

RAPPORT FINAL

PSI SPARROW



« Conception d'une centrale inertielle à bas coût »



SOMMAIRE

RÉSUMÉ	3
ABSTRACT	4
ETAT DE L'ART	6
Présentation	6
les Technologies	7
<i>Les gyromètres*</i>	7
<i>Les accéléromètres*</i>	8
Les dérives inertielles	10
NOS EQUIPEMENTS	11
La centrale inertielle	11
Le microcontrôleur	13
Les logiciels	16
ETUDE THEORIQUE	17
Theorie D'euler	17
Application	19
PROTOCOLES DE COMMUNICATION	24
I2C*: INTER Integrated Circuit	24
UART*: Universal Asynchronous Receiver Transmitter	27
ORGANISATION DU PROJET	30
Diagramme de gantt	30
Méta Projet Management	32
CONCLUSION	33
RESSOURCES	34
GLOSSAIRE	41
ANNEXE	42

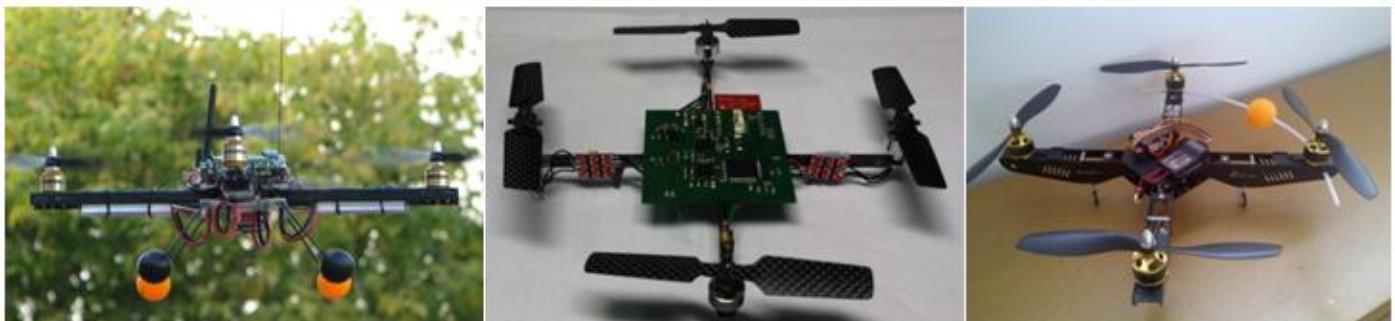
RÉSUMÉ

DE PLUS EN PLUS PETIT, DE MOINS EN MOINS CHER ...

Le pôle Acquisition et Traitement des Images et du signal (ATIS) de l'Ecole Supérieure d'Informatique Electronique Automatique (ESIEA) est une structure de Recherche et Développement travaillant sur des projets innovants.

Au sein de ce laboratoire, nous nous occupons de l'optimisation de la navigation du micro drone SPARROW. Celui-ci s'inscrit dans la continuité des projets aérodynamiques d'ATIS : Faucon Noir et Moustik.

Nous collaborons avec le groupe PAIR qui s'occupe de la conception mécanique et électronique du micro drone. Sa fonction principale sera de voler dans un milieu clos. Mais il sera aussi utilisé pour l'acquisition et la transmission de données en temps réel sur le terrain.



Faucon noir (ATIS), Moustik (ATIS) et Quad Flyer 330 X-S (Gau)

Dans une optique économique, les chercheurs du pôle ATIS nous ont proposé de réduire la dépense majeure concernant l'équipement de navigation qui s'élève à plus de \$ 2000. Ainsi, ils ont trouvé une alternative pour l'acquisition des données spatiales : une centrale inertuelle* à \$ 300.

Dans ce contexte, nous sommes chargés de l'implémentation de cette nouvelle carte, l'optimisation du traitement des données et la comparaison des résultats avec ceux de l'ancienne carte. Nous avons surtout découvert deux protocoles de communication UART* et I2C*. Ils nous ont servis à travailler avec la centrale inertuelle* et le microcontrôleur* et d'avoir un visuel sur l'ordinateur via un émulateur de terminal.

MOTS CLES

Drone, Navigation, Centrale inertuelle*, Système embarqué, Gyroscope, Accéléromètre*, Angle d'Euler...

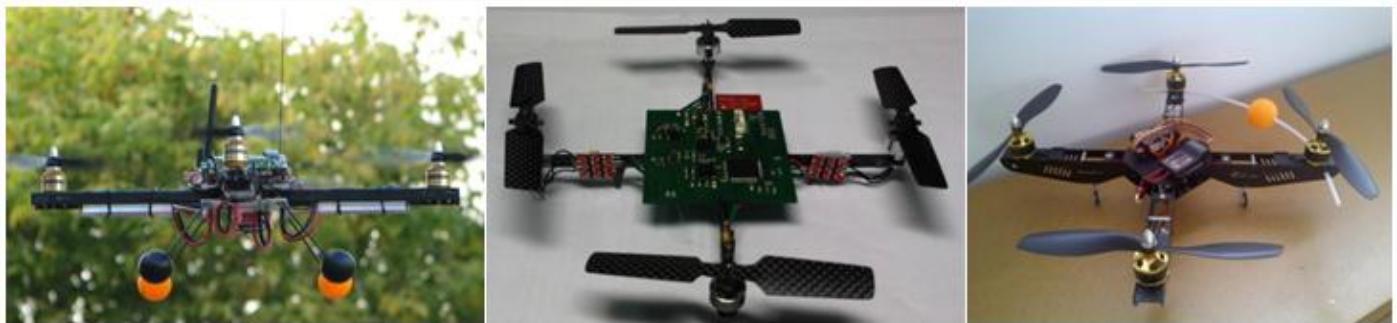
ABSTRACT

SMALLER AND CHEAPER ...

The department of research “Acquisition et Traitement des Images et du Signal” (ATIS) based at the French engineering school ESIEA “Ecole supérieure d’ informatique électronique automatique” is a laboratory that works on innovative projects in the field of image, signal recording and processing.

Inside this laboratory, we are in charge of the navigation optimization concerning the micro drone SPARROW. This project is another step for ATIS aerodynamics after: Faucon noir and Moustik.

We collaborate with a PAIR group who takes care of the mechanical and electronic design. The drone main function will be to fly indoors, in a confined environment but it will still be used to collect and transmit real time data from the field.



Faucon noir (ATIS), Moustik (ATIS) et Quad Flyer 330 X-S (Gau)

For economic reasons the ATIS lab researcher suggested that we to reduce the main expense concerning the navigation equipment. To achieve our goal, they gave us an alternative for the spatial data acquisition. In fact the \$ 2000 inertial central must be replaced by a \$ 300 one.

In this context, we are in charge of the implementation of this new card, the optimization of the data processing and the comparison of the results coming from the new card with those coming from the old one. Mostly, we have studied two communication protocols UART* and I2C*. Therefore, we have been able to interact with the inertial measurement unit and the microcontroller thanks to a terminal emulator.

KEY WORDS

UAV (unmanned aerial vehicle), Navigation, IMU* (inertial measurement unit), Embedded System, Gyroscopes, Accelerometer, Euler's Angle

INTRODUCTION

Dans le cadre de notre PSI proposé par le Laboratoire ATIS de l'ESIEA, nous travaillons sur l'optimisation de la navigation du micro drone SPARROW. Celui-ci s'inscrit dans la continuité des projets aérodynamiques d'ATIS: Faucon noir et Moustik.

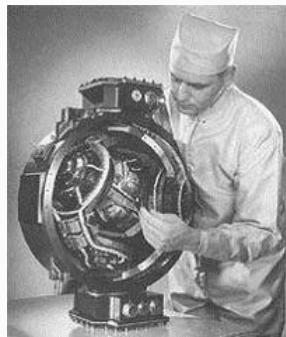
Nous travaillons en parallèle avec le groupe PAIR qui s'occupe de la conception mécanique et électronique du micro drone. Sa fonction sera de voler dans un milieu clos ou confiné. Il sera aussi utilisé l'acquisition et transmission de données en temps réel sur le terrain.

Dans une optique de logique industrielle, les chercheurs du pôle ATIS nous ont proposé de réduire dépense majeure concernant l'équipement de navigation qui s'élève à \$ 2000. Ainsi, ils ont trouvé une alternative pour l'acquisition des données spatiales avec une centrale inertielle* à \$ 300.

Dans un premier temps, nous chercherons à comprendre le fonctionnement de cette nouvelle carte électronique et de ses composants. Nous poursuivrons par l'implémentation d'un programme qui traite ces données et récupère les informations de position souhaitées. Afin de vérifier l'exactitude de nos résultats, comparons avec les valeurs de l'ancien dispositif de navigation.

ETAT DE L'ART

PRESENTATION



Les centrales inertielles sont à la base de nombreux systèmes de navigation notamment ceux des navires, des objets volants, des bateaux et des missiles.

Dans le cas du projet SPARROW, le vol de notre drone nécessite une centrale inertielles*.

Plate-forme inertielles ST-124-M installée sur le lanceur Saturn V (1970)

<http://www.de-la-terre-a-la-lune.com/>

SYSTEMES DE NAVIGATION

Dans les domaines aéronautiques, spatiaux, maritimes et terrestres, les systèmes de navigation des engins civils et militaires nécessitent de connaître précisément la position et la vitesse de ceux-ci.

Les deux principales approches pour ces applications sont :

- ✓ La navigation par satellite qui repose sur les systèmes de positionnements GPS*. La position du mobile est donnée en longitude et en latitude. Le GPS* s'est rapidement développé du fait de son faible cout et de sa précision acceptable.
- ✓ La navigation inertielles qui repose sur l'utilisation des centrales inertielles (Inertial measurement unit en anglais). Il s'agit d'un composant électronique de navigation de précision qui mesure l'accélération et la vitesse angulaire à l'aide de gyromètres* et d'accéléromètres. Grâce à ces capteurs, et à partir des conditions initiales, nous pouvons calculer à chaque instant la position et la vitesse du véhicule.

Les centrales inertielles sont souvent composées de trois accéléromètres* et trois gyromètres*.

Dans leur utilisation normale, les centrales inertielles définissent l'attitude (les mouvements et l'orientation) d'une plateforme. On peut donc aussi les utiliser pour des manettes de jeux vidéo, des Smartphones, des lunettes pour aveugles... Cependant, les IMU* donnent des mesures qui divergent au cours du temps par rapport au déplacement et à la position réelle.

GPS

Avantages : faible coût, performances importantes, pas de divergence du positionnement

Inconvénients : faible précision

Centrale inertielles

Avantages : grande précision, autonomie, indépendance par rapport aux conditions extérieures.

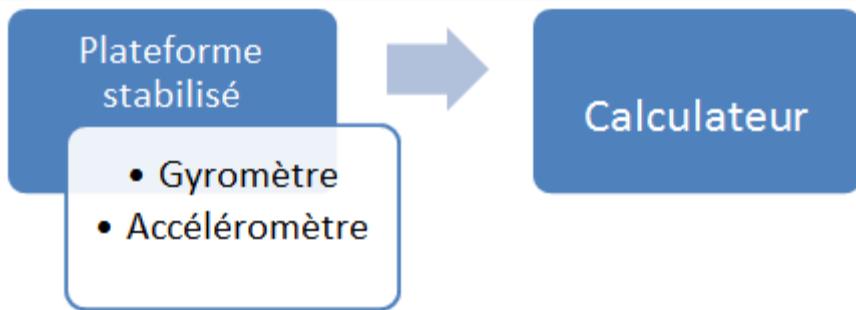
Inconvénients : cout, divergence de l'estimation des capteurs

Au vu de cette analyse comparative, il convient de chercher à coupler ces deux systèmes pour combler chacun des désavantages (on empêche la dérivation du système inertielles en corrigeant sa position à l'aide du GPS*). La combinaison des deux technologies est utilisée pour de nombreux systèmes.

La précision est un facteur clé pour assurer la maîtrise du vol. Cependant, dans le cadre de la réalisation d'un système à cout faible, nous utiliserons une centrale inertielles* seule.

LES TECHNOLOGIES

Les centrales inertielles sont composées de gyroscopes et de capteurs d'accélération. Ils fournissent la vitesse angulaire et calculent, à partir de ces mesures, l'évolution du vecteur vitesse ainsi que de son attitude (roulis, tangage, lacet).

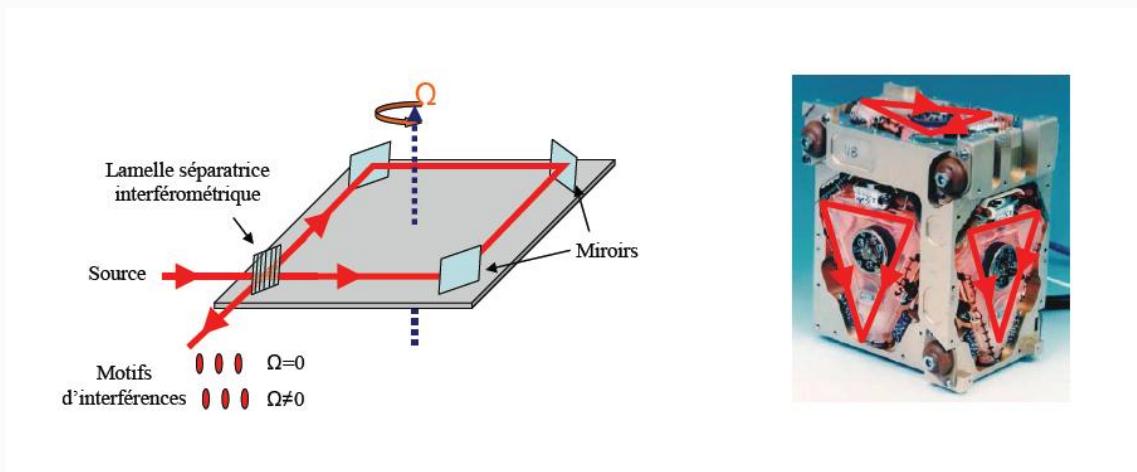


LES GYROMETRES*

Les gyromètres* permettent de déterminer les évolutions d'attitude, c'est-à-dire les variations angulaires. Ils fournissent une vitesse de rotation dont l'intégration donne la variation angulaire. On distingue plusieurs technologies.

Les gyromètres*laser:

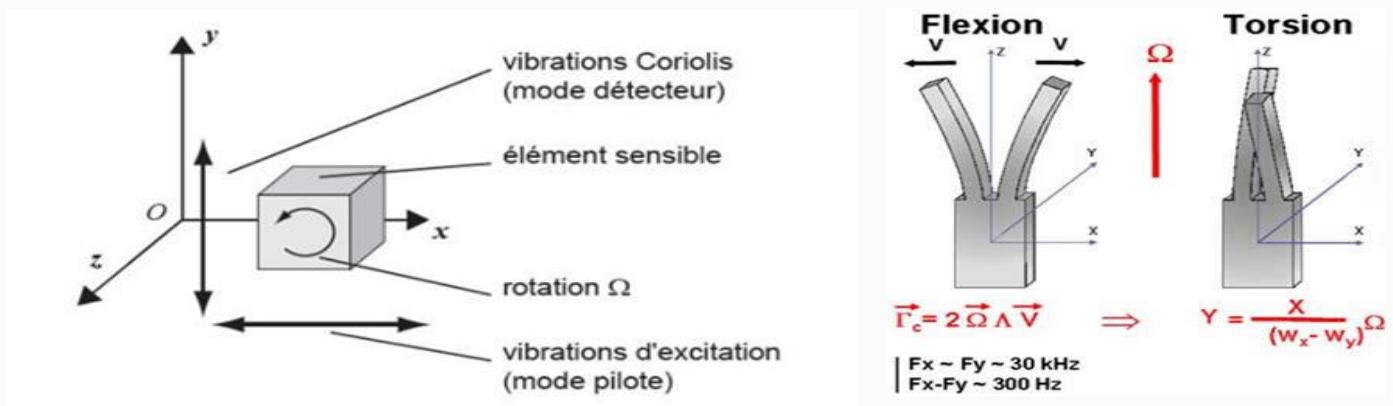
« Il s'agit d'un triangle isocèle qui comporte un capteur de lumière au sommet des cotés identiques. Au centre du côté opposé, un faisceau laser est divisé en deux et rejoint le sommet opposé par deux chemins différents. Au repos, les deux faisceaux arrivent en même temps, ce qui n'est plus le cas si le triangle est soumis à une rotation. La différence de temps est alors une image de la vitesse angulaire. »



Gyromètre* laser : <http://hal.archives-ouvertes.fr/>

Les gyromètres*vibrants

«Un gyromètre* vibrant est composé d'un élément sensible possédant la capacité de pouvoir osciller selon deux axes (x et y) orthogonaux. Lorsque l'élément sensible est en vibration le long de l'axe x, si le gyromètre* est soumis à une rotation autour de l'axe z, une vibration apparaît le long de l'axe y. Cette vibration est excitée par la force de Coriolis et son amplitude est proportionnelle à la rotation.»



