

Rapport personnel

rapport personnel Loïs



# Table des matères

Définition du projet 3

 répartition des tâches 3

organisation 4

 Teams 4

 GitHub 5

 Memory of moments 6

déroulement du projet 6

 prise en main de SRS 2022 6

 Paramétrage d’automation studio 7

 Premiers tests 8

Recettes 8

Importation du fichier XDD 8

Prépartation du robot 9

# Définition du projet

Ce projet a été fait sur la demande de l’entreprise Castel Frères et a été réalisé sur deux ans.

Actuellement le système de déboxage de l’entreprise est géré par un humain qui a pour tâche de prendre les bouteilles d’un box une a unes afin de les déposer sur un convoyeur. Cela pose des problèmes musculo squelettiques pour la personne en charge du déboxage étant donné la répétition et la fatigue que cette tache engendre.

Pour pallier a ce problème un système de déboxage automatique a été pensé. En effet ce ne serait plus un humain mais un robot à qui cette tâche serait confiée.

Le système pensé est divisé en 3 parties principales : une partie Vision qui permet de prendre les coordonnées de chaque objet a déboxer en temps réel, une partie robotique avec un bras robotisé permettant de prendre les bouteilles et de les poser sur un convoyeur et enfin une partie base de données permettant de stocker toutes les informations relatives au système.

## répartition des tâches

Une image contenant texte, diagramme, capture d’écran, Plan

Description générée automatiquementSur le schéma ci-dessous est affiché la répartition des tâches entre les 3 membres du groupe.

# organisation

L’organisation au sein du projet s’est faite grâce à 3 outils principaux ; Teams, GitHub et l’utilisation d’un MOM (Memory Of Moments)

## Teams

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquementDans un premier temps nous avons crée un canal sur Teams afin de pouvoir y déposer nos fichiers. Le but principal de ce canal était de permettre de stocker tous les rapports et annexes que nous aurions pu créer au cours du projet.

Chacun des membres de l’équipe avait son propre dossier pour ses fichiers persos relatifs à la partie qu’il traitait (ici BDD, VISION et ROBOT). Un dossier était dédié au MOM, un aux rapports communs et un dernier aux différentes ressources d’ont nous pourrions avoir besoin.

Teams avais aussi pour fonction de pouvoir mettre en place des Visio si jamais un membre de l’équipe était absent mais nous ne l’avons pas utilisé de cette manière au final.

Nous nous sommes servis un petit moment de Teams avant de passer sur GitHub qui était plus simple d’accès sur les PC, qui se mettais à jour automatiquement et qui était plus simple d’pour la gestion globalement.

## GitHub

GitHub a été notre moyen de gestion des fichiers principal. Il permet de travailler directement dans un dossier sur notre ordinateur en étant sur que chaque personne possède les mêmes fichiers. Il est aussi plus rapide d’utilisation que Teams qui était assez lent dans certains moments en raison de la faible puissance des ordinateurs.

Pour l’utiliser nous avons installé la version desktop du logiciel qui permettait de mettre à jour les fichiers directement dans un dossier sur le pc.

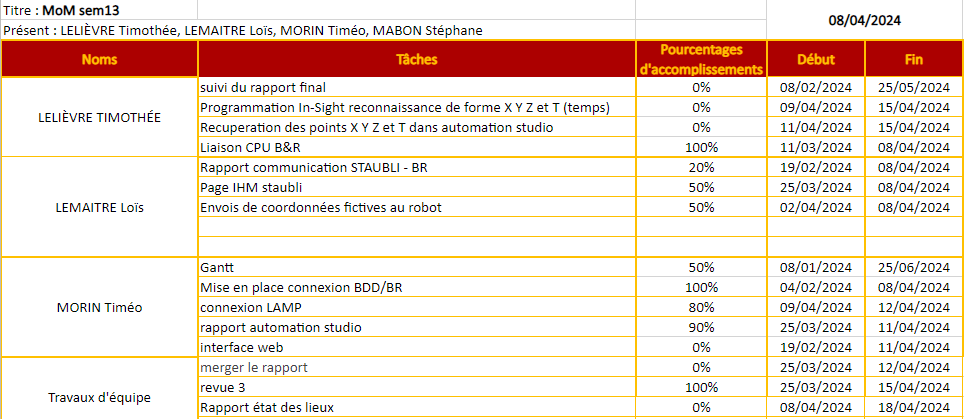
Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Ici nous retrouvons les dossiers de chaque membre ainsi que les dossier avec le github de l’année dernière

## Memory of moments

Le Memory Of Moments est un tableau que nous mettons a jour chaque semaine. Il nous permet de faire un suivi de nos taches et rend la rédaction du GANTT plus simple.



# déroulement du projet

Dans la première phase du projet j’ai passé une grosse partie de mon temps à prendre connaissance des documents laissés par les étudiants de l’année précédente ainsi que du cahier des charges.

J’ai aussi préparé mon poste de travail en installant tous les logiciels qui me seraient possiblement utiles pour la suite du projet tels que Teams, GitHub et autres.

## prise en main de SRS 2022

Une fois cette première partie de prise de connaissance des attendus du projet, j’ai commencé à apprendre à me servir de Staubli Robotic Suite qui est le logiciel de programmation du Robot TX40 que nous utilisons.

Pour ce faire J’ai utilisé la documentation laissée par les étudiants qui ont travaillés sur le projet l’année dernière ainsi que des ressources trouvées sur le web.

Globalement je n’ai pas eu de problèmes et les ressources que j’avais à disposition étaient assez claires pour m’aider à faire fonctionner le robot facilement. J’ai créé une simulation de programme pour prendre et déposer une pièce à l’aide d’un outil assez rapidement. Une fois que la simulation a marchée correctement je suis passé à la rédaction d’un rapport expliquant toutes les fonctions et outils du logiciel utilisées dans mon programme.

L’étape suivante a été de transférer mon programme dans le CS8 qui est l’ordinateur permettant de contrôler le robot. Cela s’est passé sans problème et le robot a marché comme prévu.

Suite à cela j’ai entamé la partie sur Automation Studio.

## Paramétrage d’automation studio

Afin que l’ensemble du système communique il a été chopisi d’utiliser AS (Automation Studio). Le logiciel va en effet servir à lier le robot, la caméra, la base de données et le power pannel (interface tactile programmable sur automation studio) entre eux.

La première chose que j’ai fait en créant le projet sur AS (Automation Studio) a été d’ajouter le power pannel au projet. J’ai eu une erreur au moment ou j’ai voulu me connecter au power pannel. En effet il m’était impossible de voir le pannel sur le réseau. Pour résoudre ce problème je l’ai simplement réinitialisé en utilisant une clé USB.

Après avoir réussi à paramétrer le pannel comme il se doit, j’ai cherché a importer le fichier XDD du robot qui est le fichier qui va permettre de gérer les entrées et sorties. La seconde erreur que j’ai eu a été lors de l’importation de ce fichier XDD. Le menu permettant d’importer le fichier ne répondait pas et même lorsque je compilais le projet j’avais plus d’une centaine d’erreurs.

La solution a cette erreur a été de changer de version d’AS en passant de la 4.6 a la 4.12 en même temps qu’en supprimant un dossier dans les fichiers d’AS qui générait des erreurs.

La dernière erreur que j’ai eu ensuite avec automation studio a été la liaison physique entre le power pannel et le CS8 du robot. La liaison powerlink n’était tout simplement pas paramétrée sur le bon nœud ce qui a été assez vite fixé par la suite.

## Premiers tests

Une fois qu’automation studio et SRS ont été paramétré de la bonne manière j’ai commencé des tests de liaison entre les deux. Pour ce faire j’ai d’abord essayé de comprendre comment je pouvais faire passer une entrée du robot en TRUE via l’appui d‘un bouton sur le power pannel.

Grace aux ressources mises à ma disposition j’ai pu comprendre comment réaliser cette tâche pour finalement avoir un test concluant.

Par la suite j’ai commencé à réfléchir à une manière de stocker des coordonnées dans un tableau et de les envoyer au CS8. La solution que j’ai trouvée est de convertir les coordonnées reçues en binaire avant de les envoyer bit par bit sur les sorties du CS8 qui va ensuite les retransformer en coordonnées décimales. Etant donné que la liaison avec la caméra n’est pas encore faite j’ai fait en sorte de pouvoir entrée des coordonnées fictives sur le power pannel a des fins de test.

# Recettes

### Importation du fichier XDD

|  |  |
| --- | --- |
| Titre | Importation d’un fichier XDD |
| Objectif | Importer un fichier XDD sur le logiciel AS |
| Préconditions | Le logiciel est ouvert avec le projet et vous avez installé le fichier XDD |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N° | Démarche | Données | Comportement attendu | OK ? |
| 1 | Aller dans l’onglet « Tools » puis appuyer sur « Manage 3rd-Party Devices » |  | Une fenêtre s’ouvre |  |
| 2 | Appuyer sur le bouton « Import DTM Device(s) » |  | Une fenêtre de l’explorateur de fichier s’ouvre |  |
| 3 | Choisissez le fichier d’extension « .xdd » et appuyer sur »Ouvrir » | Prenez le fichier « staubli\_io.xdd » | Vous allez voir apparaître le fichier «.xdd » dans les propositions de la toolbox |  |
| 4 | Appuyer sur le bouton « Close » |  | Vous êtes redirigée sur le projet |  |
| 5 | Aller dans la barre de recherche de la toolbox et taper le nom de votre fichier « .xdd »  (Cela est présent dans la partie droite de l’écran) |  | Vous allez voir apparaître le fichier « .xdd » dans les propositions de la toolbox |  |
| 6 | Ensuite faites un glisser/déposer du matériel comprenant le nom de votre fichier, présent dans la toolbox, dans le visuel du projet |  | Vous devez voir votre matériel comprenant la connectique voulu |  |

|  |  |
| --- | --- |
| Test réalisé par : | Réalisé le : |
| Commentaire : | Approbation : |

### Prépartation du robot

|  |  |
| --- | --- |
| Titre | Préparer le robot |
| Objectif | Démarrer le TX40 afin qu’il soit opérationnel |
| Préconditions | Le robot est bien câblé au réseau |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N° | Démarche | Données | Comportement attendu | OK ? |
| 1 | Mettre autotransfo, transfo et général pc sur "On" |  | Les disjoncteurs sont fermés |  |
| 2 | Mettre l’interrupteur du PC5 sur "On" |  | Le disjoncteur est fermé |  |
| 3 | Mettre la poignée bleue de mise en air à l’horizontale |  | Les manos affichent une pression de 6 bars |  |
| 4 | Allumer le système chaine d’embouteillage |  | On peut voir l’écran de la chaine s’allumer |  |
| 5 | Lancer ARwin Start up via un double toucher sur l’écran | Attendre 2 min | La barre du bas passe au vert |  |
| 6 | Lancer MappView via un double toucher sur l’écran |  | Une page s’ouvre |  |
| 7 | Cliquer en haut a gauche sur connexion et sélectionner stéphane |  | Une représentation du système s’affiche |  |
| 8 | Réaligner les barrières de sécurité si elles sont affichées en rouge sur l’IHM |  | Toutes les barrières sont en vert |  |
| 9 | Ouvrir les logs et supprimer les erreurs éventuelles |  | Il n’y a plus de messages d’erreur dans les logs |  |
| 10 | Allumer le CS8 |  | Attendre d’arriver sur le menu principal du MCP |  |

|  |  |
| --- | --- |
| Test réalisé par : | Réalisé le : |
| Commentaire : | Approbation : |

# annexes

Présentation Staubli 12

 historique 12

 generalites 12

choix du robot/comparatif 13

Généralités robotique 15

 les differents types de bras robotisés 15

 les principaux constructeurs 15

GitHub 16

 créer un compte et s’y connecter 16

 créer un repository 17

premiere methode 17

deuxième methode 18

 ajouter des collaborateurs 19

 acceder aux fichiers sur son pc 20

connecter l’application a un repository 21

gerer des fichiers avec l’application 21

Automation studio 23

 infos générales 23

 informations sur le logiciel 23

Création d’un projet 23

Mise en place d’un système de visualisation 24

Mise en forme de la page de visualisation 26

Mise en place de la connexion avec le pp70 27

Rapport d’erreur 28

 problème de build AS 28

problème avce le 3rd-party device manager 29

solution pour les bugs 29

 Problème de connexion au cs8 30

Connexion Powerlink vers cs8 = false 30

Solution connexion 30

SRS 2022 31

 infos générales 31

 informations sur le logiciel 31

Création d’un projet 31

créer une application 34

Ajouter des positions au robot 36

Modifier la vitesse du robot 39

créer un outil 40

Ajouter un programme 45

utiliser le controleur 47

Mise sous tension du robot 49

 etapes de mise sous tension 49

mise sous tension des éléments principaux 49

Mise en service DE LA barrière immatérielle (bi) 49

Mise sous tension du controleur du robot 51

Liaison automation - robot 52

exporter le fichier xdd sur le pc 52

importer le fichier xdd dans automation studio 52

communication avec le robot 54

# Présentation Staubli

## historique

Le groupe, fondé en 1892 à Horgen sous le nom de Schelling & Stäubli, est aujourd'hui basé à Pfäffikon, en Suisse.

La société suisse s'installe en 1909 en France, dans le bassin industriel de Faverges, à proximité du lac d'Annecy en Haute-Savoie.

## generalites

Fort de ses 6.000 collaborateurs, Stäubli est présent dans 28 pays (filiales commerciales, SAV, PR) et à un réseau de distribution dans 50 pays et 4 continents. Il comprend 15 sites de production industrielle

Quatre divisions composent l'activité du groupe :

* **Electrical Connectors** développe des solutions avancées de connecteurs électriques, basées sur la technologie MULTILAM éprouvée et extrêmement fiable. Les solutions sont utilisées dans des secteurs tels que les énergies renouvelables, les applications industrielles et d’automatisation, la transmission et la distribution d’énergie, le ferroviaire, l'automatisation de soudage, les tests et mesures, les dispositifs médicaux et la mobilité électrique.
* **Fluid Connectors** est un leader dans la conception et la fabrication de systèmes de connexion rapide, Stäubli couvre les applications de connexion de tous types d’énergies fluides et électrique. Standard ou spécifiques, les gammes de produits incluent des mono-raccords, des multi-raccords et connecteurs multipolaires, des changeurs d’outils et des systèmes de changement rapide de moule. Toutes ces solutions allient performance, qualité, sécurité et fiabilité.
* **Robotics** est un acteur mondial majeur du secteur de l’automatisation industrielle, qui fournit un support d’ingénierie et technique reconnu pour son efficacité et sa fiabilité. Qu’il s’agisse de SCARA, de robots et cobots six axes ou encore de de systèmes de robot mobile et d’AGV, la haute précision et la puissance de des produits dans de nombreux domaines industriels permettent à Stäubli de relever avec ses clients le défi de l’Industrie 4.0.
* **Textile** utilise des solutions Stäubli techniquement avancées pour coordonner l’ensemble de votre processus textile, avec tout d’une seule et même source. Qu’il s’agisse de la préparation du tissage, du tissage par cadres et Jacquard, du tissage de moquettes et du tissage technique, ou encore du tricotage de chaussettes, Stäubli offre des systèmes et solutions de machines innovantes pour une production textile haut de gamme. Tirez parti du savoir-faire de Stäubli et orientez votre production textile haut de gamme avec passion.

# choix du robot/comparatif

Le robot TX40 de Stäubli est un robot industriel conçu pour une variété d'applications, y compris le déboxage de pièces. Voici une comparaison avec d'autres robots industriels pouvant être utilisés pour le déboxage :

|  |  |  |
| --- | --- | --- |
| Robot Industriel | Caractéristique clé | Prix |
| Stäubli TX40 | Haute précision, vitesse et répétabilité, programmation facile, capacité de charge utile de 2,3 kg, construction robuste pour une longue durée de vie | Environ 20 000€-30 000€ |
| ABB IRB 4600 | Capacité de charge utile de 40 kg, haute vitesse et précision, programmation intuitive, options de sécurité avancées | Environ 50 000€-70 000€ |
| Fanuc M-20iB/25 | Capacité de charge utile de 25 kg, grande plage de mouvement, haute vitesse et précision, programmation facile et intuitive | Environ 30 000€-40 000€ |
| KUKA KR 210 | Capacité de charge utile de 210 kg, grande plage de mouvement, haute vitesse et précision, options de sécurité avancées | Environ 80 000€-100 000€ |

En termes de précision, vitesse et répétabilité, le Stäubli TX40 est un robot très performant, avec une capacité de charge utile de 2,3 kg. Sa construction robuste et sa longue durée de vie sont également des avantages.

L'ABB IRB 4600 est un autre robot industriel populaire pour le déboxage, avec une capacité de charge utile de 40 kg. Il dispose également d'options de sécurité avancées pour une utilisation en toute sécurité dans des environnements industriels.

Le Fanuc M-20iB/25 est un robot industriel avec une capacité de charge utile de 25 kg et une grande plage de mouvement pour une flexibilité maximale. Sa programmation facile et intuitive en fait également un choix populaire pour le déboxage.

Le KUKA KR 210 est un robot industriel avec une capacité de charge utile de 210 kg, ce qui en fait le plus grand robot de cette liste. Il est également équipé d'options de sécurité avancées et d'une grande plage de mouvement pour une flexibilité maximale.

En fin de compte, le choix du robot industriel pour le déboxage dépendra de plusieurs facteurs tels que la capacité de charge utile requise, la précision et la vitesse nécessaires, la programmation facile et intuitive et les options de sécurité avancées nécessaires pour une utilisation sûre dans un environnement industriel. Le Stäubli TX40 est un excellent choix pour le déboxage.

En raison de la faible charge dont nous avons besoin le Stäubli TX40 nous suffit. En revanche si nous devons changer de produits pour un produits plus lourd le robot n’aura pas la capacité de le soulever. S’orienter vers un autre type de robot sera donc nécessaire.

# Généralités robotique

## les differents types de bras robotisés

Les bras robotisés sont énormément utilisés dans l’industrie mais ne sont pas pour autant tous semblables.

On retrouve deux grand types de bras robotisés :

* Les robots industriels
* Les cobots (ou robots collaboratifs)

Les robots industriels sont des robots qui visent la rapidité, la précision et la répétition. Ces robots ne font pas attention à leur environnement et sont indépendants. Ils vont juste exécuter la tache qui leur a été confiée, ce qui fait que de travailler a coté pour un Homme est dangereux.

A l’inverse, les cobots sont des robots fait pour travailler avec l’Homme. Ils n’ont pas pour vocation d’être indépendants, ou programmés pour une tache qu’ils répèteront éternellement dans leur coin. Ce type de robot est généralement équipé pour s’arrêter quand il entre contact avec l’opérateur avec qui il travaille.

## les principaux constructeurs

**Kuka**

La société KUKA figure parmi les premiers spécialistes de l’automatisation du monde. C’est un des principaux constructeurs de Cobots et de robots industriels.

**Comau**

Comau fabrique ces machines dans le but d’intégrer et d’améliorer les applications innovantes dans tous les secteurs industriels. Leurs robots exploitent aussi l’IA afin de fonctionner de manière automnome.

**Stäubli**

Stäubli ans de nombreux domaines industriels, les hautes performances des robots industriels SCARA et 6 axes, cobots, systèmes de robot mobile et AGV permettent à Stäubli et ses clients de relever le défi de l’Industrie 4.0.

