

## Alarme de voiture

### Compétences visées

Un des objectifs de l'enseignement de SNT est de développer et de coder des scripts PYTHON afin d'apporter une réponse à une problématique précise. A travers le thème " informatique embarquée et objets connectés", nous pouvons notamment travailler les compétences suivantes dans l'activité proposée :

- Coder des scripts simples d'acquisition de données.
- Gérer des entrées/sorties à travers les ports utilisés par le système.
- Écrire et développer des algorithmes pour résoudre une problématique.
- Identifier des algorithmes de contrôle des comportements physiques à travers les données des capteurs.

### Situation déclenchante



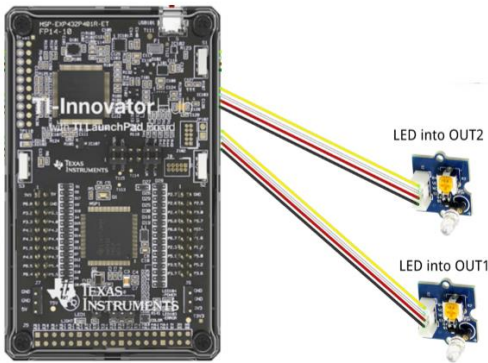
Aujourd'hui, de nombreuses voitures sont équipées d'alarmes. Ces alarmes ne sont pas nécessairement associées à la défense du véhicule. Qu'il s'agisse d'une alarme sonore ou d'une alarme visuelle, une alarme peut être un simple indicateur d'un événement particulier.

### Problématique

Comment concevoir une solution technologique pour simuler le déclenchement d'une alarme de voiture ?

## Fiche méthode

### Matériel nécessaire



- Calculatrice TI-83 Premium CE
- Câble (calculatrice/Hub)
- TI-Innovator Hub
- Câble x2
- LED blanche x2

### Déroulement possible du projet

Travail de groupe possible. Chaque groupe disposera du matériel ci-dessus et devra concevoir une réponse à la problématique sous forme d'une maquette munie d'une documentation qui explique les scripts pilotant la maquette. A la fin du projet, les groupes pourront voter pour élire la production qui répond le mieux à la problématique.

La maquette devra répondre aux critères suivants :

Critère 1 : Jouer 2 sons de 1 seconde chacun dans une boucle pour simuler une alarme sonore.

Critère 2 : Faire clignoter 2 leds externes pour simuler l'allumage des clignotants.

### Proposition de résolution

Critère 1 : Jouer 2 sons de 1 seconde chacun dans une boucle pour simuler une alarme sonore.

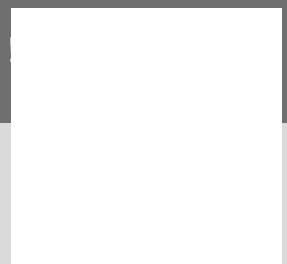
- Importation des bibliothèques *sound* et *time* (voir le paragraphe importation des bibliothèques ci-après).
- L'instruction **sound.tone (440,1)** permet d'émettre un son à une fréquence de 440 hertz pendant 1 seconde.
- L'instruction **sleep (1)** permet de temporiser (1s) l'exécution du programme pendant l'émission du son précédent.

```

ÉDITEUR : ALARME
LIGNE DU SCRIPT 0011
from time import *
import sound

def son():
    for i in range(1,11):
        sound.tone(440,1)
        sleep(1)
        sound.tone(880,1)
        sleep(1)
    
```

Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus



## Fiche méthode

Critère 2 : Faire clignoter 2 leds externes.

- Importation de la bibliothèque *led* (voir remarque).
- L'instruction : `a=led("OUT 1")` permet d'associer la variable a, à la led (objet physique) connectée au port OUT 1 du hub.
- L'instruction : `b=led("OUT 2")` permet d'associer la variable b, à la led (objet physique) connectée au port OUT 2 du hub.
- L'instruction `a.on()`, permet d'allumer la led branchée au port 1.
- L'instruction `a.off()`, permet d'éteindre la led branchée au port 1.

```

ÉDITEUR : ALARME
LIGNE DU SCRIPT 0011
from led import *
def diode():
  a=led("OUT 1")
  b=led("OUT 2")
  for i in range(1,31):
    a.on()
    b.on()
    sleep(1)
    a.off()
    b.off()
    sleep(1)
  
```

Critère 2 (seconde version) : Faire clignoter 2 leds externes.

