

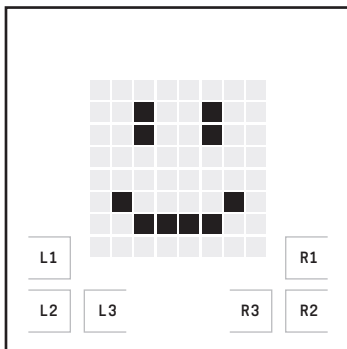
# FICHE A006

## MACHINE À SMILEY

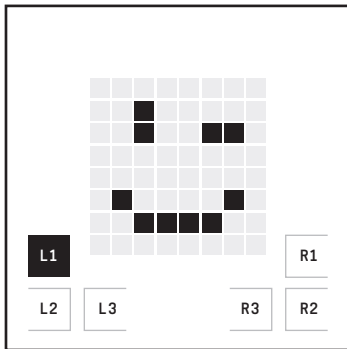
Pour contrôler des appareils, on utilise des capteurs capables de détecter des perturbations de pression, de mouvement ou de volume sonore. L'Oxocard dispose de tous ces types de capteurs. Définir le comportement à adopter lors de la détection d'une telle perturbation relève par contre du travail du programmeur.

### CONSIGNE

Change l'image lors d'une pression sur un bouton.



Le smiley content s'affiche lorsqu'aucun bouton n'est pressé.

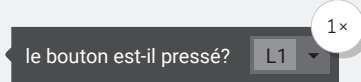
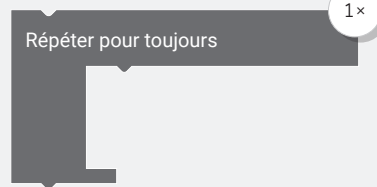


Le smiley clignotant s'affiche pendant quelques secondes après que le premier bouton de gauche (L1) a été pressé.

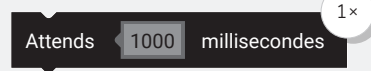
### CONSEIL

Utilise le bloc « Si / alors » en mettant la condition à vérifier derrière le bloc « si » et les instructions à exécuter si la condition est satisfaite dans la partie « alors ».

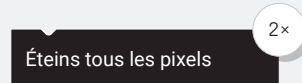
#### LOGIQUE



#### TEMPS



#### MATRICE LED



#### LISTE DES PIÈCES LISTE DES BLOCS UTILISÉS



NIVEAU FACILE

### DÉFI SUPPLÉMENTAIRE:

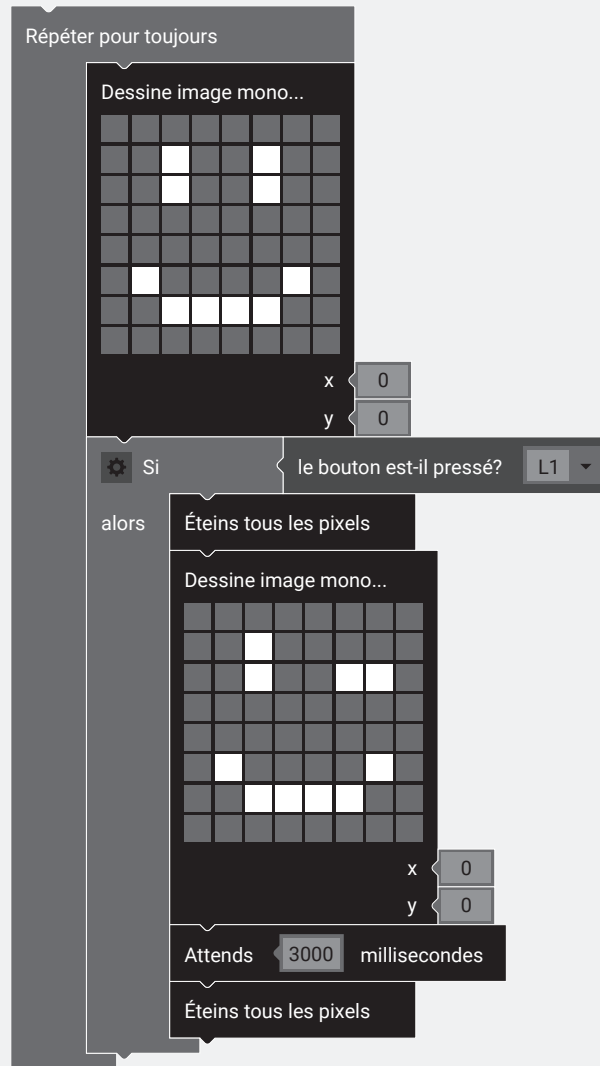
Conçois et programme un smiley différent pour chacun des boutons de l'Oxocard.

