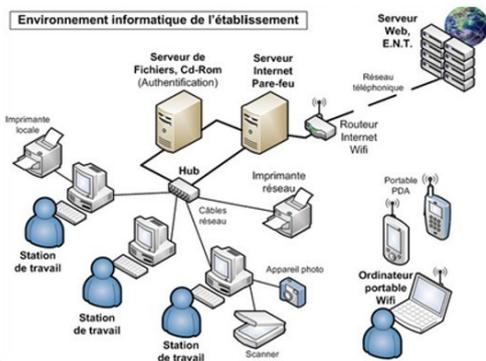


Connaissance : Composants d'un réseau, architecture d'un réseau local.

Pour pouvoir se servir du **matériel informatique** du **réseau local** du collège, il faut en **comprendre son architecture** et son **fonctionnement**. Un réseau informatique est un ensemble d'ordinateurs et de périphériques reliés entre eux pour partager des informations et accéder à des services.
Exemple: Internet...



Le serveur : c'est un ordinateur qui est choisi pour organiser l'ensemble du réseau. Il gère l'accès aux ressources et aux périphériques et les connexions des différents utilisateurs.

Le poste client : c'est un ordinateur connecté au réseau par l'intermédiaire d'une carte réseau (avec ou sans fils) qui utilise les moyens informatiques partagés.



Un **Environnement Numérique de Travail** est un espace numérique de travail personnalisable. ARGOS est L'ENT de l'Académie de Bordeaux.

Un **réseau** est un ensemble d'**équipements électroniques** (ordinateurs, imprimantes, scanners, modems, routeurs, commutateurs...) **interconnectés** avec des **câbles réseaux** ou avec des technologies **sans fils** (wifi, bluetooth...) et capables de **communiquer des informations** et de **partager des périphériques**.

Connaissance : Moyens de connexion d'un moyen informatique.

Pour que les composants du réseau **communiquent entre eux**, il faut des **moyens de connexion** : câbles électriques, transmissions sans fil.



Le **câble Ethernet** est le type de câble le plus utilisé pour connecter des ordinateurs entre eux dans un réseau local (RJ45). Il relie généralement un ordinateur personnel à un routeur avec des prises RJ45.

Une **fibre optique** est un fil en verre ou en plastique très fin qui a la propriété d'être un conducteur de la lumière et sert dans la transmission de données et de lumière. Elle est utilisée pour connecter les serveurs et les Hub pour sa rapidité de transmission du signal.



Le Wi-Fi est le moyen de transmission de données sans fil par ondes radios le plus utilisé. Sa portée ne peut pas dépasser les 200 mètres en espace ouvert et sa vitesse de débit théorique est de plus de 100 mégabits par seconde.

L'infrarouge est un autre moyen de transmission des données sans fil, qui exploite la lumière. (Très utilisé pour les télécommandes)



Le Bluetooth utilise la diffusion d'ondes radio entre les équipements électroniques. Sa portée est de 20 mètres.



Le Li-fi utilise le spectre optique à l'aide d'une LED capable de transmettre des données numériques par la lumière.



Les **composants d'un réseau informatique** nécessitent d'être **connectés**. De nombreux moyens matériels : **câbles, fibre optique...**, et immatériels : **WiFi, Bluetooth, LiFi, infra rouge...** permettent de réaliser ces connexions.

Connaissance : Notion de protocole, d'organisation de protocoles en couche, d'algorithme de routage.

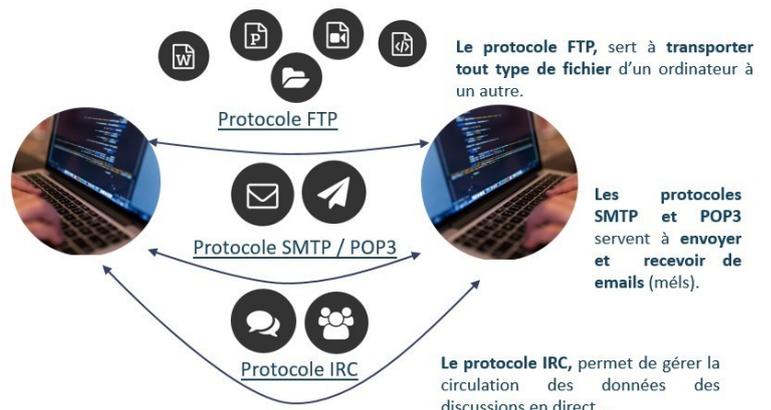
Notion de protocole :

Internet, réseau mondial, utilise plusieurs protocoles pour assurer les divers types d'échanges et la communication entre tous les ordinateurs et serveurs qui sont connectés.

Les protocoles sont donc des langages de communication.



Le protocole HTTP, permet de transporter des données, notamment des pages web, qui sont affichées (décodées) par un navigateur web sur l'ordinateur de l'utilisateur.



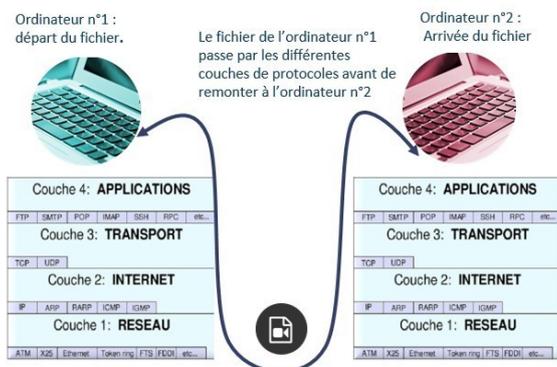
Les protocoles sont des langages de communication permettant d'assurer les échanges de divers types de données et la communication entre les ordinateurs et serveurs sur le réseau Internet. Le plus utilisé est le protocole HTTP permettant de transporter les pages web.

Connaissance : Notion de protocole, d'organisation de protocoles en couche, d'algorithme de routage.

Organisation de protocoles en couche :

Les protocoles sont structurés en couches. Chaque couche s'occupe d'apporter un plus, permettant la transmission de données, et de fournir des éléments aux couches supérieures.

Exemple : trajet parcouru par un fichier sur le réseau internet.



Couche 4: Applications
Transfert des fichiers des applications s'exécutant sur l'ordinateur.

Couche 3: Transport
Gestion du transfert des données vers le destinataire.

Couche 2: Internet
Chemin parcouru

Couche 1: Réseau
Description physique du transport des données à travers des câbles, des hubs...

Algorithme de routage:

Le routage est le mécanisme par lequel des chemins sont sélectionnés dans un réseau pour acheminer les données en plusieurs paquets d'un expéditeur jusqu'à un ou plusieurs destinataires. Le routage est une tâche exécutée dans de nombreux réseaux, tels que le réseau téléphonique, les réseaux de données électroniques comme Internet, et les réseaux de transports



Visualisation des multiples chemins à travers Internet



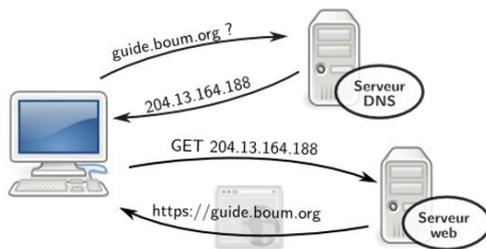
Le but d'un algorithme de routage est de trouver le meilleur chemin possible entre la source et le destinataire des informations pour satisfaire les différents critères de qualité imposés (Débits, taux de perte...)

La diffusion des informations sur internet se fait à travers des protocoles en couches qui permettent d'organiser tous les transferts... Les algorithmes de routages permettront de trouver le meilleur chemin sur le réseau pour acheminer les information le plus rapidement possible par exemple.

Connaissance : Internet

C'est le **réseau informatique mondial** accessible au public, composé de **millions de réseaux interconnectés**, aussi bien **publics** que **privés**.
Le web (raccourci de « world wide web », traduit en français par toile mondiale) correspond à une des applications utilisant ce réseau, la principale.
Le web est un **système de publication** et de **consultation de documents** (textes, sons, images) faisant appel aux **techniques de l'hypertexte** qui utilisent des renvois permettant de passer directement d'une partie d'un document à une autre, ou d'un document à d'autres documents.

L'information est transmise par **internet** grâce à un ensemble standardisé de protocoles de transfert de données, qui permet l'élaboration d'applications et de services variés.

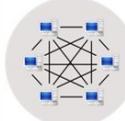


Exemple:

L'ordinateur se connecte au serveur DNS pour connaître l'adresse IP de la page d'accueil d'un site. Il lui envoie une requête qui signifie « envoie-moi la page d'accueil du site web guide.boum.org ». Les paquets qui véhiculent la demande partent et passent alors par la « box » pour arriver au routeur du fournisseur d'accès. La réponse revient et l'ordinateur peut maintenant s'adresser au serveur Web pour rapatrier la page d'accueil souhaitée.



