

Khepera IV

Manuel d'utilisation

BASCOUR Gwenaël – CAUSSE Jade – DA COSTA Yacine
NAYET Morgan – PARES Lucien

Table des matières

1	Description du robot	2
1.1	Les actionneurs	2
1.1.1	Roues	2
1.1.2	Haut-parleurs	2
1.1.3	Leds	2
1.2	Les capteurs	3
1.2.1	Odomètre	3
1.2.2	Gyroscope	3
1.2.3	Accéléromètre	3
1.2.4	Capteurs infrarouges – Bottom	4
1.2.5	Capteurs infrarouges	5
1.2.6	Microphone	5
1.2.7	Capteurs à ultrasons	6
1.2.8	Capteurs de chocs	7
2	Connexion du robot à l’ordinateur (sous Windows)	8
3	Programmes de manipulation du robot	11
3.1	Roues	11
3.2	Haut-parleurs	12
3.3	Leds	13
3.4	Odomètre	14
3.5	Gyroscope	15
3.6	Accéléromètre	18
3.7	Capteurs infrarouges – Bottom	20
3.8	Capteurs infrarouges	21
3.9	Microphone	22
3.10	Capteurs à ultrasons	24
3.11	Capteurs de chocs	25

1 Description du robot

1.1 Les actionneurs

1.1.1 Roues

Le robot possède deux roues gérées par deux moteurs ayant pour fonction de le mouvoir selon quatre directions (avant, arrière, droite, gauche) et une vitesse choisie. La vitesse du robot peut être incrémentée de 17 mm/s ou 67,8 mm/s en une fois selon l'option choisie. De plus, la vitesse minimale est de 0 mm/s, c'est-à-dire à l'arrêt jusqu'à un maximum de 813,8172 mm/s. La valeur par défaut de la vitesse lors du lancement du programme est de 135,6362 mm/s. Ces actionneurs sont situés sous le robot aux extrémités gauche et droite.

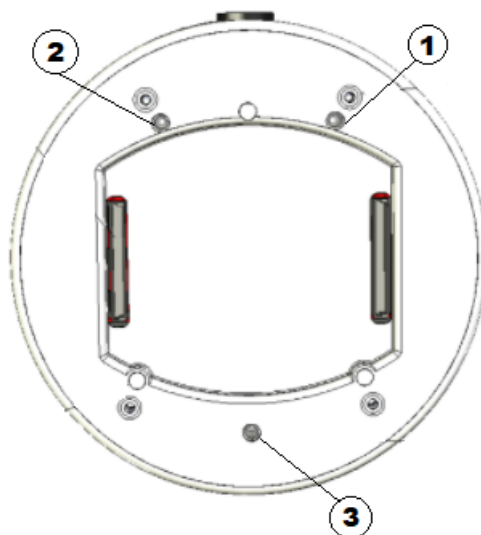
1.1.2 Haut-parleurs

Le robot dispose de haut-parleurs permettant l'écoute de fichier audio au format .wav. Ces derniers peuvent être désactivés si souhaité et leur volume est réglable en pourcentage, allant de 0% (cas dans lequel le haut-parleur est désactivé) jusqu'à 100%. Le volume est défini par défaut à 80%.

1.1.3 Leds

Le robot possède trois LEDs RGB positionnées sur le dessus de celui-ci. Elles sont entièrement contrôlables par le programmeur. Ces leds sont positionnées sur les sommets d'un triangle isocèle, elles peuvent donc servir à localiser et identifier le robot à l'aide d'une caméra, ainsi que de savoir où est l'avant.

FIGURE 1 – Position des LEDs



1.2 Les capteurs

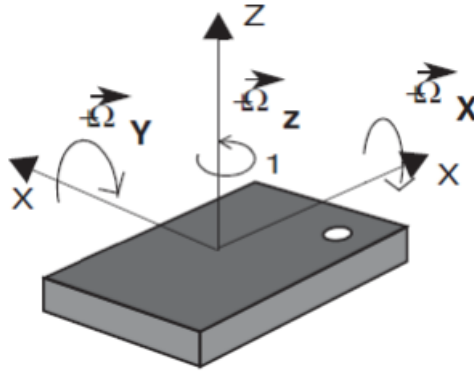
1.2.1 Odomètre

Le robot est équipé d'un odomètre permettant de mesurer la distance parcourue par le robot. Ce capteur prend en compte les révolutions des deux roues du robot, information utilisée pour calculer la distance parcourue.

1.2.2 Gyroscope

Le robot est équipé d'un gyroscope qui sert à mesurer la vitesse angulaire sur 3 axes, en deg/s.

FIGURE 2 – Axes de mesure du gyroscope (vus du dessus)

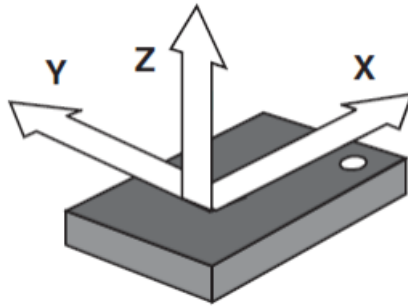


Le capteur mesure 10 valeurs par 10 valeurs, et ce toutes les 105ms.

1.2.3 Accéléromètre

Le robot est équipé d'un accéléromètre qui sert à mesurer l'accélération du robot sur 3 axes, en g. La valeur est positive sur l'axe X quand le robot avance, positive sur l'axe Y quand le robot va sur sa gauche, et négative sur l'axe Z avec la gravité.

FIGURE 3 – Axes de mesure de l'accéléromètre (vus du dessus)

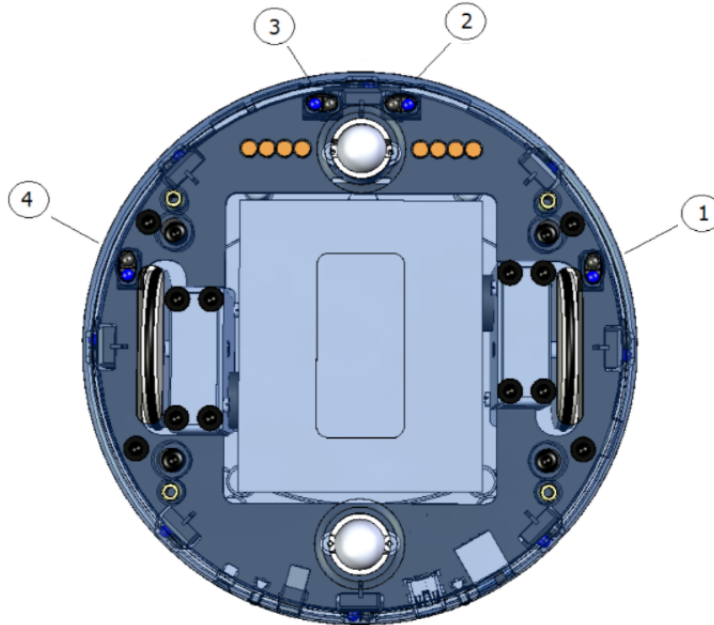


Le capteur mesure 10 valeurs par 10 valeurs, et ce toutes les 100ms.

