

# Baccalauréat général

## Enseignement de spécialité sciences de l'ingénieur

### Sujet zéro

Aucun document n'est autorisé

**Information aux candidats** : les candidats qui disposent d'une **calculatrice** avec **mode examen** devront l'activer le jour des épreuves et les calculatrices **dépourvues de mémoire** seront autorisées. Ainsi, tous les candidats composeront sans aucun accès à des données personnelles pendant les épreuves.

Coefficient 16 – Durée 4 heures

#### ROBOT UBBO MAKER

#### Constitution du sujet

Le sujet comporte deux parties indépendantes qui peuvent être traitées dans un ordre indifférent.

- **Sujet** (*mise en situation et questions à traiter par le candidat*)
  - **partie 1 : sciences de l'ingénieur (3 heures)** ..... pages 2 à 16
    - **documents réponses** ..... pages 17 à 20
  - **partie 2 : sciences physiques (1 heure)** ..... pages 21 à 29

Dans la partie 1 : sciences de l'ingénieur :

- les sous-parties 1 et 4 sont à traiter obligatoirement par tous les candidats ;
- les candidats devront choisir de traiter seulement l'une des deux sous-parties :
  - la sous-partie 2 (choix 1),
  - la sous-partie 3 (choix 2).

Dans la partie 2 : sciences physiques les candidats choisiront de traiter deux des trois exercices proposés.

Les documents réponses DR1 à DR4 (pages 17 à 20) seront à rendre agrafés aux copies.

# Partie 1 : sciences de l'ingénieur

## Mise en situation

« AXYN » est une société française qui conçoit, fabrique et commercialise des robots de service spécialisés dans la téléprésence. Le robot « UBBO Maker » fait partie d'une gamme destinée à l'apprentissage et aux technophiles.

Dans le cadre du maintien à domicile d'un élève (voir figure 1), celui-ci doit pouvoir suivre ses cours à distance. Pour cela, un robot de téléprésence équipé d'une tablette numérique va simuler sa présence dans son établissement scolaire.

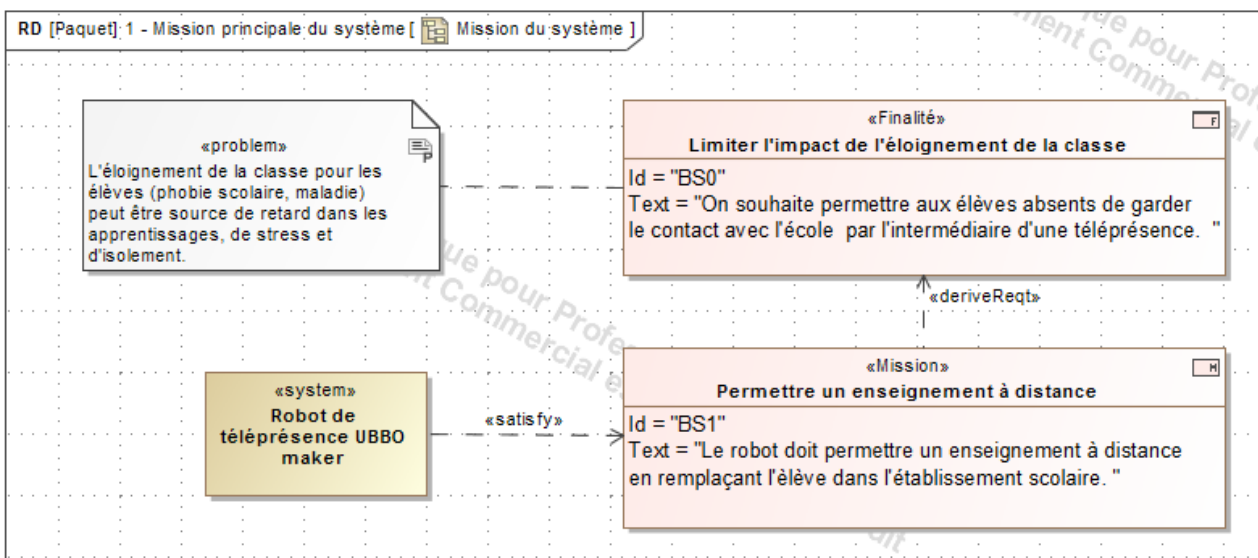


Figure 1 : mission du système.

Schéma de principe de la téléprésence retenue par l'entreprise :

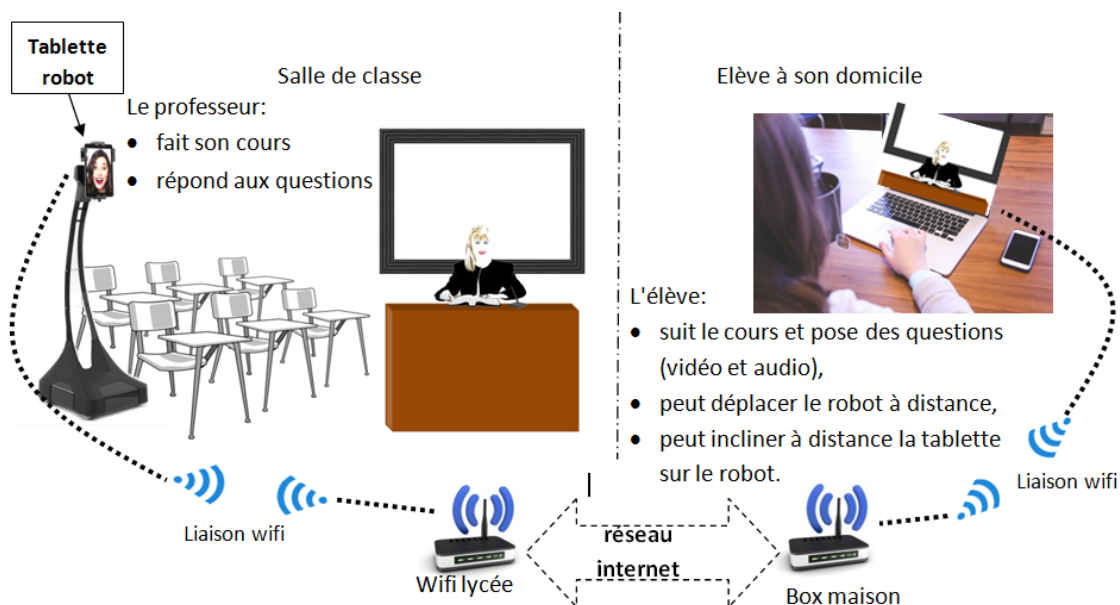


Figure 2 : schéma de principe de la téléprésence.

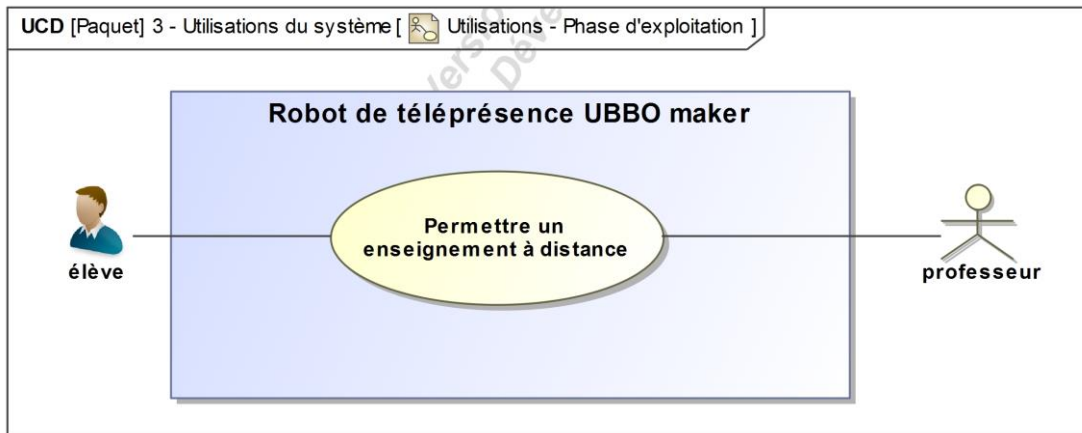


Figure 3 : diagramme des cas d'utilisation.