Gyromètre* vibrant : <http://www.onera.fr>

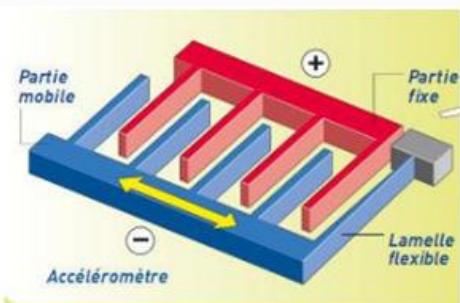
<http://hal.archives-ouvertes.fr/>

«Afin d'augmenter la sensibilité du gyromètre*, les deux vibrations orthogonales ont des fréquences de résonances proches (le maxIMU*m de sensibilité correspondant à l'égalité des deux fréquences). Le premier mode, dénommé mode pilote, est excité à sa résonance avec une amplitude asservie pour être constante. Dans ces conditions, l'amplitude de la vibration du second mode, appelé mode détecteur, est donc uniquement proportionnelle à la vitesse de rotation. La mesure de l'amplitude de vibration du mode détecteur réalise ainsi la mesure de la vitesse de rotation.»

«Ces gyromètres* mesurent les accélérations de Coriolis Σ_c qui apparaissent lors d'une rotation. Dans ce but, on excite continûment à sa fréquence de résonance (mode pilote de pulsation Ω_x et d'amplitude X) un résonateur, au moyen d'un circuit oscillateur. Ces accélérations de Coriolis sont alors alternatives à la pulsation Ω_x et excitent en vibration forcée un mode transverse (de pulsation Ω_y) dont l'amplitude Y est proportionnelle à la vitesse de rotation Ω subie.»

LES ACCELEROMETRE*S

Les accéléromètres* mesurent l'accélération suivant l'axe de l'objet qui les supportent. Ces capteurs peuvent être mono, bi ou triaxial. On distingue deux grandes familles d'accéléromètres* : les accéléromètres*non asservis et les accéléromètres*à asservissement.



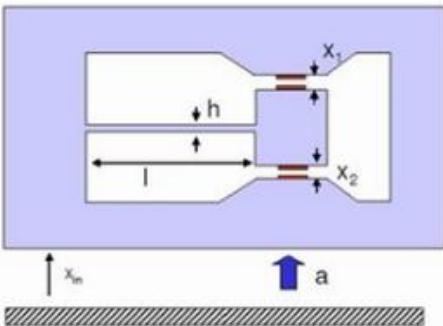
Accéléromètre* de type MEMS
<http://www.wilcoxon.com>

Les accéléromètres*de type MEMS

«L'accéléromètre* Mems est composé de deux peignes complémentaires conducteurs en Silicium. L'un (en rouge) est fixe. L'autre en bleu est mobile, suspendu à une lamelle flexible, et constitue une masse sensible à l'accélération. Le peigne mobile peut se déplacer d'une vingtaine de nanomètre par inertie lorsque le véhicule change de vitesse»

Les accéléromètres*capacitifs:

«Ils mesurent la variation d'une capacité. Il s'agit d'une masse mobile en équilibre dans un cadre. Lors d'une accélération, la masse bouge et se rapproche d'un des bords du cadre. Comme la capacité d'un condensateur est inversement proportionnelle à la distance séparant ses deux armatures, on peut dire que la capacité est fonction de l'accélération $C = f(a)$.»



Accéléromètre*s capacitatifs <http://www.forum-auto.com/>

«Il existe également d'autres technologies d'accéléromètres*tels que les accéléromètres*piézoélectriques, piezorésistifs et les accéléromètres*à effet de Hall.»

«Le problème lié au rapport poids/performance des objets volant a poussé la recherche à s'orienter vers la miniaturisation des composants. Ainsi, les entreprises de recherche se penchent désormais sur la réalisation de micro-accéléromètres*vibrants et de micro-gyromètres*vibrants. Toutefois, la réalisation de centrale inertielles* pour micro-drones lance un nouveau défi puisqu'il s'agit de gagner un à deux ordres de grandeur sur l'encombrement sans trop altérer la précision des capteurs.

Actuellement, le quartz est la matière privilégiée mais les recherches tendent à s'orienter vers une matière aux propriétés immenses : le silicium.»

LES DERIVES INERTIELLES

Les dérives inertielles dépendent des caractéristiques des capteurs utilisés. On peut les recensées selon quatre paramètres :

- ❖ Erreur de biais
- ❖ Erreur de facteur d'échelle
- ❖ Erreur d'alignement des axes
- ❖ Bruit

BAIS

Le biais est une valeur additionnelle qui est déterminée en appliquant une entrée nulle.

Néanmoins, il arrive qu'il soit différent à chaque fois que le capteur est remis en marche. Par ailleurs, sa valeur n'est pas fixe au cours du temps. Ceci est dû aux variations de la température. Ainsi, la relation obtenue est : **Mesurée = vraie + biais**

FACTEUR D'ECHELLE

Cette erreur est due essentiellement à la fabrication du matériel. Sa détermination se fait le plus souvent de façon expérimentale.

$$\text{Mesurée} = K_a * \text{vraie}$$

On obtient donc en combinant les facteurs d'échelle et de biais :

$$a\text{Mesurée} = K_a * \text{vraie}.$$

Avec $a\text{Mesurée}$ = analogique mesurée

NON ORTHOGONALITE DES AXES

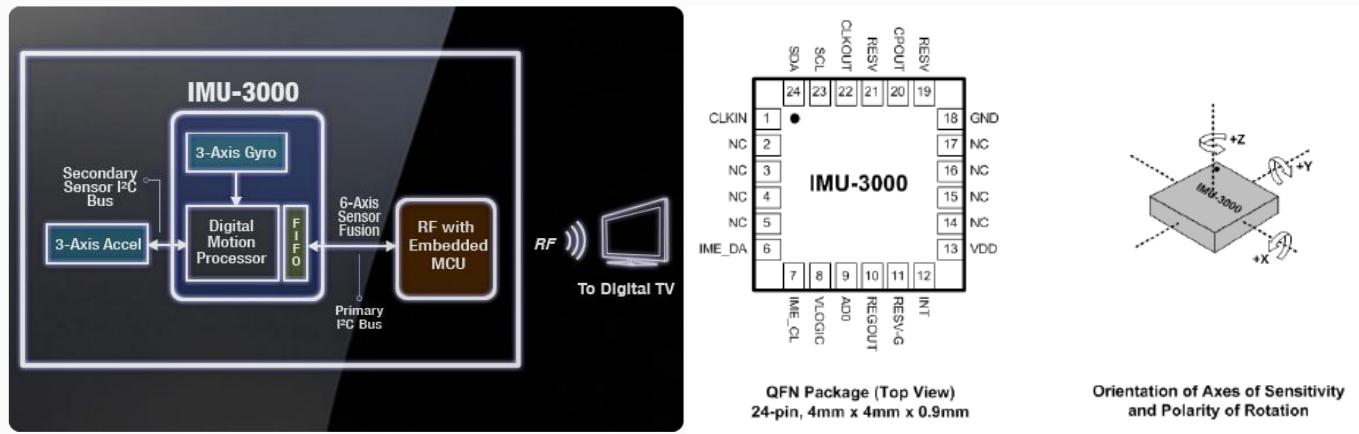
Cette erreur est liée à la fabrication de la centrale. En effet les axes des capteurs peuvent ne pas être alignés correctement avec les bords du boîtier ou ne pas être parfaitement orthogonaux. Dans ce cas il y aura une différence entre les valeurs reçues et les valeurs attendues.

BRUIT

Le bruit est un signal parasite présent dans tous les circuits électroniques. Celui-ci perturbe les réponses du système. Dans notre cas il sera donc nécessaire d'utiliser un ou plusieurs correcteurs pour compenser les perturbations et obtenir la sortie voulue.

NOS EQUIPEMENTS

LA CENTRALE INERTIELLE* (REF : IMU3000*)



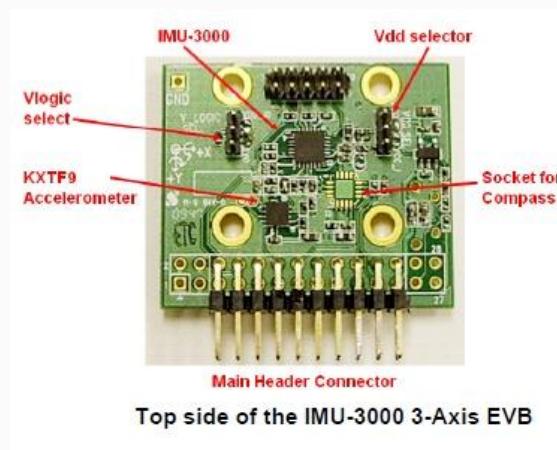
COMPOSANTS ET CARACTERISTIQUES DES DONNEES

- ❖ 3 axes numériques gyroscopes vibratoires, échelle de mesure programmable : ± 250 à ± 2000 ° /sec
- ❖ 3 convertisseurs ADC
- ❖ 3 filtres passe-bas
- ❖ DMP (digital motion processor) : sert à la gestion du temps et de la puissance de traitement.
- ❖ FIFO (first in first out sur 512 bytes) détermine les données qui seront accessibles par l'IMU*.

Communication I²C Mode Maître (2 bytes par axes) et/ou Esclave :

- “I²C Primaire” communication entre les interfaces
- “I²C Secondaire» pour communication avec les 3 accéléromètres*

- ❖ Alimentation VLOGIC : 3,3Volt DC ou 3Volt avec un régulateur
- ❖ Technologie utilisée MEMS (Micro Electro-Mechanical Systems) 24 pins
- ❖ Dimension : 4 x 4 x 0.9mm
- ❖ Fréquence de l'horloge 1MHz (Synchronisation)
- ❖ Traitement de signal : amplification, démodulation, filtrage du signal mesurant les vibrations dues à l'effet de Coriolis par l'intermédiaire des angles. Pour avoir une tension proportionnelle à la vitesse angulaire.
- Kionix KXTF9 : 3 accéléromètres* de type MEMS, 1 pour chaque axe
- AKM AK8975 : 3 boussoles, 1 pour chaque axe



CHEMINEMENT DES DONNEES

Les signaux analogiques provenant directement des gyroscopes (interne à l'IMU*3000*) et des accéléromètres* sont transmis aux ADC pour être converti en signaux numériques.

Ces signaux (sauf celui de la température) sont ensuite traités (signal conditioning) afin d'éliminer les perturbations et limiter les erreurs de mesure. Les données traitées passent ensuite dans le Registre des capteurs... et sont ensuite stockées dans la FIFO (first in first out).

La FIFO possède une mémoire de 512 octets (bytes). Chaque information récupérée a une taille de 2 octets. L'ensemble d'une mesure fait donc 14 octets :

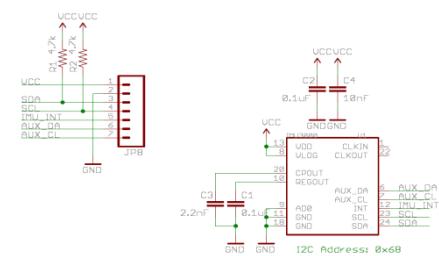
Cependant lorsque l'on veut lire les données il faut aussi lire le FIFO FOOTER (sauf pour la première lecture) qui fait lui-même 2 octets. Donc un groupement de données prendra donc 16 octets dans la mémoire du FIFO, ce qui nous donne la possibilité de stocker jusqu'à 32 groupements de données.

ORDRE DE RECUPERATION DES DONNEES

Register (Hex)	(Decimal)	
1B	27	TEMP_OUT_H
1C	28	TEMP_OUT_L
1D	29	GYRO_XOUT_H
1E	30	GYRO_XOUT_L
1F	31	GYRO_YOUT_H
20	32	GYRO_YOUT_L
21	33	GYRO_ZOUT_H
22	34	GYRO_ZOUT_L
23	35	AUX_XOUT_H
24	36	AUX_XOUT_L

EN PRATIQUE :

La centrale inertielle* fournit des données analogiques venant des gyromètres*, d'un thermomètre interne et des accéléromètres* externes. Donc il faudra calibrer, étalonner les capteurs pour que les résultats soient cohérents avec la réalité.



LE MICROCONTROLEUR* (REF : DSPIC*30F4011)

CARACTERISTIQUES



C'est un microcontrôleur* faisant partie de la famille des dsPic* produit par Microchip. Il est utilisé pour contrôler des moteurs ainsi que pour convertir et traiter des données (filtre numérique, FFT*...). Il se caractérise par son architecture de type « Modified Harvard* ».

Pour ce type de microcontrôleur* il existe plusieurs périphériques parmi lesquels:

- UART* : Liaison série asynchrone qui permet d'interfacer la carte avec l'ordinateur sous la forme d'un port série et d'afficher du texte pour du débogage
- I2C* : Liaison numérique fonctionnant sur le principe de bus et sert à communiquer entre des composants électroniques.
- Timer : Le dsPic*30F4011 est composé de 5 timers. Ce sont des compteurs internes au microcontrôleur*. Les timers du dsPic*30F4011 sont de 3 catégories différentes (cf image ci-dessous).

Timer Differences

	Type A (Timer 1)	Type B (Timers 2,4)	Type C (Timers 3,5)
Features	Can use 32 kHz osc, best for RTC operation	Clock Sync is after prescaling	Clock sync is before prescaling Can trigger an ADC
Clock Sources	External (Txck) LPOCS (32Khz) Tcy	External (Txck) Tcy	External (Txck) Tcy
Operating Modes	16-bit timer, 16-bit sync counter, 16-bit async counter	16-bit timer, 16-bit sync counter, 32-bit timer (LB), 32-bit counter(LB)	16-bit timer, 16-bit sync counter, 32-bit timer (UB), 32-bit counter(UB)

© 2003 Microchip Technology Incorporated. All Rights Reserved. Name of presentation 7

Timer du dsPic*30F4011

Output compare : mode du microcontrôleur* qui permet notamment de contrôler les servos moteur.

ADC : Analog Digital Conversion. C'est un convertisseur analogique numérique utilisé pour des capteurs tels que les gyroscopes.