# FICHE A006

MACHINE À SMILEY

## SOLUTION

PROPOSITION DE SOLUTION



# FICHE A006

## MACHINE À SMILEY

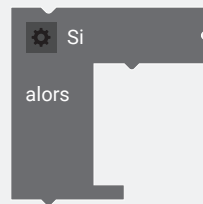
### Objectifs d'apprentissage:

Les élèves apprennent à programmer des instructions conditionnelles.

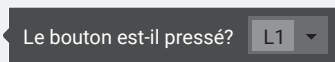
## STRATÉGIE DE RÉSOLUTION

1. Il faut à nouveau utiliser une boucle « Répète pour toujours ». Tous les blocs doivent être insérés dans cette boucle.
2. On affiche ensuite à l'écran l'image qui doit être affichée lorsqu'aucun bouton n'est pressé.
3. Le bloc « Si/alors » est le bloc le plus importante dans ce programme. Il permet de dire presque à l'ordinateur presque littéralement: « Si le premier bouton à gauche est enfoncé, alors efface l'image et dessine-en une nouvelle, attends quelques secondes et efface-la à nouveau ».

## NOUVELLES COMMANDES

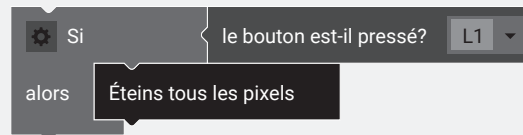


Les instructions se trouvant dans la partie « alors » ne sont exécutées que si la condition placée après le bloc « si » est remplie.



Ce bloc conditionnel vérifie si un bouton (L1, L2, L3, R1, R2, R3) est actuellement pressé.

## COMPLÉMENT DE THÉORIE: CONDITIONS (SI/ALORS)



Les instructions conditionnelles sont incontournables en programmation. Leur structure obéit toujours au même principe: on formule toujours d'abord une condition qui est vraie ou fautive suivie des instructions que l'ordinateur doit exécuter si cette condition est satisfaite (vrai). Dans notre exemple, on interroge l'état d'un bouton. Très souvent, on interroge (lit) également les valeurs fournies par d'autres capteurs que l'on peut comparer entre elles (plus grand que, plus petit que, égalité).

Si l'on clique sur le menu des options en-haut à gauche de ce bloc, on peut étendre le bloc « si » avec un bloc « sinon » ou avec plusieurs blocs « si-sinon ». Les blocs « sinon » ne sont exécutés que lorsque la condition précédente est fautive. Cela permet de construire des structures très complexes. Dans notre exemple, on n'utilise cependant que la forme simple avec une seule condition.



### Exemples de conditions simples:

Lorsque le soleil brille, on va se baigner. « Le soleil brille » est la condition et « nous allons nous baigner » constitue la gestion. Pour permettre plus d'options (branches possibles), on utilise tout simplement plusieurs « sinon si », respectivement « sinon ». Voici un petit exemple: « Si le soleil brille, nous allons nous baigner; S'il pleut on va à la piscine et, sinon, nous restons à la maison ».