Le courrier électronique (courriel, e-mail, mail) est un service de transmission de messages écrits et de documents envoyés électroniquement via un réseau informatique dans la boîte aux lettres électronique d'un destinataire choisi par l'émetteur.



Le pair à pair (peer-to-peer, abrégé « P2P ») est un modèle de réseau informatique proche du modèle client-serveur mais où chaque client est aussi un serveur.



Le World Wide Web (WWW), communément appelé le Web, et parfois la Toile, est un système hypertexte public fonctionnant sur Internet. Le Web permet de consulter, avec un navigateur, des pages accessibles sur des sites.

On trouve l'origine d'**Internet** dans **Arpanet**, le premier réseau à transfert de paquets de données développé aux États-Unis en 1972 qui permet l'acheminement de proche en proche de **messages découpés en paquets indépendants**. L'**internet** est aujourd'hui un gigantesque **réseau composé de millions de réseaux publics et privés**, universitaires, commerciaux, gouvernementaux...

Connaissance : Notions d'algorithme et de programme

Les **objets connectés** sont souvent programmés pour fonctionner **automatiquement**. Chaque fonction numérique de l'objet connecté peut être assimilée à un « problème » à résoudre. La **résolution d'un problème** par un programmeur peut s'effectuer en **trois étapes** :

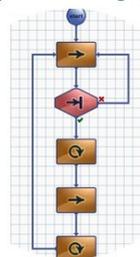
-1- Ecriture d'un algorithme : Suites logique d'opérations ou d'instructions, souvent rédigées sur feuille de papier en utilisant le langage naturel et des mots clés : si, alors, tant que, jusqu'à ...

Exemple : Un robot évitant un obstacle.



~ Si le robot détecte un obstacle avec son capteur de pare-choc, alors tourner à gauche de 90° ; avancer de 10cm puis tourner à droite de 90°.
~ Sinon avancer indéfiniment.

-2- Construction à l'aide d'un logiciel d'une représentation graphique de l'algorithme



