**DIFFUSION RESTREINTE**

Une image contenant Graphique, graphisme, Police, logo

Description générée automatiquement

Notice d’utilisation de l’interface homme-machine

Propulsion / Banc de test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Document initial** | |  |  |  |
| **Version** | **A** | B | C | D |
| **DATE**  **VISAS** | 29/07/24 |  |  |  |
| **Rédacteur** | Mehdi Delouane |  |  |  |
| **Validation** |  |  |  |  |

**LISTE DE DIFFUSION**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Noms | Organisme | Nb. | Pour | | |
| Approb. | Accept. | Info. |
| Diffusion interne | | | | | |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| Diffusion externe | | | | | |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**ENREGISTREMENT DES EVOLUTIONS DU DOCUMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Edition** | **Date** | **Pages** | **Référence et résumé des modifications** |
| A | 26/07/24 | 21 | Document initial |

**TABLE DES MATIERES**

[I. Terminologie 5](#_Toc164449343)

[Références 6](#_Toc164449344)

[II. Guide utilisateur 7](#_Toc164449345)

[1. Introduction 7](#_Toc164449346)

[2. Description de l’électronique du banc 9](#_Toc164449347)

[3. Prise en main 11](#_Toc164449348)

[3.a. Lancement du logiciel 11](#_Toc164449349)

[3.b. Interface utilisateur 11](#_Toc164449350)

[4. Exemple d’utilisation 15](#_Toc164449351)

[Documents annexes 21](#_Toc164449352)

[Annexe A : Synoptique fluide du banc sans débitmètres (1er) et avec débitmètres (2e) 21](#_Toc164449353)

[Annexe B : Schéma de câblage 23](#_Toc164449354)

**Liste des figures**

[Figure II‑I : Interface utilisateur du logiciel de pilotage du banc d'essai 9](#_Toc164428438)

[Figure II‑II : Exemple de fichier de séquence 10](#_Toc164428439)

[Figure II‑III : Affichage du choix et du lancement d’un essai 14](#_Toc164428440)

# Terminologie

Les sigles, abréviations et acronymes utilisés dans le présent document sont listés ci-dessous.

|  |  |
| --- | --- |
| **Acronyme** | **Signification** |
| EV | Electrovannes |
| IDE | Integrated development environment |
| NO | Normally open |
| NC | Normally close |
| PC | Poste de contrôle |
| GPIO | General Purpose Input/Output |
| IHM | Interface homme-machine |

# Références

|  |  |  |
| --- | --- | --- |
| **ID** | **Titre du document** | **Référence** |
| REF 1 |  |  |
| REF 2 |  |  |

# Guide utilisateur

## Introduction

Cette interface a été développé pour offrir une solution complète afin de contrôler et de surveiller les tests d’allumeur torche du moteur-fusée. Réalisé en Python et doté d’une interface graphique intuitive, ce logiciel permet d’agir en temps réel avec le banc d’essai via une connexion sans fil à une carte Beagle Bone Black, elle-même reliée en Ethernet à une carte STM32 Nucléo. Le logiciel utilise le multithreading pour répondre à des contraintes de temps réel ainsi que diverses bibliothèques Python permettant un niveau d’abstraction plus élevé.

**Objectif du Logiciel :**

Ce logiciel vise à simplifier et à optimiser le processus de test d’allumage de l’allumeur torche du moteur-fusée en fournissant une plateforme conviviale pour contrôler les électrovannes, acquérir les données des différents capteurs et effectuer diverses actions pendant les tests. Le but étant de pouvoir confirmer le bon fonctionnement de l’allumeur torche.

**Principales fonctionnalités :**

* Contrôle des électrovannes (manuel et automatique) ;
* Acquisition en temps réel des données des capteurs ;
* Ecriture des données en temps réel dans un fichier csv avec horodatage des mesures ;
* Interface graphique intuitive permettant une visualisation pratiquedes données et des interactions faciles ;
* Personnalisation de l’essai grâce à une banque de séquence d’essai

**Configuration requise :**

* Système d’exploitation : Windows ;
* Python ;
* Bibliothèques Python : sys, os, socket, csv, threading, functools, numpy, datetime, PyQt5 (QtWidgets, QtGui, QtCore), matplotlib (pyplot, backends, figure).
* .

**Exécution du logiciel :**

Pour exécuter le logiciel, ouvrez votre IDE préféré tel que VScode et chargez le fichier Python principal (*IHM.py*) du logiciel. Vous pourrez ensuite exécuter le script Python directement depuis l'IDE. Assurez-vous que tous les documents nécessaires (Logo\_*AndroMach.png, Synoptique\_Allumeur.png, Tableau.png ainsi que les différents fichiers (.txt) de séquence)* sont présents dans le même répertoire que le programme principal pour un fonctionnement correct du logiciel.

Ce logiciel vous permettra de contrôler les électrovannes, d'acquérir les données des capteurs et d'effectuer diverses actions pendant les tests d’allumage de l’allumeur. Nous vous encourageons à lire attentivement cette notice d'utilisation pour une utilisation optimale du logiciel.

## Description de l’électronique du banc

L’électronique du banc s’articule autour des éléments suivants :

* 3 capteurs de pression ;
* 2 débitmètres ;
* 2 thermocouples
* 7 électrovannes (6NC et 1 NO) ;
* 1 bougie d’allumage
* 1 carte électronique STM32 F207ZG ;
* 1 carte électronique Beable Bone Black ;
* 1 poste de contrôle (ordinateur).

Les capteurs de pression permettent de connaître la pression des fluides à l’intérieur du banc. Ces mesures sont effectuées conformément au synoptique fluide (Annexe A). Les mesures des capteurs sont récupérées et traitées directement par la carte électronique STM32 sur des pins analogiques de l’ADC1 (ADC1\_IN3, ADC1\_IN10 et ADC1\_IN13) pour les mesures de pression.

Les débitmètres sont basés sur un capteur à effet hall et une petite turbine. Le capteur à effet Hall permet de détecter les passages des pales de la turbines. Ce capteur émet une pulsation à chaque passage de pale. Ces pulsations sont détectées par les pins GPIO (PB8 et PB9). Suivant le nombre d’impulsion détecté dans un délai de temps connu, il est possible de déduire le débit du fluide.

Un thermocouple est un capteur de température qui fonctionne sur le principe de l'effet Seebeck. Il est composé de deux fils métalliques différents, soudés ensemble à une extrémité pour former une jonction. Lorsque cette jonction est exposée à une température, elle génère une tension électrique proportionnelle au différentiel de température entre cette jonction (jonction chaude) et une jonction de référence (jonction froide) située à l'autre extrémité des fils. Cette tension peut être mesurée et convertie en une valeur de température grâce à des tables de référence ou des appareils de mesure électroniques.

Les débitmètres et les capteurs de pression sont alimentés en parallèle depuis une pin « 5V » de la carte. Les thermocouples n’ont pas besoin d’être alimenté du fait de leurs fonctionnements.

La bougie d’allumage est branchée suivant le schéma suivant (schéma de câblage de la bougie). Il faut noter que, la bougie ne fonctionne pas en continue, elle doit impérativement alterner entre l’état ON/OFF pour générer une étincelle d’où la présence d’un relai.