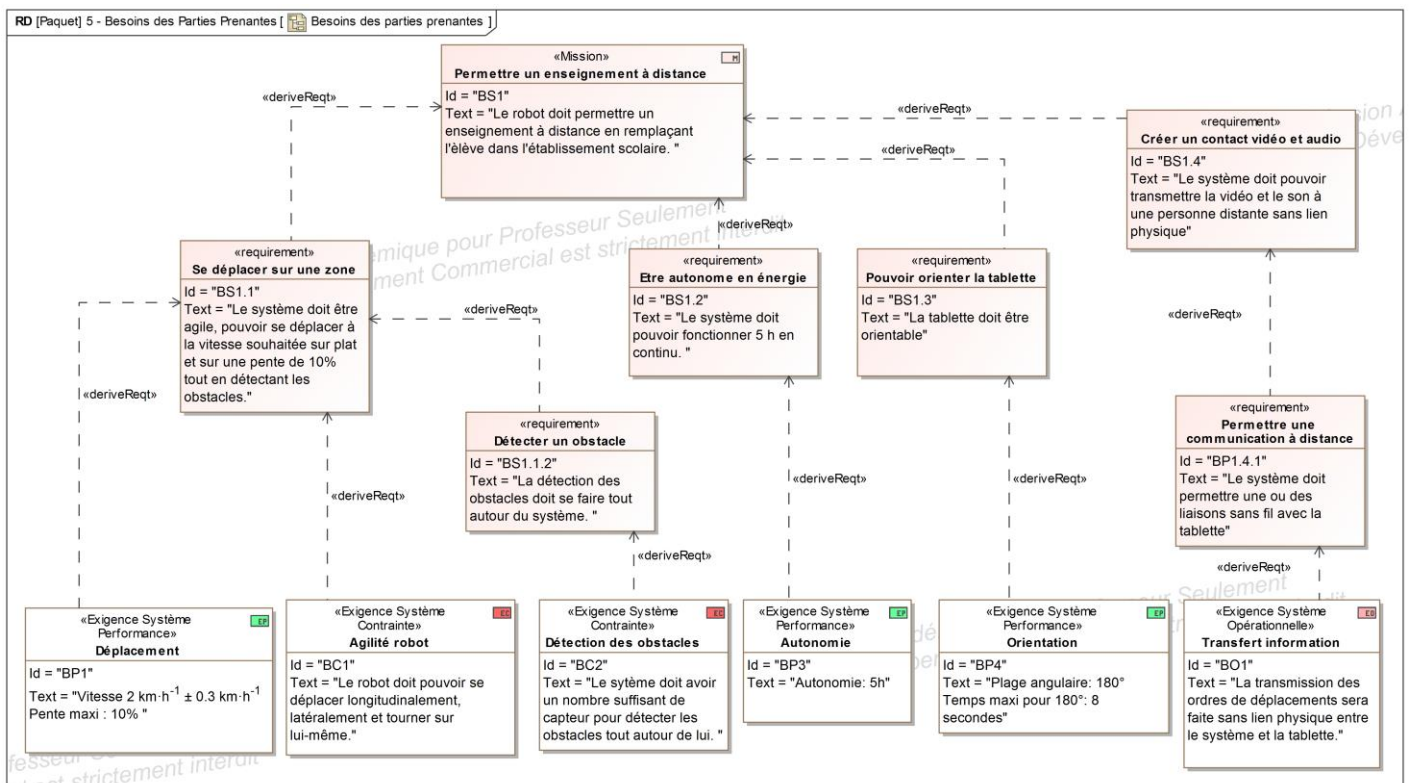


Figure 4 : diagramme des exigences.

La problématique abordée dans ce sujet est la suivante :

- comment positionner le robot dans la salle de classe et orienter la tablette pour permettre l'interaction entre un professeur et un élève situés dans des lieux distants ?

Le sujet va s'attacher à répondre à cette question par la résolution de trois problèmes techniques :

- comment permettre le déplacement du système dans son environnement, afin de pouvoir l'installer et le faire circuler dans la salle de classe quelle que soit la configuration du lieu ;
- comment assurer son autonomie de fonctionnement, afin que l'élève puisse assister à ses cours pendant au moins une demi-journée (le constructeur impose 5 heures) ;
- comment piloter et orienter l'interface de communication, afin d'assurer une bonne visibilité de la salle de classe pour l'élève ?

Enfin, une synthèse permettra de conclure et de répondre à la problématique.

## Sous-partie 1 : à traiter obligatoirement

### Permettre le déplacement du système dans son environnement.

L'objectif de cette partie est d'analyser la solution retenue par l'entreprise pour le déplacement du robot, de valider le choix des moteurs et enfin de garantir un déplacement en ligne droite.

La circulation dans un établissement scolaire et plus particulièrement dans les salles de classe est difficile pour un robot en raison des obstacles (personnes, tables, chaises, cartables, etc.).

**Question 1.1** À l'aide du diagramme des exigences figure 4, **relever** les critères chiffrés de déplacement à respecter pour la circulation dans un établissement scolaire.

La solution retenue par l'entreprise pour le déplacement du robot est l'utilisation de roues suédoises (voir figures 5 et 6). Chaque roue est associée à un moteur et permet un déplacement du robot dans toutes les directions en fonction du sens de rotation des roues.

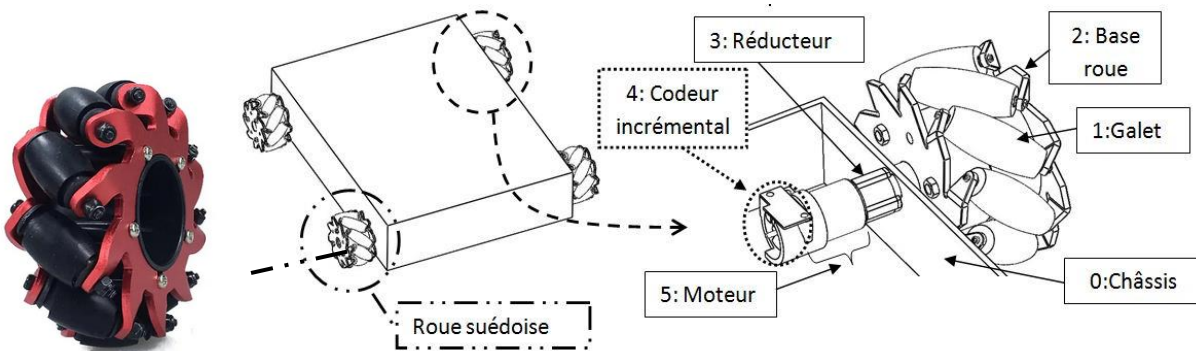


Figure 5 : vue du système de déplacement et détail pour une roue.

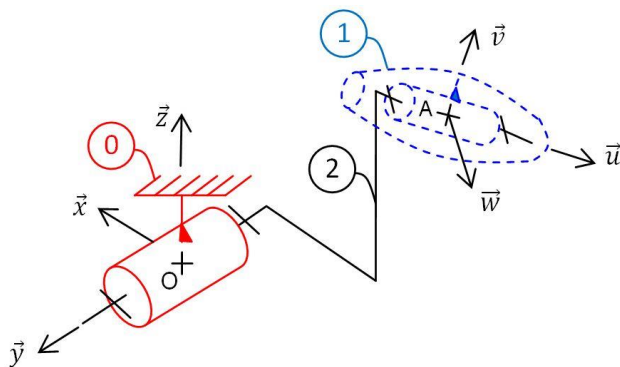


Figure 6 : schéma cinématique pour le guidage d'une roue.

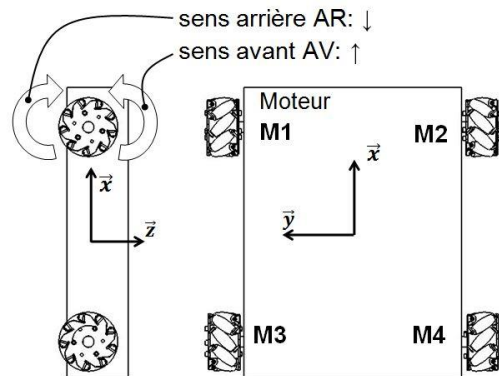


Figure 7 : numéro et sens de rotation des moteurs.

**Question 1.2** À l'aide des figures 5 et 6, **compléter** sur le document réponse DR1, le diagramme de bloc interne avec le nom des composants (repérés avec un point d'interrogation) qui contribuent au déplacement du robot.  
Voir DR1

Le robot doit suivre la loi de vitesse simplifiée suivante pour un **déplacement en ligne droite**.

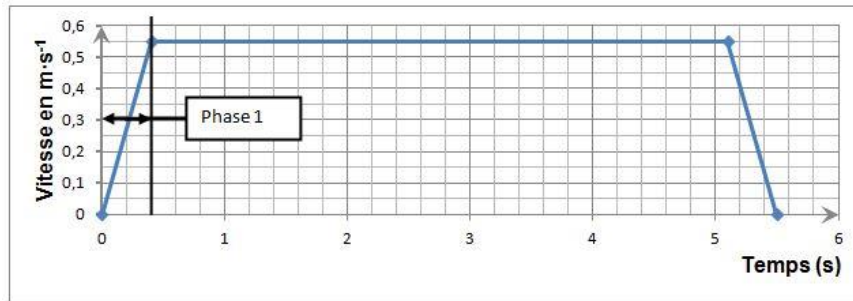


Figure 8 : loi de vitesse simplifiée pour un déplacement du robot.

Question 1.3 À partir de la figure 8, **indiquer** la nature du mouvement pour la phase 1. **Déterminer** l'accélération  $a_G$  du robot lors de cette phase de démarrage.

La figure 9 correspond à la modélisation plane de l'étude du déplacement du robot.

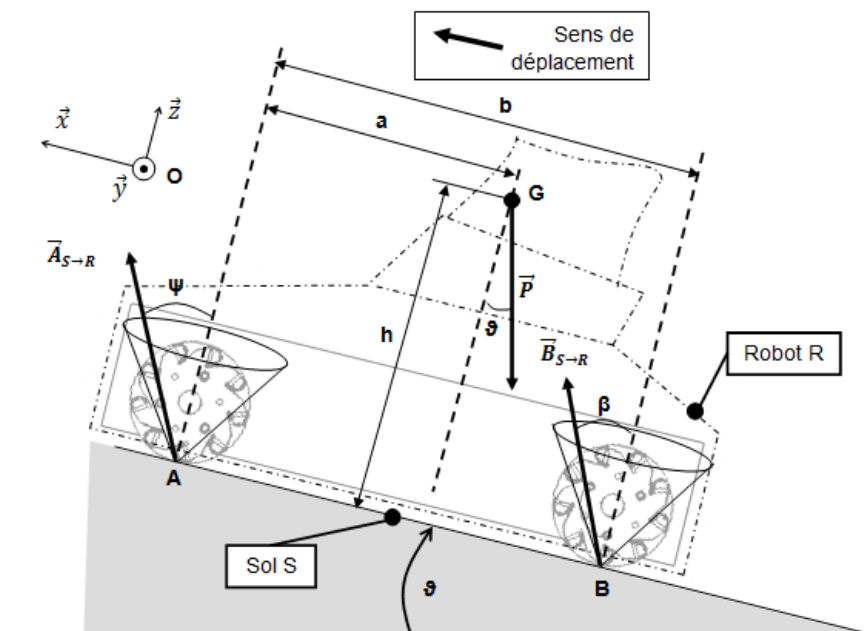


Figure 9 : modélisation des actions mécaniques extérieures exercées sur le robot dans le plan  $(O, \vec{x}, \vec{z})$ .

