
Manuel d'utilisation

ProSE B1 2024 - CANvengers
Passerelle Android-CAN vers banc CAN réel ou simulé

Responsable du document	Paul TREMOUREUX
État du document	En révision
Version	0.1
Révision	0

Date	Actions	Auteur	Version	Révision
09/06/2023	Création du document	Paul TRÉMOUREUX	0.0	0
09/06/2023	Travail sur la partie "But"	Paul TRÉMOUREUX	0.0	1
12/06/2023	Travail sur la partie "Utilisation"	Paul TRÉMOUREUX	0.0	2
12/06/2023	Rédaction de la partie "Branchement"	Elisa DECLERCK	0.0	3
13/06/2023	Travail sur les partie "Utilisation" et "Amélioration"	Paul TRÉMOUREUX	0.0	4
13/06/2023	Rédaction du démarrage de ICSim et de CANGateway	Elisa DECLERCK	0.0	5
13/06/2023	Rédaction du lancement des tests unitaires CMocka	Elisa DECLERCK	0.0	6
13/06/2023	Rédaction de la partie "JMeter"	Thomas ROCHER	0.0	7
13/06/2023	Ajout d'un dictionnaire de domaine et d'un tableau de référence	Thomas ROCHER	0.0	8
13/06/2023	Relecture global du dossier	Thomas ROCHER	0.1	0

TABLE 2 – Table des évolutions et validations internes du document

Table des matières

1	Introduction	5
1.1	Objectif du document	5
1.2	Portée	5
1.3	Copyright	5
1.4	Références	5
2	Utilisation de CANDroid/CANgateway	7
2.1	Matériel nécessaire	7
2.2	Logiciels nécessaires	7
2.3	Préparation	7
2.3.1	Application CANDroid seule	7
2.3.2	Utilisation de la Raspberry Pi	7
2.3.3	Utilisation du Simulateur ICSim	8
2.3.4	Utilisation du Banc de Test	8
2.3.5	Utilisation du Simulateur ICSim et du Banc de Test	8
2.3.6	Tests	9
2.4	Branchement	9
2.4.1	Simulateur ICSim	9
2.4.2	Banc de Test	10
2.4.3	Banc de Test et Simulateur ICSim	10
2.5	Démarrage	10
2.5.1	Banc de Test	10
2.5.2	Simulateur ICSim	11
2.5.3	Programme CANgateway	13
2.5.4	Application CANDroid	13
2.6	Manipulation	13
2.6.1	Application CANDroid	14
2.6.1.1	Partie interaction avec le simulateur	14
2.6.1.2	Partie basse de l'écran	14
2.6.2	Programme CANgateway	14
2.6.3	Simulateur ICSim	15
2.6.4	Programme CANgateway	15
2.6.5	Simulateur ICSim	15
2.6.6	Banc de Test	15
2.7	Tests	15
2.7.1	JMeter	15
2.7.2	Robot Framework	16
2.7.3	Tests unitaires CMocka	16
3	Amélioration de CANDroid/CANgateway	17
3.1	État du projet	17
3.2	Améliorations	17
3.2.1	Spécifications	17

3.2.2	Création de scénarios	17
-------	---------------------------------	----

1 Introduction

1.1 Objectif du document

Ce document est un manuel d'utilisation définissant l'ensemble des directives à suivre par le client pour l'utilisation du projet « Passerelle Android-CAN vers banc CAN réel ou simulé » du groupe B1 2024 « CANvengers » à la suite de sa livraison.

Ce document est disponible sur le Référentiel Documentaire Projet (RDP) dans le répertoire [manuel/manuel_utilisation] sous le nom [Manuel_Utilisation_SANS_B1_2024].

1.2 Portée

Ce document est destiné :

- À l'équipe projet ;
- Au client KEREVAL.

Il pourra être consulté par les consultants de la société FORMATO sur leur demande.

1.3 Copyright

Le présent document est un document à but pédagogique. Il a été réalisé sous la direction de Paul TRÉMOUREUX dans le cadre du projet de l'Équipe ProSE B1 2024 CANvengers.

