





Table des matières



01	Rappel du projet et des objectifs	1
02	GPS RTK	2
	• Filtrage par moyenne	2
	• Survey-In	3
	• Résultats	4
	• Points GPS à l'ICAM	7
03	Amélioration du système de pilotage	8
	• Etude du DGPS avec les GPS M8N	9
	• Nouveau algorithme de pilotage	10
	◦ Fonctionnement	10
	◦ Organisation des tests	11
	◦ Récupération des données	11
	◦ Premier aperçu de l'application	13
04	Problèmes rencontrés	14
	• Mise à jour des pins	14
	• Problème d'alimentation par le BEC	14
	• Nouveaux ESC	16
	• Changement de télécommande	17
05	Conclusion	18

01 - Rappel du projet & des objectifs

DESCRIPTION DU PROJET

Le projet **RoBoNav** a pour objectif de développer un système innovant de bouées autonomes, capables de se positionner avec précision sur un plan d'eau grâce à la technologie GPS.

NOS OBJECTIFS

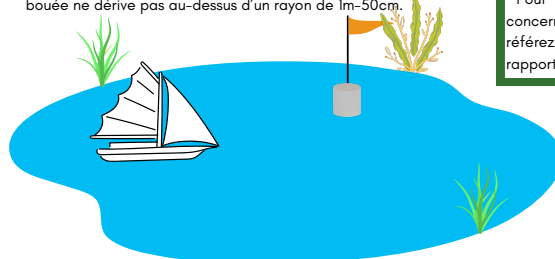
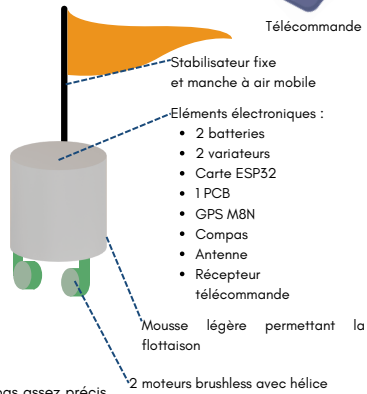
Actuellement, la bouée est radio-commandable, est capable de passer en mode ancrage virtuel (asservir sa position) et de revenir au bord du plan d'eau (en mode Return To Home).

Cependant, son asservissement n'est pas assez précis et la bouée a tendance à trop dériver en mode ancrage.

Notre objectif est donc d'améliorer l'ancrage virtuel en améliorant le GPS ou par d'autres moyens afin que la bouée ne dérive pas au-dessus d'un rayon de 1m-50cm.



Télécommande



Pour plus de détails concernant le projet, référez-vous au rapport de la phase 1

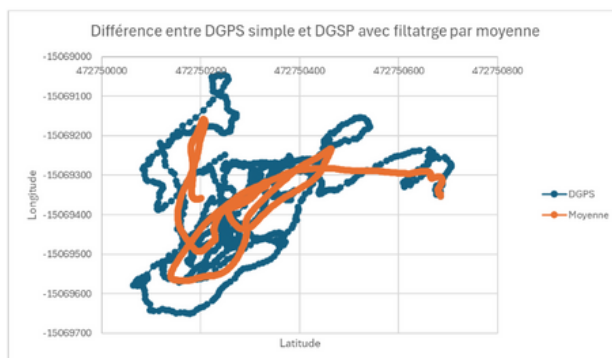
02 - GPS RTK

Rappel : les GPS RTK sont des GPS beaucoup plus précis pouvant indiquer une position exacte au centimètre près. L'ICAM a investi dans deux modules RTK de Quectel (24 €) ainsi que dans deux antennes (28 €). Lors de la phase 1, nous avons appris à configurer les modules (un en mode base et un second en mode rover). Cependant, la précision n'était pas encore celle souhaitée. Dans cette partie, nous allons vous détailler les améliorations que nous avons effectuées.

1. Filtrage par moyenne

Un filtrage par moyenne (moyenne glissante) permet de réduire le bruit aléatoire. En effet, les signaux GPS peuvent contenir des petites erreurs dues à l'environnement (réflexions sur les bâtiments, interférences, variations atmosphériques...). Le filtrage par moyenne permet de lisser ces variations rapides et de stabiliser la position calculée.

Ci-dessous, vous pouvez observer les résultats d'une expérience avec 100 points dans la moyenne avec le GPS M8N :



Observations : nous voyons bien que la moyenne concentre les points.