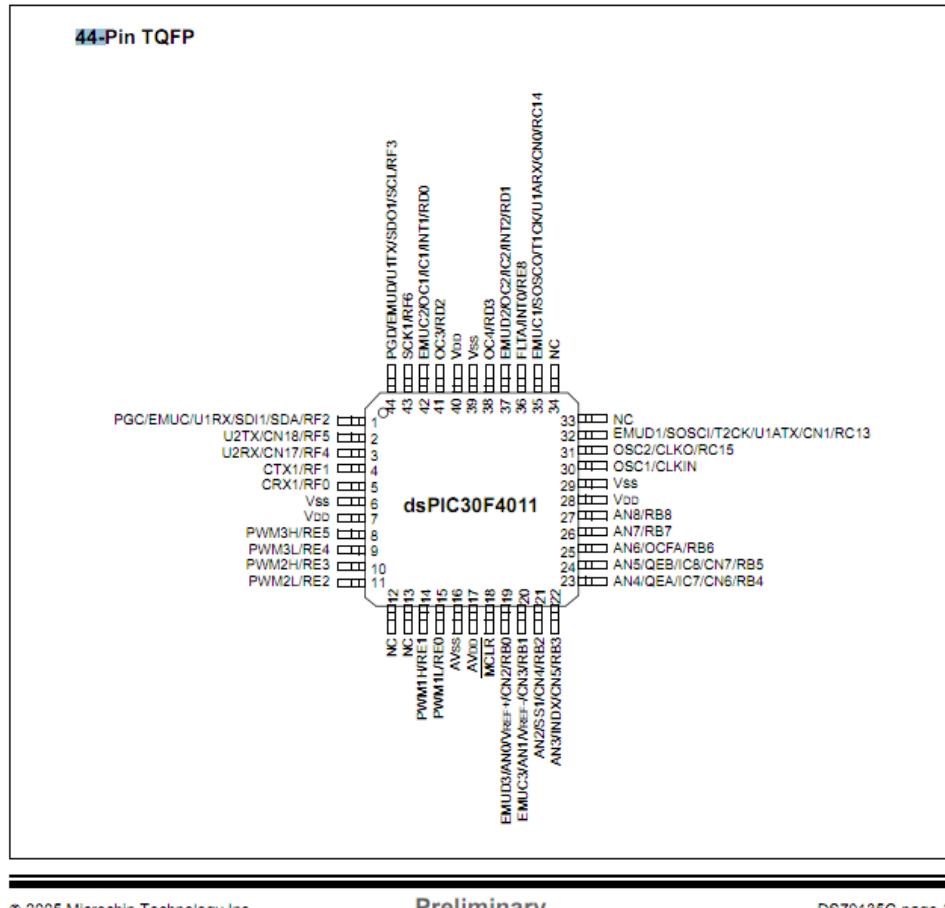
On trouve également sur le dsPic*30F4011 3 types de mémoire différents. Une mémoire Flash de 24KB, une mémoire EEPROM de 1KB et mémoire programme de 48K bytes.

dsPIC30F Motor Control and Power Conversion Family⁽¹⁾

Device	Pins	Program Mem. Bytes/ Instructions	SRAM Bytes	EEPROM Bytes	Timer 16-bit	Input Cap	Output Comp/Std PWM	Motor Control PWM	A/D 10-bit 1 Msps	Quad Enc	UART	SPI	I ² C TM	CAN
dsPIC30F4011	40/44	48K/16K	2048	1024	5	4	4	6 ch	9 ch	Yes	2	1	1	1

Caractéristique des mémoires

Le dsPic* 30F4011 possède aussi 30 pin E/S qui sont des GPIO (General Purpose Input/Out



Schémas descriptif des différentes pins du dsPic*30F4011

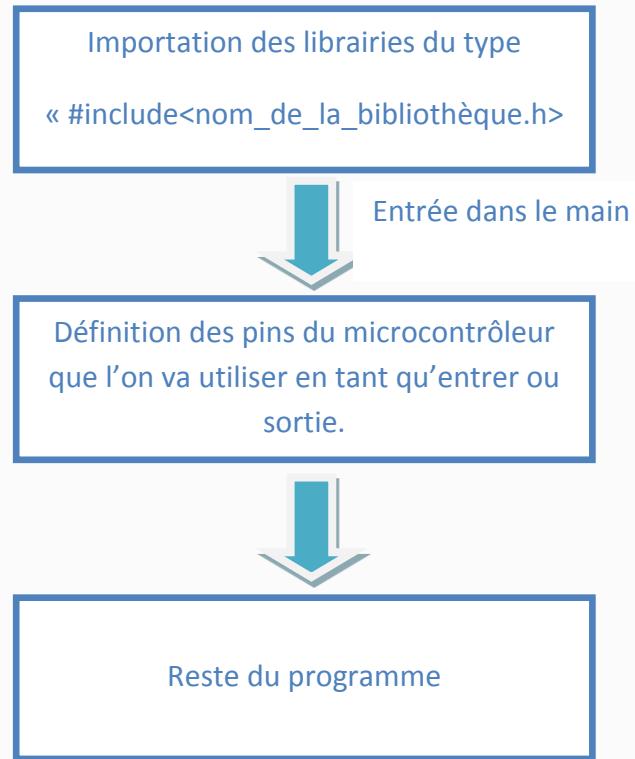
FONCTIONNEMENT

Alimentation

Pour son fonctionnement le dsPic*30F4011 doit être alimenté par une tension comprise entre 2.5 et 5V.

Structure des programmes

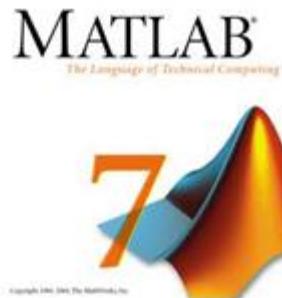
Sa programmation peut se faire en assembleur ou en langage C. En C le programme peut être découpé en 3 grandes étapes :



Objectifs

Ce microprocesseur aura pour rôle dans le projet SPARROW de récupérer les données des gyroscopes et accéléromètre* stocker dans l'IMU* 3000 dans le but de les traiter afin de les rendre utilisables.

LES LOGICIELS



MPLAB

MPLAB est un outil de développement pour les PIC. Il est disponible sur le site de Microchip et il est gratuit. C'est avec cet outil que nous mettons en œuvre notre code en langage C. Il nous a servi d'interface entre notre objectif et les composants électroniques.

EAGLE

EAGLE est un programme pour la conception de carte électronique. Il nous permet de visualiser la configuration, l'emplacement de composants sur des cartes déjà modélisées. Il nous a été utile en particulier pour les connections entre les différents équipements.

MINICOM

MINICOM est un émulateur de terminal, qui permet de communiquer avec des composants via les ports série. Nous l'utilisant, pour vérifier sur l'ordinateur les données transmises entre l'IMU* 3000 et le dsPic*. Celui-ci est configuré sous linux.

TERA TERM

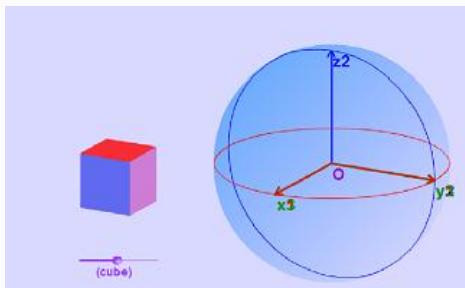
TERA TERM a la même fonctionnalité que MINICOM mais s'implémente sous Windows.

MATLAB

MATLAB est un outil de développement et un langage de programmation, qui permet de modéliser et réaliser des équations numériques complexes à base d'algorithme. Il permet de calculer l'attitude de notre centrale inertie* lorsque que nous la manipulons selon différents angles en utilisant les données des capteurs et équations d'Euler avec la matrice.

ETUDE THEORIQUE

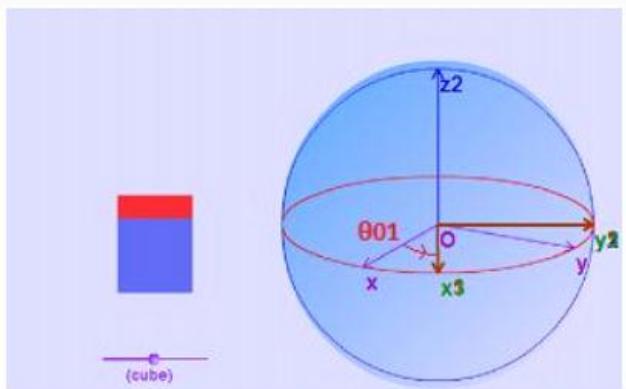
THEORIE D'EULER



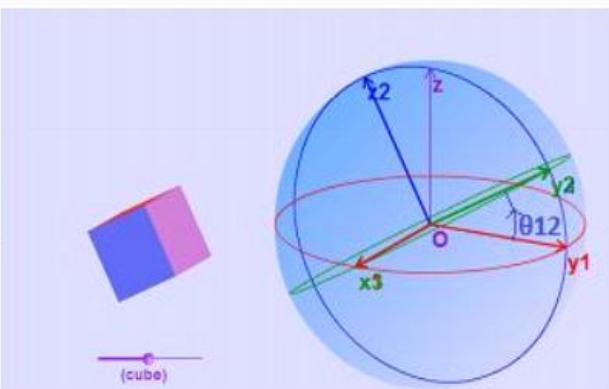
Les angles d'Euler permettent de définir l'orientation d'un objet dans l'espace. Une quelconque orientation peut être décomposée en trois rotations élémentaires. Pour cela on définit trois repères successifs.

Soit $R_0(0, xo, yo, zo)$ le repère initial définissant l'origine de référence.
 $\theta_{01}=0 \text{ rad}, \theta_{12}=0 \text{ rad}, \theta_{23}=0 \text{ rad}$

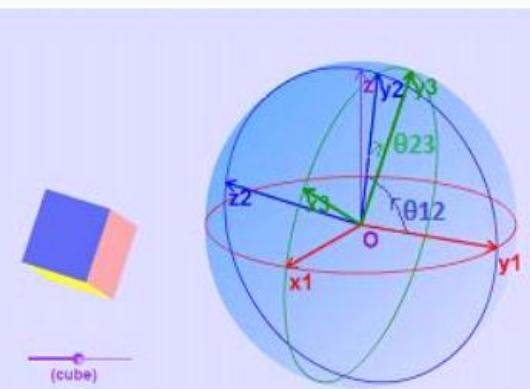
TRANSFORMATIONS ELEMENTAIRES



$\theta_{01}=0.5 \text{ rad}, \theta_{12}=0 \text{ rad}, \theta_{23}=0 \text{ rad}$



$\theta_{01}=0 \text{ rad}, \theta_{12}=0.5 \text{ rad}, \theta_{23}=0 \text{ rad}$



$\theta_{01}=0, \theta_{12}=1.45, \theta_{23}=0.69 \text{ rad}$

$R_1(01, x_1, y_1, z_1)$ le repère intermédiaire 1 définissant toutes les orientations possibles dans plan $(0, xo, yo)$.

R_1 est lié à R_0 par l'angle de précession θ_{01} souvent appelé ψ qui définit la rotation autour l'axe z_0 .

$R_2(02, x_2, y_2, z_2)$ le repère intermédiaire 2 définissant toutes les orientations possibles dans plan $(0, y_1, z_1)$.

R_2 est lié à R_1 par l'angle de nutation θ_{12} souvent appelé θ qui définit la rotation autour l'axe x_1 .

$R_3(03, x_3, y_3, z_3)$ le repère final (celui du drône) définissant toutes les orientations possibles dans plan $(0, x_3, y_3)$.

R_3 est lié à R_2 par l'angle rotation propre θ_{23} souvent appelé φ qui définit la rotation autour l'axe z_2 .

RELATIONS ELEMENTAIRES

Premier changement de repère	Deuxième changement de repère	Troisième changement de repère
$x_1 = \cos\psi * x_0 + \sin\psi * y_0$	$x_2 = x_1$	$x_3 = \cos\varphi * x_2 + \sin\varphi * y_2$
$y_1 = -\sin\psi * x_0 + \cos\psi * y_0$	$y_2 = \cos\theta * y_1 + \sin\theta * z_1$	$y_3 = -\sin\varphi * x_2 + \cos\varphi * y_2$
$z_1 = z_0$	$z_2 = -\sin\theta * y_1 + \cos\theta * z_1$	$z_3 = z_2$
Ou	Ou	Ou
$x_0 = \cos\psi * x_1 - \sin\psi * y_1$	$x_1 = x_2$	$x_2 = \cos\varphi * x_3 - \sin\varphi * y_3$
$y_0 = \sin\psi * x_1 + \cos\psi * y_1$	$y_1 = \cos\theta * y_2 - \sin\theta * z_2$	$y_2 = \sin\varphi * x_3 + \cos\varphi * y_3$
$z_0 = z_1$	$z_1 = \sin\theta * y_2 + \cos\theta * z_2$	$z_2 = z_3$

Nous passons ensuite, ses relations scalaires en équations matrices. Les propriétés mathématiques sur les relations des matrices de passages et les changements de repère permettent d'obtenir un produit de matrices élémentaires afin de passer de la première base à la dernière. De plus, grâce aux propriétés mathématiques concernant les matrices de changement de repère nous arrivons à de simples relations entre les matrices de passages et les vitesses de rotation autour de chaque axe.

$$\begin{bmatrix} \dot{G_x} \\ \dot{G_y} \\ \dot{G_z} \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \equiv J^{-1} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}$$

A la fin, nous obtenons une expression de la matrice d'Euler « J^{-1} » simplifiée

$$\begin{bmatrix} \dot{G_x} \\ \dot{G_y} \\ \dot{G_z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \sin(\theta) \\ 0 & \cos(\varphi) & -\sin(\varphi) \cdot \cos(\theta) \\ 0 & \sin(\varphi) & \cos(\varphi) \cdot \cos(\theta) \end{bmatrix} \cdot \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}$$

Ainsi en inversant la matrice on obtient bien les vitesses angulaires du repère fixe en fonction de la matrice d'Euler et des vitesses angulaires du repère mobile.

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = J \begin{bmatrix} \dot{G_x} \\ \dot{G_y} \\ \dot{G_z} \end{bmatrix} = \begin{bmatrix} 1 & (\sin\phi \tan\theta) & (\cos\phi \tan\theta) \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \end{bmatrix} \begin{bmatrix} \dot{G_x} \\ \dot{G_y} \\ \dot{G_z} \end{bmatrix}$$

Contraintes

Par contre, les angles θ_{01} et θ_{23} sont définis entre 0 et 2π . Et entre 0 et π pour θ_{12} . Car si l'on ne respecte pas ces conditions on n'aura pas une manière unique de définir une position.

Lorsque l'angle θ_{12} est nul, la rotation suivant l'axe z_0 est la même que la rotation suivant l'axe z_2 . Ainsi, on a deux possibilités pour définir une orientation c'est le phénomène du blocage de cadran.

« Il s'agit de la perte d'un degré de liberté, connue aussi sous le nom anglais de « **gimbal lock** », qui survient quand les axes de deux des trois cardans nécessaires pour appliquer ou compenser les rotations dans l'espace à trois dimensions sont portés par la même direction. »

APPLICATION

Utilisation des accéléromètres*

Les accéléromètres* sont des capteurs mesurant des accélérations (m/s^2). Cependant, nous pouvons relier cette grandeur physique à une force (Newton ou en g) grâce au principe fondamental de la dynamique.

Enoncé du principe fondamental de la dynamique

L'accélération subie par ce corps dans un référentiel galiléen est proportionnelle à la résultante des forces qu'il subit, et inversement proportionnelle à sa masse m .

$$\sum \vec{F}_i = m\vec{a}$$

Dans le cas idéal, cette force mesurée que nous représenterons par le vecteur F_m est la résultante de trois forces élémentaires représentant les projections sur les trois axes de F_m : F_{mx} , F_{my} et F_{mz} .

On définit ainsi les angles associés β_x , β_y , et β_z entre F_m et les axes du repère initial $R_0(0, x_0, y_0, z_0)$.

Propriétés

Géométrie

$$F_m^2 = F_{mx}^2 + F_{my}^2 + F_{mz}^2$$

$$\beta_x = \arccos(F_{mx}/F_m) \quad \beta_y = \arccos(F_{my}/F_m) \quad \beta_z = \arccos(F_{mz}/F_m)$$

Cosinus directeur

$$\cos(\beta_x)^2 + \cos(\beta_y)^2 + \cos(\beta_z)^2 = 1$$

Caractéristiques de l'accéléromètre* utilisé (Kionix KXTF9) :



Digital I2C* Communication Rate KHz 400 max
Sensibilité (counts/g) : 64, 32, 16 pour un 8-bit et 1024, 512, 256 pour un 12-bit.
(1 g = 9.80665 m/sec²).