1.2.4 Capteurs infrarouges – Bottom

Il y a 4 capteurs infrarouges se situant sous le robot Khepera IV. Chacun d'eux est capable de mesurer les ondes lumineuses ambiantes. Les capteurs infrarouges pulsent des ondes lumineuses infrarouges et captent la réflexion de celles-ci sur l'objet se trouvant sous chaque capteur.

FIGURE 4 – Position des capteurs infrarouges sous le robot

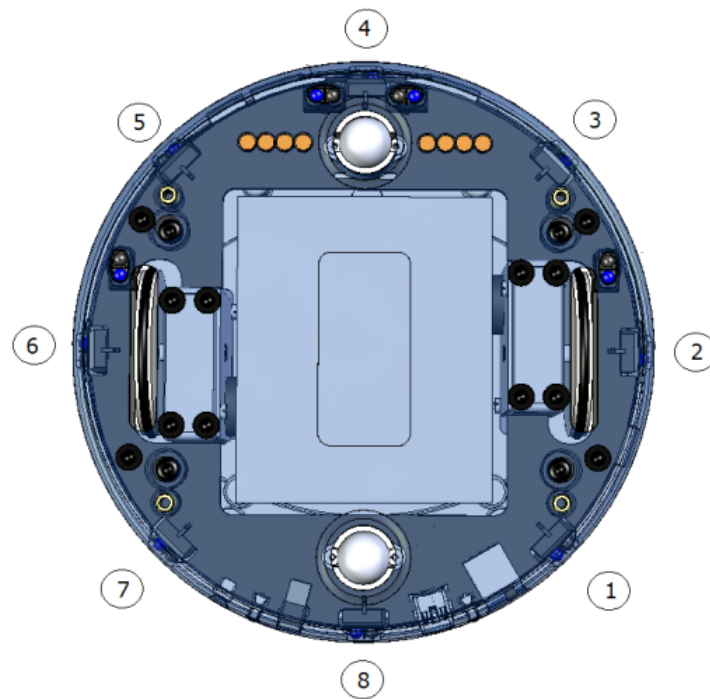


1.2.5 Capteurs infrarouges

Le robot possède 8 capteurs infrarouges situés tout autour de lui, chacun espacé de 45° (cf. Figure 5), lui permettant de détecter des obstacles ou alors de mesurer la luminosité ambiante. Il est capable de détecter des objets situés entre 2mm et 250mm en fonction de la luminosité du lieu et de la couleur de l'objet.

Les valeurs renvoyées par les capteurs sont normalement comprises entre 0 et 1023. Plus la valeur est élevée plus un objet est proche du capteur. Si valeur s'approche de 0 alors aucun objet ne se trouve à proximité du robot.

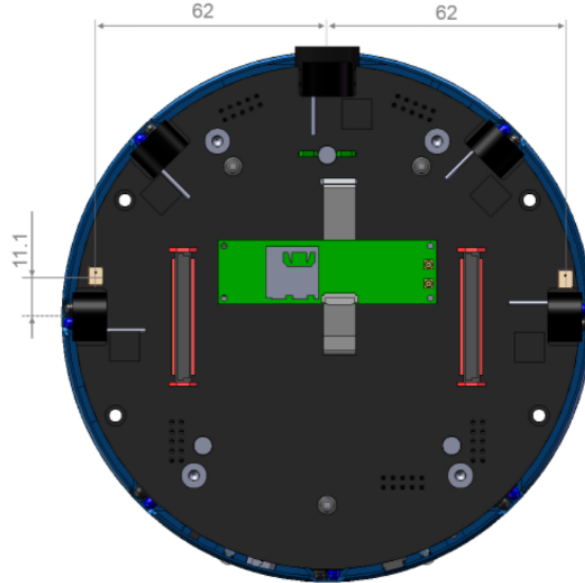
FIGURE 5 – Position des capteurs infrarouges sur les côtés du robot



1.2.6 Microphone

Le robot possède des microphones ayant pour fonction d'enregistrer des sons durant une durée donnée. Le volume de ces microphones peut être choisi entre 0 et 100 ou simplement initialisé par défaut, c'est-à-dire à 80 sur 100. De plus, il est possible de sauvegarder le fichier créé au format .wav et de choisir son nom. Ces deux microphones sont placés sous le robot, aux extrémités gauche et droite.

FIGURE 6 – Position des microphones sur le robot

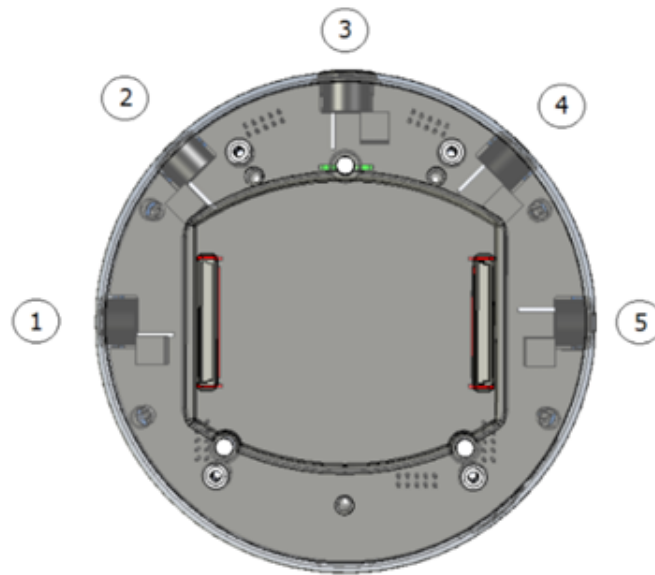


1.2.7 Capteurs à ultrasons

Le robot possède 5 capteurs à ultrasons, chacun espacé de 45° (cf. Figure 7), lui permettant de mesurer la distance qui le sépare d'un objet (ex : mobilier, mur). Les capteurs à ultrasons émettent à intervalles réguliers de courtes impulsions sonores à haute fréquence. Ces impulsions se propagent dans l'air à la vitesse du son. Lorsqu'elles rencontrent un objet, elles se réfléchissent et reviennent sous forme d'écho aux capteurs. Celui-ci calcule alors la distance le séparant de la cible sur la base du temps écoulé entre l'émission du signal et la réception de l'écho. La fréquence de ces capteurs est d'environ 40kHz.