Pour centrer encore plus les points, nous pourrions augmenter les points dans la moyenne. Seulement, ici, nous étions statiques. Si la bouée se met à bouger, avec la moyenne de plus de 100, nous observerons le déplacement seulement au bout d'un certain temps.

Aujourd'hui, sur la bouée, la moyenne est faite à partir d'une soixantaine de points. Nous avons souhaité faire plusieurs tests avec les bouées, en la simulant sur l'eau, avec plusieurs tailles de moyenne, afin de voir avec quelle quantité nous obtenons les meilleurs résultats. Seulement, nous n'avons pas pu les faire à cause de problèmes techniques détaillés à la suite de ce rapport.

2. Survey-In

Le Survey-In en RTK (Real-Time Kinematic) est une phase d'initialisation utilisée par une station de base GNSS pour déterminer avec précision sa position fixe. Cette phase est importante pour être sûre que la station de base connaît sa position exacte pour pouvoir envoyer des corrections différentielles aux rovers (bouée). Si cette position est incorrecte, toutes les corrections seront biaisées.

Comment ça marche ?

- Pendant le Survey-In, la base moyenne des positions GPS brutes sur une durée définie (ex : 1 heure) et un seuil de précision (ex : écart-type < 1 m).
- À la fin, si les critères sont remplis, la station enregistre cette position comme "fixe", ce qui permet ensuite un fonctionnement RTK correct.

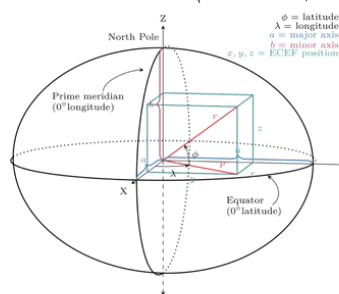
`$PQTMCFGSVIN,W,1,3600,1,0,0,0*16` : cette commande à envoyer à RTK permet de lancer un Survey-In de 1h (3600s) avec un seuil de précision de 1m.

`$PQTMCFGMSGRATE,W,PQTMSTATUS,1,1*58` : cette commande permet d'autoriser la réception et donc, d'afficher les trames dédiées aux statuts du Survey-in. A activer avant de lancer le Survey-In

Output :

`$PQTMCFGSVIN,W,2,0,0,X,Y,Z` : A chaque nouvelle valeur, une trame comme celle-ci s'affiche avec des valeurs en X, Y et Z. Ces valeurs sont des coordonnées ECEF (earth-centered, earth-fixed).

Ces coordonnées sont les informations que les satellites envoient aux GPS. Il est possible de les convertir en latitude et en longitude sur ce site par exemple : <https://convertecef.com/>

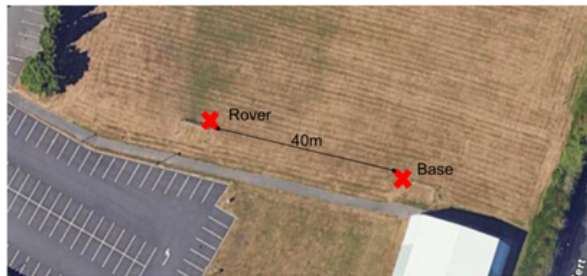


3. Résultats

Avec ces deux configurations supplémentaires (Survey-in pour la base et un filtrage pour le rover), nous avons pu faire un test. Cette fois-ci, nous nous sommes rapproché.e.s des conditions d'utilisation de la bouée sur le plan d'eau en augmentant la distance de la base et de la bouée (ici 10m). Pour cela, nous sommes allés sur le terrain de foot de l'ICAM. La base est alimentée sur le secteur du gymnase grâce à une rallonge. Le rover, quand à lui, est alimenté par un PC afin de récupérer les données.

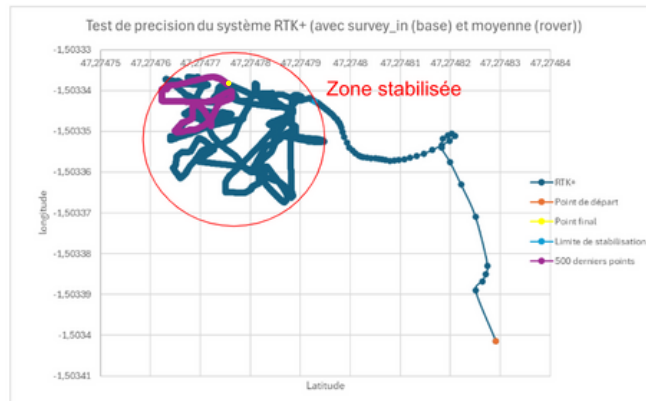


BASE



04

Ci-dessous, les résultats de ce test. A noter qu'ici, nous observons l'évolution des points du rover (statique) dans le temps. La base fait un Survey-In tout au long de l'expérience tout en envoyant ses trames au rover. En amélioration, il faudrait commencer par un Survey-In de la base, puis lancer le système RTK entre la base et le rover.



