|  |
| --- |
|  |
| Projet Robot Ambiant |
|  |

|  |
| --- |
| Zheng, Raphaël Bresson, Charles Roger  19/02/2016 |

Table des matières

**Introduction1**

1. **Cahier des charges et analyse fonctionnelle :4**

Tapez le titre du chapitre (niveau 2)5

Tapez le titre du chapitre (niveau 3)6

1. **Designs possibles :4**
2. Design ROS + microcontrôleur5
3. Design FPGA5
4. Comparaison des designs 5
5. **Cahier de recettes :4**

**Conclusion1**

1. Cahier des charges et analyse fonctionnelle
2. Designs possibles
3. Design de l’application JAVA

Classes de gestion de capteurs reliés directement à la carte HIGH LEVEL (caméra, lidar)

Ordres

Centre de décision

Classes de gestion de communication vers la carte LOW LEVEL

(ETHERNET ou UART)

Données brutes des capteurs internes

Données brutes des capteurs externes

Classes de génération de la carte des obstacles

(OPENCV)

Carte des obstacles

Code IA

1. Design ROS + microcontrôleur

Wifi

HIGH LEVEL (Carte Firefly avec ROS)

Rôle :

* Décision en fonction des acquisitions de la caméra, du lidar et des capteurs du LOW LEVEL
* Envoi des ordres de haut niveau au LOW LEVEL

LOW LEVEL (MBED ARCH PRO)

Rôle :

* Transformation des ordres de haut niveau en un ordre de bas niveau compréhensible pour les actionneurs.
* Envoi des données des capteurs au HIGH LEVEL

Valeurs brutes des capteurs

Ethernet

Ordres de haut niveau

Données brutes

Images

Valeurs brutes

Ordres de bas niveau

Lidar

Actionneurs

Capteurs

Caméra

Figure 1 : Design haut niveau design ROS + microcontrôleur

Caméra

Lidar

IMU

Proximité

I2C

USB

Ethernet

MBED ARCH PRO

Multiplexeur

Firefly

Ethernet

Appli IA

DirG

PwmG

7 pins GPIO

DirD

PwmD

Alimentation Firefly et Mbed

H-Bridge

Bras articulé

Alimentation actionneurs

Moteurs

roues

Bras articulé

Lidar

*Actionneur Capteur Ordre Données des capteurs*

Figure 2 : Design détaillé pour ROS + microcontrôleur

Données des capteurs

Ordres High Level

Communication

Décision

Données des capteurs

Ordres Low Level

Thread 1 : Aller à gauche

Thread 2 : Accélérer

Thread 3 : Bouger le bras de n degrés

…

…

Figure 3 : design du programme C pour le LOW LEVEL

1. Design FPGA

HIGH LEVEL (Processeur JAVA embarqué)

Rôle :

* Décision en fonction des acquisitions de la caméra, du lidar et des capteurs du LOW LEVEL
* Envoi des ordres de haut niveau au LOW LEVEL

LOW LEVEL (Circuits dédiés)

Rôle :

* Transformation des ordres de haut niveau en un ordre de bas niveau compréhensible pour les actionneurs.
* Envoi des données des capteurs au HIGH LEVEL

Valeurs brutes des capteurs

Axi Bus

Ordres de haut niveau

MOJO V3 (SPARTAN 6)

Ordres de bas niveau

Valeurs brutes

Images

Données brutes

Lidar

Caméra

Capteurs

Actionneurs

Figure 4 : design design haut niveau pour FPGA

Lidar

Proximité

IMU

Caméra

Shield

I2C

JAVA EMBEDDED PROCESSOR

AXI

CIRCUITS DEDIES

Multiplexeur

AXI

Appli IA

DirG et PwmG

2 std\_logic\_vector

(15 downto 0)

DirD et PwmD

Std\_logic\_vector

(15 downto 0)

FPGA

Alimentation FPGA

Servo Shield

Bras articulé

H-Bridge

Alimentation actionneurs

Moteurs

roues

Bras articulé

Lidar

*Actionneur Capteur Ordre Données des capteurs*

Figure 5 : Design détaillé pour FPGA

Données des capteurs

Ordres High Level

Communication

+

Buffers

Décision (gestion de la table de vérité)

Données des capteurs

Ordres Low Level

Circuit 1 : Aller à gauche

Circuit 2 : Accélérer

Circuit 3 : Bouger le bras de n degrés

…

Figure 6 : Design du bloc des circuits dédiés …

1. Comparaison des design FPGA/ROS + microcontrôleur

|  |  |  |
| --- | --- | --- |
| Critère de comparaison | FPGA | ROS + µContrôleur |
| Performances | ++ | -- |
| Temps de développement | ++ | -- |
| Complexité du développement | ++ | -- |
| Programmation dynamique | OUI | NON |
| Disponibilité immédiate du matériel | NON | OUI |

Tableau 1 : Comparaison des designs FPGA / ROS + µContrôleur

1. Cahier de recettes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Fiche de test unitaires | | | | |
| Numéro de test | Composants | Description | Scénario | Résultat |
| 1 | Lidar + HIGH LEVEL | Test de l'acquisition du Lidar | Mettre un objet à une distance connue du Lidar et vérifier à l'aide du Debug de Microvision si la valeur est cohérente |  |
| 2 | Caméra + HIGH LEVEL | Test de l'acquisition de la Caméra | Afficher la sortie de la caméra à l'écran |  |
| 3 | Lidar + Caméra + HIGH LEVEL | Génération de la carte des obstacles | Générer la carte des obstacles à l'aide des valeurs des capteurs (caméra + lidar) |  |
| 4 | LOW LEVEL + HIGH LEVEL | Communication ETHERNET ou UART | Envoyer une trame de données et vérifier la cohérence de ce qu'on reçoit via affichage( LOW LEVEL vers HIGH LEVEL) ou via debug de Microvision (HIGH LEVEL vers LOW LEVEL) |  |
| 5 | LOW LEVEL + Moteurs | Commande des moteurs | Vérifier la bonne mise en marche et extinction des moteurs en parallèle |  |
| 6 | LOW LEVEL + Bras articulé | Commande du bras articulé | Vérifier la bon fonctionnement en parallèle des servo-moteurs du bras articulé |  |
| 7 | LOW LEVEL + HIGH LEVEL + Moteurs + Bras Articulé | Commande des moteurs et du bras à haut niveau | Vérifier que les moteurs et le bras sont bien commandés |  |
| 8 | LOW LEVEL + HIGH LEVEL + Moteurs + Bras Articulé | Test de saisie d'objet | Saisir un objet donné |  |
| 9 | LOW LEVEL + HIGH LEVEL + Moteurs + Bras Articulé | Recherche de chemin | Se déplacer vers un point sans heurter d'obstacle |  |
| Tableau 2 : Fiche de tests unitaires | | | | |
|  |  |  |  |  |
| Interactions avec l'environnement | | | | |
| Numéro | Capteurs utilisés | élements externes utilisés | Scénario | Résultat |
| 1 | Caméra + Lidar | QRCodes ou Aruco | Le robot doit saisir l'objet ayant un code |  |
| 2 | Caméra | Aruco ou QRCode | Le robot doit suivre l'utilisateur qui porte un code |  |
| 3 | Tous les capteurs | Aucun | Le robot doit trouver le plus court chemin pour arriver à une destination décidée à l'avance |  |
| 4 | Caméra | éventuellement codes Aruco | Le robot doit sortir d'une pièce tapisée de codes aruco tout seul |  |

Tableau 3 : Fiche de tests d’interaction avec l’environnement