Ces capteurs permettent de détecter un objet se situant entre 25 cm et 250 cm, avec une marge d'erreur d'approximativement 2 cm.

FIGURE 7 – Position des capteurs à ultrasons sur le robot



1.2.8 Capteurs de chocs

Le robot ne comporte pas de capteurs de chocs mais, grâce aux capteurs infrarouges situés sur les côtés du robot, il est possible de programmer des comportements de prévention des chocs.

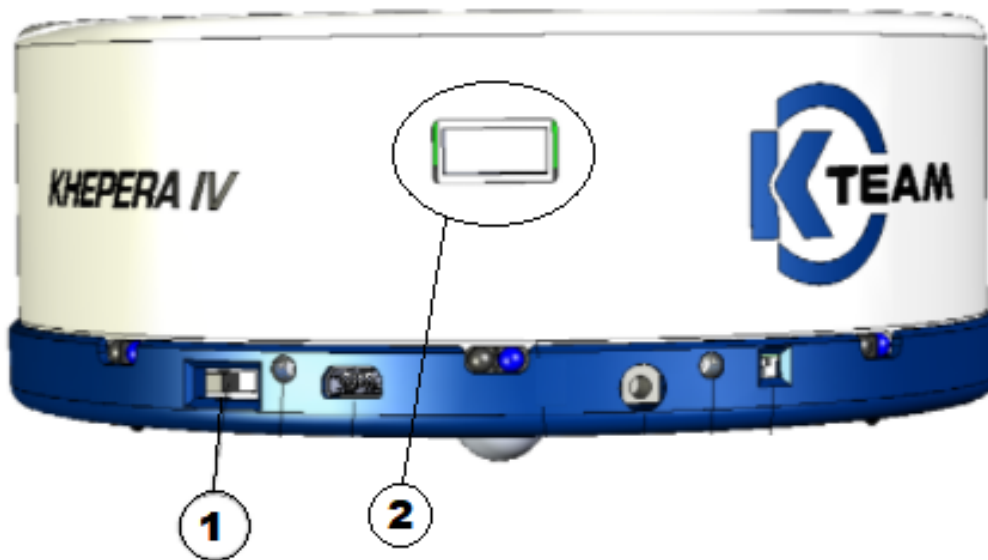
2 Connexion du robot à l'ordinateur (sous Windows)

Dans un premier temps, il faudra s'assurer que le logiciel TeraTerm soit correctement installé sur votre ordinateur. Si TeraTerm n'est pas installé, télécharger le fichier `teraterm.exe` depuis le lien suivant :

<https://osdn.net/projects/ttssh2/releases/>

Installer ensuite le logiciel en suivant les instructions pour une installation classique. Une fois le logiciel installé, allumez le robot avec l'interrupteur On/Off situé sur l'arrière (cf. Figure : Connectiques robot(1)) et connectez ensuite le robot à votre ordinateur en utilisant le câble USB fourni (cf. Figure Connectiques robot(2)).

FIGURE 8 – Connectiques du robot



Lancer maintenant l'application TeraTerm. Une fenêtre indiquant la mise en place d'une nouvelle connexion. Dans notre cas, ce sera l'option "**Série**" qui sera à prendre en compte. Vérifiez bien que le port utilisé soit : **COM3 : USB Serial Port (COM3)** (cf. Figure : Nouvelle Connexion TeraTerm)

FIGURE 9 – Nouvelle Connexion TeraTerm

Tera Term: Nouvelle connexion

☐ TCP/IP Hôte: myhost.example.com

☒ Historique

Service ☐ Telnet TCP port#: 22

☒ SSH SSH version: SSH2

☐ Autre IP version: AUTO

☒ Série Port: COM3: USB Serial Port (COM3)

OK Effacer Aide

Une fois ceci fait, rendez-vous dans l'onglet **"Configuration : Port Serie..."**. Une nouvelle fenêtre de configuration apparaîtra, assurez vous que les réglages soient similaires à la figure ci-dessous (cf. Figure : Configuration du Port Serie), puis cliquez sur **"New Setting"**

FIGURE 10 – Configuration du port série

Tera Term: Serial port setup and connection

Port: COM3

Vitesse: 115200

Données: 8 bit

Parité: none

Stop bits: 1 bit

Ctrl. de flux: none

Délai de transmission

0 msec/car 0 msec/ligne

New setting

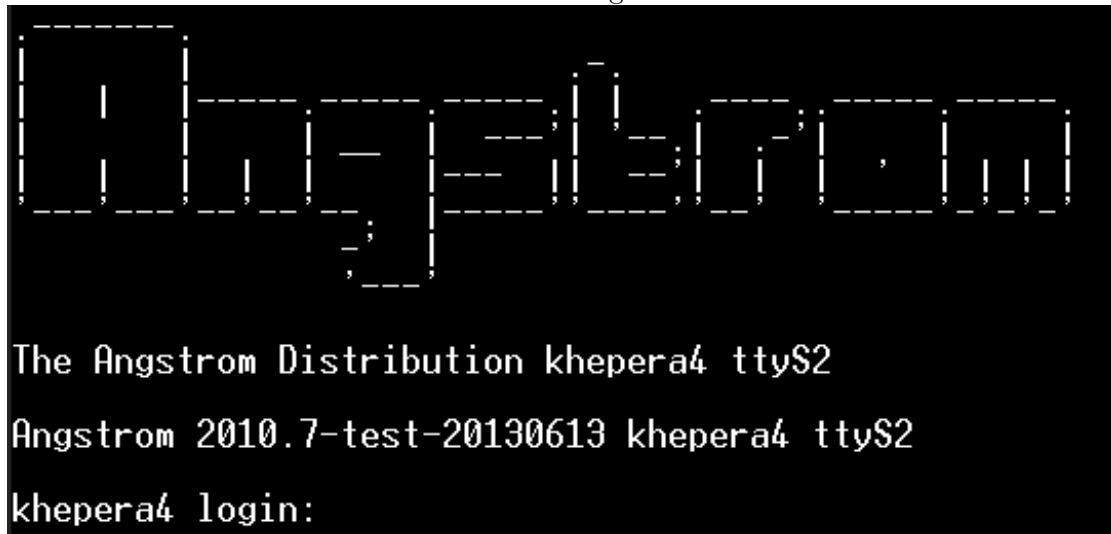
Effacer

Aide

Device Friendly Name: USB Serial Port (COM3)
 Device Instance ID: FTDIBUS\VID_0403+PID_6015+DJ002YBFA10C
 Device Manufacturer: FTDI
 Provider Name: FTDI
 Driver Date: 7-5-2021
 Driver Version: 2.12.36.4