A partir de la limite de stabilisation			500 derniers points			(Soit pendant 500 s)	
	Latitude	Longitude		Latitude	Longitude		
Min	47,27476229	-1,503367401	Min	47,27476229	-1,503350281		
Max	47,27479476	-1,503336578	Max	47,27477629	-1,503336594		
Moyenne	47,27477716	-1,503348968	Moyenne	47,27476935	-1,50334364		
Delta	3,24707E-05	3,06227E-05	Delta	1,39974E-05	1,36872E-05		
Dispersion (en m)	3,61	2,33	Dispersion (en m)	1,56	1,03		

En effet, nous voyons bien à quel point les points se stabilise plus le Survey-In est long. Sur les 8 dernières minutes (500 s), la dispersion se fait dans un rayon d'environ 1m. Cela est encore bizarrement élevé sachant que c'est avec un système RTK. Ce modèle est dispose d'un grand nombre de configurations que nous n'avons pas encore toutes testées.

03 - Modèles de GPS

Dans la suite du projet, nous n'allons plus travailler sur ce système RTK puisqu'il est trop cher pour le projet RoboNav (soit 52€ pour 1 bouée comparés à 14€ pour le GPS M8N. Pour 8 bouées et 1 base, cela reviendrait à plus de 450€). Cependant, vis à vis de l'ICAM, cela reste un projet de recherche intéressant. Il est possible de développer une base fixe, positionnée sur le toit qui permettrait à l'ICAM de connaître des points du site précisément (et de rejoindre le réseau Centipède).

Voici une matrice de décision qui montre pourquoi nous avons choisi le M8N pour ce projet lowtech Robonav :

Critère / GPS	Neo-M8N (U-blox)	Quectel LC29HDA (RTK)	ZED-F9P (U-blox RTK)	NEO-M9N (U-blox)
Prix (en France)	~10-15 €	~24 €	~150-180 €	~30-40 €
Précision (sans correction)	~2.5 m	~2.5 m	~1 m	~1.5 m
Précision (avec DGPS / RTK)	(~1-2 m max)	~10-30 cm	~1 cm (RTK Fix)	(pas conçu pour RTK)
Support DGPS / RTK	(pas RTK natif)	RTCM, base/rover	RTCM + multiband GNSS	(pas compatible RTK)
Facilité d'intégration	(très documenté)	(configuration sensible)	(complexe, nécessite support)	(bon support Arduino)
Compatibilité Arduino / UART	Oui	Oui	Oui	Oui
Utilisation idéale	Tests de base GPS	DGPS expérimental / RTK	Projets de navigation haute précision	Navigation standard améliorée

D'autres GPS permettraient d'avoir de meilleures positions, mais ce qui fait bien pencher la balance ce sont les prix. Des GPS plus performant que le M8N ne rentrent pas dans l'enveloppe du développement amateur pour améliorer les courses de petits voiliers, mais dans un projet professionnel.

06

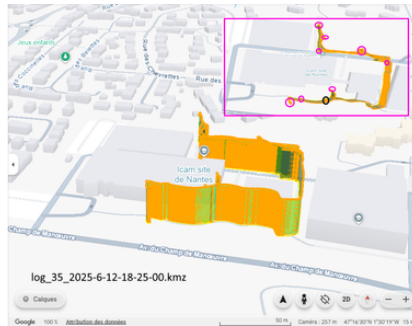
04 - Acquisition de points GPS RTK

4. Points GPS à l'ICAM

Lors de la première phase de notre projet, nous avons beaucoup analysé les GPS et les DGPS que nous avions conçus. Pour cela, nous avions besoin de comparer nos résultats avec la réalité (les vraies coordonnées). Ainsi, nous prenions soit des points sur Google Maps soit avec une application sur nos téléphones avec leur GPS interne. Cependant, rien ne nous indiquait que ces points étaient corrects.

