

<p><b>Term spé S.I</b> Séquence n°6 <b>AE : Activité Expérimentale</b></p>	<p><b><i>Thème sociétal : L'homme assisté</i></b> <b><i>Intelligence artificielle au service du recyclage</i></b></p>	
<p><b>CONNAISSANCES ASSOCIEES</b></p>		
<p>Algorithme, programme, langage informatique, I.A</p>		
<p><b>PREREQUIS</b></p>		
<p>Apport de connaissance sur l'I.A</p>		
<p><b>Objectif :</b> Réaliser le tri de pièce à l'aide d'un bras robotisé</p>		
<p><b>Compétences terminales visées :</b></p>		
<div style="display: flex; justify-content: space-between; align-items: center;">  </div> <div style="display: flex; justify-content: center; align-items: center; margin-top: 20px;"> <div style="text-align: center;"> <p><b>INNOVER</b></p> </div> <div style="text-align: center;"> <p><b>ANALYSER</b></p> <div style="border: 1px solid black; padding: 5px; margin-top: 5px;"> <p><b>Compétences développées :</b></p> <ul style="list-style-type: none"> <li>- Analyser le traitement de l'information ;</li> <li>- Analyser le comportement d'un objet.</li> </ul> </div> </div> <div style="text-align: center;"> <p><b>EXPERIMENTER &amp; SIMULER</b></p> <div style="border: 1px solid black; padding: 5px; margin-top: 5px;"> <p><b>Compétences développées :</b></p> <ul style="list-style-type: none"> <li>- Instrumenter tout ou partie d'un produit en vue de mesurer les performances ;</li> <li>- Modifier les paramètres influents et le programme en vue d'optimiser les performances d'un objet.</li> </ul> </div> </div> <div style="text-align: center;"> <p><b>COMMUNIQUER</b></p> <div style="border: 1px solid black; padding: 5px; margin-top: 5px;"> <p><b>Compétences développées :</b></p> <ul style="list-style-type: none"> <li>- Documenter un programme informatique .</li> </ul> </div> </div> <div style="text-align: center;"> <p><b>MODELISER &amp; RESOUDRE</b></p> <div style="border: 1px solid black; padding: 5px; margin-top: 5px;"> <p><b>Compétences développées :</b></p> <ul style="list-style-type: none"> <li>- Traduire le comportement attendu ou observé d'un objet.</li> </ul> </div> </div> </div> <p><b>Construction pédagogique :</b></p> <div style="margin-top: 10px;"> <p>Thématique(s) générale(s) : L'homme assisté, réparé, augmenté</p> <p>Thème(s) : I.A au service du recyclage</p> <p>Problématique technique : <b>Comment robotiser une chaîne de tri de déchets recyclables ?</b></p> </div>		
	<p>Niveau Term</p>	
<p>Système réel et les matériels : PC, Dobot, une carte microbit</p>	<p>Groupe de 2 élèves par poste informatique</p>	<p> <b>2h</b></p>
<p> Poste informatique en réseau Logiciels : DobotStudio</p>	<p> Fiches ressources : Aucune</p>	

**Copier le répertoire « AE\_dobot\_tri »**

## 1- Mise en situation

L'intelligence artificielle pourrait aussi apporter une autre nouveauté avec l'introduction de bras robotisés dans les centres de tri. La technologie est apparue aux États-Unis où trois entreprises, AMP Robotics, BHS et Machinex, ont déjà installé quelques dizaines d'unités.

La technologie a, semble-t-il, conquis par sa compacité (des machines bien moins volumineuses que les trieurs optiques) et sa rapidité d'installation (les fabricants assurent pouvoir monter un robot sur une chaîne en quelques jours sans modifier les convoyeurs).



En France, BHS a installé quatre unités sur deux sites exploités par Veolia (un robot à Amiens et deux à Nantes) et un site Paprec (une unité à Lyon). Machinex finalise également la vente de six unités. Par ailleurs, les spécialistes européens des équipements de tri se lancent aussi : Tomra vient tout juste de présenter un modèle spécifique au tri des bouteilles en PET (polyéthylène terephthalate). Pellenc ST travaille activement sur le sujet.