Le robot terminera ensuite son initialisation et vous aurez un affichage vous demandant le login afin de vous connecter au robot et avoir accès à ses fichiers. Vous devrez alors saisir le login **"root"** (cf. Figure : login robot) et appuyer sur la touche **"entrée"** de votre clavier. Vous êtes maintenant connecté au robot. Vous pouvez dès à présent vous rendre dans la section **"Programmes de manipulation du robot"**, pour pouvoir exécuter les différents programmes de tests disponibles pour vérifier le bon fonctionnement des capteurs et actionneurs du robot Khepera IV.

FIGURE 11 – Login robot



3 Programmes de manipulation du robot

Tous les programmes de tests créés sont présents dans un dossier `testProgram` sur le robot. Ce dossier est accessible via la commande `cd testProgram/` et chaque programme de test est exécutable à partir de ce dossier à l'aide de la commande `./xxxxxx`, où `xxxxxx` est le nom du programme à lancer. La liste des programmes est accessible via la commande `ls -d test_*`.

3.1 Roues

```
./test_wheels
```

Une fois le test lancé, le programme tente dans un premier temps d'accéder au microcontrôleur `pic` en initialisant la librairie `libkhepera` et en se connectant au robot. Si l'accès à cette librairie, permettant de récupérer les fonctions et données nécessaires au lancement du test, ou la connexion au robot échouent, un message d'erreur est affiché sur le terminal et le programme se ferme.

Si aucun problème n'est à déplorer, le programme commence tout d'abord par afficher un menu sur le terminal avec les informations permettant de manipuler le robot. C'est-à-dire choisir la direction des roues, la vitesse de déplacement du robot, et la possibilité de quitter le test.

Ensuite, le programme attend que l'utilisateur presse un des boutons permettant la manipulation du robot et selon la touche enfoncée, l'action liée va être effectuée. Par exemple, pour le déplacement du robot selon une direction choisie, si l'utilisateur presse la touche "z" alors le programme va initialiser la vitesse par défaut, si l'utilisateur ne l'a pas changée auparavant, et orienter les deux roues du robot vers l'avant et commencer à déplacer le robot selon cette direction. Les trois autres directions, à savoir arrière, droite et gauche, sont initialisées en suivant ce même schéma.

En appuyant sur la touche "a", le programme va couper la vitesse des deux moteurs. Pour contrôler la vitesse de déplacement, on peut incrémenter et décrémenter selon deux valeurs, soit 16,95 mm/s soit 67,82 mm/s. Les touches "b" et "n" se chargent de l'incrément et décrément par 16,95 mm/s. Lorsqu'une des deux touches est pressée, le programme vérifie que la vitesse actuelle du robot ne dépasse pas la vitesse maximale ou minimale selon le cas puis va changer la vitesse de déplacement du robot. Ce schéma est le même pour l'incrément et décrément par 67,82 mm/s avec les touches "h" et "j". Enfin, en appuyant sur la touche "x", le programme se ferme tout en remettant le robot dans son état initial, c'est-à-dire en coupant la vitesse des moteurs.

FIGURE 12 – Exemple d'utilisation du programme de test des roues & moteurs

```
-----  
TEST ROUES & MOTEURS  
-----  
  
Direction avec les touches ZQSD, arret avec A  
Augmentation de la vitesse (+25) avec B, Reduction (-25) avec N  
Augmentation de la vitesse (+100) avec H, Reduction (-100) avec J  
Quitter le programme avec X.  
  
Vitesse actuelle : 225 (152 mm/s)  
Vitesse actuelle : 200 (135 mm/s)  
Vitesse actuelle : 300 (203 mm/s)  
Vitesse actuelle : 200 (135 mm/s)
```

3.2 Haut-parleurs

```
./test_speakers [filename.wav]
```

Ce programme possède deux mode d'exécution. Le premier sans spécifier de fichier audio, dans ce cas là ce sera le fichier audio par défaut "beep.wav" qui sera lu. Sinon, il est possible de tester deux autres sons en remplaçant [filename.wav] par le nom du fichier. Les deux autres fichiers disponibles sont "beep2.wav" et "beep3.wav".

Une fois le test lancé, le programme tente dans un premier temps d'accéder au microcontrôleur pic en initialisant la librairie libkhepera et en se connectant au robot. Si l'accès à cette librairie permettant de récupérer les fonctions et données nécessaires au lancement du test ou la connexion au robot échouent, un message d'erreur est affiché sur le terminal et le programme se ferme.

Si aucun problème n'est à déplorer, le programme initialise et configure le système de son du robot et désactive tous les capteurs à ultrasons pour éviter les problèmes qui pourraient survenir lors de l'activation des haut-parleurs.

Il demande ensuite de saisir le volume du haut-parleur gauche ainsi que du haut-parleur droit du robot, le volume est compris entre 0% et 100%.

Le programme regarde ensuite si un fichier a été mentionné en argument, si tel est le cas il va vérifier qu'il existe et ensuite va faire le lire, on pourra ainsi entendre le son souhaité. Dans le cas où le fichier n'existerait pas, le son par défaut sera joué. (cf. Figures 13 & 14)

Une fois la lecture terminée, le programme remet le robot dans son état initial et réactive les capteurs à ultrasons.

