

Projet C++ : Simulateur d'un système carburant d'avions de chasse

Le système carburant d'un avion alimente ses moteurs en carburant. Il est composé en général de plusieurs réservoirs, d'un système de pompes, de vannes et de tuyaux permettant d'acheminer le carburant aux moteurs. Comme tout système, le système carburant peut avoir des pannes, par exemple, la panne d'une pompe pendant le vol, un réservoir qui se vide, une vanne qui se bloque, etc. Pour assurer l'acheminement du carburant à tous les moteurs pendant le vol, il est possible de re-configurer le système pour mettre en marche une pompe de secours ou alimenter un moteur par un autre réservoir. A tout moment pendant le vol, le pilote peut voir l'état du système carburant et agir sur ce dernier. Il peut notamment mettre en marche les pompes de secours, ouvrir ou fermer les vannes pour changer l'acheminement du carburant. Pour éviter les erreurs au cours d'un vol, le pilote a besoin de s'entraîner.

Le but de ce projet est de développer un mini-simulateur du système carburant d'un avion. Ce simulateur a pour but d'entraîner les pilotes à gérer le système carburant en cas de crise.

1 Description du système

Le système carburant : le système de carburant simplifié d'un avion de chasse est représenté sur la Figure 1. Il est composé de 3 réservoirs indépendants : *Tank1*, *Tank2*, *Tank3*. Avec un fonctionnement normal du système, chaque réservoir alimente son moteur : *Tank1* alimente le moteur *M1*, *Tank2* alimente le moteur *M2* et *Tank3* alimente le moteur *M3*. Il faut noter que les réservoirs *Tank1* et *Tank3* ont une plus grande capacité que le réservoir *Tank2*. Le carburant est acheminé aux moteurs à l'aide d'un système de pompes. Pour chaque réservoir il existe 2 pompes, une pompe primaire et une pompe de secours. Dans le fonctionnement normal, les pompes de secours sont arrêtées.

Pendant le vol, des événements imprévus peuvent survenir. Nous pouvons considérer le cas d'un réservoir qui se vide ou le cas d'une pompe qui tombe en panne. Pour remédier à ces pannes deux systèmes de re-configuration sont prévus :

1. Les trois réservoirs sont reliés par des tuyaux pour équilibrer le niveau de carburant dans chaque réservoir. Si les vannes VT12 et VT23 sont fermées, le carburant peut circuler d'un réservoir à l'autre. Si l'un des réservoirs se vide, le pilote peut fermer les vannes VT12 et/ou VT23 pour remplir le réservoir vide avec le carburant provenant des autres réservoirs.
2. Chaque pompe peut alimenter n'importe lequel des trois moteurs. Chaque moteur a besoin d'une seule pompe pour être alimenté. Si l'une des pompes tombe en panne, le pilote peut démarrer la pompe secondaire. Si les 2 pompes sont en panne, il est possible d'acheminer le carburant au moteur depuis un autre réservoir. Il suffit que le pilote démarre la pompe secondaire et ferme la vanne correspondante (V13, V12 ou V23).

Le tableau de bord du pilote : la partie du tableau de bord du pilote permettant de gérer le système carburant de l'avion est représentée sur la Figure 2. Le pilote peut réaliser les actions suivantes :

- Ouvrir ou fermer les vannes VT12 et VT23 (boutons VT12 et VT23 Figure 2) ;
- Ouvrir ou fermer les vannes V12, V13, V23 (boutons V12, V13 et V23 Figure 2) ;
- Démarrer ou arrêter les pompes de secours (boutons P12, P22 et P32 Figure 2).

2 Travail à réaliser

Il s'agit donc de développer un mini-simulateur du système carburant d'un avion. Ce simulateur a pour but d'entraîner les pilotes à gérer le système carburant en cas de crise. L'application comprendra deux interfaces :

- Une interface représentant le tableau de bord de l'avion. Elle permet à l'utilisateur d'agir sur le système carburant de l'avion. Chaque action de l'utilisateur modifie l'état du système.
- Une interface qui décrit l'état du système pendant la simulation. Elle permet à l'utilisateur d'observer l'état du système pendant la simulation. Cette interface doit aussi permettre à l'utilisateur d'injecter les pannes dans le système : vidange d'un réservoir ou défaillance d'une pompe.

2.1 Pour commencer

Tout d'abord, définissez et implémentez le modèle de données pour représenter le système carburant de l'avion. Pour simplifier, on suppose que le réservoir peut être dans deux états : plein ou vide ; que les pompes peuvent être soit en marche, soit en arrêt, soit en panne ; les vannes peuvent être soit ouvertes, soit fermées (on ne prendra pas en compte les éventuelles défaillances des vannes).

Le pilote peut effectuer les actions suivantes :

- ouverture et fermeture des vannes ;
- démarrage et arrêt des pompes.

L'utilisateur doit être capable de déclencher les événements de pannes du système suivants :

- panne des pompes ;
- vidange des réservoirs.

2.2 Pour continuer

Implémentez le simulateur de votre modèle. Si toutes les pompes sont en marche et l'un des réservoirs se vide, le pilote doit privilégier la fermeture des vannes VT12 et/ou VT23 pour rééquilibrer le niveau de carburant dans les réservoirs. Si par contre les deux pompes d'un réservoir sont en panne, il est possible d'acheminer le carburant par une pompe de secours d'un autre réservoir. Pour cela, le pilote doit mettre en marche une pompe supplémentaire et ouvrir la vanne correspondante (V12, V13 ou V23).

