

Ecole Nationale Polytechnique



المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique



Département d'Électrotechnique

Projet de Fin d'Études

Pour l'Obtention du Diplôme d'Ingénieur d'État en Électrotechnique

Étude et Régulation d'un Groupe Électrogène avec Supervision via un Système SCADA sous AGI Creator

Réalisé par :
Mme. AIT OUMEGHAR Meriem
Mme. ROUABAH Hadjer

Encadré par :
M. TADJINE Mohamed
M. BELHADDAD Abdellah

Soutenu le 8 Juillet 2025, Devant le jury composé de :

M. TEGUAR MADJID : ENP - Président
M. SEBAA KARIM : ENP - Examineur

ENP 2025

Ecole Nationale Polytechnique



المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique



Département d'Électrotechnique

Projet de Fin d'Études

Pour l'Obtention du Diplôme d'Ingénieur d'État en Électrotechnique

Étude et Régulation d'un Groupe Électrogène avec Supervision via un Système SCADA sous AGI Creator

Réalisé par :
Mme. AIT OUMEGHAR Meriem
Mme. ROUABAH Hadjer

Encadré par :
M. TADJINE Mohamed
M. BELHADDAD Abdellah

Soutenu le 8 Juillet 2025, Devant le jury composé de :

M. TEGUAR MADJID : ENP - Président
M. SEBAA KARIM : ENP - Examineur

ENP 2025

ملخص

يتناول هذا العمل دراسة تحسين معلمات PID لتنظيم سرعة مجموعة توليد كهربائي، باستخدام تقنيات متقدمة مثل تحسين بواسطة أسراب الجسيمات (PSO) والخوارزميات الجينية (GA). تم أيضًا تطوير نظام SCADA باستخدام بروتوكول RTU Modbus للتواصل، لتمكين الإشراف في الوقت الفعلي مع واجهة رسومية تم إنشاؤها باستخدام AGI Creator. أظهرت النتائج التي تم الحصول عليها تحسنًا ملحوظًا في التنظيم، مما يضمن أداءً مستقرًا وفعالًا، حتى في مواجهة تقلبات الحمل، مع تطبيقات محتملة سواء لأنظمة متصلة بالشبكة أو للأنظمة المستقلة في المناطق المعزولة.

الكلمات المفتاحية : تحسين PID، تحسين بواسطة أسراب الجسيمات (PSO)، الخوارزميات الجينية (GA)، معيار الحد الأدنى للخطأ ITAE، SCADA، RTU، Modbus، AGI Creator.

Abstract

This work investigates the optimization of PID parameters for speed regulation of a diesel generator, using advanced techniques such as Particle Swarm Optimization (PSO) and Genetic Algorithms (GA). A SCADA system, utilizing the Modbus RTU protocol for communication, was also developed to enable real-time supervision with a graphical interface created in AGI Creator. Experimental results demonstrated a significant improvement in regulation, ensuring stable and responsive performance even with load variations. This system shows great potential for industrial applications, whether connected to the grid or operating autonomously in remote areas.

Keywords : PID Optimization, PSO, GA, Minimal Error Criterion ITAE, SCADA, AGI Creator, Modbus RTU.

Résumé

Ce travail explore l'optimisation des paramètres PID pour la régulation de la vitesse d'un groupe électrogène, en utilisant des techniques avancées telles que l'Optimisation par Essais Particuliers (PSO) et les Algorithmes Génétiques (GA). Un système SCADA, utilisant le protocole Modbus RTU pour la communication, a également été développé afin de permettre une supervision en temps réel, avec une interface graphique réalisée sous AGI Creator. Les résultats obtenus montrent une amélioration notable de la régulation, garantissant une performance stable et efficace, même face aux variations de charge, avec des applications potentielles tant pour les systèmes connectés au réseau que pour les installations autonomes en zones isolées.

Mots clés : Optimisation PID, PSO, GA, critère d'erreur minimale ITAE, SCADA, AGI Creator, Modbus RTU.

REMERCIEMENTS

Avant toute chose, nous rendons grâce à Allah le Tout-Puissant qui nous a accordé la force, la volonté et la persévérance nécessaires pour mener à bien ce projet.

Nous tenons à adresser nos remerciements à notre encadrant **M. TADJINE Mohamed**, pour avoir accepté de superviser ce travail et pour l'attention qu'il y a portée. Sa rigueur et son sens de l'organisation ont contribué au bon déroulement de ce projet.

Nous exprimons également notre profonde gratitude à **M. BELHADDAD Abdelah**, co-encadrant de ce mémoire, pour son implication continue, sa disponibilité, ses conseils précieux et ses encouragements constants. Son accompagnement humain et scientifique a été d'une aide inestimable à chaque étape de cette étude.

Nous remercions également les membres du jury pour l'honneur qu'ils nous font en acceptant d'évaluer ce travail et pour le temps qu'ils y consacrent.

Nous exprimons notre reconnaissance à nos camarades, nos amis et toutes les personnes qui nous ont soutenus de près ou de loin. Vous êtes nombreux à avoir contribué à rendre cette aventure plus agréable et enrichissante.

Nous tenons à exprimer notre profonde gratitude à l'ensemble de l'équipe pédagogique ainsi qu'à notre chef de département, M. BOUGHRARA kamel et à l'équipe administrative de l'École Nationale Polytechnique pour la qualité de l'enseignement dispensé et leur disponibilité tout au long de notre parcours.

À toute personne ayant contribué, directement ou indirectement, à la réalisation de ce travail, nous adressons nos sincères remerciements.

DÉDICACE

Nous dédions ce mémoire:

*À nos parents — que Dieu les protège — qui nous ont
encouragés à poursuivre nos études,*

À nos frères, nos sœurs, et à toute notre famille,

*À Jedou ROUABAH Salah, qui nous a quittés le 29 juin
2025. Allah yerhamou.*

À toutes nos amies et tous nos amis,

À la promotion de génie électrique (Électrotechnique),

À tous ceux qui nous sont chers, à vous tous.

Merci.

– Meriem & Hadjer

Sommaire

Liste des Figures

Liste des Tableaux

Liste des Abréviations

Introduction Générale 18

Chapitre 1: Amimer Énergie et Aperçu de ses Groupes Électrogènes 20

1 Présentation de l'Entreprise 21

1.1 Présentation Succincte 21

1.2 Services de l'Entreprise 22

1.3 Structure Générale de l'Entreprise 23

2 Qu'est-ce qu'un Groupe Électrogène ? 24

3 Les Types du Groupe Électrogène 24

3.1 Groupes de Production 24

3.2 Groupes de Secours 25

4 Les Composants d'un Groupe Électrogène Diesel 25

4.1 Moteur Diesel 25

4.2 Alternateur 26

4.3 Dispositifs de Régulation 27

4.3.1 Le Régulateur de Tension Automatique 27

4.3.2 Le Régulateur de Vitesse 29

4.4 Système de Contrôle/Commande et Protection 30

5 Fonctionnement des Groupes Électrogènes 31

5.1 Vérifications et Opérations à réaliser avant le Fonctionnement 31

5.2	Principe de Fonctionnement	32
6	Processus de Fabrication d'un Groupe Électrogène	34
Chapitre 2: Optimisation Intelligente de la Régulation PID d'un Groupe Électrogène		35
1	Modélisation du Groupe Électrogène	36
1.1	Modèle Simplifié d'un Générateur Synchrone	36
1.2	Modèle de la Charge	37
1.3	Modèle de Moteur Diesel	38
1.4	Modèle Global du Groupe Électrogène	39
2	Structure de Régulage PID	40
2.1	Structure du Régulateur	40
2.2	Méthodes de Réglage des Paramètres PID	42
3	Synthèse des Gains PID: Approche Classique de Ziegler–Nichols	43
4	Synthèse des Gain PID: Approche Methaheuristique	44
4.1	Position du Problème d'Optimisation: Critères d'Erreur Minimales . .	44
4.2	Procédure de Réglage Optimale	45
4.3	Solutions Approche Proposées	46
5	Optimisation par Essaims Particulaires OEP	46
5.1	Principe de Fonctionnement de l'Algorithme OEP	46
5.2	Étapes de l'Algorithme PSO	48
6	Algorithme Génétique AG	48
6.1	Principe de Fonctionnement de l'Algorithme AG	49
6.2	Étapes de l'Algorithme AG	51
Simulation, Résultats et Discussion		52

7	Modèle Simulink du Groupe Électrogène Utilisé	52
7.1	Application de la Méthode Ziegler-Nichols	52
7.1.1	Calcul les Paramètres du Régulateur PID	53
7.1.2	Les Résultats de Simulation	55
7.2	Application du Critère ITAE Optimisé par PSO et GA	57
7.2.1	Formulation du Problème d’Optimisation PID par AG	57
7.2.2	Formulation du Problème d’Optimisation PID par PSO	58
8	Les Résultats du Critère ITAE avec ses Algorithmes d’Optimisation	59
9	Discussion	62
 Chapitre 3: Fondements Théoriques des Systèmes SCADA et du Pro- tocol Modbus RTU		 63
1	Présentation Générale du Système SCADA	64
1.1	Définition d’un Système SCADA	64
1.2	Éléments Constitutifs d’un Système SCADA	64
1.3	Architecture d’un Système SCADA	66
1.4	Fonctions Principales	67
2	Protocole ModBus	67
2.1	Définition	67
2.2	Variantes du Protocole Modbus	68
2.2.1	Modbus sur Ligne Série	68
2.2.2	Modbus via Wifi ou Ethernet	69
2.2.3	Extensions Avancées du Protocole Modbus	69
3	Protocole Modbus RTU	69
3.1	Définition	69
3.2	Caractéristique du Réseau Modbus RTU:	70
3.3	Mode de Communication Maître/Esclave	70

3.4	Interfaces de Communication Série de Modbus RTU	72
3.4.1	RS232	72
3.4.2	RS422	72
3.4.3	RS485	72
3.5	Structure de la Trame Modbus RTU	73
3.6	Transaction Modbus RTU	74
3.6.1	Déroulement d'une Transaction	74
3.6.2	Vérification d'Intégrité: CRC	75
3.6.3	Codes d'Exception	75
3.6.4	Codes d'Erreur Modbus RTU	76
3.7	Données Echangées via Modbus RTU	76
3.7.1	Type de Données	76
3.7.2	Codes Fonctions Modbus Courants	77
3.7.3	Codes Fonctions Modbus Avancés	78
3.8	Application Simple de la Communication Modbus RTU en Environ- nement Simulé	79
3.9	Diagnostic et Dépannage d'un Réseau Modbus RTU	79
3.9.1	Diagnostic de la Couche Hardware	79
3.9.2	Diagnostic au Niveau Logiciel	80
3.9.3	Résolution des Problèmes de Cohérence des Données	80
3.9.4	Bonnes Pratiques pour le Dépannage	80
Chapitre 4: Implémentation sous AGI Creator du Système SCADA		82
1	Présentation Générale du Logiciel AGI Creator	83
1.1	Présentation du Workspace	83
1.2	Création et Gestion d'un Projet	84
1.3	Utilisation du Panneau Project View	84
1.4	Création des Protocoles de Communication	86
1.5	Création des Tags de Communication	87

1.6	Insertion d'Objets Graphiques et Liaison avec un Tag de Communication	87
1.7	Animation Graphique des Éléments	89
1.8	Écriture dans un Tag	90
2	Simulation du Système SCADA en Environnement Virtuel	91
2.1	Rôle des Outils Logiciels dans la Simulation	91
2.2	Mise en Place de la Communication	91
2.3	Création des Tags de Communication	93
2.4	Conception de l'Interface HMI	94
2.4.1	Structure de l'interface	94
2.4.2	Liaison des Objets Graphiques aux Données	98
2.5	Exécution de la Simulation	98
2.5.1	Observation des Valeurs Simulées sur EasyModbus et AGI . .	98
2.5.2	Vérification de la Commande d'Écriture	100
3	Mise en œuvre du Système SCADA sur un Groupe Électrogène Réel	100
3.1	La Carte SGC420	100
3.1.1	Afficheur et Boutons de Commande	101
3.1.2	Menus de Surveillance et de Configuration	102
3.1.3	Entrées / Sorties	102
3.1.4	Modes de Fonctionnement	103
3.1.5	Communication	103
3.2	Mise en œuvre de la Communication réelle	103
3.2.1	Câblage Physique	104
3.2.2	Configuration du Port Série dans AGI Creator	104
3.3	Tests Fonctionnels et Validation	104
3.3.1	Commandes via l'Interface HMI	104
3.3.2	Comparaison des Valeurs Affichées	105

Conclusion Générale	107
Bibliographie	109
Annexe	112

Liste des Figures

1	Gamme de produits Amimer Énergie	21
2	Centrale diesel mobile d'AMIMER ENERGIE au Sahara	22
3	Groupe Électrogène TASSILI 750 KVA - 2000 KVA	24
4	Moteur diesel General-Electric.	26
5	Alternateur synchrone Leroy-Somer	27
6	Régulateur de tension électromécanique chromé	28
7	Régulateur de tension analogique Leroy-Somer	28
8	Régulateur de tension numérique Leroy-Somer	29
9	Gouverneur mécanique	29
10	Gouverneur électronique	30
11	Le modèle linéaire du GS avec $\Omega(s)$	37
12	Le modèle linéaire du GS avec la charge	38
13	Le modèle du moteur diesel	39
14	Le modèle global du groupe électrogène	39
15	Schéma bloc d'un régulateur PID	41
16	Système sous étude: Ziegler-Nichols	43
17	Déplacement d'une Particule.	47
18	Modèle du groupe électrogène avec ses valeurs numériques	52
19	Schéma bloc du GE pour l'application de la méthode de Ziegler-Nichols	53
20	La réponse du système avec la méthode de ZN pour différentes per- turbations	56
21	Schéma bloc du GE pour l'application du critère ITAE	57
22	Réponse du système avec un réglage optimisé par PSO pour $\Delta P_l = 1\%$	59
23	Réponse du système avec un réglage optimisé par AG pour $\Delta P_l = 1\%$	60
24	Réponse du système avec un réglage optimisé par PSO pour $\Delta P_l = 5\%$	60
25	Réponse du système avec un réglage optimisé par AG pour $\Delta P_l = 5\%$	61
26	Réponse du système avec un réglage optimisé par PSO pour $\Delta P_l = 10\%$	61
27	Réponse du système avec un réglage optimisé par AG pour $\Delta P_l = 10\%$	62

28	Mode broadcast	71
29	Mode unicast	71
30	Trame générale de Modbus	73
31	Structure trame Modbus RTU	74
32	Comparaison des trames avec et sans contrôle de la parité	74
33	Transaction Modbus sans erreur	75
34	Transaction Modbus avec exception	76
35	Interface du logiciel AGI Creator	83
36	Panneu Project View dans AGI Creator	85
37	Ajout d'un protocole de communication dans AGI Creator	86
38	Modification des paramètres du protocole dans AGI Creator	86
39	Création des tags dans AGI Creator	87
40	Panneau d'objets graphiques dans AGI Creator	88
41	Objet Multi-State Image dans AGI Creator	89
42	Animation graphique d'un interrupteur dans AGI Creator	90
43	Simulation de l'interrupteur On-Off	90
44	Création des ports COM virtuels via VSPE	92
45	Affectation de port COM1 → EasyModbus	92
46	Affectation de port COM2 → AGI Creator	93
47	Correspondance d'adressage Modbus entre la carte SGC420 et AGI Creator.	93
48	Liste des tags configurés dans AGI Creator	94
49	Interface d'accueil esthétique du projet	95
50	Page des détails électriques	95
51	Page des détails mécaniques	96
52	Page des détails de la génératrice	96
53	Page des détails mécaniques	97
54	Liste des alarmes	97
55	Activation du groupe par bouton Démarrage	98

56	Visualisation des tensions simples du générateur	99
57	Affichage de la pression d'huile moteur simulée	99
58	Carte SGC 420	101
59	Câblage de la carte SGC 420 avec le PC via un convertisseur US- B/RS485	104
60	Validation de la mesure de pression d'huile entre AGI et SGC.	105
61	Validation de la mesure de tension batterie entre AGI et SGC.	105
62	Validation des tensions génératrice entre AGI et SGC.	105
63	Synchronisation de la vitesse moteur affichée entre AGI et SGC.	106
64	Interface du logiciel Modbus Poll	114
65	Menu de connexion dans Modbus Poll	114
66	Configuration de la communication dans Modbus Poll	115
67	Communication série dans Modbus Poll	116
68	Paramètres avancés de la communication série dans Modbus Poll	117
69	Communication Ethernet dans Modbus Poll	117
70	Accès à la fenêtre Read/Write Definition dans Modbus Poll	118
71	Fenêtre Read/Write Definition dans Modbus Poll	120
72	Accès à la fenêtre Communication Traffic dans Modbus Poll	120
73	Fenêtre Communication Traffic dans Modbus Poll	121
74	Fenêtre Log Setup dans Modbus Poll	122
75	Fenêtre Excel Log Setup dans Modbus Poll	123
76	Interface du logiciel EasyModbus	123
77	Configuration de la communication dans EasyModbus	124
78	Métadonnées du simulateur EasyModbus Server	124
79	Fonctions supportées dans EasyModbus	125
80	Types des données dans EasyModbus	125
81	Protocole d'information dans EasyModbus	126
82	Interface de l'émulateur VSPE	126
83	Création d'un périphérique dans VSPE	127

84	Type des périphériques dans VSPE	127
85	Création de ports virtuels avec VSPE	128
86	Lancement d'EasyModbus RTU Server	129
87	Configuration de Modbus Poll Maître	129
88	établissement de la communication série RTU	129
89	Communication Modbus RTU utilisant la fonction 03	130
90	Communication Modbus RTU utilisant la fonction 16	130
91	Échange de trames Modbus – Fonction 03	131

Liste des Tableaux

1	Effets de L'augmentation des Gains PID sur les Performances du Système	42
2	Réglages Empiriques selon la Méthode de Ziegler–Nichols	44
3	Réponse du système pour différentes variations de charge	56
4	Les performances du système face à une faible perturbation $\Delta P_l = 1\%$	60
5	Les performances du système face à une moyenne perturbation $\Delta P_l = 5\%$	61
6	Les performances du système face à une forte perturbation $\Delta P_l = 10\%$	62
7	Comparaison des rôles du Maître et de l'Esclave dans une communication Modbus	70
8	Codes d'exception Modbus RTU	76
9	Fonctions Modbus RTU les plus courantes	78
10	Codes fonctions Modbus avancés	78
11	Types d'adressage Modbus	80
12	Rôles des différents outils dans la simulation SCADA	91
13	Correspondance des commandes AGI / Interprétation par le SGC420	100
14	Exemple d'enregistrement des trames dans le journal de communication Modbus Poll	121
15	Trame envoyée (Tx) – Fonction 03	131
16	Trame reçue (Rx) – Fonction 03	131

Liste des Abréviations

- **EPC:** Ingénierie, Approvisionnement et Mise en Service.
 - **IPP:** Producteur Indépendant d'Électricité.
 - **SCADA:** Système de Contrôle et d'Acquisition de Données.
 - **AVR:** Régulateur Automatique de Tension.
 - **PID:** Proportionnel Intégral Dérivé.
 - **RID:** Interface à Relais.
 - **PLC:** Automate Programmable Industriel.
 - **ECU:** Unité de Contrôle Électronique.
 - **AC:** Courant Alternatif.
 - **DC:** Courant Continu.
 - **MCC:** Centre de Contrôle des Moteurs.
 - **IMC:** Commande par Modèle Interne.
 - **IAE:** Intégrale de l'Erreur Absolue.
 - **ISE:** Intégrale de l'Erreur au Carré.
 - **ITAE:** Intégrale de l'Erreur Absolue Pondérée par le Temps.
 - **PSO:** Optimisation par Essaim Particulaire.
 - **GA:** Algorithme Génétique.
 - **BFGS:** Méthode Broyden–Fletcher–Goldfarb–Shanno.
 - **IEEE:** Institut des Ingénieurs Électriciens et Électroniciens.
 - **RTU:** Unité Terminale Distantée.
 - **MTU:** Unité Terminale Maître.
 - **HMI/SCADA:** Interface Homme-Machine / Supervision.
 - **ERP:** Planification des Ressources de l'Entreprise.
 - **GMAO:** Gestion de Maintenance Assistée par Ordinateur.
 - **MODBUS:** Protocole de Communication Industriel MODBUS.
 - **ADU:** Unité de Données d'Application.
 - **PDU:** Unité de Données de Protocole.
-

-
- **CRC:** Contrôle de Redondance Cyclique.
 - **VSPE:** Émulateur de Port Série Virtuel.
 - **MDI:** Interface à Documents Multiples.
 - **RTS:** Requête pour Émission.
 - **DSR:** Périphérique Prêt.
 - **CTS:** Clair pour Émettre.
 - **TCP:** Protocole de Contrôle de Transmission.
 - **TCP/IP:** Protocole de Contrôle de Transmission / Protocole Internet.
 - **EMI:** Interférences Électromagnétiques.
 - **AGI:** Intelligence Artificielle Générale.
 - **DEIF:** Société Danoise d'Électronique et d'Instrumentation (DEIF).
 - **SGC:** Single genset controller.
-

Introduction Générale

La production d'énergie est essentielle pour satisfaire une demande croissante, stimulée par l'évolution rapide de la technologie et des besoins industriels. Toutefois, cette production reste vulnérable à des contraintes imprévisibles telles que les surcharges, les défauts électriques ou les pannes techniques, conduisant souvent les responsables à recourir à des coupures d'électricité. Ces interruptions affectent particulièrement des secteurs sensibles comme les hôpitaux, les centres de données et les industries, où la continuité de l'alimentation est importante. Par ailleurs, l'implantation de centrales électriques conventionnelles dans des zones isolées ou difficiles d'accès s'avère complexe et coûteuse.

Face à ces défis, **les groupes électrogènes** émergent comme une solution alternative et simple. Ces systèmes autonomes, composés d'un moteur à combustion couplé à un générateur synchrone, permettent de produire localement une électricité de qualité. Ils remplacent efficacement les centrales dans les régions reculées ou servent de source d'alimentation de secours lors des défaillances du réseau. Cependant, l'énergie fournie par ces groupes doit répondre aux mêmes exigences de stabilité que le réseau public, notamment en termes de **régulation précise de la vitesse (fréquence)**, en particulier lors de variations de charge. Cette dernière repose sur l'**optimisation des paramètres des régulateurs PID (Proportionnel-Intégral-Dérivé)**, qui sont chargés de corriger dynamiquement les écarts de performance.

De plus, pour des applications critiques comme les centrales électriques ou les infrastructures industrielles, **la supervision et le contrôle à distance** des groupes électrogènes deviennent indispensables. Un système **SCADA (Supervisory Control and Data Acquisition)** permet de centraliser la surveillance, d'automatiser les commandes et d'anticiper les pannes via une interface homme-machine (IHM) intuitive.

Dans ce contexte, notre projet de fin d'études s'articule autour de deux axes principaux:

1. L'optimisation intelligente de la régulation de vitesse du groupe électrogène

- Après une modélisation dynamique du groupe électrogène (moteur diesel, alternateur, charge), nous avons appliqué une méthode classique de réglage PID (Ziegler-Nichols) pour définir des paramètres initiaux afin de guider cette optimisation.
- Pour affiner ces réglages classiques et garantir une stabilité optimale sous différents niveaux de perturbations, nous avons exploité des métaheuristiques d'optimisation (algorithme génétique et essaim particulaire) en minimisant le critère d'erreur ITAE (Intégrale du Temps multiplié par la Valeur Absolue de l'Erreur), améliorant ainsi la rapidité et la précision de la réponse.

2. Le développement d'un système SCADA dédié à la supervision du groupe électrogène

- En partenariat avec Amimer Énergie, une entreprise leader algérienne dans la fabrication de groupes électrogènes et la réalisation de projets énergétiques, nous avons conçu une IHM sous AGI Creator.
- Cette interface permet de contrôler à distance le groupe (démarrage, arrêt, mode automatique), de visualiser en temps réel les paramètres électromécaniques (tension, vitesse, pression, température) et de gérer les alarmes. Elle s'appuie sur le protocole Modbus RTU pour communiquer avec le contrôleur embarqué (SGC420) du groupe.

Ce travail s'articule autour de l'amélioration de l'efficacité des groupes électrogènes à travers l'optimisation de leur régulation et l'intégration de systèmes de supervision modernes renforce la fiabilité des groupes électrogènes, en faisant des solutions autonomes et résilientes pour les zones isolées ou les secours d'urgence.

Chapitre 1:

Amimer Énergie et Aperçu de ses Groupes Électrogènes

1 Présentation de l'Entreprise

1.1 Présentation Succinete

Amimer Énergie est une entreprise algérienne spécialisée dans la fabrication et l'installation de groupes électrogènes, ainsi que dans la réalisation de centrales électriques. Fondée en **1989** à Seddouk, dans la wilaya de Béjaïa, sous le nom d'**Établissement Boukheddami**, elle a débuté dans la fabrication de postes à souder avant d'élargir ses activités au secteur énergétique.

Aujourd'hui, Amimer Énergie est un acteur majeur du marché avec plus de **450 employés**, **10 points de vente et de représentation** à travers l'Algérie, et un portefeuille de réalisations comprenant **200 MW de centrales thermiques** en Algérie et en Afrique, ainsi que plus de **24 MW de centrales photovoltaïques** en EPC. L'entreprise produit également plus de **1 000 groupes électrogènes** par an, avec des puissances allant de **10 kVA à 2 000 kVA**. Engagée dans l'innovation et le développement durable, elle investit dans la formation de ses équipes et l'amélioration continue de ses solutions énergétiques.[1]

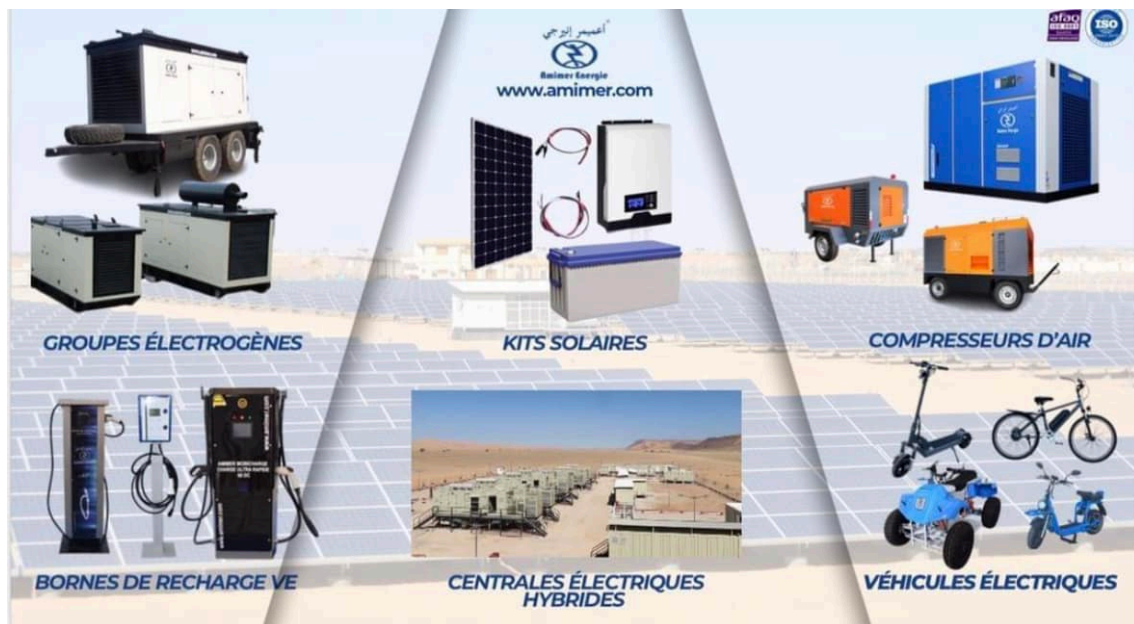


Figure 1: Gamme de produits Amimer Énergie

1.2 Services de l'Entreprise

Amimer Énergie est composée de trois divisions principales, chacune dédiée à des domaines spécifiques:

1. **Amimer Power Generation (APG)**, spécialisée dans la conception, la fabrication et l'installation de groupes électrogènes et d'autres produits tels que: les armoires électriques, les moto-soudeuses, les compresseurs d'air, les kits solaires, les conteneurs et les bornes de recharge.
2. **Amimer Power System (APS)**, chargée de la conception et de la réalisation de projets de centrales électriques, ainsi que de la gestion des projets en cours acquis par le groupe.
3. **Amimer Construction**, assure le développement des activités de charpente métallique, de bâtiment et de génie civil.