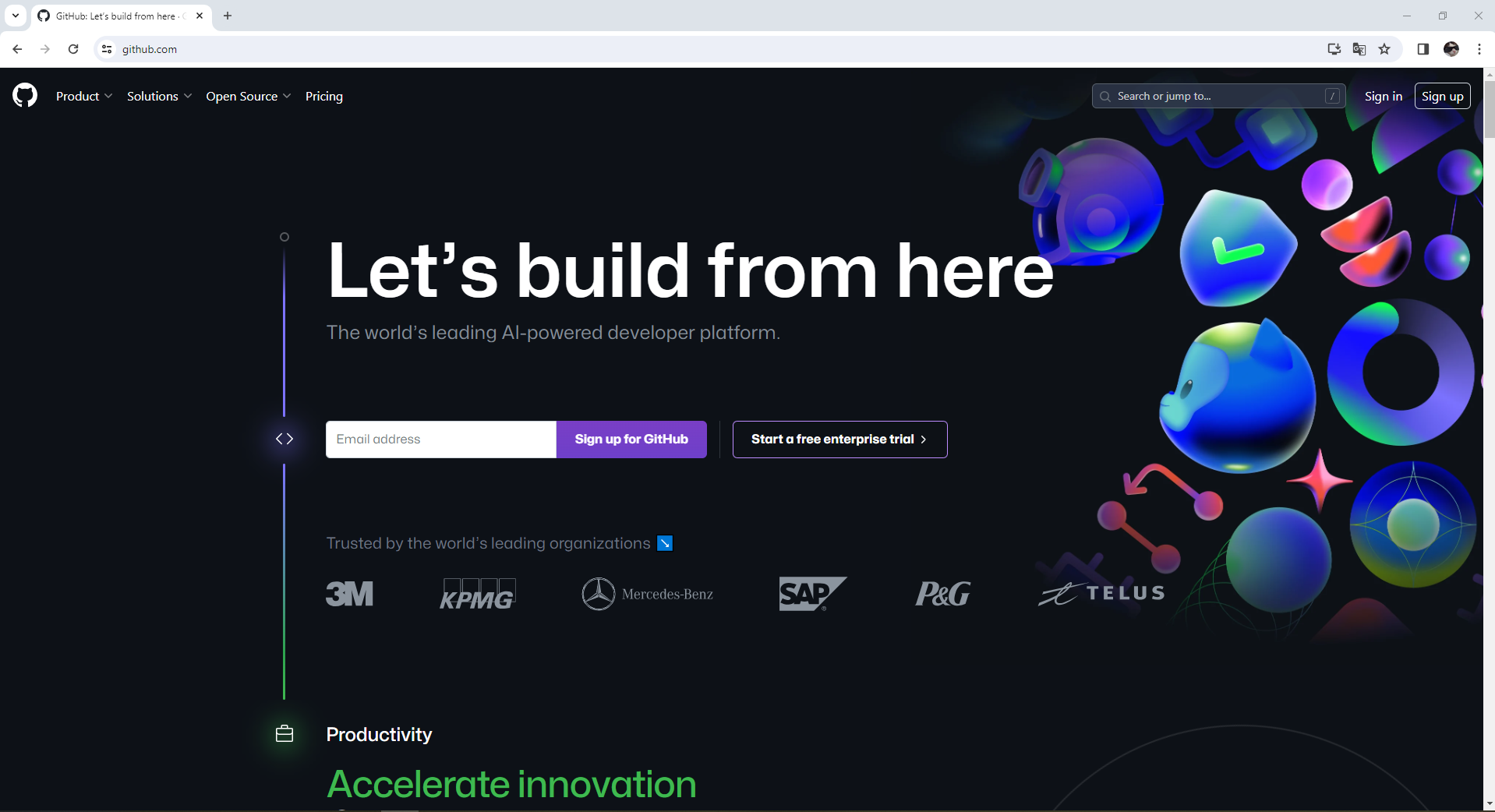
**ABB**

ABB Robotics est un pionnier de la robotique, de l'automatisation des machines et des services numériques, qui fournit des solutions innovantes pour une gamme variée d'industries, de l'automobile à l'électronique en passant par la logistique.

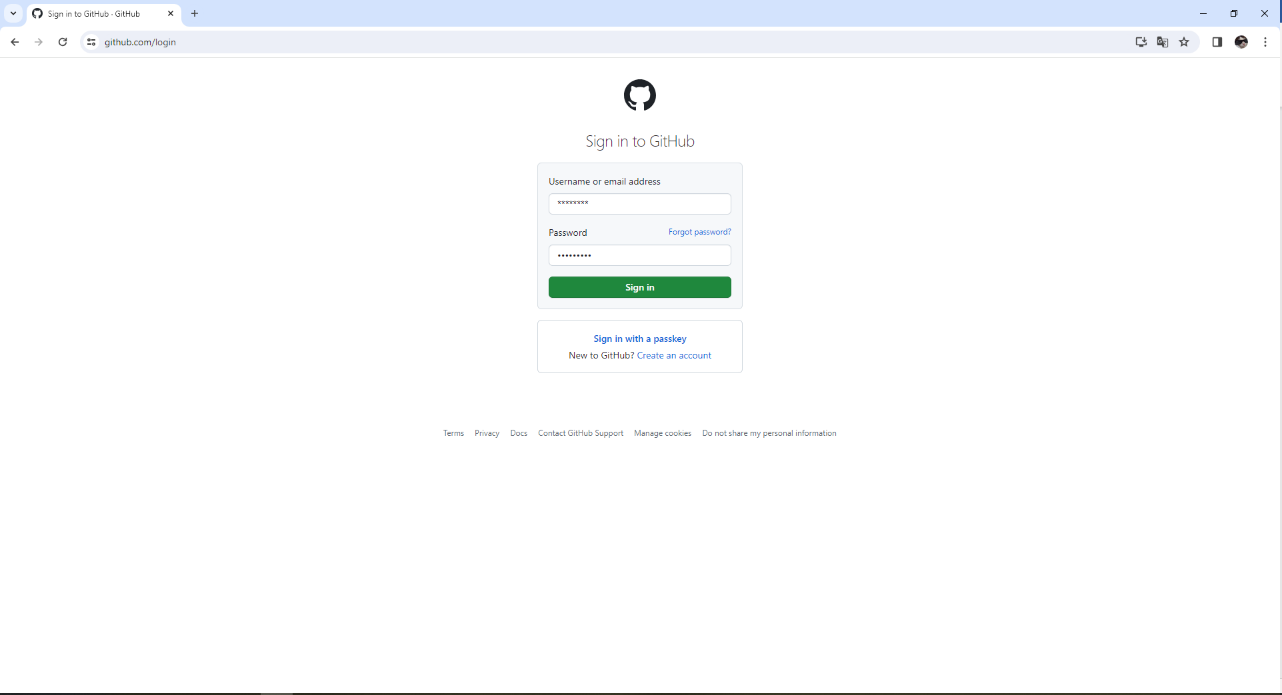
# GitHub

## créer un compte et s’y connecter

Se rendre sur <https://github.com>



Cliquer sur : « sign in » si vous possédez déjà un compte sinon cliquer sur :  « sign up » et remplir le formulaire qui vous sera demandé. Ensuite allez sur « sign in ».

La page affichée ci-dessous s’ouvrira et vous demandera de vous connecter afin d’accéder a votre compte.