Globalement, ce qui nous a beaucoup manqué, c'est de ne pas avoir de point de référence au l'ICAM.

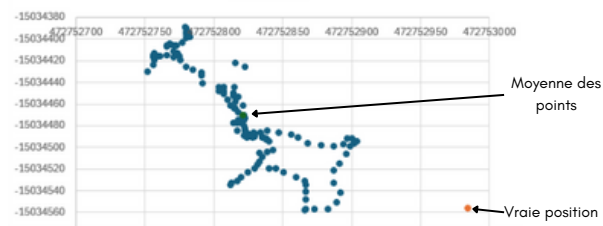
C'est pourquoi, nous avons fait appel à un ami de M. Jean FRUITET : M. Ronan BLANCHARD, qui a lui aussi développé une bouée à ancrage virtuel et à déplacement autonome avec un système RTK. Il est venu à l'ICAM et ainsi, avons pu utiliser son système RTK pour récupérer des points GPS précis du site de l'ICAM.



05 - Performances insuffisantes des GPS Ublox M8N

Dans ce projet, notre objectif principal avec les modules GPS est de permettre à la bouée de connaître sa position en temps réel. Cette information est essentielle pour une fonction clé : pour qu'elle puisse corriger automatiquement sa trajectoire ou maintenir une position cible face aux courants et au vent. Nous avons donc cherché à améliorer la précision des données GPS afin de rendre ces fonctions plus fiables et avons développé un DGPS lors de la phase 1. Nous avons poursuivi nos tests en phase 2.

Afin d'évaluer l'efficacité de notre système et la précision des données GPS corrigées (DGPS), nous avons réalisé plusieurs tests. Voici le résultat d'un de ces tests (le plus précis que l'on ait réussi à obtenir), qui illustre le comportement des positions relevées – à la fois brutes et corrigées – par rapport à la position réelle.



Cela montre que le GPS varie d'environ **1,7 m à 1,9 m** autour d'un point fixe (dispersion).

Différence entre position moyenne et vraie position :

- Lat : 472752821.4 (moyenne) vs 472752984 (vraie)
- → Erreur en m : 1.810 m
- Long : -15034469.94 (moyenne) vs -15034556 (vraie)
- → Erreur en m : 0.958 m

Donc, même si le GPS reste relativement stable (écart **~1.8 m**), sa position moyenne s'éloigne de la vraie de **1 à 1.8 mètres** selon l'axe.

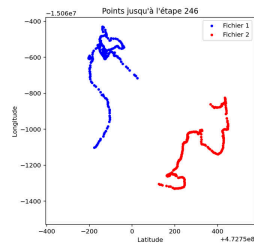
Nous avons étudié la variation des positions relevées par les modules GPS afin d'analyser leur stabilité et leur dispersion autour de la position réelle.

1. Etude du DGPS avec les GSP M8N

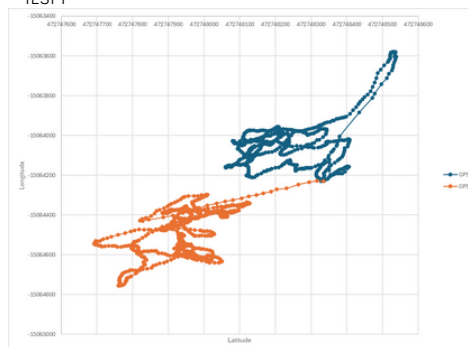
Nous avons voulu comprendre pourquoi nous n'avions pas des résultats meilleurs avec les DGSP (voir phase 1). Pour cela, nous avons décidé d'analyser les deux GPS en simultané. En effet, si les GPS varient de manières indépendante un algorithme de DGPS basé sur les mesures acquises par une base fixe et transmise aux rovers pour appliquer les mêmes corrections ne peut fonctionner.

Nous avons fait tourner 2 GPS en même temps pendant 10 minutes sur le toit de l'ICAM afin d'avoir un minimum de perturbation.

En voici les résultats :



TEST 1



TEST 2

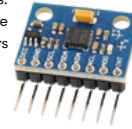
Ces observations nous ont permis de constater que les modules GPS M8N présentent des écarts aléatoires significatifs et ne dévient pas de manière homogène, ce qui limite leur efficacité pour une correction différentielle précise.