Des technologies éprouvées : concrètement, la technologie combine un bras robotique à l'intelligence artificielle. Le bras ne pose pas de réelles difficultés, les fabricants se tournant vers des technologies éprouvées, notamment dans l'agro-alimentaire ou l'automobile. Le système de préhension des objets, basé sur une ventouse, est un peu plus délicat à mettre au point, compte tenu de la grande variabilité des déchets à capter. Mais, là aussi, les solutions techniques semblent robustes. Finalement, ce sont les progrès réalisés par l'intelligence artificielle qui ont permis d'appliquer ces technologies aux déchets.

On souhaite réaliser le prototype du robot dans un centre de tri. (voir la vidéo

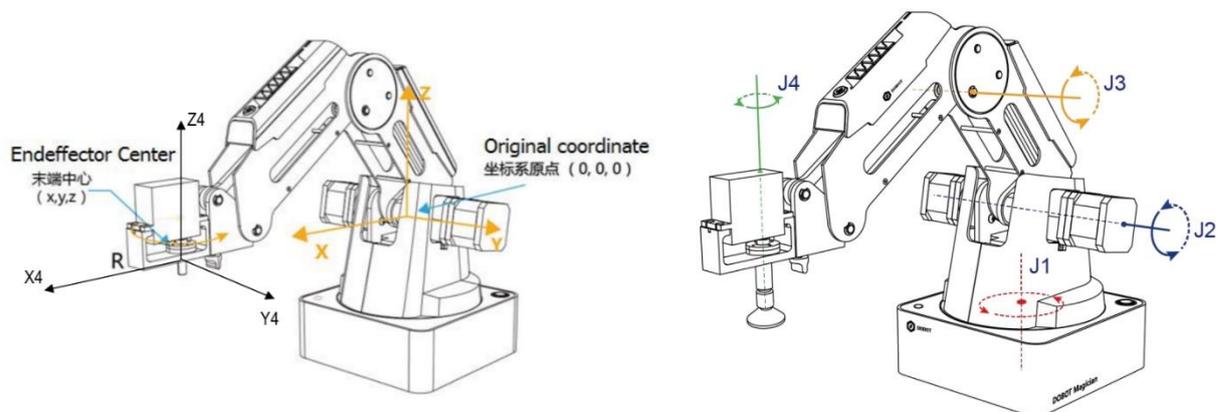
dans le répertoire « AE\_dobot\_tri »

Dans cette activité, vous allez programmer le robot Dobot Magician 4 axes afin de répondre à la demande de l'entreprise de recyclage. Le cahier des charges est le suivant :

- Les produits arrivent les uns après les autres dans l'axe du robot ;
- La caméra identifie le produit (canette ou bouteille plastique) ;
- Le robot saisit le produit et l'évacue vers le conteneur correspondant.

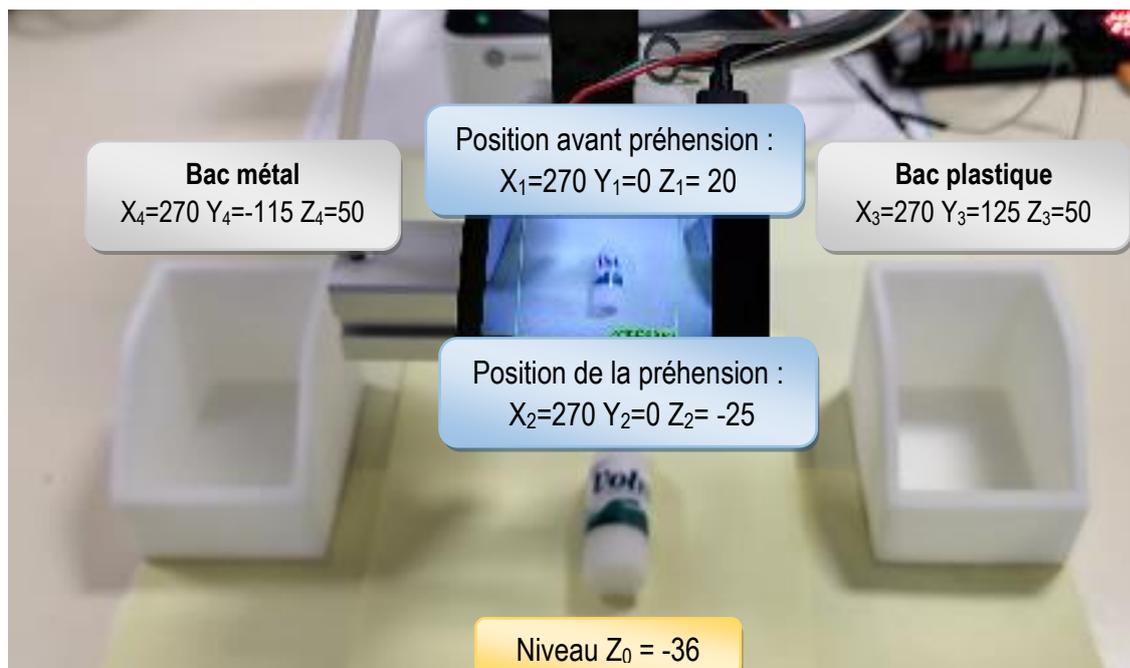
Prendre connaissance du diagramme des exigences SYSML : voir dossier ressources

