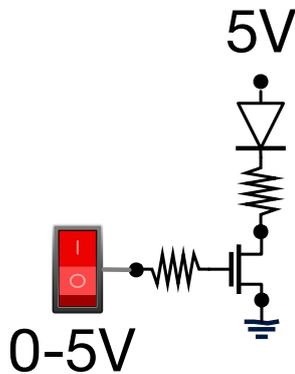


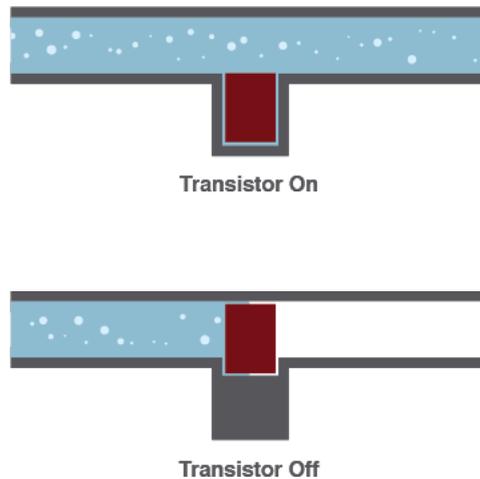
Fonctionnement du transistor

La polarisation de la base le fait passer à l'état PASSANT. Un courant s'établit entre le collecteur et l'émetteur. Dans le montage ci-dessous, le courant allume la diode.

Circuit électronique



Analogie hydraulique



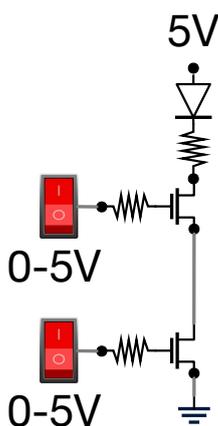
Abstraction binaire : Porte logique OUI

| E | S |
|---|---|
| 0 | 0 |
| 1 | 1 |

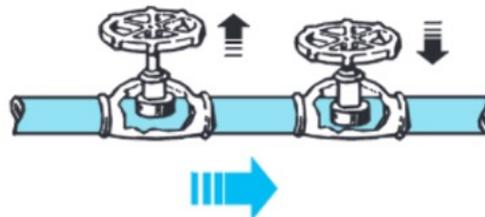
Fonction logique AND

La polarisation de la base est indépendante pour chaque transistor. Observons la diode. Celle-ci s'allume lorsque le transistor T1 est passant ET lorsque le transistor T2 est aussi passant. C'est la fonction logique AND.

Circuit électronique



Analogie hydraulique



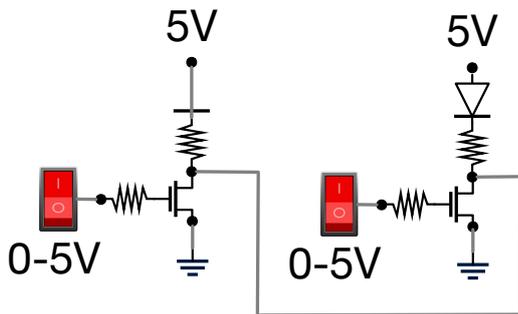
Abstraction binaire : Porte logique AND

| E1 | E2 | S |
|----|----|---|
| 0 | 0 | |
| 0 | 1 | |
| 1 | 0 | |
| 1 | 1 | |

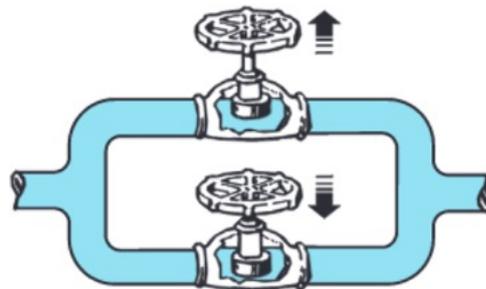
Fonction logique OR

La polarisation de la base est indépendante pour chaque transistor. Observons la diode. Celle-ci s'allume lorsque l'un des 2 transistors, T1 OU T2, est passant. C'est la fonction logique OR.

Circuit électronique



Analogie hydraulique



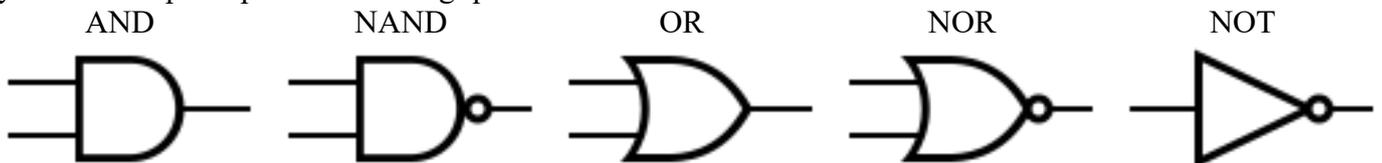
Abstraction binaire : Porte logique OR

| E1 | E2 | S |
|----|----|---|
| 0 | 0 | |
| 0 | 1 | |
| 1 | 0 | |
| 1 | 1 | |

Portes logiques

Une porte logique est un petit circuit électronique composé d'un ou deux transistors et utilisé pour traiter des données binaires. Les portes logiques telles que la porte AND, la porte OR, la porte NOT, la porte XOR et les portes NAND sont les éléments de base de tout système informatique.

Symboles des principales Portes Logiques



L'UAL, Unité Arithmétique et Logique

L'UAL permet de réaliser les opérations arithmétiques en base 2. Il prend en entrée deux mots de 2 bits (les opérandes A et B). Il peut effectuer les opérations +, -, *, /. L'opération à effectuer lui est communiquée par un code opération qui lui est communiqué par l'unité de contrôle (F). La sortie D permet de gérer l'overflow.

Symbole

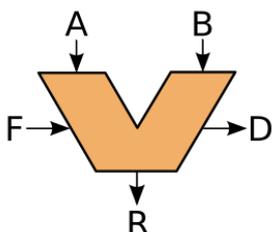
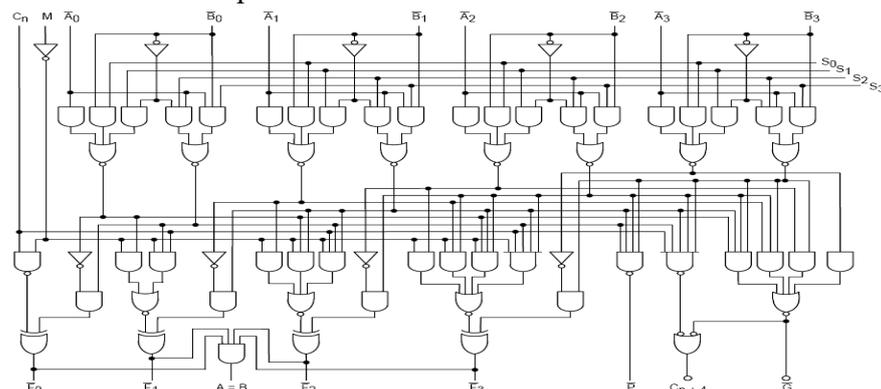


