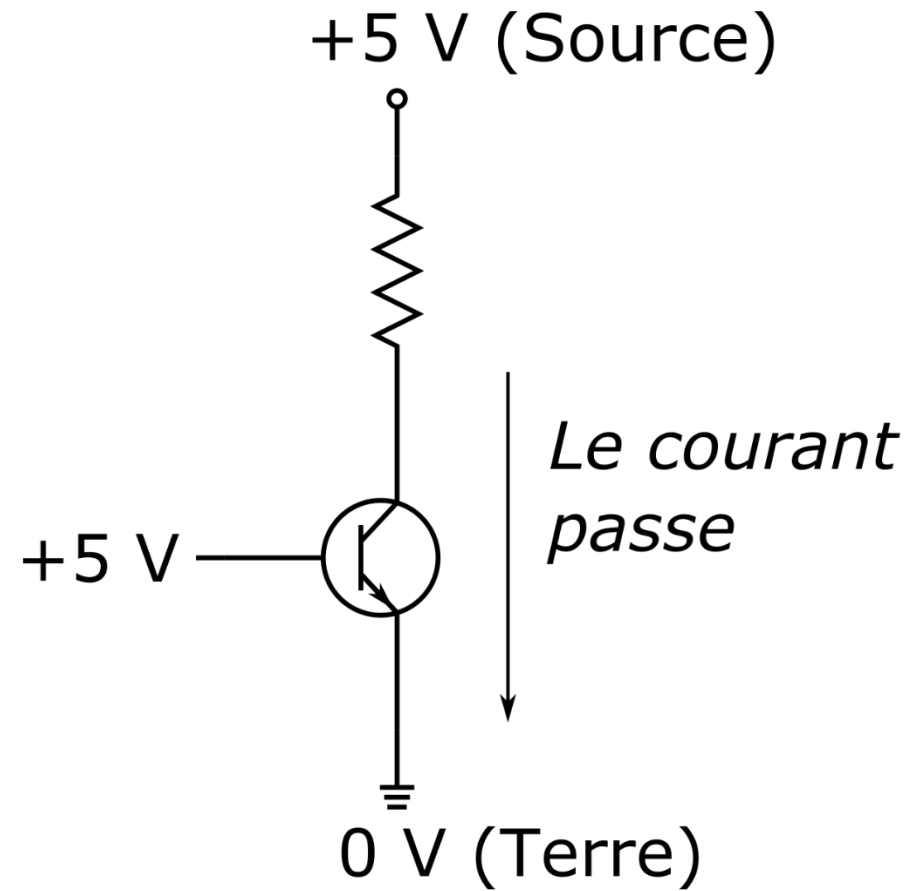
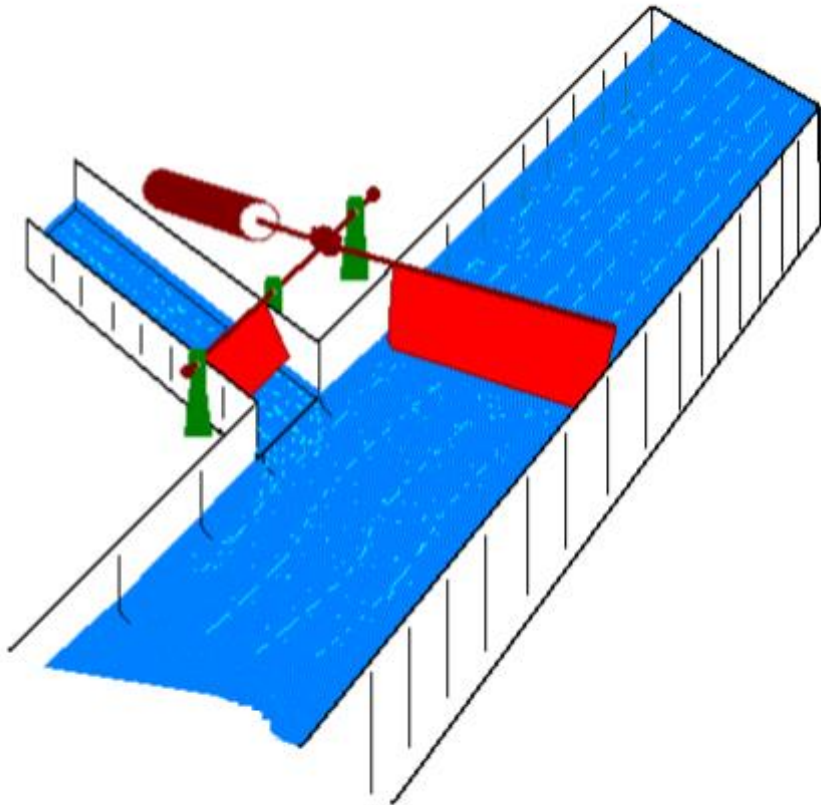