Les positions données sur le schéma ci-dessous représentent les coordonnées absolues de la pince dans le repère  $R_0(X,Y,Z)$ . **! \ la position sur l'axe Z (= -36) est donnée lorsque la pince est juste au dessus de la table.**



L'orientation de la pince est portée par l'axe de rotation J4 et est défini en programmation par la commande « SetR ».

Schéma du positionnement des différents éléments sur le centre de tri :



## 2- Prise en main du robot

Déplacez-vous à l'endroit indiqué par le professeur afin de manipuler le robotDobot

Démarrez le robot en appuyant sur le bouton d'alimentation du robot



Lancez le logiciel "Dobot studio" sur le bureau :  l'interface suivante doit s'ouvrir

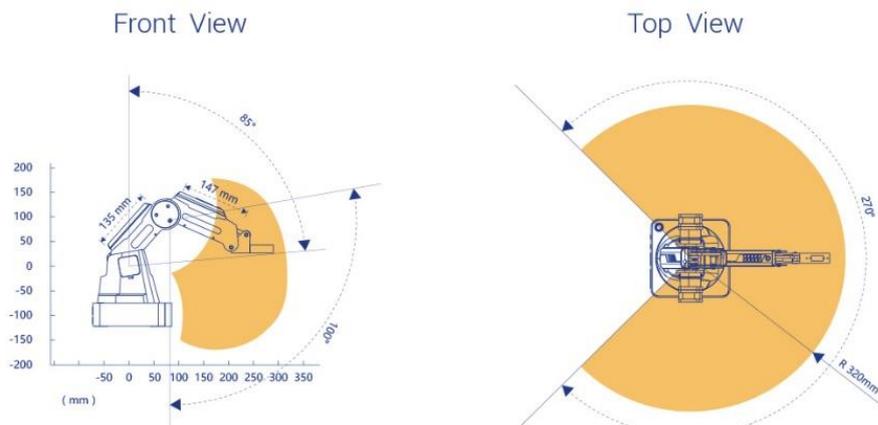


Dans un premier temps, il faut connecter le robot au PC, pour ce faire, cliquez sur "connect" :



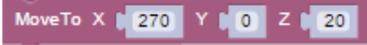
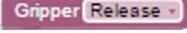
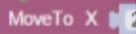
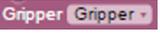
Une fois connecté, vous allez faire les prises d'origine machine en cliquant sur "home", **ATTENTION** :

l'espace doit être **dégagé autour du robot** (voir les débattements ci-dessous) :



### 3- Programmation du robot afin de saisir une bouteille ou canette et l'évacuer

3.1- Programmation par bloc afin de trier une bouteille plastique :

- SetR=0 (Axe de rotation J4 (pince) à 0°) ; 
- Position de départ X1=270, Y1=0, Z1=20 ; 
- Ouvrir la pince ; 
- Se déplacer à la position X2=270, Y2=0, Z2=-25 ; 
- Fermer la pince ; 
- Se déplacer à la position X3=270, Y3=125, Z3=50 ;
- Ouvrir la pince ;
- Se déplacer à la position X1, Y1, Z1 ;
- .....