2. Nouvelles pistes

Ainsi, nous avons réalisé qu'un système DGPS avec le GPS M8N n'a aucun sens. Nous avons démontré qu'avec ce type de système (bas de gamme) il n'est pas possible d'améliorer les valeurs de position du GPS. Il faudrait passer par d'autres systèmes plus performant (type RTK) mais en ce cas nous dépasserions le budget convenu pour une bouée.

Nous avons dû rebondir et nous demander ce que nous pouvions faire pour améliorer l'ancrage de la bouée.

Nous avons pensé à utiliser un accéléromètre, qui est utilisé pour les drones. Nous avons allumé un accéléromètre et l'avons fait bouger en simulant le déplacement de la bouée. Une bouée dérive très lentement et les valeurs affichées étaient souvent 0m/s^2 ou bien très légèrement a-dessus.

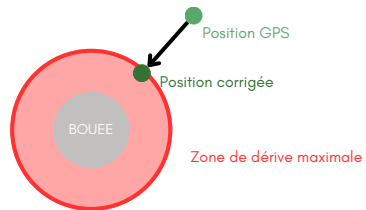


06 - Mesures de dérive et de consommation

En gardant la logique de la dérive, nous nous sommes fait la remarque que la bouée à une vitesse de dérive maximale. En analysant les données GPS que le M8N fournit en restant statique, nous avons remarqué qu'il est possible d'avoir un écart de plus de 25cm entre deux points (en 1 seconde). Or, nous doutons que la bouée dérive aussi vite. C'est pourquoi nous avons pensé à un nouvel algorithme pour filtrer les positions GPS.

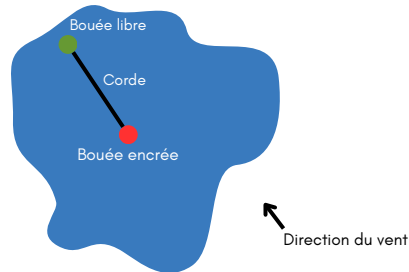
A. Fonctionnement de l'algorithme

Grâce à une multitude de tests (détaillés à la suite) sur le plan d'eau, nous allons déterminer la dérive maximale (en m/s) que peut avoir la bouée. Si la position GPS sort de la zone maximale alors, nous rectifions la position comme si elle était à la limite de la zone tout en gardant sa direction. Voici, un schéma plus compréhensif :



B. Organisation des tests

Pour tester la vitesse de dérive, nous allons laisser la bouée RoBoNav dériver sous le vent sur une distance connue. Pour cela, on dispose sur le plan d'eau de repères visuels matérialisant cette distance. Une première bouée ancrée physiquement au fond du lac indique la position de départ ; une seconde bouée, dite "libre" car non ancrée, est reliée avec un fil de la longueur voulue (20 mètres) à la première bouée ; cette bouée se place naturellement sous le vent de la bouée ancrée.



Une fois ce système installé, nous positionnons la bouée RoBoNav au niveau de la bouée ancrée et nous la laisserons dériver jusqu'à la bouée libre. Nous mesurons ainsi une distance et un temps, nous donnant une vitesse de dérive pour une force de vent donnée.

Nous profiterons de cette installation pour faire également des tests de consommation des batteries. Cette fois-ci, le test consistera à faire tourner les moteurs à une certaine puissance face et dos au vent sur la même distance (entre les bouées).

Le but est de faire beaucoup de tests sur différentes journées avec différentes conditions de vent afin d'avoir le plus possible de données et de les traiter par la suite.

C. Récupération des données

En parlant des données voici comment nous avons pensé à les récupérer et à les stocker : nous connecterons la bouée en wifi à une application PC qui récupérera les données envoyées (la tension et l'intensité des batteries, la position GPS et la commande des moteurs).

D. Récupération des données

Une fois les informations reçues, l'application les utilisera en transformant la tension et l'intensité en puissance et les stockera sous forme de fichier JSON.

Qu'est-ce qu'un fichier JSON ?

Le fichier JSON (JavaScript Object Notation) est un format de fichier texte léger permettant de représenter des données structurées sous forme de paires clé-valeur. Il est largement utilisé en informatique pour :

- échanger des données entre un serveur et une application web (notamment via des API),
- stocker des données de configuration,
- enregistrer des résultats sous une forme lisible à la fois par l'humain et par la machine.

