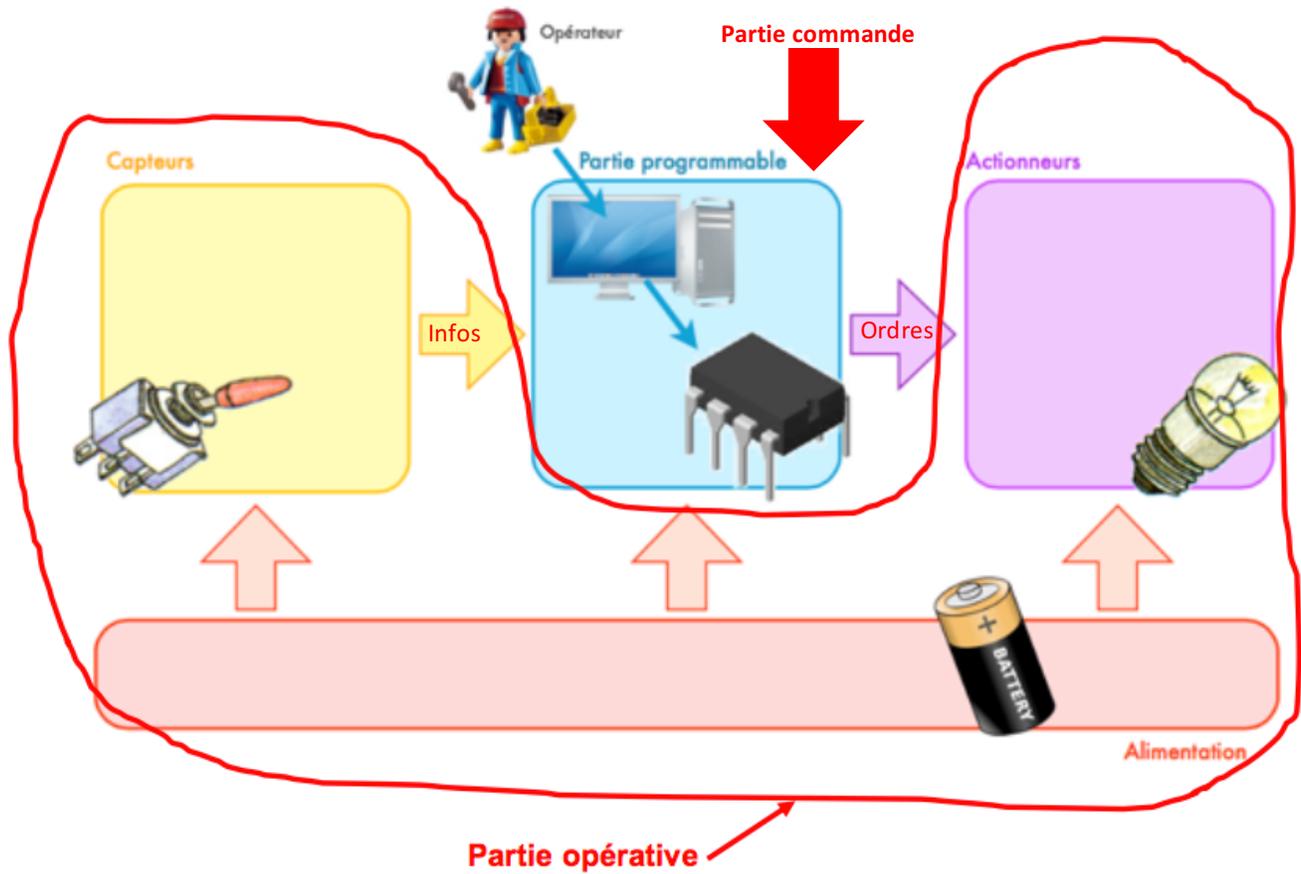


# Synthèse programmation

Schéma d'un système automatisé



## Capteurs



Les **capteurs** mesurent une grandeur physique (présence, position, température, lumière, ...) et la convertissent en **information électrique**. Ils rendent compte de l'état du système à un moment donné.

L'information est délivrée sous la forme d'un **signal** qui sera exploité par la chaîne d'information.