3.2- Sous le logiciel



cliquez sur l'icone

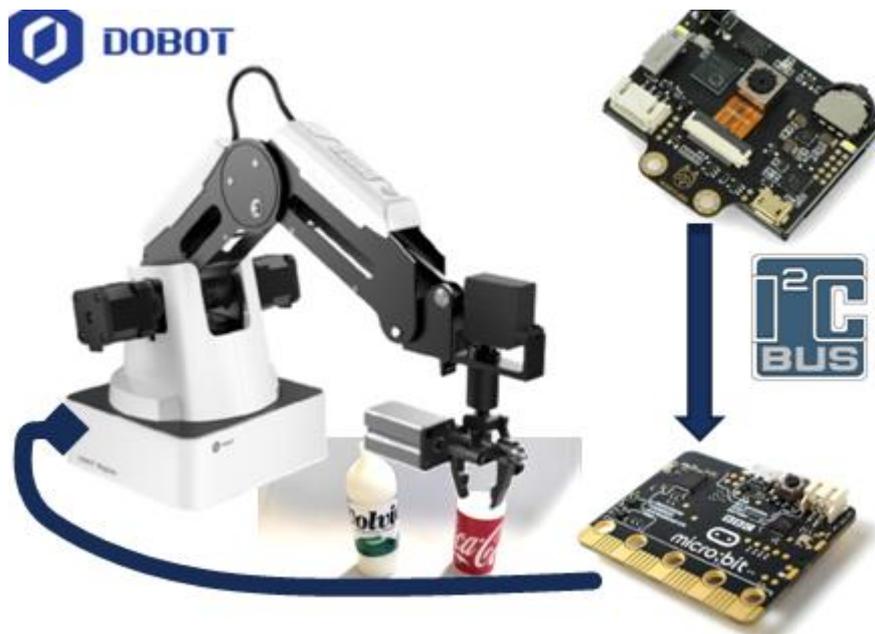


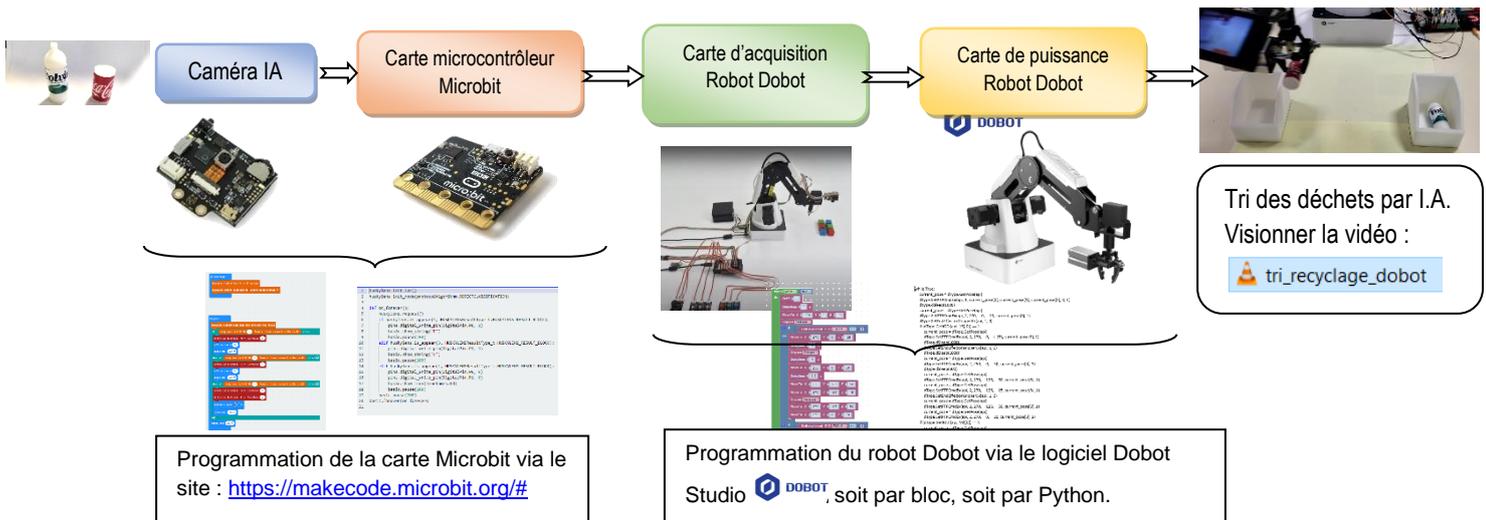
et réaliser le programme par bloc du robot Dobot.