Traitement des données

Soit Fmxa, Fmya, et Fmza les valeurs analogiques mesurées.
amplitudeMax=2^{(nombre de bit)-1}. Vzero est donnée dans la datasheet.

$$Fmx = (Fmxa * Vref/amplitudeMax - Vzero)/sensibilité$$

$$Fmy = (Fmya * Vref/amplitudeMax - Vzero)/sensibilité$$

$$Fmz = (Fmza * Vref/amplitudeMax - Vzero)/sensibilité$$

Les données provenant des capteurs analogiques sont échantillonnées et converties en données numériques Fmx, Fmy, Fmz. Celles-ci représentent la sortie directe des capteurs.

Pin Number	Pin Name	Pin Description
6	AUX_DA	Interface to a 3rd party accelerometer, SDA pin. Logic levels are set to be either VDD or VLOGIC. See Section 7 for more details.
7	AUX_CL	Interface to a 3rd party accelerometer, SCL pin. Logic levels are set to be either VDD or VLOGIC. See Section 7 for more details.

Utilisation des gyroscopes

Le IMU*-3000 est composé de trois indépendants gyroscopes vibrant (technologie : MEMS), qui détectent la rotation autour de leurs axes respectifs. Pendant la rotation, l'effet de Coriolis provoque une vibration. Celle-ci est détectée par un capteur d'angle capacitif.

Le signal résultant est amplifié, démodulé et filtré pour produire une tension qui est proportionnelle à la vitesse angulaire. Cette tension est numérisée à l'aide d'un convertisseur analogique numérique à 16 bit
L'échelle des capteurs gyroscopiques peut être numériquement programmée de ± 250 , ± 500 , ± 1000 , ou ± 2000 degrés par seconde ($^{\circ}/s$). Le taux d'échantillonnage du CAN est programmable sur 8000 échantillons par seconde.

Caractéristiques du gyroscope :

Parameter	Typical	Unit
Full-Scale Range	± 250 à ± 2000	$^{\circ}/s$
Gyro ADC Word Length	16	Bits
Sensitivity Scale	131	LSB/ $(^{\circ}/s)$

Les données des gyroscopes sont avant d'être traitées par le registre 22 DLPF, Full Scale. Dans celui-ci, on configure deux paramètres. Le premier FS_SEL consiste à choisir la plage d'échelle de l'acquisition qui va entre $\pm 250^{\circ}/sec$ et $\pm 2000^{\circ}/sec$. Le second DLPF_CFG sert à déterminer la bande passante du filtre passe bas et le taux d'échantillonnage analogique.

Ensuite les registres 12 à 17 Gyro Offset permettent de supprimer les biais des mesures des capteurs. Enfin c'est les registres 27 à 40 Sensor Registers qui traitent les données.

Echelle de conversion : 0 représente 0,0 Volts, et $2^n - 1$ représente environ 5 volts (n bits de résolution)

Traitement des données

Soit α_x , α_y , et α_z les valeurs analogiques mesurées. amplitudeMax= 2^n (nombre de bit)-1. Vzero est donnée dans la datasheet.

$$\begin{aligned}\alpha'x &= (\alpha_x * Vref/amplitudeMax - Vzero)/sensibilité \\ \alpha'y &= (\alpha_y * Vref/amplitudeMax - Vzero)/sensibilité \\ \alpha'z &= (\alpha_z * Vref/amplitudeMax - Vzero)/sensibilité\end{aligned}$$

Utilisation combinée de gyroscopes et d'accéléromètres*

Faire attention à la configuration d'usine de IMU* pour savoir si les axes sont bien orientés comme on a le pense. Le meilleur moyen de vérifier est de tester.

On n'utilise pas uniquement l'accéléromètre* car il n'est pas à 100% fiable. En effet, il ne détecte que les forces inertielles et pas les forces gravitationnelles. De plus, il est sensible au vibration et au bruit. Contrairement au gyroscope qui est moins sensible aux mouvements linéaires mais qui rencontre des soucis de réinitialisation. Finalement, la combinaison des deux technologies, et de leur méthode d'acquisition de la position du système permet d'avoir une information constamment stable et fiable.

Procédure d'étalonnage

Lors de l'utilisation des équipements nous prévoyons d'étalonner les divers composants. De plus, nous avons aussi implémenté des vérifications afin de savoir l'état de fonctionnement des différents matériels. Souvent, un simple clignotement d'une LED nous satisfait.

ALGORITHME

- Récupération de l'information de l'accéléromètre*
- Vérification
- Correction de l'erreur avec les informations du gyroscope => Redéfinition de la position estimée
- Approbation de la position estimée comme étant la meilleure position

Nous avons une alternative avec le logiciel que le constructeur peut nous fournir : le SDK IMU 3000. Il permettrait de faire faire les calculs par le Digital Motion Processor ou d'avoir accès à des algorithmes déjà chargés. En attendant, nous avons réalisé un petit algorithme afin de simuler le calculateur et ainsi avoir la position du micro drone.

%Données utilisées via la datasheet

```
vref=5;
n=16;
amplitudeMax=2^n-1;
vzero=2.5;
sensibilite=131;
```

%Les angles d'euler

```
phi=1; %angle de la troisième rotation entre la base 3 et la base 2
teta=2; %angle de la deuxième rotation entre la base 2 et la base 1
```

%Matrice de passage de la base fixe à la base mobile %%Lecture en ligne

```
M1=[1 0 sin(teta); 0 cos(phi) -sin(phi)*cos(teta); 0 sin(phi) cos(phi)*cos(teta)];
```

%Matrice inverse

```
I=inv(M1);
```

%Matrice de passage de la base mobile à la base fixe

```
M2=[1 sin(phi)*tan(teta) cos(phi)*tan(teta); 0 cos(phi) -sin(phi); 0 sin(phi)/cos(teta) cos(phi)/cos(teta)];
```

%verification justel: M2=inv(M1)

%Les données à tester

%Les vitesses de rotation des gyroscopes analogiques

```
Gxa=90%Gx=(0:2)%Gx=(0:180)
```

```
Gya=0%Gy=(0:2)%Gy=(0:180)
```

```
Gza=0%Gz=(0:2)%Gz=(0:180)
```

Ga=[Gxa Gya Gza]

%Conversion en numérique

```
Gxn=(Gxa*vref/amplitudeMax-vzero)/sensibilite
```

```
Gyn=(Gya*vref/amplitudeMax-vzero)/sensibilite
```

```
Gzn=(Gza*vref/amplitudeMax-vzero)/sensibilite
```

%Multiplication de la matrice de passage pour obtenir le vecteur de rotation dans la base fixe

W=Ga*M2;

%Intégration de la vitesse de rotation pour avoir la position angulaire

PROTOCOLES DE COMMUNICATION

Nous avons vu dans les parties précédentes les équipements que nous avons utilisés au cours de ce projet. Cependant, il est essentiel d'interagir avec eux. Pour cela, au niveau le plus élémentaire on intervient avec les cartes électroniques. Toutefois, il existe aussi des outils normalisés qui permettent la communication entre des composants. Nous en avons découvert deux principaux

I2C*: INTER INTEGRATED CIRCUIT

C'est un protocole de communication entre plusieurs cartes électroniques grâce à 3 fils : un signal de données (SDA), un signal d'horloge synchrone (SCL), et un signal de référence électrique (GND). C'est une liaison série dont la vitesse de transfert est de 100 kbits/sec.

L'avantage:

L'optimisation des connexions entre les composants. La charge maxIMU*m des SDA et CL est 400pF.

Contraintes:

Utiliser les mêmes références de potentiels.

Transmission:

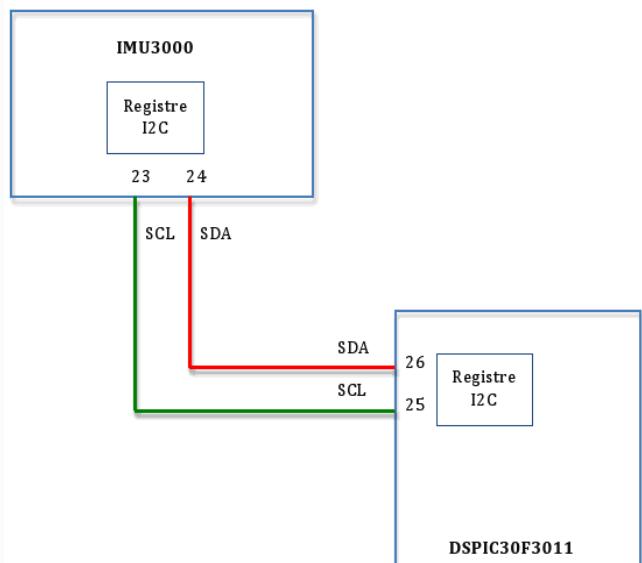
-Condition de départ : SDA et SCL sont au repos = état haut (1)

-Prise de contrôle : SDA à l'état bas (0) =>contrôle verrouillé c'est le MASTER

-Condition d'arrêt

GLOSSAIRE

ABONNE	tout élément connecté sur le bus.
EMETTEUR	tout abonné qui envoie des données sur SDA.
RECEPTEUR	tout abonné qui reçoit des données de SDA.
MAITRE	tout abonné qui démarre et termine un échange.
ESCLAVE	tout abonné adressé par un maître.
ADRESSE	numéro attribué à un esclave.
ECHANGE	dialogue entre un maître et un esclave.
ARBITRAGE	résolution du conflit d'un accès simultané par 2 maîtres.

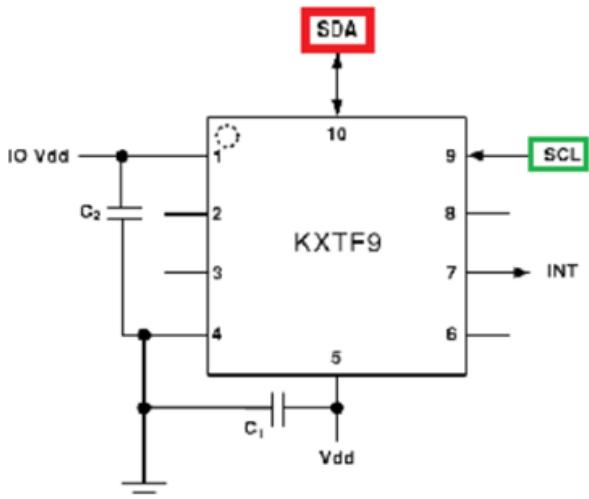


CABLAGE

¤Les lignes SCL et SDA sont à VCC si personne ne parle.

¤Pour mettre 1 sur SCL ou SDA, un abonné programme le port en entrée, la résistance se charge de tirer la ligne à 1

¤Pour mettre 0 sur SCL ou SDA, un abonné doit écrire un 0, c-à-d relier la ligne à la masse.



Pin	Name	Description
1	IO Vdd	The power supply input for the digital communication bus.
2	DNC	Reserved – Do Not Connect
3	DNC	Reserved – Do Not Connect
4	GND	Ground
5	Vdd	The power supply input.
6	DNC	Reserved – Do Not Connect
7	INT	Physical Interrupt
8	DNC	Reserved – Do Not Connect
9	SCL	I ² C Serial Clock
10	SDA	I ² C Serial Data

KXTF9 Pin Descriptions

Relation avec les accéléromètre*

IMPLEMENTATION

1. Le maître envoie la condition de démarrage (S) et contrôle le signal d'horloge
2. Le maître envoie une adresse unique de 7-bit correspondant à l'identifiant de l'esclave
3. Le maître signale si le mode lecture ou écriture (R / W). Ce qui implique que l'esclave est soit en mode réception (0), soit en mode émission (1).
4. Le récepteur envoie un bit d'accusé de réception (ACK)
5. Le transmetteur envoie 1 octet de données
6. Le récepteur émet un bit d'accusé de réception (ACK) pour l'octet reçu
7. Envoie et réception des données.
- 8.a) Le maître envoie la condition d'arrêt en écriture(P) après le dernier octet de données.
- 8.b) Le maître envoie le bit de non accusé de réception (NACK) puis la condition d'arrêt en lecture(P) après le dernier octet de données.

Single-Byte Read Sequence

Master	S	AD+W		RA		S	AD+R			NACK	P
Slave			ACK		ACK			ACK	DATA		

Burst Read Sequence

Master	S	AD+W		RA		S	AD+R			ACK		NACK	P
Slave			ACK		ACK			ACK	DATA		DATA		

I²C Terms

Signal	Description
S	Start Condition: SDA goes from high to low while SCL is high
AD	Slave I ² C address
W	Write bit (0)
R	Read bit (1)
ACK	Acknowledge: SDA line is low while the SCL line is high at the 9 th clock cycle
NACK	Not-Acknowledge: SDA line stays high at the 9 th clock cycle
RA	IMU-3000 internal register address
DATA	Transmit or received data
P	Stop condition: SDA going from low to high while SCL is high

Premier programme

Programme de test qui initialise :
les registres de contrôle de l'IMU*3000*
le protocole de communication I2C*
récupère dans la FIFO la donnée du gyroscope X.

On programme les méthodes suivantes qui serons
appelés dans le main() :

Initialisation du protocole I2C* avec la fonction : void
I2C*_Init()
Mise du dsPic* en mode Master

Configuration de la FIFO avec la fonction :
MI2C*InitialisationFIFOenable()
Configure le registre 18 FIFO enable.

On Envoi la valeur 01000000 soit le GYRO_OUT 1
l'adresse 12 pour récupérer uniquement une valeur
du gyroscope contenu dans la FIFO.

Configuration de l'interruption :
MI2C*InitialisationInterruptConfiguration()
On configure le registre 23 interrupt configuration

MI2C*InitialisationUserControl()
On configure le registre 61 user control.
On Envoi la valeur 11000000 l'adresse 3D qui
correspond au registre user control.

Mise en place de l'interruption : MI2C*Interrupt().

Interruption du main() par cette fonction
pour récupérer les données dans la pile FIFO
A partir de ce point on la donnée du gyroscope 1.

Second programme

Ce programme a les mêmes initialisations que le premier programme. De plus, il a comme fonction
de récupérer toutes les données et de les traiter. En cours de développement.

Nous avons rencontrés comme problème principal dans cette partie notre absence de connaissance sur la méthode à employer pour tester un code sur DSPIC. Nous avons été habitué à débugger en utilisant la console, ici, nous avons dû trouver de nouvelles stratégies comme faire clignoter des leds ou afficher le résultat grâce à l'UART.

UART*: UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER

C'est un protocole de communication asynchrone très courant entre plusieurs composants électroniques via un standard d'interface RS232. Celui-ci repose sur l'utilisation de 4 fils : un signal de transmission (TX), un signal de réception (RX), un signal de référence électrique (GND) et un signal d'alimentation (VCC). C'est une liaison série dont la vitesse de transfert peut varier entre 110 et 3 686 400 baud/s.

Avantage:

- Il est facile à mettre en œuvre.
- Il est peu perturbé par le bruit.

Contrainte:

- il a une faible vitesse de transfert comparé aux liaisons série synchrone.

Dans notre cas, l'UART* va nous permettre de vérifier sur le pc les données que la centrale inertie* manipule.