Figure 2: Centrale diesel mobile d'AMIMER ENERGIE au Sahara

Ces divisions collaborent pour offrir plusieurs services:

- **Engineering:** Un bureau d'études composé de plusieurs sections en génie électrique, génie civil et contrôle qualité, contrôle-commande et énergies renouvelables, approvisionnement et génie mécanique. Sa mission est d'analyser les besoins des clients pour proposer des solutions optimales adaptées à leurs projets.
- **Accompagnement technique:** Un support technique complet tout au long des projets, incluant l'élaboration de bilans énergétiques et des études de besoins pour répondre aux exigences actuelles et futures des clients.
- **Réparation et maintenance:** Un réseau d'experts assure le service après-vente des produits de l'entreprise sur tout le territoire national.

-
- **Location:** Un service dédié à fournir aux clients l'énergie nécessaire pour leurs projets ponctuels ou urgents, sans mobiliser leurs propres ressources, grâce à des solutions de location allant des groupes électrogènes individuels ou synchronisés aux solutions énergétiques fonctionnant par différentes sources d'énergies.
 - **Formation:** Des programmes de formation adaptés pour une meilleure prise en main des produits et une utilisation optimale.

La société ambitionne d'être devenir un acteur majeur dans les équipements et solutions énergétiques en Algérie et dans la région de l'Afrique du nord et de l'Afrique subsaharienne. Ainsi, elle vise d'être un acteur actif dans le domaine des énergies renouvelables en tant que EPC fournisseur de solutions mais également en tant que IPP, développeur propriétaire de centrales de production d'électricité, société d'exploitation et maintenance de centrales électriques solaires.

1.3 Structure Générale de l'Entreprise

Amimer Énergie est composée de:

- Une usine principale de fabrication située à Seddouk, dans la wilaya de Béjaïa.
- Des ateliers de montage et de maintenance située à Beni Merad, dans la wilaya de Blida.
- Un siège administratif basé à Alger.
- Plus de 10 agences de commercialisation réparties sur l'ensemble du territoire national.

Dans le cadre de notre projet de fin d'études, nous avons effectué notre stage pratique de mars à juin sur les deux sites d'Amimer Énergie:

Le Site de Blida, où nous avons réalisé notre étude sur la conception de l'armoire de synchronisation pour les quatre groupes électrogènes de 2MW, avec une supervision sur un système SCADA.

Le Site de Béjaïa, où nous avons assisté au processus de fabrication de l'armoire.

2 Qu'est-ce qu'un Groupe Électrogène ?

Un groupe électrogène est un système autonome de production d'électricité qui sert à produire de l'énergie électrique à partir de l'énergie chimique. Il est composé d'un moteur à combustion ou à explosion, d'un alternateur (généralement une génératrice synchrone, plus rarement génératrice asynchrone), ainsi que de tout appareillage de commande et de coupure et d'équipements auxiliaires associés, conformément à la Norme ISO 8528-1.[2]

Sa durée de vie est généralement de 10 à 15 ans, dépend de la qualité, de l'utilisation et de la maintenance.



Figure 3: Groupe Électrogène TASSILI 750 KVA - 2000 KVA

3 Les Types du Groupe Électrogène

Les groupes électrogènes se distinguent selon plusieurs critères, notamment leur poids, taille, le type de carburant, la mobilité, le mode de démarrage, l'isolation acoustiques et la tension de sortie.[3]

Toutefois, l'un des paramètres les plus critiques est la notion de puissance, qui définit le régime de fonctionnement du groupe et ses applications.

3.1 Groupes de Production

Les groupes électrogènes de production sont destinés pour une alimentation électrique continue ou semi-continue, indépendamment du réseau. Ils sont généralement constitués de plusieurs unités fonctionnant en parallèle, assurant ainsi la continuité du service même en cas de maintenance ou de panne d'un générateur.

Lorsque les groupes sont la seule source d'énergie sur un site isolé, ils fournissent 100% de leur puissance nominale sans limite de temps. En revanche, en milieux industriels où les charges varient, ils fonctionnent en moyenne à 70-80% de leur puissance nominale et une possibilité de surcharge de 10% pendant 1 heure toutes les 12 heures.[4]

Ces groupes sont largement utilisés dans les industries minières, les plateformes pétrolières, les exploitations agricoles et les sites isolés, où une autonomie énergétique complète est indispensable.

3.2 Groupes de Secours

Ces groupes sont utilisés uniquement en cas de coupure du réseau électrique. Ils sont dimensionnés pour fournir leur puissance maximale pendant une durée limitée, souvent inférieure à 500 heures par an, sans possibilité de surcharge prolongée.

Pour éviter toute surcharge, une marge de sécurité d'au moins 10 % est recommandée. Ces groupes sont essentiels pour les infrastructures critiques comme les hôpitaux, les centres de données, les télécommunications et les industries nécessitant une alimentation d'urgence.[4]

4 Les Composants d'un Groupe Électrogène Diesel

4.1 Moteur Diesel

Le moteur diesel est un moteur à pistons qui fonctionne selon un cycle à quatre temps (l'admission, la compression, la détente et l'échappement) et alimenté en gasoil. Il peut être conçu en version monocylindre ou multicylindre. Pour assurer son bon fonctionnement, ce moteur nécessite les circuits auxiliaires suivants:

- **Système d'alimentation:** Ce système comprend le réservoir, la pompe d'alimentation, la pompe d'injection, les filtres et les injecteurs, assurant une alimentation en diesel propre et continue pour une combustion efficace et une performance optimale du moteur.[5]
- **Système de démarrage:** Ce système comprend une solénoïde, une batterie et un démarreur à corde pour assurer un démarrage manuel ou un démarreur électrique pour un démarrage électrique. Pour certains modèles, un système pneumatique peut être ajouté pour des raisons de sécurité. L'ajout d'un tableau inverseur de source automatique permet d'assurer un démarrage automatique dès qu'il détecte une coupure de réseau électrique. Par ailleurs, l'intégration de systèmes de supervision permet désormais de démarrer les groupes à distance, assurant une gestion centralisée et une réactivité optimisée.[6]

-
- **Système de lubrification:** Ce système comprend un réservoir d'huile, une pompe à huile et des filtres, assurant une lubrification efficace des pièces métalliques mobiles du moteur pour minimiser l'usure et prévenir les surchauffes.[6]
 - **Système de refroidissement:** Ce système comprend généralement un radiateur, une pompe, un vase d'expansion, un thermostat, des ventilateurs et un liquide de refroidissement, qui est souvent de l'eau ou un mélange eau-additif pour dissiper efficacement la chaleur générée pendant le fonctionnement du moteur et maintenir les températures de ses différents organes à des niveaux acceptables.
Dans certains modèles, un système de refroidissement par air peut être utilisé, assurant le refroidissement sans recourir à une pompe ni à un échangeur de chaleur traditionnel.[5]
 - **Système d'échappement:** Ce système comprend souvent un silencieux pour réduire le bruit et un catalyseur ou filtre pour diminuer les émissions polluantes. Il garantit ainsi une évacuation sécurisée des gaz de combustion produits par le moteur tout en respectant l'environnement.



Figure 4: Moteur diesel General-Electric.

4.2 Alternateur

L'alternateur est une machine électrique couplée au moteur, exploitant l'énergie mécanique générée par celui-ci pour produire de l'électricité. Il existe deux principaux types d'alternateurs:

- **Alternateurs asynchrones**, peu courants en production autonome, nécessitent une alimentation initiale pour magnétisation.
- **Alternateurs synchrones**, largement utilisés dans les groupes électrogènes, nécessitent un régulateur de tension.

Ces alternateurs sont principalement composés d'un **stator**, la partie fixe contenant des enroulements triphasés dans lesquels le courant alternatif est induit et d'un **rotor**, partie mobile entraînée par le moteur, équipée d'un système d'excitation pour créer un champ magnétique tournant.

Son système d'excitation peut être:[5]

- **Avec bagues et balais**, la tension produite par l'excitatrice qui est directement posée sur l'arbre de l'alternateur, est acheminée au rotor grâce aux balais et des bagues collectrices.
- **Sans balais (auto-excité)**, une technologie récente utilisant une excitatrice (alternateur auxiliaire) associée à des redresseurs rotatifs à semi-conducteurs pour alimenter le rotor de l'alternateur principal.

Les groupes de grande puissance utilisent des alternateurs synchrones à trois étages, comprenant une sous-excitatrice (à aimants permanents), une excitatrice et un alternateur principal.[7]

Pour assurer son bon fonctionnement et éviter toute surchauffe, un ventilateur interne est intégré à l'alternateur, garantissant un refroidissement efficace de ces composants.



Figure 5: Alternateur synchrone Leroy-Somer

4.3 Dispositifs de Régulation

4.3.1 Le Régulateur de Tension Automatique

Il est communément appelé AVR, est un dispositif électronique à semi-conducteurs conçu pour **stabiliser automatiquement la tension de sortie de l'alternateur, notamment en cas de variations de charge.**

Il est composé de plusieurs composants tels que des diodes, des condensateurs, des résistances, des potentiomètres et parfois des microcontrôleurs. En général, l'AVR est installé au niveau de la boîte de jonction de l'alternateur.

Le régulateur de tension se trouve dans plusieurs catégories selon leur technologie:

- **Régulateur électromécanique**, basé sur des contacts mécaniques pour ajuster la tension. Ce type de régulateur présente une faible précision et une durée de vie limitée, ce qui a conduit à son remplacement par des technologies plus modernes.



Figure 6: Régulateur de tension électromécanique chromé

- **AVR analogique (Analog AVR)**, basé sur des circuits électroniques (amplificateurs opérationnels, transistors, résistances, condensateurs, etc.) pour surveiller la tension. Ces régulateurs se caractérisent par une fréquence de commutation de transistor élevée, des circuits simples et robustes, répondant largement aux besoins standards, avec un coût souvent inférieur à celui des AVR numériques.[8]



Figure 7: Régulateur de tension analogique Leroy-Somer

- **AVR numérique (Digital AVR)**, basé sur des microcontrôleurs ou des microprocesseurs qui traitent numériquement les informations relatives de la tension. Outre la régulation de base, un AVR numérique peut intégrer une multitude de protection (sur-tension, sous-tension, surchauffe, sur-excitation, etc.), implémenter des algorithmes de contrôle avancés tel que la compensation PID, enregistrer les données de fonctionnement et de performance de l'alternateur et d'autres fonctions supplémentaires telles que la surveillance par sondes RTD, l'intégration de fonctionnalités de type PLC (Automate Programmable), et la capacité de couper l'excitation en cas de conditions critiques.[8]



Figure 8: Régulateur de tension numérique Leroy-Somer

Le choix de l'AVR dans la fabrication des groupes électrogènes dépend principalement des exigences fonctionnelles et économiques de l'application.

4.3.2 Le Régulateur de Vitesse

Il est souvent appelé gouverneur, est un dispositif qui combine des capteurs, des actionneurs et une unité de contrôle ECU pour **maintenir une vitesse de rotation stable dans la plage de fonctionnement du moteur, afin de garantir une fréquence de sortie stable du générateur**. En général, le gouverneur est intégré dans **le volant moteur** et actionne sur la pompe à injection du moteur thermique.

Il existe de nombreux systèmes de régulation de la vitesse capables de réguler dynamiquement la soupape d'admission de carburant, les plus couramment utilisés sont:[9]

- **Gouverneur mécanique**, est un modèle traditionnel repose sur un ensemble de volants centrifuges qui, sous l'effet de la force centrifuge, modifient l'ouverture d'un clapet ou la position d'un injecteur via des ressorts et des leviers. Il se caractérise par leur simplicité, leur robustesse et leur coût abordable, mais offre une précision moindre et est sensible à l'usure des composants.



Figure 9: Gouverneur mécanique

-
- **Gouverneur électronique**, est un système moderne intègre une ECU utilisant des algorithmes de régulation, généralement PID, pour ajuster précisément l'injection de carburant. Un servo-moteur ou une électrovanne applique ces corrections en temps réel.

Ce régulateur offre une grande précision, une réactivité accrue et des capacités de diagnostic avancées, bien qu'il nécessite un entretien spécialisé et présentent un coût plus élevé.



Figure 10: Gouverneur électronique

Pour garantir un fonctionnement sécurisé, des protections sont intégrées dans le gouverneur afin d'éviter les sur-injections susceptibles d'endommager le moteur et de détecter toute anomalie du système de contrôle.

4.4 Système de Contrôle/Commande et Protection

Ce système contient de plusieurs éléments essentiels pour assurer un fonctionnement stable, fiable et sécurisé. On y trouve principalement:

- **Disjoncteur Motorisé**, sert à protéger l'alternateur en déconnectant automatiquement la charge en cas de surcharge ou de court circuit.
- **Tableau de commande**, est un boîtier équipé principalement d'un contrôleur qui sert d'interface principale entre l'opérateur et le système.
- **Bouton poussoir d'arrêt d'urgence**, permet d'interrompre immédiatement le fonctionnement du groupe électrogène en cas de situation critique.
- **Clé de commande (Key Switch)**, est un commutateur à clé permet de commander le démarrage et le fonctionnement du moteur.
- **Appareillage de protection et de commande**, comme les disjoncteurs, les relais, les fusibles ainsi que les instruments de comptage complétés par des boutons de commande et des LED d'alarme à l'extérieur, garantissent une protection et une commande sécurisées.

Le choix et le dimensionnement de ces dispositifs, ou d'autres équipements peuvent être ajoutés, doivent être effectués en fonction des spécificités de l'application des groupes électrogènes, tout en tenant compte des exigences opérationnelles et des normes électrique en vigueur.

5 Fonctionnement des Groupes Électrogènes

5.1 Vérifications et Opérations à réaliser avant le Fonctionnement

Avant de démarrer un groupe électrogène, il est essentiel d'effectuer plusieurs contrôles pour assurer un fonctionnement optimal et en toute sécurité.

Voici les différentes vérifications à faire:[10]

1. **Vérification Générale:** Il s'agit du contrôle visuel du fonctionnement générale du moteur et du groupe électrogène, en s'assurant que les conditions sont optimales pour l'admission et l'émission d'air, ainsi que l'évacuation du gaz d'échappement.
2. **Vérification des fluides:** Il faut d'abord ouvrir le bouchon du radiateur pour vérifier le niveau d'eau et ajouter un mélange 50 % eau 50 % antigel si nécessaire. Ensuite, il est important de contrôler le niveau d'huile moteur à l'aide de la jauge et de compléter si besoin. Enfin, il faut vérifier le niveau de carburant et faire l'appoint si nécessaire afin de garantir une alimentation continue du moteur.
3. **Vérification des filtres et des systèmes d'air:** Il faut contrôler l'état des filtres et la circulation de l'air en commençant par vérifier l'indicateur du filtre à air afin de s'assurer qu'il n'est pas obstrué. Si le filtre est encrassé, il doit être remplacé pour éviter une baisse des performances du moteur. Enfin, il est important de vérifier que les volets du radiateur et ceux de l'admission d'air propre sont bien ouverts, pour une bonne ventilation et un refroidissement efficace.
4. **Vérification des connexions électriques:** Vérifier le câblage et les connexions de la batterie afin de détecter d'éventuelles anomalies. Ensuite, il est important de s'assurer que les bornes sont bien serrées et propres pour éviter toute perte de contact. Enfin, si la batterie nécessite un entretien, il faut contrôler le niveau d'eau distillée et compléter si nécessaire.
5. **Vérification des dispositifs de sécurité:** Avant de mettre en service le groupe électrogène, il est essentiel de s'assurer que le disjoncteur de sortie est en position OFF afin d'éviter tout risque électrique. Ensuite, il faut vérifier que le bouton d'arrêt d'urgence n'est pas activé, garantissant ainsi un démarrage sans interruption. Enfin, vérifier l'absence de fuites, fissures ou desserrages pouvant affecter la sécurité et le fonctionnement du groupe électrogène.

-
6. **Vérification de l'environnement du groupe électrogène:** S'assurer qu'aucun objet étranger (clés, outils, papiers) ne se trouve sur le moteur ou l'alternateur.

5.2 Principe de Fonctionnement

Un groupe électrogène diesel génère de l'électricité en convertissant l'énergie chimique du carburant en énergie électrique par un processus de transformation électromécanique. Dans un premier temps, le moteur thermique transforme l'énergie chimique du diesel en énergie mécanique. Ensuite, cette dernière est convertie en électricité à travers un alternateur, selon les principes de l'induction électromagnétique. Le fonctionnement d'un générateur diesel peut être décomposé en plusieurs étapes essentielles, qui sont détaillées dans les sections suivantes:

1. **Démarrage:**

Il repose sur une source électrique externe permettant d'actionner le démarreur. Tout d'abord, un chargeur de batterie, composé d'un abaisseur de tension, d'un redresseur, d'un filtre et d'appareils de mesure de courant et de tension, convertit le AC en DC. Ce dernier alimente des batteries qui fournissent l'énergie nécessaire à un MCC, jouant le rôle de démarreur.

Celui-ci génère un couple suffisant pour amener le moteur diesel à sa vitesse minimale de démarrage. Une fois cette vitesse atteinte, le moteur à courant continu est automatiquement déconnecté, laissant le moteur diesel fonctionner de manière autonome.

2. **Combustion (Chimique $\rightarrow$ Mécanique):**

Le moteur diesel convertit l'énergie chimique du carburant en énergie mécanique grâce à un processus de combustion interne basé sur l'auto-inflammation, où le diesel s'enflamme spontanément sous l'effet de la compression.

3. **Fonctionnement de l'alternateur (Mécanique $\rightarrow$ Électrique):**

Le vilebrequin, qui convertit le mouvement linéaire du piston en mouvement rotatif, est connecté au moteur ainsi qu'à l'arbre de l'alternateur, qu'il entraîne pour mettre en rotation le générateur à aimant permanent (PMG). Ce dernier, possédant son propre champ magnétique, induit une tension alternative dans les enroulements de son stator lorsqu'il tourne.

Cette tension est ensuite envoyée vers l'AVR, qui la convertit en tension continue à l'aide d'un redresseur à thyristors.

Le DC obtenu alimente l'enroulement du stator de l'excitatrice, générant un champ magnétique qui interagit avec le rotor de l'excitatrice. Cette interaction induit à son tour une tension alternative dans le rotor.

La tension produite passe ensuite par un redresseur rotatif, qui la transforme en tension continue avant de l'envoyer vers les enroulements du rotor principal. Cela génère un champ magnétique tournant, lequel interagit avec les enroulements du stator principal, produisant ainsi une tension alternative.

Enfin, cette tension est envoyée vers le jeu de barres, permettant d'alimenter les charges en électricité.

4. **Régulation du groupe:** Il est essentiel d'intégrer un système de régulation afin d'assurer un fonctionnement stable du groupe et une alimentation électrique fiable et continue.

- **Régulation de la tension:**

Son principe repose sur le rôle de l'AVR, qui contrôle le courant d'excitation afin de stabiliser la tension de sortie du générateur.

Des transformateurs de courant et de tension sont installés sur le jeu de barres pour mesurer en continu la tension et le courant délivrés aux charges. Ces valeurs sont ensuite envoyées à l'AVR, où elles sont comparées à une tension de référence.

Si une différence est détectée, un signal d'erreur est généré. Ce signal est amplifié puis redressé par un thyristor, qui ajuste l'angle de d'amorçage pour réguler le courant d'excitation du générateur.

- **En cas de chute de tension**, il augmente l'excitation pour éviter un dysfonctionnement des équipements et une charge insuffisante de la batterie.
- À l'inverse, **si la tension est trop élevée**, il réduit ce courant afin de prévenir toute surcharge ou détérioration des appareils alimentés.

- **Régulation de la vitesse:**

Il s'agit de l'action du Governor, qui ajuste la position de la vanne d'injection de carburant afin de maintenir la vitesse de rotation du moteur et, par conséquent, la fréquence du générateur.

Un capteur de vitesse magnétique mesure la vitesse de rotation et envoie cette valeur au régulateur. Cette vitesse est comparée à une valeur de consigne.

Si une différence est détectée, un signal d'erreur est généré ensuite amplifié puis envoyé à un actionneur, qui ajuste la position de la vanne d'injection de carburant.

- **Si la vitesse du moteur est trop basse**, l'actionneur ouvre davantage la vanne de carburant, permettant à une plus grande quantité de diesel d'entrer dans le moteur. Cela augmente la combustion et accélère le moteur, stabilisant ainsi la fréquence du générateur.
- À l'inverse, **si la vitesse du moteur est trop élevée**, l'actionneur réduit l'apport en carburant, ce qui ralentit le moteur et ajuste la fréquence à la valeur nominale.

6 Processus de Fabrication d'un Groupe Électrogène

Les différentes parties constituant le groupe électrogène sont fabriquées et assemblées au sein des différents ateliers de production suivante:

1. **Atelier chaudronnerie:** Situé sur le site de Béjaïa, cet atelier a pour rôle le découpage et le façonnage des tôles métalliques pour fabriquer les pièces nécessaires destinées à l'unité de production du site de Blida.
2. **Atelier électromécanique:** Les produits semi-finis arrivant sont utilisés pour faire l'assemblage des châssis. L'accouplement du moteur avec la génératrice en monobloc est réalisé, suivi de son montage sur le châssis et de sa fixation. Les tuyauteries des différents liquides du groupe sont ensuite raccordées. Actuellement, même le remplissage avec l'huile et liquide de refroidissement se fait au niveau de cette atelier.
3. **Atelier électrotechnique:** En parallèle avec l'atelier mécanique, d'autres pièces provenant de l'atelier chaudronnerie sont acheminées vers cet atelier pour permettre la fabrication des armoires de commande, armoires disjoncteurs, armoires inverseurs.
La préparation des faisceaux (faisceaux de plaque, faisceaux moteur, câbles de puissance, câbles de résistance de préchauffage, câbles de batterie, etc.) est effectuée. Ensuite, les composants (bornier, relais, porte-fusibles, inverseurs, etc.) sont montés sur chaque plaque. Enfin, l'accouplement des plaques câblées avec les armoires assemblées marque la fin de cette étape.
4. **Assemblage final:** Une fois toutes les étapes précédentes terminées, l'assemblage du bloc Générateur-moteur, son système commande et le capotage est effectué pour former un groupe électrogène fonctionnel.
5. **Banc d'essai:** Après la fin de la chaîne de production, cette étape consiste en un contrôle qualité où chaque groupe est soumis à une série de tests pour valider son bon fonctionnement:
 - Testes de performances
 - Testes de charge résistive.

À la fin des tests, un bilan de tests est rédigé, indiquant toutes les valeurs mesurées telles que le courant, la tension, la fréquence, la température, etc. Ces informations sont ensuite intégrées dans le dossier technique du groupe. En cas de problème, des ajustements seront effectués sur le groupe. Sinon, il est envoyé à la section commercialisation et remis au client accompagné de son dossier technique complet.

Chapitre 2:

Optimisation Intelligente de la Régulation PID d'un Groupe Électrogène

1 Modélisation du Groupe Électrogène

1.1 Modèle Simplifié d'un Générateur Synchrone

Le modèle simplifié de la machine synchrone est une approximation du modèle complet, il s'agit d'un modèle qui ne tient compte que des dynamiques du rotor. Cette dynamique est décrite par la loi fondamentale de la mécanique:[11]

$$T_m - T_e = J \frac{d\Omega_m}{dt} = J \frac{d^2\delta_m}{dt^2} \quad (1)$$

Où:

- T_m : Couple mécanique fourni par le Moteur (N.m).
- T_e : Couple électromagnétique demandé par la charge (N.m).
- J : Moment d'inertie du système tournant ($\text{kg} \cdot \text{m}^2$).
- Ω_m : Vitesse angulaire du rotor (rad/s).
- δ_m : Position angulaire du rotor (rad).

Ainsi, l'énergie cinétique du rotor est:

$$W = \frac{1}{2} J \Omega_m^2$$

D'où:

- $M = J \Omega_m$, représente une constante d'inertie.

Alors, en multipliant l'équation (1) par Ω_m , on obtient:

$$P_m - P_e = M \frac{d^2\delta_m}{dt^2} \quad (2)$$

On sait que:

- L'angle angulaire électrique $\delta_e = \frac{2}{p} \delta_m$.
- La vitesse angulaire électrique $\Omega_e = \frac{2}{p} \Omega_m$.

Donc, l'équation (2) devient:

$$P_m - P_e = \frac{2}{p} M \frac{d^2\delta_e}{dt^2} \quad (3)$$

On définit:

- la constante d'inertie $H = \frac{0.5 J \Omega_m^2}{S_B}$.

On déduit:

$$M = \frac{HpS_{\text{base}}}{\Omega_e} \quad (4)$$

En remplaçant l'équation (4) dans l'équation (3) et après quelques simplifications mathématiques, on obtient l'expression suivante dans le système des unités relatives (pu):

$$P_m - P_e = \frac{H}{\pi f_0} \frac{d^2 \delta_e}{dt^2} \quad (5)$$

Où:

- f_0 : La fréquence nominale du réseau (50 Hz)

Introduisant une perturbation légère Δ à partir de point nominal de fonctionnement du générateur. L'équation (5) devient:

$$\Delta P_m - \Delta P_e = \frac{H}{\pi f_0} \frac{d^2 \Delta \delta_e}{dt^2}$$

Où:

$$\Delta P_m - \Delta P_e = \frac{2H}{f_0} \frac{d^2 \Delta f}{dt^2} \quad (6)$$

En appliquant la transformée de Laplace à l'équation (6), on obtient l'expression suivante:

$$\Delta f(s) = \frac{f_0}{2Hs} [\Delta P_m - \Delta P_e]$$

Et donc le modèle linéaire d'un générateur synchrone soumis à une légère perturbation s'écrit:

$$\Omega_e(s) = \frac{1}{2Hs} [\Delta P_m - \Delta P_e] \quad (7)$$

Ce modèle est représenté dans la figure suivante:

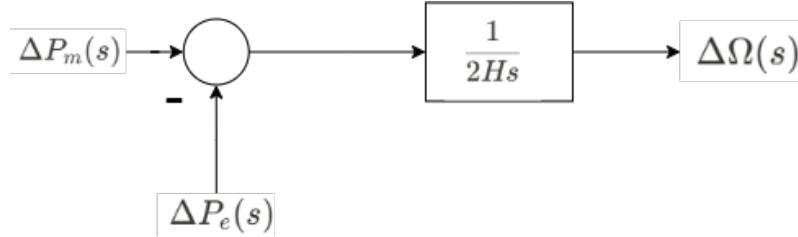


Figure 11: Le modèle linéaire du GS avec $\Omega(s)$

1.2 Modèle de la Charge

La charge dans un système électrique peut être constituée de deux types principaux:

- Les charges résistives, telles que l'éclairage ou le chauffage, dont la puissance consommée est indépendante de la fréquence.

- Les charges motorisées, plus sensibles aux variations de fréquence, avec une dépendance qui reflète la caractéristique vitesse-charge des différents dispositifs d'entraînement.

La caractéristique fréquence-charge d'une charge composite peut être approximée par l'expression suivante:[11]

$$\Delta P_{lec}(pu) = \Delta P_L(pu) + D(pu)\Delta f$$

Où:

- D : Le coefficient de sensibilité de la charge à la fréquence, exprimant le pourcentage de variation de la charge par rapport à celui de la fréquence.
- ΔP_L ; La charge insensible à la fréquence (résistive).
- $D\Delta f$: La charge sensible à la fréquence (motorisée).

En intégrant le modèle de charge dans le modèle du générateur, on obtient une représentation modifiée, telle qu'illustrée dans la figure suivante:

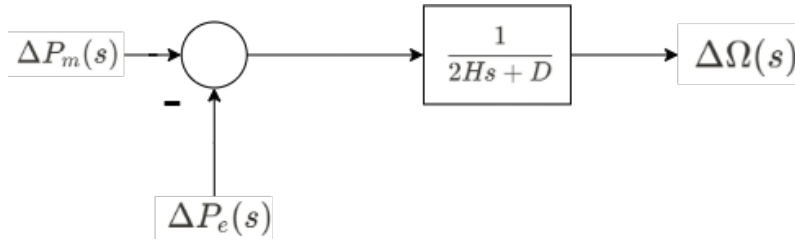


Figure 12: Le modèle linéaire du GS avec la charge

1.3 Modèle de Moteur Diesel

Le moteur diesel est modélisé comme un système gouverné produisant une puissance mécanique. Ce modèle représente la dynamique entre la position de la commande de carburant (injection) et la vitesse du moteur.

Le modèle comprend: [12]

- **Bloc de gouverneur de vitesse:** Ce bloc est représenté par un contrôleur PID .
- **Bloc de l'actionneur de vanne:** Ce bloc modélise la réponse dynamique de l'actionneur qui ajuste la quantité de carburant injectée dans le moteur. Il est représenté par un système du premier ordre:

$$G_{\text{actionneur}}(s) = \frac{1}{1 + sT_{sm}}$$

Où:

- T_{sm} est la constante de temps de l'actionneur, représentant sa lenteur de réponse mécanique.

- **Bloc du moteur diesel:** Ce bloc modélise la dynamique du moteur thermique, c'est-à-dire la conversion du carburant en puissance mécanique:

$$G_{\text{Moteur}}(s) = \frac{1}{1 + sT_d}$$

Où:

- T_d est la constante de temps d'inertie du moteur diesel, représente le retard de répondre à une variation de la commande de carburant.