Hypothèses :

- l'étude est effectuée avec l'hypothèse du roulement sans glissement ;
- le modèle proposé pour le robot figure 9 présente une symétrie de géométrie et d'efforts suivant le plan  $(O, \vec{x}, \vec{z})$  ;
- masse robot,  $M_r = 12 \text{ kg}$  ;
- accélération de la pesanteur,  $g = 9,81 \text{ m}\cdot\text{s}^{-2}$  ;
- accélération du robot,  $a_G = 1,375 \text{ m}\cdot\text{s}^{-2}$  ;
- actions de l'air sur le robot et résistance au roulement négligées ;
- dimensions :  $a = 0,180 \text{ m}$  ;  $b = 0,336 \text{ m}$  et  $h = 0,3 \text{ m}$ .

Actions mécaniques extérieures appliquées sur le robot :

- $\vec{A}_{S \rightarrow R}$ , action de contact du sol sur le robot appliqué au point A ;
- $\vec{B}_{S \rightarrow R}$ , action de contact du sol sur le robot appliqué au point B ;
- $\vec{P}$ , action à distance de la pesanteur sur le robot appliqué au centre de gravité G.

Question 1.4 Lors de la phase d'accélération, **tracer** sur le document DR1 les projections des actions mécaniques  $\vec{A}_{S \rightarrow R}$ ,  $\vec{B}_{S \rightarrow R}$  et  $\vec{P}$  dans le plan  $(0, \vec{x}, \vec{z})$  (exemple pour  $\vec{A}_{S \rightarrow R}$  elles seront notées  $XA$  pour la composante sur  $\vec{x}$ ,  $ZA$  pour celle sur  $\vec{z}$ ). **Retrouver** en détaillant votre démarche, les relations ci-dessous par application du théorème de la résultante dynamique.

Voir DR1

$$XA + XB - P \cdot \sin(\vartheta) = Mr \cdot a_G \text{ (équation 1)}$$

$$ZA + ZB - P \cdot \cos(\vartheta) = 0 \text{ (équation 2)}$$

L'application du théorème du moment dynamique au point A permet de trouver l'équation suivante :

$$ZB \cdot b - P \cdot a \cdot \cos(\vartheta) - P \cdot h \cdot \sin(\vartheta) - a_G \cdot Mr \cdot h = 0 \text{ (équation 3)}$$

Le cas limite se produit lors d'un déplacement sur une pente de 10 % (cela correspond à un angle de 5,71 degrés par rapport à l'horizontale), départ arrêté.

En considérant que l'on se situe à la limite de l'adhérence en A, les angles  $\psi$  et  $\beta$  ont pour valeurs  $\psi = 20$  degrés et  $\beta = 11,3$  degrés.

Question 1.5 **Déterminer** les relations liant  $ZA$ ,  $XA$  et l'angle  $\psi$  ainsi que  $ZB$ ,  $XB$  et l'angle  $\beta$ .

À partir des équations 1, 2 et 3 et des relations précédemment trouvées, **déterminer** pour le cas limite quelle est l'action tangentielle  $XB$  ou  $XA$  la plus importante.

La chaîne de puissance peut être décomposée de la manière suivante.

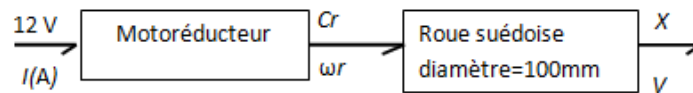


Figure 10 : chaîne de puissance partielle.

Quelles que soient les valeurs obtenues précédemment, les données à utiliser sont :

- l'action tangentielle (X) au niveau du contact de la roue avec le sol dans le cas limite aura pour valeur 8,9 N ;
- vitesse du robot par rapport au sol  $V = 0,55 \text{ m}\cdot\text{s}^{-1}$ .

Question 1.6 **Déterminer** la fréquence de rotation  $\omega_r$  en  $\text{rd}\cdot\text{s}^{-1}$  puis en  $\text{tr}\cdot\text{min}^{-1}$  ainsi que le couple  $C_r$  en  $\text{N}\cdot\text{m}$  en sortie du motoréducteur. **Conclure** sur le choix du moteur fait par l'entreprise (voir figure 11).

Les caractéristiques du motoréducteur choisi par l'entreprise sont les suivantes :

Motoréducteur	Namki coreless motor
Modèle	22CL-3501PG
Tension	12 V
Puissance	15 W
Réducteur	$r=1/80$
Vitesse sortie	$120 \text{ tr}\cdot\text{min}^{-1}$
Couple fonctionnement normal	$0,49 \text{ N}\cdot\text{m}$
Masse	140 g

Figure 11 : données constructeur du motoréducteur.

Les roues suédoises ainsi que l'incidence des choix de la rotation des 4 moteurs (sens avant : AV ou arrière : AR) présentées sur les figures 5, 6 et 7 permettent au robot de se déplacer dans la direction souhaitée. Pour se déplacer en ligne droite, les quatre roues du robot doivent tourner dans le même sens et à la même vitesse.

Un essai de déplacement en **ligne droite** sur une pente de 10 % est mis en œuvre. Une modélisation du fonctionnement en chaîne directe (figure 12) est réalisée sur les roues arrière pour les deux chaînes de puissance (moteurs M3 et M4, voir figure 7). Des perturbations modélisées par des frottements visqueux (proportionnels à la vitesse de rotation) différents sur chaque roue dus aux divers composants (réducteur, etc.) sont appliquées et donnent la simulation figure 13.

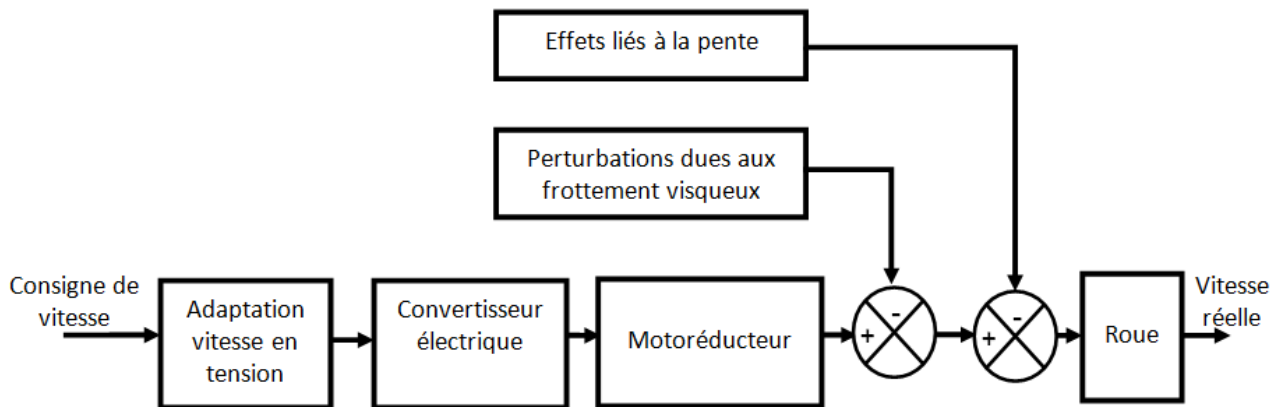


Figure 12 : schéma de principe de commande d'une roue en chaîne directe.