GLOSSAIRE

TXD : signal binaire transmis par l'émetteur

TXC : horloge de transmission de l'émetteur (signal non transmis)

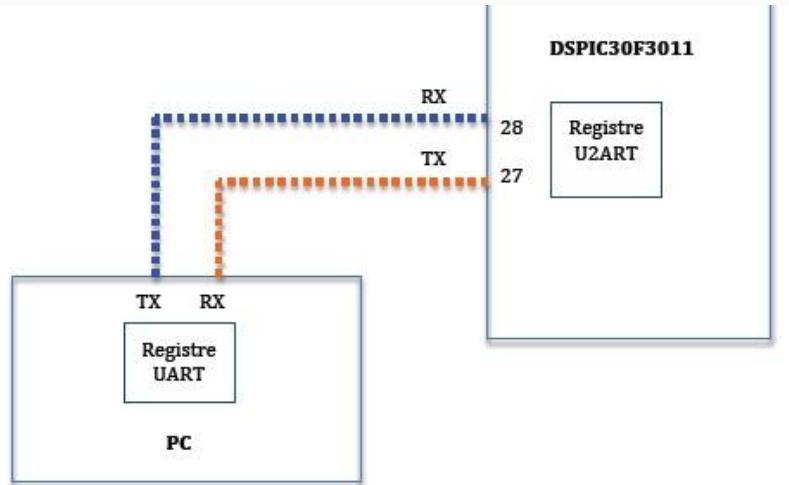
RXD : signal binaire reçu par le récepteur. En principe identique à TXD s'il n'y a pas d'erreur

RXC : horloge de réception reconstituée dans le récepteur

START : indicateur de début de la transmission d'un symbole

PARITE : indicateur d'erreur de transmission

STOP : indicateur de fin de transmission d'un symbole



CABLAGE DES LIAISONS DE COMMUNICATION

CABLAGE

Condition de départ : TXD est état haut (1) tandis que START est toujours état bas (0)

Condition d'arrêt : STOP est toujours à l'état haut (1)

CONFIGURATION

Il faut configurer :

Le registre de réception : UxMODE

UART*EN sera mis à 1 afin d'activer l'UART*.

Nous n'utiliserons qu'un bit de stop et aucun bit de parité, donc PDSEL à 00 et STSEL à 0.

Toutes les autres options ne nous intéressent pas alors nous les laissons à 0.

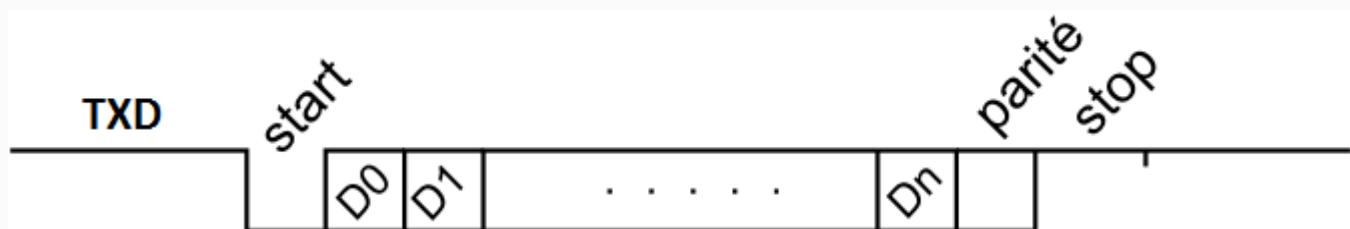
Finalemenent on a UxMODE=0x8000.

Le registre de transmission : UxSTAT

L'activation de bit UTXEN permet la transmission. La vitesse de transfert : UxBRG attention à ne pas dépasser une trame de 16bits $UxBRG = (PLL * FOSC / 4) / (16 * baudrate) - 1$ Dans notre cas, FOSC = 7352MHz et PLL=16 donc on a UxBRG=15 par approximation.

IMPLEMENTATION:

Description de la structure



Sachant que nous avons une fréquence de traitement de 400KHz, on en déduit qu'il y a un cycle à 2.5µs pour 1 bit. Par conséquent, pour un train défini de :

« start+adress+R/W+ACK+CMD+ACK+XOUTH+ACK+XOUTL+ACK+YOUTH+ACK+YOUTL+ACK+ZOUTH+ACK+ZOUTL+ACK+stop »

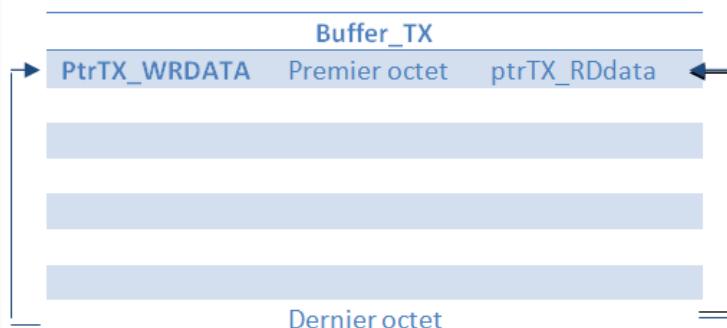
Soit 74 bits et donc un cycle à 185µs.

L'UART* n'utilise aucun protocole pour transmettre ou recevoir des données, d'où sa simplicité de mise en œuvre. Nous utilisons des interruptions pour éviter de perdre des données ou pour maximiser sa vitesse.

Les fonctions de transmission et de réceptions utilisent des tampons mémoire circulaire.

Tampon Circulaire en transmission

« ptrTX_WRDATA » et « ptrTX_RDdata » pointeurs qui contient une adresse pointant une case mémoire du buffer.

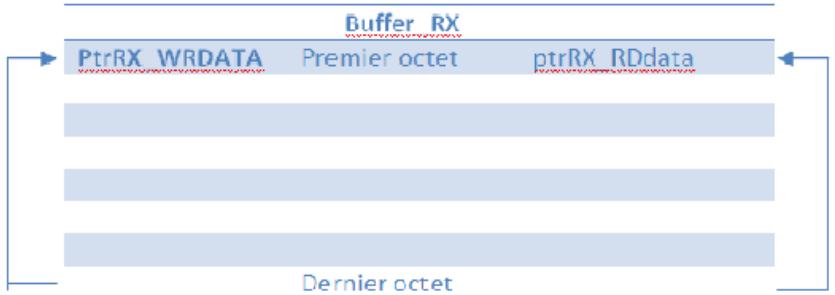


« Reset » du microcontrôleur : Initialisation pointeurs « ptrTX_WRdata » et « ptrTX_RDdata » à l'octet de départ du « buffer »

Données transmises : premier octet envoyés
Octet rangé dans le buffer à l'adresse « ptrTX_data »
Incrémantation pour pointer sur la case vide qui suit.
Si le pointeur dépasse la plage mémoire du buffer, il est affecté par l'adresse de la case une.
Si aucune transmission, on provoque une interruption :
-Si le Buffer est vide les pointeurs sont égaux
-Lecture du second octet pointée par « ptrTX_RDdata »
-Transmission de l'octet par le coupleur UART du dsPIC
-Incrémantation du pointeur RDdata si plage mémoire dépasser
-Acquittement de l'interruption

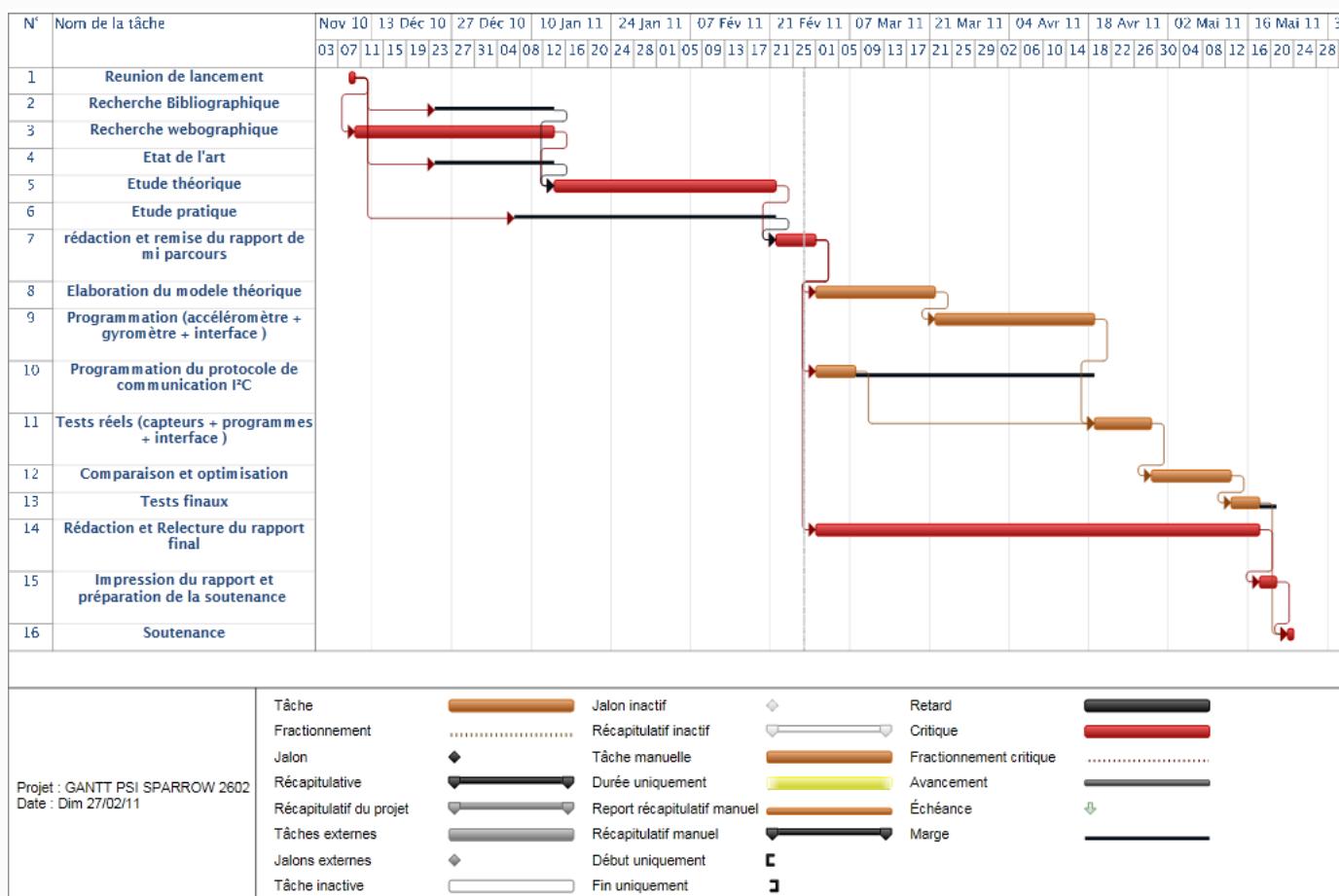
Transmission circulaire en reception

« Reset » du microcontrôleur : Initialisation pointeurs « ptrRX_WRdata » et « ptrRX_RDdata » à l'octet de départ du « buffer »
Réception des données et interruption à chaque octet reçu
Acquittement de l'interruption
Octet reçu et placer dans l'adresse pointée par « ptrRX_WRdata »
Incrémantation de ptrRX_WRdata. Si contenu du pointeur dépasse la taille du buffer.
Indiquer que le buffer est vide donc « ptrRX_WRdata » et « ptrRX_RDdata » pointent la même case
S'ils ne sont pas égaux :
Lire le second octet pointé par « ptrRX_RDdata »
-Incrémenter le contenu du pointeur « ptrRX_RDdata ». Si dépassement de la plage mémoire du buffer.



ORGANISATION DU PROJET

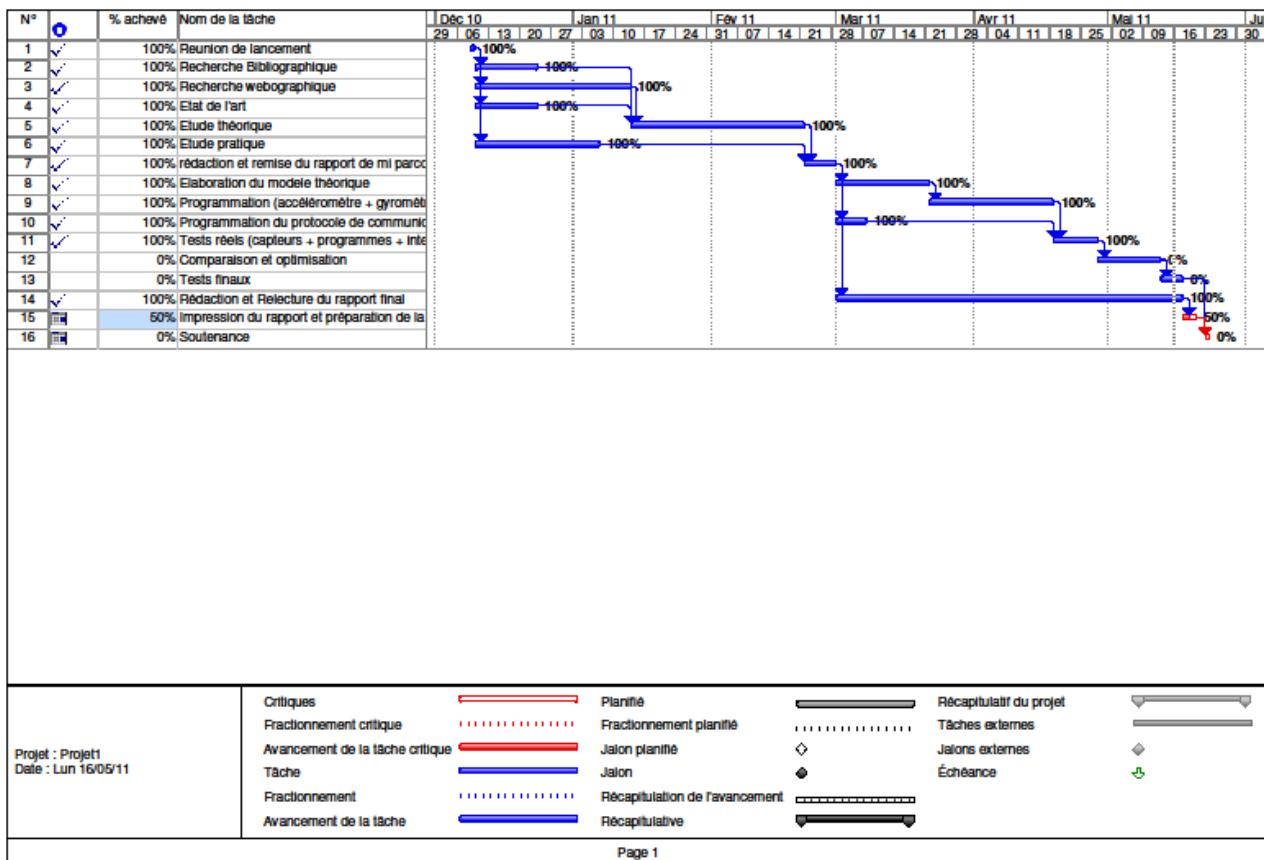
DIAGRAMME DE GANTT



Nous avons choisi d'élaborer un support de suivi de projet, pour nous permettre d'être plus rigoureux et d'éviter d'être dépassé par le temps.

Nous avons essayé d'appliquer au mieux les conseils de Mr Vedel, professeur de management que nous avons suivi lors du second semestre. Ainsi, nous avons tenté d'analyser les étapes principales de ce PSI. Nous les avons ordonné et estimé le temps de travail que nous allions consacrer à chaque étape à partir de notre maigre expérience.