Ce document et la propriété de l'Équipe ProSE B1 2024 CANvengers. En dehors des activités pédagogiques de l'ESEO, ce document ne peut être diffusé ou recopié sans l'autorisation écrite de son propriétaire.

1.4 Références

Voici un tableau récapitulatif des documents utilisés pour le manuel d'utilisation ainsi que les liens permettant d'accéder aux fichiers.

[CdC_KEREVAL_2023]	Société KEREVAL "Cahier des charges : développement d'une Passerelle Android-CAN vers banc CAN réel ou simulé", 2023.
[ISO/IEC/IEEE 29148 : 2018]	International standard, systems and software engineering life cycle processes requirements engineering, 2018, https://standards.ieee.org/standard/29148-2018.html .
[UML_2.5]	OMG, Unified Modeling Language, version 2.5, 2015.
[Simulateur ICSim]	Société KEREVAL, Simulateur d'un tableau de bord de voiture, version 3, 2007.

[PAQL_B1_2024]	P. Trémoureux et T. Bénard, Plan d'Assurance Qualité Logicielle, version 0.3.0, 2023, Git/doc/qualite/PAQL/version/.
[plan_de_test_TEST_B1_2024]	P. Trémoureux et T. Bénard, Plan de test, version 0.5.1, 2023, Git/doc/test/plan_test/livrables/.
[dossier_de_specification_SPEC_B1_2024]	CANvengers, Dossier de specification, version 1.0, 2023, Git/doc/specification/livrables/.
[Manuel_Installation_SANS_CAN_2024]	CANvengers, Manuel d'installation, version 1.0, 2023, Git/doc/manuel/livrables/

2 Utilisation de CANdroid/CANgateway

2.1 Matériel nécessaire

L'utilisation de l'application CANdroid et du programme CANgateway nécessite un matériel spécifique. Ce matériel a été défini avec le client et a été validé via tests. CANvengers ne garantit pas le bon fonctionnement du projet « Passerelle Android-CAN vers banc CAN réel ou simulé ». Le matériel testé est le suivant :

- Banc de Test : KERAVAL Banc de Test ;
- Ordinateur : Linux Ubuntu (64-bit) ;
- Passerelle Raspberry-CAN : PEAK PCAN ;USB IPEH-002021 175459 ;
- Passerelle Tableau de Bord-CAN : RS485/CAN HAT ;
- Raspberry : pi 3B+ ;
- Smartphone : Samsung Galaxy A20e sous système Android 9.

2.2 Logiciels nécessaires

L'utilisation du système développé dans ce projet nécessite l'installation de certains logiciels :

- Application CANdroid : version 2.0.0 ;
- Programme CANgateway : version 2.0.0 ;
- Simulateur ICSim : version du 12/06/2020.

Pour installer ces logiciels veuillez suivre les instructions du Manuel Installation_SANS_B1_2024.

2.3 Préparation

En fonction de l'utilisation souhaitée, Utilisateur peut n'avoir besoin que d'une partie du matériel.

2.3.1 Application CANdroid seule

Pour une utilisation de l'application CANdroid seule, les éléments suivants sont nécessaires :

- Application CANdroid ;
- Smartphone.

2.3.2 Utilisation de la Raspberry Pi

Pour une utilisation de l'application CANdroid avec la Raspberry Pi, les éléments suivant sont nécessaires :

- Application CANdroid ;
- Programme CANgateway ;
- Raspberry ;
- Smartphone.

2.3.3 Utilisation du Simulateur ICSim

Pour une utilisation de l'application CANdroid avec la Raspberry Pi et le Simulateur ICSim, les éléments suivant sont nécessaires :

- Application CANdroid ;
- Ordinateur ;
- Passerelle Raspberry-CAN ;
- Passerelle Tableau de Bord-CAN ;
- Programme CANgateway ;
- Raspberry ;
- Simulateur ICSim ;
- Smartphone.