Les électrovannes sont pilotées par la carte STM grâce à des relais 5V. Ces relais sont alimentés depuis le bloc d’alimentation 24V du banc. Un convertisseur de tension permet de passer à une tension de 5V. Le signal de commande est transmis depuis des pins GPIO (PF13, PE9, PE11, PF14, PE13, PF15, PG14). Les électrovannes sont raccordées en Normally Open (NO) ce qui signifie que lorsque le signal de commande est actif, le relais se ferme. A l’exception du relai connecté au pin PE13, quand le relais se ferme, la vanne s’ouvre. Le relai du pin PE13 contrôle une vanne NO, la fermeture du relai va donc fermer la vanne. Les électrovannes sont alimentées en parallèle en 24Vdc. Les connecteurs des électrovannes présentent un troisième fil n’étant pas utilisé dans notre cas. Le contrôle des vannes se fait par l’intermédiaire de nombre binaire ou chaque bit correspond à un relai. Un bit portant la valeur « 1 » va envoyer un signal au relai et donc le fermer. Inversement, une valeur « 0 » signifie l’absence de signal donc un relai ouvert. Ce nombre binaire est traité par la carte qui renvoie un message de validation « OK » pour confirmer à l’interface le changement d’état.

Une fois correctement traitées, une mise en forme est appliquée afin de pouvoir communiquer les données de mesure au poste de contrôle (PC). Les 15 données (1 pour le temps, 7 pour les capteurs et les 7 bits pour l’état des vannes) sont envoyées en un seul message. Les valeurs sont séparées par des virgules et précédées par un nom associatif. Ci-contre, un extrait du message envoyé.

Time=1000, PS1=10, PS2=103, …

Il faut noter que l’envoi de nombre décimal est difficile. Pour faciliter l’envoi, certaines valeurs ont été multiplié pour permettre l’accès à une meilleur précision des valeurs. Elles seront retraitées par l’IHM.

Il est également important de noter que chaque pin possède un rôle particulier. En effet, bien que les capteurs soient identiques, les mesures récupérées par la carte sont traitées afin d’être ordonnées. Par exemple, si le pin du capteur PS1 est échangé avec celui du capteur PS2, les valeurs affichées sur l’IHM seront inversées. **Il est donc essentiel de se référer au schéma de câblage (voir Annexe B) et de le respecter lors du montage des capteurs**. Pour assurer un bon branchement, un étiquetage est présent sur les câbles sortant du boîtier électronique permettant d’indiquer quel capteur doit être branché sur chaque câble.

L’entièreté des échanges entre la carte STM et le PC se font via la Beagle Bone. La connexion STM-Beagle Bone est effectué via un câble Ethernet et la connexion PC-Beagle Bone se fait sans fil.

## Prise en main

### Lancement du logiciel

* Ouvrez l’IDE de votre choix (ex : Spyder).
* Chargez le fichier Python « *IHM.py* ».
* Assurez-vous que les documents « *Tableau.png »*, « Logo\_*AndroMach.png », « Synoptique\_Allumeur.png »* se trouvent dans le même répertoire que « *IHM.py* ».
* *Rédiger les fichiers de séquence (au moins un) dans des .txt en suivant le format décrit précédemment. Assurez vous également qu’ils se trouvent dans le même répertoire que* « *IHM.py* »
* Connecter la carte STM à la Beagle Bone via un câble Ethernet et connecter
* Paramétrer les adresses de communication sur la STM, Beagle Bone et l’IHM
* Vérifiez la bonne transmission en utilisant la commande « ping » de l’invité de commande. *Pingez les adresses IP de la Beagle Bone et de la STM*
* Une fois ces modifications effectuées, exécutez le script « *IHM.py* ».
* Si l’interface apparaît correctement et qu’aucun message d’erreur n’est affiché dans le terminal de votre IDE, l’application est prête à usage.
* Si des problèmes persistent, il est probable que les adresses IP utilisées soient la source du problème.

### Interface utilisateur

Ci-dessous, un aperçu de l’interface utilisateur du logiciel :

A diagram of a machine

Description automatically generated

Figure III‑I : Interface utilisateur du logiciel de pilotage du banc d'essai

Cette interface est composée de 5 sections principales.