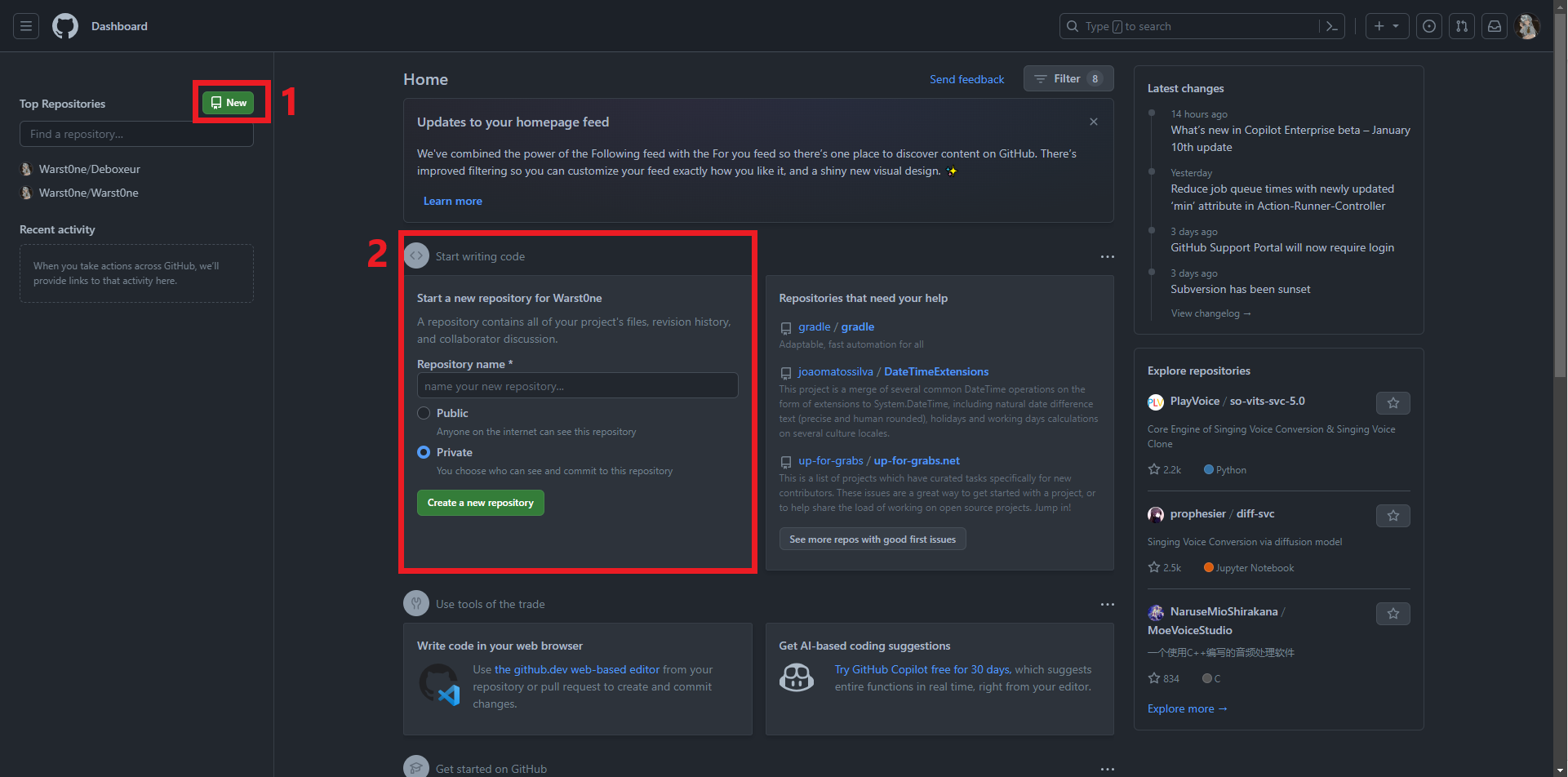
## créer un repository

Un repositories sur GitHub est un lieu permettant de rendre accessible facilement les fichiers d’un projet.

### premiere methode

Une fois connecté la page suivante s’ouvrira et vous permettra de créer une « Repositories » de deux manières différentes :

-En cliquant sur new(1) ou alors en choisissant un nom et la visibilité du repository avant de cliquer sur « create a repository »(2).



### deuxième methode

Une image contenant texte, capture d’écran

Description générée automatiquementUne image contenant texte, capture d’écran, gadget, multimédia

Description générée automatiquement

Cliquer en haut à droite sur votre image de profil (1) puis sur « Your repositories(2) »

Cliquer sur « new »

Une image contenant capture d’écran, texte, femme, Logiciel multimédia

Description générée automatiquement

Choisir un nom pour le repository (1), une description(2), sa visibilité(3) et enfin cocher la case permettant d’ajouter un fichier README(4).

Ensuite valider en cliquant sur « create repository » en bas de page

Une image contenant texte, capture d’écran, logiciel, conception

Description générée automatiquement

## ajouter des collaborateurs

Se rendre dans le dossier du repository crée et aller dans les Settings

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Puis aller dans « collaborators » et cliquer sur « Add people »

Une image contenant texte, capture d’écran

Description générée automatiquementRépéter l’opération j’jusqu’à ce que tout le monde soit ajouté.

Une image contenant texte, capture d’écran, Police, conception

Description générée automatiquement

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

## acceder aux fichiers sur son pc

### connecter l’application a un repository

Se rendre sur : <https://desktop.github.com> et installer l’application de bureau GitHub.

Une fois installé, exécuter le programme d’installation. L’lorsqu’elle sera installée, l’application se lancera automatiquement.

Cliquez sur « Clone a repository from the Internet »

Attention il faut bien avoir été invité dans repository (ou en avoir créé) pour qu’il soit affiché !

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquementUne image contenant texte, capture d’écran, Police, conception

Description générée automatiquement

Ensuite connectez-vous en cliquant sur « sign in »

Une image contenant texte, capture d’écran, logiciel, multimédia

Description générée automatiquementUne image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Après, choisissez le repository que vous avez crée

Précédemment (ici SFL3Deboxeur-24) et cliquer sur

Clone.

### gerer des fichiers avec l’application

Les fichiers du repository cloné se trouvent dans : Documents/GitHub/NomDuRepository

Pour ajouter un fichier dans le repository il suffit de l’ajouter directement dans ce dossier.

Ici en exemple le fichier exemple.txt a été créé.

Une image contenant texte, capture d’écran, logiciel, nombre

Description générée automatiquement

Ensuite dans l’application nous voyons bien que le fichier a été ajouté (1) et son contenu (3).

Pour mettre à jour le GitHub et y ajouter le fichier il suffit de cliquer sur « commit to main » (4).

Une image contenant texte, capture d’écran, Logiciel multimédia, logiciel

Description générée automatiquement

# Automation studio

## infos générales

Automation Studio est un logiciel de conception et de simulation d'automatismes industriels développé par Famic Technologies. Il permet la création de schémas électriques, pneumatiques et hydrauliques, la simulation du comportement des systèmes automatisés, la programmation de contrôle pour divers équipements, et génère automatiquement des documents techniques. Utilisé pour la formation, il facilite l'intégration avec d'autres outils de conception et de gestion de projet, contribuant ainsi à améliorer l'efficacité et la fiabilité des processus automatisés industriels.

## informations sur le logiciel

### Création d’un projet

Lancez la version 4.9 du logiciel Automation Studio sur votre ordinateur.

Sélectionnez l'option pour créer un nouveau projet puis nommez le et choisissez le dossier dans lequel il sera sauvegardé.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Une image contenant texte, capture d’écran, logiciel, affichage

Description générée automatiquementSur la page suivante sélectionner : "define a new hardware configuration manually"

Sur la page suivante rechercher et sélectionner notre power panel (ici le 4PPC70.0702-20B).

Une image contenant texte, logiciel, affichage, Icône d’ordinateur

Description générée automatiquement

### Mise en place d’un système de visualisation

Une image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquementOuvrir la toolbox d’automation studio en haut a droite et sélectionner "visual components" et double cliquer sur "VC4 Visualisation". Dans la fenêtre qui s’ouvre cliquer sur suivant jusqu’à ce que la fenêtre se ferme.

Ensuite se rendre dans la Physical View et faire un clic droit sur ETH --> Configuration

Sur cette page, sélectionner dans "**VC object name**" la visu ajoutée précédemment puis enregistrer la page.

Une image contenant texte, capture d’écran, nombre, logiciel

Description générée automatiquement

Activer aussi **SNMP parameters** afin de faire en sorte que le power panel soit reconnaissable sur le réseau quand on le recherche.

Une image contenant texte, Police, ligne, nombre

Description générée automatiquement

### Mise en forme de la page de visualisation

Pour paramèter les différentes pages du control panel il suffit de se rendre dans la **Logical view,** de double cliquer sur Visu puis d’ouvrir **init\_Page** qui se trouve dans **Pages**

Une image contenant capture d’écran, texte, affichage, logiciel

Description générée automatiquement

Une image contenant texte, capture d’écran, Icône d’ordinateur, logiciel

Description générée automatiquementLe logiciel met à notre disposition juste en dessous divers outils permettant de mettre en place l’interface.

Une image contenant texte, capture d’écran, affichage, nombre

Description générée automatiquement

Exemple d’assignation d’une variable a un bouton :

Pour qu’un bouton fasse varier la valeur d’une variable lorsqu’il est cliqué il faut, sous la rubrique **"Keys"** mettre le bouton en mode **ToggleDatapoint** (bouton reste enfoncé après appui) ou **SetDatapoint** (relâchement du bouton après appui). Il suffit ensuite dans l’onglet "**Value**" de choisir la **variable** à éditer et la valeur qui sera définie au clic.