Le modèle global du moteur diesel est illustré dans la figure suivante:

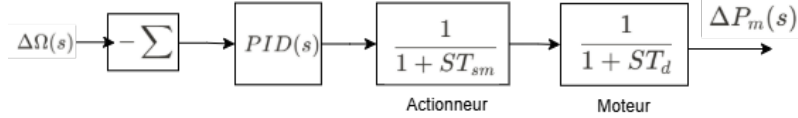


Figure 13: Le modèle du moteur diesel

1.4 Modèle Global du Groupe Électrogène

Le modèle global du groupe électrogène représente l'interaction complète entre les sous-systèmes précédents, reliés en cascade et intégrés dans une boucle de rétroaction.

Le système est également soumis à une perturbation mécanique externe modélisée par une variation soudaine de la charge (ΔP_L).

Cette perturbation est injectée à l'entrée du générateur synchrone sous forme d'une puissance électrique soustraite à la puissance mécanique développée par le moteur. Cela permet de simuler le comportement réel d'un groupe électrogène soumis à des variations de charge.

Le modèle global du groupe électrogène est montré dans la figure suivante:

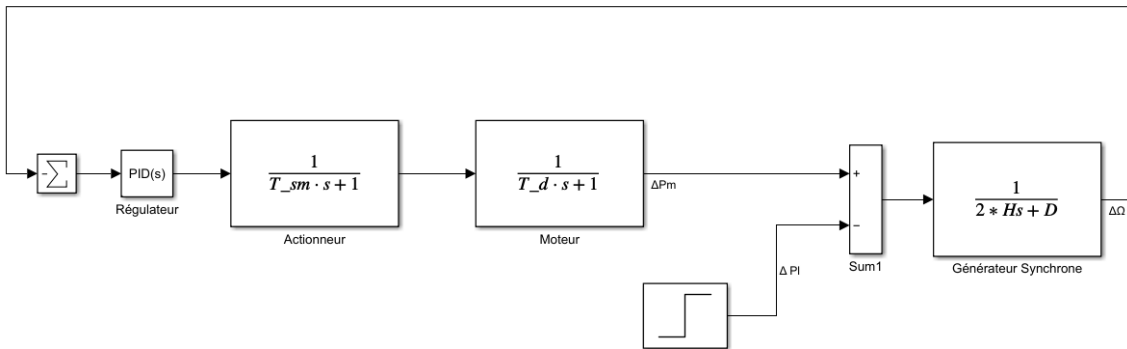


Figure 14: Le modèle global du groupe électrogène

2 Structure de Régulage PID

La régulation PID (Proportionnelle – Intégrale – Dérivée) est l’une des techniques de contrôle les plus répandues dans les systèmes automatisés, répondant à plus de 90% des besoins en régulation industrielle.[13]

Elle a pour mission de minimiser l’écart entre la consigne et la variable mesurée. Pour cela elle agit sur le système asservi en fournissant un signal de commande qui corrige l’erreur en temps réel.

L’objectif est d’obtenir à la fois:

- La précision: Erreur statique nulle ou très faible.
- La rapidité: Réponse dynamique rapide sans lenteur excessive.
- La stabilité: Pas d’oscillations persistantes ou divergences.

Dans les groupes électrogène, et en particulier dans le contrôle de vitesse, l’algorithme est généralement implémenté dans l’unité de commande électronique ECU, située au niveau du gouverneur. Son rôle est de surveiller en permanence la vitesse du moteur et de la maintenir à sa valeur nominale, quelles que soient les perturbations de charge pouvant survenir durant le fonctionnement. [13]

2.1 Structure du Régulateur

Un régulateur combine trois actions correctrices, chacune modulée par un gain ajustable:

- **Action proportionnelle (P):** Elle agit instantanément en fonction de l’erreur actuelle.

La sortie proportionnelle est donnée par:

$$P = K_p \cdot e(t)$$

Cette action permet une réponse rapide à l’écart, mais ne suffit pas à elle seule pour annuler l’erreur statique.

- **Action intégrale (I):** Elle prend en compte l’erreur cumulée dans le temps pour corriger les erreurs persistantes.

La sortie intégrale est exprimée par:

$$I = K_i \int_0^t e(t) dt$$

Cette action est essentielle pour éliminer l’erreur de régime permanent, mais si elle est mal réglée, elle peut provoquer un dépassement (overshoot) et ralentir la réponse du système.

- **Action dérivée (D):** Elle anticipe l'évolution de l'erreur en agissant sur sa dérivée temporelle.

Elle est donnée par:

$$D = K_d \frac{de(t)}{dt}$$

Cette action améliore la stabilité et permet d'amortir les variations rapides de l'erreur, mais elle est sensible au bruit et peut entraîner une sollicitation excessive de l'organe de commande.

La fusion des termes P, I et D est également appelée forme parallèle (ou forme non interactive).

Dans cette configuration, le signal de commande total appliqué au système $u(t)$ est simplement la somme des trois contributions:

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt}$$

La figure ci-dessous montre un schéma-bloc d'un régulateur:

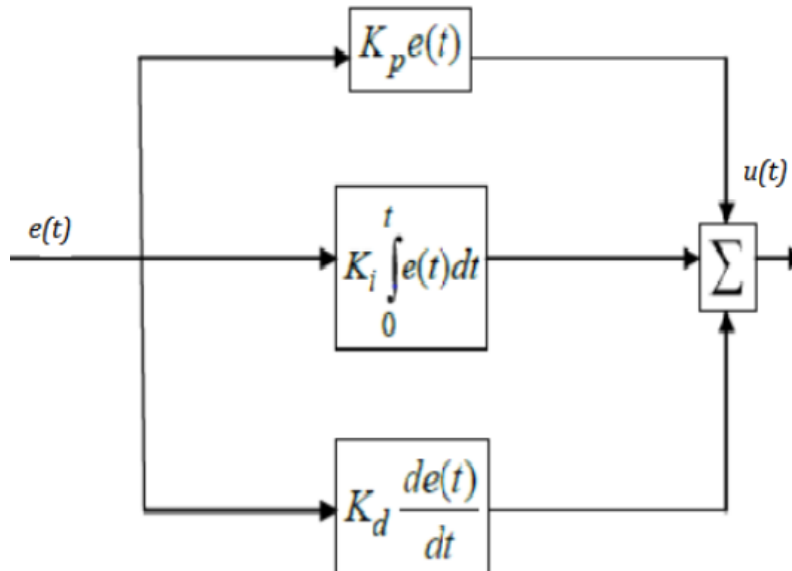


Figure 15: Schéma bloc d'un régulateur PID

En notation de Laplace, le signal de commande s'écrit:

$$U(s) = K_p E(s) + K_i \frac{dE(s)}{s} + K_d s E(s)$$

Autrement exprimée:

$$U(s) = K_p E(s) \left(1 + \frac{1}{T_i s} + T_d s\right)$$

Alors, sa fonction de transfert s'écrit:

$$G_R(s) = K_p + \frac{K_i}{s} + K_d s = K_p \left(1 + \frac{1}{T_i s} + T_d s\right)$$

Où:

- $T_i = \frac{K_p}{K_i}$
- $T_d = \frac{K_d}{K_p}$

Le tableau suivant présente les effets de l'augmentation des gains PID sur les performances du système:[13]

Table 1: Effets de L'augmentation des Gains PID sur les Performances du Système

	Augmentation de	Stabilité	Précision	Rapidité
[13]	K_p	Diminue	Augmente	Augmente
	K_i	Diminue	Augmente	Diminue
	K_d	Augmente	Diminue	Augmente

Ainsi, l'intérêt du régulateur PID est d'intégrer les effets positifs des trois actions précédentes (proportionnelle, intégrale et dérivée), Cela implique un choix judicieux des paramètres K_p , K_i et K_d ce qui représente souvent une tâche complexe, mais essentielle pour garantir des performances satisfaisantes du système.

2.2 Méthodes de Réglage des Paramètres PID

Plusieurs méthodes permettent d'ajuster les paramètres d'un régulateur PID à l'aide de formules ou d'algorithmes spécifiquement conçus pour la configuration du régulateur.

Ces méthodes sont généralement classées comme suit:[14]

1. **Les méthodes classiques empiriques en boucle fermée:** Ces méthodes utilisent la rétroaction pour extraire des paramètres caractéristiques (gain critique, période d'oscillation) directement en boucle fermée.

On y distingue notamment:

- Méthode de Ziegler–Nichols.
- Méthode d'oscillation amortie.
- Méthode Tyreus–Luyben.

2. **Les méthodes classiques empiriques en boucle ouverte:** Ces méthodes sont basées sur la configuration en boucle ouverte.

Parmi lesquelles on peut citer:

- Méthode Cohen et Coon.
- Méthode Cohen et Coon modifiée.
- Méthode Fertik.

-
3. **Les méthodes analytiques:** Elles utilisent un modèle mathématique du système pour calculer les gains PID.

Parmi ces méthodes, on trouve notamment:

- Méthode IMC (Internal Model Control).
- Critères d'erreur minimum (IAE, ISE, ITAE).

4. **Les méthodes intelligentes:** Ces approches ne nécessitent pas de modèle précis du système et sont particulièrement adaptées aux systèmes complexes, non linéaires ou incertains.

On peut notamment utiliser:

- La logique floue.
- Les réseaux de neurones.

D'autres approches peuvent également être utilisées et le choix de la méthode la plus appropriée dépend principalement de la complexité du système et des objectifs à atteindre.

3 Synthèse des Gains PID: Approche Classique de Ziegler–Nichols

La méthode d'oscillation de Ziegler–Nichols est une technique empirique de réglage des régulateurs PID qui repose sur la mise en boucle fermée du système et la recherche des conditions d'oscillation soutenue. Pour cela, on s'appuie sur la notion de stabilité critique.

Considérons le schéma du système sous l'étude Ziegler-Nichols illustré dans la figure ci-dessous:[14]

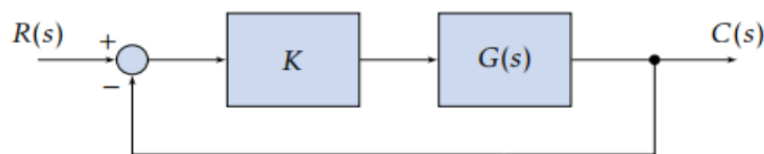


Figure 16: Système sous étude: Ziegler-Nichols

Cette méthode se déroule en trois phases:

1. Mise en place un régulateur P seul ($T_i = \infty$, $T_d = 0$) et on augmente progressivement le gain proportionnel K_p à partir d'une faible valeur, jusqu'à ce que la sortie du système entre en oscillation de façon soutenue (oscillations de même amplitude et de même période, non amorties ni croissantes).
2. Détermination de gain critique K_{cu} et la période des oscillations T_u , telque:
 - K_{cu} est la valeur du gain proportionnel K_p pour laquelle les oscillations sont soutenues.

- T_u est la période des oscillations observées à $K_p = K_{cu}$.

Ces deux paramètres caractérisent approximativement la dynamique du système en boucle fermée et servent ensuite à calculer les réglages du régulateur PID.

3. Calcul des paramètres PID selon les formules empiriques en fonction de K_{cu} et T_u de Ziegler–Nichols suivantes:[14]

Table 2: Réglages Empiriques selon la Méthode de Ziegler–Nichols

	K_p	T_i (sec)	T_d (sec)
P	$0,5 K_{cu}$	∞	0
PI	$0,45 K_{cu}$	$T_u/1,2$	0
PID	$0,6 K_{cu}$	$T_u/2$	$T_u/8$

En conséquence, les actions du PID sont déterminées comme suit:

$$K_p = 0.6K_{cu}, \quad K_i = \frac{K_p}{T_i}, \quad K_d = K_p T_d.$$

4 Synthèse des Gain PID: Approche Methaheuristique

4.1 Position du Problème d’Optimisation: Critères d’Erreur Minimas

Cette approche vise à optimiser les paramètres PID (Proportionnel, Intégral, Dérivé) en minimisant une mesure intégrale de l’erreur entre la consigne et la réponse du système. Elle repose sur des indicateurs mathématiques pour quantifier l’erreur globale durant le régime transitoire.

- **ISE (Integral of Squared Error)**

Mathématiquement données par:[14]

$$\text{ISE} = \int_0^{\infty} e^2(t) dt$$

Ce critère présente les points suivants:

- **Pénalisation des grandes erreurs:** Grâce à son effet quadratique, ISE pénalise fortement les grandes erreurs ce qui le rend particulièrement efficace pour atténuer les pics de perturbation ou les écarts brusques.

-
- **Tolérance aux petites erreurs en régime permanent:** En privilégiant les grandes erreurs, l'ISE devient moins sensible aux écarts faibles et durables une fois le régime établi, ce qui peut conduire à une précision statique moindre.

Souvent cela conduit à des réponses rapides, mais avec une amplitude considérable, une faible atténuation des petites erreurs, et parfois des oscillations.

- **IAE (Integral of Absolute Error)**

Il est défini par:[14]

$$\text{IAE} = \int_0^{\infty} |e(t)| dt$$

Ce critère se caractérise par:

- **Pénalisation uniforme:** L'IAE somme la valeur absolue de l'erreur, pénalisant équitablement les déviations positives et négatives.
- **Sensibilité aux erreurs persistantes:** Contrairement à l'ISE, qui privilégie l'élimination rapide des erreurs de forte amplitude, l'IAE pénalise de façon équitable et continue même les plus petites déviations, ce qui favorise une réduction plus complète de l'erreur à l'état stationnaire.

L'IAE conduit généralement à une réponse plus lente que les systèmes optimaux ISE, mais généralement avec une oscillation moins durable.

- **ITAE(Integral of Time-weighted Absolute Error)**

Mathématiquement, ce critère s'exprime par:[14]

$$\text{ITAE} = \int_0^{\infty} t |e(t)| dt$$

- **Accent sur la stabilité à long terme:** L'erreur tardive (après la phase transitoire) est beaucoup plus pénalisée que l'erreur initiale.
- **Réduction rapide du temps de réponse au voisinage de la consigne:** En minimisant la surface sous la courbe $t |e(t)|$, on obtient généralement un temps de stabilisation plus court que pour ISE ou IAE.

l'ITAE accélère la stabilisation du système tout en imposant un démarrage plus progressif, indispensable pour maîtriser les oscillations.

4.2 Procédure de Réglage Optimale

1. Identifier ou estimer la fonction de transfert ou le modèle d'état du système à contrôler.
2. Définir le critère d'erreur $J(K_p, K_i, K_d)$ à minimiser en fonction de l'objectif principal du système de contrôle.

-
3. Utiliser des méthodes numériques d'optimisation pour ajuster les gains du régulateur K_p, K_i, K_d basées sur le gradient ou quasi-Newton comme BFGS.
 4. Simuler la réponse du système face à différentes perturbations valider les performances obtenues.

4.3 Solutions Approche Proposées

Utiliser des méthodes intelligentes d'optimisation pour ajuster les gains du régulateur K_p, K_i, K_d basées sur Algorithmes Évolutionnaires (Metaheuristiques), notamment:

- Algorithmes Génétique (GA).
- Optimisation par essaims particulaires (PSO).

5 Optimisation par Essaims Particulaires OEP

L'optimisation par essaims particulaires (Particle Swarm Optimization PSO) est un algorithme métaheuristique basé sur la population et inspirée de l'intelligence en essaim plus précisément le comportement social des bancs de poissons ou des vols groupés d'oiseaux pour développer une solution optimale au problème. Elle repose sur les principes d'auto-organisation où un groupe d'organismes peut adopter un comportement globalement intelligent et coordonné à partir de règles individuelles simples, sans supervision centrale.

Il est largement utilisé pour résoudre des problèmes complexes d'optimisation dans de nombreux domaines, notamment l'apprentissage automatique, l'électronique de puissance et le contrôle.

5.1 Principe de Fonctionnement de l'Algorithme OEP

Dans PSO, on considère une population (essaims) de N agents simples, appelés particules, évoluant dans un espace de recherche de dimension D . Chaque particule i constitue une solution candidate, représentée par son vecteur de position $\mathbf{x}_i$ dans cet espace de décision.

Chaque particule est caractérisée par les vecteurs suivants:

- $\mathbf{x}_i$: La position actuelle de la particule dans l'espace de recherche.
- $\mathbf{p}_i$: La meilleure position visitée par la particule elle-même.
- $\mathbf{g}$: La meilleure position visitée par l'ensemble de l'essaim.
- $\mathbf{v}_i$: La vitesse de la particule (vélocité), indiquant la direction du déplacement.

Au cours de chaque itération, la mise à jour de la vitesse puis la position de la particule i obéit à la règle générale suivante:[15]

$$\mathbf{v}_i(t+1) = w \mathbf{v}_i(t) + c_1 r_{1,i}(\mathbf{p}_i - \mathbf{x}_i(t)) + c_2 r_{2,i}(\mathbf{g} - \mathbf{x}_i(t)) \quad (8)$$

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1) \quad (9)$$

Où:

- w est le coefficient d'inertie.
- c_1 et c_2 sont les coefficients d'accélération.
- r_1 et r_2 sont deux nombres aléatoires entre 0 et 1.

Ainsi, le déplacement d'une particule résulte de la combinaison de trois composantes:

1. **La composante d'inertie du déplacement** ($w \mathbf{v}_i(t)$), où le paramètre ω contrôle l'influence de la direction de déplacement sur le déplacement futur.
2. **La composante cognitive du déplacement** ($c_1 r_1(\mathbf{p}_i - \mathbf{x}_i(t))$), où le paramètre c_1 contrôle le comportement cognitif de la particule.
3. **La composante sociale du déplacement** ($c_2 r_2(\mathbf{g} - \mathbf{x}_i(t))$), le paramètre c_2 contrôle l'aptitude sociale de la particule.

La stratégie de déplacement d'une particule est illustrée dans la figure suivantes:

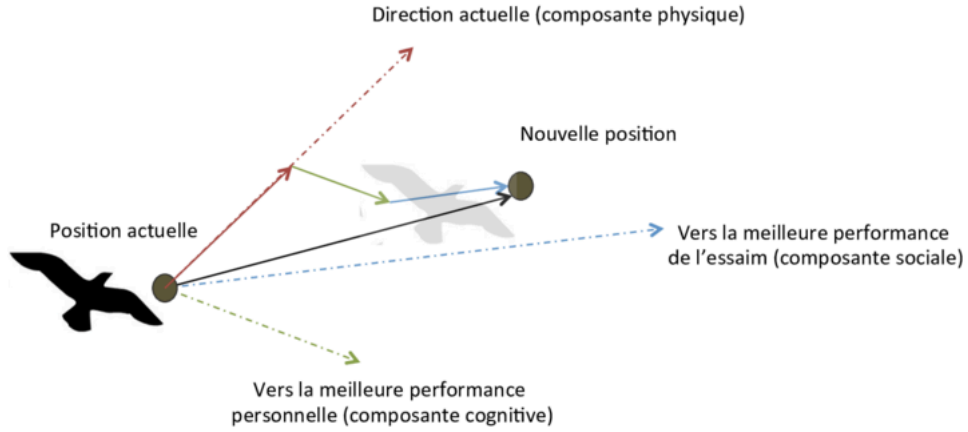


Figure 17: Déplacement d'une Particule.

Une fois le déplacement des particules effectué, les nouvelles positions sont évaluées et les deux vecteurs $\mathbf{p}_i$ et $\mathbf{g}$ sont mis à jour comme suit: [16]

$$\vec{p}_i(t+1) = \begin{cases} \vec{p}_i(t), & \text{si } f(\vec{x}_i(t+1)) \geq f(\vec{p}_i(t)) \\ \vec{x}_i(t+1), & \text{sinon} \end{cases} \quad (10)$$

$$\vec{g}(t+1) = \arg \min_{\vec{p}_i} f(\vec{p}_i(t+1)), \quad 1 \leq i \leq N. \quad (11)$$

Grâce à cette coopération entre particules, l'essaim converge progressivement vers les zones optimales de l'espace de recherche, guidé par les succès individuels et collectifs de ses membres.

5.2 Étapes de l'Algorithme PSO

Le PSO vise à optimiser une fonction objectif, définie par l'utilisateur en fonction de l'application ciblée. Cette fonction sert de critère pour évaluer les performances des solutions proposées par les particules.

Les étapes de l'algorithme sont les suivantes: [16]

1. **Initialiser** aléatoirement N particules: x_i et v_i ;
2. **Évaluer** la fonction objectif pour chaque position initiale des particules;
3. **Pour** chaque particule i , **assigner** la meilleure position personnelle connue $\vec{p}_i = \vec{x}_i$
4. **Déterminer** la meilleure position globale $\vec{g}$ selon la formule (11);
5. **tant que** le critère d'arrêt n'est pas satisfait, **répéter**:
 - (a) **Mettre à jour** la vitesse et la position de chaque particules selon les formules(8) et (9);
 - (b) **Réévaluer** les positions des particules selon la fonction objectif;
 - (c) **Mettre à jour** es meilleures positions personnelles et globale $\vec{p}_i$ et $\vec{g}$ selon les formules (10) et (11);
6. **fin**

6 Algorithme Génétique AG

L'algorithme génétique (Genetic Algorithm) est une méta-heuristique d'optimisation inspirée de la biologie évolutive. Il vise à trouver des solutions approchées à des problèmes complexes pour lesquels aucune méthode exacte n'est pratique, en faisant évoluer une population de candidats grâce à des opérateurs empruntés à la génétique.

Cet algorithme trouve de nombreuses applications dans le domaine électrique, notamment pour traiter des problématiques de stabilité, de sécurité, ainsi que de régulation de la tension et de la fréquence (vitesse).[17]

6.1 Principe de Fonctionnement de l'Algorithme AG

L'algorithme génétique GA s'appuie sur une population de taille N , composée d'individus représentant des solutions potentielles au problème étudié.

Ces individus sont codés sous forme de chromosomes, selon deux approches possibles: [18]

- **Binaire:** Séquence de 0 et 1, nécessitant des opérations d'encodage et de décodage.
- **Réel:** Valeurs directement exploitables x_i , évitant les biais de discrétisation.

La population initiale est générée aléatoirement au sein de limites définies, puis évaluée par une fonction de fitness selon l'objectif:

- **Pour un objectif de minimisation:** La fonction de fitness est:

$$f(x) = \frac{1}{1 + g(x)}$$

- **Pour un objectif de maximisation:** La fonction de fitness devient:

$$f(x) = g(x) \quad \text{ou} \quad f(x) = g(x) + C$$

Où:

- $g(x)$ est la fonction coût à minimiser.
- $f(x)$ la fitness.
- C une constante pour éviter les valeurs négatives.

Sur la base de ces évaluations, certaines solutions sont sélectionnées comme des **parents** pour former la nouvelle population **enfants** p_i .

Les méthodes de sélection les plus courantes sont:

- **Roulette proportionnelle**, probabilité de sélection:

$$p_i = \frac{f_i}{\sum_{k=1}^N f_k}$$

- **Sélection par rang**, probabilité linéaire selon le rang r_i :

$$p_i = \frac{2(N - r_i + 1)}{N(N + 1)}$$

Les opérateurs génétiques appliqués pour produire les enfants sont:

-
1. **Croisement**, est la combinaison de deux chromosomes parents P_1, P_2 pour créer des enfants E_1 et E_2 avec un taux de croisement P_c généralement entre 60 et 90%.

Le croisement peut être:

- **Croisement à un point**: Échange des segments après un point de coupure $c \in [1, L - 1]$:

$$E_1 = [P_1^{(1:c)}, P_2^{(c+1:L)}], \quad E_2 = [P_2^{(1:c)}, P_1^{(c+1:L)}]$$

- **Croisement uniforme**: Chaque gène j est choisi aléatoirement chez P_1 ou P_2 :

$$E_1[j] = \begin{cases} P_1[j] & \text{proba } 0.5 \\ P_2[j] & \text{proba } 0.5 \end{cases}.$$

2. **Mutation**, une introduction de diversité pour éviter les optima locaux, avec un taux de mutation P_m faible (0,1–5%):

- **Si le codage binaire**, on utilise bit-flip:

$$b'_j = \begin{cases} 1 - b_j & \text{proba } P_m, \\ b_j & \text{proba } 1 - P_m \end{cases}$$

Où:

- b_j : un gène binaire $\in \{0,1\}$

- **Pour le codage réel**, on applique une perturbation gaussienne:

$$x'_j = x_j + \mathcal{N}(0, \sigma)$$

Où:

- x_i , un gène réel.
- σ détermine l'amplitude des modifications.

Les enfants ainsi générés forment la nouvelle population selon la stratégie de remplacement choisie:

- **Génération complète**: Remplacement total de la population par les enfants.
- **Remplacement partiel**: Seuls les k meilleurs enfants remplacent les k moins performants des parents.
- **Élitisme**: Copie directe des k meilleurs parents (élites) dans la nouvelle génération sans modification (pas de croisement ni mutation): $P_{t+1}[:k] = P_t[:k]$

Où:

- P_t = population à la génération t .

Cette boucle de sélection, croisement, mutation et remplacement se répète jusqu'à satisfaire un critère de convergence, par exemple:

- Nombre maximal d'itérations.
- Tolérance sur l'amélioration (stagnation).
- Atteinte d'un objectif de performance cible.

6.2 Étapes de l'Algorithme AG

L'algorithme génétique a pour objectif d'évaluer une population de solutions potentielles vers des individus de plus en plus adaptés selon le critère donné, au sein d'un processus itératif de recherche exploratoire dont les étapes sont les suivantes: [19]

1. Initialisation

Générer aléatoirement une population de N individus:

$$P^{(0)} = \{x_1^{(0)}, x_2^{(0)}, \dots, x_N^{(0)}\}$$

2. Évaluation

Pour chaque individu $x_i^{(t)}$, calculer sa fitness $f(x_i^{(t)})$;

3. Tant que le critère d'arrêt n'est pas atteint faire:

(a) Sélection

Construire une nouvelle population de parents selon une probabilité proportionnelle à la fitness selon la méthode choisi;

(b) Croisement

Pour chaque paire de parents (p_1, p_2) , appliquer l'opérateur de croisement selon la technique choisie avec probabilité p_c ;

(c) Mutation

Pour chaque gène j de chaque enfant, avec une faible probabilité p_m ;

(d) Remplacement

Former la population $P^{(t+1)}$ à partir des enfants générés;

4. Retourner la meilleure solution trouvée.

Simulation, Résultats et Discussion

7 Modèle Simulink du Groupe Électrogène Utilisé

Pour la simulation, notre groupe a adopté les paramètres suivants, extraits d'un article de référence:[12]

- Temps de réponse de l'actionneur: $T_{sm} = 0.05$ s
- Constante de temps du moteur: $T_d = 0.5$ s
- Inertie de la génératrice: $H = 1.5$ s
- Coefficient d'amortissement de la génératrice: $D = 0.667$ pu

La figure ci-après présente notre modèle de groupe électrogène, annoté avec ces valeurs numériques:

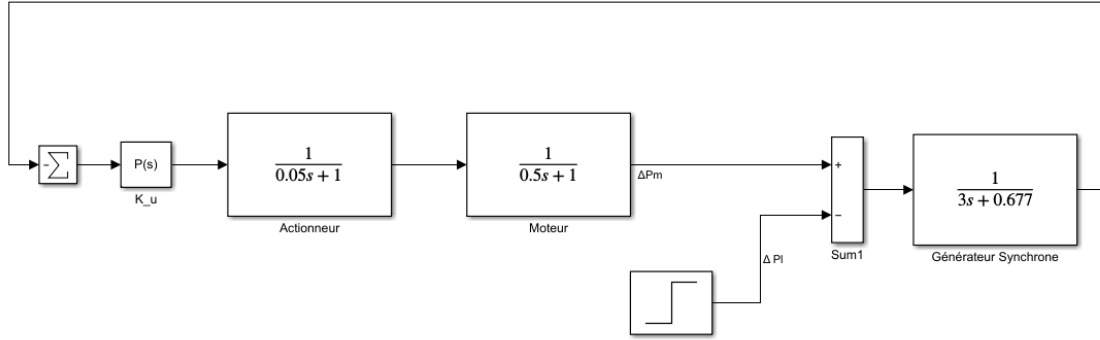


Figure 18: Modèle du groupe électrogène avec ses valeurs numériques

Nous appliquons typiquement des variations de +1% à $\pm 5\%$ (0.01 à 0.05 à 0.1 pu) de charge ΔP_l ainsi qu'une valeur de 0.1 pu ($\pm 10\%$) pour simuler des perturbations particulièrement sévères, afin de tester notre régulation de vitesse selon le critère ITAE avec deux algorithmes d'optimisation PSO et GA.

Avant de lancer les algorithmes d'optimisation, nous utilisons d'abord les réglages de la méthode de Ziegler–Nichols pour initialiser les gains du PID. Cette étape garantit des performances de contrôle acceptables dès le départ, même si elle n'élimine pas entièrement le risque de convergence vers un minimum local.