- Deux entrées: Le collecteur et la base.
- Une sortie: l'émetteur.

Le transistor

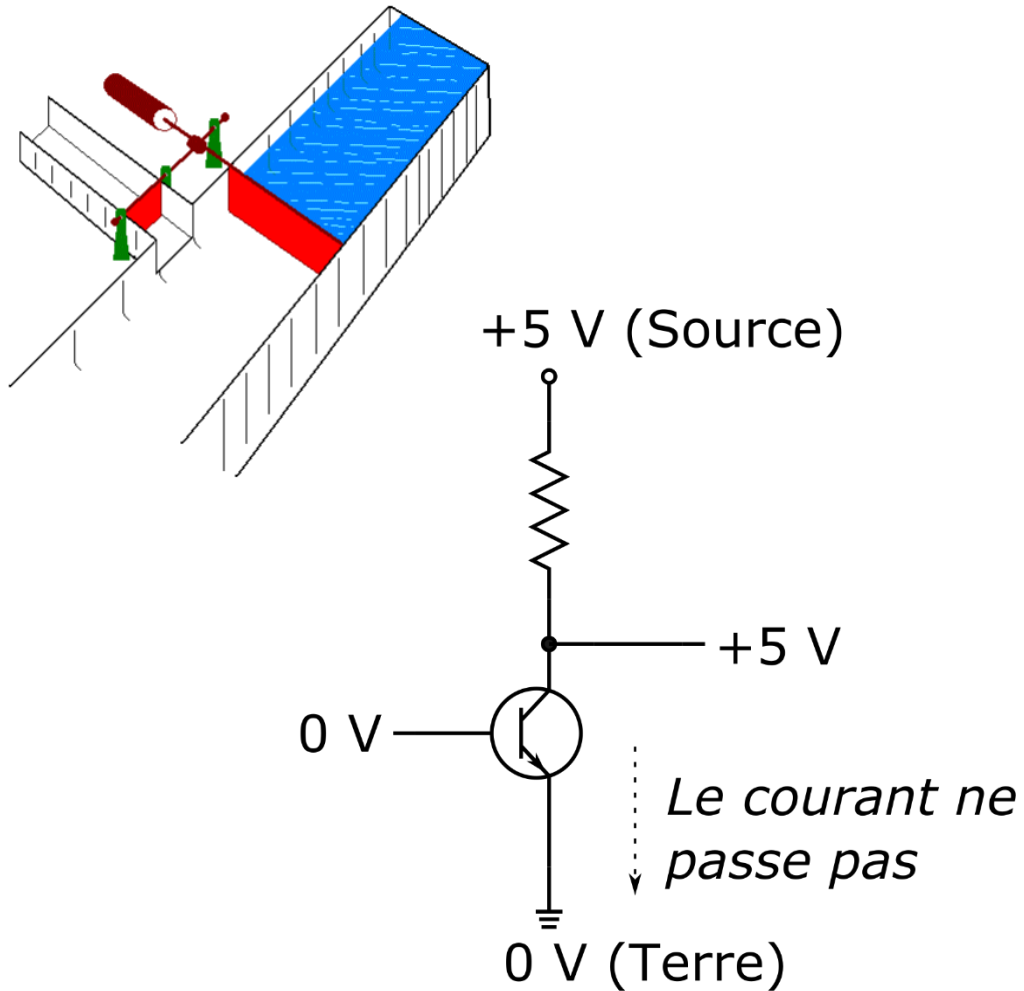