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

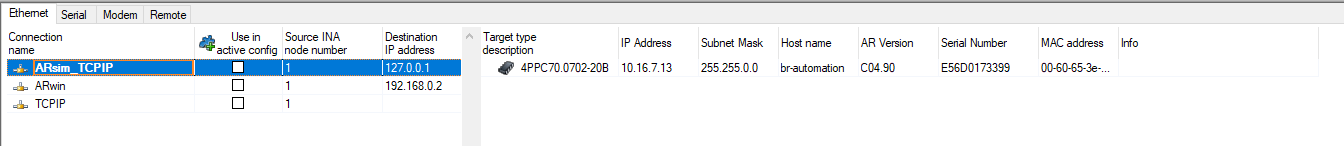
### Mise en place de la connexion avec le pp70

Pour se connecter au pp70 aller dans : **Online/Settings**

Une image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement

Une fois dedans le pp70 devrai apparaitre dans la liste des composants.



Il suffit ensuite de faire un clic droit et de se connecter au pp70.

# Rapport d’erreur

## problème de build AS

Erreurs rencontrées lors d’un build alors que le projet viens juste d’etre crée avec un power panel 4PPC70.0702-20B.

Une image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquementDe nombreuses erreurs apparessent dans la console.

Une image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquement

Ver : 4.6.3.55 SP

Automation runtime : B4.62

Mapp Services : V4.62.0

### problème avce le 3rd-party device manager

Après être passé sur AS412 :

Impossibilité d’interagir avec l’interface et même de la fermer (obligation de passer par le gestionnaire des taches). Quand n’importe quel bouton du menu est cliqué rien ne se passe et un avertissement est écris dans la console.

Une image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquement

### solution pour les bugs

Une image contenant texte, Police, ligne, nombre

Description générée automatiquementAller dans C:/ProgramData/BR/*votreVersion (ici AS412)* et supprimer le fichier Harware avant de redémarrer automation studio.

## Problème de connexion au cs8

### Connexion Powerlink vers cs8 = false

Ver : 4.12.2.93

Automation runtime : B4.93

Mapp Services : V4.72.6

Une image contenant texte, capture d’écran, nombre, logiciel

Description générée automatiquement

Lorsque l’on est connectés au PP70 et en mode monitoring, dans I/O mapping le paramètre « ModuleOk » attestant de la bonne communicaton avec le robot est en FALSE.

### Solution connexion

Une image contenant texte, Police, nombre, ligne

Description générée automatiquementPour régler le problème : mettre staubli.io en ST1 et non en ST3.

Cela va changer le nœud sur lequel se trouve le CS8

# SRS 2022

## infos générales

Le logiciel SRS (Stäubli Robotics Suite) est un logiciel développé par Stäubli fonctionnant sous Windows dédié au développement et au débogage de systèmes robotisés dans un environnement 3D.

Pour appendre le logiciel, je me suis appuyé sur les rapports de l’année dernière ainsi que sur d’autres ressources en ligne. Le logiciel a fait l’objet d’une mise a jour majeure par rapport a l’année dernière (passé de la version 2013 a la version 2022) ce qui fait que certaines ressources récupérées ne sont plus à jour.

## informations sur le logiciel

**Attention les explications qui von suivre ont été réalisées sur la version 2022 de SRS. Aussi l’agencement des fenêtres a été modifiée et pourrai ne pas être identique au votre.**

### Création d’un projet

1. Lancer le logiciel SRS 2022
2. Cliquer sur « nouvelle cellule »
3. Choisir **assistant nouvelle cellule**

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

1. Nommer la cellule
2. Sélectionner ajouter un contrôleur local
3. Une image contenant prise

   Description générée automatiquementUtiliser la barre de recherche à droite pour sélectionner le **TX40**. Attention à ne pas prendre le TX2\_40. Bien prendre le modele ci-dessous.
4. Une image contenant texte, capture d’écran, affichage, nombre

   Description générée automatiquementDans l’onglet version, **sélectionner la version s7.4** (c’est celle du contrôleur que nous utilisons). **Cocher** aussi **les cases pour BIO et BIO2/MIO** juste en dessous.
5. Cliquer sur « **continuer** » puis sur « **terminer** » j’jusqu’à arriver sur l’interface du logiciel.
6. Ajouter un objet
7. Faire un clic droit n’importe où dans la **vue** **3D** et sélectionner le **catalogue**.

Une image contenant texte, capture d’écran, ligne, diagramme

Description générée automatiquement

1. Une fenêtre **catalog** s’ouvre sur la droite de l’écran avec plusieurs catégories

Une image contenant capture d’écran, texte, Tracé, diagramme

Description générée automatiquement

1. Pour ajouter un objet a la scène il suffit de le **glisser** depuis le **catalog** jusqu’a la scène à l’endroit souhaité.

### créer une application

Afin de pouvoir contrôler les différents comportements du robot pour un cas particulier il va falloir y ajouter une application.

Dans la fenêtre cellule, faire un clic droit sur le contrôleur et sélectionner « **nouvelle** **application** ».

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

Nommer l’application et appuyer sur « **OK** »

Une image contenant texte, capture d’écran, Police, ligne

Description générée automatiquement

Une image contenant texte, capture d’écran, Police, logiciel

Description générée automatiquementL’application a bien été crée et se trouve dans l’arborescence.

### Ajouter des positions au robot

#### Avec jointRX

Dans la cellule, faire un clic droit sur l’application puis cliquer sur « **ajouter** » et enfin sélectionner « **nouvelle** **donnée** ».

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

Une image contenant texte, capture d’écran, affichage, logiciel

Description générée automatiquementSélectionner JointRx puis cliquer sur « **OK** ».

Une image contenant texte, capture d’écran, affichage, logiciel

Description générée automatiquementLe Joint a bien été créé. Il permet de contrôler indépendamment les articulations du robot avec **une** **valeur pour chaque axe**.

#### Avec PointRX

Une image contenant texte, Appareils électroniques, capture d’écran, affichage

Description générée automatiquementRefaire comme au début de la méthode pour ajouter un JointRx mais a la place sélectionner le PointRx.

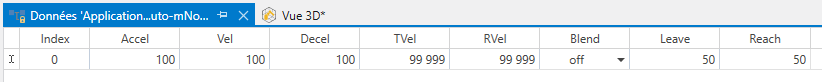
PointRx permet de déplacer le robot à partir d’une position cartésienne.

### Modifier la vitesse du robot

Le paramètre permettant de modifier la vitesse se trouve dans mdesc.

Une image contenant texte, capture d’écran, logiciel, affichage

Description générée automatiquement

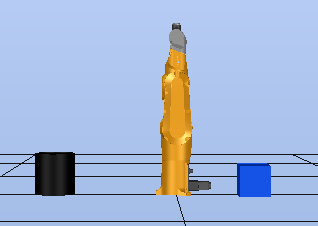
Dans ce fichier nous pouvons modifier la **vitesse**, **l’accélération** et la **décélération**.

Il est possible de créer plusieurs fichiers permettant de gérer la vitesse afin que le robot puisse s’adapter à différentes situations.