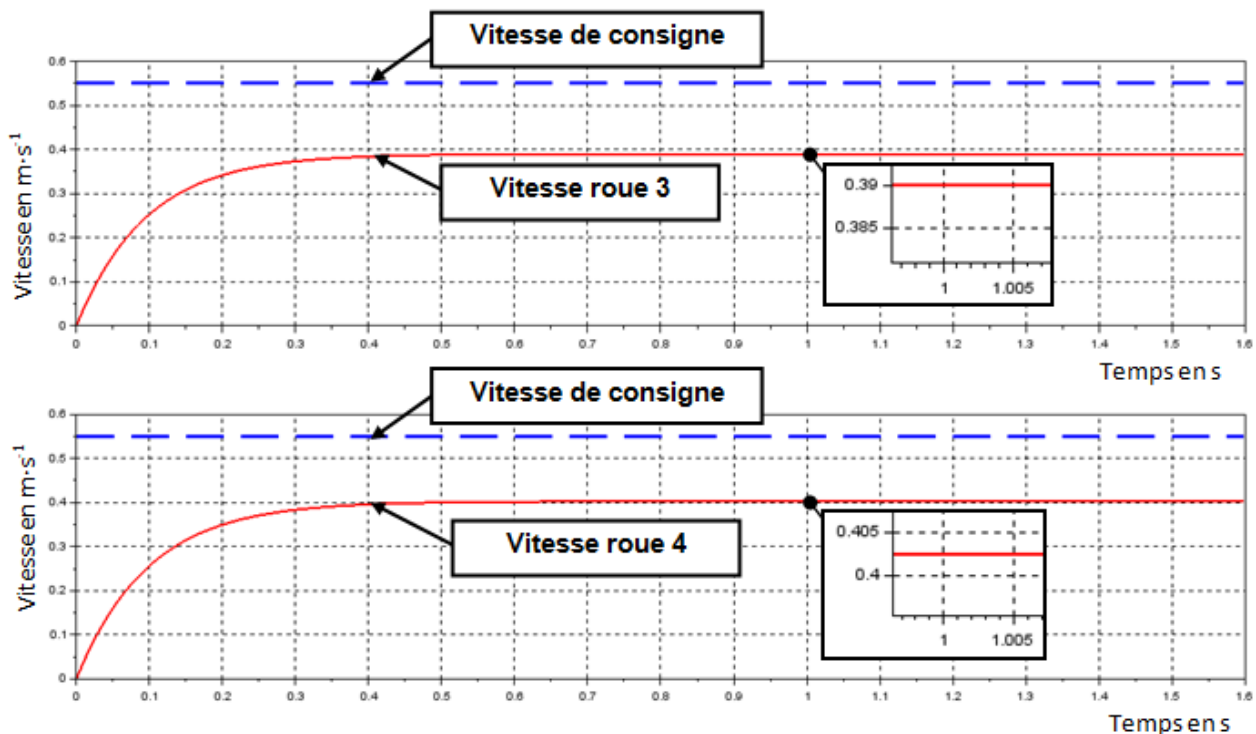


Figure 13 : résultats de la simulation (moteurs M3 et M4) de la chaîne de propulsion.

Question 1.7 À l'aide de la simulation figure 13, **déduire** les conséquences des perturbations sur la vitesse des roues et la trajectoire de déplacement du robot. **Proposer** une solution qui permet de remédier à ce problème.

La mesure de l'angle de rotation moteur, et donc indirectement de la vitesse, est obtenue par des codeurs incrémentaux (voir figure 5). Une régulation en vitesse et l'ajout d'un correcteur proportionnel sont réalisés. Un modèle dont le schéma de principe est donné figure 14 permet d'obtenir la simulation figure 15.

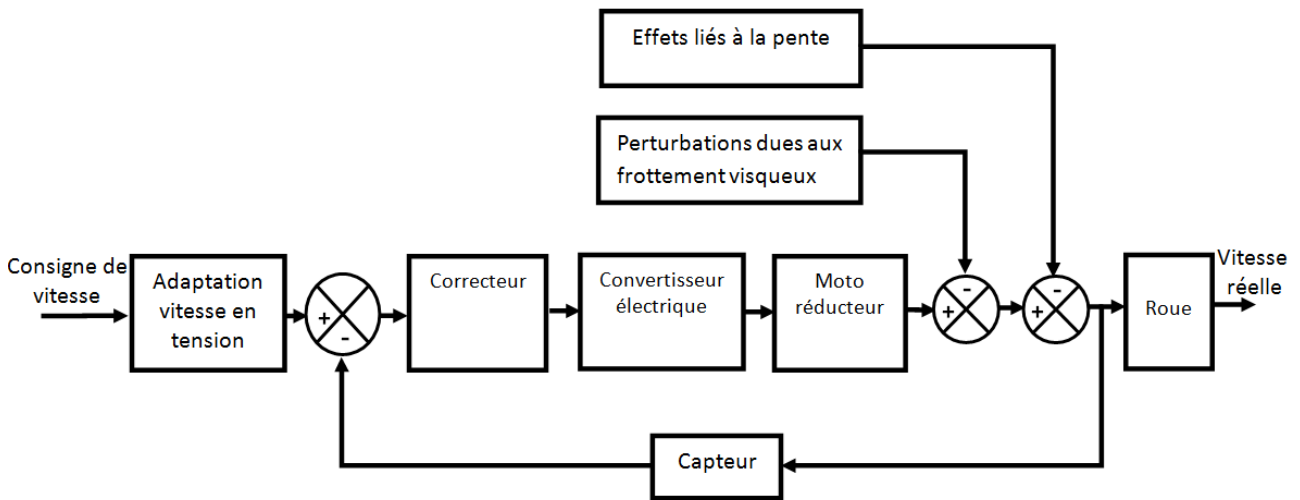


Figure 14 : schéma de principe de commande d'une roue avec régulation de vitesse.

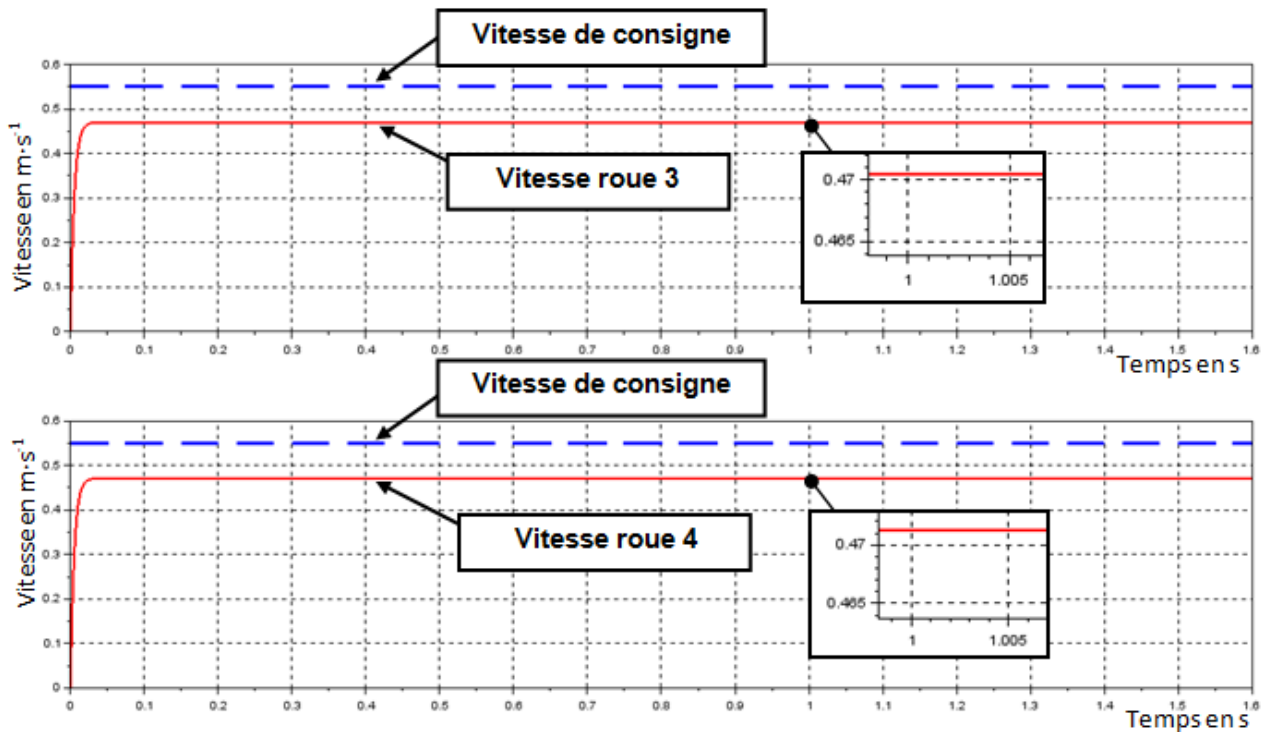


Figure 15 : résultats de la simulation de la chaîne de propulsion (moteurs M3 et M4) avec régulation de vitesse.

Question 1.8 À partir de la simulation figure 15, **déterminer** les écarts pour les vitesses des roues 3 et 4 entre le modèle simulé et l'exigence attendue. **Expliquer** l'intérêt d'avoir utilisé une régulation avec correcteur proportionnel.

Question 1.9 En justifiant votre réponse, **conclure** sur la capacité de déplacement du robot pour BP1 et BC1 (uniquement pour le déplacement longitudinal) de la figure 4.



## 2. Assurer son autonomie de fonctionnement

L'objectif de cette partie est d'estimer la consommation énergétique du robot, d'en déterminer son autonomie afin de valider le choix de la batterie aux vues des exigences attendues.

La consommation énergétique du robot peut se répartir en trois principaux points :

- la consommation due aux moteurs pour le déplacement ;
- la consommation due à la tablette ;
- la consommation due à l'électronique embarquée (carte électronique, capteurs, etc.).

Pour affiner l'estimation de l'autonomie, le modèle multi-physique complet du robot de la figure 16 a permis d'obtenir l'intensité consommée par un moteur lors d'un fonctionnement test (figure 17) ainsi que l'allure de la décharge de la batterie (figures 18 et 19).