7.1 Application de la Méthode Ziegler-Nichols

Dans le cadre de l'application de la méthode de Ziegler–Nichols, notre modèle de groupe électrogène, présenté à la figure suivante, intègre un régulateur proportionnel $P(s)$ afin de déterminer les paramètres du contrôleur PID selon cette méthode:

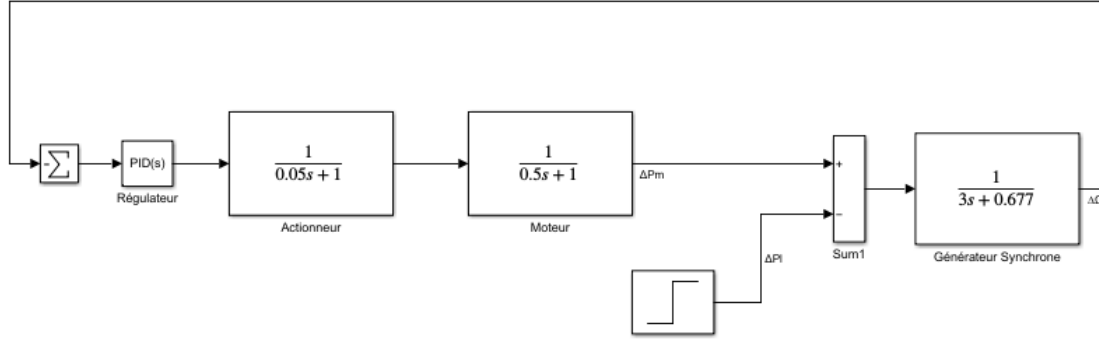


Figure 19: Schéma bloc du GE pour l'application de la méthode de Ziegler–Nichols

7.1.1 Calcul les Paramètres du Régulateur PID

La Fonction de Transfert en Boucle Fermée du Système:

Pour déterminer le gain critique K_{cu} et la période des oscillations T_u , il est nécessaire de calculer la fonction de transfert en boucle fermée du système. Celle-ci est donnée par:

$$G_{BF}(s) = \frac{G_{BO}(s)}{1 + G_{BO}(s)}$$

Où la fonction de transfert en boucle ouverte $G_{BO}(s)$ est:

$$G_{BO}(s) = K_u \cdot \frac{1}{0,05s + 1} \cdot \frac{1}{0,5s + 1} \cdot \frac{1}{3s + 0,677}$$

Avec une rétroaction unitaire, la fonction de transfert en boucle fermée devient:

$$G_{BF}(s) = \frac{G_{BO}(s)}{1 + G_{BO}(s)} = \frac{K_u}{(0,05s + 1)(0,5s + 1)(3s + 0,677) + K_u}$$

En développant le dénominateur, on obtient:

$$G_{BF}(s) = \frac{K_u}{0,075s^3 + 1,667s^2 + 3,372s + (0,677 + K_u)}$$

Détermination du Gain Critique et la Période des Oscillations

Afin de déterminer la valeur du gain critique K_{cu} :

On applique le **critère de Routh-Hurwitz**, qui permet d'identifier la valeur critique du gain proportionnel K_p pour laquelle le système présente des oscillations soutenues, marquant ainsi la limite de stabilité.[14]

1. Le polynôme caractéristique du système:

$$D(s) = 0,075s^3 + 1,667s^2 + 3,372s + (0,677 + K_u)$$

2. Tableau de Routh-Hurwitz:

$$\begin{array}{c|cc} s^3 & 0,075 & 3,372 \\ s^2 & 1,667 & 0,677 + K_u \\ s^1 & b_1 & 0 \\ s^0 & 0,677 + K_u & \end{array}$$

Tel que, le terme b_1 est:

$$b_1 = \frac{1,667 \cdot 3,372 - 0,075 \cdot (0,677 + K_u)}{1,667}$$

Donc:

$$b_1 = \frac{5,5703 - 0,075K_u}{1,667}$$

3. Conditions de stabilité:

Pour que toutes les racines aient une partie réelle négative, tous les éléments de la première colonne doivent être strictement positifs:

$$0,075 > 0 \quad (\text{toujours vrai})$$

$$1,667 > 0 \quad (\text{toujours vrai})$$

$$\frac{5,5703 - 0,075K_u}{1,667} > 0 \Rightarrow K_u < 74,27$$

$$0,677 + K_u > 0 \Rightarrow K_u > -0,677 \quad (\text{toujours vrai si } K_u > 0)$$

Alors, le système est asymptotiquement stable si:

$$0 < K_u < 74,27$$

Le gain critique est donc $K_{cu} = 74,27$

Afin de déterminer la valeur du période des oscillations T_u :

On calcule la fréquence d'oscillation du système avec le gain critique. On utilise le polynôme de degré 2 de la table de Routh, puis on isole ω_c en remplaçant $s = j\omega_c$.

1. Le polynôme de degré 2 de la table de Routh:

$$P(s) = 1,667s^2 + (0,677 + K_{cu})$$

Pour $K_{cu} = 74,27$, alors:

$$P(s) = 1,667s^2 + 74,947$$

En remplaçant $s = j\omega_c$, on obtient:

$$P(s) = 1,667j\omega_c^2 + 74,947$$

2. Calcul de la pulsation critique ω_c :

On isole ω_c :

$$1,667j\omega_c^2 + 74,947 = 0 \Rightarrow \omega_c^2 = \frac{74,947}{1,667} \Rightarrow \omega_c = \sqrt{\frac{74,947}{1,667}}$$

$$\omega_c \approx 6,7 \text{ rad/s}$$

3. Calcul de la période des oscillations:

$$T_u = \frac{2\pi}{\omega_c} = \frac{2\pi}{6,7}$$

$$T_u \approx 0,94 \text{ secondes}$$

Les Valeurs de Paramètres PID:

À partir des valeurs du K_{cu} et T_u suivantes:

$$K_{cu} = 74,27 \quad \text{et} \quad T_u = 0,94 \text{ s}$$

Les valeurs de réglage PID selon les formules empiriques de Ziegler–Nichols sont:

$$K_p = 0,6 K_{cu} \quad K_p = 44,56$$

$$T_i = \frac{T_u}{2} \quad T_i = 0,47 \text{ s}$$

$$T_d = \frac{T_u}{8} \quad T_d = 0,1175 \text{ s}$$

Donc:

$$K_i = \frac{K_p}{T_i} \quad K_i \approx 94,81$$

$$K_d = K_p T_d \quad K_d \approx 5,24$$

Ainsi, les constantes du régulateur PID sont:

$$K_p = 44,56 \quad K_i = 94,81 \quad K_d = 5,24$$

7.1.2 Les Résultats de Simulation

Après avoir implémenté les valeurs du régulateur calculées précédemment, nous avons appliqué des variations de charge de 1%, 5% et 10%.

La réponse du système obtenue est illustrée dans la figure suivante:

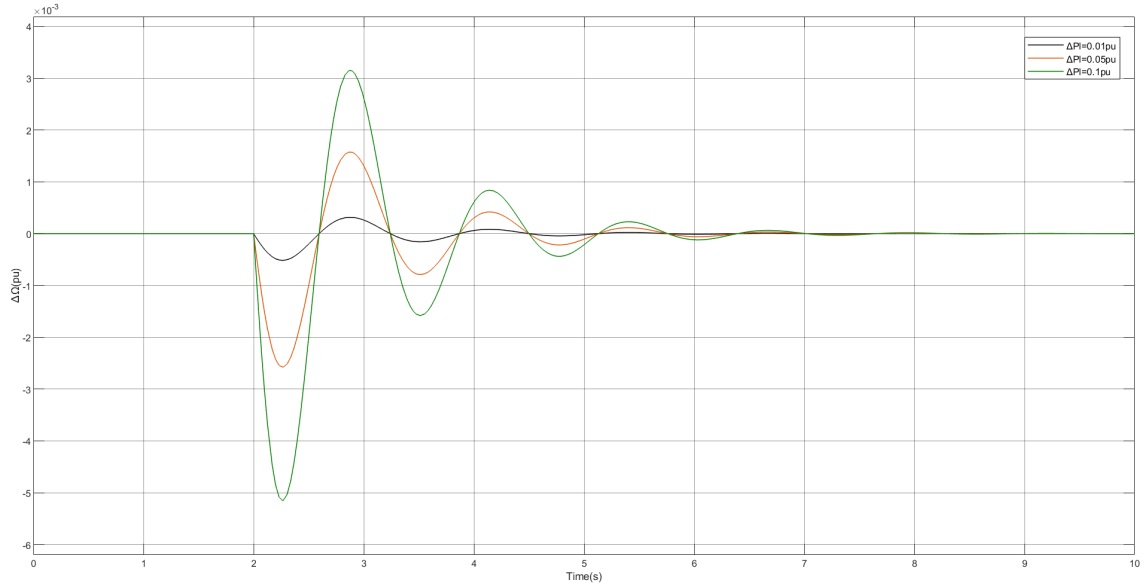


Figure 20: La réponse du système avec la méthode de ZN pour différentes perturbations

La réponse du système, réglée selon la méthode de Ziegler–Nichols et illustrée dans la figure précédente, se caractérise par un comportement sous-amorti avec des oscillations et un temps de stabilisation relativement long.

Le tableau ci-dessous récapitule, pour plusieurs niveaux de variation de charge, l'amplitude maximale de la réponse en vitesse et le temps de stabilité:

Table 3: Réponse du système pour différentes variations de charge

ΔP (pu)	$\Delta\Omega_{\max}$ (%)	Temps de stabilisation (s)
0,01	0,0315	6
0,05	0,1574	7
0,10	0,3148	8

On observe que l'amplitude de la variation de vitesse croît proportionnellement à l'intensité de la perturbation, tout en restant dans des limites acceptables pour la stabilité. En revanche, le retour à l'équilibre devient lent lorsque la charge varie de manière significative. Pour améliorer les performances dynamiques, notamment en réduisant les oscillations et le temps de stabilisation, il est nécessaire d'ajuster les gains du régulateur PID.

Dans la suite de notre étude, les valeurs issues de la méthode de Ziegler–Nichols serviront de point de départ pour une optimisation intelligente, basée sur Le critère d'erreur minimale ITAE.

7.2 Application du Critère ITAE Optimisé par PSO et GA

Pour notre étude, nous avons retenu le critère ITAE afin de garantir un compromis optimal entre rapidité de réponse et atténuation des erreurs à l'état permanent.

Le schéma Simulink intègre ce critère de la façon suivante:

- **Bloc d'intégration**, implémenté par une fonction de transfert $\frac{1}{s}$ pour calculer l'intégrale du signal d'erreur.
- **Bloc de valeur absolue**, assure que l'erreur soit toujours prise en valeur positive avant intégration.
- **Horloge (Clock)**, configurée sur la durée de simulation de 20 s.
- **Afficheur (Display)**, visualise en temps réel la valeur accumulée de l'ITAE.
- **Bloc fonction de coût**, récupère la mesure d'ITAE pour la transmettre aux algorithmes d'optimisation.

Le schéma prend alors la configuration suivante:

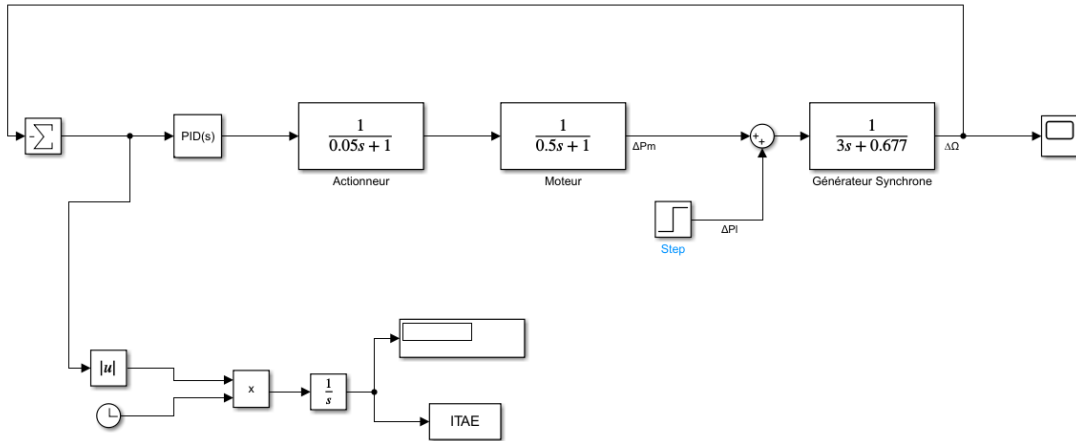


Figure 21: Schéma bloc du GE pour l'application du critère ITAE

7.2.1 Formulation du Problème d'Optimisation PID par AG

Définition du problème:

- **Variables à optimiser:** Trois variables $n_v = 3$ qui sont:
 - Gain proportionnel K_p .
 - Gain intégral K_i .
 - Gain dérive K_d .

-
- **Bornes de recherche:**
 - $\mathbf{lb} = [0 \ 0 \ 0]$, définit les bornes inférieures de chaque variable.
 - $\mathbf{ub} = [50 \ 100 \ 10]$, définit les bornes inférieures de chaque variable.
 - **Le point de départ:** $[44,56 \ 94,81 \ 5.24]$
 - **Fonction objectif:** Minimiser le critère d'erreur ITAE en maximisant la fonction de fitness:

$$\text{Fitness} = \frac{1}{1 + \text{ITAE}}$$

Choix des paramètres AG:

- **Population initiale:** $n_{pop} = 30$
- **Nombre de générations:** $\text{MaxGen} = 50$
- **Sélection**, avec la roulette proportionnelle.
- **Croisement uniforme**, avec un taux de croisement: $P_c = 0.9$
- **Mutation gaussienne**, avec un taux de mutation: $P_m = 0.1$
- **Élites:** $N_{\text{élite}} = 1$ meilleur individu est conservé à chaque génération.
- **Critère d'arrêt** Nombre maximal de générations: $\text{MaxGen} = 50$

7.2.2 Formulation du Problème d'Optimisation PID par PSO

Définition du problème:

- **Variables à optimiser:** Trois variables $n_v = 3$ qui sont:
 - Gain proportionnel K_p .
 - Gain intégral K_i .
 - Gain dérive K_d .
- **Bornes de recherche:**
 - $\mathbf{lb} = [0 \ 0 \ 0]$, définit les bornes inférieures de chaque variable.
 - $\mathbf{ub} = [50 \ 100 \ 10]$, définit les bornes inférieures de chaque variable.
 - **Le point de départ:** $[44.56 \ 94.81 \ 5.24]$
- **Fonction objectif:** Minimiser le critère de l'erreur ITAE

Choix des paramètres PSO:

- Nombre de particules: $n_{oP} = 20$
- Nombre maximal d'itérations: $Max_{Iter} = 50$
- Coefficients d'inertie:
 - Coefficient d'inertie maximale $w_{\max} = 0.9$
 - Coefficient d'inertie minimale $w_{\min} = 0.1$
- Coefficients d'accélération:
 - Le terme cognitif $c_1 = 1.5$
 - Le terme social $c_2 = 2$
- Vitesse maximale et minimale de particule:
 - Vitesse maximale $v_{\max} = 0.3(u_b - l_b)$
 - Vitesse minimale $v_{\min} = -v_{\max}$

8 Les Résultats du Critère ITAE avec ses Algorithmes d'Optimisation

Les réponses du système optimisé à l'aide des algorithmes PSO et AG, pour différents niveaux de perturbation:

Pour une perturbation de 0.01 pu:

Le système a réagi à une faible perturbation de 0.01 pu et s'est stabilisé, comme le montrent les figures ci-après:

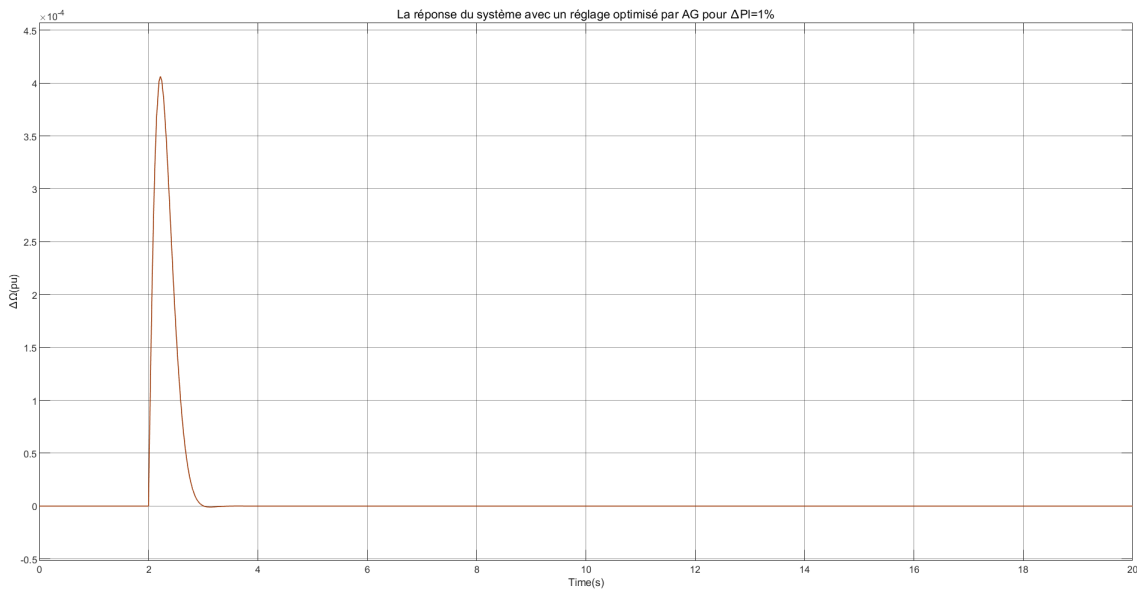


Figure 22: Réponse du système avec un réglage optimisé par PSO pour $\Delta P_l = 1\%$

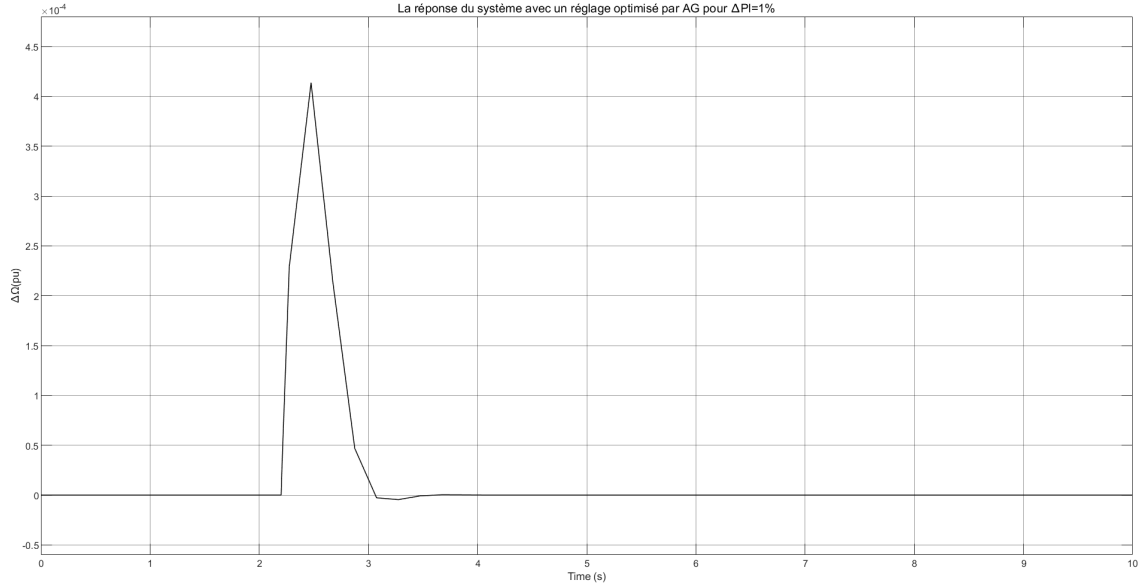


Figure 23: Réponse du système avec un réglage optimisé par AG pour $\Delta P_l = 1\%$

Les performances du système face à cette faible perturbation peuvent être résumées dans le tableau suivant:

Table 4: Les performances du système face à une faible perturbation $\Delta P_l = 1\%$

Algorithme	Dépassement max (%)	Temps de stabilisation (s)
PSO	0.04%	0.5s
AG	0.04%	1s

Pour une perturbation de 0.05 pu

Le système réagit à une perturbation moyenne de 0.05 pu et se stabilise de manière similaire à une faible perturbation. Les figures suivantes illustrent la réponse obtenue:

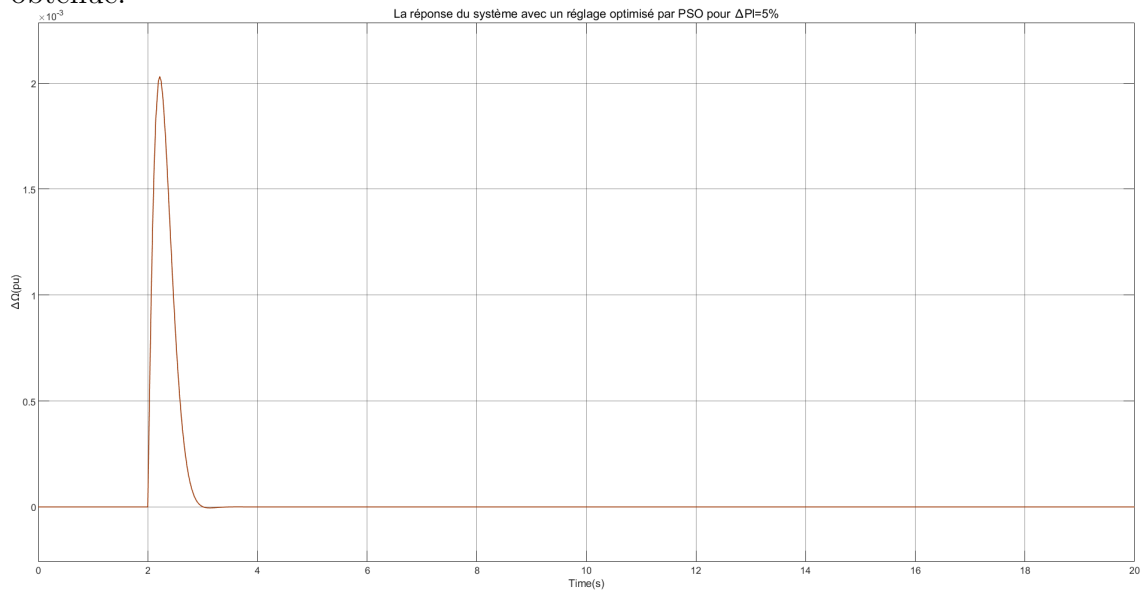


Figure 24: Réponse du système avec un réglage optimisé par PSO pour $\Delta P_l = 5\%$

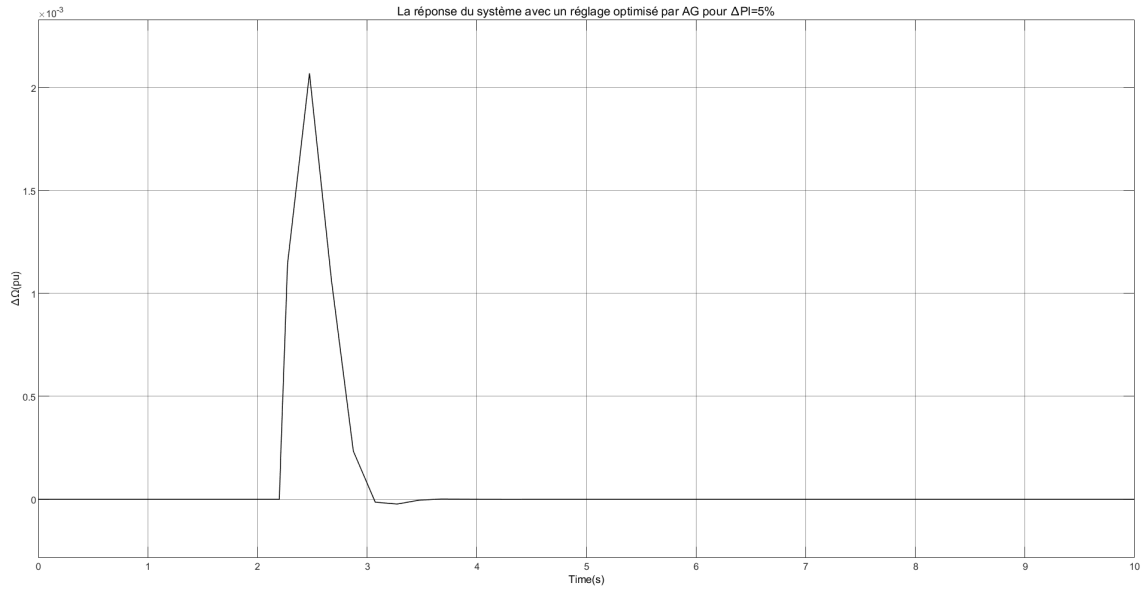


Figure 25: Réponse du système avec un réglage optimisé par AG pour $\Delta P_l = 5\%$

Les performances du système face à une moyenne perturbation peuvent être résumées dans le tableau suivant:

Table 5: Les performances du système face à une moyenne perturbation $\Delta P_l = 5\%$

Algorithme	Dépassement max (%)	Temps de stabilisation (s)
PSO	0.2%	0.5s
AG	0.2%	1s

Pour une perturbation de 0.1 pu

Le système réagit également à une forte perturbation et parvient à se stabiliser de manière similaire aux autres niveaux. Les figures suivantes illustrent la réponse du système:

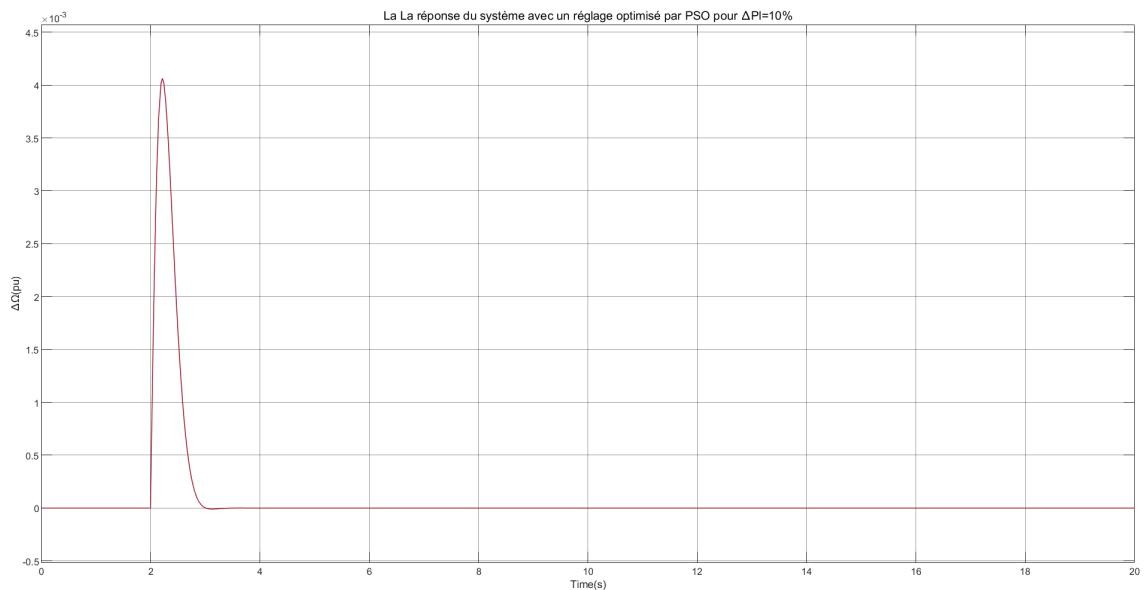


Figure 26: Réponse du système avec un réglage optimisé par PSO pour $\Delta P_l = 10\%$

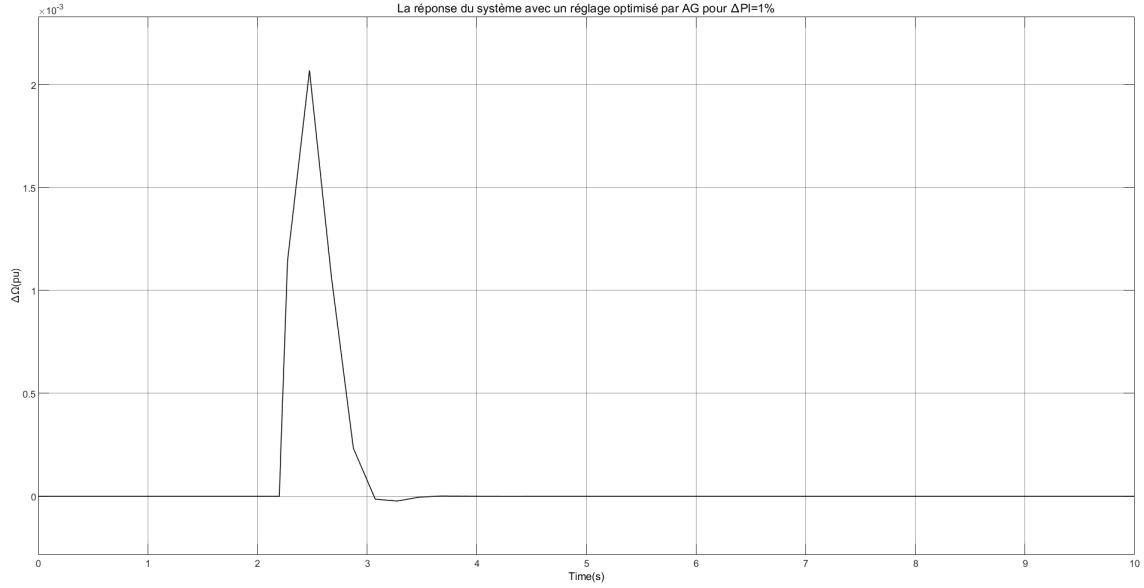


Figure 27: Réponse du système avec un réglage optimisé par AG pour $\Delta P_l = 10\%$

Les performances du système face à une forte perturbation peuvent être résumées dans le tableau suivant:

Table 6: Les performances du système face à une forte perturbation $\Delta P_l = 10\%$

Algorithme	Dépassement max (%)	Temps de stabilisation (s)
PSO	0.4%	0.5s
AG	0.4%	1s

9 Discussion

Le système, réglé de manière optimale à l'aide des algorithmes PSO et AG, réagit efficacement à tous les niveaux de perturbation, avec des oscillations réduites et un temps de réponse rapide, avec:

- Un dépassement modéré, avec un pic d'amplitude est également élevé a chaque niveau de perturbation, mais toujours il reste acceptable dans la marge de la régulation de vitesse.
- Le retour à l'état d'équilibre est rapide ($< 4s$), ce qui est un bon indicateur de performance dynamique.