### créer un outil

Pour créer un outil il va falloir d’abord avoir un objet que nous allons définir come outil. Cet objet peut être une forme quelconque ou alors un modèle importé. Dans la démarche écrite ci-dessous je vais montrer comment ajouter un outil de type ventouse.

Tout d’abord il faut ajouter un cylindre qui va nous servir d’outil. (Pour ajouter un cylindre se référer à la partie « ajouter un objet ».



Une image contenant ciel, dessin humoristique, plein air

Description générée automatiquementFaire un clic droit sur le cylindre et cliquer sur **propriétés**. Régler le **TopRadius** a 4mm, le **BottomRadius** a 20mm et la **Length** a 80mm. Le résultat devra donner l’image ci-dessous.

Faire un clic gauche sur le cône pour le sélectionner et cliquer sur **l’icône** **create** et sélectionner **Tool**. Le cône est maintenant défini comme un outil.

Une image contenant texte, capture d’écran, Police, conception

Description générée automatiquement

Deux repères, appelés **Handle** et **Tcp**, sont associés au cône (déclaré en tant qu’outil).

Une image contenant outil, conception

Description générée automatiquementLe repère **Handle** va permettre d’attacher l’outil à la flasque du robot, ce qui sera le cas lorsque ce repère coïncidera avec le repère **Handler** du robot (à savoir, le repère associé à la flasque du robot), voir la figure qui suit.

Une image contenant texte, conception

Description générée automatiquementPour éditer la position du **Handle** et du **Tcp** il faut selectionner le cône et cilquer sur l’icone  **Edit Component.**

Cela va ouvrir une fenetre avec une vue du composant avec les point **Tcp** et **Handle** qui sont mis en évidence.

Une image contenant capture d’écran, ciel, cerf-volant

Description générée automatiquement

Une image contenant texte, diagramme, conception, origami

Description générée automatiquementIci les deux repères sont bien paramétrés mais il est toujours possible de les modifier dans l’onglet **Editer le repère de référence**.

Une image contenant ciel, cône, plein air, bouteille

Description générée automatiquementPour attacher cet outil a robot , Cliquer sur **Magnet mode** et faire un drag and drop en gilssant **le point d’origine** de l’outil vers la flasque du robot.

Vous pouvez vérifier que l’outil est bien attaché a la flasque en bougeant manuellement le robot.

Il reste à indiquer que l’outil est une ventouse. Pour cela, après avoir sélectionné l’outil, appuyer sur l’icône Une image contenant texte, logo, Police, Graphique

Description générée automatiquement, puis cliquez sur , ce qui va permettre à l’outil de se comporter comme une ventouse.

Maintenant dans la fenêtre **propriétés** des caractéristiques propres a une ventouse sont normalement apparues notamment dans la section « **Grasp** », la distance de saisie (**GraspDistance**), par défaut égale à 200 mm, qui correspond à la distance maximale de saisie de la ventouse. On retrouve aussi « **ActionSignal** » qui permet d’activer l’outil lorsqu’il est appelé.

Pour mettre en place une action pour l’outil, cliquer sur les 3 points à droite de ActionSignal et sélectionner **valve1** sous **UserIO/Sorties digitales/valve1**. Ensuite bien être sur que le carré bleu à droite des 3 points est bien sur le 1 et non le 0.

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquementEnsuite faire un clic droit sur votre **application** puis sélectionner Ajouter puis **Nouvelle donnée.**

Dans la fenêtre ci-dessous, sélectionner **tool** puis appuyer sur **OK**.

Une image contenant texte, Appareils électroniques, capture d’écran, affichage

Description générée automatiquement

Ensuite double cliquer sur l’outil crée et **mettre le Z a 80** ce qui correspond à la hauteur de l’outil.



Et voilà votre outil est paramétré et prêt à être utilisé.

### Ajouter un programme

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquementPour ajouter un nouveau programme au robot, faire un clic droit sur votre **application** puis sélectionner **Ajouter** et enfin **Nouveau programme**.

Ensuite, nommer votre programme et cliquez sur **OK.**

Vous devriez trouver la même fenêtre avec a droite au-dessus de **start()** et **stop()** le nom du programme que vous avez créé.

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

Pour faire faire un mouvement basique au robot on utilise la fonction "**movej**" avec comme paramètres (**les coordonnées** (on peut utiliser JointRx ou PointRx), **l’outil utilisé**, la **vitesse de déplacement** du robot).

**Attention a bien utiliser "moveJ" pour un joint et "movel" pour un point.**

Pour contrôler l’ouverture ou non de l’outil nous utiliseront les variables **open()** pour l’enclencher et **close()** pour le mettre au repos.

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquementOn mettra toujours **waitEndMove()** a chaque fins de programme pour être sur que le robot a bien fini son mouvement.

Une image contenant texte, Police, capture d’écran, algèbre

Description générée automatiquement

**Attention il faudra bien penser à enregistrer son programme a chaque modification !**

Sur l’exemple ci-dessus **pPointRx** correspond aux coordonnées visées, **tTool** à la ventouse que nous avons installée et **mNomSpeed** a la vitesse du robot.

Enfin pour faire en sorte que notre programme soit exécuté il faut ajouter la fonction "**call"** **suivie du nom** de notre programme dans le programme **start().**

Une image contenant texte, Police, capture d’écran, ligne

Description générée automatiquement

### utiliser le controleur

Le contrôleur sert à **exécuter** les programmes que nous avons fait. Ici est explique comment faire une **simulation** de ce contrôleur.

Pour démarrer une simulation il faut se rendre dans le panel "**Control Pad**" et mettre la vitesse en dessous de configuration 1 à **40ms.**

Une image contenant texte, capture d’écran, Police, ligne

Description générée automatiquement

Ensuite il suffit de cliquer sur la **flèche verte** qui se trouve à droite et de démarrer la simulation du contrôleur. Pour ce faire il suffit de faire un clic droit sur le contrôleur et de sélectionner "**Afficher l’émulateur**".

Une image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement

Pour démarrer le robot appuyer sur  pour que le voyant devienne **vert**. Cela indique que le robot est démarré. Ensuite appuyer sur  jusqu’à ce qu’il se trouve sur la position **automatique**.

Pour naviguer dans les menus il faut utiliser les **flèches directionnelles** ainsi que les touches **F1, F2, F3, etc** et **entrée**.

Pour sélectionner un programme naviguez dans **Application Manager/ Val3 Applications/ +Disks/ NomDeVotreProgramme Manual**. Vous devriez voir une fenêtre similaire d’affichée avec a la place de **"+ApplicationTuto"** le nom de votre programme.

Une image contenant texte, capture d’écran, logiciel, nombre

Description générée automatiquement

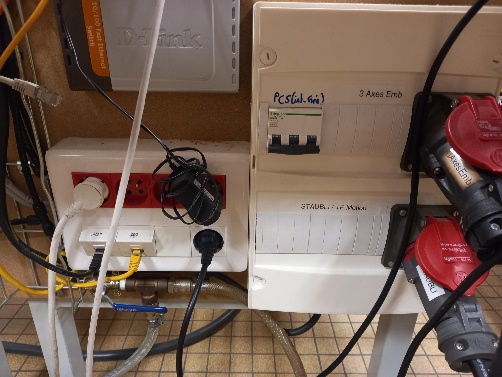
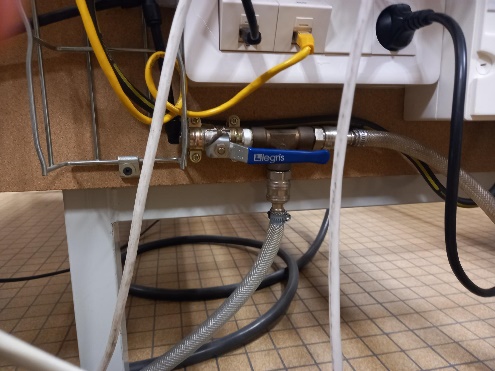
Maintenant nous pouvons lancer le programme. Pour cela appuyez sur **"run"** puis sur **" move"**.