**FAITES VERIFIER VOTRE PROGRAMME PAR LE PROFESSEUR**

### 4- Optimisation de la programmation du robot à l'aide de la caméra I.A

La chaîne de tri reproduit les mêmes mouvements que précédemment. L'objectif maintenant est d'introduire de l'intelligence artificielle à l'aide de la caméra Huskylens (cf AE n°1). Afin de bien comprendre la chaîne d'informations, voici un synoptique du fonctionnement des différents éléments :





#### 4.1- Identification des entrées / sorties à configurer afin de commander le robot Dobot

Dans cette partie il est essentiel de bien comprendre le fonctionnement de la chaîne d'information. La caméra préalablement configurée identifie les objets de nature plastique ou métallique (cf AE n°1). Une fois identifiés, la caméra envoie ces informations à la carte Microbit qui, en fonction de l'objet identifié, va mettre une de ses sorties (Pin0 ou Pin1) à un niveau logique « 1 » (c.a.d : 3.3V). Ces deux sorties de la Microbit sont câblées à deux entrées du robot Dobot qui, à l'aide de sa programmation, va identifier quelle entrée est à un niveau logique « 1 » ce qui correspond à un déchet plastique ou métallique et de ce fait va l'envoyer dans le conteneur correspondant. Le schéma ci-contre montre la solution technologique retenue :



## 4.1.1- Identification de la correspondance des sorties de la carte Micro:bit en fonction des déchets

Ci-dessous le programme de la carte Microbit en Python, identifiez quelle(s) sortie(s) ainsi que son état (c.a.d : à « 1 » ou « 0 ») permettant d'identifier une bouteille plastique ou une canette métallique (cf AE n°1). (compléter le tableau ci-après)

```

huskylens.init_i2c()
huskylens.init_mode(protocolAlgorithm.OBJECTCLASSIFICATION)

def on_forever():
    huskylens.request()
    if huskylens.is_appear(ID1, HUSKYLENSResultType_t.HUSKYLENS_RESULT_BLOCK):
        pins.digital_write_pin(DigitalPin.P0, 1)
        basic.show_string("B")
        basic.pause(200)
    elif huskylens.is_appear(ID2, HUSKYLENSResultType_t.HUSKYLENS_RESULT_BLOCK):
        pins.digital_write_pin(DigitalPin.P1, 1)
        basic.show_string("C")
        basic.pause(200)
    elif huskylens.is_appear(ID3, HUSKYLENSResultType_t.HUSKYLENS_RESULT_BLOCK):
        pins.digital_write_pin(DigitalPin.P0, 0)
        pins.digital_write_pin(DigitalPin.P1, 0)
        basic.show_icon(IconNames.NO)
        basic.pause(200)
    basic.pause(200)
basic.forever(on_forever)

```

	Bouteille plastique	Canette métallique	Aucun objet
Sortie Pin0			0
Sortie Pin1			0
Entrée EI015			0
Entrée EI019			0

A l'aide du schéma de câblage précédent, identifiez l'état des entrées du robot Dobot en fonction des sorties Pin0 et Pin1, complétez le tableau ci-dessus.

## 5- programmation de la chaîne de tri

Dans cette partie, vous allez programmer le robot Dobot afin qu'il effectue le tri des déchets plastiques ou métalliques en fonction des informations qui arrivent sur les entrées EI015 et EI019.

5.1- Sous le logiciel  cliquez sur l'icône , puis ouvrez le fichier  tri\_dechets\_eleve .