2.3.4 Utilisation du Banc de Test

Pour une utilisation de l'application CANdroid avec la Raspberry Pi, les éléments suivant sont nécessaires :

- Application CANdroid ;
- Banc de Test ;
- Passerelle Tableau de Bord-CAN ;
- Programme CANgateway ;
- Raspberry Pi ;
- Smartphone.

2.3.5 Utilisation du Simulateur ICSim et du Banc de Test

Pour une utilisation de l'application CANdroid avec la Raspberry Pi, les éléments suivant sont nécessaires :

- Application CANdroid ;
- Banc de Test ;
- Ordinateur ;
- Passerelle Raspberry-CAN ;
- Passerelle Tableau de Bord-CAN ;
- Programme CANgateway ;
- Raspberry Pi ;
- Simulateur ICSim ;
- Smartphone.

2.3.6 Tests

Pour l'exécution des tests, les logiciels suivants sont nécessaires :

- Tests du programme CANgateway :
 - GNU Compiler Collection : Logiciel utilisé pour compiler les tests ;
 - CMocka : Module de tests (notamment utilisé pour les tests unitaires) ;
 - Apache JMeter : Logiciel utilisé pour les tests de serveurs.
- Tests de l'application CANDroid :
 - JUnit : Logiciel utilisé pour les tests unitaires ;
 - Robot Framework : Logiciel utilisé pour les tests de validation.

2.4 Branchement

Pour l'utilisation du matériel veuillez suivre le branchement décrit ci-dessous selon votre utilisation.

2.4.1 Simulateur ICSim

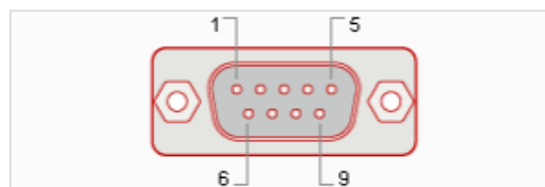
Pour utiliser le Simulateur ICSim avec notre système, il faut pouvoir connecter votre PC avec la Raspberry Pi. Pour cela, vous pouvez utiliser le PCAN-USB de PEAK. Connectez la partie USB à votre PC et la partie CAN à la Raspberry Pi.

Voici le lien du site de PCAN-USB : Interface CAN pour USB.

Il y a 3 pins importantes sur les 9 pins du connecteur CAN :

- CAN_L : pin 2 ;
- CAN_H : pin 7 ;
- GND : pin 6.

Description D-Sub



- | | |
|---|--------------------------------|
| 1 | Non connectée / En option +5 V |
| 2 | CAN-L |
| 3 | GND |
| 6 | GND |
| 7 | CAN-H |
| 9 | Non connectée / En option +5 V |

FIGURE 1 – Description des pins du PCAN-USB

Pour connecter ses pins à la Raspberry Pi, vous pouvez utiliser un connecteur DB9 femelle et souder des fils sur les pins 2, 7 et 6.

Connectez ensuite les fils CAN_H et CAN_L aux pins du RS485 CAN HAT et GND à une pin GND de la Raspberry Pi.

Attention, pour utiliser ICSim, il vous faut utiliser un RS485 CAN HAT dont la résistance de terminaison est soudée.

2.4.2 Banc de Test

Pour utiliser le Banc de Test, vous n'avez pas besoin d'un connecteur DB9 supplémentaire, celui-ci est déjà présent sur le Banc de Test.

Trouvez le connecteur DB9 dans le Banc de Test, de la même manière que pour le connecteur DB9 précédent, les pins 2 et 7 sont soudées à des fils que vous pouvez directement connecter au RS485 CAN HAT.

Attention cependant, pour utiliser le Banc de Test, il vous faut utiliser un RS485 CAN HAT dont la résistance de terminaison est désoudée.

2.4.3 Banc de Test et Simulateur ICSim

Pour utiliser à la fois le Banc de Test et le simulateur ICSim, il vous suffit de connecter le connecteur DB9 du Banc de Test au PCAN-USB que vous reliez à votre PC. Reliez également les pins 2 et 7 du Banc de Test au RS485 CAN HAT.

De la même manière que pour les deux autres utilisations, vous devez utiliser un RS485 CAN HAT dont la résistance de terminaison est désoudée.