Vous devriez voir le robot bouger sur la simulation 3D.

# Mise sous tension du robot

## etapes de mise sous tension

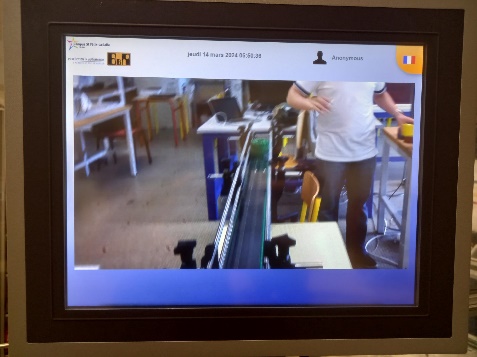
### mise sous tension des éléments principaux

1. Mettre sous tension la cellule robotique en 400V : **Mettre le Disjoncteur PC5 en ON.**
2. Mise en air :

* Source : Mettre la poignée bleue a l’horizontale
* Robot staubli : vérifier sur le mano dédié « 6bars » minimum et le bouton rotatif rouge sur on.
* Robot carthésien : vérifier sur le mano dédié « 6bars » minimum.

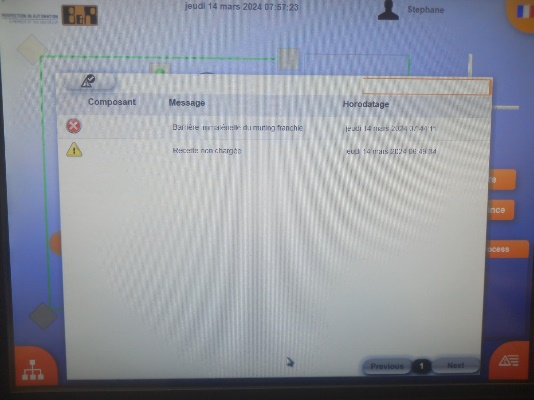
### Mise en service DE LA barrière immatérielle (bi)

1. Mettre sous tension le système « chaine d’embouteillage » avec le **sectionneur ON/OFF** sur **ON**
2. Lancer sur l’IHM B&R : **ARWin Start up** par un double touché sur l’écran (attendre 2min) / Attendre que la barre passe au jaune puis au vert. Fermer ensuite le menu.



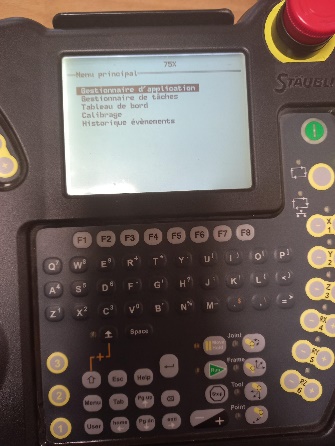
1. Lancer sur l’IHM B&R : **mappView** par un double touché sur l’écran :
2. Après l’identification : STEPHANE (sans mot de passe), vérifier la mise en place des BI avec la présence de lignes verte sur l’IHM du système. Dans le cas contraire, **des lignes rouges nous informent d’une mauvaise orientation des BI**. Dans ce cas il faut les régler manuellement pour qu’elles passent au vert.

Vérifier aussi que tous les boutons d’arrêt d’urgence soient désactivés.



1. Reseter les messages d’erreurs du système sur l’IHM avec les **touches tactiles appropriées**.

### Mise sous tension du controleur du robot

Tourner le sélectionneur du CS8 en position ON puis Attendre l’apparition du menu principal du MCP

**Vérifier les indicateurs visuels de fonctionnement :**

⮱ WMS : 1 LED sur 3 allumée avec la couleur verte fixe

⮱ MCP : LEDs clignotantes

⮱ Attendre que les leds s’éteigne ou arrête de clignoter

⮱ CS8 : Attendre que l’afficheur 7 segments Affiche (Doc CS8 p226)

❑ « **c** » : arrêt d’urgence enclenché

⮱ BI de la cellule robotique à Mettre en service (voir ci-dessous 4))

❑ « **U** » : Mémorisation AU, s’annule à la prochaine mise sous tension du bras

❑ «  **.** »  : **point clignotant = système OK et sous puissance**

❑ « **I .** » : Petit trait verticale et **point clignotant = système OK** sans la puissance ?



# Liaison automation - robot

### exporter le fichier xdd sur le pc

Tout d’abord pour importer le fichier .xdd il faut que le robot soit allumé.

1. Lancer le logiciel SRS 2022
2. SRS : Sélectionner l’application / onglet Accueil (Home) / IO physiques
3. SRS : (onglet Général /) Ajouter une carte I/O
4. SRS : Choisissez le protocole Powerlink / OK
5. SRS : Sélectionner « Hilscher »
6. SRS : Ajouter un module
7. SRS : Choisissez le nombre d’entrées et de sorties digitales souhaités.
8. Maintenant nous devons récupérer le fichier xdd dans le CS8, nous ferons ça avec FileZilla.

Dans FileZilla : Hôte : **ftp://10.16.7.18/usr**

Identifiant : **default**

Une image contenant texte, capture d’écran, logiciel, Police

Description générée automatiquement Laisser les autres champs vide

1. Dans le CS8, dans le dossier **usr/configs,** il y a le fichier **staubli\_io.xdd**
2. Transférez-le dans le dossier de votre choix, ça y est vous l’avez !

### importer le fichier xdd dans automation studio

A screenshot of a computer

Description automatically generated

1. Aller dans l’onglet **Tools / Manage 3rd Party Device / Import Fieldbus Device**, sélectionner le fichier .**xdd** puis ouvrir.
2. Aller dans la physical view
3. Sélectionner PLK (Powerlink)
4. Dans la partie droite de l’écran, dans le Catalog, double cliquer sur le fichier .**xdd**
5. A screenshot of a computer

   Description automatically generatedRésultat :

#### configuration de la carte staubli.io

Bien mettre le mode **supervised** = **off** dans **l’onglet Confi**g (clic droit sur la carte **staubli/ Config**)

Une image contenant texte, capture d’écran, nombre, logiciel

Description générée automatiquement

### communication avec le robot

Afin de pouvoir communiquer avec le robot j’ai utilisé les IO du powerlink.

Avant de pouvoir communiquer avec le robot il faut se connecter au power pannel via le menu **Online/Settings**.

Une fois en mode **Monitor** il est possible de changer les valeurs physiques. Pour ce faire, il faut entrer une valeur dans le **force activated** (ici true) et l’activer. Cela fera que la sortie sélectionnée prendra la valeur définie.

Une image contenant texte, logiciel, Icône d’ordinateur, Logiciel multimédia

Description générée automatiquement