En comparant les deux algorithmes d'optimisation intelligente, appliqués à une perturbation initiale et initialisés avec les paramètres obtenus par Zighlas Nicholas pour le réglage:

- L'optimisation par PSO permet un retour à l'équilibre plus rapide et plus précis, tandis que AG présente une réponse plus lente avec plus d'oscillations.
- Les deux méthodes produisent un pic similaire en termes d'amplitude, mais la gestion du transitoire diffère. AG conserve une oscillation plus persistante, tandis qu'PSO amortit mieux ces fluctuations.

Chapitre 3:

Fondements Théoriques des Systèmes SCADA et du Protocole Modbus RTU

1 Présentation Générale du Système SCADA

1.1 Définition d'un Système SCADA

Le terme SCADA, acronyme de *Supervisory Control And Data Acquisition*, désigne un système informatique dédié à la supervision, au contrôle et à l'acquisition de données en temps réel.

Il repose sur une combinaison de solutions logicielles et d'équipements matériels, permettant de surveiller en temps réel l'état d'un processus, de collecter des informations à partir de dispositifs de terrain répartis sur le site, puis de les transmettre à un poste de supervision, généralement un ordinateur central (local ou distant), chargé d'assurer le pilotage global de l'installation. Ce poste peut être assisté par d'autres stations secondaires pour une gestion collaborative et décentralisée.

Les interfaces SCADA offrent en général une représentation graphique ou synoptique de l'état du système, facilitant ainsi l'analyse des informations, la détection des anomalies et la prise de décision par les opérateurs.[20][21]

1.2 Éléments Constitutifs d'un Système SCADA

Le fonctionnement d'un système SCADA repose sur une chaîne coordonnée de composants matériels et logiciels interconnectés à travers des réseaux de communication, organisés en plusieurs niveaux. Les principaux éléments constitutifs sont les suivants:

Capteurs et Actionneurs

Les capteurs représentent les dispositifs de base utilisés pour effectuer des mesures physiques directement sur le procédé industriel. Ils permettent de relever, en continu ou à intervalles réguliers, des grandeurs électriques et mécaniques afin de caractériser l'état du système et détecter d'éventuels écarts par rapport aux valeurs de consigne.

Les actionneurs, quant à eux, assurent l'exécution des ordres de commande issus du système de contrôle. Ils interviennent physiquement sur le procédé pour modifier son comportement, en exécutant des actions telles que l'ouverture de vannes, l'activation de relais ou le démarrage de moteurs..

Dispositifs d'Automatisation et de Télémétrie

Le pilotage local et la remontée d'informations dans un système SCADA sont assurés par un ensemble de périphériques d'automatisation.

Unité Terminale Maître (MTU)

Les MTU constituent le centre de supervision principal dans un système SCADA. Elles regroupent un ensemble de ressources matérielles et logicielles, notamment des serveurs, des ordinateurs industriels, des automates maîtres et des infrastructures réseau, chargées de superviser et de coordonner l'ensemble du système. Ses fonctions principales sont :

- L'initiation et la gestion des communications avec les unités distantes (RTU et PLC),
- La centralisation des données collectées sur l'ensemble du réseau,
- L'enregistrement et le traitement des informations acquises (tri, mise en forme, stockage, analyse),
- L'interface avec les opérateurs via les logiciels de supervision (HMI/SCADA),
- L'interconnexion avec d'autres systèmes (ERP, GMAO, bases de données, plateformes cloud, etc.) pour une intégration dans une architecture industrielle globale.

Logiciel SCADA

Le logiciel SCADA constitue l'environnement de développement central du système de supervision. Il est généralement installé sur les serveurs de l'unité terminale maître (MTU) et assure l'ensemble des tâches logicielles liées à l'acquisition, au traitement, à l'affichage et à l'archivage des données.

De nombreux logiciels SCADA sont proposés sur le marché, parmi lesquels on peut citer Siemens WinCC, Schneider Citect, Ignition, ou encore LabVIEW, avec des niveaux de complexité, de performance et de coût très variables. Quel que soit le logiciel utilisé, certaines fonctionnalités de base doivent toujours être présentes pour garantir une supervision efficace:

- **Communication:** compatibilité avec les protocoles standards et capacité à interfacier divers équipements industriels.
- **Données de procédé:** consultation des données en temps réel et des données historiques, avec affichage des tendances, alertes et alarmes.
- **Rapports:** génération de rapports personnalisés pour le suivi de production, la qualité ou la maintenance.
- **Architecture:** prise en charge des solutions multi-utilisateurs, avec accès possible à distance via un navigateur ou une application mobile.
- **Interface homme-machine (HMI):** outils graphiques pour créer des vues synoptiques dynamiques et intuitives, facilitant l'interprétation par l'opérateur.

Interface Homme-Machine (HMI)

L'HMI représente le point d'interaction direct entre le système SCADA et l'opérateur. Elle permet de visualiser graphiquement l'état du procédé, de suivre en temps réel l'évolution des grandeurs mesurées, et d'agir sur les équipements à travers des commandes manuelles.

Conçue à l'aide du logiciel SCADA, l'HMI offre des vues synoptiques animées, des tableaux de bord, des alarmes visuelles et sonores, ainsi que des historiques de fonctionnement. Elle facilite la compréhension du système et la prise de décision rapide par l'opérateur.

1.3 Architecture d'un Système SCADA

Un système SCADA repose sur une architecture hiérarchisée composée de plusieurs couches fonctionnelles. Cette organisation vise à assurer une gestion efficace des flux d'information, depuis la collecte des données brutes sur le terrain jusqu'à leur traitement, leur visualisation et leur exploitation par les opérateurs.[22] On distingue généralement trois niveaux:

- **Couche de Données:** À la base de l'architecture se trouve la couche de données, qui regroupe l'ensemble des dispositifs en interaction directe avec le processus physique. Elle assure l'acquisition des grandeurs mesurées par les capteurs. Ces données sont ensuite collectées par les RTU ou les PLC, qui agissent comme interfaces locales intelligentes, puis transmises vers les couches supérieures pour traitement.
- **Couche de Traitement:** La couche intermédiaire joue un rôle central dans l'architecture SCADA. Elle assure la centralisation, le traitement et la supervision des données remontées depuis les équipements de terrain. Cette fonction est assurée par la MTU, les serveurs de supervision et les réseaux de communication internes.
La MTU coordonne l'ensemble du système, traite les données, déclenche les alarmes, effectue les calculs nécessaires et alimente le logiciel SCADA. Elle permet également une diffusion optimisée de l'information vers les stations opérateurs, tout en limitant le trafic sur le réseau.
- **Couche Client:** Au sommet de l'architecture se trouve la couche client, qui assure l'interface entre le système SCADA et les opérateurs. Elle repose sur des postes de supervision équipés de logiciels SCADA et d'interfaces homme-machine (HMI), permettant de visualiser l'état du système, de recevoir les alertes et d'agir en temps réel. Cette interaction directe facilite la prise de décision, la maintenance et la gestion globale du procédé supervisé.

1.4 Fonctions Principales

Grâce à l'interconnexion entre les équipements de terrain, les unités de traitement et les interfaces utilisateurs, un système SCADA assure de manière coordonnée plusieurs fonctions clés, parmi lesquelles:[22]

- **Fonction d'Acquisition des Données:** Cette fonction constitue la base informationnelle du système. Elle consiste à collecter en temps réel les informations provenant des équipements de terrain (capteurs, automates, RTU, etc.). Ces données sont ensuite mises à disposition pour l'analyse, la visualisation et la prise de décision.
- **Fonction de Surveillance:** À partir des données acquises, le système SCADA assure la surveillance en temps réel du procédé. Cette fonction permet de détecter les anomalies, d'en identifier les causes et de proposer des solutions pour rétablir un fonctionnement normal. Elle repose sur des alarmes, des historiques d'événements et des interfaces de diagnostic intégrées au système SCADA.
- **Fonction de Commande:** Elle constitue l'interface opérationnelle entre la supervision logicielle et l'action physique sur les installations. Elle permet d'agir sur le procédé en envoyant des ordres vers les équipements de terrain (actionneurs, moteurs, vannes, disjoncteurs, etc.).

Dans un système SCADA, la communication entre les différents équipements (capteurs, RTU, PLC, HMI, etc.) repose sur des protocoles standards assurant la fiabilité et la rapidité des échanges. Ces protocoles définissent les règles de transmission des données entre les équipements de terrain et les unités de supervision.

2 Protocole ModBus

2.1 Définition

Modbus, acronyme de *Modicon Bus*, est un protocole de communication introduit en 1979 par la société MODICON, sous l'initiative de Richard E. Morley. Il a été conçu pour permettre l'échange de données entre différents équipements industriels, notamment les PLC, les capteurs, et les systèmes de supervision. Sa simplicité de mise en œuvre, sa robustesse en milieu industriel et son architecture ouverte ont favorisé son adoption à grande échelle, faisant de Modbus l'un des protocoles les plus répandus dans les systèmes d'automatisation industrielle.

2.2 Variantes du Protocole Modbus

Le protocole MODBUS a connu plusieurs déclinaisons afin de s'adapter à divers environnements de communication, qu'ils soient industriels, sécurisés ou à haute vitesse. Ces variantes se distinguent selon le type de support utilisé, la structure des messages et les exigences en matière de performance ou de sécurité.

2.2.1 Modbus sur Ligne Série

Dans ce mode, les données sont transmises bit par bit via une ligne de communication série, selon deux types principaux de transmission:[23]

1. Transmission série synchrone

- L'émetteur et le récepteur partagent une horloge commune, garantissant une cadence identique.
- La transmission est régulière, avec un flux continu de données.
- Ce mode assure une synchronisation précise entre les deux extrémités de la communication.

2. Transmission série asynchrone

- Solution plus simple et économique, couramment utilisée dans l'industrie.
- Les bits composant un mot sont transmis successivement suivant le principe de sérialisation.
- Le récepteur reconstruit le mot via une transformation série/parallèle.
- La transmission peut être irrégulière dans le temps, mais l'intervalle entre deux bits reste constant.
- Chaque trame de données est encadrée par des bits de synchronisation (Start&Stop).

Dans les communications Modbus établies via une liaison série, deux formats principaux de transmission sont généralement utilisés, à savoir:

1. Modbus ASCII (American Standard Code for Information Interchange):

Dans ce mode, les données sont initialement exprimées en format hexadécimal, puis converties en caractères ASCII pour la transmission. Chaque octet est ainsi représenté par deux caractères ASCII, ce qui facilite le décodage mais augmente la taille des trames. Ce format est particulièrement adapté aux environnements où la clarté du signal prime sur la vitesse.

2. Modbus RTU (Remote Terminal Unit):

Modbus RTU (Remote Terminal Unit) est une variante du protocole Modbus utilisée dans les communications série. Il permet l'échange de données entre

des unités de contrôle et des dispositifs de terrain.

Ce protocole se distingue par son format binaire compact, qui optimise le débit et la rapidité des transmissions tout en réduisant la surcharge de données. Modbus RTU est un protocole ouvert, largement adopté dans les environnements industriels.

2.2.2 Modbus via Wifi ou Ethernet

1. **Modbus TCP/IP (Transmission Control Protocol / Internet Protocol):**

Cette version est adaptée aux applications nécessitant une transmission fiable, grâce à un mécanisme de confirmation (handshake) garantissant la réception correcte de chaque paquet de données.

2. **Modbus UDP/IP (User Datagram Protocol / Internet Protocol):**

Variante plus rapide que MODBUS TCP/IP, utilisant le protocole UDP, qui ne vérifie pas la réception des données. Ce mode sacrifie la fiabilité au profit de la rapidité, et reste rarement utilisé dans les environnements industriels sensibles.

2.2.3 Extensions Avancées du Protocole Modbus

1. **Modbus Plus:**

Protocole orienté peer-to-peer, basé sur un échange entre nœuds via une gestion par jetons (token passing). Il se distingue par sa rapidité et nécessite une formation ou une licence spécifique fournie par Schneider Electric.

2. **Modbus Security:**

En réponse aux faiblesses de sécurité du MODBUS original. Il intègre des fonctionnalités de protection contre les intrusions. Parmi les extensions notables figure ModSecurity, solution open source visant à renforcer la sécurité des communications industrielles.

3 Protocole Modbus RTU

3.1 Définition

Modbus RTU (Remote Terminal Unit) est une variante du protocole Modbus utilisée dans les communications série. Il permet l'échange de données entre des unités de contrôle et des dispositifs de terrain.

Ce protocole se distingue par son format binaire compact, qui optimise le débit et la rapidité des transmissions tout en réduisant la surcharge de données. Modbus RTU est un protocole ouvert, largement adopté dans les environnements industriels.

3.2 Caractéristique du Réseau Modbus RTU:

- **Vitesse de communication:** De 1200 à 115200 bauds (vitesses les plus courantes est de 9600 ou 19200 bauds).
- **Distance de transmission:** La distance maximale de communication dépend du type de câble et de la vitesse de communication configurée.
- **Topologie de réseau:** Modbus RTU utilise une topologie en bus ou en chaîne, permettant de connecter jusqu'à 32 dispositifs (1 maître et 31 esclaves) sur un réseau. Au-delà, des répéteurs ou convertisseurs sont nécessaires pour étendre la portée ou ajouter plus d'appareils.[24]

3.3 Mode de Communication Maître/Esclave

Le protocole Modbus RTU repose sur une architecture maître/esclave, qui suit les principes suivants:[25]

- **Communication unidirectionnelle:** L'appareil maître initie toutes les communications, et les appareils esclaves ne répondent qu'aux demandes du maître.
- **Mécanisme d'interrogation:** Le dispositif maître interroge en permanence chaque dispositif esclave pour obtenir des données ou des mises à jour d'état, ce qui garantit la disponibilité d'informations actualisées.
- **Temps de réponse déterministes:** Étant donné que l'appareil maître contrôle la communication, les temps de réponse peuvent être prévisibles et cohérents.

Le tableau suivant présente une comparaison des rôles du maître et de l'esclave dans un réseau Modbus RTU:

Table 7: Comparaison des rôles du Maître et de l'Esclave dans une communication Modbus

Caractéristiques	Maître	Esclave
Rôle principal	- Initie la communication.	- Répond uniquement aux requêtes.
Fonctions possibles	- Lire des registres. - Écrire des valeurs. - Attendre une réponse.	- Répondre à une lecture. - Exécuter une commande. - Envoyer un accusé de réception.
Limitations	- Ne peut pas écouter sans requête. - Ne reçoit pas d'infos spontanées.	- Ne communique jamais seul. - Ne prend pas d'initiative.
Exemples d'équipements	- PC industriel. - PLC. - HMI. - Système SCADA. - Logiciels (LabVIEW, MATLAB).	- Capteurs Modbus. - Variateurs. - Relais de protection. - Modules E/S. - Groupes électrogènes.

Deux modes d'adressage sont pris en charge:[24]

1. **Mode broadcast:** Le maître peut également envoyer un message de diffusion (broadcast) à tous les esclaves simultanément. Ce type de message n'obtient pas de réponse de la part des esclaves, il est utilisé pour leur envoyer des commandes communes.

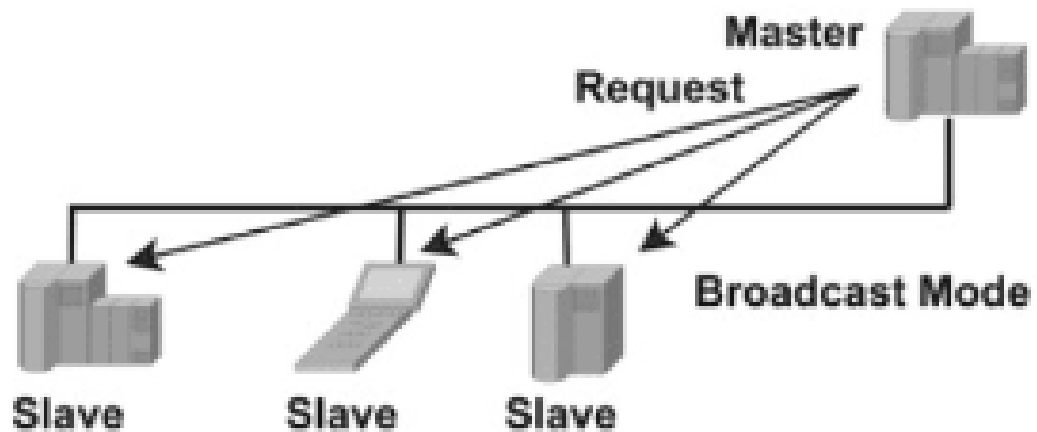


Figure 28: Mode broadcast

2. **Mode unicast:**

La communication est dirigée vers un seul esclave qui exécute la commande et renvoie une réponse.

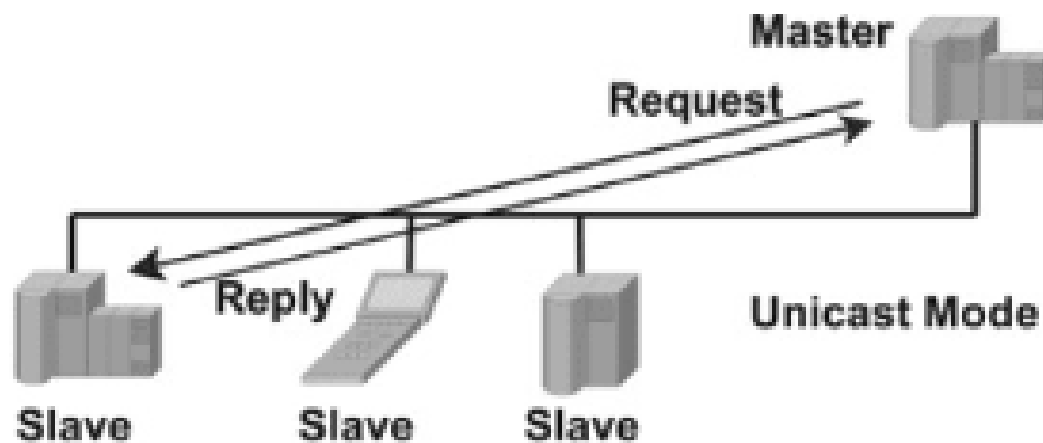


Figure 29: Mode unicast

3.4 Interfaces de Communication Série de Modbus RTU

Le protocole Modbus RTU repose sur une communication série, qui s'effectue via différentes interfaces électriques normalisées. Ces interfaces définissent le support physique de transmission des données entre les équipements:[23]

3.4.1 RS232

Le RS232 est l'interface série la plus ancienne et la plus répandue. Elle est de type point à point, c'est-à-dire qu'elle ne permet la communication qu'entre un seul maître et un seul esclave.

- Communication full duplex (émission et réception simultanées).
- Composée de 3 fils Tx, Rx et GND.
- Distance maximale: 15 mètres.
- Vitesse: jusqu'à 115 kbps.
- Sensible aux parasites et interférences, donc peu fiable dans les environnements industriels bruyants.

3.4.2 RS422

Le RS422 est une amélioration du RS232, conçu pour des transmissions plus rapides et sur de plus longues distances.

- Communication full duplex.
- Utilise 2 paires différentielles pour transmettre les signaux.
- Permet de connecter 1 émetteur et jusqu'à 10 récepteurs.
- Distance maximale: jusqu'à 1200 m.
- Vitesse: jusqu'à 10 Mbps.
- Utilisé dans certains équipements industriels et informatiques.

3.4.3 RS485

Le RS485 est l'interface la plus adaptée aux réseaux industriels utilisant Modbus RTU. Elle permet la communication multipoint sur une ligne unique, ce qui en fait la solution privilégiée pour les topologies maître/esclaves.

- Communication half duplex sur 2 fils (ou full duplex sur 4 fils).

- Supporte jusqu'à 32 périphériques sans répéteurs (et jusqu'à 247 avec répéteurs dans le cas de Modbus).
- Distance maximale: jusqu'à 1200 m.
- Immunité élevée contre les bruits et interférences industrielles.
- Possibilité de communication entre esclaves (sous certaines implémentations).
- Topologie multidrop: plusieurs nœuds connectés à la même ligne.

3.5 Structure de la Trame Modbus RTU

Dans le protocole Modbus RTU, les échanges de données entre maître et esclaves s'effectuent sous forme de trames structurées, transmises via une liaison série. Ces trames respectent une organisation standardisée basée sur l'Application Data Unit (ADU).

Le protocole définit une unité de données de protocole (PDU), indépendante des couches physiques sous-jacentes. Lorsqu'il est adapté à un support de communication spécifique, cette PDU est encapsulée dans une ADU, qui peut inclure des champs supplémentaires pour répondre aux exigences du média utilisé.[22]

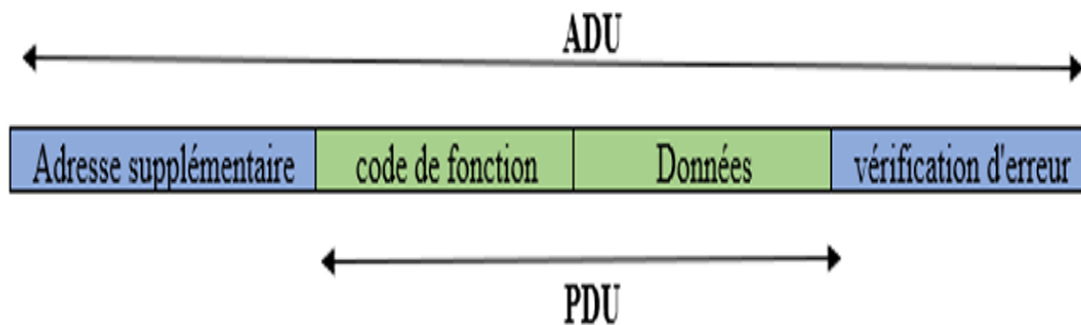


Figure 30: Trame générale de Modbus

La trame Modbus RTU est constituée comme suit:

- Adresse de l'esclave (1 octet): identifie l'équipement cible de la requête.
- PDU (Protocol Data Unit), comprenant:
 - Code de fonction (1 octet): indique l'opération à effectuer (lecture, écriture, etc.).
 - Données associées (N octets): paramètres nécessaires à l'exécution de la commande.

- CRC (Cyclic Redundancy Check) (2 octets): champ de contrôle d'intégrité, permettant de détecter les erreurs de transmission.

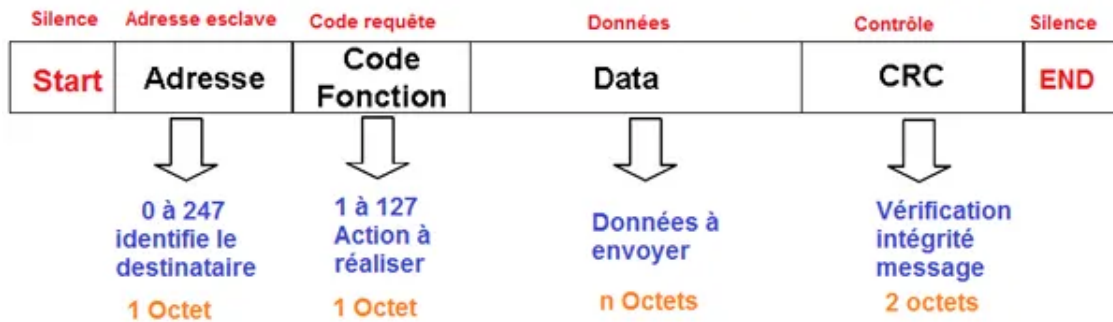


Figure 31: Structure trame Modbus RTU

Sur le plan de la transmission:

- L'affichage de la trame se fait en caractères hexadécimaux (1 octet = 2 caractères hex).
- Lorsqu'elle est envoyée sur la ligne série, chaque octet est converti en 11 bits (1 bit de start, 8 bits de données, éventuellement 1 bit de parité, et 1 bit de stop).

Sans contrôle de la parité:

Start	B0	B1	B2	B3	B4	B5	B6	B7	Stop	Stop
-------	----	----	----	----	----	----	----	----	------	------

Avec contrôle de la parité:

Start	B0	B1	B2	B3	B4	B5	B6	B7	Parité	Stop
-------	----	----	----	----	----	----	----	----	--------	------

Figure 32: Comparaison des trames avec et sans contrôle de la parité

3.6 Transaction Modbus RTU

Dans le protocole Modbus RTU, une communication repose sur une transaction structurée entre un maître et un ou plusieurs esclaves. Le maître est l'unique initiateur, il construit et transmet une trame de requête contenant les informations nécessaires à l'exécution d'une commande. L'esclave concerné analyse cette trame, exécute l'action demandée (s'il le peut), puis retourne une réponse.

3.6.1 Déroulement d'une Transaction

En Modbus RTU, la transaction débute toujours par l'envoi d'une trame par le maître, contenant l'adresse de l'esclave, un code de fonction, des données associées, et un CRC pour vérifier l'intégrité.

Si l'esclave reçoit une trame valide, il exécute l'instruction et répond avec les données demandées ou un accusé de réception, accompagné d'un CRC recalculé.

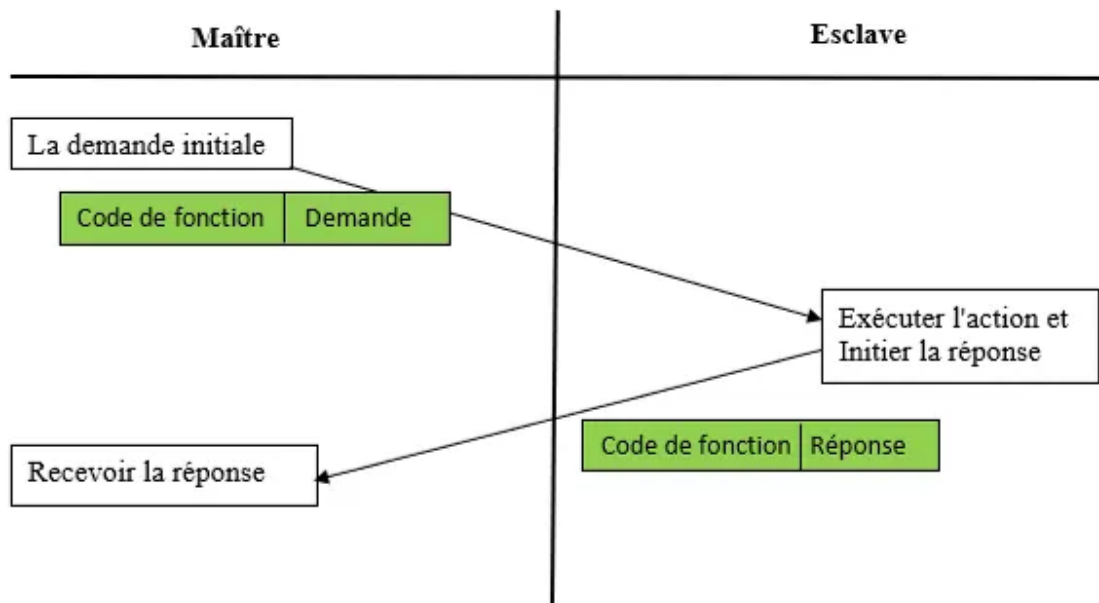


Figure 33: Transaction Modbus sans erreur

En cas de diffusion (broadcast), tous les esclaves reçoivent la trame, mais aucun ne répond, car le broadcast ne nécessite pas de retour.

