

# M2 EEA SME

Projet Barre Franche - Novembre 2019

---

## EIEAS3GM SYNTHESE ET MISE EN ŒUVRE DES SYSTEMES

---

**Auteurs :**

Nicolas OTAL

Antoine ROUTIER

**Encadrants :**

M. PERISSE



UNIVERSITÉ  
TOULOUSE III  
PAUL SABATIER



Université  
de Toulouse

## Introduction

*Dans le cadre de notre UE "Synthèse et Mise en œuvre de système", un bureau d'étude nous a été proposé avec pour objectif de mettre en œuvre une solution logicielle/matérielle qui répond au besoin de gestion et contrôle de trajectoire d'un voilier de barre franche.*

*A l'aide des bases acquises, durant notre formation, en VHDL, simulation, et conception systèmes l'objectif de cette UE sera la réalisation d'un contrôleur de Barre-France de voilier par FPGA (Altera). Il sera nécessaire d'étudier, décomposer, coder et implémenter chaque fonctions une à une et d'intégrer la globalité du projet à l'aide d'un Bus Avalon permettant l'interconnexion des fonctions au MCU intégré au FPGA.*

*Pour réaliser cela, nous avons respecter le processus de développement consistant à réaliser une analyse des besoins et du contexte permettant d'identifier les différentes interfaces du système, nous avons par la suite réaliser la conception du système en décomposant notre fonction principale en différents blocs. Ces différents blocs ont fait l'objet d'une description fonctionnelle avant d'être implémenter. Suite à cela, nous avons réaliser un ensemble de simulation et test sur maquette pour vérifier le bon fonctionnement de chaque module intégré pour valider finalement notre projet sur une maquette.*

## Sigles et acronymes

<b>CPGPS</b>	<i>Code Phase Global Positioning System</i>
<b>GPIO</b>	<i>General Purpose Input/Output</i>
<b>GPS</b>	<i>Global Positioning System</i>
<b>GSM</b>	<i>Global System for Mobile</i>
<b>I2C</b>	<i>Inter Integrated Circuit</i>
<b>IHM</b>	<i>Interface Homme Machine</i>
<b>NMEA</b>	<i>National Marine Electronics Association</i>
<b>UART</b>	<i>Universal Asynchronous Receiver Transmitter</i>
<b>UML</b>	<i>Unified Modeling Language</i>
<b>RTK</b>	<i>Real Time Kinematic</i>
<b>SPI</b>	<i>Serial Peripheral Interface</i>
<b>SysML</b>	<i>Systems Modeling Language</i>

## Table des matières

<b>Introduction</b>	<b>2</b>
<b>1 Cahier des charges du Projet "Barre-Franche"</b>	<b>5</b>
1.1 Introduction technique . . . . .	5
1.2 Cahier des charges général . . . . .	5
1.3 Cahier des charges technique . . . . .	6
<b>2 Conception Matérielle</b>	<b>7</b>
2.1 Réalisation fonction simple - Gestion anémomètre . . . . .	7
2.2 Réalisation fonction complexe - Gestion vérin . . . . .	8
<b>3 Conception Logicielle</b>	<b>9</b>
<b>4 Validation des Fonctions</b>	<b>10</b>
<b>5 Conclusion</b>	<b>11</b>

# 1 Cahier des charges du Projet "Barre-Franche"

## 1.1 Introduction technique

Le projet qui nous est demandé est basé sur les SoC et plus particulièrement sur un FPGA, de chez Altera, embarqué dans un voilier pour en piloter la barre-franche. Ce dispositif électronique fonctionnera à l'aide de différentes entrées/sorties (gyroscope, anémomètre, GPS, convertisseur analogique/numérique, vérin, boutons, buzzer).

## 1.2 Cahier des charges général

Durant ce projet, nous allons utiliser les différentes compétences acquises à travers les différents cours de l'année et les mettre en corrélation pour mener à bien ce dernier. Le projet devra respecter certains critères présentés ci-dessous :

1. Le dispositif devra utiliser un appareil de mesure pour capter la valeur de la vitesse du vent.
2. Le dispositif devra utiliser un appareil de mesure pour capter la direction du vent.
3. Le dispositif devra réceptionner des données GPS, traiter ces données et agir sur le dispositif en fonction des résultats obtenus après traitement.
4. Le dispositif devra avoir une interface entre l'opérateur et le voilier (boutons, Leds et buzzer).
5. Le dispositif devra piloter la barre-franche du voilier à l'aide des différents appareils pour venir piloter un vérin (Vérin et compas).