FIGURE 13 – Exemple d'utilisation du programme des haut-parleurs : sans fichier choisi

```
-----  
TEST HAUT-PARLEURS  
-----  
  
Entrez le volume des haut-parleurs (entre 0 et 100)  
Haut-parleur gauche : 80  
Haut-parleur droit : 80  
  
Aucun fichier choisi. Utilisation du fichier par default.  
Le fichier lu sera : beep.wav  
Lecture en cours...  
Lecture terminee.
```

FIGURE 14 – Exemple d'utilisation du programme des haut-parleurs : avec fichier choisi

```
root@khepera4:~/testProgram# ./test_speakers beep2.wav  
-----  
TEST HAUT-PARLEURS  
-----  
  
Entrez le volume des haut-parleurs (entre 0 et 100)  
Haut-parleur gauche : 60  
Haut-parleur droit : 60  
  
Le fichier lu sera : beep2.wav  
Lecture en cours...  
Lecture terminee.
```

3.3 Leds

```
./test_leds
```

Une fois le test lancé, le programme tente dans un premier temps d'accéder au microcontrôleur pic en initialisant la librairie libkhepera et en se connectant au robot. Si l'accès à cette librairie permettant de récupérer les fonctions et données nécessaires au lancement du test ou la connexion au robot échouent, un message d'erreur est affiché sur le terminal et le programme se ferme.

Si aucun problème n'est à déplorer, le programme affiche sur le terminal un message demandant à l'utilisateur de choisir les codes décimaux RGB des trois leds présentes sur le robot. Ces codes sont chacun composés de trois valeurs allant de 0 à 255.

Si l'utilisateur n'a pas écrit toutes les valeurs des codes décimaux RGB, le programme renvoie un message d'erreur sur le terminal et éteint les leds en remplaçant les valeurs entrées par 0. Sinon, le programme affiche sur le terminal les valeurs choisies par l'utilisateur pour chaque led, puis affecte les valeurs aux leds et active les leds avec les couleurs choisies par l'utilisateur durant cinq secondes. Enfin, le programme éteint les leds et leur affecte la valeur 0 puis se ferme.

FIGURE 15 – Exemple d'utilisation du programme des Leds

```
-----  
TEST DES LEDS RGB  
-----  
  
Entrez les valeurs RGB des trois leds a la suite  
(Ex : 120 0 0 0 122 0 0 0 120)  
  
Saisie : 50 12 10 0 60 0 55 0 48  
  
RGB :  
      - LED 1 : 50 12 10  
      - LED 2 : 0 60 0  
      - LED 3 : 55 0 48
```

3.4 Odomètre

```
./test_odometer
```

Une fois le test lancé, le programme tente dans un premier temps d'accéder au microcontrôleur pic en initialisant la librairie libkhepera et en se connectant au robot. Si l'accès à cette librairie, permettant de récupérer les fonctions et données nécessaires au lancement du test, ou la connexion au robot échouent, un message d'erreur est affiché sur le terminal et le programme se ferme.

Si aucun problème n'est à déplorer, le programme commence tout d'abord par afficher un menu sur le terminal avec les informations permettant de manipuler le robot. C'est-à-dire choisir la direction des roues, la vitesse de déplacement du robot, la possibilité de quitter le test ainsi que la distance parcourue par le robot lors du test.

Ensuite, le programme attend que l'utilisateur presse un des boutons permettant la manipulation du robot et selon la touche enfoncée, l'action liée va être effectuée. Par exemple, pour le déplacement du robot selon une direction choisie, si l'utilisateur presse

la touche "z" alors le programme va initialiser la vitesse par défaut, si l'utilisateur ne l'a pas changée auparavant, et orienter les deux roues du robot vers l'avant et commencer à déplacer le robot selon cette direction. Les trois autres directions, à savoir arrière, droite et gauche, sont initialisées en suivant ce même schéma.

En appuyant sur la touche "a", le programme va couper la vitesse des deux moteurs. Pour contrôler la vitesse de déplacement, on peut incrémenter et décrémenter selon deux valeurs, soit 16,95 mm/s, soit 67,82 mm/s. Les touches "b" et "n" se chargent de l'incrément et décrément par 16,95 mm/s. Lorsqu'une des deux touches est pressée, le programme vérifie que la vitesse actuelle du robot ne dépasse pas la vitesse maximale ou minimale selon le cas puis va changer la vitesse de déplacement du robot. Ce schéma est le même pour l'incrément et décrément par 67,82 mm/s avec les touches "h" et "j". Ajouté à cela, afin de tester l'odomètre du robot, lorsque les touches "z" ou "s" sont pressées, le programme va incrémenter une variable pour la distance parcourue et l'afficher continuellement sur le terminal jusqu'à ce que le programme soit fermé.

Enfin, en appuyant sur la touche "x", le programme se ferme tout en remettant le robot dans son état initial, c'est-à-dire en coupant la vitesse des moteurs.

FIGURE 16 – Exemple d'utilisation du programme de l'odomètre

```
-----
TEST ODOMETRE
-----

Direction avec les touches ZQSD, arrêt avec A
Augmentation de la vitesse (+25) avec B, Reduction (-25) avec N
Augmentation de la vitesse (+100) avec H, Reduction (-100) avec J
Quitter le programme avec X.

Distance parcourue : 91 mm
```

3.5 Gyroscope

```
./test_gyroscope
```

Une fois le test lancé, le programme tente dans un premier temps d'accéder au microcontrôleur pic en initialisant la librairie libkhepera et en se connectant au robot. Si l'accès à cette librairie, permettant de récupérer les fonctions et données nécessaires au lancement du test, ou la connexion au robot échouent, un message d'erreur est affiché sur le terminal et le programme se ferme.

Si aucun problème n'est à déplorer, le programme commence tout d'abord par afficher sur le terminal un menu présentant deux modes pour manipuler le gyroscope. Le premier mode affiche une seule valeur par axe, plus précisément la moyenne des dix valeurs récupérées par le gyroscope pour chaque axe. Le second mode quant à lui, affiche toutes ces valeurs ainsi que leur moyenne pour chaque axe. L'utilisateur est invité à choisir un

des deux modes en pressant soit la touche "a" pour le premier mode soit la touche "b" pour le second mode.

FIGURE 17 – Exemple d'utilisation du programme du gyroscope : Choix du mode

```
-----  
TEST DU GYROSCOPE  
-----  
  
Quel mode souhaitez-vous choisir ?  
a -> Afficher la moyenne des valeurs du gyroscope  
b -> Afficher les 10 valeurs recuperees par le gyroscope et leur moyenne  
  
Choix : 
```

Avant d'afficher la vue liée à l'un des deux modes choisi par l'utilisateur, le programme commence par récupérer les dix valeurs pour chaque axe x y et z dans un tableau.

Dans le premier mode, affichant seulement la moyenne des valeurs pour chaque axe, le programme lit les dix premières valeurs contenues dans le tableau pour l'axe x et calcule la moyenne de ces valeurs puis l'affiche à l'écran. Le même schéma est répété pour les deux autres axes. L'utilisateur peut décider à tout moment de quitter le test en appuyant sur n'importe quelle touche.

FIGURE 18 – Exemple d'utilisation du programme du gyroscope : Choix "a"

```
-----  
TEST DU GYROSCOPE  
-----  
  
Moyenne des 10 valeurs de X :  
    0.5  
  
Moyenne des 10 valeurs de Y :  
   -1.8  
  
Moyenne des 10 valeurs de Z :  
    0.7  
  
Appuyez sur une touche pour stopper le programme.  

```

Dans le second mode, affichant les dix valeurs ainsi que leur moyenne pour chaque axe, le programme lit les dix premières valeurs contenus dans le tableau pour l'axe x puis

les affiche à la suite sur l'écran. La moyenne de ces valeurs est ensuite calculée et affichée sur le terminal. Le même schéma est répété pour les deux autres axes et l'ensemble des valeurs pour chaque axe est ensuite rafraîchi toutes les 105 ms. L'utilisateur peut décider à tout moment de quitter le test en appuyant sur n'importe quelle touche.

FIGURE 19 – Exemple d'utilisation du programme du gyroscope : Choix "b"

```
-----  
TEST DU GYROSCOPE  
-----  
  
Valeurs de X [deg/s] :  
  0.4  
  0.7  
  0.8  
  0.9  
  0.7  
  0.5  
  0.5  
  0.7  
  0.5  
  0.4  
Moyenne des 10 valeurs de X :  
  0.6  
  
Valeurs de Y [deg/s] :  
 -1.5  
 -1.0  
 -1.1  
 -1.8  
 -2.2  
 -1.8  
 -1.5  
 -1.4  
 -1.6  
 -1.9  
Moyenne des 10 valeurs de Y :  
 -1.6  
  
Valeurs de Z [deg/s] :  
  0.7  
  1.1  
  1.1  
  0.7  
  0.4  
  0.5  
  0.5  
  0.4  
  0.6  
  0.7  
Moyenne des 10 valeurs de Z :  
  0.7  
  
Appuyez sur une touche pour stopper le programme.
```

3.6 Accéléromètre

```
./test_accelerometer
```

Une fois le test lancé, le programme tente dans un premier temps d'accéder au microcontrôleur pic en initialisant la librairie libkhepera et en se connectant au robot. Si l'accès à cette librairie, permettant de récupérer les fonctions et données nécessaires au lancement du test, ou la connexion au robot échouent, un message d'erreur est affiché sur le terminal et le programme se ferme.

Si aucun problème n'est à déplorer, le programme commence tout d'abord par afficher sur le terminal un menu présentant deux modes pour manipuler l'accéléromètre. Le premier mode affiche une seule valeur par axe, plus précisément la moyenne des dix valeurs récupérées par l'accéléromètre pour chaque axe. Le second mode quant à lui, affiche toutes ces valeurs ainsi que leur moyenne pour chaque axe. L'utilisateur est invité à choisir un des deux modes en pressant soit la touche «a» pour le premier mode soit la touche «b» pour le second mode.

FIGURE 20 – Exemple d'utilisation du programme de l'accéléromètre : Choix du mode

```
-----  
TEST ACCELEROMETRE  
-----  
  
Quel mode souhaitez-vous choisir ?  
a -> Afficher la moyenne des valeurs de l'accelerometre  
b -> Afficher les 10 valeurs recuperees par l'accelerometre et leur moyenne  
Choix : █
```

Avant d'afficher la vue liée à l'un des deux modes choisi par l'utilisateur, le programme commence par récupérer les dix valeurs pour chaque axe x y et z dans un tableau.

Dans le premier mode, affichant seulement la moyenne des valeurs pour chaque axe, le programme lit les dix premières valeurs contenues dans le tableau pour l'axe x et calcule la moyenne de ces valeurs puis l'affiche à l'écran. Le même schéma est répété pour les deux autres axes. L'utilisateur peut décider à tout moment de quitter le test en appuyant sur n'importe quelle touche.

FIGURE 21 – Exemple d'utilisation du programme de l'accéléromètre : Mode "a"

```
-----  
TEST ACCELEROMETRE  
-----  
  
Moyenne des 10 valeurs de X :  
    0.01  
  
Moyenne des 10 valeurs de Y :  
   -0.00  
  
Moyenne des 10 valeurs de Z :  
    0.96  
  
Appuyez sur une touche pour stopper le programme.
```

Dans le second mode, affichant les dix valeurs ainsi que leur moyenne pour chaque axe, le programme lit les dix premières valeurs contenus dans le tableau pour l'axe x puis les affiche à la suite sur l'écran. La moyenne de ces valeurs est ensuite calculée et affichée sur le terminal. Le même schéma est répété pour les deux autres axes et l'ensemble des valeurs pour chaque axe est ensuite rafraîchi toutes les 105 ms. L'utilisateur peut décider à tout moment de quitter le test en appuyant sur n'importe quelle touche.

FIGURE 22 – Exemple d'utilisation du programme de l'accéléromètre : Mode "b"

```
-----
TEST ACCELEROMETRE
-----

Valeurs de X [g] :
0.02
0.02
0.02
0.02
0.02
0.02
0.02
0.02
0.02
0.02
0.01
Moyenne des 10 valeurs de X :
0.02

Valeurs de Y [g] :
0.00
0.00
0.00
0.00
0.00
-0.00
0.00
0.00
0.00
0.00
Moyenne des 10 valeurs de Y :
-0.00

Valeurs de Z [g] :
0.97
0.96
0.96
0.96
0.96
0.96
0.96
0.96
0.96
0.96
Moyenne des 10 valeurs de Z :
0.96

Appuyez sur une touche pour stopper le programme.
```

3.7 Capteurs infrarouges – Bottom

test_ir_bottom

Une fois le test lancé, le programme tente dans un premier temps d'accéder au microcontrôleur pic en initialisant la librairie libkhepera et en se connectant au robot. Si

l'accès à cette librairie, permettant de récupérer les fonctions et données nécessaires au lancement du test, ou la connexion au robot échouent, un message d'erreur est affiché sur le terminal et le programme se ferme.

Si aucun problème n'est à déplorer, le programme commence tout d'abord par récupérer les valeurs des capteurs dans un tampon, puis attribue ces valeurs à chaque capteur correspondant. Ensuite, le programme affiche sur le terminal chaque valeur liée à un capteur à la suite, et rafraîchi continuellement les valeurs reçues.