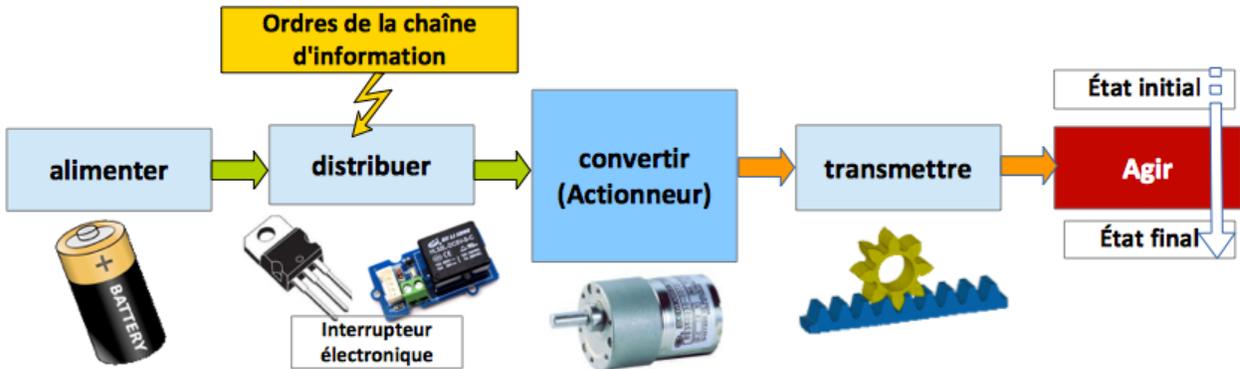


## Exemples de capteurs :

capteur de T°C	capteur de luminosité	contact de porte	détecteur de présence	contact de position	capteur éolien



Un **actionneur** est un élément de la chaîne d'énergie qui produit une action (mouvement, bruit, lumière, ...) après un événement de la chaîne d'information.



## Exemples d'actionneurs : DEL, Lampe, Moteur électrique, Résistance...

### Algorithmes et programmes



Un **programme** informatique est une suite d'instructions déterminées par l'Informaticien pour répondre à un problème (jeux, application, système réel, ...). Il est mis au point, testé puis corrigé avant d'être mémorisé puis traité par un **microcontrôleur**.

Un programme peut être exprimé sous différentes formes :

Langage naturel ou Algorithme	Logigramme ou Langage graphique	Code
Allumer la DEL entrée 2 Attendre 1 seconde Eteindre la DEL entrée 2 Attendre 1 seconde Allumer la DEL entrée 2 Attendre 1 seconde Eteindre la DEL entrée 2 Attendre 1 seconde ...		<pre>                     void setup(){                     pinMode(2,OUTPUT);                     digitalWrite(2,1);                     delay(1000*1);                     pinMode(2,OUTPUT);                     digitalWrite(2,0);                     delay(1000*1);                     pinMode(2,OUTPUT);                     digitalWrite(2,1);                     delay(1000*1);                     pinMode(2,OUTPUT);                     digitalWrite(2,0);                     delay(1000*1);                     }                 </pre>

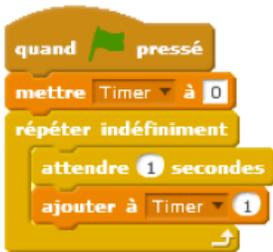
Ces différentes formes de programmes servent à faciliter le travail du programmeur. Elles seront ensuite traduites en langage compréhensible par le microcontrôleur, « 0 » et « 1 » : le code **binaire**.

Un **algorithme** est une suite d'instructions qui permet de résoudre un problème et d'obtenir rapidement un résultat. Il est écrit à la main ou à l'aide d'un logiciel dans un langage compréhensible par tous. Il sert à préparer l'écriture d'un programme informatique

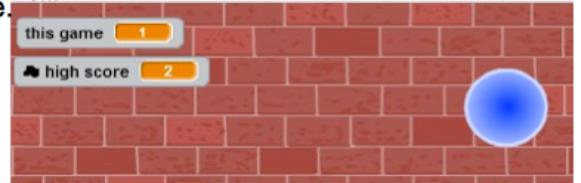
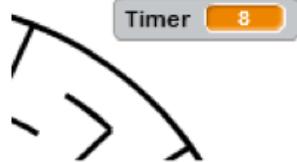
## Variables



Une **variable** est une donnée (information) associée à un nom. Elle est mémorisée et elle peut changer dans le temps, lors de l'exécution du programme.



Exemple : timer



Exemple : score et meilleur score pour un jeu

## Boucles



Lorsque des instructions sont répétées, on utilise des **boucles** pour optimiser le programme.

**Exemple de boucles : TANT QUE, JUSQU'À, REPETER ...**

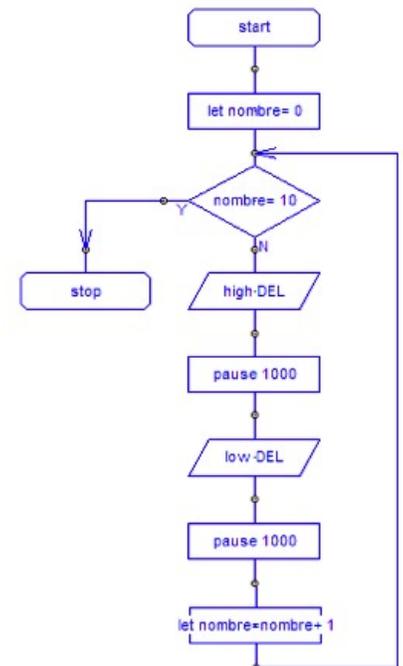


Il est possible d'imbriquer plusieurs boucles les unes dans les autres pour répondre au problème.