Complétez le programme ci-dessous à l'aide du programme par bloc (page suivante) et faites vérifier avant de modifier le code sous le logiciel.

```

while .....:
    current_pose = dType.GetPose(api)
    dType.SetPTPCmdEx(api, 1, current_pose[0], current_pose[1], current_pose[2],
    0, 1)
    dType.dSleep(1000)
    current_pose = dType.GetPose(api)
    dType.SetPTPCmdEx(api, 2, 270, 0, 20, current_pose[3], 1)
    dType.SetEndEffectorGripperEx(api, 1, 0)

    if (dType.GetIODI(api, 15)[0]).....1:
        current_pose = dType.GetPose(api)
        dType.SetPTPCmdEx(api, 2, ....., 0, (-.....), current_pose[3], 1)
        dType.dSleep(1000)
        dType.SetEndEffectorGripperEx(api, 1, 1)
        dType.dSleep(.....)#pause en ms
        current_pose = dType.GetPose(api)
        dType.SetPTPCmdEx(api, 2, ....., 0, ....., current_pose[3], 1)
        dType.dSleep(500)
        current_pose = dType.GetPose(api)
        dType.SetPTPCmdEx(api, 2, ....., ....., ....., current_pose[3], 1)
        current_pose = dType.GetPose(api)
        dType.SetPTPCmdEx(api, 2, 270, 125, 15, current_pose[3], 1)
        dType.SetEndEffectorGripperEx(api, 1, 0)
        current_pose = dType.GetPose(api)
        dType.SetPTPCmdEx(api, 2, 270, 125, 50, current_pose[3], 1)
        current_pose = dType.GetPose(api)
        dType.SetPTPCmdEx(api, 2, 270, 0, 20, current_pose[3], 1)
    if (dType.GetIODI(api, 19)[0]) ..... 1:
        current_pose = dType.GetPose(api)
        dType.SetPTPCmdEx(api, 2, 270, 0, (-25), current_pose[3], 1)
        dType.dSleep(1000)
        dType.SetEndEffectorGripperEx(api, 1, 1)
        dType.dSleep(1000)
        current_pose = dType.GetPose(api)
        dType.SetPTPCmdEx(api, 2, 270, 0, 50, current_pose[3], 1)
        dType.dSleep(500)
        current_pose = dType.GetPose(api)
        dType.SetPTPCmdEx(api, 2, 270, (-115), 50, current_pose[3], 1)
        current_pose = dType.GetPose(api)
        dType.SetPTPCmdEx(api, 2, 270, (-115), 15, current_pose[3], 1)
        dType.SetEndEffectorGripperEx(api, 1, 0)
        current_pose = dType.GetPose(api)
        dType.SetPTPCmdEx(api, 2, 270, (-115), 50, current_pose[3], 1)
        current_pose = dType.GetPose(api)
        dType.SetPTPCmdEx(api, 2, 270, 0, 20, current_pose[3], 1)

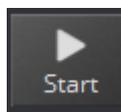
```

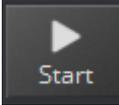
## Programmation par « bloc » de la chaîne de tri du robot Dobot

```

repeat while true
do
  SetR 0
  Delaytime 1 s
  MoveTo X 270 Y 0 Z 20
  Gripper Release
  if GetLevelInput EIO EIO15 = 1
  do
    MoveTo X 270 Y 0 Z -25
    Delaytime 1 s
    Gripper Gripper
    Delaytime 1 s
    MoveTo X 250 Y 0 Z 50
    Delaytime 0.5 s
    MoveTo X 270 Y 125 Z 50
    MoveTo X 270 Y 125 Z 15
    Gripper Release
    MoveTo X 270 Y 125 Z 50
    MoveTo X 270 Y 0 Z 20
  if GetLevelInput EIO EIO19 = 1
  do
    MoveTo X 270 Y 0 Z -25
    Delaytime 1 s
    Gripper Gripper
    Delaytime 1 s
    MoveTo X 270 Y 0 Z 50
    Delaytime 0.5 s
    MoveTo X 270 Y -115 Z 50
    MoveTo X 270 Y -115 Z 15
    Gripper Release
    MoveTo X 270 Y -115 Z 50
    MoveTo X 270 Y 0 Z 20

```



Après vérification, cliquez sur  afin de lancer la chaîne de tri. Concluez sur la possibilité de trier les déchets de façon robotisée.

## DOSSIER RESSOURCES