3.6.2 Vérification d'Intégrité: CRC

Le CRC (Contrôle de Redondance Cyclique) est un code de 16 bits ajouté à la fin de chaque trame. Il permet de détecter les erreurs de transmission :

- Le CRC est calculé sur tout le contenu de la trame (adresse, fonction, données).
- À la réception, l'appareil recalcule le CRC et le compare à celui reçu :
 - Si les deux CRC concordent, la trame est considérée comme valide.
 - Sinon, la trame est ignorée sans réponse.

Dans ce cas, le maître peut réémettre la demande après expiration d'un timeout.

3.6.3 Codes d'Exception

Quand un esclave reçoit une trame avec bon CRC, mais que la requête est mal formée ou impossible à exécuter, il ne renvoie pas une réponse normale, mais une réponse d'exception structurée comme suit:

- Le code de fonction retourné est modifié : fonction + 0x80
- Un code d'erreur est inclus dans le champ de données

- La trame est terminée par un CRC

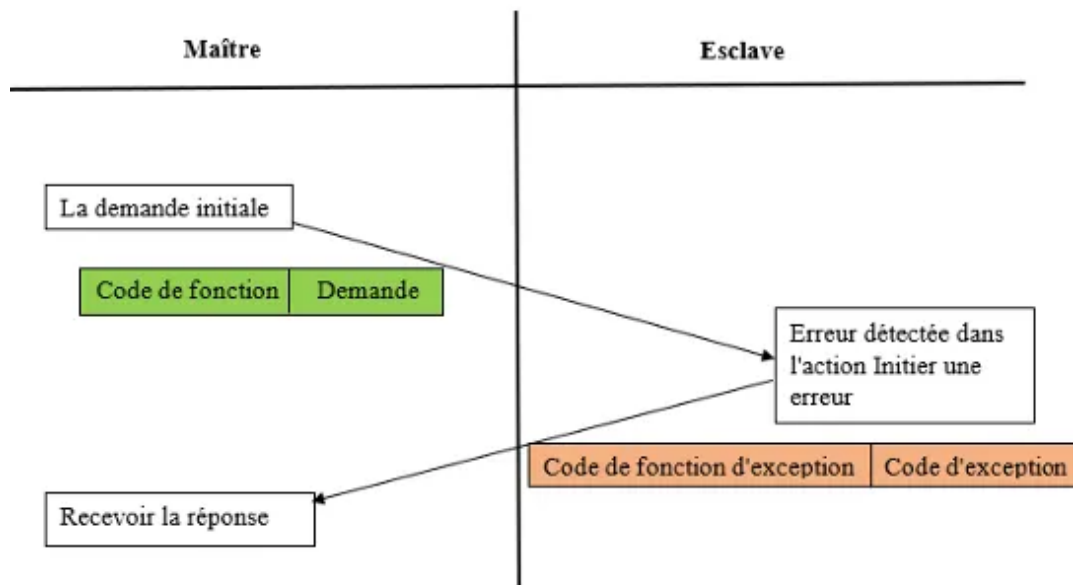


Figure 34: Transaction Modbus avec exception

3.6.4 Codes d'Erreur Modbus RTU

Lorsqu'un appareil esclave rencontre une erreur lors du traitement d'une requête, il répond avec un message d'exception contenant un code d'erreur. Ces codes fournissent des informations sur la nature de l'erreur. Voici quelques codes d'exception courants:

Table 8: Codes d'exception Modbus RTU

Code	Signification
0x01	Fonction illégale
0x02	Adresse de données illégale
[24] 0x03	Valeur de données illégale
0x04	Défaillance du dispositif esclave
0x05	Accusé de réception
0x06	Dispositif esclave occupé

3.7 Données Echangées via Modbus RTU

3.7.1 Type de Données

Dans le protocole Modbus RTU, la communication repose sur la lecture ou l'écriture de données dans la mémoire de l'esclave. Cette mémoire est organisée en quatre

zones principales, appelées objets Modbus, chacune associée à un type de données bien précis, avec des droits d'accès.[24]

1. **Bobines:** ce sont des éléments booléens (1 bit) dans la mémoire de l'esclave. Elles peuvent être activées (1) ou désactivées (0) par le maître. Elles servent à commander des actionneurs tels que: Relais, Moteurs et Voyants lumineux.
2. **Entrées discrètes:** elle représentent des signaux binaires accessibles en lecture seule, comme l'état de capteurs ou de boutons.
3. **Registres d'entrée:** ils stockent des valeurs mesurées (ex. : tension, température). Chaque registre fait 16 bits (2 octets) et est accessible uniquement en lecture.
4. **Registres de maintien:** ils sont utilisés pour stocker des consignes, des paramètres de configuration ou des états internes. Ils ont également une taille de 16 bits et sont accessibles en lecture et écriture.

Chaque zone est identifiée par une adresse, et les échanges se font selon le format Big-Endian, c'est-à-dire que l'octet le plus significatif est transmis en premier lors de l'envoi de données multi-octets.

3.7.2 Codes Fonctions Modbus Courants

Afin d'accéder aux différentes zones mémoire (coils, registres, etc.), le protocole Modbus RTU utilise des codes de fonction bien définis. Ces codes spécifient l'opération à effectuer (lecture, écriture...) sur une zone de données particulière.

Le tableau suivant récapitule les fonctions les plus utilisées dans l'accès aux objets Modbus:

Table 9: Fonctions Modbus RTU les plus courantes

Code	Description
01	Lire les bobines : Lit l'état des variables binaires (0 ou 1) dans un dispositif esclave (sorties).
02	Lire les entrées discrètes : Lit l'état des entrées discrètes dans un dispositif esclave, généralement utilisé pour lire des capteurs.
03	Lire les registres de maintien : Lit la valeur des registres de 16 bits, souvent pour lire des données ou paramètres.
[24] 04	Lire les registres d'entrée : Lecture seule des valeurs de mesure issues de capteurs.
05	Forcer une bobine : Active ou désactive une sortie (relais, moteur, etc.).
06	Écrire dans un registre : Écrit une valeur dans un registre de configuration.
15	Écrire plusieurs bobines : Modifie plusieurs sorties binaires en une seule commande.
16	Écrire plusieurs registres : Met à jour simultanément plusieurs registres de paramètres.

3.7.3 Codes Fonctions Modbus Avancés

Le tableau suivant présente les fonctions avancées de protocole Modbus RTU:

Table 10: Codes fonctions Modbus avancés

Code	Description
07	Lire l'état du bus : Interroge un mot de 8 bits sur l'état d'alarmes ou erreurs. Utilisé pour le diagnostic rapide.
08	Diagnostics : Permet de vérifier la liaison maître-esclave et de réinitialiser certains compteurs.
17	Lire/écrire registres : Réalise simultanément des lectures et écritures sur plusieurs registres pour améliorer l'efficacité.
[24] 20	Lire un fichier : Lit des enregistrements de fichiers dans des dispositifs compatibles, souvent utilisés dans l'automatisation.
21	Écrire dans un fichier : Permet de modifier des fichiers spécifiques dans certains dispositifs esclave.
22	Écriture masquée : Modifie certains bits d'un registre tout en conservant les autres, utile pour les changements partiels.
23	Lire/écrire registres multiples : Combine lecture et écriture dans une seule transaction pour un gain de temps.

Choix de la fonction Modbus

- Pour **lire** l'état ou les données: Utilisez les fonctions **01, 02, 03 ou 04** selon que vous travaillez avec des bobines, entrées discrètes, ou registres.
- Pour **écrire** des commandes ou paramètres: Utilisez **05 ou 06** pour une commande unique, ou **15 et 16** pour écrire sur plusieurs bobines ou registres en une seule fois.
- Pour des **diagnostics** ou des besoins avancés: Les fonctions plus avancées comme **07, 08, ou 22** sont utiles pour vérifier l'état du réseau, tester la communication ou gérer des modifications complexes.

3.8 Application Simple de la Communication Modbus RTU en Environnement Simulé

Afin d'approfondir la compréhension du protocole Modbus RTU, une simulation a été réalisée à l'aide d'outils logiciels. L'objectif de cette expérimentation est de permettre une mise en pratique concrète des notions théoriques liées à la communication maître-esclave. Elle est basée sur l'utilisation de trois logiciels principaux: (voir annexe1)

- **Modbus Poll:** le maître.
- **EasyModbus RTU Server:** l'esclave.
- **Virtual Serial Ports Emulator (VSPE):** la liaison série virtuelle.

3.9 Diagnostic et Dépannage d'un Réseau Modbus RTU

La fiabilité d'un réseau Modbus RTU repose sur une configuration correcte et un diagnostic précis en cas de dysfonctionnement. Cette section présente les approches de dépannage courantes, classées par couches et par type de symptôme, afin d'aider à localiser rapidement l'origine du problème.

3.9.1 Diagnostic de la Couche Hardware

- **Vérification du câblage RS-485:**
 - Fils A et B torsadés, sans dérivations en étoile.
 - Polarité correcte.
 - Résistances de terminaison de 120Ω aux extrémités.
- **Alimentation des équipements:**
 - Vérifier la tension et la stabilité de l'alimentation.
 - Éviter les chutes de tension ou redémarrages intempestifs.
- **Bruits électromagnétiques (EMI):**

-
- Identifier les sources de perturbation (moteurs, variateurs...).
 - Mise à la terre du blindage à un seul point.
 - Utiliser des ferrites ou isolateurs galvaniques si besoin.

3.9.2 Diagnostic au Niveau Logiciel

- **Analyse des codes d'exception:** L'esclave répond avec un code de fonction modifié (ex : 0x83) et un code d'exception (de 0x01 à 0x06).
- **Vérification des paramètres série.**
- **Correspondance des adresses:** L'adresse appelée par le maître doit correspondre à celle configurée sur l'esclave.
- **Temporisation et timing:**
 - Timeout maître adapté au temps de réponse maximal.
 - Silence minimum de 3,5 caractères entre les trames.

3.9.3 Résolution des Problèmes de Cohérence des Données

- **Adressage correct:**

Table 11: Types d'adressage Modbus

Adressage	Nom de la donnée	Code fonction
0xxxx	Bobines	0x01
1xxxx	Entrées discrètes	0x02
3xxxx	Registres d'entrée	0x04
4xxxx	Registres de maintien	0x03

- **Offset et longueur:**
 - 30001 correspond souvent à l'offset 0.
 - Vérifier le nombre d'éléments demandés.
- **Mappage des variables:** S'assurer que le type logique (BOOL, INT, REAL...) correspond bien au format réel de la donnée reçue.

3.9.4 Bonnes Pratiques pour le Dépannage

- Tester en local avec un seul maître et un seul esclave.
- Utiliser un logiciel de test Modbus (Modbus Poll, QModMaster, ModScan).

-
- Activer les journaux (logs) côté maître ou SCADA.
 - Réduire la vitesse (ex : de 115200 à 9600 bps).
 - Segmenter le réseau pour localiser un câble ou périphérique défectueux.

Chapitre 4:

Implémentation sous AGI Creator du Système SCADA

1 Présentation Générale du Logiciel AGI Creator

AGICreator est un logiciel conçu par DEIF pour la création d'interfaces homme-machine (HMI) graphiques. Basé sur un principe de « glisser-déposer », il permet de composer rapidement des pages complexes en bénéficiant des conventions et des contrôles familiers des applications Windows classiques.[26]

1.1 Présentation du Workspace

L'environnement de travail d'AGI Creator se compose de plusieurs zones aux fonctionnalités distinctes, chacune de ces zones peut être masquée, détachée ou ancrée, offrant une flexibilité maximale dans l'agencement de l'interface. Les dispositions ainsi définies sont automatiquement restaurées à l'ouverture ultérieure d'AGI Creator.

Les zones principales de l'environnement de travail sont les suivantes:

1. **Project View:** Affiche l'arborescence hiérarchique des pages, protocoles, alarmes, etc.
2. **Object View:** Liste des widgets disponibles sur la page active (boutons, voyants, graphes...).
3. **Tag Cross Reference:** Indique pour chaque balise les emplacements où elle est utilisée.
4. **Working Area:** Zone centrale d'édition, où l'on dispose et configure les éléments graphiques.
5. **Properties:** Panneau des propriétés de l'objet ou de la page sélectionnée.
6. **Widget Gallery:** Bibliothèque d'objets graphiques et symboles prêts à l'emploi, classés par type.

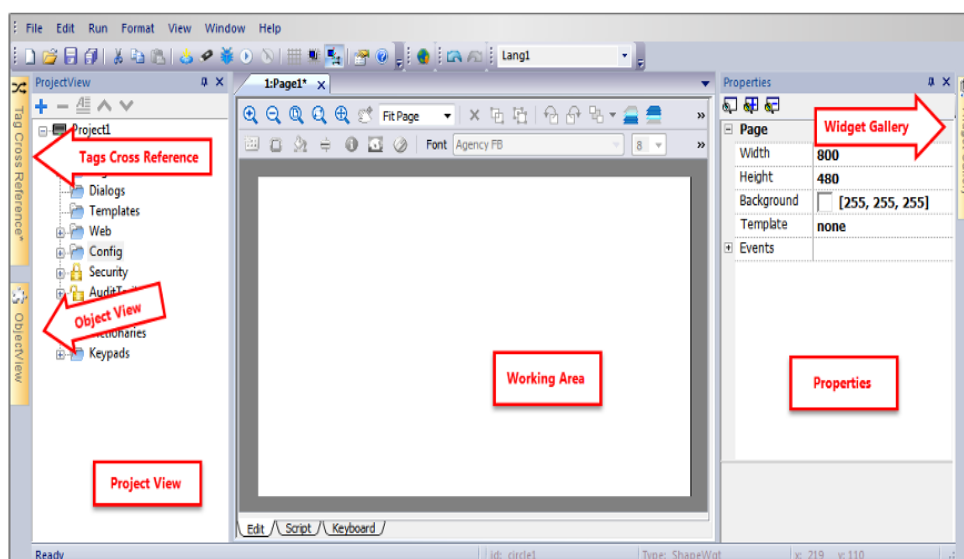


Figure 35: Interface du logiciel AGI Creator

L'utilisateur peut adapter son espace de travail en activant ou désactivant des barres d'outils via le menu **View** → **Toolbars** → **Docking Windows**. De plus, les propriétés globales du projet peuvent être modifiées à tout moment via l'onglet **Project Properties**.

1.2 Création et Gestion d'un Projet

La création d'un nouveau projet dans AGI Creator repose sur un assistant **Project Wizard** qui guide l'utilisateur à travers les étapes nécessaires à l'initialisation du projet HMI:

1. Ouvrir AGI Creator puis cliquer sur **File** → **New Project**.
2. Une fenêtre **Project Wizard** s'ouvre :
 - Saisir le nom du projet.
 - Choisir l'emplacement de sauvegarde sur le disque local.
3. Cliquer sur **Next** pour accéder à la sélection du matériel HMI :
 - Sélectionner un modèle d'écran compatible (ex. : AGI 400, AGI 500...).
 - Choisir l'orientation de l'écran (*portrait* ou *paysage*).
4. Choisir le type de structure du projet :
 - **Unified** : les mêmes pages sont utilisées à la fois pour l'écran HMI et les clients Web.
 - **Native & Web** : permet de définir séparément des pages pour l'interface locale HMI et pour l'accès Web.
 - **Custom** : structure libre avec arborescence manuelle.
5. Cliquer sur **Finish** pour finaliser la configuration du projet.

Une fois ces étapes terminées, le projet est automatiquement ouvert dans l'espace de travail. L'utilisateur peut alors créer des pages HMI, définir les balises de communication, insérer des objets graphiques et configurer les interactions avec le système supervisé.

1.3 Utilisation du Panneau Project View

Le panneau **Project View**, situé sur le côté gauche de l'interface, constitue l'arborescence hiérarchique de l'ensemble des éléments du projet. Il permet d'accéder rapidement à toutes les sections importantes du projet et d'y effectuer des opérations de création, de configuration ou de suppression.

1. Pages et fenêtres

- **Pages**: liste de toutes les pages principales du projet.

- **Dialog:** pages contextuelles (fenêtres pop-up).
- **Templates:** modèles de mise en page réutilisables sur plusieurs pages.

2. Configuration

- **Protocols:** configuration des interfaces physiques de communication.
- **Tags:** création des balises de communication (tags) associées aux adresses des variables échangées.
- **Trends:** définition des courbes de tendance pour la visualisation de données historiques.
- **Alarms:** configuration des alarmes, rapports d'événements et niveaux de gravité.
- **Events Buffer:** gestion de tampons pour stocker les alarmes ou les journaux..
- **Schedules:** planification d'actions automatiques selon des plages horaires définies.
- **Data Transfer:** transfert de données entre plusieurs dispositifs connectés.

3. Paramètres généraux

- **Recipes:** gestion de jeux de données ou configurations à appliquer dynamiquement.
- **Dictionaries:** configuration multilingue via des dictionnaires de textes.
- **Keypads:** création de claviers virtuels personnalisés pour la saisie de données.

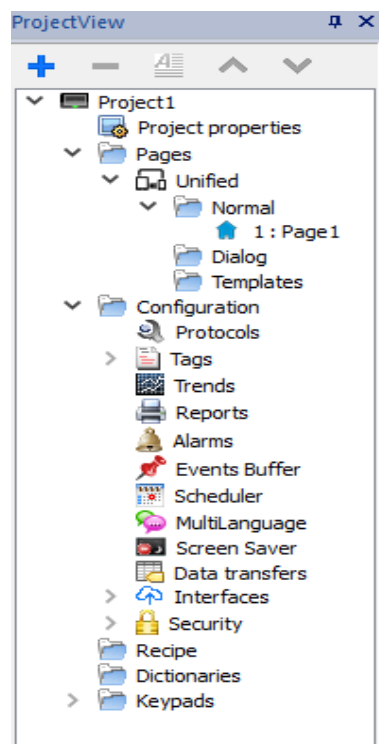


Figure 36: Panneau Project View dans AGI Creator

Chaque élément du Project View peut être développé ou replié pour faciliter la navigation. La majorité des actions s'effectue par un clic droit sur l'élément concerné. Ce panneau constitue le point d'entrée principal pour la structuration et la gestion globale du projet HMI.

1.4 Création des Protocoles de Communication

Dans AGI Creator, la configuration des protocoles permet d'établir la communication entre l'interface HMI et les équipements externes. Cette configuration se fait via le panneau Project View → Config → Protocols. L'ajout et la configuration d'un protocole s'effectuent selon les étapes suivante:

1. Dans l'arborescence Project View, accéder à Config → Protocols.

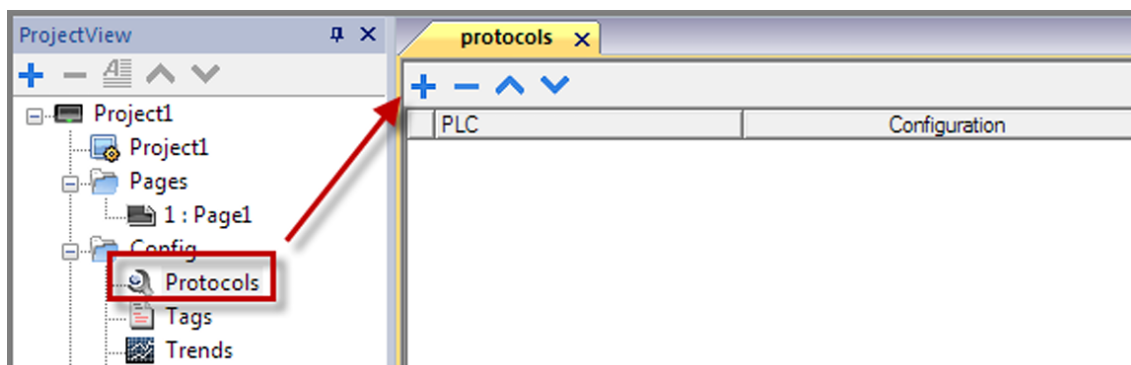


Figure 37: Ajout d'un protocole de communication dans AGI Creator

2. Cliquer sur le bouton + pour ajouter un nouveau protocole:
 - Une liste de protocoles supportés s'affiche (Modbus RTU, Modbus TCP..).
 - Sélectionner le protocole adapté à l'équipement cible.
 - Saisir les paramètres de communication requis (ID, port, débit..).
3. Cliquer sur l'icône de configuration (colonne « Configuration ») pour accéder aux paramètres détaillés.

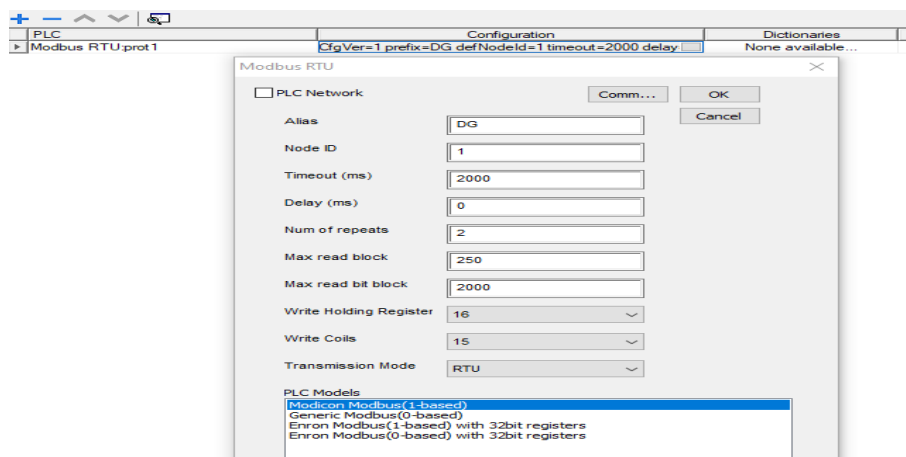


Figure 38: Modification des paramètres du protocole dans AGI Creator

4. Confirmer l'ajout et valider la configuration.

1.5 Création des Tags de Communication

Dans AGI Creator, les tags représentent des variables utilisées pour lire ou écrire des données depuis ou vers un automate ou un dispositif connecté. Chaque tag est lié à une adresse mémoire via un protocole de communication. La création et la gestion des tags s'effectuent depuis Project View → Config → Tags:

1. Cliquer sur l'icône + pour créer une nouvelle balise.
2. Donner le nom à la variable.
3. Double clic, une boîte de dialogue s'ouvre pour saisir les informations.

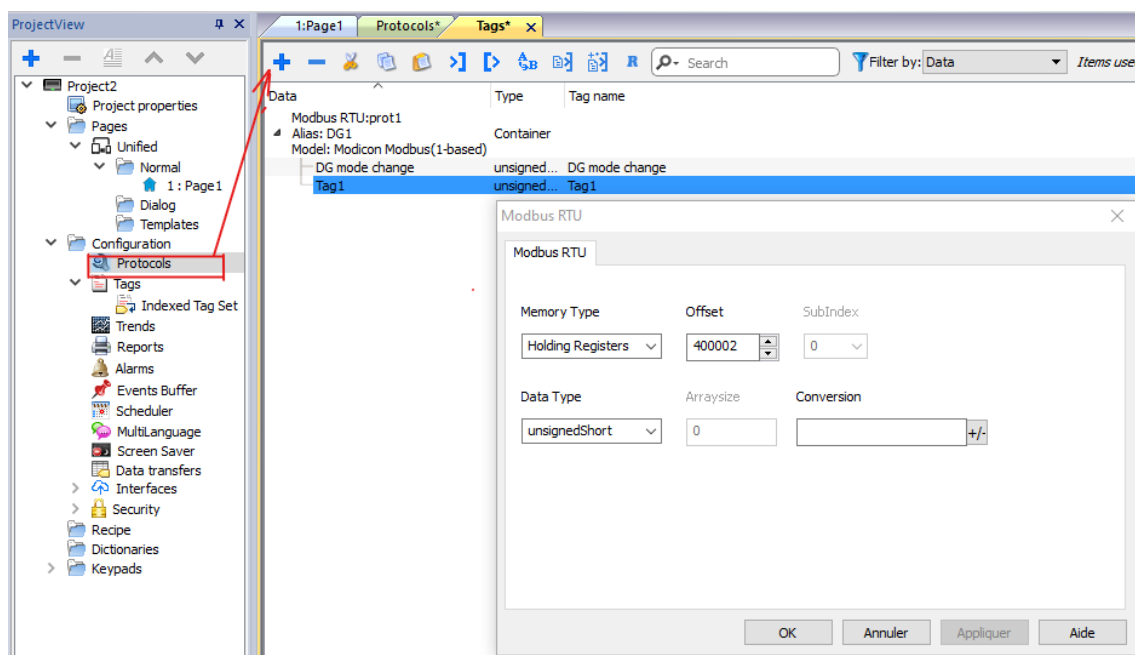


Figure 39: Création des tags dans AGI Creator

AGI Creator offre la possibilité d'importer tags à partir de fichiers au format CSV ou XML, généralement issus de dictionnaires générés dans des logiciels tiers tels qu'Excel ou des environnements de programmation d'automates. L'importation se déroule en deux étapes:

1. Intégration des définitions dans un dictionnaire.
2. Importation des balises de ce dictionnaire vers le projet.

À l'inverse, les balises du projet peuvent être exportées au format XML via le bouton **Export Tags** pour être réutilisées ou modifiées à l'extérieur.

1.6 Insertion d'Objets Graphiques et Liaison avec un Tag de Communication

La création de l'interface graphique dans AGI Creator repose sur l'ajout et la configuration d'objets HMI (ou widgets), disponibles dans la Widget Gallery. Cette

bibliothèque regroupe une variété d'éléments prédéfinis tels que des jauges, voyants, compteurs, boutons, zones de texte, etc., permettant de représenter visuellement les données du système supervisé.

1. Pour insérer un widget, il suffit d'accéder à la Widget Gallery, puis de sélectionner l'élément souhaité.



Figure 40: Panneau d'objets graphiques dans AGI Creator

2. Le widget peut ensuite être glissé-déposé directement dans la zone de travail. Une fois positionné, il peut être redimensionné ou stylisé selon les besoins via le panneau Properties. Certains widgets proposent plusieurs styles visuels, accessibles depuis des boutons spécifiques dans la galerie.
3. Les objets complexes, tels que les boutons combinant plusieurs éléments (texte, cadre, icône...), peuvent être personnalisés individuellement. Chaque sous-composant est accessible dans le panneau Object View, sans qu'il soit nécessaire de dissocier manuellement le widget complet.

L'étape suivante consiste à connecter le widget à un tag de communication, afin qu'il reflète une valeur réelle provenant d'un PLC ou qu'il permette d'agir sur celui-ci. La procédure est la suivante:

1. Dans le panneau **Properties**, cliquer sur l'icône + située à côté du champ **Value** ou de toute autre propriété pertinente.
2. Choisir la source de données: Tag, System, Widget ou Recipe.
3. Sélectionner un protocole de communication puis le tag souhaité.
4. Définir le type d'accès, lecture seule, écriture seule ou lecture/écriture.
5. En cas de tag de type tableau, spécifier un index, soit fixe, soit indirect via un autre tag.
6. Valider la configuration en cliquant sur OK.

Une fois la configuration validée, le widget est fonctionnel et interagit dynamiquement avec le processus surveillé, affichant les données en temps réel ou permettant des actions sur le système à travers l'interface utilisateur.

1.7 Animation Graphique des Éléments

L'animation graphique permet de représenter visuellement l'état de certains équipements à travers des changements d'apparence dynamiques.

1. Insérer un widget de type **Multi-State Image** depuis la **Widget Gallery** sur la page souhaitée.

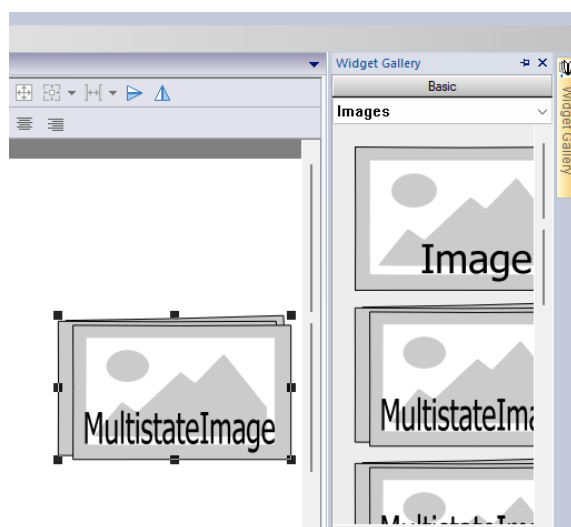


Figure 41: Objet Multi-State Image dans AGI Creator

2. Dans les propriétés du widget, cliquer sur **Image** pour ouvrir la fenêtre de sélection des états.
3. Sélectionner plusieurs fichiers SVG correspondant à différents états de fonctionnement.
4. Chaque image est automatiquement associée à un index (0, 1, 2, ...) qui représentera une valeur du tag à surveiller.
5. Pour simuler un changement d'état, insérer un **Two-State Switch** sur la même page.
6. Configurer les propriétés du commutateur (labels, couleurs, valeur 0/1...).
7. Associer le widget **Multi-State Image** à un tag de communication, de manière à refléter la valeur du tag contrôlé.
8. Associer également le **Two-State Switch** au même tag pour pouvoir modifier son état depuis l'interface.