Programme en « scratch »

Exemple d'une DEL qui clignote 10 fois



Programme sous forme d'organigramme

## Déclenchement d'actions par évènements



L'enchaînement des opérations et le **déclenchement d'actions** se fait toujours par un **évènement** :

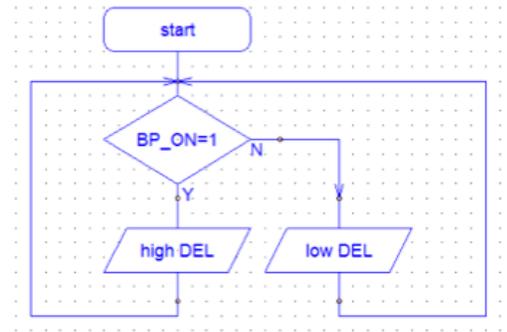
- interne au programme (début programme, variable, ...)
- externe au programme (capteur, touche du clavier, ...)

On écrit l'algorithme sous la forme de « blocs » :

**SI ...**

**ALORS ...**

**SINON ...**

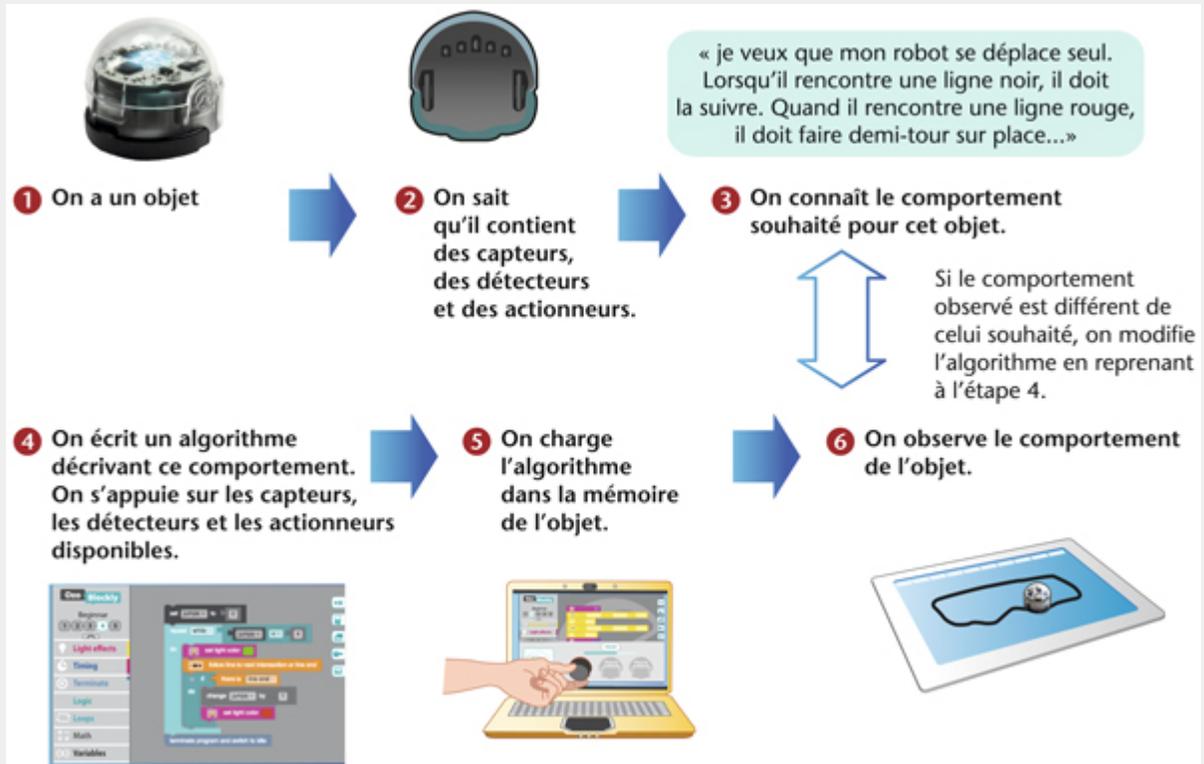


Complétez à l'aide des mots clés suivants : variables, systèmes embarqués, algorithme, détecteurs, boucles

- ✓ De nos jours, les objets doivent être capables d'évoluer dans leur environnement et de réfléchir seuls. C'est notamment le cas pour les .....
- ✓ Pour cela, ils utilisent toutes les informations qu'ils reçoivent de leurs capteurs et ....., et les traitent afin de savoir comment réagir.
- ✓ Ce traitement est réalisé par le microprocesseur. Il s'appuie sur un ....., qui est une série d'instructions que le concepteur de l'objet lui a mise en mémoire.
- ✓ Dans les algorithmes, on peut utiliser des ....., des instructions conditionnelles, et faire des calculs avec des .....informatiques. Ainsi, le comportement de l'objet peut être très riche et complexe.
- ✓ On peut aussi utiliser des algorithmes pour que les objets attendent des signaux extérieurs. Grâce à cette possibilité, on peut modifier en temps réel leur comportement.