2.5 Démarrage

Afin que le système fonctionne au mieux, nous vous proposons de suivre les étapes suivantes dans l'ordre indiqué.

2.5.1 Banc de Test

Dans le cadre de notre projet, nous tenons à préciser que nous n'avons pas utilisé le Banc de Test, mais uniquement le simulateur ICSim. Par conséquent, nous ne disposons pas des informations nécessaires pour décrire en détail le cycle d'allumage du dispositif. Notre travail s'est principalement concentré sur l'utilisation du simulateur et la mise en uvre des fonctionnalités requises.

2.5.2 Simulateur ICSim

Si vous souhaitez utiliser le Simulateur ICSim, commencez pas brancher le PCAN-USB à votre PC et à la Raspberry Pi.

Si vous le branchez sur un OS Linux, vous n'avez rien à faire, le driver permettant d'utiliser le PCAN-USB est déjà installé sur le noyau Linux.

Sinon, vous avez besoin de télécharger le driver sur le site de PEAK : [Support PEAK](#).

Si le PCAN-USB est branché et que le driver est installé, la LED rouge du PCAN-USB doit être allumée.

Commencez par activer le CAN sur votre ordinateur :

- Dans un terminal, tapez :

```
sudo ip link set can0 down
sudo ip link set can0 type can bitrate 125000
sudo ip link set can0 up
```

- Vérifiez que le CAN est bien activé :

```
sudo ip -details link show can0
```

Vous devriez voir ceci :

```
3 : can0 : <NOARP,UP,LOWER_UP,ECHO> mtu 16 qdisc pfifo_fast state UP
mode DEFAULT group default qlen 10
link/can promiscuity 0 minmtu 0 maxmtu 0
can state ERROR-ACTIVE (berr-counter tx 0 rx 95) restart-ms 0
bitrate 125000 sample-point 0.875
tq 500 prop-seg 6 phase-seg1 7 phase-seg2 2 sjw 1
pcan_usb : tseg1 1..16 tseg2 1..8 sjw 1.4 brp 1..64 brp-inc 1
clock 8000000 numtxqueues 1 numrxqueues 1 gso_max_size 65536
gso_max_segs 65535 parentbus usb parentdev 2-2.1 :1.0
```

Pour rappel, le bus CAN est fonctionnel quand il est dans l'état ERROR-ACTIVE. Dans l'état STOPPED, il n'a pas démarré. Dans l'état ERROR-PASSIVE, il est en erreur et il faut le redémarrer pour qu'il fonctionne à nouveau.

Lorsque le bus CAN est fonctionnel, la LED rouge du PCAN-USB clignote lentement (elle cligonte rapidement lorsque des trames sont émises).

Dès que le bus CAN est fonctionnel, vous pouvez lancer le Simulateur ICSim :

1. Ouvrez un terminal et placez vous dans le dossier ICSim ;
2. Tapez :

```
./icsim can0
```

3. Ouvrez un second terminal et placez vous dans le dossier ICSim ;
4. Tapez :

```
./controls can0
```

Cela est censé vous ouvrir deux fenêtres, une pour le Simulateur ICSim et une pour le contrôleur :

