

Documentation Centrale Inertielle

I. Centrale Intertielle

La Centrale inertielle est un pôle important de notre projet, en effet, cette dernière nous permet d'avoir un autre référentiel. Ce dernier nous permet d'obtenir de nouvelles informations de positionnement, autres que celles obtenues avec les tags RFID. Grâce à cette centrale, nous obtenons de nouveaux types de mesures tels que des mesures d'accélération (avec l'accéléromètre), de température, d'inclinaison et bien d'autres.

II. Fichier Header

Pour l'utilisation de la centrale inertielle, nous avons adopté une stratégie basée sur le découpage en tâches de notre système. En effet, nous avons utilisé un OS en temps réel (FreeRTOS) pour gérer la communication au sein de système (lecteur -> centrale et réciproquement). Ici, nous avons choisi d'utiliser un mode de communication par I2C, avec le modèle maître-esclave. On peut aussi utiliser une communication via SPI, mais celle-ci n'est pas optimale au vu des ports présents sur notre lecteur. Pour se faire, nous avons un fichier header dans notre code qui nous permet d'initialiser toutes les communications en I2C. Dans ce fichier header, nous relions et utilisons un grand nombre de registres de la librairie MPU. En effet, il existe un découpage particulier des registres, et l'on doit suivre à la lettre l'ordre des registres. Les 3 premiers sont réservés par le Gyroscope, les registres 13 14 et 15 sont réservés par l'accéléromètre. Il existe aussi pour le Gyroscope des registres d'offset, qui sont ceux de 19 à 24. Nous n'utilisons pas moins de 126 registres, qui nous permettent de recueillir toutes les informations nécessaires sur cette centrale. S'en suit alors tout un tas de définitions de variables pour écrire dans les registres. Il existe aussi une phase de calibration de la Centrale Inertielle, il nous faut donc alors définir des variables de sensibilité, qui vont définir notre précision pour chaque calibration. Nous avons le même principe de sensibilité pour la mesure des angles. Enfin, la dernière définition ici repose sur la sensibilité du magnétomètre.

A ce stade, tout a été défini et est prêt à être utilisé. L'utilisation de la librairie et le remplissage de ce fichier Header a été en grande partie réalisé grâce à l'aide de M.Latorre.

III. Fichier Source

Ici repose la plus grande partie du code de notre centrale inertielle. En effet, c'est là que se trouve l'ensemble du code de fonctionnement du module.

Nous utilisons une seule fonction : **BSP_MPU9520_Init()**. Cette dernière nous permet d'initialiser et de relier tous les registres entre eux, en fonction du milieu environnant dans lequel la mesure est effectuée, la marge d'erreur sur les valeurs mesurées etc.

Dans ce fichier, on gère l'ensemble des interruptions, toutes les initialisations (magnétomètre, I2C maître et I2C esclave). Ensuite, nous gérons la lecture des données reçues via le magnétomètre. Dans la première partie du code, il y a toute une étape d'initialisation et de récupération d'états de capteurs. Ensuite, nous gérons la partie filtrage, avec un filtre de type Passe-Bas. Les interruptions sont ici pour régler la bande passante du filtre passe-bas numérique (DLPF). La partie magnétomètre qui comprend les capteurs magnétiques servant à détecter le magnétisme dans les axes X,Y et Z, D'autre

part, nous avons configuré la sortie de la centrale inertielle pour être compatible avec le module de détection qui se base sur un circuit d'amplification de signal et une autre partie arithmétique pour traiter les signaux de chaque capteur. Enfin, les données mesurées du magnétomètre sont récupérées en continu, pour cela il suffit de programmer la partie protocole I2C pour transférer les données traitées.