Algorithme : organigramme de programmation



Logiciel de représentation graphique par bloc (ou briques) comme Scratch

-3- Traduction de la représentation graphique en langage de programmation qui lui-même sera converti en langage machine (code binaire) que le microprocesseur peut exécuter

```

17 // ... (NPU) ;
18 // ... (10*analogRead(A7) > ...
19 // ... (1000);
20 // ... (100*analogRead(A7)>1070:1;
21 // ... (100*analogRead(A7)>1070:1;
22 // ... (100*analogRead(A7)>1070:1;
23 // ... (100*analogRead(A7)>1070:1;
24 // ... (100*analogRead(A7)>1070:1;
25 // ... (100*analogRead(A7)>1070:1;
26 // ... (100*analogRead(A7)>1070:1;
27 // ... (100*analogRead(A7)>1070:1;
28 // ... (100*analogRead(A7)>1070:1;
29 // ... (100*analogRead(A7)>1070:1;
30 // ... (100*analogRead(A7)>1070:1;
31 // ... (100*analogRead(A7)>1070:1;
32 // ... (100*analogRead(A7)>1070:1;
33 // ... (100*analogRead(A7)>1070:1;
34 // ... (100*analogRead(A7)>1070:1;
35 // ... (100*analogRead(A7)>1070:1;
36 // ... (100*analogRead(A7)>1070:1;
37 // ... (100*analogRead(A7)>1070:1;
38 // ... (100*analogRead(A7)>1070:1;
39 // ... (100*analogRead(A7)>1070:1;
40 // ... (100*analogRead(A7)>1070:1;
41 // ... (100*analogRead(A7)>1070:1;
42 // ... (100*analogRead(A7)>1070:1;
43 // ... (100*analogRead(A7)>1070:1;
44 // ... (100*analogRead(A7)>1070:1;
45 // ... (100*analogRead(A7)>1070:1;
46 // ... (100*analogRead(A7)>1070:1;
47 // ... (100*analogRead(A7)>1070:1;
48 // ... (100*analogRead(A7)>1070:1;
49 // ... (100*analogRead(A7)>1070:1;
50 // ... (100*analogRead(A7)>1070:1;
51 // ... (100*analogRead(A7)>1070:1;
52 // ... (100*analogRead(A7)>1070:1;
53 // ... (100*analogRead(A7)>1070:1;
54 // ... (100*analogRead(A7)>1070:1;
55 // ... (100*analogRead(A7)>1070:1;
56 // ... (100*analogRead(A7)>1070:1;
57 // ... (100*analogRead(A7)>1070:1;
58 // ... (100*analogRead(A7)>1070:1;
59 // ... (100*analogRead(A7)>1070:1;
60 // ... (100*analogRead(A7)>1070:1;
61 // ... (100*analogRead(A7)>1070:1;
62 // ... (100*analogRead(A7)>1070:1;
63 // ... (100*analogRead(A7)>1070:1;
64 // ... (100*analogRead(A7)>1070:1;
65 // ... (100*analogRead(A7)>1070:1;
66 // ... (100*analogRead(A7)>1070:1;
67 // ... (100*analogRead(A7)>1070:1;
68 // ... (100*analogRead(A7)>1070:1;
69 // ... (100*analogRead(A7)>1070:1;
70 // ... (100*analogRead(A7)>1070:1;
71 // ... (100*analogRead(A7)>1070:1;
72 // ... (100*analogRead(A7)>1070:1;
73 // ... (100*analogRead(A7)>1070:1;
74 // ... (100*analogRead(A7)>1070:1;
75 // ... (100*analogRead(A7)>1070:1;
76 // ... (100*analogRead(A7)>1070:1;
77 // ... (100*analogRead(A7)>1070:1;
78 // ... (100*analogRead(A7)>1070:1;
79 // ... (100*analogRead(A7)>1070:1;
80 // ... (100*analogRead(A7)>1070:1;
81 // ... (100*analogRead(A7)>1070:1;
82 // ... (100*analogRead(A7)>1070:1;
83 // ... (100*analogRead(A7)>1070:1;
84 // ... (100*analogRead(A7)>1070:1;
85 // ... (100*analogRead(A7)>1070:1;
86 // ... (100*analogRead(A7)>1070:1;
87 // ... (100*analogRead(A7)>1070:1;
88 // ... (100*analogRead(A7)>1070:1;
89 // ... (100*analogRead(A7)>1070:1;
90 // ... (100*analogRead(A7)>1070:1;
91 // ... (100*analogRead(A7)>1070:1;
92 // ... (100*analogRead(A7)>1070:1;
93 // ... (100*analogRead(A7)>1070:1;
94 // ... (100*analogRead(A7)>1070:1;
95 // ... (100*analogRead(A7)>1070:1;
96 // ... (100*analogRead(A7)>1070:1;
97 // ... (100*analogRead(A7)>1070:1;
98 // ... (100*analogRead(A7)>1070:1;
99 // ... (100*analogRead(A7)>1070:1;
100 // ... (100*analogRead(A7)>1070:1;

```

Programme : lignes de codes en langage C

Pour résoudre un problème, le programmeur commence par **écrire un algorithme** dans lequel il donne des **ordres en fonction de conditions (état des capteurs)**. Puis il construit sur un ordinateur une **représentation graphique de l'algorithme** (Algorithme ou par bloc avec Scratch). Le logiciel va ensuite traduire la représentation graphique en **ligne de code (le programme)** que le système va exécuter.

Connaissance : Notion de variable informatique

5^{ème} 4^{ème} 3^{ème}

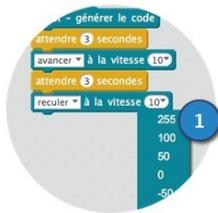
La programmation des **objets connectés** nécessite la gestion de **situations complexes** (déplacements, trajectoires, mesures des capteurs...). Pour résoudre ces problèmes plus « évolués », les programmeurs vont introduire deux types de **variables informatiques** dans leurs algorithmes : les variables dites « **statiques** » et « **dynamiques** ».

Les variables statiques :

les variables statiques sont tout simplement des **valeurs constantes**. Elles sont **stockées** (enregistrées) dans la mémoire de l'objet connecté (comme dans une clé USB).

Exemple : Utilisation de variables statiques pour régler la vitesse d'un robot.

-Cas 1- La plupart du temps, pour les logiciels de représentation graphiques, les variables sont prédéfinies. Le programmeur a le choix entre plusieurs valeurs (1) pour régler la vitesse de son robot.