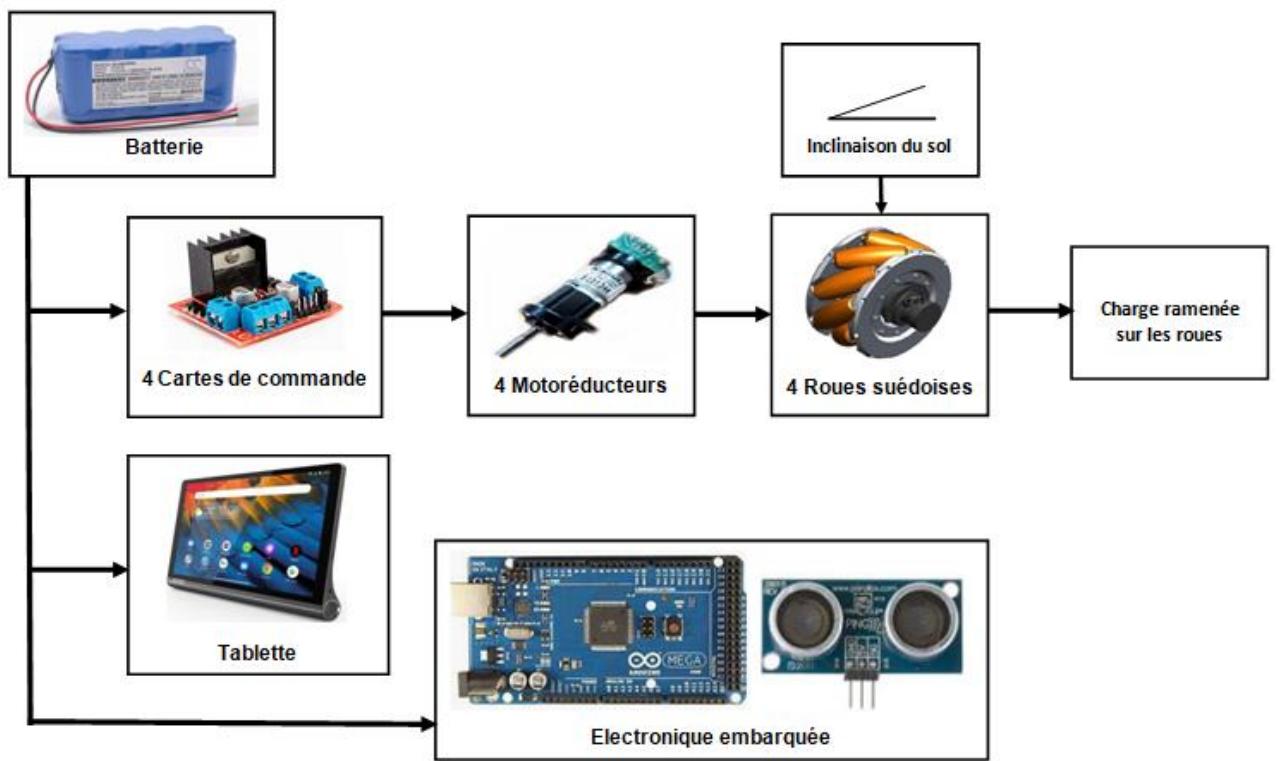


Figure 16 : schéma de principe simplifié du modèle multi-physique complet incluant les quatre chaînes fonctionnelles du robot.

À partir du modèle multi-physique présenté par le schéma de principe simplifié ci-dessus, la simulation suivante est réalisée ; celle-ci représente l'évolution de l'intensité consommée par **un moteur de l'essieu arrière** en ampère en fonction du temps en seconde pour un fonctionnement test.

De 0 s à 20 s le robot se déplace sur un sol plat, puis de 20 s à 40 s sur un sol en pente de 10 %, à partir de 40 s le robot s'arrête et reste en position fixe (seules la tablette et l'électronique embarquée consomment).

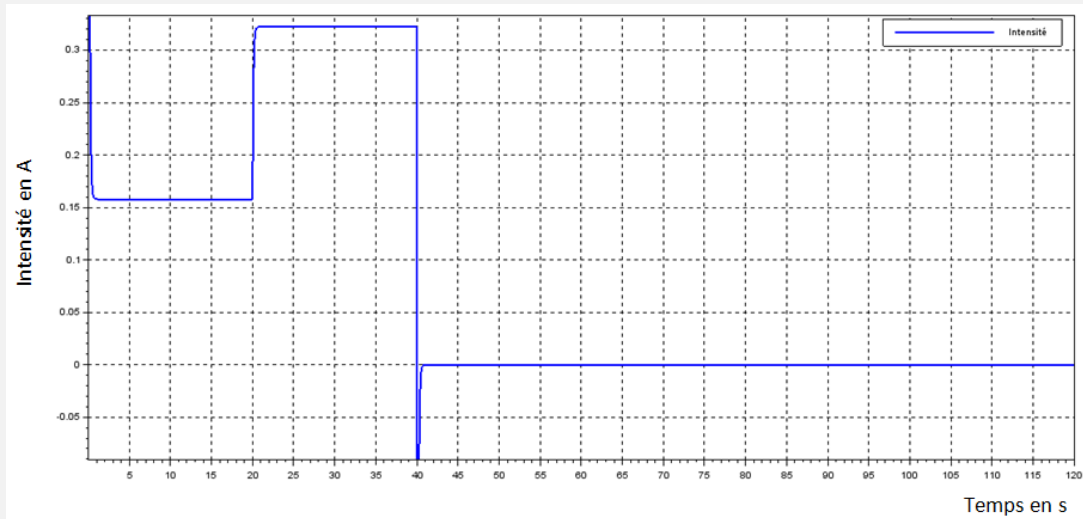


Figure 17 : intensité absorbée par un moteur de l'essieu arrière fournie par la modélisation pour un fonctionnement test et une tension d'alimentation de 14,8V.

Une étude sur l'essieu avant a permis de trouver une consommation en énergie pour les moteurs M1 et M2 :  $Co_{mot-avant} = 0,065 \text{ W}\cdot\text{h}$ .

**Question 1.10** *Déterminer la consommation en énergie des deux moteurs de l'essieu arrière (M3 et M4)  $Co_{mot-arrière}$  en  $\text{W}\cdot\text{h}$  pour un fonctionnement test à partir de la simulation de la figure 17, puis **déterminer** la consommation en énergie des 4 moteurs  $Co_{4mot}$  pour une période de 2 min correspondant au fonctionnement test de la figure 17.*

*Remarque : ne pas tenir compte de la pointe de courant au démarrage du moteur ni à  $t = 40 \text{ s}$  et assimiler le tracé à des segments de droite.*

Les caractéristiques de consommation des différents éléments du robot sont données ci-dessous, les accessoires fonctionnant en continu :

- tension d'alimentation des accessoires 14,8 V ;
- tablette intensité absorbée 0,9 A ;
- carte microprogrammée intensité absorbée 12 mA ;
- capteurs intensité absorbée 5 mA par chacun des deux capteurs ;
- les autres éléments ont une consommation négligeable.

**Question 1.11** *Déterminer la consommation des accessoires  $Co_{acc}$  (tablette, carte microprogrammée et capteurs) pour une durée de 2 min en  $\text{W}\cdot\text{h}$ . **En déduire** la consommation énergétique totale (moteurs + accessoires) dans le cas d'un fonctionnement test ramené à 1 h.*

**Question 1.12** *À partir des caractéristiques de la batterie dans le diagramme de bloc interne (voir DR1), **calculer** l'énergie stockée  $E_{batt}$  dans la batterie en  $\text{W}\cdot\text{h}$ , puis **déterminer** l'autonomie du robot de téléprésence en heure.*

Pour affiner avec plus de précision l'autonomie du robot, une tension minimum ( $U_{batmin}$ ) est nécessaire pour que le robot puisse fonctionner correctement. De plus, une batterie au lithium ne doit pas être déchargée de plus de 95 % pour conserver une durée de vie optimale. La tension minimum  $U_{batmin}$  est de 12,5 V.

À partir du schéma de principe du modèle multi-physique (figure 16), les simulations suivantes ont été réalisées ; la première représente l'évolution de la tension de la batterie en volt en fonction du temps en seconde et la seconde représente le pourcentage de charge lors d'un fonctionnement permanent du robot, à partir des estimations moyennes de consommation d'énergie déjà calculées.

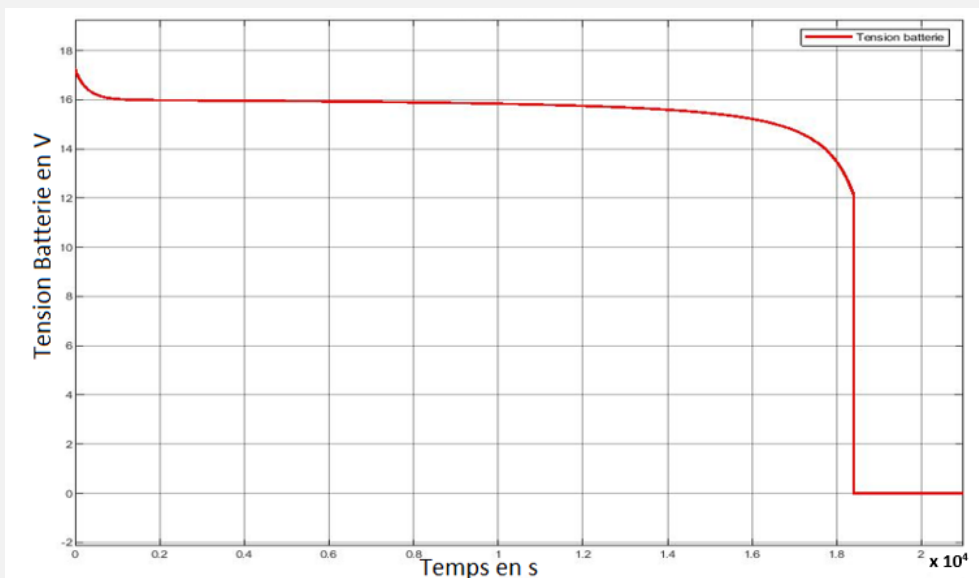


Figure 18 : décharge de la batterie en volt.

