**Rapport de Projet Temps Réel**

**Vincent BAURES** : conception, codage des fonctions

**Alicia CALMET** : conception, rédaction du compte-rendu

**Robin MOINE** : conception, codage des fonctions

**Loïc ROBERT** : secrétaire, testeur, implémentation/intégration des fonctions sur le robot, assistant général

Table des matières

[1 Conception 3](#_Toc67303130)

[1.1 Diagramme fonctionnel général 3](#_Toc67303131)

[1.2 Groupe de threads gestion du moniteur 3](#_Toc67303132)

[1.2.1 Diagramme fonctionnel du groupe gestion du moniteur 3](#_Toc67303133)

[1.2.2 Diagrammes d’activité du groupe gestion du moniteur 3](#_Toc67303134)

[1.3 Groupe de threads gestion du robot 3](#_Toc67303135)

[1.3.1 Diagramme fonctionnel du groupe gestion 3](#_Toc67303136)

[1.3.2 Diagrammes d’activité du groupe 3](#_Toc67303137)

[1.4 Groupe de threads vision 3](#_Toc67303138)

[1.4.1 Diagramme fonctionnel du groupe vision 3](#_Toc67303139)

[2 Transformation AADL vers Xenomai 3](#_Toc67303140)

[2.1 Thread 3](#_Toc67303141)

[2.1.1 Instanciation et démarrage 3](#_Toc67303142)

[2.1.2 Code à exécuter 3](#_Toc67303143)

[2.1.3 Niveau de priorités 3](#_Toc67303144)

[2.1.4 Activation périodique 3](#_Toc67303145)

[2.2 Donnée partagée 3](#_Toc67303146)

[2.2.1 Instanciation 3](#_Toc67303147)

[2.2.2 Accès en lecture et écriture 3](#_Toc67303148)

[2.3 Port d’événement 3](#_Toc67303149)

[2.3.1 Instanciation 3](#_Toc67303150)

[2.3.2 Envoi d’un événement 2.3.3 Réception d’un événement 3](#_Toc67303151)

[2.4 Ports d’événement-données 3](#_Toc67303152)

[2.4.1 Instanciation 3](#_Toc67303153)

[2.4.2 Envoi d’une donnée 3](#_Toc67303154)

[2.4.3 Réception d’une donnée 3](#_Toc67303155)

[3 Analyse et validation de la conception 3](#_Toc67303156)

# 1 Conception

## Diagramme fonctionnel général

## Groupe de threads gestion du moniteur

### Diagramme fonctionnel du groupe gestion du moniteur

### Diagrammes d’activité du groupe gestion du moniteur

|  |  |  |
| --- | --- | --- |
| **Nom du Thread** | **Rôle** | **Périodicité** |
| th\_receiveFromMon | Permet de recevoir des messages depuis le moniteur | Non |
| th\_SendTo Mon | Permet d’envoyer un message au moniteur | Contrainte de temps de 10 ms |
| th\_Server | Permet la connexion au serveur | Non |

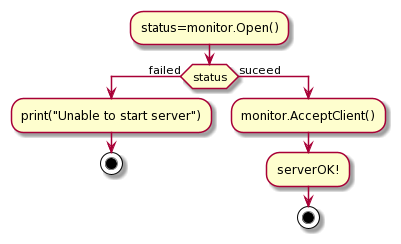
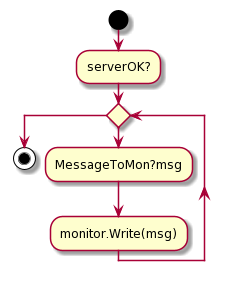


Figure : Diagramme d'activité du thread th\_server



## 

Figure : Diagramme d’activité du thread th\_sendToMon

## Il est a noté que nos diagrammes d’activité ne sont pas exhaustifs. Afin de ne pas surcharger les digrammes, nous n’avons donc pas représenter les nœuds d’exécution des sémaphores et des mutex.