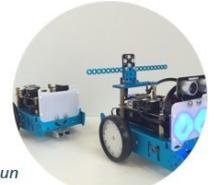


- Cas 2 – Les variables peuvent aussi être créées par le programmeur. Elles porteront un nom précis en fonction des choix du programmeur (exemple : var, B0, B1, vitesse...)

Le programmeur commence par créer la variable et lui donne un nom. Ici il la nomme « vitesse » (2)



Dans l'algorithme, il est ensuite possible d'attribuer des valeurs (3) à la variable vitesse pour choisir la vitesse de déplacement (4) du robot.



Une **variable informatique** est une case mémoire stockant une donnée qui peut être fixe ou varier au cours de l'exécution du programme.

En programmation, les **variables statiques** correspondent à des **valeurs constantes et fixes** que l'on peut utiliser dans les algorithmes pour fixer des valeurs numériques comme pour définir les vitesses de déplacements des robots. Les variables sont **stockées** (enregistrées) dans la **mémoire** de l'objet connecté.

Connaissance : Notion de variable informatique

5^{ème} 4^{ème} 3^{ème}

Les variables dynamiques affectées à des capteurs :

Les capteurs, quand ils réalisent leurs mesures, envoient pour **stockage (enregistrement)** leurs résultats dans des variables. Par exemple, un capteur de luminosité va régulièrement détecter une variation de lumière, un capteur ultrason va mesurer le changement de la distance de l'obstacle et un contacteur sera en position ouverte ou fermée. Etant donné que les mesures des capteurs sont susceptibles de changer dans le temps, les valeurs stockées dans les variables feront de même. Ces variables seront dites **dynamiques**.

Exemple : Utilisation de variable pour gérer l'arrêt d'un robot devant un obstacle.

- La variable « **obstacle** » (1) permettra de stocker la valeur correspondant à la **distance des obstacles** mesurée par le capteur ultrasons
- Remarque : La variable « **route** » (2) pourra stocker la valeur correspondant à la couleur de la route mesurée par le capteur infrarouge.



1- Création d'une variable

- Dans l'algorithme, on va commencer par lire et stocker la valeur mesurée (3) par le capteur ultrason dans la variable.
- Puis on compare (4) la valeur de la variable avec un seuil correspondant à la distance de l'obstacle à laquelle le robot doit s'arrêter.

Ici, la valeur de la variable « **obstacle** » va **varier** quand le robot se déplacera car la boucle est répétée indéfiniment.



2- Utilisation d'une variable

Il est possible de **visualiser** pendant le fonctionnement les valeurs des variables (5) à l'aide des logiciels de programmation graphiques (exemple : Scratch)



Les variables **affectées à des capteurs** sont dites « **dynamiques** » car leurs **valeurs changent dans le temps** en fonction de la variation de la mesure du capteur.

Connaissance : Notion de variable informatique

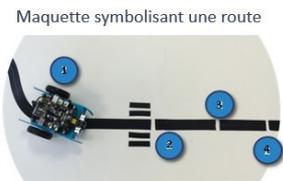
5^{ème} 4^{ème} 3^{ème}

Les variables dynamiques destinées au comptage :

Lors de la résolution de **problèmes complexes**, il peut être nécessaire réaliser des **opérations de comptage** (additions, soustractions...), grâce à des variables. Par exemple, un robot, pour se repérer dans un itinéraire, peut compter les intersections. Ou encore, il peut être nécessaire de compter le nombre de personnes ou d'objets qui passent devant un capteur. La plupart du temps la **variable sera incrémentée** en lui ajoutant une valeur fixée, souvent 1.

Exemple : Utilisation d'une variable pour compter les intersections.

1. Le robot (1) suit la route et passe devant l'intersection n°1 (2), puis croise les autres intersections (3,4).
2. Une variable nommée « intersection » sera incrémentée quand le robot passera à chaque intersection de la maquette. Le robot détectera la rupture de la ligne noire. Ses capteurs verront du blanc à la place de la route noire.



3. Dans le programme associé, la première étape, (5) consiste à remettre la variable à 0 (réinitialiser).

```

mBot - générer le code
attendre jusqu'à le bouton est p
mettre vitesse à 100
mettre intersection à 0
répéter indéfiniment
mettre route à état du suiv
si route = 0 alors
    avancer à la vitesse
    
```

4. Ensuite, quand le capteur mesure une rupture de la ligne noire (6) correspondant à une intersection, alors on ajoute 1 à la variable « intersection » (7).

```

route = 3 alors
    ajouter à Intersection 1
    avancer à la vitesse 100
    attendre 0.3 secondes
si Intersection = 3 alors
    tourner à gauche à la vitesse 100
    attendre 0.7 secondes
    reculer à la vitesse 100
    attendre 1 secondes
    
```

5. Enfin, on peut comparer (8) la variable « intersection » à un nombre précis de carrefour (ici 3) pour donner l'ordre de tourner.