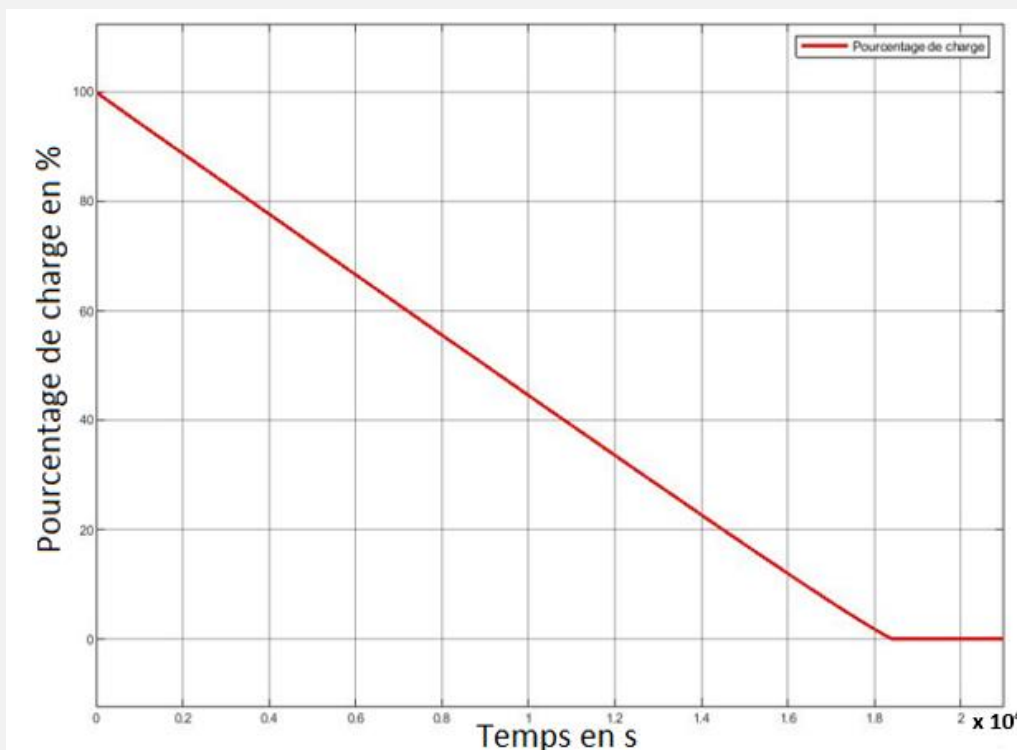


Figure 19 : décharge de l'énergie de la batterie en %.

Question 1.13 À partir des relevés (figure 18 et 19), **relever** les temps correspondant à  $U_{batmin.}$  et à une décharge de 95 % notés respectivement  $T_{U_{batmin}}$  et  $T_{95\%}$ . **Déterminer** la durée pendant laquelle la batterie peut assurer le fonctionnement du robot.

Question 1.14 La batterie choisie par le constructeur permet-elle dans le cas d'un déplacement type de garantir les exigences attendues ? **Justifier** la réponse et **proposer** des améliorations si nécessaire.

### 3. Piloter et orienter l'interface de communication

L'objectif de cette partie est de valider le choix de la technologie pour la transmission des informations entre la tablette et la chaîne d'information du robot, et de garantir une commande d'inclinaison de la tablette conforme au fonctionnement attendu.

La figure 20 représente les différentes liaisons informatiques reliant les différents éléments à l'établissement scolaire (voir aussi la figure 1 : schéma de principe de la téléprésence) :

- de chez lui, l'élève pilote le robot présent dans la salle de cours à partir de l'application dédiée (à partir de son ordinateur, d'une tablette ou de son téléphone) le robot présent dans la salle de cours ;
- dans l'établissement, une borne Wi-Fi permet les échanges entre la tablette et l'élève resté à son domicile ;
- au niveau du robot, la communication entre la tablette et la chaîne d'information du robot se fait en Bluetooth ;

#### Dans la salle de classe :

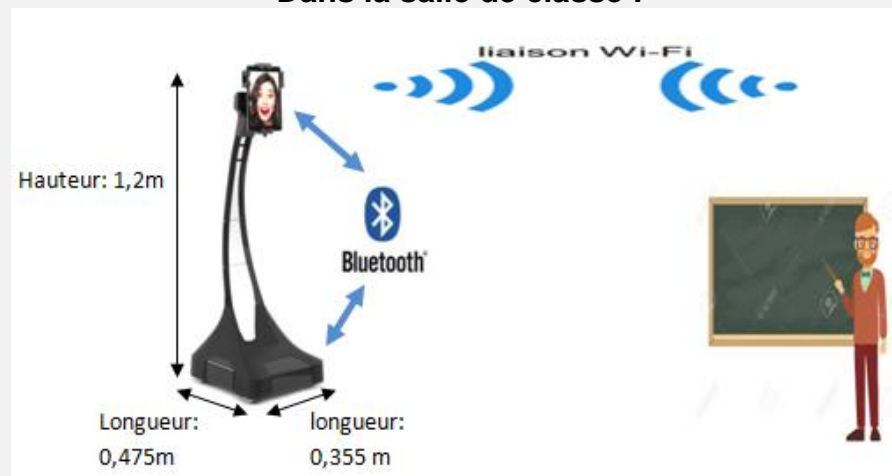


Figure 20 : liaisons informatiques reliant les différents éléments.

#### Présentation de l'interface Bluetooth :

- le Bluetooth est une technologie de **réseau personnel sans fil** (noté **WPAN** pour **Wireless Personal Area Network**). Il s'agit d'une technologie de réseaux sans fil à **faible portée** (quelques dizaines de mètres). Elle permet de relier plusieurs appareils entre eux sans liaison filaire, en utilisant les **ondes radio** comme support de transmission. La liaison radio fonctionne dans une bande de fréquence située autour de **2,45 GHz** ;
- vitesse de transmission  $1 \text{ Mb}\cdot\text{s}^{-1}$  ;
- faible consommation (de 7 fois à 10 fois moins que la WiFi) ;
- portée : la puissance d'émission permise pour un équipement Bluetooth est réglementée. Trois classes de puissance sont définies par la norme.

Classe	Puissance d'émission maximale	Portée en champ libre
1	100 mW	100 m
2	2,5 mW	40 m
3	1 mW	10 m


Module Bluetooth :	Bee TEL0023	
Bluetooth	V2.0	
Sensibilité	Livré avec une antenne intégrée	
Brochage	Compatible avec le XBEE	
Fréquence de fonctionnement	2,4 à 2,48 GHz	
Modulation	GFSK	
Puissance de transmission	= 4 dBm, classe-2	
Distance de transmission	20 à 40 m maximum (en espace libre)	
Débit de transfert de données	Synchrone : 1 Mbits/s	
Prise en charge des profils	Port série Bluetooth	
Tension d'entrée	+3,3 CC/50 mA	
Température fonctionnement	-20 à +55 °C	

Figure 21 : module Bluetooth implanté dans le robot.

Question 1.15 À partir de la présentation de l'interface Bluetooth ci-dessus, **valider** le choix de cette technologie au regard du critère de distance de transmission. **Indiquer** quel autre critère aussi prépondérant dans le choix d'une communication Bluetooth entre la tablette et le robot a été retenu par le concepteur.

L'inclinaison de la tablette du robot (voir figure 2) est réalisée par un servomoteur décrit figure 22.

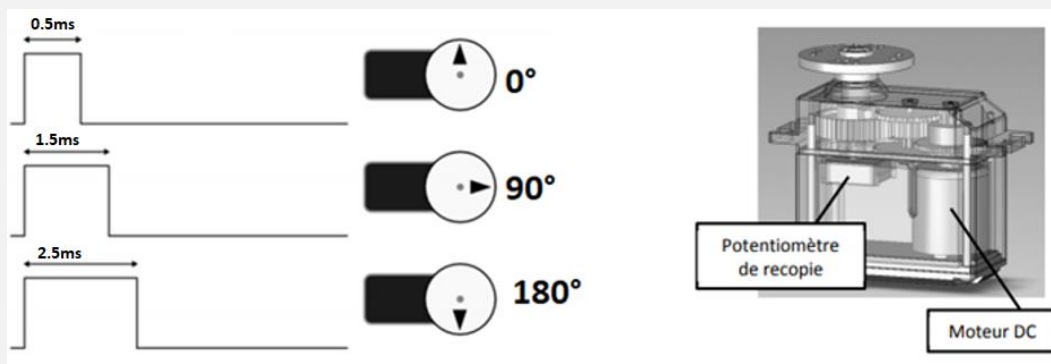


Figure 22 : servomoteur.

Un servomoteur est un système électromécanique, asservi en position, servant à actionner les parties mobiles d'un modèle-réduit, et répondant à une commande externe de type MLI (modulation à largeur d'impulsion) généralement transmise par une radiocommande. C'est la largeur de ces impulsions, générées périodiquement, qui détermine la position angulaire de l'axe de sortie. L'amplitude angulaire du servomoteur varie de 0° à 180° (pour une largeur d'impulsion qui varie entre 0,5 ms et 2,5 ms) pour une période de 20 ms.

Servomoteur	Référence : FEETECH FT835BL
Masse	72 g
Vitesse de fonctionnement	0,14 sec / 60 degrés (6 V)
Couple de calage	30 kg·cm/416,61 oz·in (6 V)
Tension de fonctionnement	6 V
Angle de fonctionnement	180 degrés
Impulsion requise	500 à 2500 µs
Longueur du fil du connecteur	30 cm

(Attention : les unités peuvent être différentes du système international.)

La largeur de l'impulsion de commande est codée sur 12 bits. La commande d'un angle nul correspond à 12 bits à 0 soit 000 en hexadécimal, la commande d'un angle de 180° correspond à 12 bits à 1 soit FFF en hexadécimal.