Pour vérifier le bon fonctionnement des capteurs, il est possible d'approcher des objets des différents capteurs présents sous le robot afin de vérifier que les valeurs des capteurs concernés par la présence de l'objet augmentent, ou diminuent si l'objet est éloigné de ces capteurs.

Enfin, l'utilisateur est invité, s'il le souhaite, à presser n'importe quelle touche pour quitter le programme.

FIGURE 23 – Exemple d'utilisation du programme des capteurs infrarouges – Bottom

```
-----  
TEST CAPTEURS INFRAROUGES SOUS LE ROBOT  
-----  
  
La valeur du capteur droit: 959  
La valeur du capteur gauche est 960  
La valeur du capteur avant gauche est 958  
La valeur du capteur avant droit est 958  
  
Appuyez sur une touche pour stopper le programme.
```

3.8 Capteurs infrarouges

```
./test_ir
```

Une fois le test lancé, le programme tente dans un premier temps d'accéder au microcontrôleur pic en initialisant la librairie libkhepera et en se connectant au robot. Si l'accès à cette librairie permettant de récupérer les fonctions et données nécessaires au lancement du test ou la connexion au robot échouent, un message d'erreur est affiché sur le terminal et le programme s'arrête.

Si aucun problème n'est à déplorer, le programme récupère les valeurs des différents capteurs infrarouges et les stocke dans un tampon. Après ça, les valeurs sont associées à chaque capteur et affichées sur la sortie standard et s'actualisent en temps réel.

Pour vérifier le bon fonctionnement des capteurs, approchez des objets des différents capteurs afin de vérifier que les valeurs des capteurs concernés par la présence de l'objet

augmentent.

Il suffit ensuite d'appuyer sur n'importe quelle touche du clavier afin de stopper le programme.

FIGURE 24 – Exemple d'utilisation du programme des capteurs infrarouges

```
-----  
TEST CAPTEURS INFRAROUGES  
-----  
  
Affichage valeurs de chaque capteur :  
  
Capteur Arriere Droit : 4  
Capteur Droit : 7  
Capteur Avant Droit : 13  
Capteur Avant : 5  
Capteur Avant Gauche : 22  
Capteur Gauche : 6  
Capteur Arriere Gauche : 4  
Capteur Arriere : 3  
  
Appuyez sur une touche pour stopper le programme.
```

3.9 Microphone

```
./test_microphone
```

Une fois le test lancé, le programme tente dans un premier temps d'accéder au microcontrôleur pic en initialisant la librairie libkhepera et en se connectant au robot. Si l'accès à cette librairie permettant de récupérer les fonctions et données nécessaires au lancement du test ou la connexion au robot échouent, un message d'erreur est affiché sur le terminal et le programme se ferme.

Si aucun problème n'est à déplorer, le programme commence tout d'abord par arrêter les haut-parleurs s'ils étaient en marche.

Suite à cela, l'initialisation du son créé par le robot se fait, et si un problème survient, un message d'erreur est affiché sur le terminal et un code d'erreur -2 est renvoyé.

Le programme coupe ensuite le son des haut-parleurs et des capteurs à ultrasons afin

d'éviter un effet Larsen, qui se produit par exemple lorsque les microphones captent le son amplifié par les haut-parleurs et le réenregistrent.

Dans un second temps, l'utilisateur est convié à entrer le volume des deux microphones séparément, dans un intervalle de 0 à 100 compris, ou par défaut s'il entre 0. Les valeurs entrées sont vérifiées et si une valeur incorrecte est rentrée, l'utilisateur est invité à rentrer de nouvelles valeurs correctes. Le volume des microphones est mis à jour et si un problème de configuration du son avec ces paramètres survient, un message d'erreur est envoyé sur le terminal et le volume est remis par défaut.

Dans un troisième temps, la durée de l'enregistrement souhaité est demandée à l'utilisateur qui peut entrer une valeur en seconde, si cette valeur est négative ou nulle, le programme l'invite à recommencer. Pour débiter l'enregistrement, une touche quelconque doit être pressée, sauf la touche "a" qui permet de redonner la durée d'enregistrement au besoin. S'ensuit l'enregistrement avec en amont l'extinction des haut-parleurs, toujours pour éviter l'effet Larsen.

Enfin, pour écouter le son qui vient d'être enregistré, le programme coupe les microphones et rallume les haut-parleurs puis demande à l'utilisateur d'entrer le volume de ces derniers. Comme les microphones, les valeurs doivent être comprises entre 0 et 100, 0 pour la valeur par défaut de 80 et la vérification des valeurs entrées est faite.

Le volume des haut-parleurs est mis à jour puis le son est joué par le robot jusqu'à sa fin et le programme sauvegarde dans un fichier audio d'extension .wav le son enregistré avec un nom de fichier qui doit être entré par l'utilisateur. Après avoir réalisé la sauvegarde, le programme se ferme et le test est terminé.

FIGURE 25 – Exemple d'utilisation du programme de test des microphones

```
TEST MICROPHONES
-----
Entrez le volume des microphones (entre 0 et 100, entrez 0 pour valeur par default) :
Microphone gauche : 60
Microphone droit : 60

Entrez une duree d'enregistrement (en sec) : 5

Enregistrement de 5s -- Entrez 'a' pour ressaisir, autre chose pour lancer l'enregistrement
Saisie : m

Debut de l'enregistrement:

Entrez le volume des haut-parleurs (entre 0 et 100, entrez 0 pour valeur par default) :
Haut-parleur gauche : 80
Haut-parleur droit : 80

Ecoute du son enregistre

Choisissez un nom de fichier (sans l'extension) pour le sauvegarder :
nouveauSon
```


3.10 Capteurs à ultrasons

`./test_us`

Une fois le test lancé, le programme tente dans un premier temps d'accéder au microcontrôleur pic en initialisant la librairie libkhepera et en se connectant au robot. Si l'accès à cette librairie permettant de récupérer les fonctions et données nécessaires au lancement du test ou la connexion au robot échouent, un message d'erreur est affiché sur le terminal et le programme se ferme.

Si aucun problème n'est à déplorer, le programme demande dans un premier temps le(s) capteur(s) à activer pour effectuer le test. Chaque capteur possède une valeur propre. L'addition de ces valeurs permet d'activer plusieurs capteurs en même temps. Dans le cas où aucune valeur ou une valeur incorrecte est saisie, tous les capteurs seront activés par défaut.