**Synoptique et contrôle des vannes :**

La section principale est le synoptique du banc de l’allumeur. Il résume les principaux éléments constituant la fluidique du banc tels que les vannes (solénoïde ou manuel), les réservoirs, les capteurs, l’allumeur et les différents tuyaux reliant ces éléments. Pour l’accompagner, 7 sous-sections sont présentes. Chacune d’entre elle permet de commander manuellement les électrovannes. Une sous-section est composée : d’une frame, d’un label qui rappelle le nom de la vanne, un label qui rappelle l’état actuel de la vanne (ouvert/fermé) et de deux boutons servant respectivement à ouvrir et fermer la vanne. L’état d’une vanne est affiché sur sa sous-section associée et dans un tableau récapitulatif. Ce tableau a pour but de résumer l’état de toutes les vannes évitant ainsi de chercher en permanence la sous-section d’une vanne pour voir son état. Le bouton « SPARK » permet à l’utilisateur d’allumer la bougie. Ce bouton change de couleur en fonction de son état. Un bouton rouge signifie une bougie éteinte et un bouton vert signifie une bougie allumée. Enfin, des labels sont visibles au niveau des différents capteurs pour afficher en temps réel leurs valeurs.

**Choix et lancement d’une séquence**

Sur la partie droite de l’IHM se trouve 4 zones, la première concerne le choix et le lancement d’une séquence d’allumage. Une combobox permet de répertorier les différents fichiers (.txt) que vous aurez rempli à l’avance. Ces fichiers doivent suivre un format bien précis.

A screen shot of a computer

Description automatically generated

Figure III‑II : Exemple de fichier de séquence

La première ligne représente l’état initial de chaque relai grâce au format binaire. Il est important de considérer le type de vanne (NO/NC) puisque qu’un relai fermé n’implique pas nécessairement une vanne fermée.

Les lignes suivantes représentent les délais de « toggle » des vannes. Chaque ligne commence par le nom de la vanne et les temps en millisecondes ou la vanne doit changer d’état. Ainsi, le relai associé à EV0 sera fermé puis 1 seconde après le lancement de la séquence du test, il s’ouvrira pour se rouvrir 1 seconde après ce changement. 500ms plus tard, il se refermera et finira ouvert 500ms après soit 3 secondes depuis le début de la séquence.

*Attention : Prenez garde au format. Ne remplacer pas les virgules, ne rajouter pas de ligne ou d’espace et assurer vous que le nombre de chiffre dans la première ligne correspond bien avec le nombre de vanne. Les temps doivent forcément être rangés dans l’ordre. Il est fortement conseiller de garantir un certain délai entre deux changements d’état.*

Il suffira de cliquer sur l’un d’eux pour le sélectionner. Le bouton « View checklist » permet d’afficher un chronogramme de la séquence. Enfin, le bouton « Start » permet d’entamer le processus d’allumage. En cliquant dessus, la STM va configurer les vannes dans un état initial et attend la confirmation de l’utilisateur pour continuer. Un chronomètre va alors s’afficher. Le premier bouton « Start Countdown » permet d’entamer le décompte avant allumage et le bouton « Stop » permet d’interrompre le décompte. Lorsque le décompte tombe à zéro, le contrôle des vannes sur le synoptique est bloquée et un chronomètre se lance pour afficher le temps depuis allumage.

A screenshot of a computer

Description automatically generatedA screenshot of a clock

Description automatically generated

Figure III‑III : Affichage du choix et du lancement d’un essai

**Affichage des graphiques**

La deuxième zone permet d’afficher les graphiques. Cette nouvelle fenêtre peut être décomposé en deux sections : les graphiques en temps réel et les options d’affichage. Les options d’affichage concernent la taille de l’axe « temps » c’est à dire le nombre de valeur à afficher à chaque instant. L’utilisateur a ainsi le choix d’afficher les 10 ou 30 dernières secondes, la dernière minute ou l’intégralité des valeurs depuis le lancement de l’IHM. Chaque grandeur (pression, température et débit) possède plusieurs capteurs impliquant donc plusieurs courbes. Par souci de lisibilité, il est possible de choisir les courbes à afficher. Il faut noter que cette fenêtre peut être ouverte ou fermée à n’importe quel moment même pendant le test.