# Schéma bilan

## Processus d'enregistrement et d'exécution d'un programme



	<b>TECHNOLOGIE</b> <i>Ce que je dois retenir</i>	<b>L'ALGORITHME</b> <b>ORGANIGRAMME OU LOGIGRAMME</b>	<b>CYCLE</b> <b>4</b>
CT 1.3 – CT 2.5 – CT 2.7 DIC 1.5	Imaginer des solutions pour produire des objets et des éléments de programmes informatiques en réponse au besoin.		
CT 3.1 OTSCIS 2.1	Exprimer sa pensée à l'aide d'outils de description adaptés : croquis, schémas, graphes, diagrammes, tableaux.		
CT 4.2 – CT 5.5 IP 2.3	Écrire un programme dans lequel des actions sont déclenchées par des événements extérieurs.		

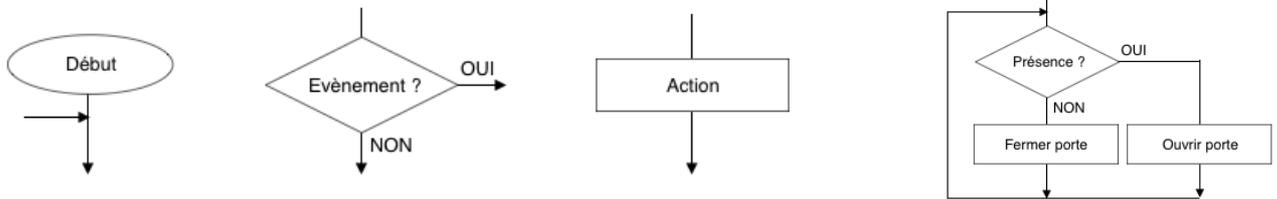
### Symboles de base



Un algorithme est une suite d'instructions précises et structurées qui décrit la manière dont on résout un problème.

Cette description peut être textuelle (si, alors, sinon, tant que ...) ou graphique (appelé également organigramme ou logigramme).

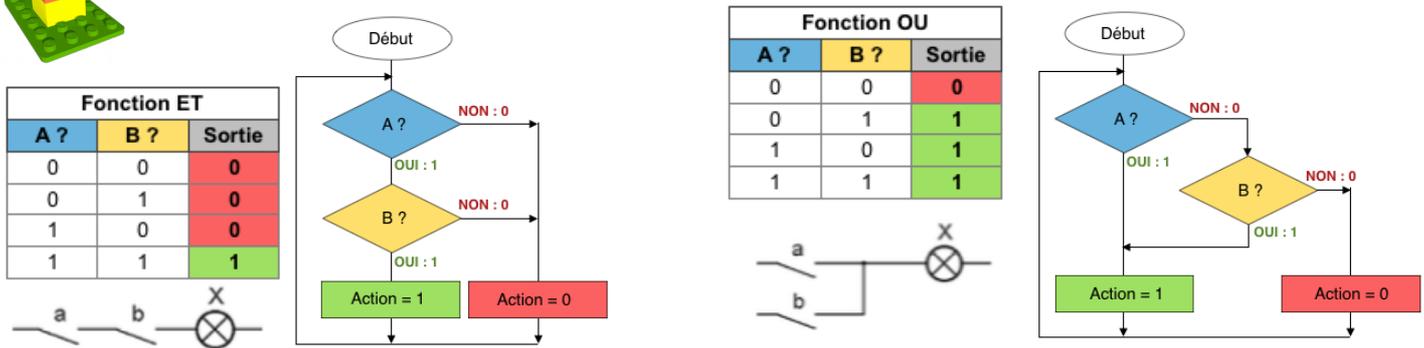
Dans ce cas des normes d'écritures sont à respecter :



**Début**  
**Si** Présence  
**Alors** ouvrir porte  
**Sinon** fermer porte  
**Retour** au début

### Fonctions ET et OU

L'utilisation des fonctions ET et OU sont essentielles pour présenter correctement une solution.



### Algorithme et gestion des sous-problèmes



L'utilisation des sous-problèmes est idéale pour une meilleure lisibilité, pour alléger l'algorithme lors de succession d'actions identiques, pour faciliter le travail en collaboration, pour faciliter une recherche d'erreur (test individuel des sous-problèmes).