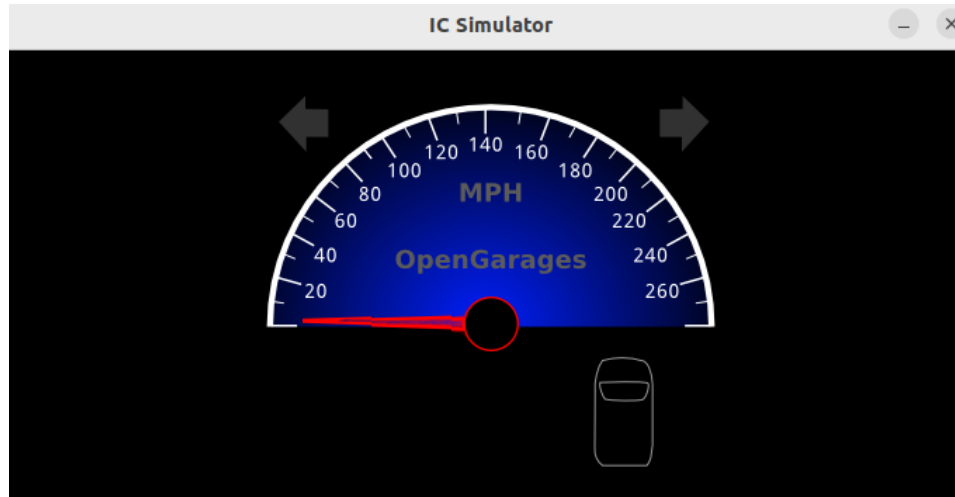


FIGURE 2 – ICSim

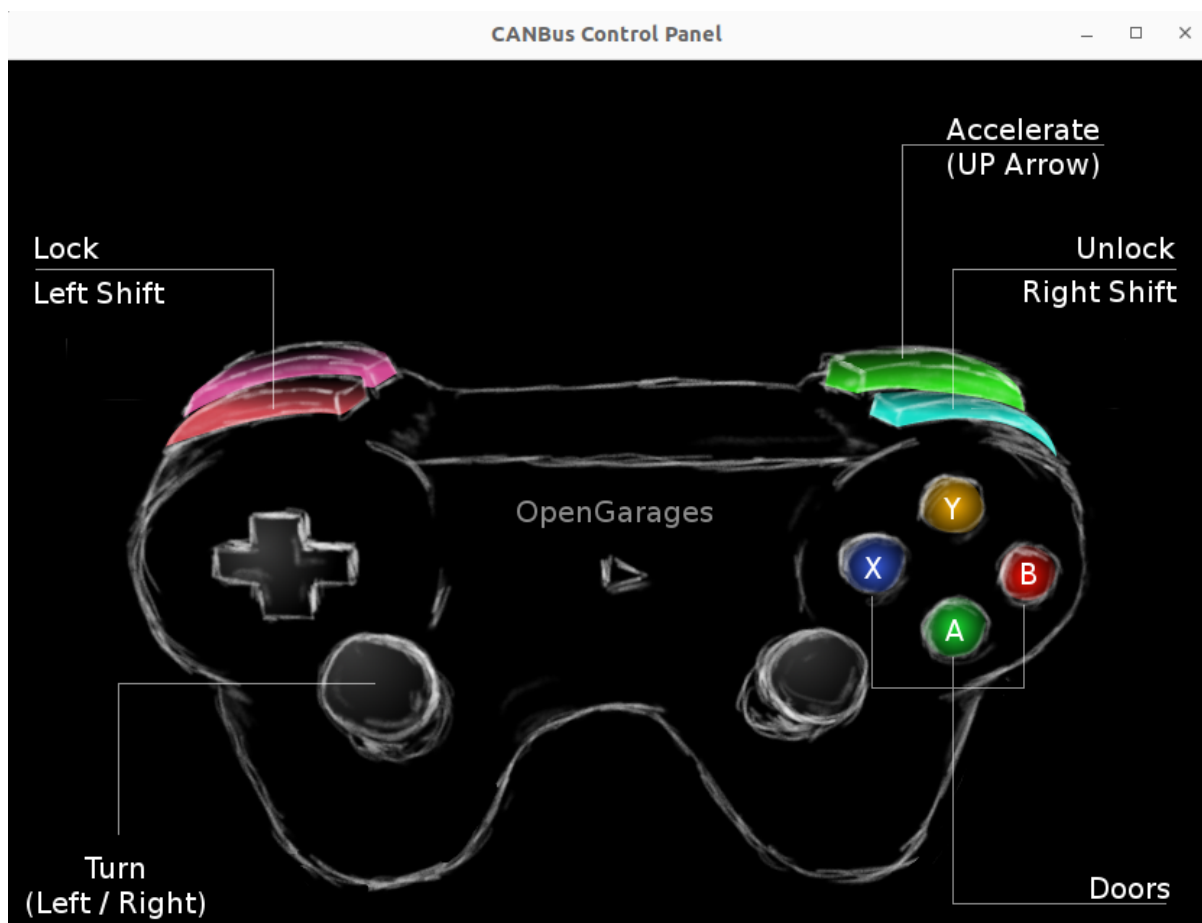


FIGURE 3 – Contrôleur

le Simulateur ICSim est démarré, vous pouvez voir toutes les trames émises entre le simulateur et le contrôleur grâce aux commandes suivantes :