La figure 1 modélise le système qui sera réalisé avec ses différentes fonctions et les différents flux d'informations nécessaires au système "Barre-Franche".

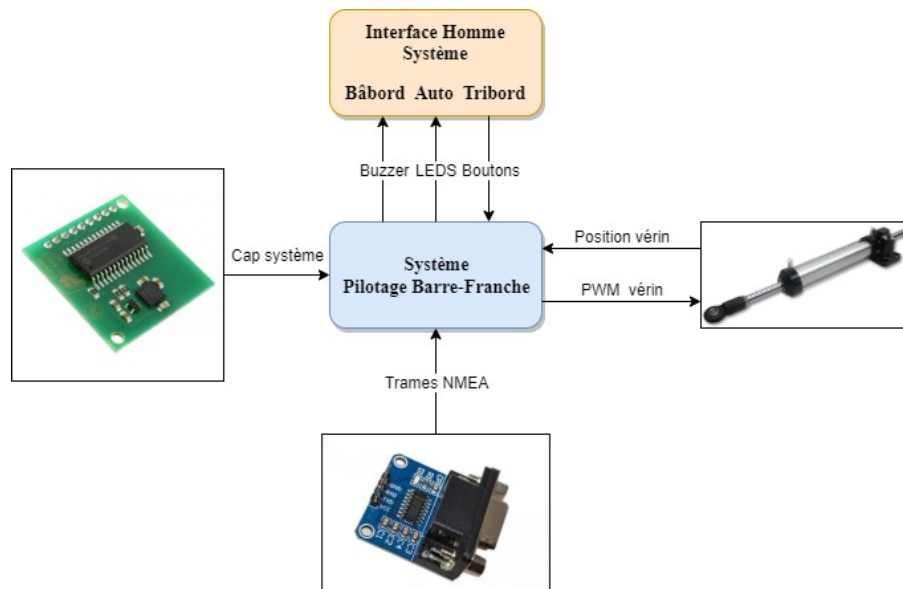


FIGURE 1 – Diagramme contexte du système

Pour des raisons pratiques et de temps nous avons implémenté la mesure de la vitesse du vent, la gestion du vérin, la gestion du compas ainsi que l'asservissement du vérin, nous n'avons pas implémenté la gestion des trames NMEA, mais avons ajouté l'interface hommes système (boutons, leds et buzzer).

### 1.3 Cahier des charges technique

Le projet se porte sur la réalisation d'un dispositif embarqué à base de FPGA, de chez Altera. Il sera composé de deux parties principales une partie Hardware sur le FPGA et une partie Software intégrée/développée dans le FPGA (SOPC). Les deux parties communiqueront par le biais du Bus Avalon, qui est le Bus développé par Altera pour leur SOC. Le cœur du projet sera composé d'un FPGA Cyclone IV EP4CE22F17C6N du fondeur Altera permettant le développement du projet Hardware et Software sur la même carte d'évaluation. Différentes fonctions seront implémentées :

1. Une fonction qui permettra de lire la mesure de la vitesse du vent (0-250km/h). La fonction devra lire la sortie de l'anémomètre qui est une sortie logique de fréquence variable (0 à 250 Hz).
2. Une fonction générant un signal PWM qui sera utilisé dans plusieurs parties du projet. Cette fonction sera intégré plus tard dans le SOPC du FPGA permettant la génération d'un signal PWM qu'on utilisera au travers du Bus Avalon.
3. Une fonction de gestion vérin gérant le pilotage de la barre franche.
4. Une fonction qui utilise un compas pour récupérer les mesures d'angles sur le plan horizontale du voilier, permettant de donner un cap à celui-ci.
5. Une fonction qui permet la gestion de l'interface Homme système (composée de différents boutons[Bâbord, Tribord, Auto/Manuel], des LEDS ainsi qu'un buzzer).
6. Une implémentation d'un MCU dans le FPGA grâce à l'outil SOPC du logiciel d'Altera. Celui-ci permettra le traitement et l'affichage des différentes variables du projet (cap du voilier, position vérin, position GPS, gestion des PWM et la vitesse du vent).

