



Lycée Julien Wittmer

SNT

OBJETS CONNECTES



et Informatique embarquée

1

Les repères historiques

p. 2

2

Architecture des systèmes embarqués

p. 2

3

Les IHM

p. 4

4

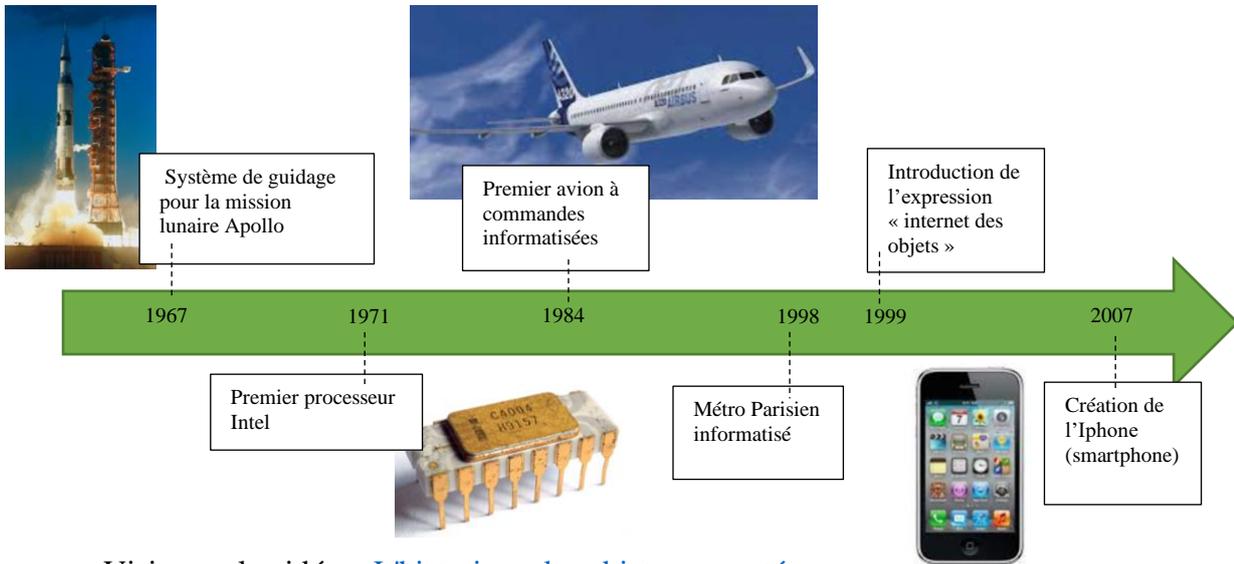
Les objets connectés

p. 6





Repères historiques



- Visionner la vidéo : [L'historique des objets connectés](#)

1. Relever les informations suivantes :

a. Quelle invention permet de miniaturiser l'informatique ?

.....

b. Quel est le rôle d'un processeur ?

.....

c. Quel est le premier objet à avoir été connecté ?

.....



2. Cours : Récupérer et traiter l'information.

Dans les systèmes informatiques embarqués l'information est le plus souvent récupérée par des **capteurs**.



Un **capteur** est un dispositif qui permet de mesurer des grandeurs du monde réel et éventuellement de les numériser pour pouvoir être utilisées par un ordinateur.

Exemples : Capteur de température, de luminosité, de pression, ...

Les systèmes vont ensuite mémoriser et traiter ces informations à l'aide d'algorithmes.

Puis des **actionneurs** permettent ensuite au système d'agir à son tour sur le monde physique.

Les actionneurs réalisent des actions.

Exemples : Les moteurs, les lampes, les buzers, ...





2

Application 1 – L’architecture de l’informatique embarquée

NOVICE LEVEL

- Visionner la vidéo : [Des robots dans nos maison ?](#)

1. Quels sont les trois composants indispensables à un objet connecté ?

.....