**Emergency**

La troisième zone concerne l’arrêt d’urgence, mis en évidence par un bouton rouge. Il permet tout simplement d’interrompre toute action en cours que la carte STM pourrait être en train d’effectuer. Ce bouton est à utiliser principalement durant la phase de test si besoin. Une fois l’arrêt enclenché, une séquence est activée. Toutes les vannes sont, dans un premier temps, fermées puis une purge est effectuée en ouvrant les vannes nécessaires. Il est alors possible de recontrôler les vannes en cas de besoin. Ce bouton d’arrêt d’urgence interrompt également le chronomètre du test.  
  
**Fermeture de l’interface**

La dernière zone est le bouton fermeture. Afin d’éviter tout action involontaire, il a été décidé de désactiver le bouton de fermeture de fenêtre habituellement localisé en haut à droite de la fenêtre. Un nouveau bouton a été créé afin de le remplacer. Si celui-ci est sélectionné, une petite fenêtre warning apparaitra afin que l’utilisateur confirme la fermeture du programme.

## Exemple d’utilisation

Je souhaite mener un essai sur un allumeur torche. Après avoir vérifié les connectiques, le fonctionnement des différents éléments du banc et mis en place un périmètre de sécurité, je dois établir un ou plusieurs fichiers .txt au format adéquat ou vérifier le format de ceux déjà existant. Dans le programme « IHM.py », je dois vérifier les paramètres de lancement, notamment les adresses de communication et leur port et le type de nom du fichier csv (daté ou défaut). Je peux alors exécuter le script « *IHM.py* ».

L’interface graphique s’ouvre. Je vérifie rapidement la console de l’IDE pour s’assurer qu’aucun message d’erreur n’est présent. Je vérifie ensuite que les capteurs reçoivent des données et que celle-ci sont cohérentes. Il est également conseillé de vérifier le bon fonctionnement la fenêtre des graphiques notamment que l’affichage a bien lieu en temps réel.

En accord avec les procédures rédigées à l’avance, je teste l’ouverture et la fermeture des vannes.

*Attention : en cas de problème, ne fermer pas immédiatement l’interface car aucun reset de position n’est programmé. Il est donc possible qu’une vanne reste dans une position non désirée. Si toutefois, l’interface venait à fermer (volontairement ou non), l’interface est programmée pour forcer une position initiale des vannes a son lancement. Relancer le programme peut donc permettre un reset mais il reste préférable de ne pas recourir à cette solution.*

Une fois les premières vérifications effectuées et validées, je peux choisir la séquence voulu pour mon test. Je peux vérifier que la séquence correspond au test voulu grâce au bouton « View Checklist ». Je vérifie que les vannes sont positionnées correctement à partir du tableau récapitulatif. Je poursuis si aucune anomalie n’a été détecté. Lorsque le test est lancé, je peux ouvrir la fenêtre des graphiques pour observer les courbes et potentiellement détecté un problème. Si un important problème est détecté, fermer ou minimiser la fenêtre des graphiques et appuyer sur le bouton « EMERGENCY ». En accord avec les procédures et une fois la purge finie, ouvrir ou fermer des vannes.

Dans le cas où le test se déroule correctement, toujours en accord avec les procédures, réaliser les différentes actions sur les vannes (purge, vidange, …).

Pour relancer un essai, je ferme l’interface et la relance pour assurer l’initialisation des vannes et l’écriture des données dans un nouveau fichier csv.

Je peux également accéder au fichier de sauvegarde et appliqué les traitements et analyses nécessaires.

# Documents annexes

## Annexe A : Synoptique fluide du banc

A diagram of a machine

Description automatically generated

## Annexe B : Schéma de câblage

A diagram of a computer

Description automatically generated