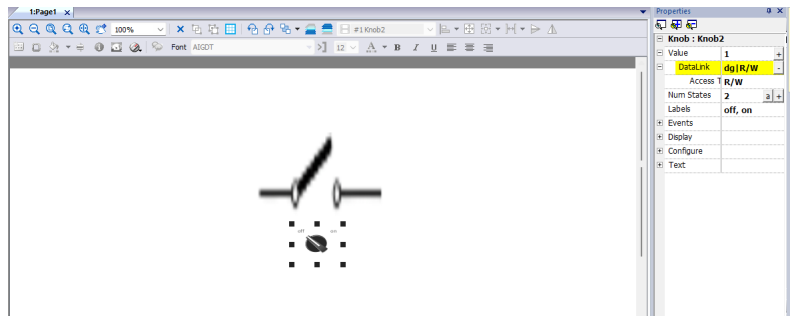


Figure 42: Animation graphique d'un interrupteur dans AGI Creator

9. Pour tester l'animation, lancer le mode simulation en accédant au menu **Run** → **Start Simulation**.

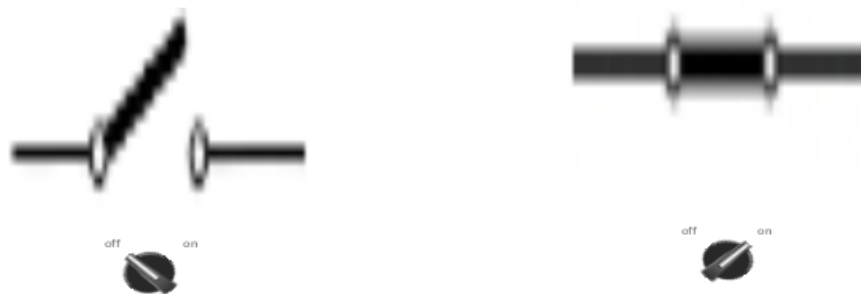


Figure 43: Simulation de l'interrupteur On-Off

Cette méthode permet d'obtenir une interface dynamique et intuitive, offrant une représentation visuelle fidèle de l'état réel du système supervisé.

1.8 Écriture dans un Tag

Cette fonctionnalité permet de modifier la valeur d'un tag (bit ou mot) depuis l'interface HMI, par l'intermédiaire d'un widget interactif. Elle s'effectue selon les étapes suivantes:

1. Insérer un widget de type **0/1 Bit Switch** depuis la bibliothèque graphique.
2. Personnaliser l'apparence du commutateur via le panneau **Properties**, afin de l'adapter à l'élément commandé.
3. Ajouter un widget **Numeric Display** qui permettra d'afficher en temps réel la valeur actuelle de tag.
4. Pour renforcer la lisibilité, il est également possible d'ajouter une **Multi-State Image** représentant visuellement l'état du dispositif qui sera liée au même tag.
5. Associer la propriété **Value** du widget **Bit Switch** à un tag de communication défini au préalable.
6. Lancer la simulation pour valider le bon fonctionnement de l'ensemble.

Cette méthode permet de commander des éléments à distance via l'interface HMI, tout en assurant un retour visuel clair sur l'état des variables contrôlées. Elle illustre l'un des principes fondamentaux du contrôle-commande dans un environnement SCADA.

2 Simulation du Système SCADA en Environnement Virtuel

Dans le cadre de la mise en œuvre du système SCADA, une simulation en environnement virtuel a été réalisée à l'aide du logiciel AGI Creator. Cette étape vise à valider la communication entre l'interface SCADA et les équipements simulés, en reproduisant des conditions proches du réel. Elle permet de vérifier le bon fonctionnement de l'ensemble du système avec des données représentatives, tout en s'assurant que la transition vers l'installation réelle pourra s'effectuer sans modification, ni au niveau des balises, ni de la structure de l'interface.

2.1 Rôle des Outils Logiciels dans la Simulation

Trois outils logiciels complémentaires ont été utilisés pour mettre en œuvre la simulation:

Table 12: Rôles des différents outils dans la simulation SCADA

Outil	Rôle
AGI Creator	Sert d'interface SCADA et agit comme maître Modbus RTU . Il génère les requêtes de lecture/écriture.
EasyModbus Server	Émule un esclave Modbus en simulant les réponses d'un automate/contrôleur.
VSPE	Permet de créer une liaison série virtuelle RS-485 entre les deux logiciels via une paire de ports COM.

2.2 Mise en Place de la Communication

La communication a été établie à travers plusieurs étapes, réparties selon les outils utilisés:

1. **Création des ports COM virtuels via VSPE:**

Deux ports virtuels (COM1 et COM2) ont été créés pour simuler une liaison maître-esclave locale.

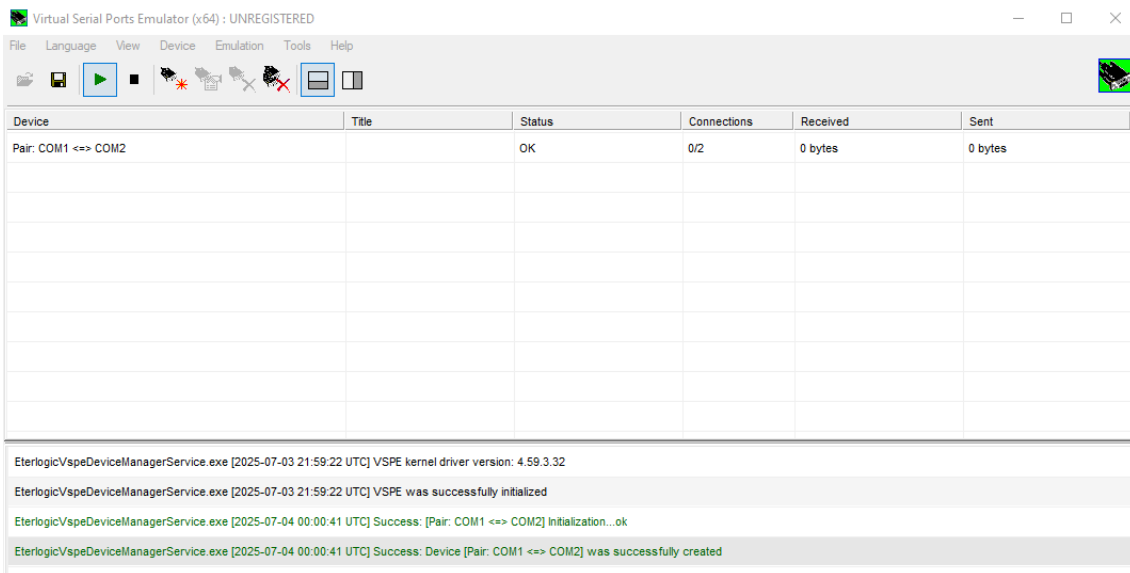


Figure 44: Création des ports COM virtuels via VSPE

2. Affectation du port à EasyModbus:

EasyModbus est configuré comme esclave RTU, avec affectation du port COM1.

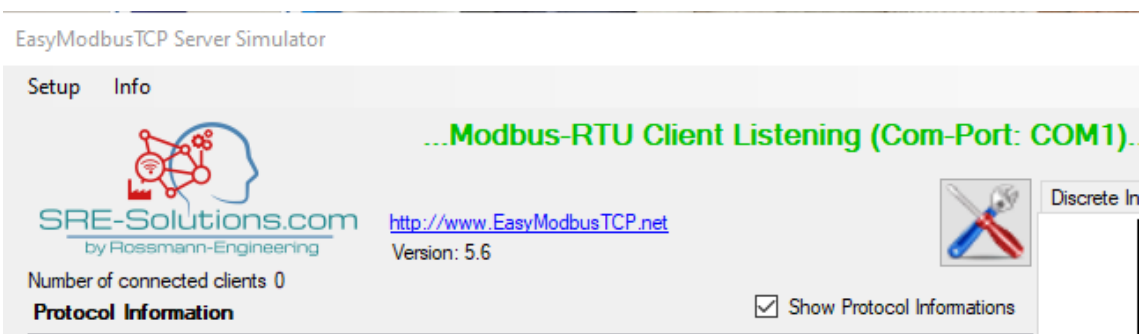


Figure 45: Affectation de port COM1 → EasyModbus

3. Ajout et configuration du protocole Modbus RTU:

Un protocole Modbus RTU est ajouté. Le port COM2 est sélectionné, avec paramétrage des options série: baudrate, parité, bits de données et bits de stop.

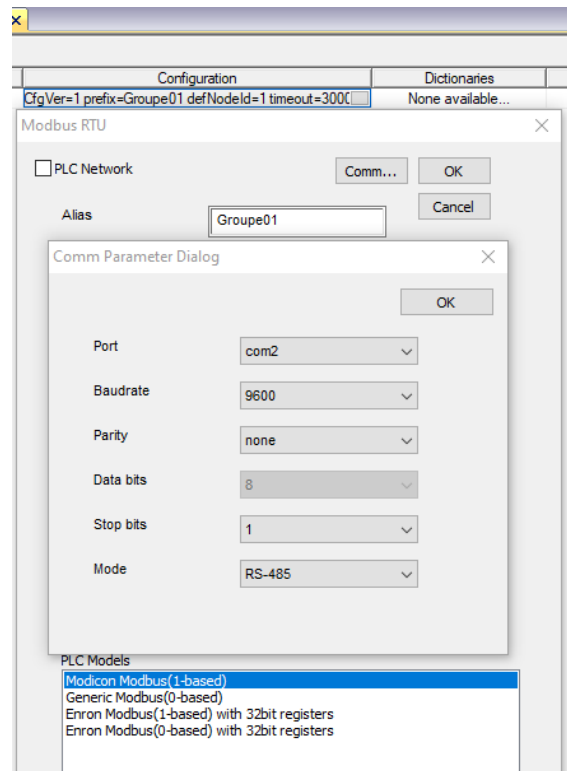


Figure 46: Affectation de port COM2 → AGI Creator

2.3 Création des Tags de Communication

Après avoir mis en place le protocole de communication Modbus RTU dans AGI Creator, l'étape suivante consiste à créer les tags de communication correspondant aux différents paramètres à superviser sur le groupe électrogène.

Dans le cadre de ce projet, les adresses de ces tags ont été directement extraites de la table Modbus fournie par le constructeur de la carte de commande SGC420[27]. Ces adresses définissent l'ensemble des registres disponibles pour l'échange de données avec le système SCADA.

Afin d'assurer une compatibilité avec AGI Creator, qui applique une indexation 1-based (propre au protocole Modbus), chaque adresse a été ajustée en conséquence. Par exemple, une adresse 0 mentionnée dans la documentation technique de la carte SGC420 correspond à 1 dans AGI Creator.

Holding register (03;16)							
Function group	Modbus address	AGI address	Modbus function code	Parameter	Data type	Scaling	Unit
DG mode change	0	1	16	DG mode change command	Unsigned	-	SGC STOP KEY (0x01), Decimal value:01 SGC START KEY (0x02), Decimal value:02 SGC AUTO KEY (0x04), Decimal value:04 SGC ACK KEY (0x40), Decimal value:64 Only one value at a time

Figure 47: Correspondance d'adressage Modbus entre la carte SGC420 et AGI Creator.

En complément, une capture d'écran récapitulative des tags créés dans AGI Creator est présentée ci-après. Elle met en évidence l'organisation fonctionnelle des variables (mesures, alarmes, commandes, etc.) ainsi que leur correspondance précise avec les adresses Modbus définies.

Data	Type	Tag name
Modbus RTU:prot2		
Alias: Groupe01	Container	
Model: Modicon Modbus(1-based)		
Battery voltage	unsignedShort	Battery voltage
Battery voltage & Battery Vtg from ECU	unsignedShort	Battery voltage & Battery Vtg from ECU
Charge alternator voltage	unsignedShort	Charge alternator voltage
Coolant temperature	unsignedShort	Coolant temperature
Coolant temperature & CLNT Temp from ECU	unsignedShort	Coolant temperature & CLNT Temp from ECU
DG status	unsignedShort	DG status
Eng run hrs & Run hrs from ECU (more than 65535 from ECU)	unsignedShort	Eng run hrs & Run hrs from ECU (more than 65535 from ECU)
Engine speed & Speed from ECU	unsignedShort	Engine speed & Speed from ECU
ETAT reseau	unsignedShort	ETAT reseau
Fail To Start Alarm Status	unsignedShort	Fail To Start Alarm Status
Fail To Stop Alarm Status	unsignedShort	Fail To Stop Alarm Status
Fuel level	unsignedShort	Fuel level
Fuel Level in lit	unsignedShort	Fuel Level in lit
Generator average power factor	unsignedShort	Generator average power factor
Generator B frequency	unsignedShort	Generator B frequency
Generator cumulative apparent energy	unsignedInt	Generator cumulative apparent energy
Generator cumulative energy	unsignedInt	Generator cumulative energy
Generator cumulative reactive energy	unsignedInt	Generator cumulative reactive energy
Generator L3-L1 voltage	unsignedShort	Generator L3-L1 voltage
Generator L3-n voltage	unsignedShort	Generator L3-n voltage
Generator L2-L3 voltage	unsignedShort	Generator L2-L3 voltage
Generator L2-n voltage	unsignedShort	Generator L2-n voltage
Generator L1-L2 voltage	unsignedShort	Generator L1-L2 voltage
Generator L1-n voltage	unsignedShort	Generator L1-n voltage
Generator power factor L1	unsignedShort	Generator power factor L1
Generator power factor L2	unsignedShort	Generator power factor L2
Generator power factor L3	unsignedShort	Generator power factor L3
Generator R frequency	unsignedShort	Generator R frequency
Generator Y frequency	unsignedShort	Generator Y frequency
Load L1 current	unsignedShort	Load L1 current

Figure 48: Liste des tags configurés dans AGI Creator

2.4 Conception de l'Interface HMI

Une fois les tags de communication définis dans AGI Creator, l'étape suivante consiste à concevoir l'interface HMI (Human Machine Interface). Cette dernière vise à offrir à l'utilisateur une visualisation claire et interactive de l'état du groupe électrogène, tout en assurant la commande à distance des équipements connectés via Modbus RTU.

2.4.1 Structure de l'interface

L'interface est organisée en plusieurs pages thématiques accessibles par navigation via des boutons. Chaque page regroupe des données spécifiques (mécaniques, électriques, alarmes, etc.), facilitant l'analyse et la prise de décision.

Une page d'accueil a été conçue comme introduction visuelle du système. Elle joue le rôle d'une interface soignée et professionnelle, affichant les informations académiques liées au projet, tout en donnant accès aux pages fonctionnelles suivantes:



Figure 49: Interface d'accueil esthétique du projet

1. **Page des détails électriques:** Cette page présente les grandeurs électriques principales ainsi que les commandes essentielles du groupe électrogène. Un synoptique dynamique illustre le couplage réseau-groupe. Elle intègre les boutons de commande manuelle : démarrage, arrêt, mode automatique (AUTO), acquittement des alarmes (ACK), ainsi que des boutons de navigation.

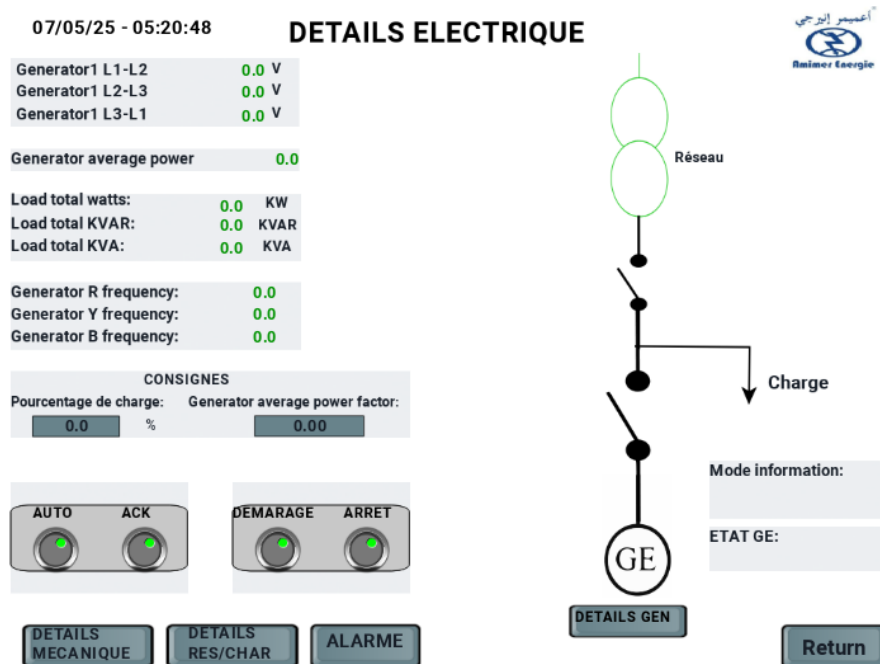


Figure 50: Page des détails électriques

2. **Page des détails mécaniques:** Elle regroupe les paramètres moteurs avec insertion de jauges graphiques analogiques pour une meilleure lisibilité.

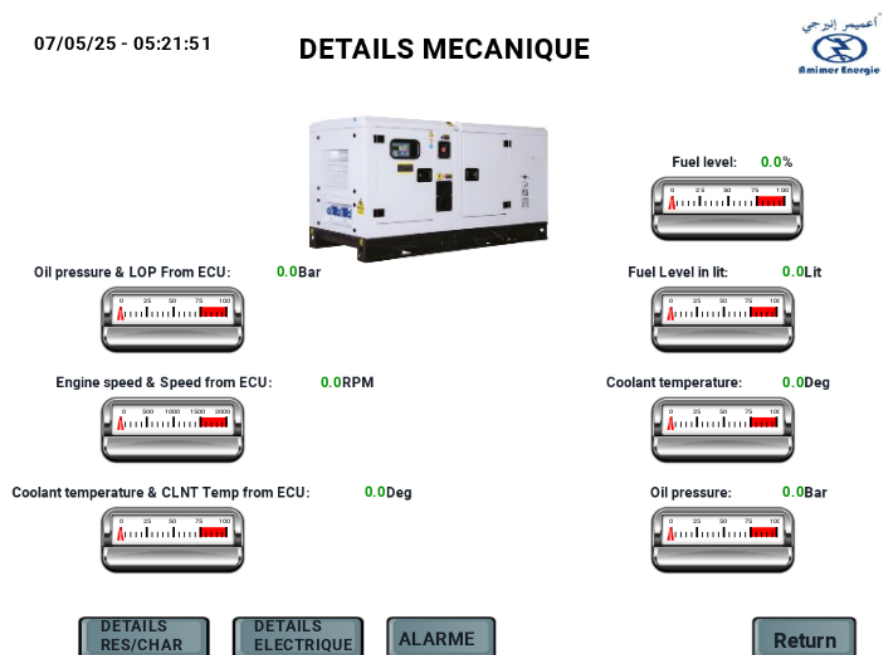


Figure 51: Page des détails mécaniques

3. **Page des détails génératrice:** Cette interface affiche les valeurs mesurées côté génératrice (avant couplage), les énergies cumulées, la tension batterie et la charge alternateur.

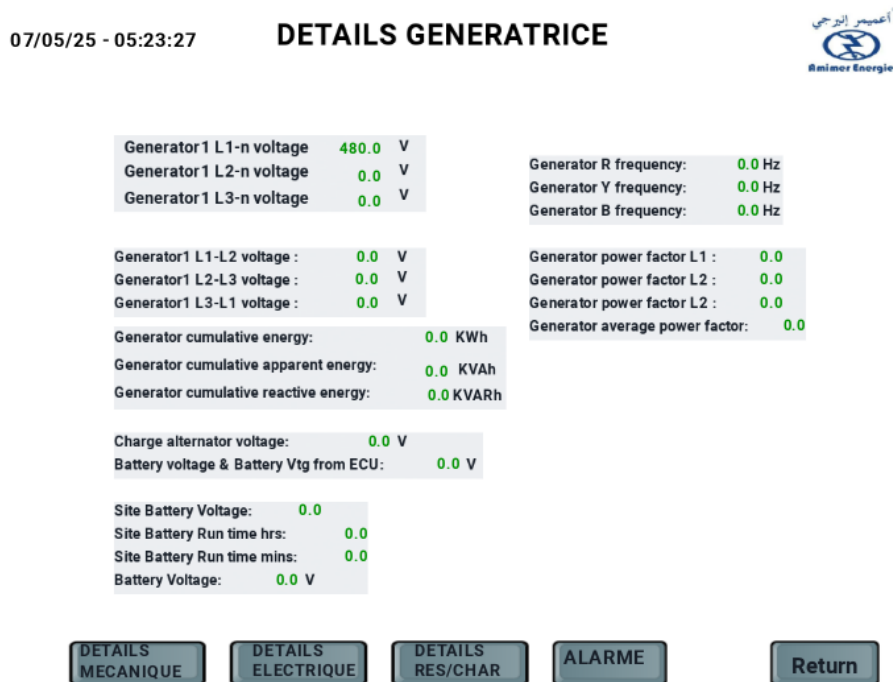


Figure 52: Page des détails de la génératrice

4. **Page détails réseau / charge:** Elle regroupe les mesures côté réseau, permettant de visualiser l'état du couplage réseau-groupe et d'évaluer la stabilité des charges..

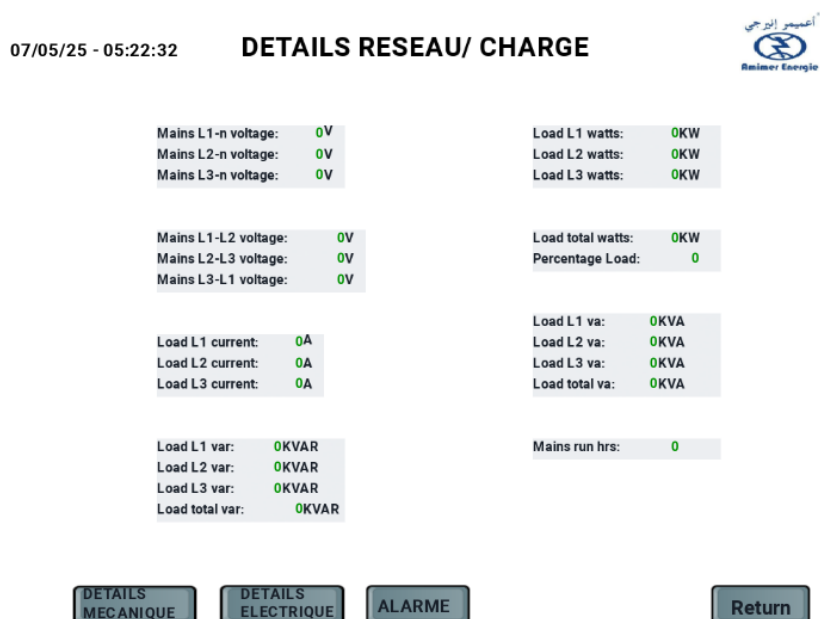


Figure 53: Page des détails mécaniques

5. **Page des alarmes:** Un tableau dynamique récapitule les alarmes actives, leurs horodatages, codes et descriptions. L'utilisateur peut effectuer des filtres temporels ou acquitter les défauts via un bouton spécifique.

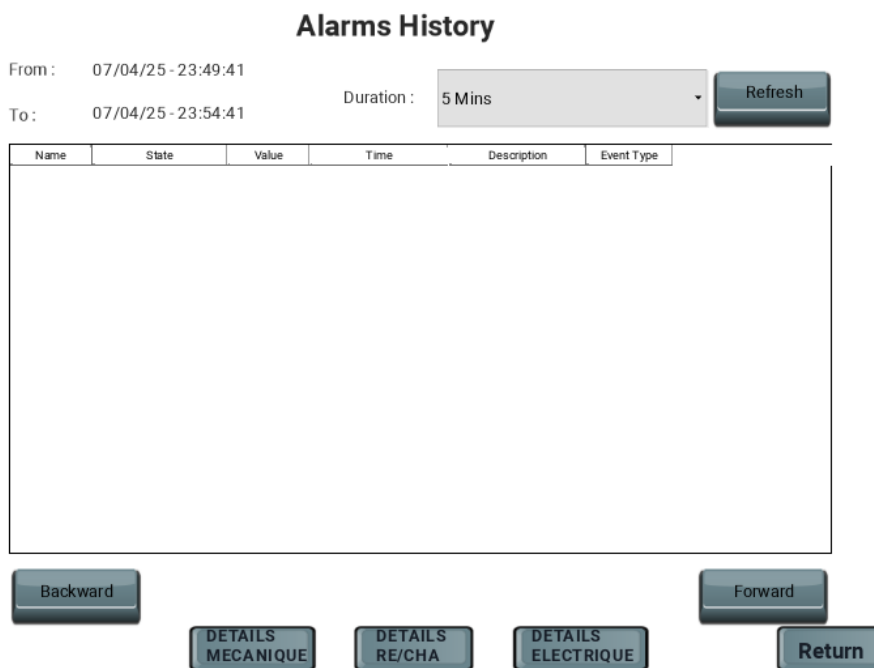


Figure 54: Liste des alarmes

2.4.2 Liaison des Objets Graphiques aux Données

Tous les objets graphiques affichés dans les pages précédentes ont été associés à des tags créés dans AGI Creator. Chaque widget (jauge, afficheur, voyant, etc.) est lié à une adresse Modbus extraite de la documentation de la carte SGC420.

La configuration a été réalisée via le panneau Properties, en spécifiant pour chaque propriété (Value, Display...) le tag correspondant, ainsi que le type d'accès (lecture, écriture ou les deux). Cela permet d'assurer une mise à jour dynamique des données pendant la simulation ou en exploitation réelle.

2.5 Exécution de la Simulation

Après avoir conçu l'interface HMI, défini les tags de communication et établi la liaison série virtuelle, la phase finale consiste à exécuter la simulation . Cette dernière est initiée via le menu **Run → Start Simulation** ce qui active les échanges Modbus entre le maître (AGI Creator) et l'esclave (EasyModbus).

2.5.1 Observation des Valeurs Simulées sur EasyModbus et AGI

Les données simulées sont observées à la fois dans la fenêtre EasyModbus et sur les widgets de l'interface AGI Creator.

Des captures d'écran sont présentées ci-après à différents moments de la simulation, illustrant l'état des variables, les actions de commande et les réponses de l'esclave.

Démarrage:

Lorsque l'utilisateur clique sur le bouton **Démarrage** de l'interface SCADA, la commande est transmise à l'esclave qui met à jour la valeur du registre associé, déclenchant visuellement l'état **groupe en marche**.

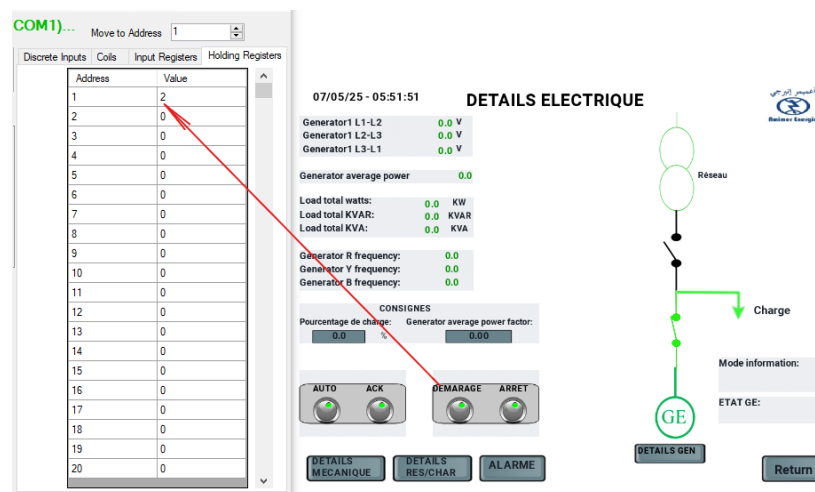


Figure 55: Activation du groupe par bouton Démarrage

Mesures électriques:

Lorsque les tensions (L1-N, L2-N, L3-N) sont mises à jour dans EasyModbus, l'interface AGI Creator reflète immédiatement les valeurs, confirmant une lecture cohérente.

