**SPÉCIFICATIONS**

**Projet IEC61499 – Universal Automation**

*Formation Informatique et Systèmes Intelligents Embarqués*

**Année 2025 – 2026**

PRI 5A

Membres de l’équipe :

Damien LORIGEON – Chef de projet/Dev IEC61131 & IEC61499

Client :

Jean Paul CHEMLA – Professeur Polytech

Arthur OUSSOUNKIRI ELIEZER GAMBO – Doctorant Université de Reims

Bernard RIERA – Professeur Université de Reims

Stéphane LECASSE – Professeur Université de Reims

Auteur : Damien LORIGEON

Version 1.0 - 18/09/25

**Objectifs**

Ce document a pour objectif de définir les spécifications techniques et fonctionnelles du projet de preuve de concept (POC) visant à comparer deux approches de programmation d’automatismes IEC 61131-3 (ECE) et IEC 61499 (EAE) appliquées à une usine virtuelle de tri de pièces sous Factory I/O, en mettant l’accent sur : difficulté de programmation, méthode de construction de la solution, traduction du GRAFCET en 61131 puis en 61499, maintenance (messages, états bloquants, vérification, maintien), capacité d’explication du programme, et flexibilité (besoin de reprogrammer ou non lors d’évolutions).

**Référence**

1. Internes

|  |  |  |
| --- | --- | --- |
| Référence : | Titre | Lien |
| PRI Polytech Tours | Projet IEC61499 – Universal Automation | Dépôt GIT : <https://github.com/ElDLOR/PRI-IEC61499-UniversalAutomation.git>  Équipe Teams :  [Général | Projet - Universal Automation - IEC 61499 | Microsoft Teams](https://teams.microsoft.com/l/team/19%3A0k-Ai5cnEyKJAbhWmfV7GzSCosQn2K5ylgHnzRoMuGc1%40thread.tacv2/conversations?groupId=52921951-777d-4c26-b5df-34c505d96857&tenantId=16150599-ebb0-4fcf-94a5-6010823c7bd5) |

2. Externes

|  |  |  |
| --- | --- | --- |
| Référence : | Titre | Lien |
| UniversalAutomation.org | Ressources IEC 61499 / EAE | <https://universalautomation.org> |

**Définition**

* **IEC 61131** : norme historique de programmation des automates (POU, Grafcet, ST, etc.).
* **IEC 61499 (EAE)** : norme orientée événements, blocs fonctionnels distribués.
* **Factory IO** : outil de simulation 3D de systèmes industriels.
* **EAE** : EcoStruxure Automation Expert.
* **ECE** : EcoStruxure Control Expert

SPÉCIFICATIONS

« Projet IEC61499 – Universal Automation »

|  |  |  |
| --- | --- | --- |
| Descriptions | | |
| Projet : | Projet IEC61499 – Universal Automation | |
| Clients | Jean Paul CHEMLA | jean-paul.chemla@univ-tours.fr |
| Auteurs | Damien LORIGEON | damien.lorigeon@univ-tours.fr |
| Date d’émission : | 19/09/2025 | |

|  |  |  |  |
| --- | --- | --- | --- |
| Validation | | | |
| Nom | Date | Valide (O/N) | Commentaires |
| LORIGEON | 02/10/2025 | O |  |
| CHEMLA |  |  |  |
| RIERA |  |  |  |
| GAMBO |  |  |  |

|  |  |  |
| --- | --- | --- |
| Suivis des versions | | |
| Version | Date | Description de la modifications |
| 1 | 02/10/2025 | Première version |
|  |  |  |

Sommaires

[1 Architecture générale 7](#_Toc211454923)

[2 Exigences fonctionnelles 7](#_Toc211454924)

[3 Exigences techniques 7](#_Toc211454925)

[4 Conception logicielle 7](#_Toc211454926)

[5 Validation et tests 7](#_Toc211454927)

# Périmètre fonctionnel

L’étude repose sur une usine virtuelle de tri de pièces modélisée dans Factory I/O.