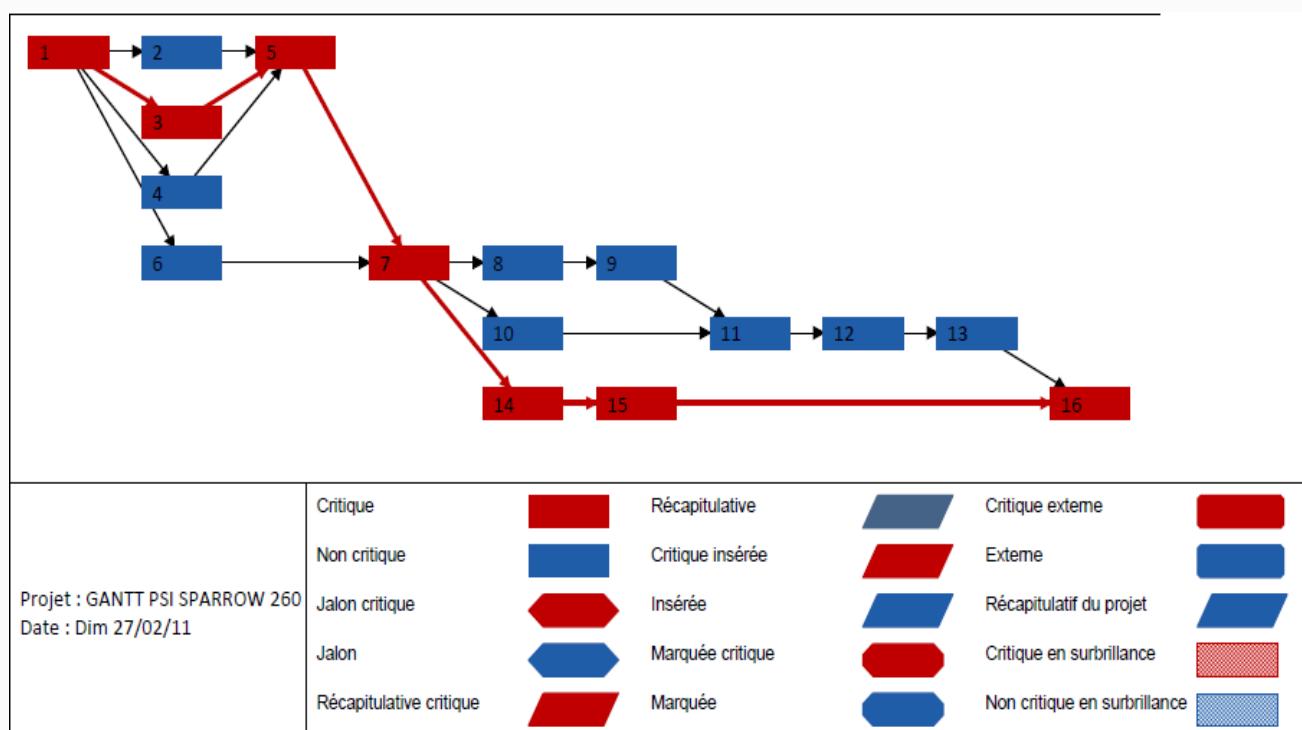
Nous avons donc tenté de réaliser un diagramme qui colle au mieux à notre projet.



A l'heure du bilan, nous constatons que notre estimation n'était pas trop mauvaise : il nous reste deux semaines avant la soutenance soit 9 jours ouvrés pour effectuer des tests, de comparer et d'optimiser notre programme.

	Nom de la tâche	Durée	Début	Fin	Prédeci	Marge totale
1	Reunion de lancement	1 jour	Jeu 09/12/10	Jeu 09/12/10		0 jour
2	Recherche Bibliographique	10 jours	Ven 10/12/10	Jeu 23/12/10	1	15 jours
3	Recherche webographique	25 jours	Ven 10/12/10	Jeu 13/01/11	1	0 jour
4	Etat de l'art	10 jours	Ven 10/12/10	Jeu 23/12/10	1	15 jours
5	Etude théorique	27 jours	Ven 14/01/11	Lun 21/02/11	2;3;4	0 jour
6	Etude pratique	20 jours	Ven 10/12/10	Jeu 06/01/11	1	32 jours
7	rédaction et remise du rapport de mi parcours	5 jours	Mar 22/02/11	Lun 28/02/11	5;6	0 jour
8	Elaboration du modele théorique	15 jours	Mar 01/03/11	Lun 21/03/11	7	3 jours
9	Programmation (accéléromètre + gyromètre + interface)	20 jours	Mar 22/03/11	Lun 18/04/11	8	3 jours
10	Programmation du protocole de communication I²C	5 jours	Mar 01/03/11	Lun 07/03/11	7	33 jours
11	Tests réels (capteurs + programmes + interface)	8 jours	Mar 19/04/11	Jeu 28/04/11	9;10	3 jours
12	Comparaison et optimisation	10 jours	Ven 29/04/11	Jeu 12/05/11	11	3 jours
13	Tests finaux	3 jours	Ven 13/05/11	Mar 17/05/11	12	3 jours
14	Rédaction et Relecture du rapport final	56 jours	Mar 01/03/11	Mar 17/05/11	7	0 jour
15	Impression du rapport et préparation de la soutenance	3 jours	Mer 18/05/11	Ven 20/05/11	14	0 jour
16	Soutenance	1 jour	Lun 23/05/11	Lun 23/05/11	15;13	0 jour

MÉTA PROJET MANAGEMENT



CONCLUSION

En ce moment, notre principale occupation est de comprendre précisément comment l'IMU* 3000 envoie les informations au DsPic* et nous avons besoin de développer un algorithme afin de collecter et de retraiter ces informations. Les différentes mesures seront alors fusionnées pour obtenir les meilleures estimations possibles. Nous utiliserons un filtre pour corriger chaque biais.

A l'issue de ces quelques mois de projet, l'heure d'un bilan s'impose. Ce projet avait pour but de nous familiariser avec les technologies liées à la navigation d'un micro drone. Cela a commencé avec des exercices assez basiques avec le dsPic*30F4011 comme le clignotement d'une LED ou la commande d'un bouton. Ceci afin de nous préparer à un projet de plus grande envergure à savoir la réalisation d'une centrale inertIELLE* à bas coût.

Pour ce projet il a fallu apprendre à s'organiser, trouver par nous-mêmes des solutions à des problèmes, mais il nous a surtout permis d'améliorer nos connaissances sur la manipulation d'un microcontrôleur*. Au final il en ressort un bilan plutôt positif, sans compter que la nouvelle centrale inertIELLE* fonctionnait plutôt bien.

Nous avons développé ce système embarqué pour la navigation du drone SPARROW. Toutefois, il peut aussi être utilisé dans toute sorte de projet nécessitant la connaissance de l'attitude moyennant un problème de poids et de prix. Dans la mesure où la précision n'est pas la première préoccupation, ce système embarqué peut avoir de nombreuses applications. Par exemple il est possible de l'utiliser à l'intérieur d'une voiture pour obtenir les variations d'une route, dans un avion ou dans un missile pour déterminer leur orientation etc...

Dans quelque temps il sera possible d'utiliser le type de technologie que l'on appelle « navigation aveugle » plutôt qu'un GPS*. De plus, un certain nombre de laboratoire de recherche étudie actuellement un système hybride. Cela permettra d'augmenter la fiabilité et ainsi de réduire de potentiel erreurs provenant de l'IMU* ou de du GPS*. Il est probable qu'au vu des progrès dans le domaine de la réalité virtuelle, divers jeux (tel l'AR Drone) se développent autour de cette technologie.

Finalement, nous remercions Mr Gademer, Mr Beaudoin, Mr Vittori, Melle Avanthey et les élèves qui nous ont aidés dans la réalisation de ce projet.

BILAN DE TRAVAIL

Séance du 25 janvier 2011 :

- Durée de la réunion 4h.
- Discussion et partage des différentes recherches effectuées.
- Mise en place d'une base de donnée partagé sur googledocs et dropbox.
- Mise en place d'un calendrier partagé google agenda.
- Réflexion générale sur la planification des grandes étapes du projet.
- Recherche et début de résumé de la théorie des angles d'euler.
- Recherche d'ouvrage au CDI.
- Planification d'un rdv hebdomadaire.
- Prévision des prochaines taches.
- Choix d'une date pour le rdv suiveur.

Objectifs pour la prochaine fois :

- Poser des questions concernant les composants utilisés.
- Avancer au plus vite dans le tutoriel.
- Se documenter sur I2C.
- Se renseigner pour appliquer le PERT et MDM.

Séance du 02 Février 2011 :

- Durée de la réunion 3h.
- Recherche des datasheets de l'IMU
- Lecture des datasheets
- Rédaction résumé sur la théorie d'Euler
- Plannification réunion du 03 Février à 14h

Questions pour la réunion 04 Février 2011 :

- Organisation :
 - Peut on faire 2 minis groupes (théorique, informatique) ?

Pas conseiller, rester souder. Ne pas trop s'éloigner trop dans la théorie savoir les limites pour mettre en application.

- Faut il suivre dans l'ordre le tutoriel ?

Il faut le suivre jusqu'à uart et après on saute à I²C. (uart c'est un protocole avec des trames, et I²C c'est un bus)

- Comment fonctionne le calendrier partagé ?

Dès que vous êtes ajouté dans le partage, vous avez accès à Google Agenda. Vous réservez votre créneaux. Pensez aux autres , ne réserver pas tous.

- Que faut il mettre dans le rapport Bilan Intermédiaire ?

Vérifier s'il faut rédiger un cahier des charges. Plan du rapport : Etat de l'art, état scientifique des solutions, bilan de l'avancement, perspectives futures , Bibliographie (sans site web)

- Technique :
 - Combien cartes d'acquisition avons nous accès ?

Il y a qu'une carte.

- *Pouvons nous empreinter ICD2 du labo d'élec?*

Faut voir mais comme il y a qu'une carte, ca sert pas à grand chose.

- *Faut il étalonner notre carte à 25\$ sur la carte_2000\$?*
- *Quelle est la position de la carte à 2000\$ par rapport à celle de 25\$?*
- *Quelles infos récupère t-on de la centrale inertuelle (type, unité, %erreur) ?*

Toutes ses infos sont dans la datasheet du IMU3000 . Mais en général on récupère les données à l'aide de l'I²C, on trouve le détails dans la datasheet. Pour chaque axe on a des variations angulaires et des accélérations et la température. (facteur essentiel en électronique)

On mesure des variations angulaires grâce aux gyroscopes et les accéléromètres rajoute de la précision.

L'un est "compétent" pour tout ce qui mesure statique, tandis que l'autre est plus adapté pour la dynamique.

Donc on fait une combinaison entre les deux et on détermine empiriquement et analytiquement l'influence de chacun. Toutes ses mesures permettent ainsi de définir le roulis et le tangage.

Bonus : penser à regarder la navigation magnéto

- *Pourquoi utiliser accéléromètre et pas un capteur de vitesse ?*

Il n'existe pas de capteur de ce genre.

Séance Mardi 8 Février 2011 :

- Rédaction du plan de rapport intermédiaire
- Etude des matrices de passage pour les angles d'Euler
- Exercice du tutoriel
- Répartition des recherches et de la rédaction :
 - Etude de l'IMU3000 - Matthieu
 - Protocole de transmission (I²C) - Niré
 - Etat de l'art - Yann
 - Etude du microcontrôleur dsPic30F - Pierrick

Séance Lundi 14 Février 2011 :

- Rédaction de l'I²C
- Tutoriel
- Rédaction de IMU3000

Séance Mardi 08 Mars 2011 :

- Compréhension du protocole I²C
- Rédaction du résumé "tutoriel"
- Implémentation du code I²C

Séance Lundi 14 Mars 2011 :

- Etude de la FIFO
- Contacter Mme Bouzefrane

Séance Mercredi 16 Mars 2011 :

- Contacter Mme Harfouche

Séance Jeudi 22 Mars 2011 :

- Rédaction du résumé français
- Recherche sur UART

Séance Jeudi 23 Mars 2011 :

- Rédaction du résumé anglais
- Implémentation du code UART

Séance Jeudi 31 Mars 2011 :

- Configuration de l'uart avec le pickit3
- Test du programme uart
- Rédaction de l'uart

Séance Vendredi 1 Avril 2011 :

- Rédaction du rapport anglais

Séance Mercredi 6 Avril 2011 :

- Préparation de la réunion suiveur

Séance Vendredi 8 Avril 2011 :

- Réunion Suiveur :
 - Questions :
 - Réponses :
 - Conseils :

Séance Lundi 25 Mai 2011 :

- Rédaction du deuxième draft en anglais

Séance Lundi 2 Mai 2011 :

- Conctacter constructeur

Séance Jeudi 5 Mai 2011 :

- Rédaction rapport final
- Installation de Minicom
- Manipulation Tera Term
- Test code I2C
- Plannification Réunion suiveur

RESSOURCES

BIBLIOGRAPHIE

Ouvrages :

- **Rogelio Lozano**
 - “*Objet Volants miniatures : modélisation et commande embarquée*”
 - Hermès - Lavoisier
 - ISBN : 978-2-7462-1466-8
 - EAN : 9782746214668
 - disponible à Lille1, Paris CNAM, Paris 6 DUMPC
- **Marc Grozel**
 - “*Drones, mystérieux robots volants : les yeux et le feu du XXIe siècle*”
 - Lavauzelle Graphic Editions
 - ISBN : 978-2-7025-1093-3
 - EAN : 9782702510933
 - disponible à Paris CNAM, Paris CDEM, Paris cité des sciences
- **Jean Christophe Zufferey**
 - “*Bio-inspired flying robots : experimental synthesis of autonomous indoors flyers*”
 - EPFL Press Lausanne
 - ISBN : 978-1-4200-6684-5
 - EAN : 9781420066845
 - disponible à Clermont, Evry, Grenoble, Toulouse par PEB
- **Microchip**
 - “*Documentation dsPic*30F Family Reference Manual*”
 - http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodId=2574
- **Vincent Vittori**
 - “*Tutoriel Programmation dsPic**”
 - Laboratoire ATIS

Articles scientifiques :

- **Philippe Martin, Ewan Salaün**
 - “*Design and implementation of a low-cost observer –based attitude and heading reference system*”
 - Elsevier, Control Engineering Practice
- **Jean Claude Radix**
 - “*Gyroscopes et gyromètres*mécaniques avec élément rotatif*”
 - Techniques de l'ingénieur
- **Zhi Shen, Jacques Georgy, Michael J. Korenberg, Aboelmagd Noureldin**
 - “*Low cost two dimension navigation using an augmented Kalman filter/Fast Orthogonal Search module for the integration of reduced inertial sensor system and Global Positioning System*”
 - Elsevier, Transportation Research Part C

Logiciels :

- **Environnement de développement pour microcontrôleur* PIC : MPLAB IDE**
 - Page de description
http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en535363
 - Page de téléchargement
http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en019469&part=SW007002
- **Outil de conception de carte électronique : EAGLE**
 - Page de description
<http://www.elektor.fr/nouvelles/eagle-version-5-10-0-avec-designlink.1389399.lynkx>
 - Page de téléchargement
<http://cadsoftusa.com/download.htm>

Liens vérifiés le 22/02/2011

Matériel :

- **Composants électroniques :**
 - Invensense : http://www.invensense.com/mems/gyro/IMU*3000*.html
- **Microcontrôleur*s :**
 - Microchip : <http://www.microchip.com>

WEBOGRAPHIE

- **I²C tutoriel, explications:**
<http://www.aurel32.net/>
- **Angles d'Euler animation:** <http://www.sciences.univ-nantes.fr>

CITATION

Définitions, illustrations :
<http://fr.wikipedia.org/>

Documentation technologique :
<http://www.onera.fr/>
<http://www.wilcoxon.com/>
Solutions et conseils :
<http://www.forum-auto.com/>
<http://forums.futura-sciences.com/>

GLOSSAIRE

- Centrale inertielle : appareil de navigation composé de gyroscopes et d'accéléromètres.
- IMU 3000 : Centrale inertielle composée de trois accéléromètres et de trois gyromètres.
- DsPic : Digital signal PICs = Microcontrôleur de la société Microchip.
- Microcontrôleur : Circuit intégré qui possède un micro-processeur, des mémoires, des unités périphériques et des interfaces d'entrée sortie
- GPS : Global Positioning System = Appareil de géolocalisation fournissant la latitude et la longitude d'un objet.
- Gyromètre : Capteur qui permet de mesurer la vitesse angulaire de l'objet sur lequel il est placé
- Accéléromètre : Capteur qui permet de mesurer l'accélération de l'objet sur lequel il est placé
- Modified Harvard : C'est une variante de Harvard architecture qui permet au contenu des instructions mémoire d'être accessible comme si c'étaient des données.
- Harvard architecture : Consiste à utiliser des mémoires système entièrement séparé pour stocker les données et les instructions.
- FFT : Fast Fourier Transform = Algorithme très efficace permettant d'obtenir une transformée de Fourier discrète.
- UART : Universal Asynchronous Receiver Transmitter = Protocole utilisé pour faire la liaison entre l'ordinateur et le port série
- I2C : Integer Integrated Circuit = Protocole de communication qui relie deux composants via un bus série asynchrone et synchrone

ANNEXE

ENGINEERING SCIENCES PROJECT

“Design of a low-cost inertial measurement unit”



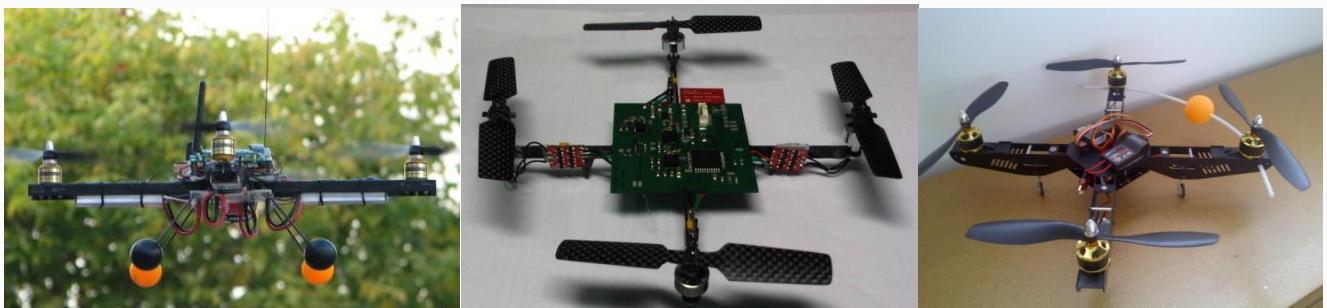
ABSTRACT

ALWAYS SMALLER AND ALWAYS CHEAPER

The department of research “Acquisition et Traitement des Images et du Signal” (ATIS) based at the French engineering school ESIEA “Ecole supérieure informatique électronique automatique” is a laboratory that works on innovative projects in the field of image, signal recording and processing.