Le transistor

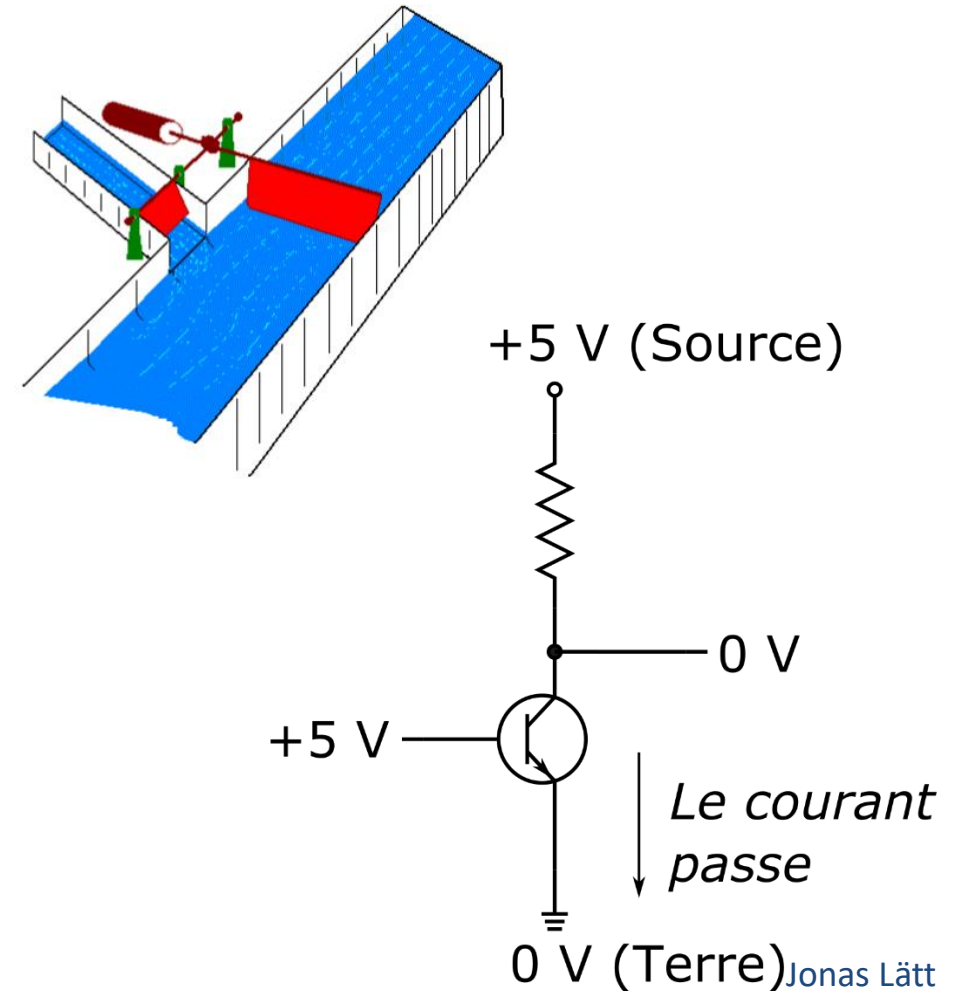


Le transistor

Fermé



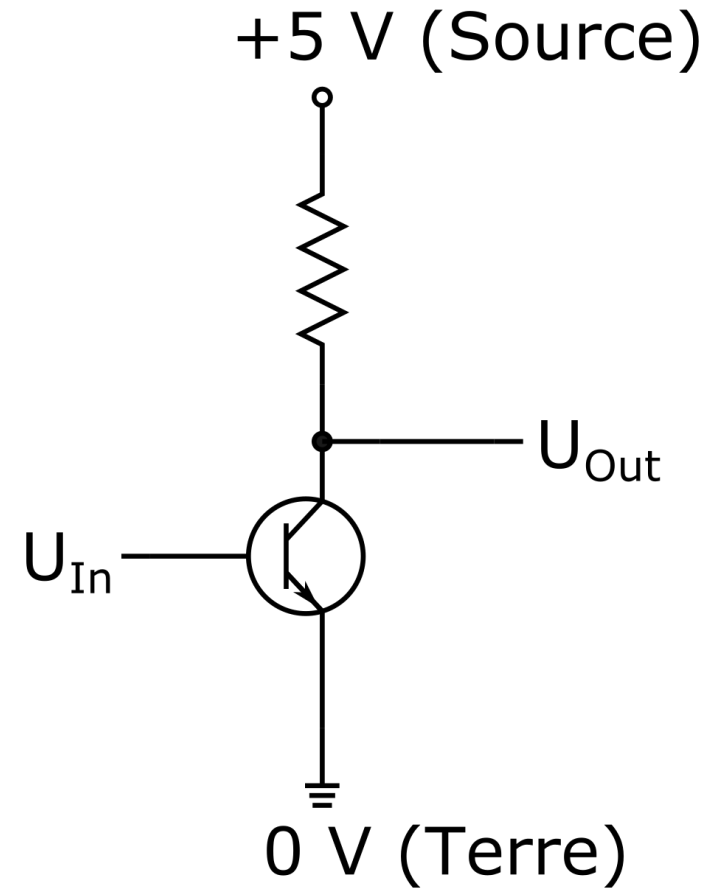
