

**Nano Ordinateurs  
Gestion de projet   
Robot Baliseur**

Mars 2018

Version 1.0

Wattin Jérôme  
Sam Mahaux

Fabio Cumbo

Timothée Simon

# Informations générales sur le document

### Contact

Pour toute question ou remarque concernant ce document, merci de contacter :

Wattin Jérôme - Etudiant

Téléphone : +32479886237

Mail : jerome.wattin@std.heh.be

HEH Campus Technique

8A Avenue Maistriau,

7000 Mons

### Confidentialité

Ce document contient des informations confidentielles et exclusives de la Haute École en Hainaut (HEH). Le service informatique ne peut divulguer les informations confidentielles contenues dans ce document à un tiers sans le consentement écrit de la HEH, hormis aux employés, enseignants ou directeurs qui ont besoin de connaître son contenu à des fins d'évaluation du document. Le service informatique se doit d'informer ces personnes de la nature confidentielle de ce document et d'obtenir leur accord pour préserver sa confidentialité.

### Termes et conditions

La HEH n'assume aucune responsabilité pour les erreurs ou omissions dans le contenu de ce document ou de tout document de tiers référencé ou associé, y compris, mais sans s'y limiter, les erreurs typographiques, les inexactitudes ou les informations périmées. Ce document et tous les renseignements qui s'y trouvent sont fournis «tels quels» sans aucune garantie, expresse ou implicite.

### Informations sur le document

|  |  |
| --- | --- |
| Nom du document | RobotBaliseurQ2.Docx |
| Version | Version 1.00 |
| Niveau de confidentialité | Utilisation interne uniquement |
| Auteur du document | Wattin Jérôme |
| Contributeur(s) |  |
| Révisé par |  |
| Approuvé par |  |

### Versions

|  |  |  |  |
| --- | --- | --- | --- |
| version | Date de parution | Modification réalisée par | Modification(s) apportée(s) |
| 1.00 | 26/03/2018 | Wattin.J | Création du document |
|  |  |  |  |
|  |  |  |  |

### Documents connexes et/ou de référence

|  |  |  |  |
| --- | --- | --- | --- |
| Nom du document | Description | Date | version |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

### Table des matières

[1. Informations générales sur le document 2](#_Toc513119002)

[Contact 2](#_Toc513119003)

[Confidentialité 2](#_Toc513119004)

[Termes et conditions 2](#_Toc513119005)

[Informations sur le document 3](#_Toc513119006)

[Versions 3](#_Toc513119007)

[Documents connexes et/ou de référence 3](#_Toc513119008)

[Table des matières 4](#_Toc513119009)

[2. Présentation générale du projet 5](#_Toc513119010)

[3. Présentation détaillée du capteur à ultrason 6](#_Toc513119011)

[4. Choix de l’IDE 7](#_Toc513119012)

[5. Algorithme de programmation 7](#_Toc513119013)

[6. Problèmes rencontrés 7](#_Toc513119014)

[7. Améliorations 7](#_Toc513119015)

[8. Conclusions 7](#_Toc513119016)

# Introduction et présentation générale du projet

L’objectif principal de ce projet a été de programmer une voiture afin de lui permettre de réaliser une cartographie autonome de son environnement.

Nous avons été poussé à choisir le robot baliseur pour le côté original du projet et parce que nous avons estimé le sujet intéressant ainsi que le travail de recherche derrière qui pourraient nous apporter de nouvelles notions.

# Matériel utilisé

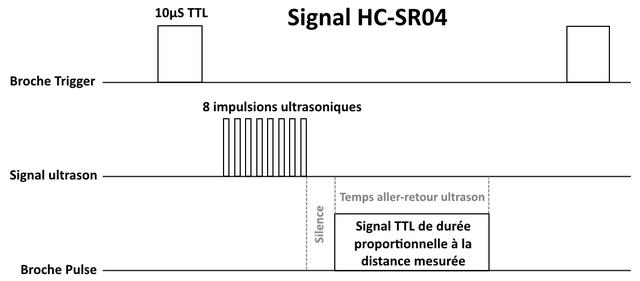
Pour réaliser cela, nous avons utilisé une base d’un robot de marque Initio composée d’un châssis, 4 roues et 2 moteurs pas à pas.

Nous avons également utilisé 2 capteurs à ultrason pour détecter les obstacles ainsi qu’un Raspberry pi 3B pour programmer le robot.

Afin de sécuriser au mieux son déplacement, nous avons fixé un capteur à ultrason à l’avant du robot et l’autre sur le côté gauche.

# Présentation détaillée du capteur à ultrason

Le capteur à ultrason fonctionne sur base de la vitesse du son.



La prise d’une mesure se déroule comme suit :

1. On envoie une impulsion HIGH durant 10 µs sur la broche Trigger (pin nommé « Trig ») du capteur
2. Le capteur répond à l’impulsion, par l’envoie d’une série de 8 impulsions ultrasonique (40KHz), inaudible pour l’être humain.
3. A l’impact d’un objet, l’ultrason retourne en sens inverse. Cela signifie que lors d’un impact face à la source, l’onde est renvoyée en direction du capteur
4. Lors de la réception de l’écho, le capteur clôture.

Afin de déterminer la durée de l’aller-retour de l’onde, la broche ECHO du capteur (pin nommé « echo ») reste sur HIGH durant les étapes 3 et 4.

# Contrôle du robot

Pour tirer parti au mieux du Raspberry pi 3 et pour le côté innovation du projet, nous avons décidé de développer un site web permettant principalement d’avoir une interface pour démarrer le robot pour cartographier une pièce et l’afficher en retour.

Nous avons donc dû installer un serveur Apache 2 sur le Raspberry pi 3 permettant de répondre aux requêtes envoyées, et nous avons également installé MySQL et PHP sur le Raspberry pi 3

Un minimum de sécurité sur le site web a été implémenté afin d’éviter que n’importe qui puisse démarrer une cartographie. Une interface de connexion a donc été mise en œuvre. Pour stocker les comptes utilisateurs et se connecter, nous avons ajouté une table appropriée à notre base de données.

Afin d’apporter un intérêt supplémentaire au site web, nous avons également créée une autre table pour stocker l’utilisateur, la pièce indiquée et sa cartographie.

Chaque utilisateur peut donc visualiser son historique de cartographie effectué.

# Choix de l’IDE

Notre choix d’IDE s’est porté sur Geany.  
En effet, celui-ci étant léger et équipé d’un interpréteur intégré, il inclut les fonctionnalités élémentaires pour la réalisation de notre projet. De plus il est multi-plateforme et supporte les langages que nous avions dû utiliser, c’est-à-dire Python, HTML, CSS et PHP.

Il répondait donc à nos exigences.

# Algorithme de programmation

## Programmation du robot

## Développement du site web

# Problèmes rencontrés

Le premier problème auquel nous avons été confrontés a été le branchement et la fixation des capteurs à ultrason.  
Nous avons dû réutiliser les fixations du toit afin de pouvoir poser les capteurs latéraux.

# Améliorations

# Conclusions