- Permet d'afficher toutes les trames sniffées :

```
candump can0
```

- Permet de n'afficher que les nouvelles trames sniffées (évite d'afficher les mêmes trames à répétition) :

```
cansniffer -c can0
```

2.5.3 Programme CANgateway

Pour utiliser le programme CANgateway, commencez par vous connecter à la Raspberry Pi via SSH, ou, si vous préférez, connectez un écran et un clavier à la Raspberry Pi.

Ouvrez un terminal de la Raspberry Pi et placez vous dans le dossier /home/pi.

Vérifiez que le programme CANgateway est bien présent dans le dossier, sinon, veuillez lire le Manuel_Installation_SANS_B1_2024, dans la section "Installation du programme CANgateway".

Pour lancer le programme CANgateway, tapez :

```
sudo ./CANGateway.out
```

(Entrez le mot de passe si nécessaire)

Lorsque le programme est lancé, vous pouvez voir le message suivant s'afficher dans la console :

```
Lancement du programme
```

2.5.4 Application CANDroid

Si l'application CANDroid est installée sur le Smartphone, cliquer sur l'icône de l'application CANDroid sur l'écran.

Si l'application CANDroid n'est pas installée sur le Smartphone, suivre le Manuel_Installation_SANS_B1_2024.

2.6 Manipulation

Lors de votre utilisation du projet, seuls les fonctionnalités décrites dans dossier_de_specification_SPEC_B1_2024 et plan_de_test_TEST_B1_2024. Toute utilisation de l'application CANDroid et du programme CANgateway sortant de ce cadre peut amener à des dysfonctionnements dont l'équipe CANvengers n'est pas responsable.

2.6.1 Application CANdroid

Durant l'utilisation normale du projet, plusieurs interactions entre Utilisateur et l'application CANdroid sont prévues. Ces interactions sont les suivantes :

- Partie haute de l'écran
 - Ajout de trame
 - Ajout d'objet
 - Défilement
 - Déroulement des menu d'objet
 - Mise en marche/arrêt de l'envoi de trames
 - Reconnexion
 - Sélections d'éléments
 - Suppression d'éléments
- Partie basse de l'écran
 - Défilement
 - Enregistrement du sniffer
 - Mise en marche/arrêt du sniffer
 - Nettoyage du sniffer

L'écran de l'application CANdroid présente une partie d'interaction avec le Simulateur ICSim et une partie sniffer CAN.

2.6.1.1 Partie interaction avec le simulateur

La partie « haute » (interaction avec le sniffer) permet d'interagir avec le Simulateur ICSim. On y observe l'état de la connexion au programme CANgateway. On peut ajouter ou supprimer des objets, et dans ces mêmes objets, ajouter ou supprimer des trames. Enfin, on peut y ordonner l'envoi de trames. Pour une explication détaillée de son fonctionnement, veuillez vous référer au dossier_de_specification_SPEC_B1_2024.

2.6.1.2 Partie basse de l'écran

La partie sniffer (partie basse de l'écran) présente les trames CAN envoyées et reçues par la Raspberry Pi lorsqu'il y en a. Lorsque la Raspberry Pi n'est pas encore connectée, un commentaire indique qu'il n'y a pas de trames actuellement. Pour une explication détaillée de son fonctionnement, veuillez vous référer au dossier_de_specification_SPEC_B1_2024.

2.6.2 Programme CANgateway

Durant l'utilisation normale du projet, aucune interaction entre Utilisateur et le programme CANgateway n'est prévue.