Ouvert



La porte NOT

In	Out
0	1
1	0

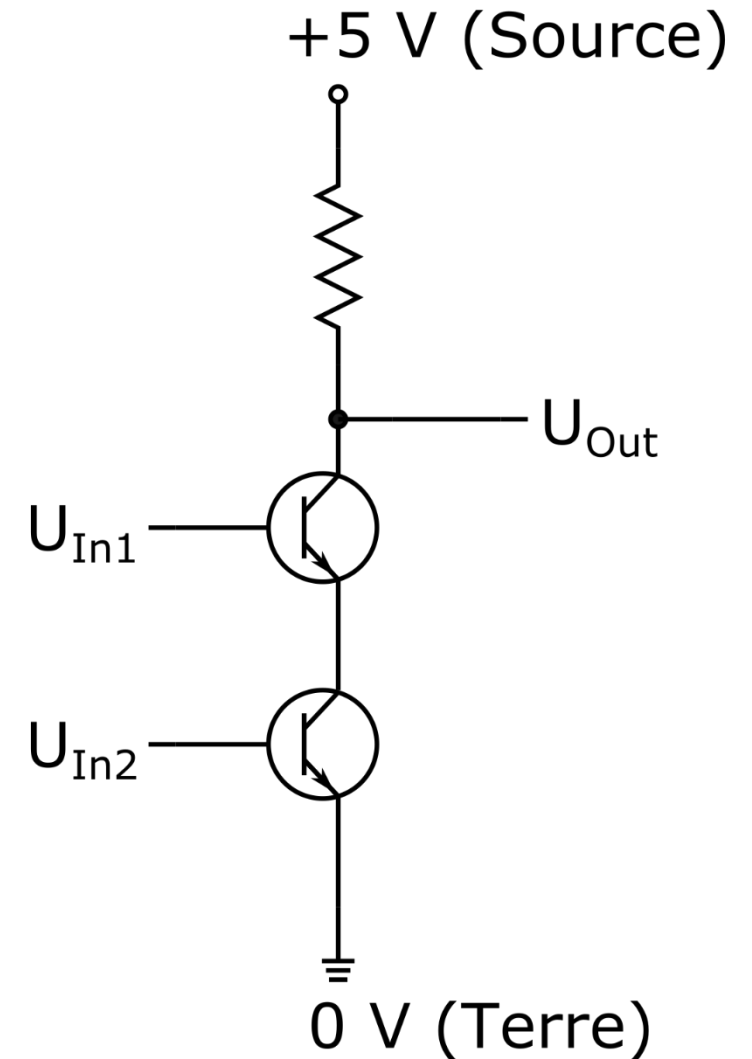
La porte NOT nécessite qu'un seul transistor.