*Question 1.16 Déterminer l'angle de déplacement de la tablette le plus petit qu'il est possible de commander, en degrés, puis le temps mis par la tablette pour pivoter de 15°. Commenter cette valeur dans le contexte d'utilisation de la tablette.*

### Constitution d'une trame Bluetooth

Les données sont transmises par paquet. Chaque paquet est constitué des informations suivantes :

	Code d'accès 72 bits	En-tête 56 bits	Données 240 bits max
Code d'accès	72 bits	Ces bits permettent de configurer la liaison afin de synchroniser les différents composants Bluetooth. (Fréquence, code canal, code de recherche.....)	
En tête	56 bits	Ce champ contient dans l'ordre l'adresse de l'esclave (codée sur 3 bits), le type de paquets et des bits de contrôle (erreurs...)	
Données	240 bits max	La taille de la donnée est variable et peut aller jusqu'à 240 bits maximum.	

Dans une trame c'est le **LSB (bit de poids faible)** qui est transmis en premier.

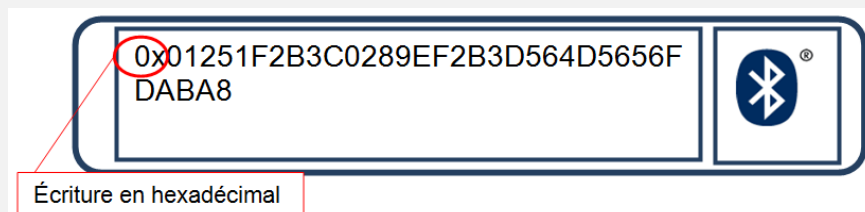


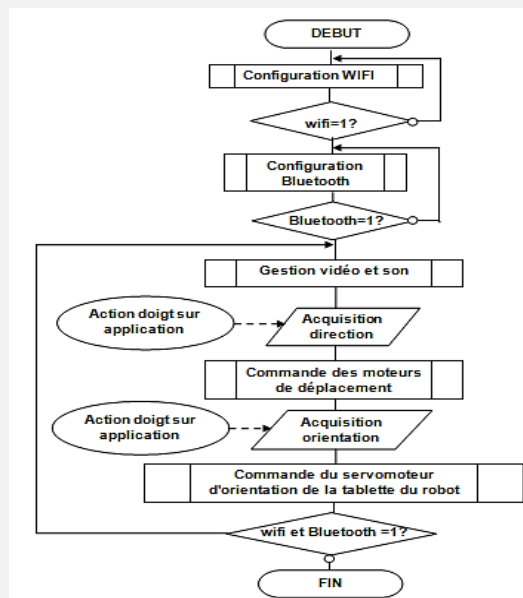
Figure 23 : relevé de la trame Bluetooth de commande d'inclinaison de la tablette.

L'information concernant la commande du sens de déplacement correspond au bit ( $bit_{sens}$ ) de poids faible du premier octet de données (si  $bit_{sens} = 0$  diminution de l'angle (déplacement négatif), si  $bit_{sens} = 1$  augmentation de l'angle (déplacement positif)).

Question 1.17 Après avoir éliminé de la trame ci-dessus les octets correspondants aux codes d'accès et à l'en-tête, **extraire** des données le sens de déplacement. **Calculer** la largeur de l'impulsion :  $T_{imp}$  à envoyer pour obtenir un positionnement de la tablette à 100 degrés.  
*Rappel : en hexadécimal un caractère représente 4 bits, un octet tient sur deux caractères hexadécimaux.*

La gestion globale du robot est représentée sur l'algorithme (figure 24). Le sous-programme <<Commande du servomoteur d'orientation de la tablette >> permet de gérer et commander l'inclinaison de la tablette.

Figure 24 : algorithme de gestion globale du système.



Question 1.18 Sur le document DR2, **compléter** l'algorithme qui permet de calculer le rapport cyclique (RC) en fonction de l'angle désiré (voir document servomoteur).  
 Voir DR2

$$\text{Rappel : } RC = \frac{\text{Largeur impulsion}}{\text{Période}} \text{ et période} = 20 \text{ ms}$$

Le déplacement étant très rapide, il faut un programme qui non seulement permet d'incrémenter et décrémenter l'angle de positionnement, mais aussi de ralentir le déplacement en faisant des temps de pause. Le programme du document DR3 permet de tester l'incrémenter, décrémenter et temps de pause sur la carte microcontrôleur au travers de deux boutons poussoirs sur deux entrées de la carte électronique qui représenteront les actions de l'élève sur la tablette.

Question 1.19 Sur le document DR3, **compléter** le sous-programme python <<commande du servomoteur d'orientation>> (voir figure 24 : algorithme de gestion globale du système) afin de diminuer l'angle d'orientation de la tablette. Sur la feuille de copie, **conclure synthétiquement** sur les exigences attendues pour orienter la tablette (voir figure 4).

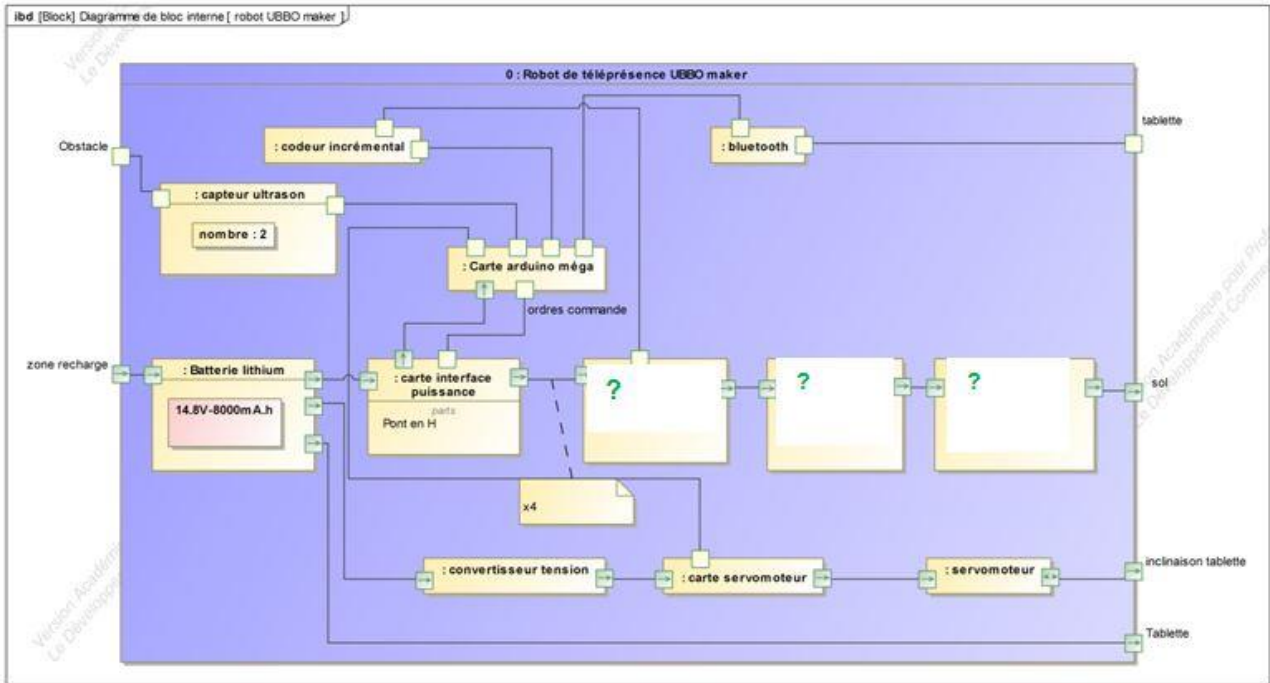
#### **Sous-partie 4. À traiter obligatoirement**

##### **Synthèse**

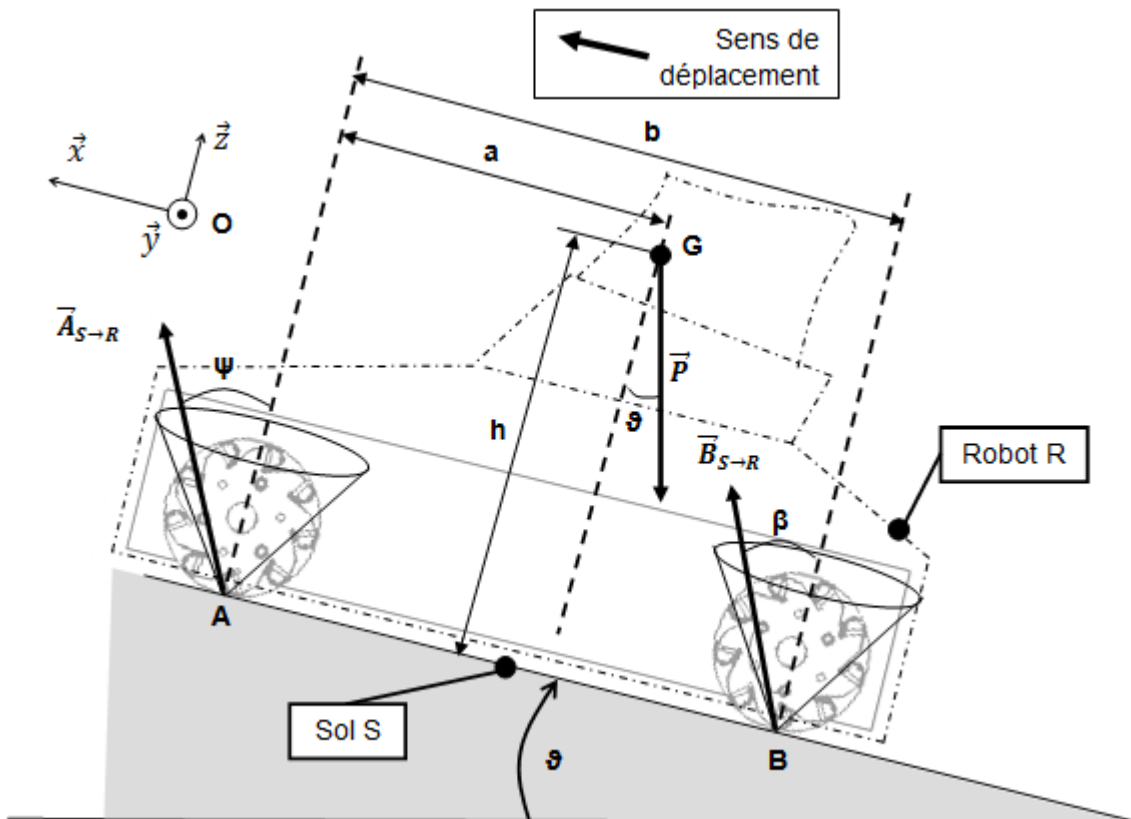
Question 1.20 En reprenant les objectifs d'étude des différentes parties, **compléter** les cases non grisées du document réponse DR4 (voir diagramme des exigences figure 4). **Quantifier** l'écart relatif sur la valeur finale de l'autonomie du robot entre d'une part les calculs effectués et les performances attendues.  
 Voir DR4