2.6.3 Simulateur ICSim

Durant l'utilisation normale du projet, Utilisateur peut être amené à interagir avec le Simulateur ICSim. Ces interactions seront l'activation d'actionneurs et la lecture des informations affichées par le Tableau de Bord.

Pour plus d'informations concernant la manipulation du Simulateur ICSim, voir document README.md disponible sur la page github du projet.

2.6.4 Programme CANGateway

Durant l'utilisation normale du projet, il vous est possible de voir les potentiels messages d'erreurs dans la console.

Vous pouvez arrêter le programme CANGateway en appuyant sur la touche "Ctrl + C" dans la console. Cela affichera le message suivant :

```
SIGINT intercepté : demande d'arrêt du programme  
Arrêt du programme
```

Le programme CANGateway s'arrête alors.

Si le programme ne s'arrête pas, il se peut que le client TCP soit encore connecté, veuillez à bien fermer l'application CANdroid avant de fermer le programme CANGateway afin de forcer la déconnexion du client TCP.

2.6.5 Simulateur ICSim

Durant l'utilisation normale du projet, Utilisateur peut être amené à interagir avec le Simulateur ICSim. Ces interactions seront l'activation d'actionneurs et la lecture des informations affichées par le Tableau de Bord.

Pour plus d'informations concernant la manipulation du Simulateur ICSim, voir document README.md disponible sur la page github du projet.

2.6.6 Banc de Test

Durant l'utilisation normale du projet, Utilisateur peut être amené à interagir avec le Banc de Test. Ces interactions seront l'activation d'actionneurs et la lecture des informations affichées sur le Tableau de Bord.

2.7 Tests

Lors de l'exécution des codes de test, certaines étapes sont à respecter.

2.7.1 JMeter

Pour exécuter les tests du serveur TCP du Postman dans JMeter, il faut tout d'abord lancer le programme CANGateway. Pour lancer le programme CANGateway, tapez :

```
sudo ./CANGateway.out
```

(Entrez le mot de passe si nécessaire)

Il faut ensuite lancer JMeter. Pour ce faire, accédez au dépôt C du projet, et allez dans le répertoire `Git/c/production/jmeter_test/`. Si ce n'est pas déjà fait, veuillez modifier le Makefile et modifier le chemin absolu vers JMeter, de façon à ce qu'il corresponde à votre installation.

Ensuite pour compiler et lancer JMeter, exécutez les commandes suivantes :

```
make all
```

```
./customLaunch.sh
```

Enfin, une fois JMeter lancé, appuyez sur la flèche verte pour démarrez le test.

2.7.2 Robot Framework

2.7.3 Tests unitaires CMocka

Vous pouvez effectuer des tests unitaires d'encodage et de décodage de chaînes de caractères en utilisant le programme CANGateway.

Ces tests sont exécutables sur cible Raspberry Pi mais également sur PC.

Selon votre choix, la compilation du programme de tests de CANGateway sera différente, veuillez vous référer au [Guide d'installation] pour la compilation.

Quelque soit la cible, les tests sont exécutables de la même manière.

Ouvrez un terminal de la Raspberry Pi ou du PC et placez vous dans le dossier contenant l'exécutable de tests, tapez :

```
./CANGateway_test.out
```

Trois tests sont alors exécutés, vous pouvez voir leur résultats dans la console.

3 Amélioration de CANdroid/CANgateway

3.1 État du projet

À la livraison, le projet est incomplet. Les éléments suivant manquent ou sont incomplet :

—

3.2 Améliorations

Plusieurs axes d'amélioration sont possibles.

3.2.1 Spécifications

Comme détaillé dans la section 3.1 le projet est incomplet à la livraison.

3.2.2 Création de scénarios

Le projet Passerelle Android-CAN vers banc CAN réel ou simulé ne permet pas de créer des séquences d'envois précises. Or, certains systèmes nécessitent la reception d'une séquence précise de trames (par exemple lors des cylce d'allumage). Une amélioration intéressante serait de donner la possibilité à Utilisateur de créer des scénario d'envois. Ce dernier pourrait décider de l'ordre d'envoi et du moment exact ou chaque trame doit être envoyée.