Inside this laboratory, we are in charge of the navigation optimization concerning the micro drone SPARROW. This project is another step for ATIS aerodynamics after: Faucon noir and Moustik.

We collaborate with a PAIR group who takes care of the mechanical and electronic design. The drones main function will be to fly indoors, in a confined environment but it will still be used to collect and transmit real time data from the field.



For economic reasons the ATIS lab researcher suggested that we to reduce the main expense concerning the navigation equipment. To achieve our goal, they gave us an alternative for the spatial data acquisition. In fact the \$2000 inertial central must be replaced by a \$25 one.

In this context, we are in charge of the implementation of this new card, the optimization of the data processing and the comparison of the results coming from the new card with those coming from the old one.

KEY WORDS

UAV (unmanned aerial vehicle), Navigation, IMU* (inertial measurement unit), Embedded System, low cost

Personal contributions

MATTHIEU BOYAULT :

This scholar project means a lot for me. In fact, it is a double challenge: personal and professional.

I have chosen to study at ESIEA because of the ATIS Laboratory which M. Beaudoin presented to me as a huge opportunity to work on an exciting projects, on the cutting edge of technology in the robotics world but also to meet interesting companies which could offer me jobs.

Therefore, during the first semester I got closer to the ATIS team in order to participate in their project. Thus, I have started the initiation tutorial and learned about my future PSI project: the micro drone SPARROW. During the second semester, I have recruited some motivated mates and presented us like a prospect. As soon as, the ATIS team had selected the people to work on PSI SPARROW, I took the project manager role. Consequently, I created a database where we can save, share and edit our documents on GoogleDocs and Dropbox. Then, I stored a common calendar on Google Agenda.

At the beginning, I looked for resources on Euler's Theory, inertial measurement unit and microprocessor DsPic*30F. My first reflex was to try to obtain useful information on the Web. Of course, it was harder than I thought. Thus, I have gone to libraries in order to find scientific books. Thanks to the administrator, I have learned a methodology and tools: Google Scholar, "Techniques de l'ingénieur", Sudoc, Pascal.

Then, given that I came from preparatory school, I worked on Euler's equations. I wrote a summary which explains each step from the acquisition of data to the determination of the position. After that, Yann and I applied the management course which we have just learned in order to optimize our work. We made Chart Bars and Metra potential method with MS Project which was inserted in the first report.

In this report, I was responsible for the inertial measurement unit part, works cited and the layout. So, I have used my research, searched and read all datasheets about the IMU*3000* to sum up its electronic specifications and workings. It is a complex component capable of doing so many tasks that it was hard for me to imagine what I'll get in the end. According to our guide Vincent Vittori, the best way to understand a component or a phenomenon is by testing it.

However, I need to get to grips with the communication protocol I2C* as results of my research, summary. I have tried to implement a program using C language but I was blocked by the FIFO working. Accordingly, I have asked for help from our OS teachers who weren't able to give us any useful answers. Hence my research and my implementation on the communication protocol UART* which will enable me to see what kind of data the IMU* receives or sends.

EULER PRESENTATION

I have studied Euler's Theory. This part of the project is essential because it will enable us to convert datas from the sensors (accelerometers and gyroscopes) to useful datas. That's to say we will get: the position and the orientation of the micro drone.

EULER DETAILS

Basically, this step is based on mathematical relations and physics phenomena. In fact, with this tool I used formulas to change references. Then, I transformed them into matrix equations and multiplied by the Euler matrix. Finally, there is a last change in order to come back to scalar equations.

RECUPERATION OF INFORMATION

I have spent a lot of time reading the datasheet. The fact is the constructor wrote plenty of information about almost every part of the IMU*. However, often the information which I need is scattered in the entire document. Thus, I easily loose myself following each key word.

One other issue, which I have encountered during the study of Euler's theory, is the conversion from analogical to digital. Indeed, during the first semester we followed a course dealing with this topic. Unfortunately, I have noticed that our acquired knowledge wasn't enough. Mostly, we studied too much signal theory and there was less focus on applications. In that context, I have faced this issue by researching practical formulas.

I2C* THEORY

At the beginning, I followed the ATIS tutorial. But I lost myself in all the code lines. Consequently, I started my own research on this topic on the Internet and the library in order to understand what points are important. After a lot of reading and filtering, I found interesting things which I have summed up in a googledocs.

I2C* IMPLEMENTATION

Then I started to implement my own I2C* program. Nevertheless, it wasn't as obvious as the general concept. I received help from Yann who took care of this part.

UART* THEORY & IMPLEMENTATION

I employ the same strategy than I2C* in order to produce our own program.

From a practical point of view, I tried hard to make MPLab and TeraTerm work together. Unfortunately, I was very disappointed and I lost a lot of time. In order to correct that, I get in touch with a colleague which I knew had already done this work. After a planning meeting and spending time finding the error, we noticed that it was a computer problem. In fact, when we used my computer with Windows to launch the

program and his computer with Ubuntu to receive information, the program worked. The fact is on Ubuntu we use another software, MiniCom, which is configured differently.

After this epic, I broke my computer. Honestly, I have been victim of my hardrive degeneration. Therefore, I have lost all my personal data and I have been forced to buy a new computer which was rather complicated.

EQUIPMENT

As soon as we finished the theoretical steps, we continued testing our program. However, Yann and I had to adapt new equipment «Pickit 3». This step consists of soldering connections between the device and our card. We used pewter, soldering iron, and a multi hands support.

MANAGEMENT

Like every project, we have to report our work to our supervisors. Thus, I used to send an email every Friday where I sum up what we have done and what is planned.

Moreover, I planned two official meetings. Of course, I prepared them by writing all the questions we have and presenting documents which were already written. After that, I wrote a clean report which is accessible on our own shared folder.

Given my non-official status of leader, I tried to keep in touch as much as possible with my associates and encourage them to work weekly on Thursday.

After following Vedel's course about Management project, Yann and I applied what we had learnt. Thus, we analyzed the main steps of the PSI. Then, we ordered them and estimated the work time according to our small experience and suggestions. In definitive, we got a ChartBar and MPM which we have tried to follow the best we are be able to.

GENERAL ISSUE

The main and recurrent issue is every step of the project deals with new component, new technology, new language, new equipment, new datasheet. We continuously discover new thing consequently it is harder and harder to rely on each notion and to extract useful information. Moreover, we don't understand all the general blocks.

Finally, I was surprised by the richness of the vocabulary. It is true that most and even though all the documents were in English. While I don't have a bad level in this language, it was a real challenge to read all the time technical helps and resources in original version.

YANN BOISGONTIER :

I wanted a PSI who matched my objectives: I wanted to work on a huge project in an interesting field. The PSI "micro-drone" was the best project according to me. We were six and only four of us had the opportunity to work on the project. I was chosen by ATIS to work on this project. It was a good point for me because I really wanted to work on a PSI which gave me the lucky occasion to work with interesting people as ATIS team. Moreover, working on a research project is very exciting for me.

We can sum up this project in two main parts: understanding technologies and concrete applications.

Before using our materials, we needed to understand how it worked. Therefore, the first task I did is the state of the art. From the synthesis of web and book documents, I had to explain the running of inertial measurement unit (IMU). The IMU needs accelerators and gyroscopes to provide its values. I had to find the highest level of development of these devices. I discovered many various technologies and, more important, a method to establish a state of the art. We inserted this research in the first report.

The second task we had to do was to learn the method of programming in C language on DsPic. We use a "DsPic30F4011" microcontroller. ATIS provided our team with a tutorial called: "Tutorial Programmation DsPic" with many technical applications. Then, we had to read the document, do the exercises and go to ATIS to try our codes. Everything was new for me, but with a large amount of work, I was able to understand and to realize most of the tutorial. At the beginning, we thought that this will be easy but we lost a lot of time trying to understand all the potential of the Dspic. At first, we were not able to determine what we will need so we had to study the whole tutorial.

From the comprehension of theory, we understood what we needed to collect data in the IMU 3000 and how we will retreat the data and make them available to the "micro drone".

After these two first parts of initiation to technologies, we were ready to begin the concrete applications. We read the datasheet of the IMU 3000 (the IMU we are using) and saw that the communication from the IMU to the DsPic is following a specific protocol: I2C bus. We began to collect information about I2C. I encoded the communication protocol. This part was the most difficult because all of us had no idea about how to start the IMU 3000 and configure the device to have the good data available. So, we asked for a meeting with Mr Azouzi who helped us a lot and gave all the information that we needed. It was interesting to see how it was easy for him to understand technical datasheet information and explain them to us. We also had a meeting with ATIS who gave us some strategy about coding and testing these devices. Unfortunately, the best way to understand how works a device is to make a lot of experimentations.

Then, it became obvious that we needed to test the values that the IMU 3000 provided to the DsPic on a computer screen. Thus, we developed UART communication protocol which allows us to print DsPic values on the computer.

We now have a lot of tests to do and we need equipments from ATIS. To be more self-sufficient, ATIS gave us a PICKIT 3 communication pack. It connects Dspic to a computer. We had to solder components between device and card. There are multiple ways to connect the Dspic and the computer. At the beginning we were using ICD3 but it is very expensive (300€) and ATIS does not have a lot of them, thereby we were asked to use PICKIT 3.

We are using two softwares: Mplab and teraterm. These softwares, are good to use Dspic because Mplab allow us to debug easily Dspic programs and Tera Term give us the opportunity to use UART protocol and make some tests.

This part is obviously the most interesting because we can see the concrete achievement after all the steps. I acquired new methods of debugging and testing code. Otherwise, I understood how to use a datasheet and I learned a lot of English technical vocabulary because most of the datasheet are not written in French.

To work in a team is not easy; we need to have a good organization. Thus, we were using specific tools and we were often meeting. During the whole project we were using tools from cloud computing like googleDocs and DropBox, there are two advantages from this technology, first, you can work from everywhere: you just need a computer, second, people from the team can see your achievement easily. We wanted to meet a lot to work together. Thus, from the beginning of the year we met every Friday. We also met with ATIS team to report and ask question to them.

From Mr Vedel management course, Matthieu and me, we scheduled the main steps of the project. It is very interesting because at the end I figure that we spend a lot of time to acquire theoretical knowledge and we run out of time to do experiments.

This project is very attractive because it deals with many varied subjects from informatics and electronics to embedded systems. We discovered all the technologies used to make an object to fly. We also acquired engineering methods of testing and developing. Moreover, we found out about the teamwork with its synergy and its bias.

PIERRICK PICON :

As a member of the drone project “sparrow”, I have worked on several parts of the development of the low-cost inertial measurement unit. In fact my team and I were asked to encode an inertial measurement unit. I shall describe my personal contribution in two parts. On one hand I shall describe the organizational choices and difficulties that the team faced during the project as well as the solutions used by the team to overcome them. On the other hand I will go over the work I accomplished for the team.

. At first we decided that everyone should work on all parts of the project. We tried to meet regularly but the differences in our schedule made it really hard to accomplish. By mutual agreement, we decided that every Thursday afternoon should be saved to meet and share our progress and ideas as well as to assign the next goals for the week to come. However, we encountered several problems. For example, the training preceding the application took us more time than expected. We realized that in order to bring the project to an end and be successful we had to come up with a solution. Therefore, we were forced to split the remaining work in two. In the meantime two sub teams were formed to take care of these two parts. I teamed up with Matthieu and we both worked on the communication protocol called “UART*”. In the meantime Yann and Niré took care of the I2C* code writing.

Concerning the work on the micro drone itself my work can be broken down into three steps:

First, I had to learn the theory of Euler’s angle. This theory is the basic mathematics necessary to understand and resolve our problem. It allows me to know how to translate the raw data that we have to recover from the accelerometers and gyroscopes into usable data. I also researched the uses and capacity of the different components of our system. These theoretical tasks were fastidious because they meant reading a lot of technical articles and datasheets and verifying every piece of information we found especially when I found them on the internet. Then I had to explain the use of the microcontroller DsPic*30F4011, its capacity and its possible different applications. The DsPic*30F4011 has many applications and it was really hard to find every pieces of information and sum them up for the other because they were scattered all over the datasheets. However, the fact that most documentation was in English made our job harder but nevertheless interesting because it pushed us to learn new technical vocabulary. Then, I started to work on the tutorial called ” Tutoriel Programmation DsPic* ”, that was provided by ATIS, to learn how to program a microcontroller in C languages. This tutorial presented me with the diverse protocol that we have to use to go through with our project via a series of practical exercises. It accompanied me through my first step of I2C* protocol as well as UART* and the uses of the different software required to implement the IMU*3000* and the microcontroller. I did a small summary on the I2C* protocol which is used to establish the link between the DsPic*30F4011 and the IMU*3000* and I attended the creation of the summary on the IMU*3000*.

Secondly, we started the concrete work. In fact, after learning the different protocols used to communicate between the components we started to program them. Until now it has been the hardest

task because I had to learn how to code by myself and I wasn't really confident about how I should proceed.

Thirdly, we began to test the program we created to check if they were working. We used two software and two pieces of equipment in order to do that. The first software is MpLab. This software is employed to implement the code in language C. The second one is TERA TERM which purpose is to check on the computer the transmitted data between the IMU* and DsPic*. The first equipment we used to link the computer with the electronic card was the ICD3 but it is really expensive and therefore ATIS owns only one. After several weeks, they found an alternative. The Pickit3 now allows us to transfer program via USB to the Pickit after implementing them on MPLAB. However, the greatest advantage is its cost (20 euros rather than 300 euros). This step is probably the most important and the most interesting of all. Foremost because it is the result of our project, then because it is known that we acknowledge our mistakes and take measures to patch our code. Moreover, it gives us an idea about where we stand and about the work that needs to be done to finish it and succeed in our project. This stage is still in motion and will continue until the end of the semester. Following the verification of our code we will have to test it directly with the drone to see if it works and makes the adjustment necessary because there always is a variance between the theory and the reality.

.

Finally, this multidisciplinary project is very interesting. It allowed us to discover new technologies and requires us to learn and to surpass ourselves. It has shown us the different steps involved in the development of a project. It also requires us to work as a team and deal with the pros and cons that this involves. I believe this project is a significant personal enrichment for everyone involved.

NIRE ESTALL

For the realization of the project, the task which was entrusted to me is the configuration of the protocol UART* (Universal Asynchronous Receiver Transmitter). It allows the connection between a computer and a microcontroller that which we use is the DSPIC*30F4011.