Il est possible d'utiliser des **variables destinées au comptage**. Elles seront en général **incrémentées d'une valeur fixe** (souvent 1) à chaque opération de comptage.

Connaissance : Séquences d'instructions, boucles

L'**algorithme** réalisé par le programmeur va permettre de répondre au problème posé (pour rendre **les objets connectés plus « intelligents »** par exemple). Il y a plusieurs « degrés de complexité » de programmation. Les **instructions** peuvent être simplement indiquées et **exécutées** une seule fois ou **répétées en boucle**. Les instructions peuvent aussi être **conditionnées** par l'apparition d'un événement **détecté par un capteur**.

Séquences d'instructions :

Les actions d'un système (exemple : robot) peuvent être déclenchés en **séquences d'instructions** sans conditions préalables : avancer, tourner à gauche, à droite, reculer... Les ordres sont enchaînés les uns à la suite des autres.



```

mBot - générer le code
joue la note C4 beat un demi
envoyer à un mBot le message Aloha
attendre 3 secondes
avancer à la vitesse 100
attendre 3 secondes
tourner à droite à la vitesse 50
attendre 2 secondes
arrêter le son
    
```

Boucles :

Les instructions peuvent aussi être **répétées en boucles** un certain nombre de fois et passer à une autre action ou **répétées indéfiniment**. Le système exécute alors le programme et ne s'arrête que lorsque l'opérateur stoppe l'exécution.



```

mBot - générer le code
répéter 4 fois
    avancer à la vitesse 100
    attendre 1 secondes
    tourner à droite à la vitesse 100
    attendre 1.5 secondes
    avancer à la vitesse 0
    
```

```

mBot - générer le code
répéter indéfiniment
    avancer à la vitesse 100
    attendre 1 secondes
    tourner à droite à la vitesse 100
    attendre 1.5 secondes
    avancer à la vitesse 0
    
```

Les **instructions** d'un **algorithme** peuvent être déclenchées en **séquences** : les ordres étant enchaînés les uns à la suite des autres sans conditions préalables (avancer, tourner...) et/ou **répétées en boucle** un nombre de fois précis, indéfiniment ou en fonction des événements détectés par les capteurs.

Connaissance : Instructions conditionnelles, déclenchement d'une action par un évènement

Instructions conditionnelles : Si – Alors – Sinon :
Dans un algorithme, les instructions peuvent être soumises à une condition pour s'exécuter.



Exemple : S'arrêter devant un piéton à une certaine distance
- Si le capteur d'obstacle du robot détecte un piéton à une certaine distance, **alors** arrêter les moteurs.
- **Sinon** avancer à la vitesse de 100.

Déclenchement d'une action par un évènement :

Les actions peuvent être déclenchées par un évènement, par exemple :

- La **variation** d'une **grandeur physique** (Changement de luminosité, de chaleur, de couleur...)
- Le **déplacement** d'un objet mesuré par un **capteur** du système.

Exemple : Dans une maison, s'il fait nuit, alors allumer les lumières.

- L'évènement est la **variation de la luminosité**.
- Le **capteur** (LDR) mesure la **quantité de lumière** et envoie cette valeur pour stockage dans une variable (« varA » ou « luminosité »).
- Puis l'algorithme **compare** cette variable avec un **seuil** (fixé ici à 150) correspondant à la **nuit**.
- Si la valeur mesurée est en **dessous** du seuil, alors on donne l'**ordre** (action) **d'allumer** les lumières.



Dans un **algorithme**, l'exécution des instructions peut être **conditionnée** par l'apparition d'un **évènement**. Dans ce cas, l'instruction s'exécute **SI** l'évènement a lieu. **SINON** une instruction différente se réalisera.