La deuxième partie de l'interface doit au moins contenir les informations suivantes :

- l'état de chaque réservoir (plein/vide) ;
- l'alimentation de chaque moteur (provenance du flux carburant) ;
- l'état des chaque pompe (marche, panne, arrêt) et de chaque vanne (ouverte/fermée).

2.3 Pour finir

Implémentez une fonctionnalité d'exercices pour le pilote et de sauvegarde de l'historique de son entraînement. Cet historique permet l'évaluation des compétences de chaque pilote et de lui attribuer une note. Vous pouvez considérer les notes dans l'intervalle $[0 - 10]$. Pour cette fonctionnalité, vous aurez à implémenter un système d'authentification et une base des utilisateurs possibles du système.

3 Remise du projet

Ce travail est à réaliser en binôme. Vous devez fournir, en plus de listing commenté et expliqué du programme, un fichier `.tar.gz` contenant les sources et un mécanisme d'installation. Un compte rendu expliquant l'architecture de votre application vous est aussi demandé. Les soutenances seront organisées pour voir le fonctionnement des applications.

Date de remise : 11 mai 2015. Attention, cette date pourra être éventuellement avancée.

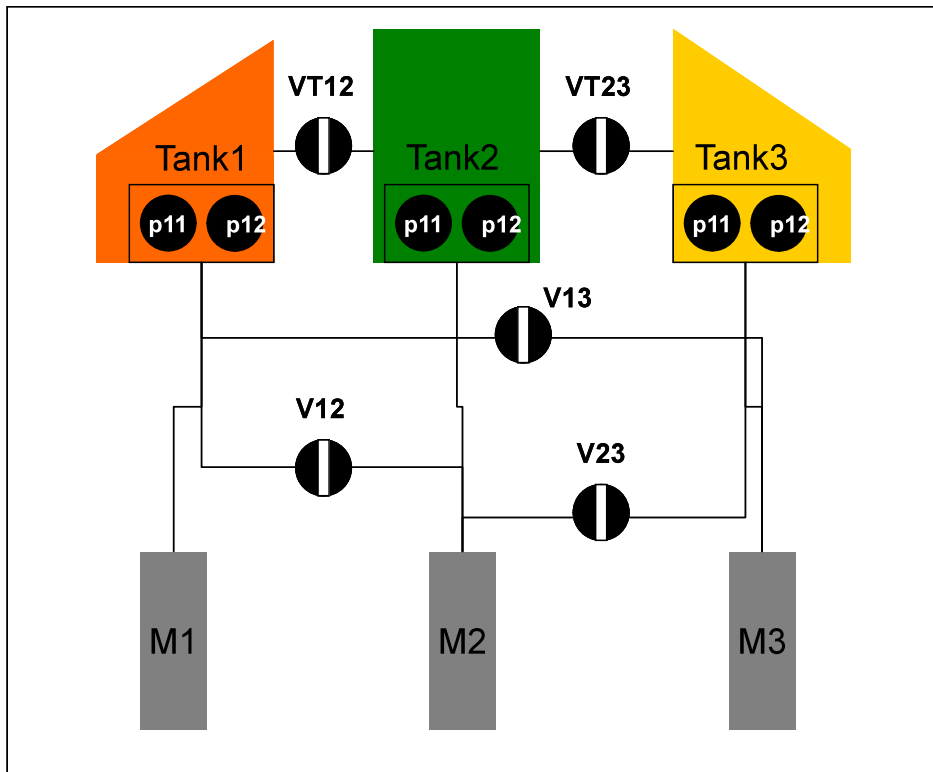


Figure 1 : Le système carburant simplifié.

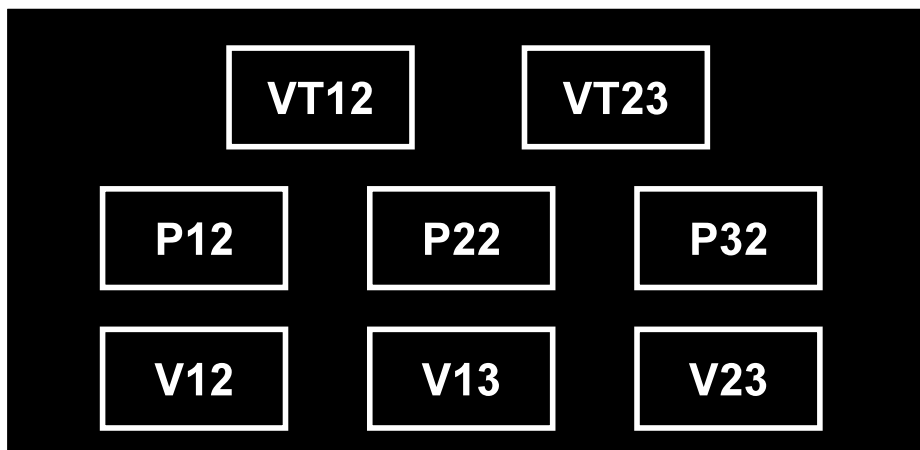


Figure 2 : Le tableau de bord de pilote, permettant de gérer le système carburant.