La porte NAND

A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

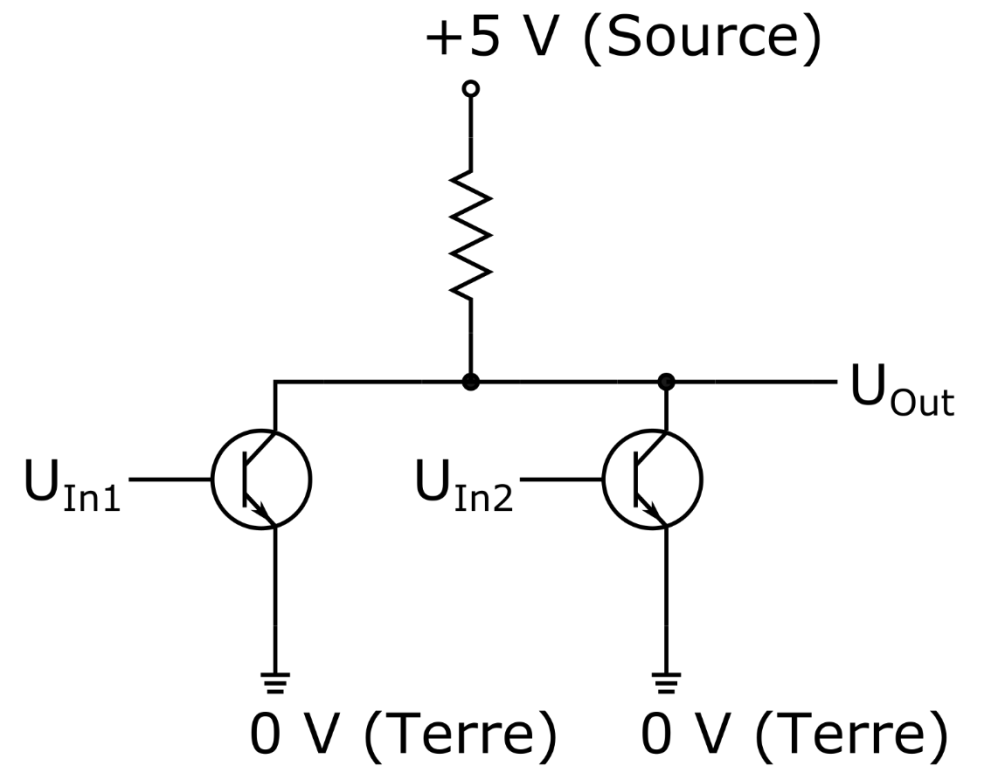
La porte NAND nécessite deux transistors. Les deux signaux d'entrée sont appliqués aux bases des deux transistors.



La porte NOR

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

La porte NOR nécessite deux transistors. Les deux signaux d'entrée sont appliqués aux bases des deux transistors.





Les portes AND et OR

- Pour construire une porte AND, on passe simplement le signal de sortie de la porte NAND à un invertisseur (NOT). La porte AND est donc plus compliquée que la porte NAND: elle nécessite trois transistors.
- De la même manière, une porte OR est obtenue en passant le signal de sortie de la porte NOR à un invertisseur (NOT).