Connaissance : Systèmes embarqués

Un **système embarqué** est un système **électronique et informatique autonome** capable souvent de **réagir en « temps réel »** et de réaliser des **tâches précises** (déplacements, préhension...). Il est intégré dans un objet et permet, à partir d'un **ordinateur** (microprocesseur, mémoires, carte mère, alimentation électrique autonome...) de ses **propres capteurs, ses actionneurs**, et d'un **logiciel** stocké dans sa **mémoire**, d'assurer un fonctionnement autonome.



Le **robot programmable** exécute le **programme** qui a été téléchargé dans sa **mémoire**. Il peut suivre une route, éviter des obstacles, jouer de la musique...

Le **robot aspirateur** est capable de nettoyer les sols et « **mémoriser** » la taille des pièces et obstacles rencontrés sur son passage afin de les éviter et d'optimiser les temps de parcours.



Le **drone** est capable de corriger sa position, de se **stabiliser** en restant en vol stationnaire. Il **exécute** aussi en temps réel les ordres envoyés par la radiocommande.

La **voiture sans conducteur** est capable de transporter des usagers sur n'importe quelle route en toute autonomie et sécurité.



Le **système embarqué** permet aux objets de réaliser des **tâches prédéfinies à l'avance** (intelligence artificielle faible) ou de rendre l'objet plus **autonome**, capable « d'apprendre » et de modifier son programme interne (intelligence artificielle forte) comme un robot aspirateur. Il se compose d'un **programme** stocké dans la **mémoire** d'un **ordinateur embarqué**.

Connaissance : Forme du signal.

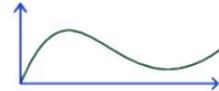
Les **objets connectés transmettent des signaux** : soit en **interne** (lors de la communication entre les capteurs, les actionneurs et l'interface) soit vers l'**extérieur** du système (en communiquant avec d'autres systèmes ou avec l'utilisateur).

Forme des signaux transmis par les capteurs :

Signal numérique : Un signal est dit **numérique** s'il ne peut prendre que deux valeurs : **0** ou **1**. Exemple un contact électrique ouvert ou fermé.



Signal Analogique : Un signal est dit **analogique** si la valeur mesurée **varie** de façon continue dans le temps. Exemple la luminosité, la température, la distance d'un obstacle...



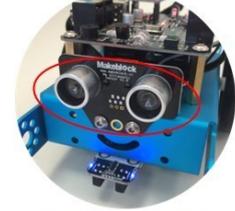
Le microinterrupteur à galet détecte si la porte est **ouverte** ou **fermée**



Contacteur de pare-choc, détecte si le pare-choc est **enfoncé** ou **relâché**



LDR (capteur de lumière) capte la variation de luminosité tout au long de la journée.



Capteur Ultrasons : capte la distance de l'obstacle.

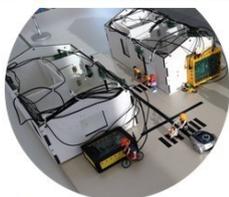
Les **capteurs** des systèmes transmettent deux types de signaux : Les **signaux numériques** qui ne prennent que deux valeurs logiques (**0** ou **1**) et les **signaux analogiques** qui varient constamment et qui peuvent prendre **une grande quantité de valeurs**.

Connaissance : Transmission du signal

Transmission du signal :

Les signaux des objets connectés sont transmis en utilisant différents supports matériels ou immatériels.

Matériels : s'ils utilisent des câbles électriques ou un autre réseau de câbles comme la fibre optique.



Les **câbles** permettent de transmettre le signal des capteurs à l'interface programmable (ici, des boîtiers Picaxes).

Les **fibres optiques** transmettent le signal sous forme d'impulsions lumineuses.



Les **câbles électriques** en cuivre transmettent le signal sous forme d'impulsions électriques ou sous forme analogique.

Immatériels : s'ils utilisent des ondes comme par exemple une télécommande infrarouge ou un Smartphone via wifi ou Bluetooth.



Le **faisceau lumineux (infrarouge)** d'une télécommande permet de piloter un robot.



Les **ondes radio, wifi ou Bluetooth**, peuvent piloter un robot depuis un Smartphone ou ordinateur.



Pour **transmettre un signal** (une information), on utilise :

- Un **signal électrique**, quand il est possible de placer un **fil conducteur**, c'est la **solution la moins coûteuse**.
- Un **signal lumineux**, grâce à de la **fibre optique**, **solution coûteuse** mais la transmission de données s'effectue avec un **très grand débit**.
- Les **ondes radio** sur de **grandes distances** ou pour traverser des obstacles : **Wifi** (100m), **Bluetooth** (10 à 20m), **3G/4G** (jusqu'à 18km)...
- Les **ondes infrarouges** sur de **petites distances** et en l'**absence d'obstacle** : souris informatique sans fil, télécommande, casque d'écoute sans fil (portée 12m).