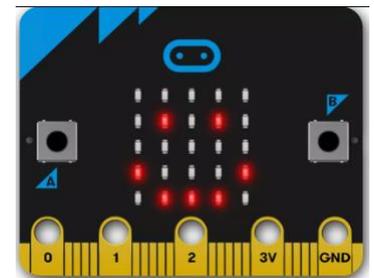
2. Quel risque de sécurité peut présenter un objet connecté ?

.....

Le microcontrôleur possède à la fois des capteurs, des actionneurs et la possibilité d’être programmé.

- A l’aide de votre microcontrôleur et du logiciel Mu flasher le programme suivant :

```
1 from microbit import *
2
3 display.show(Image.HAPPY)
```



3. Quel est l’actionneur utilisé ?

.....

- Modifier le programme précédent pour qu’il affiche successivement des images différentes.

Aide : La fonction sleep() : `sleep(2000)` permet de mettre le programme en pause un certain temps (en ms).

[Liste des images possibles](#)



2

Application 2 – L’utilisation d’un capteur

INTERMEDIATE LEVEL

Le microcontrôleur est équipé d’un accéléromètre. Il peut ainsi détecter les différents mouvements faits par la carte.

L’accéléromètre est capable de reconnaître les positions ou mouvements suivants :

haut, bas, gauche, droite, face visible, face cachée, chute libre, 3g, 6g, 8g (ce sont des accélérations), remué.

Pour obtenir le mouvement en cours, utilisez la méthode `accelerometer.current_gesture`.

Son résultat va être l’un des mouvements nommés énumérés ci-dessus (mais en anglais)

- Essayer le programme suivant :

```
1 from microbit import *
2
3 while (True):
4     message = accelerometer.current_gesture()
5     display.scroll(message)
6
```

1. Que fait la fonction `display.scroll()` ?

.....

2. Compléter le tableau suivant:

Action effectuée	Carte verticale vers le haut	Carte verticale vers le bas	Carte verticale tournée à droite	Carte verticale tournée à gauche	Carte horizontale face en haut	Carte horizontale face en bas	On agite la carte
Message reçu							

3. Compléter le programme suivant pour que la carte affiche un visage heureux si elle est tenue verticalement vers le haut et un visage triste dans le cas contraire.

```

1 from microbit import *
2
3 while (True):
4     message = accelerometer.current_gesture()
5
6     if message == "....." :
7         display.show(.....)
8
9     else:
10        display.show(.....)
11

```



3. Cours : L'IHM

Une **IHM** (Interface Homme Machine) est un ensemble de moyens informatiques qui permettent de favoriser la communication entre l'homme et la machine.

On peut utiliser un dispositif physique (boutons, interrupteurs, curseurs, ...) mais également des logiciels (fenêtre de commande, ...) pour interagir avec la machine.



3. Application 3 – Une première IHM très simple

INTERMEDIATE LEVEL

Le microcontrôleur possède également un capteur de température. Ce capteur ne mesure pas réellement la température ambiante mais celle de la puce en silicium du processeur principal. (C'est pourquoi la température indiquée peut souvent être supérieure à celle de la pièce).



- Réalisez un programme qui affiche la température de la pièce lorsque l'on presse le bouton A du microcontrôleur et la commente avec une pression du bouton B.

Les fonctions et conditions à utiliser sont les suivantes :

`display.scroll(temperature())` : fonction qui fait défiler la température. On pensera à la décaler si nécessaire pour qu'elle corresponde à celle de la pièce.

`if button_a.is_pressed()` : condition qui vérifie si on appuie sur le bouton A.

`if button_b.is_pressed()` : condition qui vérifie si on appuie sur le bouton B.

