

**H.E.P.L. : Département Informatique**

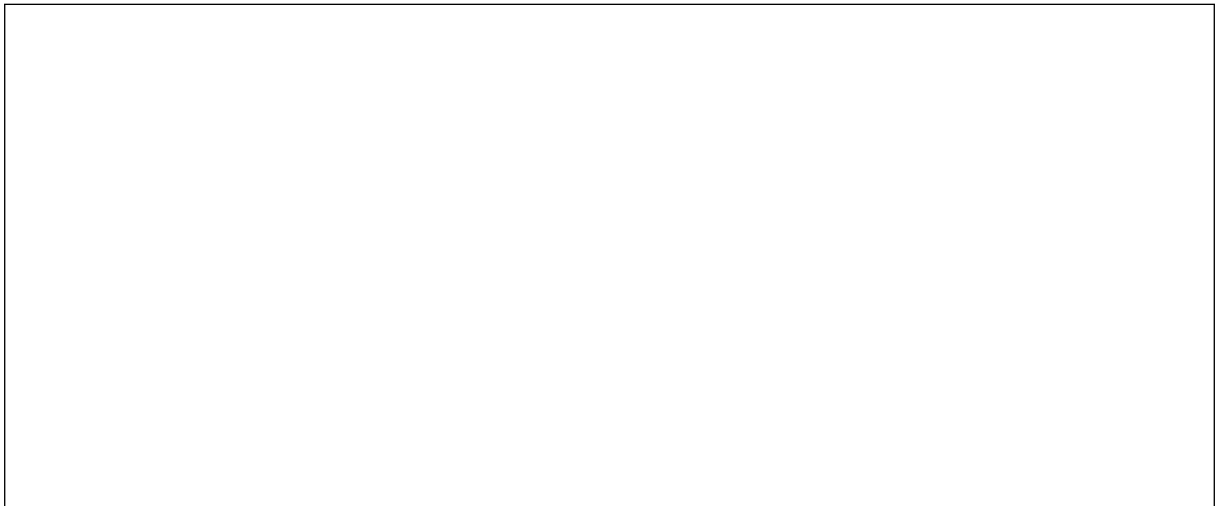
**Graduat en Informatique et Systèmes : finalité Informatique Industrielle**

**LABORATOIRE D'INFORMATIQUE INDUSTRIELLE**

**AUTOMATISATION ET ROBOTIQUE laboratoire**

2ème année

**PETRA monopiece**



Année Académique 2016-2017

ROLAND Julien  
KLEYNTSENS Tom

Groupe : 2222

ROLAND Julien

KLEYNTSENS Tom

# Automatisation & Robotique

## Dossier 1 : PETRA monopieuvre

### Variables utilisées:

struct u\_act : structure comportant la structure avec tous les actuateurs

struct u\_capt : structure comportant la structure avec tous les capteurs

short mauvaise : variable utilisée comme flag si la pièce est mauvaise (à 1 si mauvaise), elle est initialisée à 0

fd\_petra\_in : descripteur du fichier permettant de lire l'état des capteurs

fd\_petra\_out : descripteur du fichier permettant de modifier l'état des actuateurs

### Points importants du cycle :

#### Test du Slot :

Pour vérifier que la pièce possède bien un trou, nous avons attendu que la valeur du capteur S passe à 1, permettant de détecter le début de la pièce. Ensuite, nous avons attendu que cette valeur passe à 0, nous indiquant ainsi que le capteur avait détecté le trou ou la fin de la pièce. Enfin, nous avons procédé à une temporisation d'une demi seconde, suivie d'un test de la valeur du capteur. Si celui-ci était toujours à 0, le capteur avait détecté la fin de la pièce et par conséquent, elle n'avait pas de trou. La variable mauvaise était alors mise à 1. Par contre si la valeur du capteur était repassée à 1, le capteur avait précédemment détecté. Aucune opération n'était alors faite sur la variable mauvaise.

#### Test du Corner :

Pour vérifier que la pièce possède une encoche de la bonne taille, nous avons attendu que la valeur du capteur L1 passe à 1 pour détecter la présence de la pièce. Ensuite, nous avons attendu que la valeur de L1 passe à 0 et nous avons testé la valeur du capteur L2. Si L2 était à 1, la pièce avait une encoche plus petite et par conséquent la pièce était mauvaise. La variable mauvaise était alors mise à 1. Si L1 était à 0, l'encoche était plus longue et donc la pièce bonne. Aucune opération n'était alors faite sur la variable mauvaise.

#### Fonction tempo :

Nous avons créé une fonction tempo pour faciliter la modification des variables de la structure timespec. La fonction prend en paramètre time\_t sec, le nombre de secondes à attendre et unsigned long nano, le temps de nano secondes à attendre. La variable tv\_sec de la structure reçoit le contenu de sec et tv\_nsec reçoit nano. La fonction nanosleep est appelée et on lui passe la structure en paramètre.