Schéma électronique



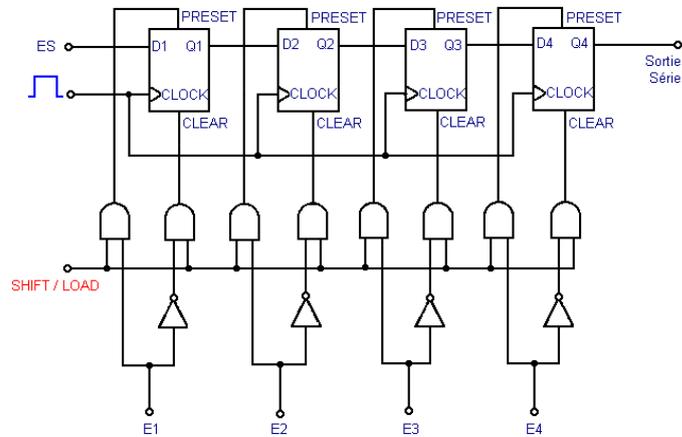
Registres

Un registre est un circuit qui permet de mémoriser plusieurs bits simultanément. Typiquement, on construit un registre en utilisant plusieurs bascules qui seront synchronisées sur le même signal d'horloge. Une bascule est un composant permettant de mémoriser un bit sous l'impulsion d'un clic d'horloge.

Symbole

| Registres |
|-----------|
| r0 |
| r1 |
| r2 |
| ... |

Schéma électronique d'un registre 4 bits



Unité de contrôle / commande

L'unité de contrôle, se charge de gérer le processeur. Il peut décoder les instructions chargées de la RAM, grâce à une unité de décodage. Il choisit les registres à utiliser. Il fait appel à l'unité arithmétique et logique (UAL). Il gère les entrée-sortie pour communiquer avec la mémoire ou les périphériques. Il possède un registre d'instructions (un compteur associé à la lecture du programme pour charger la prochaine instruction).

Symbole

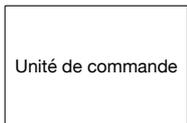
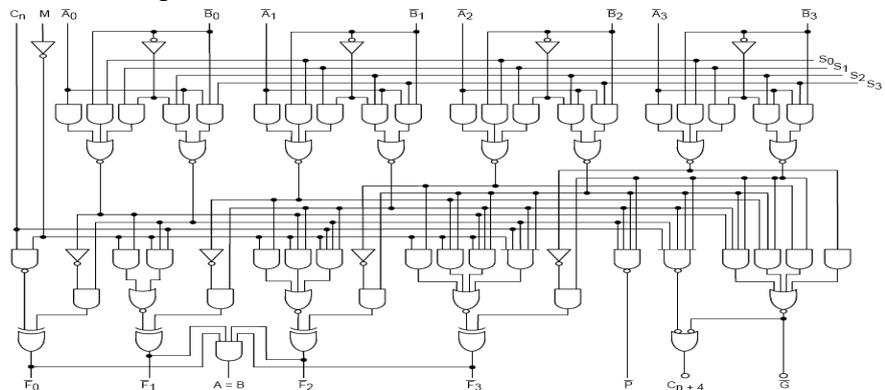
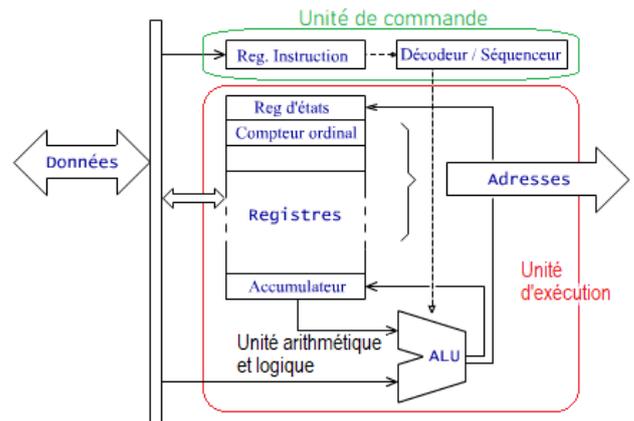