The training of this protocol was not very difficult for me, because at the time of my years studies the preceding one, I was brought to carry out a project which precisely used the UART* connection. But its installation, the realization of the program which makes it possible to manage this communication was a little more complicated, because it was necessary to study all the functions of the microcontroller.

In this project which is to use an inertial platform at low cost, we use UART* for the tests on computer, before making steal the drone in external medium.

Technologies and materials

PROGRAMMATION DsPIC*

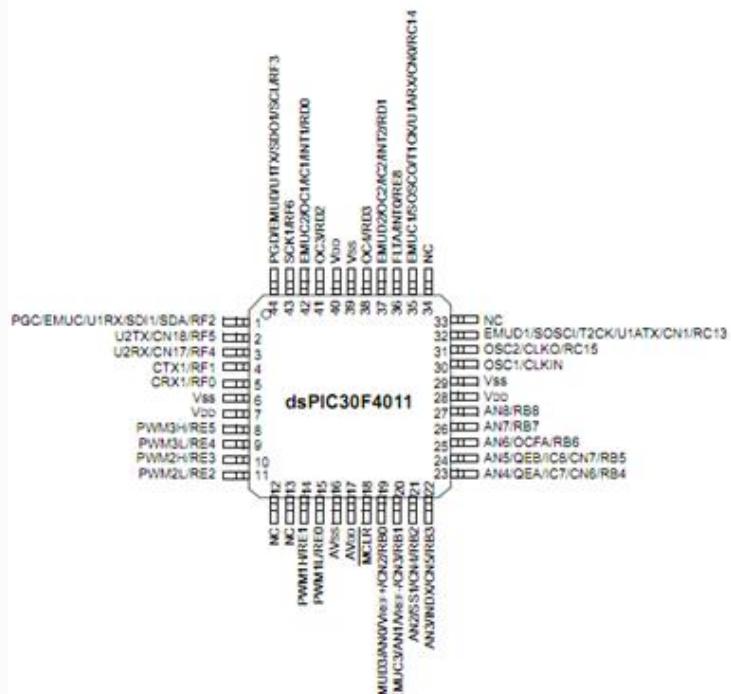
In this project we are working with a DsPic^{*}30F4011. It's a microcontroller which belongs to the DsPic^{*} family.

It is used to control engines, convert and process data. It is based on a Modified Harvard* architecture.

Specifications:

On this microchip, it is possible to find several devices:

- **UART***: (Universal Asynchronous Receiver/Transmitter) it's a serial that allow us to interface the card with a computer.
 - **I²C**: digital link that is based on bus principle and is used to communicate between electronic components.
 - **Output Compare**: microcontroller mode used to control servos engines.
 - **Timer**: The DsPic^{*}30F4011 has 5 timers. Those timers are part of the microcontroller.
 - **ADC**: Analog Digital Conversion. It's a converter employed for sensors such as gyroscopes.
 - There are also 3 types of memory on the chip. One is a flash memory of 24KB, another is an EEPROM memory of 1KB and the last one is a program memory of 48KB.
 - 30 pins I/O which are GPIO (General Purpose Input/out) In order to work the DsPic^{*}30F4011 must be powered with a voltage between 2.5 and 5V.



Implementation:

There are three different steps to program it:

First it's essential to import the library and define the shortcuts to a line or an operation.

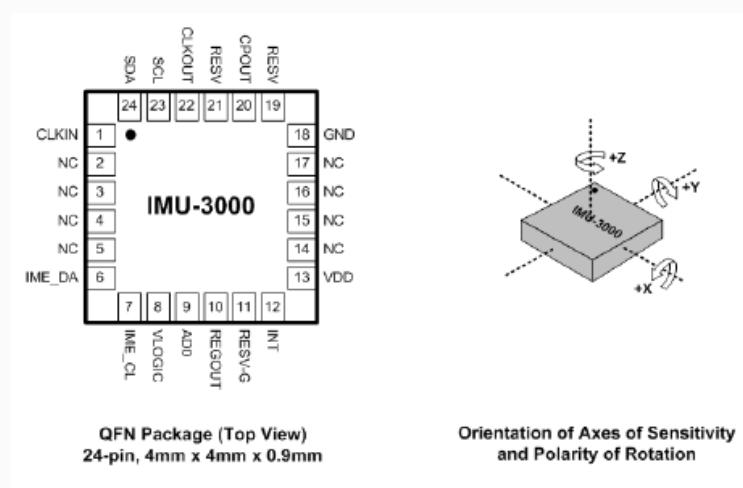
Then after getting in the main function, the pin must be defined either they are input or output.

The last step is the code that defines the purpose of the program.

In the drone the microcontroller must retrieve the data from the gyroscope and accelerometer, stock them in the IMU*3000* in order to process the data and make them usable.

DISCOVERING IMU* CHIP

Specifications :



The IMU* 3000 components are :

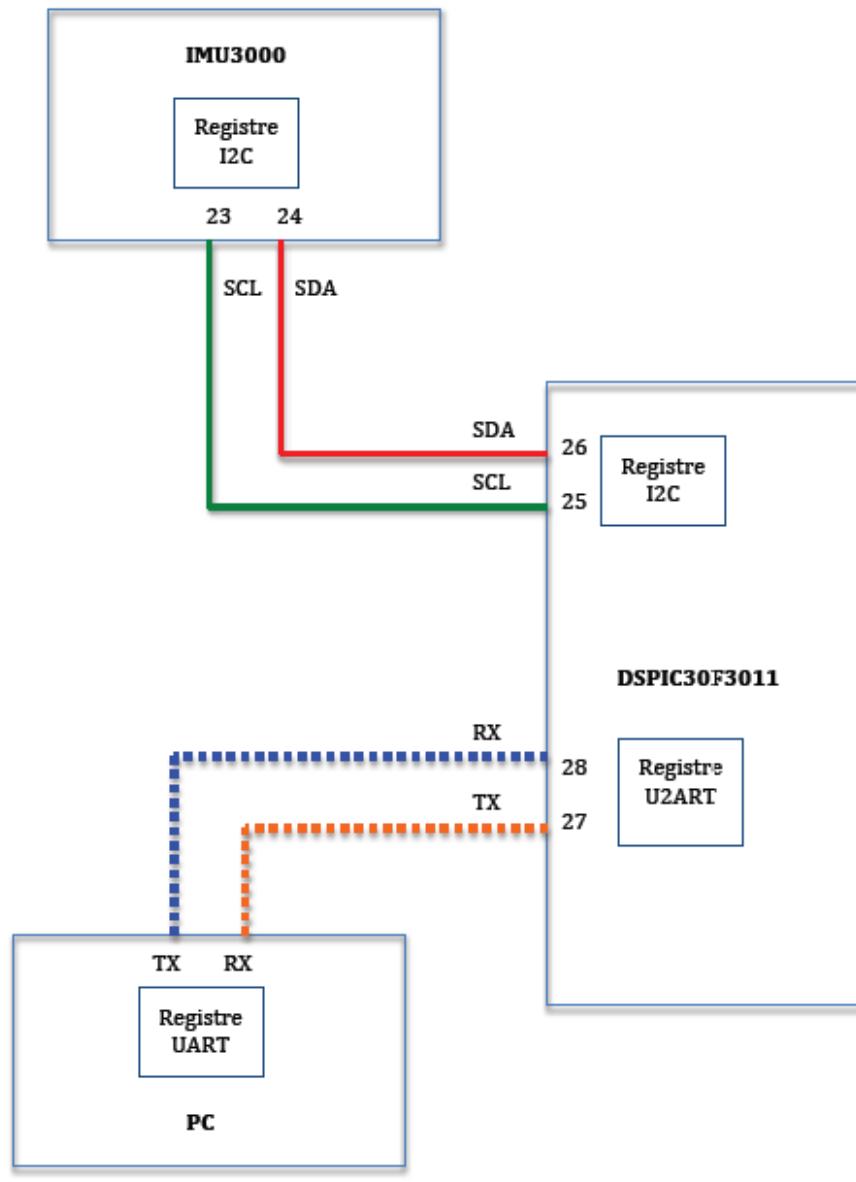
- 3-axis digital gyroscopes, measuring range: ± 250 to ± 2000 $^{\circ}$ / sec
- 3 ADC converters
- 3 Low pass filters
- DMP (processor digital motion) is used to manage time and processing power.*
- FIFO (first in first Out of 512 bytes) determines the data to be accessed by the IMU*.
- Communication I²C Master Mode (2 bytes per axis) and / or Slave:
 - "I²C Primary" communication between the interface
 - "I²C Secondary" for communication with the 3 accelerometers
- VLOGIC power: 3.3 Volt DC or 3Volt with controller
 - technology used MEMS (Micro Electro-Mechanical Systems) 24 pine
- Size: 4 x 4 x 0.9mm
- Frequency 1MHz clock (synchronization)

The IMU* communicates with essential components:

- Kionix KXTF9: 3 accelerometers, one for each axis
- AKM AK8975: 3 compasses, one for each axis

COMMUNICATION PROTOCOL

CABLAGE DES LIAISONS DE COMMUNICATION



I²C*

I²C (inter integrated circuit) is a protocol based on the bus system. A bus is a standard connection between two components. The I²C bus is a serial bus used to replace the parallel bus in order to save place when the debit of a serial bus is sufficient. Three wires are necessary to establish the I²C connection. The first one matches to a data signal (SDA), the second one to a synchronal clock signal and the last one to an electrical reference signal (GND).

The I²C* protocol can be broken down into eight steps:

- The master sends the starting condition and control the clock signal
- The master sends a 7-bit single address matching to the slave identification.

- The master reports whether the mode is reading or writing (R/W) which implies that the slave is either in reception mode (0) or in emission mode (1).
- The receptor sends a bit of acknowledgement.
- The transmitter sends a byte of data
- The receptor emits acknowledge bit for the retrieved octet
- Sends and reception of the data
 - a) The master sends the stop writing condition after the last octet of data
 - b) The master sends the not-acknowledge bit then sends the stop writing condition after the last octet of data

UART*(Universal Asynchronous Receiver Transmitter)

UART* is the protocol used to make the connections between the computer and the serial port. The computer sends data in parallel. We must therefore transform those data to pass them through a serial link that uses a single thread to run all the data bits

The UART* frame consists of the following bits:

- Start bit always 0: used to synchronize the receiver
- Data: the size may vary (usually between 5 and 9 bit)
- Possibly a bit odd or even parity
- And a stop bit always 1 (the duration can vary between 1, 1.5 and 2 bit time)
- The logic level of rest is 1.

Issues & strategies

LOGISTICAL

SHARING EQUIPMENTS

This year, many projects are in cooperation with ATIS, this is an issue because we share little equipment with many people. Therefore, we had a meeting and we adopted a sharing method: all the groups have some days where they are able to use the ATIS equipment. We have chosen every Thursday given that we don't have lessons during the afternoon.

TEAM ORGANIZATION

At the beginning of the PSI, we were six for four people on the project. We had a meeting with the ATIS team and four people were selected on the base of technical knowledge and the studies we had done before.

Working as a team can be very efficient with a good team organization. We are using a cloud-computing method: we work from different locations and share our results on a Google document collection. We also use Skype and mail box to be up to date with everyone's work.

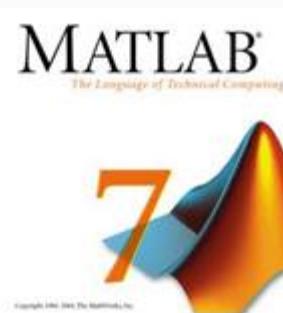
The team works quite well, but we had to balance the work between people. We are already working a lot but we are aware that we need to work more if we want to complete the project.

TECHNICAL

GRIPPING

In this project, we have to acquire knowledge on four essential softwares used in various fields: IT, electronics, and mathematics:

- MPLAB is our IDE where we implement code in C language.
- EAGLE is a program for the conception of electronic card.
- TERA TERM is used to check on the computer the transmitted data between the IMU* and DsPic*.
- MATLAB enables to calculate our position by using IMU* data and Euler's equations with matrix.



In the

meantime, we have gripped with 2 important equipments which link the computer to the electronic card. We started with a common Microchip tool the ICD3. The issue is that this component is very expensive and consequently ATIS has only one. After several weeks, they found an alternative: the Pickit3. Now we can implement our code on MPLAB as well, transfer program via USB to the Pickit. However, the most interesting advantage is its cost: 20 euros rather than 300 euros.



Current and possible future applications

Currently, we have to understand precisely how the IMU* 3000 sends information to the dsPic* and we need to develop an algorithm in order to collect and store this information. The various measurements will be “merged” to give the most precise estimations. We will use a filter to correct the different bias.

We developed this embedded system to SPARROW navigation. Nevertheless, it can be also used in all kind of projects where you need to know the attitude when cost and weight are a problem. If the precision is not the main point, you can use this embedded system for a large sample group of utilization.

For example, you can use it in a car to know the variations of the road, in a plane or in a missile.

In the future, it would be possible to use this kind of technology which is called blind navigation rather than a global positioning system. Moreover, some research labs are currently studying a mixed system. Thus, they would increase the probability of control and therefore reduce the possible errors from either IMU* or GPS*.

We could imagine that there would be applications for private individual. In fact, there is a big jump in robotics and virtual reality. In few years, some games could be equipped with this kind of technology.

Works Cited

BOOKS:

- Rogelio Lozano
 - § "Objet Volants miniatures : modélisation et commande embarquée"
 - § Hermès - Lavoisier
 - § ISBN : 978-2-7462-1466-8
 - § EAN : 9782746214668
 - § Disponible Lille1, Paris CNAM, Paris 6 DUMPC
- Marc Grozel
 - § "Drones, mystérieux robots volants : les yeux et le feu du XXIe siècle"
 - § Lavauzelle Graphic Editions
 - § ISBN : 978-2-7025-1093-3
 - § EAN : 9782702510933
 - § Disponible Paris CNAM, Paris CDEM, Paris cité des sciences
- Jean Christophe Zufferey
 - § "Bio-inspired flying robots : experimental synthesis of autonomous indoors flyers"
 - § EPFL Press Lausanne
 - § ISBN : 978-1-4200-6684-5
 - § EAN : 9781420066845
 - § Disponible Clermont, Evry, Grenoble, Toulouse par PEB
- Microchip
 - § "Documentation DsPic*30F Family Reference Manual"
 - § http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2574
- Vincent Vittori
 - § "Tutoriel Programmation DsPic* "
 - § Laboratoire ATIS

SCIENTIFIC ARTICLES

- Philippe Martin, Ewan Salaün
 - § "Design and implementation of a low-cost observer –based attitude and heading reference system"
 - § Elsevier, Control Engineering Practice
-

Jean Claude Radix
 § "Gyrosopes et gyromètres*mécaniques avec élément rotatif"

§ Techniques de l'ingénieur

- Zhi Shen, Jacques Georgy, Michael J. Korenberg, Aboelmagd Noureldin

§ "Low cost two dimension navigation using an augmented Kalman filter/Fast Orthogonal Search module for the integration of reduced inertial sensor system and Global Positioning System"

§ Elsevier, Transportation Research Part C

SOFTWARE :

- Environnement de développement pour microcontrôleur* PIC : MPLAB IDE

§ Descriptive page

http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodId=1406&dDocName=en535363

§ Download page

http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodId=1406&dDocName=en019469&part=S_W007002

- Conception electronic card : EAGLE

§ Descriptive page

<http://www.elektor.fr/nouvelles/eagle-version-5-10-0-avec-designlink.1389399.lynkx>

§ Download page

<http://cadsoftusa.com/download.htm>

Links checked 22/02/2011

EQUIPMENT :

- Electronics components :

§ Invensense : http://www.invensense.com/mems/gyro/IMU*3000.html

- Microcontrollers :

§ Microchip : <http://www.microchip.com>

WEBOGRAPHY:

- **I²C tutorial :** <http://www.aurel32.net/elec/i2c.php>

- **Animation Euler's theory:** <http://www.sciences.univ-nantes.fr/physique/perso/gtulloue/Meca/Cinematique/euler1.html>