Voici un exemple de ce à quoi peut ressembler un fichier JSON :

```
{
  "Lieu": "Le Plessis, 44980, Ste Luce / Loire",
  "date": "aaaa-mm-jj",
  "Test": [
    {
      "ID": "1234",
      "cmoteur": 50,
      "vvent": 3.0,
      "dvent": 125.0,
      "d": 20,
      "frequence": 0.1,
      "mode": "consommation",
      "duree": 600,
      "Iterations": [
        {
          "#": 1,
          "T": <timestamp>,
          "COG": 124,
          "U": 11.5,
          "I": 2000.5,
          "CMg": 45,
          "CMd": 55,
          "lon": -1.4747530,
          "lat": 47.2442500,
          "alt": 3.0000000
        },
        {
          "#": 2,
          "T": <timestamp>,
          "COG": 126,
          "U": 11.4,
          "I": 2000.0,
          "CMg": 48,
          "CMd": 52,
          "lon": -1.4747530,
          "lat": 47.2442550,
          "alt": 3.0000000
        }
      ]
    }
  ]
}
```

Pour l'image des connaisseurs, le fichier JSON se rapproche des dictionnaires en python.

07 - Une application de collecte des données de dérive

D. Premier aperçu de l'application

Voici une présentation d'une première version de l'application codée en python et des améliorations qu'elle peut subir.

The screenshot shows the 'RoBoNav - Expérimentation' application window. It features an input section on the left, a command section in the middle, and an output/console section on the right. Annotations with arrows point to specific elements:

- Zone d'entrée des input**: Points to the input fields for Date, Lieu, N°test, Vitesse, Direction, Distance à parcourir, and Commande moyeur.
- Le bouton validé**: Points to the 'Valider' button.
- Le bouton validé permet d'indiquer avec un code couleur si les données sont bien remplies**: Points to the 'Valider' button.
- Le bouton validé permet d'indiquer avec un code couleur si les données sont bien remplies. Si c'est le cas, le mode et la commande moteur est envoyé à la bouée. Le bouton START est ainsi utilisable.**: Points to the 'Valider' button.
- Les boutons START et END permettent de démarrer et d'arrêter l'enregistrement des données.**: Points to the 'START' and 'END' buttons.
- Zone d'entrée des input afin d'initialiser le fichier JSON et d'envoyer le mode de test souhaité (Consommation ou Dérivation) et la commande moteur dans le cas d'un test de consommation.**: Points to the input fields for Date, Lieu, N°test, Vitesse, Direction, Distance à parcourir, and Commande moyeur.
- Zone des output. Les nouvelles données reçues sont affichées ici.**: Points to the 'OUTPUT' section.
- Console qui affiche les données reçues et les commande effectuées. Il serait judicieux de faire deux consoles différentes : une pour les données et l'autre pour les commandes.**: Points to the 'CONSOLE' section.

The 'OUTPUT' section displays the following data:

```

Temps: 000 ms
U: 00 V I: 00 A
E: 00 J P: 00 W

Position GPS:
Lat: 00.00000000 Lon: 00.00000000
Cap: 00 °

Temps total:
CMg: 0000 CMd: 0000
  
```

The 'CONSOLE' section displays the following log:

```

24, 'direction_vent': 65, 'distance': 10, 'mode':
'Consommation', 'durée': 0, 'Itérations': [{}]}
message envoyé : Consommation
cm envoyée
message envoyé : 50
START available
('Lieu': 'ICAM', 'Date': '22/06/2025', 'Test':
('ID': 100, 'commande_moteur': 50, 'vitesse_vent':
24, 'direction_vent': 65, 'distance': 10, 'Mode':
'Consommation', 'durée': 0, 'Itérations': [{}]}
message envoyé : Consommation
cm envoyée
message envoyé : 50
START available
  
```

08 - Problèmes rencontrés

Comme dans tout projet de développement de système embarqué, plusieurs obstacles techniques ont émergé au cours de notre travail. C'est des événements qui arrivent, surtout quand on développe un système nouveau en mode DIY. Nous allons vous exposer l'ensemble des problèmes qui nous sont arrivés durant cette seconde phase. Ces derniers nous ont fait perdre beaucoup de temps sur le projet, mais ces difficultés, bien que contraignantes, ont constitué des étapes importantes dans notre progression, nous forçant à approfondir nos connaissances, à adapter notre méthodologie et à renforcer la robustesse du système final.



1. Mise à jour des pins

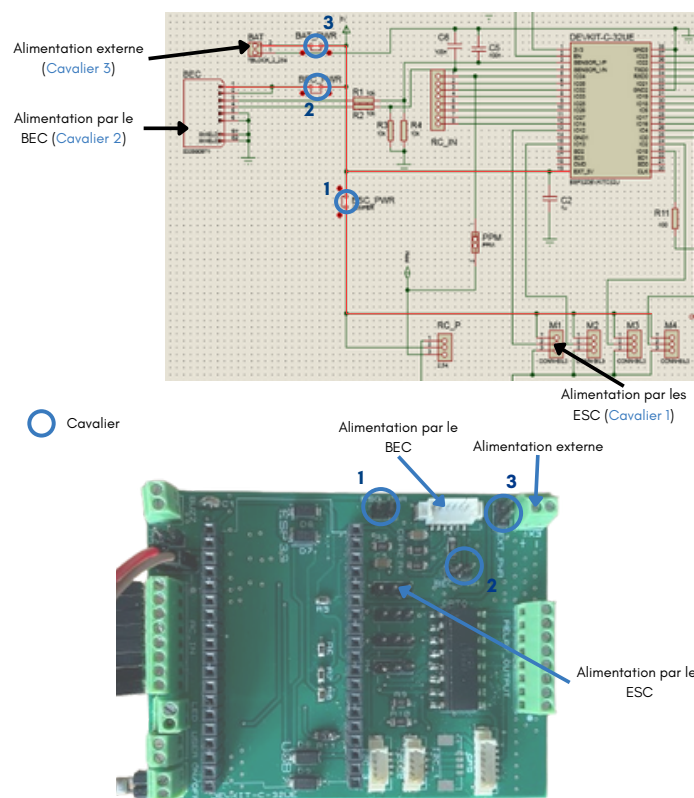
Lorsque nous avons fini les tests avec le GPS, nous souhaitions tester le programme actuel de la bouée en changeant quelques paramètres (par exemple, le nombre de points dans la liste pour faire la moyenne), afin de voir si certaines valeurs stabilisaient mieux la bouée.

Nous avons commencé par implémenter le code du GPS. Comme expliqué lors de la phase 1, nous avons réussi à configurer le GPS directement dans le code plutôt que de passer par le contrôleur Pixhawk. Seulement, impossible d'obtenir des valeurs de position. Nous avons bien passé deux journées entières à décortiquer le code pour au final s'apercevoir que la définition des pins n'était pas la bonne. Depuis le changement de carte PCB, le code n'avait pas été mis à jour.

2. Problème d'alimentation par le BEC

A présent, nous pouvions faire les tests que nous voulions sur la bouée. La carte électronique de la bouée à 4 manières d'être alimentée : par le câble USB de l'EPS32, par une alimentation externe, par les ESC (qui commande les moteurs) et par le BEC. Pour choisir par quel système on souhaite alimenter la carte, on positionne un cavalier afin de faire passer le courant. Ainsi, veuillez n'utiliser qu'un seul cavalier à la fois.

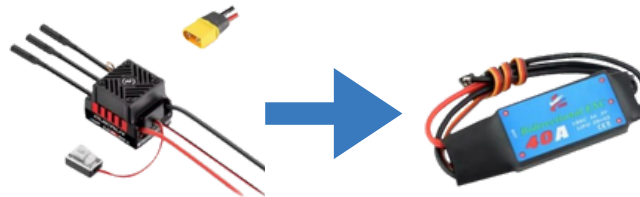




Nous avons choisi d'alimenter la carte par le BEC. Seulement, à chaque fois, cela faisait surchauffer l'ESP32 jusqu'à qu'elle soit inutilisable. Ceci nous est arrivé 2 fois, puis nous sommes passés sur l'alimentation par les ESC ce qui a réglé le problème. Nous ne savons pas encore pourquoi, si l'on passe par le BEC cela détruit l'ESP32. Nous avons pourtant vérifié les tensions. Ainsi, tant que la source du problème n'a pas été trouvée et rectifiée, **veuillez ne pas mettre le cavalier sur l'alimentation du BEC**. En revanche, le BEC sert à récupérer la tension et l'intensité des batteries, donc c'est toujours possible de le brancher sur la PCB.

3. Nouveaux ESC

Nous avons donc rallumé la bouée en alimentant la carte par les ESC quand soudain une fumée est apparue au niveau de la bouée. Après inspection, nous avons identifié que le dysfonctionnement provenait d'un des ESC (contrôleurs de moteur), qui a sûrement eu un court circuit. Pour poursuivre les tests en toute sécurité, nous avons dû remplacer et souder de nouveaux ESC, ce qui a impliqué une réadaptation du système et un léger décalage dans notre planning. Nous en avons profité pour séparer les ESC de l'électronique en les mettant sous la bouée (au fond de la bouée). L'idéal serait également d'y placer les batteries, afin de bien séparer toutes les parties "power" (alimentation) du GPS et des autres composants électroniques.



Cependant, nous ne sommes pas certains que l'algorithme actuel des ESC fonctionne avec les nouveaux. Soit il faut vérifier et modifier la partie du code des ESC ou revenir aux anciens ESC.

4. Panne de radiocommande

Après cette phase de réparation, nous nous sommes rendus compte que nous n'arrivions plus à connecter la radiocommande à son récepteur. Celui-ci a dû prendre un coup avec tous ces problèmes électriques.

Nous avons testé plusieurs radiocommandes en attendant que le récepteur grillé soit remplacé mais aucune ne disposait 8 canaux (4 voies pour le pilotage + 4 commutateurs)

5. Bilan technique

Globalement, nous avons rencontré de nombreux problèmes électriques qui se sont aggravés au fil du temps. Lors de la reprise du projet l'année prochaine, il faudra s'assurer que tous les composants sont opérationnels. Le moindre défaut peut endommager certaines pièces, comme nous avons pu le constater. Lors de notre dernier test avec la nouvelle télécommande, l'ESP32 nous a de nouveau lâchés, alors qu'elle n'a jamais été alimentée par le BEC. Nous pensons que la PCB a également été endommagée et qu'il faudrait la vérifier, voire la repenser pour plus de robustesse.

09 - Conclusion

Ce projet de bouée autonome RoBoNav, intégré dans le cadre d'OCIA à l'ICAM de Nantes, a représenté une opportunité complète de mise en œuvre de compétences en électronique, en informatique embarquée, en connaissance en GPS et en communication sans fil. L'objectif principal était d'améliorer le positionnement GPS de bouées de régate à l'aide d'un système de correction différentielle (DGPS), en combinant la technique d'une base fixe équipée d'un GPS et des rovers embarquant des modules GNSS type M8N ou LC29HDA, avec une communication par Wi-Fi.

Ce projet nous a permis d'acquérir et de renforcer plusieurs compétences techniques :

- **Systèmes GNSS et DGPS** : Nous avons découvert le fonctionnement des modules GPS, le format des trames NMEA, les paramètres de configuration, ainsi que les différents types de coordonnées utilisées. Nous avons également développé nos propres solutions de correction différentielle (DGPS), un domaine qui nous était jusque-là totalement inconnu.
- **Technologie RTK** : Nous avons exploré les systèmes RTK (Real Time Kinematic), que nous ne connaissions pas au départ. Malgré la complexité de cette technologie, nous avons su mener nos recherches et comprendre les principes fondamentaux pour concevoir un système fonctionnel.
- **Démarche de recherche** : Certaines étapes de notre travail se rapprochaient fortement d'un projet de recherche appliquée. Nous avons ainsi renforcé notre capacité à formuler des hypothèses, proposer des solutions, concevoir des expérimentations et analyser des résultats.
- **Communication réseau (Wi-Fi)** : Le projet nous a amenés à travailler de manière intensive avec des réseaux Wi-Fi pour assurer la transmission de données entre les différents modules. Cela nous a permis d'approfondir notre maîtrise des communications sans fil et de la structuration des échanges entre systèmes embarqués.
- **Gestion et analyse de données** : À travers les nombreux tests effectués (et ceux restant à venir), nous avons appris à structurer, stocker et analyser les données GPS. Nous avons notamment manipulé des fichiers JSON et développé des outils pour interpréter les résultats de positionnement.

Les points forts

Lorsque nous avons réalisé que l'utilisation du DGPS avec les modules M8N n'apportait finalement aucun gain, il a été difficile d'accepter que toute la première phase du projet ait peu servi. Malgré cette déception, nous avons su rebondir en développant une nouvelle approche basée sur un algorithme alternatif.

La fin du projet a également été marquée par de nombreuses pannes techniques, ce qui a parfois freiné notre motivation et limité les tests que nous aurions voulu mener. Néanmoins, nous avons persévéré pour résoudre les problèmes, et ces difficultés nous ont permis d'apprendre énormément sur la gestion des imprévus en électronique embarquée.

18