Schéma électronique



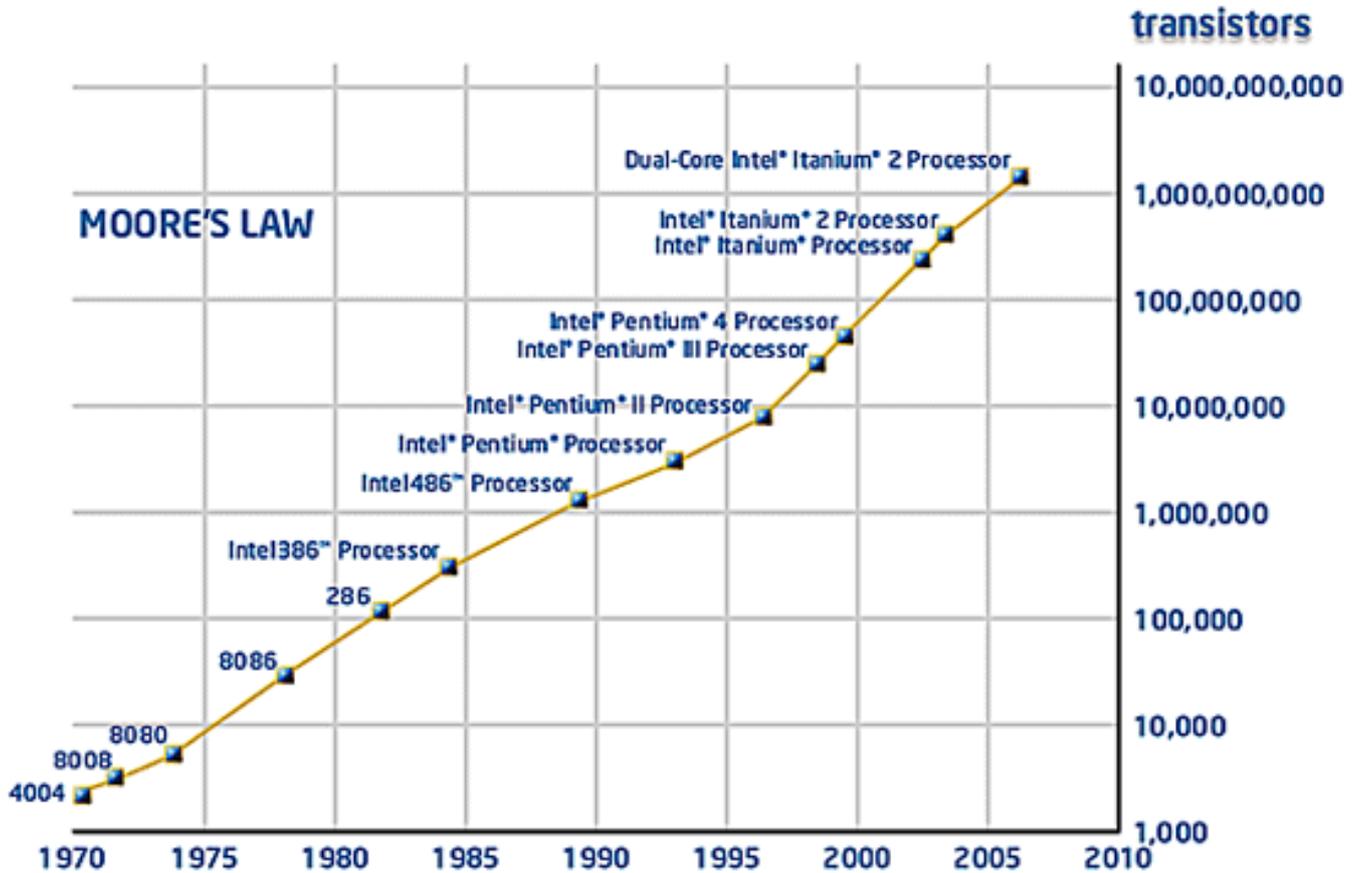
Le Microprocesseur

Le microprocesseur est un circuit intégré numérique polyvalent, piloté par horloge et basé sur une unité de contrôle, une UAL, et des registres. Il traite les données selon des instructions stockées dans sa mémoire et fournit des résultats en sortie.