Cette installation représente une chaîne de production automatisée permettant de détecter, acheminer et trier des pièces selon un critère précis.

Le système doit assurer la circulation séquentielle des pièces, la prise de décision en fonction du type détecté, et la distribution vers l’une des trois sorties prévues.

Les séquences d’exploitation incluent également les fonctions classiques d’un cycle industriel : mise en marche, arrêt, réinitialisation et gestion d’arrêts d’urgence.

Les particularités de fonctionnement de l’usine sont les suivantes :

* Tri par couleur (3 couleurs différentes). La ligne traite une pièce à la fois, sauf lorsqu’il y a deux pièces consécutives de même couleur : dans ce cas, les deux sont acheminées simultanément vers la même sortie.
* La scène Factory I/O utilisée dans le projet existe déjà. Elle ne nécessite aucune reconstruction ; seule la logique de commande sera développée dans les environnements Ecostruxure Control Expert (IEC 61131-3) et Ecostruxure Automation Expert (IEC 61499).

# Architecture

Contraintes et choix figés pour la spec :

* Tout sur le même PC (salle AUTO1 du DMS), accès possible via VPN.
* EcoStruxure Simulation utilisé comme Soft PLC.
* Modbus TCP imposé.
* Rôle Modbus : API (Soft PLC) = serveur, Factory I/O = client.

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

Le contenu généré par l’IA peut être incorrect.

# Critères de comparaison

**Difficulté de programmation**

Ce critère évalue la facilité de prise en main et la complexité de mise en œuvre de chaque méthode.

L’analyse portera sur la lisibilité du code ou des blocs fonctionnels, la cohérence de la structure logicielle, ainsi que la clarté de la logique séquentielle (GRAFCET ou équivalent).

On cherchera à déterminer dans quelle mesure un développeur peut comprendre, modifier et maintenir le programme sans ambiguïté.

**Méthode de construction**

Ce critère évalue dans quelle mesure le même GRAFCET peut être utilisé de façon intuitive pour programmer sous chaque norme.

En IEC 61131-3, la logique séquentielle du GRAFCET s’intègre naturellement grâce au langage SFC, rendant la transposition directe et simple.

En revanche, en IEC 61499, l’approche évènementielle par blocs fonctionnels rend cette correspondance moins évidente. Il s’agira donc d’analyser si le GRAFCET reste pertinent ou s’il faut adopter d’autres méthodes, comme une conception basée sur les évènements, les états ou une décomposition modulaire du système.

**Maintenance et diagnostic**

Ce critère porte sur la gestion des anomalies, des messages d’erreur et des états bloquants.

L’objectif est d’évaluer la capacité du programme à être maintenu dans un contexte industriel :

* Identification claire des fautes ou blocages,
* Gestion des messages d’alarme et de reprise,
* Visibilité des états internes et procédures de débogage.

La norme facilitant le suivi des états, la traçabilité des évènements, et la réutilisation des blocs de diagnostic sera considérée comme plus performante dans ce domaine.

**Capacité d’explication**

Ce point évalue la clarté pédagogique et la transparence du design logiciel.

Il s’agit de déterminer dans quelle mesure le programme peut être expliqué et compris par un tiers (technicien, ingénieur, enseignant).

On observera notamment :

* La structure hiérarchique du projet,
* La nomenclature des blocs fonctionnels,
* La cohérence graphique et logique du schéma global.

Une approche plus intuitive, modulaire et auto-documentée sera considérée comme plus adaptée à l’enseignement et à la maintenance.

**Flexibilité**

Ce critère mesure la capacité du système à s’adapter à des modifications sans nécessiter une refonte complète du programme.

Les cas étudiés incluront :

* L’ajout de nouveaux capteurs ou actionneurs,
* Le changement de règles de tri ou de comportements,
* La migration du projet vers un autre contrôleur.

# Conception logicielle

# Validation et tests