Figure 3:Diagramme d’activité du thread th\_receiveFromMon

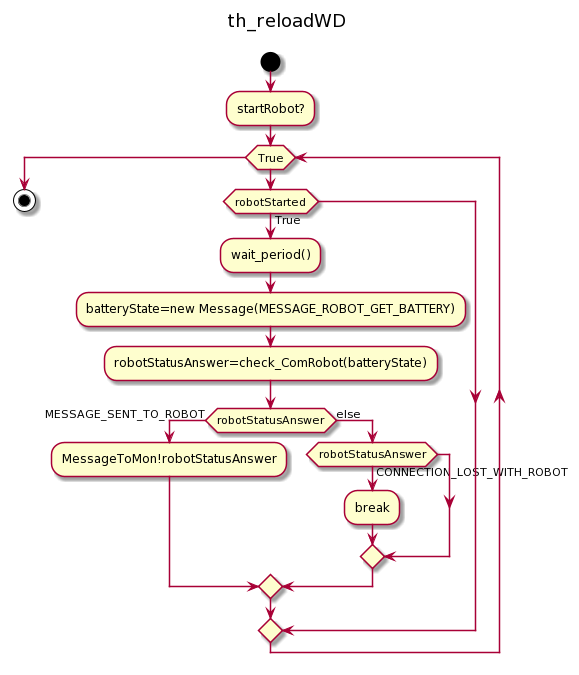
## 1.3 Groupe de threads gestion du robot

### 1.3.1 Diagramme fonctionnel du groupe gestion

### 1.3.2 Diagrammes d’activité du groupe

|  |  |  |
| --- | --- | --- |
| **Nom du Thread** | **Rôle** | **Périodicité** |
| Th\_battery | Correspond fonctionnalité 13. Permet de retourner le niveau de batterie du robot toutes les 500 ms au moniteur | 500ms |
| Th\_move | Correspond à la fonctionnalité 12. Permet le déplacement du robot : un message de déplacement est envoyé du moniteur au superviseur puis du superviseur au robot | Contrainte de temps de 100 ms |
| Th\_startRobot | Ce thread est toujours en attente d’une demande de mode de démarrage pour le robot. Dès qu’il reçoit une requête de démarrage sans watchdog il permet d’allumer le robot, s’il reçoit une requête avec watchdog, il démarre en mode watchdog. | Non |
| Th\_OpenComRobot | Permet d’ouvrir la communication entre le robot et le superviseur | Non |
| Th\_ReloadWD | Permet de confirmer la présence du watchdog au superviseur | Contrainte de temps de |

Figure : Digramme activité du thread th\_battery



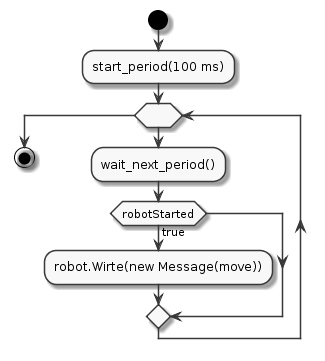


Figure : Diagramme d'activité du thread th\_move

## 

## 

Figure : Diagramme d'activité du thread th\_startRobot

## 

Figure : Diagramme d'activité du thread th\_reloadWD

Figure : Diagramme d'activité du thread th\_comRobot

## 1.4 Groupe de threads vision

1.4.1 Diagramme fonctionnel du groupe vision   
1.4.2 Diagrammes d’activité du groupe vision

|  |  |  |
| --- | --- | --- |
| Nom du Thread | Rôle | Périodicité |
| Th\_Camera |  |  |
|  |  |  |

# 2 Transformation AADL vers Xenomai

## 2.1 Thread

### 2.1.1 Instanciation et démarrage

### 2.1.2 Code à exécuter

### 2.1.3 Niveau de priorités

### 2.1.4 Activation périodique

## 2.2 Donnée partagée

### 2.2.1 Instanciation

### 2.2.2 Accès en lecture et écriture

## 2.3 Port d’événement

### 2.3.1 Instanciation

### 2.3.2 Envoi d’un événement 2.3.3 Réception d’un événement

## 2.4 Ports d’événement-données

### 2.4.1 Instanciation

### 2.4.2 Envoi d’une donnée

### 2.4.3 Réception d’une donnée

# 3 Analyse et validation de la conception

Afin démontrer que les fonctionnalités que nous avons implémentées s’exécutent correctement, nous avons réalisé une vidéo récapitulative. Tous les membres du groupe se sont exprimés dans cette vidéo.