La Loi de Moore : ACTIVITE

Le nombre de transistors sur une puce électronique devrait être *doublé tous les 2 ans*, pour le même prix de fabrication. Les transistors deviennent donc de plus en plus petits.



nombre de transistors par unité de surface pour un microprocesseur

1. Lire la valeur du nombre de transistors en 1975

- 1975 :

2. Si le nombre de transistors double tous les 2 ans, combien devrait-il y en avoir en 1977, 1979, 1981, 1983, 1985 ?

- 1977 : ...

- 1979 : ...

- 1981 :

- 1983 :

- 1985 :

3. La croissance est-elle (à peu près) régulière pour cette courbe ?

4. Conclure : la loi de Moore est-elle vérifiée entre 1970 et 2010 ?

Exercice 1 : Portes logiques

Donner les tables de vérité pour les portes logiques courantes : NOT, AND, OR, NOR

Exercice 2 : Circuits logiques

1. Pour le schéma 1 ci-dessous : Pourquoi la lampe est-elle éteinte ?

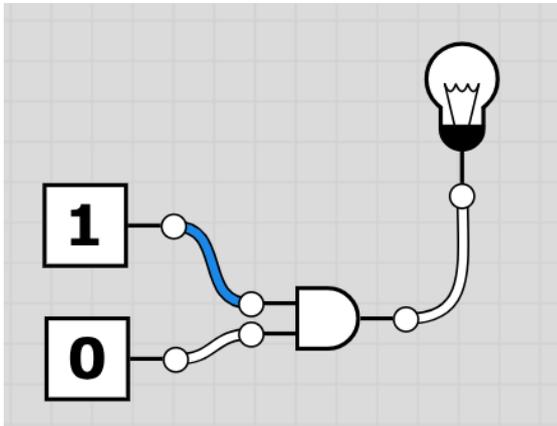


Schéma 1

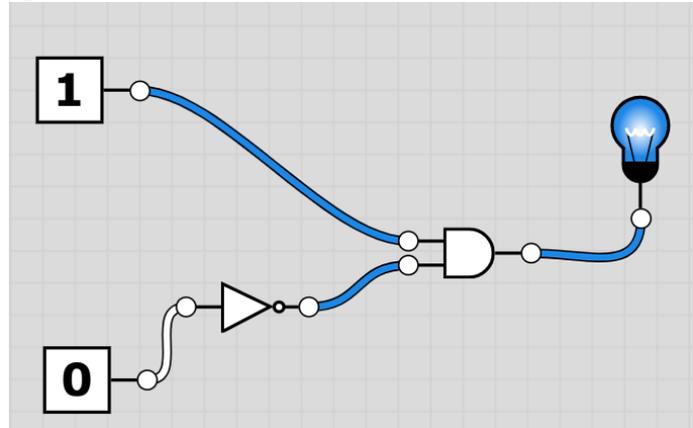
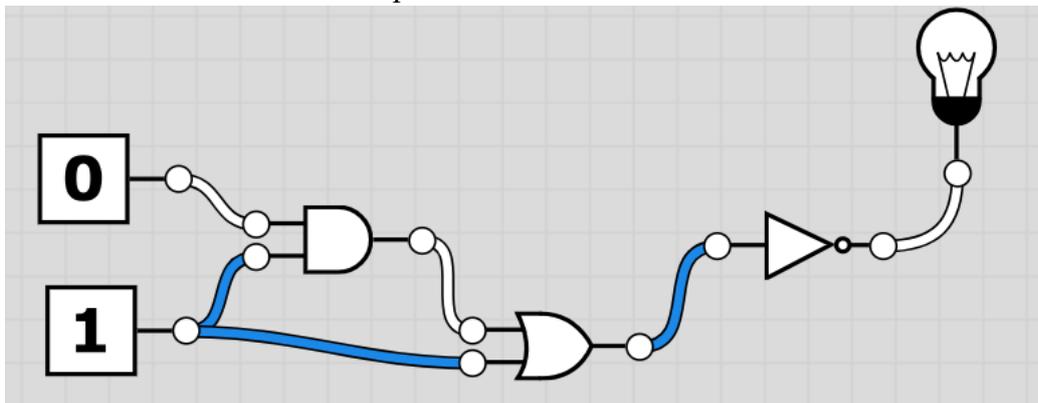