FIGURE 26 – Exemple d'utilisation du programme des capteurs à ultrasons : Choix des capteurs à activer

```
-----
TEST CAPTEURS A ULTRASONS
-----

Quel(s) capteur(s) activer ?
1  : Gauche
2  : Avant-Gauche
4  : Avant
8  : Avant-Droit
16 : Droit
31 : Tous

Ajouter les valeurs pour activer plusieurs capteurs spécifiques
Exemple : activer Gauche et Avant, 1+4, saisir 5

Si aucune valeur saisie : Tous actifs par défaut

Choix : █
```

Une fois la phase de saisie terminée, le programme présente les informations importantes concernant la compréhension des données qui seront ensuite affichées à l'écran.

Le programme affiche donc la distance en centimètre des objets se situant dans le rayon de détection des capteurs activés. Si l'objet se trouve hors de l'intervalle de détection (entre 25 et 250 cm) alors une valeur égale à 1000 sera affichée. Si le capteur est désactivé, la valeur 2000 sera affichée.

Pour quitter le programme, il suffit d'appuyer sur n'importe quelle touche du clavier.

FIGURE 27 – Exemple d'utilisation du programme des capteurs a ultrasons : Affichage des valeurs

```

-----
TEST CAPTEURS A ULTRASONS
-----

INFOS
- Distance max : 250.0 cm
- Distance min : 25.0 cm
- Distance = 2000 si capteur desactive
- Distance = 1000 si objet a moins de 25 cm ou aucune detection

- Marge d'erreur : +/- 2cm
-----

Affichage des distance (cm)

Capteur Gauche      : 124 cm
Capteur Avant-Gauche : 2000 cm
Capteur Avant       : 131 cm
Capteur Avant-Droit  : 2000 cm
Capteur Droit        : 2000 cm

Appuyez sur une touche pour stopper le programme.

```

3.11 Capteurs de chocs

```
./test_shock
```

Une fois le test lancé, le programme tente dans un premier temps d'accéder au microcontrôleur pic en initialisant la librairie libkhepera et en se connectant au robot. Si l'accès à cette librairie, permettant de récupérer les fonctions et données nécessaires au lancement du test, ou la connexion au robot échouent, un message d'erreur est affiché sur le terminal et le programme se ferme.

Si aucun problème n'est à déplorer, le programme commence tout d'abord par récupérer les valeurs des capteurs infrarouges dans un tableau. Après cela, le terminal affiche à la suite les valeurs de chaque capteur en les rafraîchissant régulièrement et vérifie qu'aucun objet n'est trop proche du robot. Si un objet est détecté comme étant trop proche du robot, le programme affiche quel capteur a détecté un potentiel choc à venir, avec la distance de l'objet par rapport au robot. Puis le programme se ferme après avoir délivré ce message.

Si aucun objet n'est détecté comme étant trop proche du robot, le programme continue à afficher les valeurs des capteurs infrarouges, en attendant que soit l'utilisateur presse

n'importe quelle touche pour le quitter soit que les capteurs à infrarouges détectent un objet se rapprochant du robot.

FIGURE 28 – Exemple d'utilisation du programme du "capteur de chocs" : pas de choc anticipé

```
-----  
TEST CAPTEUR DE CHOC  
-----  
  
Utilisation des capteurs infrarouges  
Affichage des valeurs de chaque capteur :  
  
Capteur Arriere-Gauche : 0  
Capteur Gauche : 2  
Capteur Avant-Gauche : 18  
Capteur Avant : 1  
Capteur Avant-Droit : 8  
Capteur Droit : 1  
Capteur Arriere-Droit : -3  
Capteur Arriere : -1  
  
Appuyez sur une touche pour stopper le programme.
```

FIGURE 29 – Exemple d'utilisation du programme du "capteur de chocs" : choc anticipé

```
-----  
TEST CAPTEUR DE CHOC  
-----  
  
Utilisation des capteurs infrarouges  
Affichage des valeurs de chaque capteur :  
  
Capteur Arriere-Gauche : 4  
Capteur Gauche : 500  
Capteur Avant-Gauche : 18  
Capteur Avant : 1  
Capteur Avant-Droit : 8  
Capteur Droit : 1  
Capteur Arriere-Droit : -3  
Capteur Arriere : -2  
  
Les capteurs qui ont detectes un choc a venir:  
Le capteur Gauche avec une valeur de 500 est a une distance de moins de 5 cm
```

Table des figures

1	Position des LEDs	2
2	Axes de mesure du gyroscope (vus du dessus)	3
3	Axes de mesure de l'accéléromètre (vus du dessus)	4
4	Position des capteurs infrarouges sous le robot	4
5	Position des capteurs infrarouges sur les côtés du robot	5
6	Position des microphones sur le robot	6
7	Position des capteurs à ultrasons sur le robot	7
8	Connectiques du robot	8
9	Nouvelle Connexion TeraTerm	9
10	Configuration du port série	9
11	Login robot	10
12	Exemple d'utilisation du programme de test des roues & moteurs	12
13	Exemple d'utilisation du programme des haut-parleurs : sans fichier choisi	13
14	Exemple d'utilisation du programme des haut-parleurs : avec fichier choisi	13
15	Exemple d'utilisation du programme des Leds	14
16	Exemple d'utilisation du programme de l'odometre	15
17	Exemple d'utilisation du programme du gyroscope : Choix du mode . . .	16
18	Exemple d'utilisation du programme du gyroscope : Choix "a"	16
19	Exemple d'utilisation du programme du gyroscope : Choix "b"	17
20	Exemple d'utilisation du programme de l'accéléromètre : Choix du mode .	18
21	Exemple d'utilisation du programme de l'accéléromètre : Mode "a"	19
22	Exemple d'utilisation du programme de l'accéléromètre : Mode "b"	20
23	Exemple d'utilisation du programme des capteurs infrarouges – Bottom .	21
24	Exemple d'utilisation du programme des capteurs infrarouges	22
25	Exemple d'utilisation du programme de test des microphones	23
26	Exemple d'utilisation du programme des capteurs a ultrasons : Choix des capteurs à activer	24
27	Exemple d'utilisation du programme des capteurs a ultrasons : Affichage des valeurs	25
28	Exemple d'utilisation du programme du "capteur de chocs" : pas de choc anticipé	26
29	Exemple d'utilisation du programme du "capteur de chocs" : pas choc anticipé	26