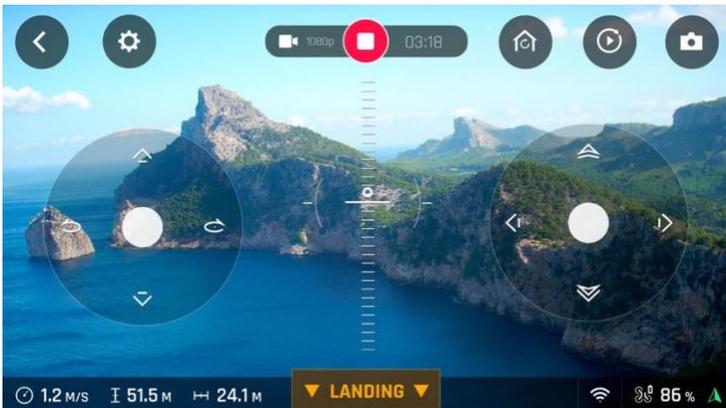
Exemple de commentaire : `if temperature()>25 : display.scroll("Il fait chaud")`



Application 4 – L' interface graphique d'une IHM

INTERMEDIATE LEVEL

Nous allons maintenant réaliser l'interface graphique d'une IHM. Les IHM avec interface graphique sont rencontrées très fréquemment. La machine est alors contrôlée par un logiciel conçu spécialement à cet effet. C'est le cas par exemple pour le pilotage d'un drone.



IHM pour le contrôle d'un drone

Attention! Nous pouvons toujours utiliser Mu mais nous utiliserons plutôt pour réaliser l'interface graphique EduPython.



Le programme EduPython:

```

from tkinter import * #importe le module de création de fenêtre avec Python

fenetre=Tk() # Création d'une fenêtre qui se nomme "fenetre"

fenetre.geometry('300x100') # Dimensions de départ de la fenêtre
6 fenetre.configure(bg='yellow')

monTexte = Label(fenetre, text="Le message à envoyer:") # création d'un petit texte
saisie = Entry () # création d'une zone de saisie
10 bouton=Button(fenetre,text="Envoyer")

monTexte.pack() #on place le texte dans la fenêtre
saisie.pack() #on place la zone de saisie dans la fenêtre
bouton.pack()

fenetre.mainloop() # Permet d'exécuter le programme plusieurs fois et de laisser la fenêtre ouverte
    
```

-Télécharger le programme précédent et répondez aux questions suivantes après l'avoir lancé.

1. Que fait la fonction configure(bg ="yellow") ?

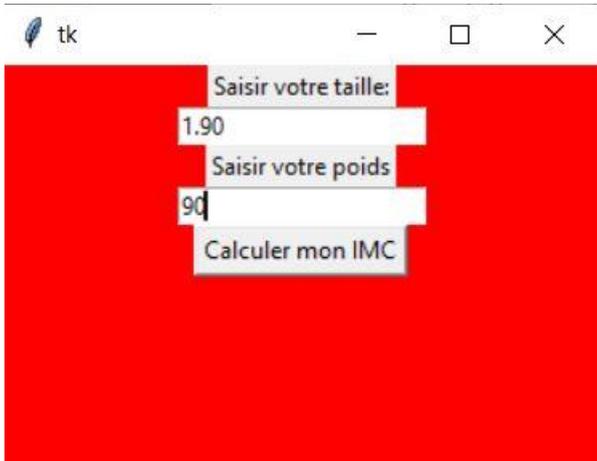
.....

2. A quoi sert la ligne 10 ?

.....

3. Que fait la fonction pack() ?

- Modifier le programme précédent pour construire une petite interface graphique ressemblant à l'image suivante:



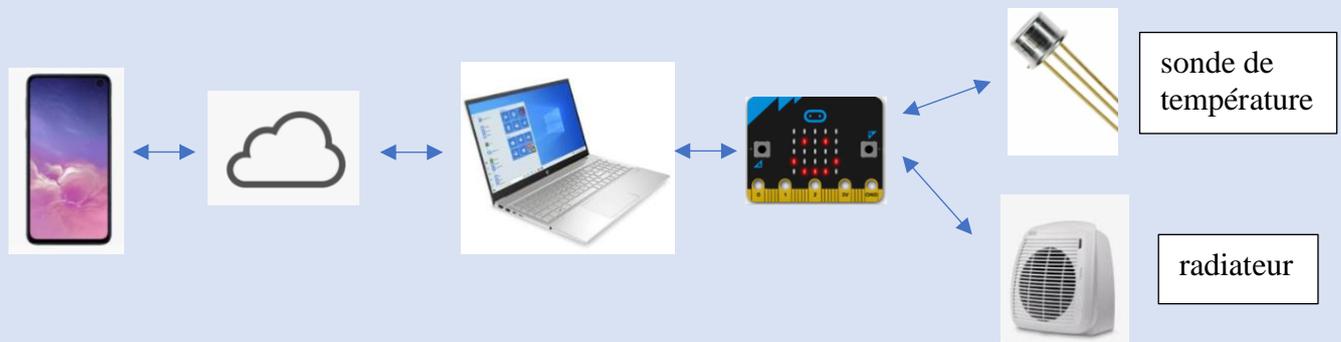
Remarque:

Pour simplifier la programmation le bouton ici n'a aucune action. Mais il est évidemment possible de faire en sorte qu'il envoie un message vers le microcontrôleur pour déclencher une action de celui-ci.



4 Cours : Les objets connectés

L'IHM réalisée précédemment permet de commander le microcontrôleur à partir d'un ordinateur et d'une liaison filaire. Mais on peut imaginer par exemple le dispositif suivant:



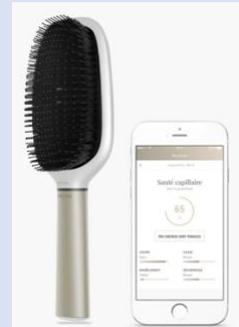
Un téléphone portable peut communiquer ou recevoir des informations sur un site internet. Un microcontrôleur est relié à ce même site internet et peut mesurer une température et commander un radiateur électrique. On peut ainsi contrôler la température de sa maison depuis un téléphone portable. On parle alors d'objets connectés.

Le plus souvent l'objet connecté contient directement le microcontrôleur et peut accéder directement à internet via une box WIFI. Ils sont de plus en plus nombreux dans notre quotidien.

Exemple d'objets connectés:



Un analyseur d'eau de piscine



Une brosse à cheveux



4. Application 5 – Des objets de plus en plus présents



1. Compléter le tableau suivant avec des objets connectés de votre choix:

Objet connecté	capteur(s)	actionneur(s)	IHM
Lampe 	aucun	lampe	Réglage de la luminosité sur son téléphone portable
enceinte 	microphone	haut-parleur	Reconnaissance vocale



4. Application 6 – Des objets qui communiquent entre eux.

📶 INTERMEDIATE LEVEL

Nous allons faire communiquer entre eux les microcontrôleurs.

- Choisissez dans la classe ceux qui seront les **émetteurs** d'un messages et ceux qui seront les **receveurs**. (On intervertira les rôles par la suite)

Attention! Le code à envoyer sur la carte n'est pas le même pour les émetteurs et les receveurs.

- Le professeur va distribuer un numéro de groupe unique pour chaque groupe "émetteur-receveur".

- Faites un copier-coller des codes ci-dessous dans Mu selon votre rôle sans oublier de **changer le numéro de groupe**.

Émetteur:

```
from microbit import *
import radio
radio.config (address=1, group=1)
radio.on()
while True:
    radio.send("Voici un message")
```

Recepteur:

```
from microbit import *
import radio
radio.config (address=1, group=1)
radio.on()
while True:
    donnees_reception = radio.receive()
    display.scroll(str(donnees_reception))
```

Pour aller plus loin:

Imaginer des modifications des programmes précédents pour envoyer des images (comme dans l'application 1) et non plus des messages.



Crédits

- Josselin Darciaux, David Laurier, Marie-Lorraine Lombard, Jean-Laurent Morati, Patrick Meunier et Stéphane Moureaux
- <https://frederic-junier.org>
- <http://algotbot-edu.org/>