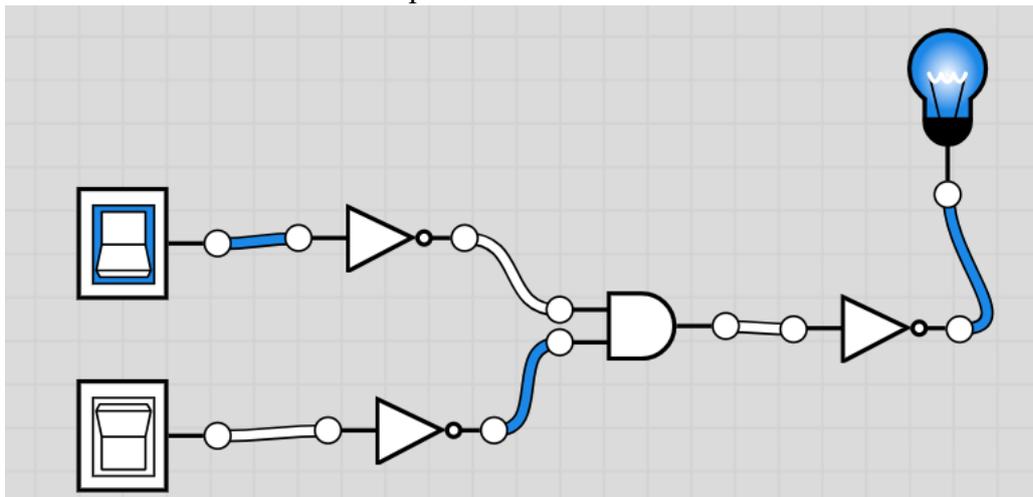
Schéma 2

2. Pour le schéma 2, pourquoi la lampe est-elle allumée ?

3. Pour le circuit suivant : la lampe sera-t-elle allumée / éteinte ?



4. Donner la table de vérité du dispositif suivant :



Exercice 3 : Porte logique XOR

La porte logique XOR, qui est un OU EXCLUSIF, donne un état logique True lorsque l'une de ses 2 entrées seulement est à l'état True, l'autre est False. Ainsi, si l'état logique des 2 entrées est True, la porte logique XOR donne False. Elle diffère alors sur ce point de la porte logique OR.

1. Donner la table de vérité de la porte logique XOR

2. La fonction logique XOR peut aussi s'écrire : $OR(AND(not(x), y), AND(x, not(y)))$. Quelle association de portes logiques va donner la porte logique OU EXCLUSIF, XOR ? Faire un schéma.

Exercice 4 : Composants d'un ordinateur

On pourra s'aider pour cet exercice des pages du site :

https://portail.lyc-la-martiniere-diderot.ac-lyon.fr/srv1/co/ordinateur_cm_composants.html

- Décrire un ordinateur en donnant sa composition : microprocesseur, horloge, mémoire vive (RAM), mémoires mortes (ROM), disque dur, carte graphique, cartes d'extension (son, réseau), connecteurs d'entrée/sortie (usb, ethernet, ...), carte Wifi, carte Bluetooth, clavier, souris, écran, enceintes, imprimante.
Structurer votre réponse entre : composants de la carte mère / périphériques de sortie / périphériques d'entrée.
- Quelle distinction faites-vous entre les différentes mémoires ? Compléter le tableau.

| mémoire | Registres | Disque dur | RAM | Mémoire morte |
|---------|-----------|------------|-----|---------------|
| usage | | | | |