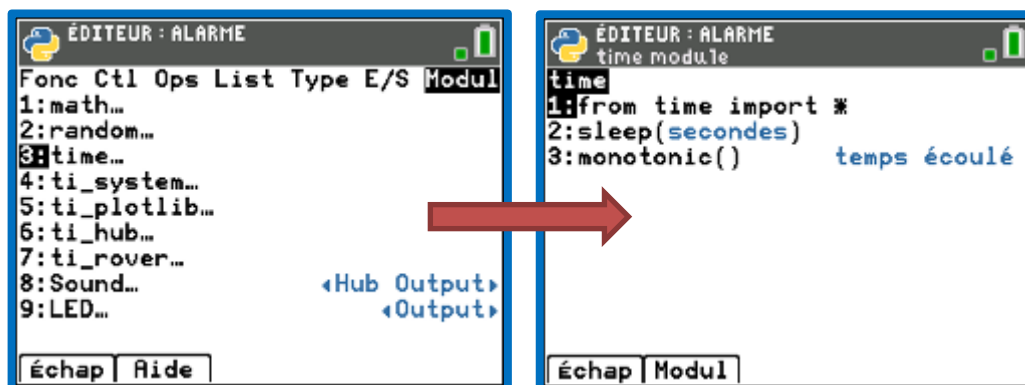
- L'instruction : `a.blink(3,30)` permet d'allumer la led branchée au port 1 et de la faire clignoter 3 fois par seconde pendant 30 secondes.
- L'instruction `sleep(30)` permet de temporiser le programme le temps de l'exécution de l'instruction précédente (le clignotement des leds).

```

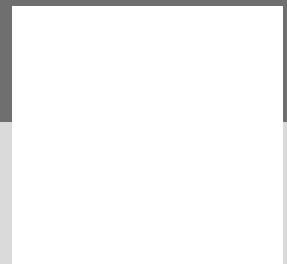
ÉDITEUR : ALARME
LIGNE DU SCRIPT 0035
def diode2():
  a=led("OUT 1")
  b=led("OUT 2")
  a.blink(3,30)
  b.blink(3,30)
  sleep(30)
a
  
```

## Importation des bibliothèques

- Une fois le nouveau script créé, pour importer la bibliothèque *time* appuyer sur la touche `F(x)`, sélectionner la rubrique Modul, puis suivre les copies d'écran suivantes :

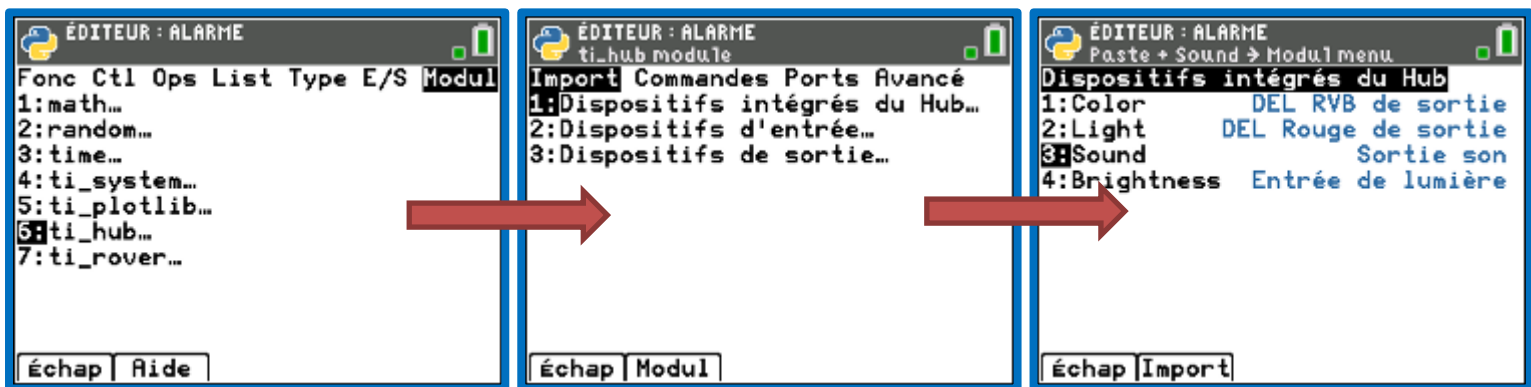
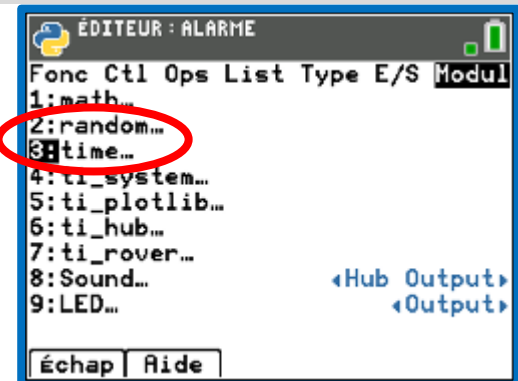


Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus !

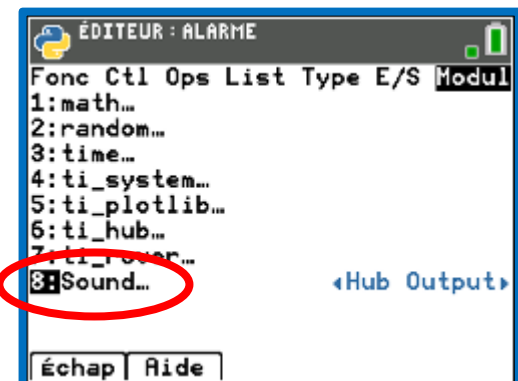


## Fiche méthode

- Pour utiliser la bibliothèque *time* importée, appuyer sur la touche **f(x)**, sélectionner la rubrique Modul puis sélectionner **3 : time...**
- Au fur et à mesure que les bibliothèques sont importées, elles s'ajoutent aux bibliothèques déjà présentes (math, random, etc.) et permettent d'accéder aux instructions qu'elles contiennent.
- Une fois le nouveau script créé, pour importer la bibliothèque *sound*, appuyer sur la touche **f(x)**, sélectionner la rubrique Modul, puis suivre les copies d'écran suivantes :



- Pour utiliser la bibliothèque *sound* importée, appuyer sur la touche **f(x)**, sélectionner la rubrique Modul, puis sélectionner **8 : Sound...**

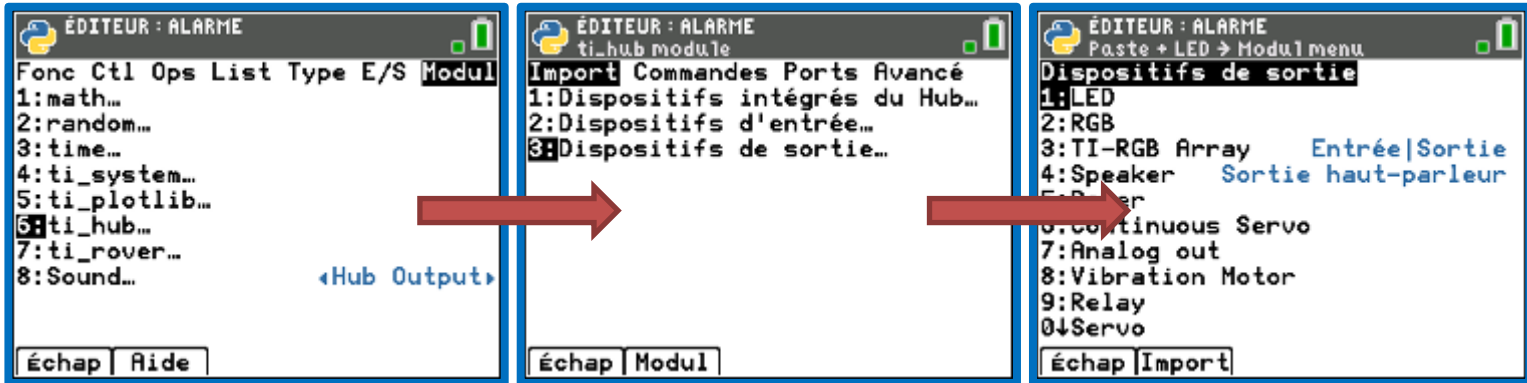


Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus

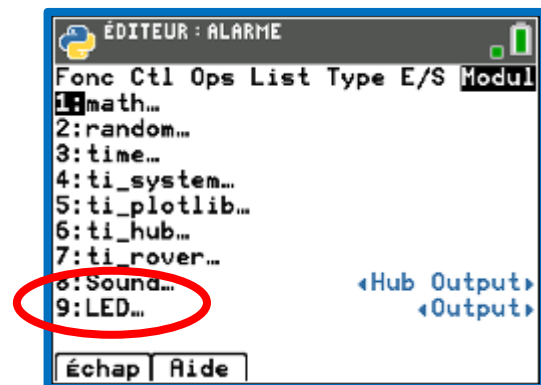


## Fiche méthode

- Pour importer la bibliothèque *led*, appuyer sur la touche `f(x)`, sélectionner la rubrique Modul, puis suivre les copies d'écran suivantes :



- Pour utiliser la bibliothèque *led* importée, appuyer sur la touche `f(x)`, sélectionner la rubrique Modul, puis sélectionner **9 : LED...**



Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus !