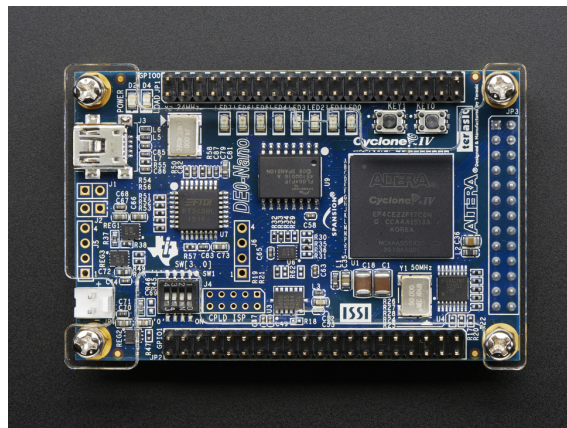


FIGURE 2 – DE0 Nano Altera

## 2 Conception Matérielle

Pour mettre en oeuvre les différentes parties du projet combinatoires et séquentielles, nous avons du passer par des étapes d'analyses comportementales de chaque blocs à réaliser pour ainsi créer des composants simples pouvant être utiliser à la composition de fonctions plus complexes plus tard. L'implémentation de chaque composants a été réalisée grâce au langage VHDL sur le logiciel de développement Quartus.

Ainsi nous avons réalisé différentes "boîtes" de composants simples pour réaliser des fonctions entières, et plus complexes, contenant les blocs simples.

### 2.1 Réalisation fonction simple - Gestion anémomètre

La fonction réalisant l'acquisition de la vitesse du vent (0 à 250 Km/h) se fait à l'aide d'un anémomètre qui sert de transducteur convertissant la vitesse du vent en fréquence variable (0 à 250 Hz).

Le composant doit fonctionner en deux modes :

1. Mode continu, dans ce mode le système actualise l'acquisition toutes les secondes de la vitesse du vent.
2. Mode mono-coup, dans ce mode le système effectue qu'une seule acquisition lorsque "Start/stop" est activé. Une fois "data\_valide" envoyé le système remet les entrées "start/stop" et "data\_valide" à zéro après une acquisition.

La figure 3 représente la décomposition des composants utilisés pour la réalisation du bloc "Captage direction vent". Composé d'un diviseur de fréquence (pour obtenir une fréquence d'un Hertz, acquisition toutes les secondes), d'un détecteur de fronts montants (permettant d'indiquer le nombre de fronts montants correspondant à la fréquence mesurée), d'une machine à état (pour décrire séquentiellement les modes continu/mono-coup), un compteur (pour l'acquisition du nombre de fronts montants toutes les secondes) et une Bascule D (permettant la mémorisation de la valeur et l'acquisition avant obtention de la nouvelle valeur).

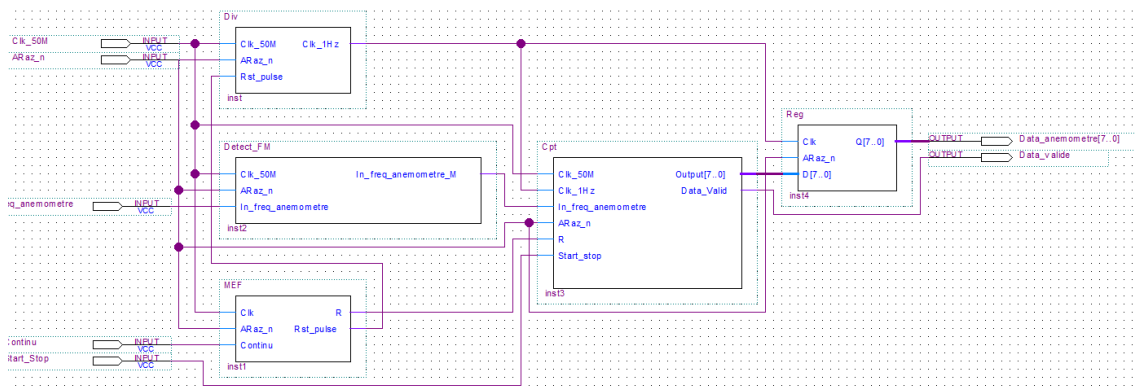


FIGURE 3 – Diagramme fonctionnel du circuit "Captation vitesse vent"

## 2.2 Réalisation fonction complexe - Gestion vérin



### 3 Conception Logicielle

## 4 Validation des Fonctions

## 5 Conclusion