Connaissance : Capteur

Dans les **systèmes automatisés**, on trouve des **interfaces** (associées à la partie commande du système) qui font le lien entre les **capteurs** (acquisition du signal) et les **actionneurs** qui réalisent l'action (transformation d'énergie).

Capteur :

Les capteurs se trouvent à « l'entrée » de la **chaîne d'information** : fonction **acquérir**. Ils reçoivent les informations extérieures au système y compris les consignes des utilisateurs et les transmettent à l'**interface** qui va **traiter** les informations.



Capteur Ultrason :
détection d'obstacle

Capteur infrarouge : détection des personnes



Le micro-rupteur à galet est contacté quand la porte s'ouvre et « appuie sur le galet »

Capteur de mouvement infrarouge



Un **Capteur** réalise l'**acquisition** d'une **grandeur physique** (température, luminosité, présence, distance, ...) qu'il **transforme** en un **signal logique, analogique ou numérique** afin qu'il puisse être traité par la **partie commande** (ordinateur + programme) du système.

Connaissance : Actionneur

L'**actionneur** : Il se situe dans la **chaîne d'énergie** pour assurer la fonction **conversion**. Il **transforme l'énergie d'entrée** pour réaliser « l'action » commandée depuis la chaîne d'information.



Motoréducteur :

Le **moteur** (de la roue) va convertir le courant **électrique** des batteries en **énergie mécanique** (mouvement de rotation)

Le **réducteur** (les engrenages) vont ensuite modifier la vitesse de rotation de l'axe du moteur, en général pour la ralentir et donner plus de force à la roue du robot.



Le **moteur électrique** permet de faire tourner les pales du **drone**. Le moteur va convertir l'électricité en énergie mécanique correspondant à la rotation de l'axe du moteur. (...) et entraîner les engrenages des pales. L'action mécanique des pales (poussée) sur l'air permet au drone de s'élever.

La **LED** permet de convertir l'énergie électrique en lumière



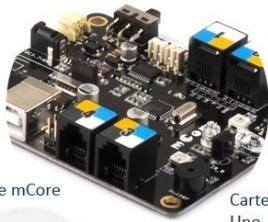
Le **Servomoteur**, ici branché sur le boîtier Groomy, est un système motorisé qui converti l'énergie électrique en une rotation, dont l'angle est choisi précisément et qui peut le maintenir.

L'**actionneur** est l'organe de la **chaîne d'énergie** qui va réaliser la **conversion de l'énergie** pour réaliser **une action**. Par exemples, le moteur transforme l'énergie électrique en énergie mécanique pour assurer un mouvement, la LED transforme l'énergie électrique en lumière pour signaler un événement.

Connaissance : Interface

L'**interface** est associée à la partie commande du système. D'un côté les **capteurs** sont connectés aux **entrées** de l'interface et d'un autre les **actionneurs** aux **sorties**. L'interface permet en partie d'assurer la fonction **communication** de la chaîne d'information ainsi qu'une partie de la fonction **transfert**. Concrètement, elle reçoit les **informations des capteurs** et **transmet les ordres aux actionneurs**.

Interfaces capteurs/partie commande/actionneurs



Carte mCore



Carte Arduino Uno



Boitier Groomy



Boitier Picaxe



Interface sur une carte programmable de type « Arduino » ou compatibles. Les capteurs et les actionneurs sont câblés directement sur la carte qui sert aussi d'interface.

Boitiers de commande programmables. Les cartes de la partie commande sont protégées par un boîtier. Les capteurs et les actionneurs sont câblés avec des prises « jacks » sur des entrées / sorties identifiés sur le boîtier interface.

Interface homme/machine



Boîtier (pupitre) de commande du thermostat du chauffage d'une maison (centrale d'ambiance) assurant l'**interface homme-machine**.

Une **interface** permet d'établir une **communication** et **assurer le dialogue** entre deux éléments :

- Soit entre **l'homme et le système**, « interface homme-machine ». Elle va permettre à **l'utilisateur de communiquer avec le système grâce à un pupitre**.
- Soit entre la **chaîne d'information (capteurs)** et la **chaîne d'énergie (actionneurs)**.