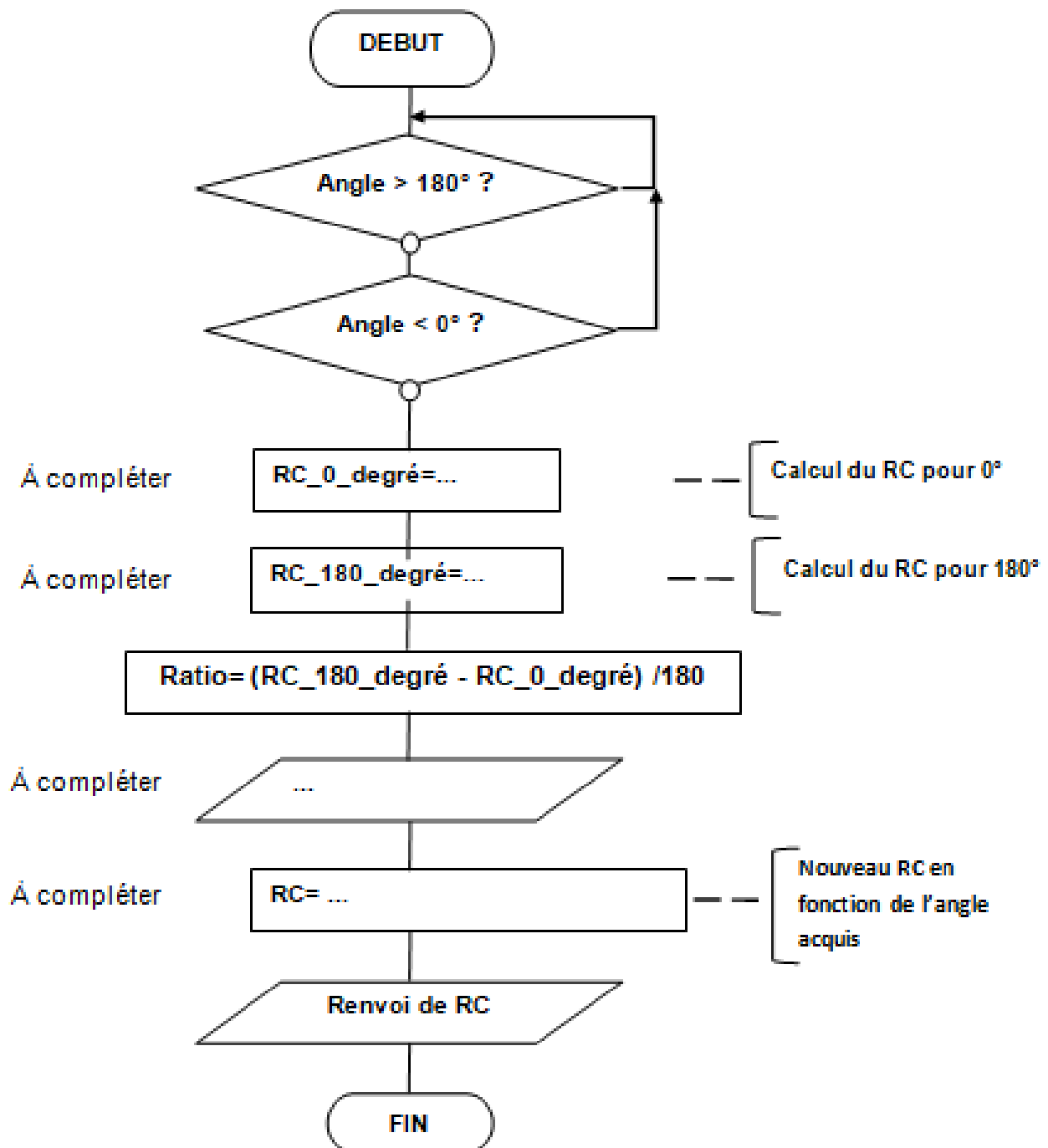
Question 1.2



Question 1.4



Question 1.18



## Question 1.19

```

1  """sous programme: Commande du servomoteur d'orientation de la tablette"""
2
3  import RPi.GPIO as GPIO          #importation de la librairie
4  import time                      #importation de la librairie time
5
6
7  angle=90                         #affectation de l'angle milieu de positionnement
8                                  #de la tablette (90°)
9
10 def rapport_cyclique_en_pourcentage(angle) :      #fonction qui permet de calculer le rapport cyclique
11                                                    #en % en fonction de l'angle
12
13     if angle > 180 or angle < 0 :
14         return False
15
16     RC_0_degre=2.5                 #rapport cyclique: impulsion pour 0° / temps
17                                   #pour un cycle= 0.5/20= 0.025=2.5%
18     RC_180_degre = 12.5           #rapport cyclique: impulsion pour 180° / temps
19                                   #pour un cycle= 2.5/20= 0.125=12.5%
20     ratio = (RC_180_degre - RC_0_degre)/180      #Calcul du ratio pour l'angle en pourcentage
21
22     RC_en_pourcentage = angle * ratio
23
24     return RC_0_degre + RC_en_pourcentage         #la fonction renvoie le nouveau rapport cyclique en %
25
26 GPIO.setmode(GPIO.BOARD)         #utilisation du mode de numérotation de la carte
27 GPIO.setwarnings(False)          #désactivé les avertissements
28
29 #utilisation de la broche 12 pour commander en PWM le servomoteur
30 pwm_gpio = 12                    #numéro du port de commande du servomoteur
31 frequence = 50                   #fréquence 50Hz
32 GPIO.setup(pwm_gpio, GPIO.OUT)   #affectation de la broche 12 en sortie
33 pwm = GPIO.PWM(pwm_gpio, frequence) #création d'une instance pwm
34
35 #utilisation des broches 15 et 16 pour récupérer l'appui sur les boutons de l'application
36 GPIO.setup(15, GPIO.IN)          #broche 15 affecter en entrée
37 GPIO.setup(16, GPIO.IN)          #broche 16 affecter en entrée
38
39 pwm.start(rapport_cyclique_en_pourcentage(angle)) #positionnement de la tablette à 90°
40 while True:
41     monter_tablette = GPIO.input(15) # renommage de l'entrée 15:ici recevra l'information d'augmenter l'angle
42     descendre_tablette = GPIO.input(16) # renommage de l'entrée 16:ici recevra l'information de diminuer l'angle
43     if (monter_tablette==True):
44         pwm.ChangeDutyCycle(rapport_cyclique_en_pourcentage(angle)) #pwm.ChangeDutyCycle permet de changer le rapport
45                                   #du cycle qui est le résultat de la fonction
46                                   #def rapport_cyclique_en_pourcentage(angle):
47                                   #RQ: Le temps d'execution de la commande pour 1° est
48                                   #de 20 ms (temps de la période) donc 3.6s pour 180°.
49                                   #incrémentation de l'angle
50                                   #temps de pause en seconde par degré: 0.020s
50                                   #(donc 3.6s de pause pour 180°).
51
52         angle= angle + 1
53         time.sleep (0.02)
54
55     elif ...                        #cas où l'on souhaite diminuer l'angle de la tablette
56
57     ...
58
59     ...
60
61 else:
62     pwm.ChangeDutyCycle(rapport_cyclique_en_pourcentage(angle)) #garde le rapport cyclique pour la position actuelle
63                                   #de la tablette
64
65     time.sleep (0.02)
66
67 #fin de programme
68 pwm.stop()
69 GPIO.cleanup()

```

## Question 1.20

\*Nature de l'écart : Attendu, Simulé, Mesuré A/S ou S/M ou A/M

Libellé de l'objectif	Nature de l'exigence	Exigence atteinte	Ecart relatif		Commentaires (pertinence de la solution retenue au regard des exigences attendues)
			Nature*	Valeur en %	
Permettre le déplacement du système dans son environnement	BP1				
	BC1		A/S		
Assurer son autonomie de fonctionnement	BP3		A/S		
Piloter et orienter l'interface de communication	BO1 BP4				