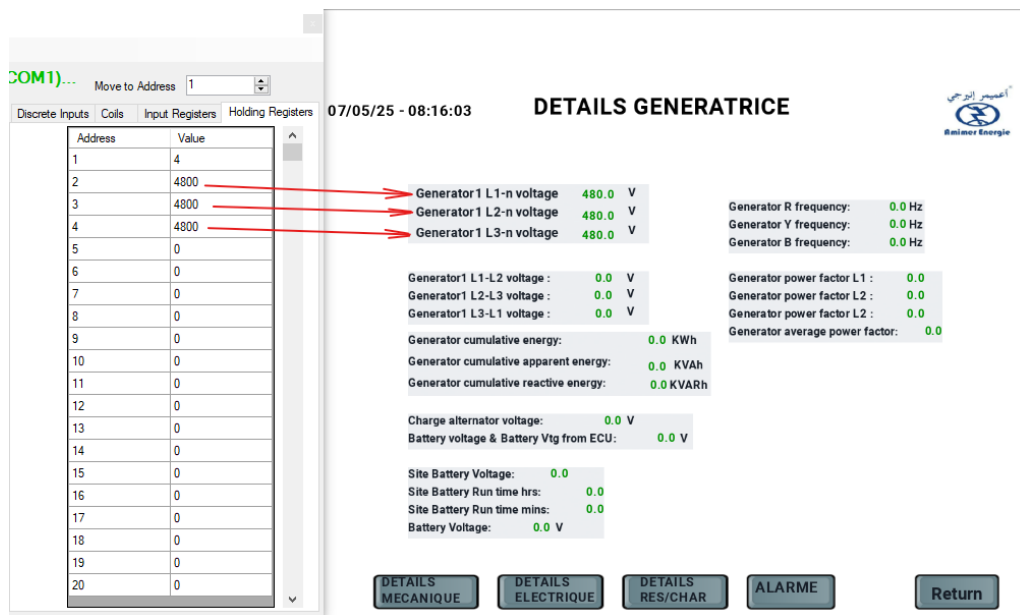


Figure 56: Visualisation des tensions simples du générateur

Mesures mécaniques:

Les données telles que la pression d'huile moteur sont introduites manuellement pour simuler des capteurs analogiques. Les jauges associées dans AGI Creator affichent alors les valeurs simulées en temps réel.

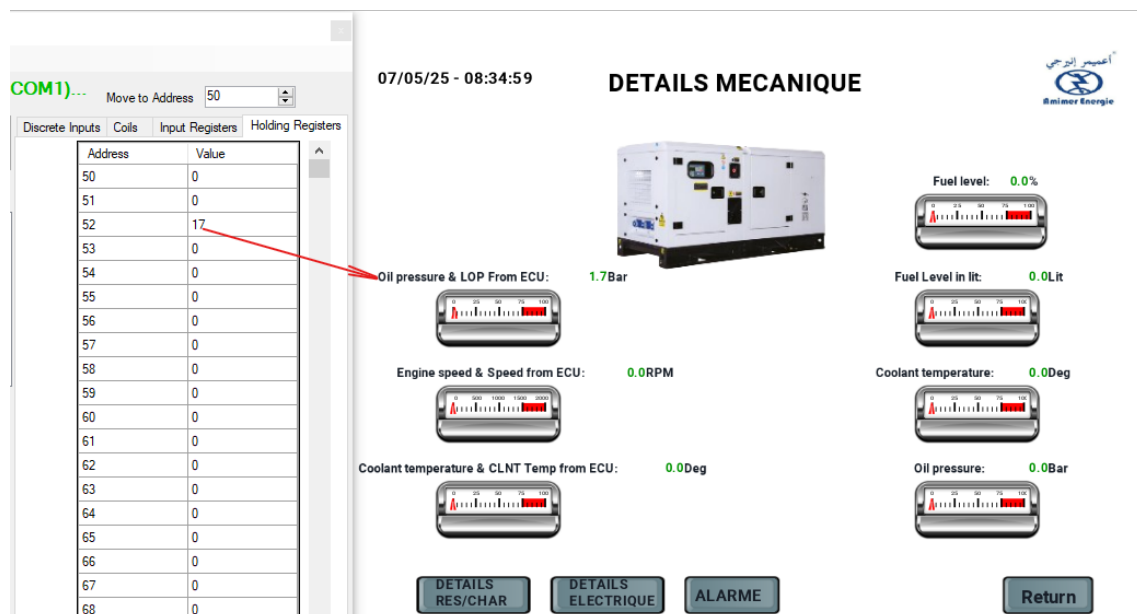


Figure 57: Affichage de la pression d'huile moteur simulée

2.5.2 Vérification de la Commande d'Écriture

Dans cette phase de la simulation, l'objectif est de valider que les commandes manuelles envoyées depuis l'interface HMI développée sous AGI Creator sont correctement interprétées et exécutées par l'esclave Modbus simulé EasyModbus Server, conformément aux spécifications de la carte de commande SGC420.

Les boutons de commande sont chacun associés à une action de type **SetBit** dans les propriétés des widgets d'AGI Creator. Cette action permet d'écrire une valeur spécifique dans un mot de 16 bits en activant un bit donné du registre Modbus concerné.

Table 13: Correspondance des commandes AGI / Interprétation par le SGC420

Commande	Bit activé (AGI)	Valeur binaire	Valeur décimale (SGC)
Arrêt	bit 0	0000 0000 0000 0001	1
Démarrage	bit 1	0000 0000 0000 0010	2
Mode AUTO	bit 2	0000 0000 0000 0100	4
Acquittement	bit 6	0000 0000 0100 0000	64

Exemple de scénario: lorsque l'utilisateur clique sur **Arrêt**, AGI Creator génère une trame Modbus RTU de type fonction 16, qui écrit la valeur décimale **1** à l'adresse 0.

L'esclave EasyModbus interprète cette requête et met à jour le registre **1** (vue côté SGC) avec la valeur **1**, simulant ainsi l'envoi d'un ordre d'arrêt au groupe électrogène.

Ce principe est identique pour toutes les autres commandes, l'action **SetBit(n)** = 1 dans AGI Creator active le *n-ième* bit du mot Modbus, ce qui correspond à une valeur entière bien définie, en accord avec la logique de commande de la carte SGC420.

3 Mise en œuvre du Système SCADA sur un Groupe Électrogène Réel

3.1 La Carte SGC420

La SGC420 est un contrôleur numérique multifonction conçu pour la gestion complète d'un groupe électrogène autonome. L'appareil assure les fonctions de commande, de surveillance, de protection et de signalisation du groupe. Elle est équipée d'un afficheur LCD et de touches frontales permettant la navigation dans les menus et le contrôle du système.

- Elle permet de piloter le groupe électrogène, son disjoncteur et, si nécessaire, un disjoncteur réseau.
- Elle permet également la surveillance de la batterie site et de la température d'abri, avec un impact positif sur la consommation de carburant.

3.1.1 Afficheur et Boutons de Commande

Le contrôleur SGC420 est équipé d'un écran LCD graphique rétroéclairé, associé à un ensemble de boutons physiques en façade qui constituent l'interface utilisateur locale. Cette interface permet à l'opérateur de surveiller les paramètres du groupe électrogène et d'effectuer les principales actions de commande sans outil externe.

Les fonctions des éléments de la façade sont les suivantes:

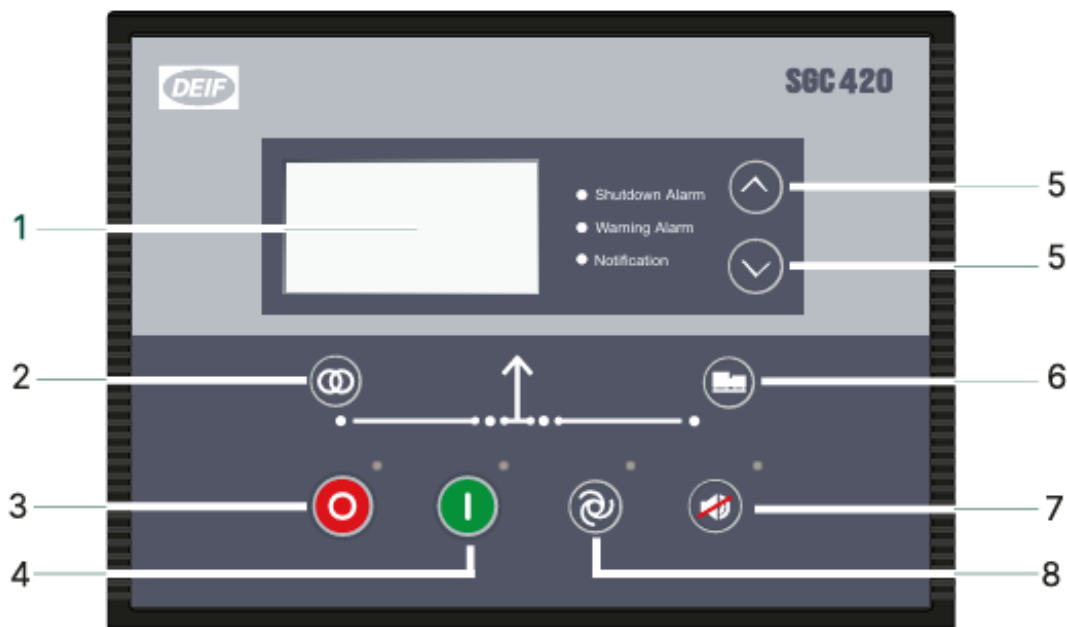


Figure 58: Carte SGC 420

1. **Écran graphique:** affiche les mesures en temps réel, les alarmes, les états et les menus de configuration.
2. **Bouton disjoncteur réseau:** permet d'ouvrir ou de fermer manuellement le disjoncteur réseau.
3. **Bouton Stop:** arrête le groupe en mode manuel. -Un appui simple déclenche la phase de refroidissement.
-Un double appui arrête immédiatement le moteur.
-Long appui Accès au menu de configuration.
-Un appui prolongé permet d'entrer ou de quitter le mode configuration.
4. **Bouton Start:** démarre le groupe électrogène en mode manuel.

-
5. **Boutons de navigation:** permettent de faire défiler les menus vers le haut ou le bas.
 6. **Bouton disjoncteur groupe:** permet d'ouvrir ou de fermer le disjoncteur du groupe.
 7. **Bouton Acknowledge:** utilisé pour acquitter une ou plusieurs alarmes actives.
 8. **Sélecteur de mode:** permet de changer le mode de fonctionnement du groupe.

3.1.2 Menus de Surveillance et de Configuration

Menus de surveillance:

Il permet à l'opérateur de consulter en temps réel les principales grandeurs mesurées par la carte. Ces informations sont affichées sur l'écran LCD soit automatiquement chaque avec un délai de mise à jour configurable, soit manuellement en appuyant sur les boutons Up et Down.

Une autre option est disponible dans ce menu est Histogramme de Charge, qui permet de visualiser les heures de fonctionnement du moteur en fonction des différents niveaux de charge. Les heures sont réparties en groupes de charge spécifiques, ce qui permet de suivre combien de temps le groupe électrogène a fonctionné à un pourcentage de charge donné.

Menus de configuration:

Il donne à l'opérateur ou à l'installateur l'autorisation de paramétrer les différents réglages du contrôleur SGC420 uniquement après une authentification via un mot de passe.

3.1.3 Entrées / Sorties

La carte SGC420 permet de se connecter à divers dispositifs externes via ses entrées et sorties, facilitant l'échange de données avec les composants du groupe électrogène:

1. **Entrées analogiques:** pour les capteurs mesurant des valeurs comme la température, la pression, le niveau de carburant, etc.
2. **Entrées numériques:** pour des informations telles que l'état du disjoncteur, la commande d'arrêt d'urgence, ou des signaux de niveau logique (haut/bas) provenant d'autres composants.
3. **Sorties relais:** pour la commande du démarrage du moteur, l'ouverture ou la fermeture du disjoncteur du groupe électrogène, et l'envoi de signaux d'alarme ou d'état.
4. **Sorties analogiques:** pour l'envoi de signaux de sortie à d'autres systèmes ou équipements connectés, tels que des systèmes de monitoring à distance.

3.1.4 Modes de Fonctionnement

Le contrôleur SGC420 offre plusieurs modes de fonctionnement pour s'adapter à différentes situations et optimiser la gestion du groupe électrogène.

Le changement de mode se fait directement depuis l'interface utilisateur de la carte via l'écran LCD et les boutons de commande. Des voyants lumineux indiquent le mode en cours de fonctionnement:

- **Mode Arrêt (OFF):** Tous les systèmes sont mis hors tension et le groupe est en arrêt total. Il est utilisé en cas de maintenance ou si la charge est alimentée globalement par le réseau.
- **Mode Manuel (MAN):** L'opérateur prend le contrôle total du groupe électrogène, notamment lors de tests ou de dépannage.
- **Mode Automatique (AUTO):** Le groupe fonctionne en autonomie, c'est-à-dire, il démarre et s'arrête selon les conditions externes.
- **Mode Test (TEST):** Il est utilisé pour tester le groupe électrogène sans couplage réel à la charge.

3.1.5 Communication

Il est équipé d'interface de communication permettant l'intégration du groupe électrogène dans un système de supervision externe, comme un système SCADA.

- **Communication RS-485 pour Modbus RTU:** Le contrôleur SGC420 utilise une interface RS-485 pour la communication Modbus RTU, permettant l'intégration avec un système SCADA pour une supervision et un contrôle à distance.
- **Communication par Bus CAN pour le moteur:** La carte prend en charge une interface CAN Bus (optionnelle) pour échanger des données en temps réel sur les paramètres moteurs du groupe électrogène, assurant une gestion optimisée.
- **Interface USB pour la communication PC:** Une interface USB permet de connecter le contrôleur à un PC pour la configuration locale, la mise à jour des paramètres et la maintenance du système.

3.2 Mise en œuvre de la Communication réelle

Dans cette configuration réelle :

- Le logiciel **AGI Creator**, installé sur PC, joue le rôle de *maître Modbus*.
- La **SGC 420**, fixée sur le groupe électrogène, agit comme *esclave*.
- La communication **hardware** est assurée via un câble **USB/RS485**.
- La communication **logicielle** repose sur le protocole **Modbus RTU**.

3.2.1 Câblage Physique

Une fois la carte SGC installée, un câble **USB/RS485** est utilisé pour établir la liaison série:

- Une extrémité est connectée au port de communication de la carte SGC.
- L'autre est branchée au port **USB** du PC.



Figure 59: Câblage de la carte SGC 420 avec le PC via un convertisseur USB/RS485

3.2.2 Configuration du Port Série dans AGI Creator

Une fois le câble connecté, le port COM correspondant apparaît dans le *gestionnaire de périphériques*. Il est ensuite configuré dans AGI Creator (menu *Protocole*), ainsi que les paramètres Modbus RtU correspondantes.

3.3 Tests Fonctionnels et Validation

3.3.1 Commandes via l'Interface HMI

Le premier test consiste à déclencher le démarrage du groupe électrogène depuis l'interface, en cliquant sur le bouton **START**. Cela fait démarrer le groupe correctement. Meme teste eétit effectuée avec les boutons **STOP** et une réponse du groupe est établie.

3.3.2 Comparaison des Valeurs Affichées

Une comparaison de données affichées dans l'interface **HMI AGI Creator**, et sur l'**afficheur local de la carte SGC 4120** est établie.

Une correspondance été observée sur toutes les pages: mesures électriques, mécaniques, génératrice, etc.

Cette validation confirme la réussite de la communication Modbus RTU, ainsi que la cohérence de l'adressage.

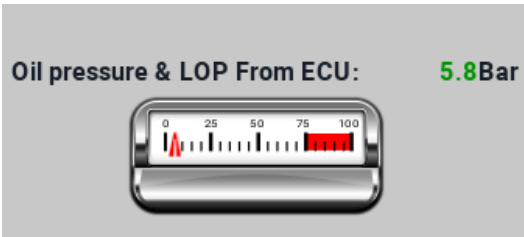
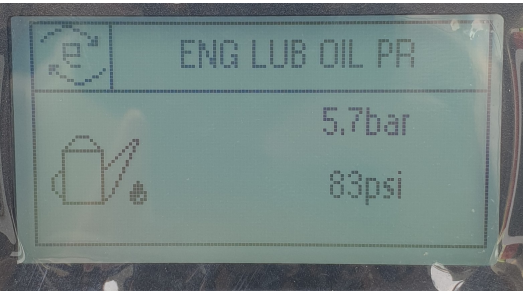


Figure 60: Validation de la mesure de pression d'huile entre AGI et SGC.

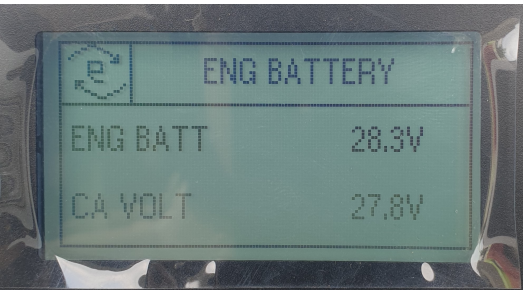


Figure 61: Validation de la mesure de tension batterie entre AGI et SGC.

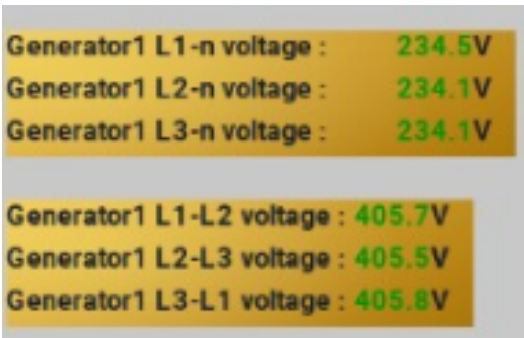
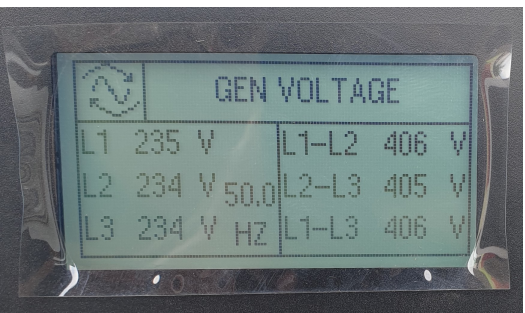


Figure 62: Validation des tensions génératrice entre AGI et SGC.

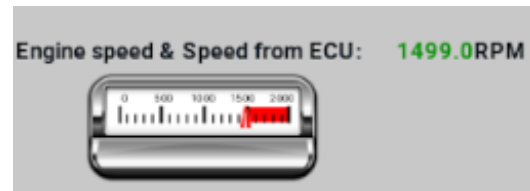
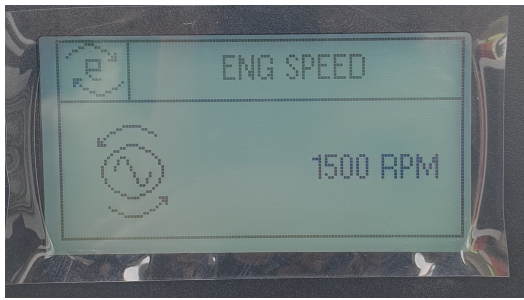


Figure 63: Synchronisation de la vitesse moteur affichée entre AGI et SGC.

Conclusion Générale

Ce projet de fin d'études a permis d'aborder et de résoudre plusieurs défis liés à l'optimisation de la régulation de vitesse d'un groupe électrogène à travers l'utilisation de méthodes avancées telles que **l'Optimisation par Essaims Particuliers (PSO)** et **les Algorithmes Génétiques (GA)**. L'optimisation des paramètres du régulateur PID, selon le **critère d'erreur minimal ITAE**, a montré une amélioration significative de la stabilité et de la réactivité du système, assurant une performance stable même lors des variations de charge.

En parallèle, un système **SCADA** a été développé avec une interface homme-machine (IHM) intuitive, utilisant le protocole **Modbus RTU** pour permettre une supervision et un contrôle à distance du groupe électrogène. Ce système a démontré sa capacité à surveiller les paramètres électromécaniques en temps réel et à anticiper les éventuelles pannes, renforçant ainsi la fiabilité de l'installation.

Les résultats expérimentaux ont prouvé l'efficacité des deux approches proposées et ont montré leur potentiel d'application, tant pour les systèmes connectés au réseau que pour les installations autonomes dans des zones isolées. Ce travail ouvre la voie à des améliorations continues dans le domaine de la régulation de groupes électrogènes et de leur supervision.

Enfin, bien que les objectifs du projet aient été atteints, des perspectives d'amélioration existent :

Aspects de régulation:

- Optimisation de la régulation par l'ajustement des paramètres PID en temps réel selon les variations du système.
- Amélioration de la régulation par l'application de l'apprentissage automatique pour une détection plus précise des patterns et une meilleure adaptation à long terme.
- Gestion simultanée de plusieurs paramètres pour une régulation plus efficace dans des systèmes industriels complexes.

Aspects SCADA et communication:

- Renforcement du SCADA par l'intégration de fonctionnalités prédictives et d'alertes automatisées pour une supervision proactive.
- Amélioration de la compatibilité SCADA avec d'autres équipements grâce à l'utilisation de protocoles de communication avancés comme Modbus TCP ou OPC-UA.
- Développement d'une solution de supervision à distance via une application mobile ou cloud pour une gestion flexible et réactive des systèmes.

Bibliographie

Bibliographie

- [1] **Site Amimer Energie**, "Qui sommes-nous?"
- [2] **La Norme ISO 8528-1-2018**, "Description générale."
- [3] **Site Genesal Energy**, "Classification et types de groupes électrogènes."
- [4] **Cahier technique n° 196**, "Génération électrique intégrée aux sites industriels et bâtiments commerciaux."
- [5] **Adjaratou LINGANI**: "Etude de la synchronisation automatique de deux groupes électrogènes de secours cummins 400 kVA.", mémoire Master d'ingénierie Génie Electrique et Energétique Titre d'I.I.E.E, de Ouagadougou, Janvier 2024.
- [6] **Messaoud DJELLOUT, Brahim DJAROUM** : "Conception et réalisation d'un système de démarrage automatique d'un groupe électrogène à base d'une carte Arduino.", mémoire de fin d'études de Master académique, Sciences et Technologies, Génie Electrique, Electrotechnique Industriel, Université A. MIRA, Béjaia, Juillet 2018 .
- [7] **BEDJOU Fayçal, CHIKOU Kaci**: "Etude d'un groupe électrogène à base d'une génératrice asynchrone.", mémoire PFE, Université A. MIRA, Béjaia, 2021.
- [8] **Site Your power guide**, "AVR for Generator."
- [9] **Site DINGO Power**, "Classification of Governor for Diesel Generator Set."
- [10] **Manuel EMSA Generator**, "Manuel de maintenance et d'utilisation des groupes électrogène a moteur diesel."
- [11]**Abdelhafid HELLAL**: "Commande et Stabilité et des Réseaux Electriques", CHAPITRE 04: COMMANDE AUTOMATIQUE DE PRODUCTION (AGC)"
- [12] **Bikash Das, V. Mukherjee, D. Das**: "Dynamic Modeling and Simulation of Diesel Generator for Stand-alone and Grid Connected Mode of Operation." , Article, Janvier 2019
- [13]**Gouasmia Khadidja, Djendli Bariza**: "Comparaison des contrôleurs intelli-

gents d'un AVR.", mémoire Master, UNIVERSITÉ BADJI MOKHTAR, ANNABA, 2023/2024.

[14] **KENNAT Mohammed, BRAHMIA Oussama:** "Modélisation et commande d'un système hydraulique à trois réservoirs.", mémoire MASTER Académique, Université 8 Mai 1945, Guelma, 19/06 /2022.

[15] **Ghiloubi Imam Barket:** "Poursuite du point de puissance maximale MPPT d'un système photovoltaïque sous les conditions PSC basée sur l'algorithme PSO", mémoire DE MASTER, Université Mohamed Khider, Biskra, jeudi 24 septembre 2020.

[16] **Abbas El Dor:** Perfectionnement des algorithmes d'optimisation par essaim particulaire: applications en segmentation d'images et en électronique., Université Paris-Est, 2012. Français.

[17] **Abdelhafid HELLAL:** "Partie 3: Métaheuristiques Techniques Evolutionnaires, Cas des Algorithmes Génétiques."

[18] **MEHIDID Fadila** "Algorithme GÈnÈtique.",mémoire Master,Université ABDELHAMID IBN BADIS, MOSTAGANEM, le 19/06/2013.

[19] Chapitre III: Les Algorithmes Génétiques Multiobjectifs

[20] **ARIGUE Imad,MADJOURI Nourdine** "Automatisation et optimisation d'un micro réseau par automate programmable base sur système SCADA « Smart Grid »", mémoire Master Académique, Université Echahid Hamma Lakhdar, El Oued, 2018/2019.

[21] **IKEDICHE Sara:** Etude et planification d'un système de supervision (SCADA) sous le logiciel Labview ,Mémoire du Projet de Fin d'Etude, Université Djilali Bounaama Khemis Miliana, 2017/2018

[22] **BELHASSANI Yasmine, SOLTANI Ourida:** "Développement d'une passerelle Modbus RTU/ Modbus TCP pour communication SCADA sécurisée", mémoire Ingénieur état, 2022/2023.

[23] "GUIDE DU MODBUS POUR LES NULS."

[24] **site Yutec:** "Modbus RTU: Fonctionnement et mise en oeuvre."

[25] Site **ip-systeme**: "Qu'est-ce que Modbus RTU ?"

[26] Site **DEIF TDocumentation for AGI 300**

[27] Carte **SGC 420**, "Table Modbus sous format excel."

ANNEXE

Annexes

Annexe 01

Modbus Poll

C'est un logiciel de simulation conçu pour tester, surveiller et diagnostiquer les communications Modbus dans un environnement maître-esclave. Il agit comme un maître Modbus capable d'envoyer des requêtes vers un ou plusieurs esclaves, et d'afficher les réponses en temps réel, sous forme de tableaux ou d'hexadécimal.

Développé pour les systèmes utilisant Modbus RTU ou Modbus TCP, il est particulièrement apprécié pour sa simplicité d'utilisation et sa flexibilité.

Principales fonctionnalités:

- Il prend en charge différentes connexions, telles que Modbus RTU/ASCII via RS232 ou RS485, Modbus TCP/IP, ainsi que les versions encapsulées sur TCP.
- Il supporte l'ensemble des fonctions Modbus classiques (lecture/écriture de bobines et de registres), ainsi que divers types de données déjà cités dans la partie RTU.
- Il permet également l'enregistrement des données dans des fichiers texte ou directement dans Excel, et leur affichage selon plusieurs formats (binaire, hexadécimal, entier signé ou non signé, flottant, etc.).
- Enfin, le logiciel intègre des fonctionnalités supplémentaires comme la surveillance du trafic de données, l'interface avec Excel via OLE Automation, l'impression, l'aperçu avant impression, et la personnalisation des polices et couleurs.

Interface utilisateur:

Modbus Poll utilise une **interface à documents multiples (MDI)** permettant d'ouvrir plusieurs fenêtres simultanément. Chaque fenêtre peut afficher les données issues de différents esclaves Modbus en parallèle.

Trois méthodes permettent d'ouvrir une nouvelle fenêtre dans Modbus Pol: Ctrl + N, File → New ou Icône *New* dans la barre d'outils.

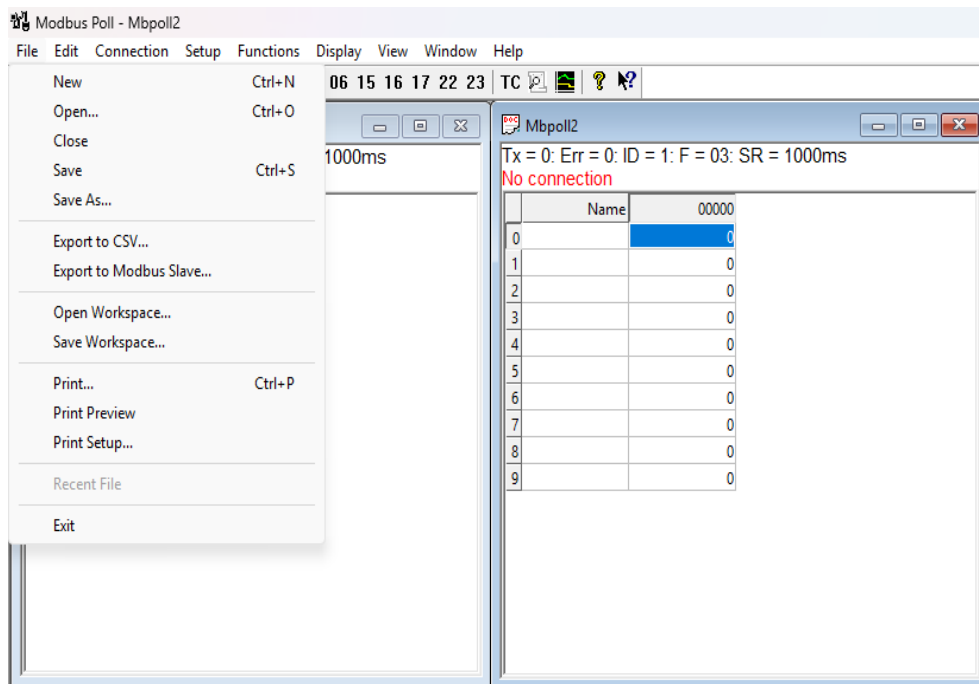


Figure 64: Interface du logiciel Modbus Poll

Configuration de la communication

Avant de pouvoir échanger des données avec un esclave Modbus, l'utilisateur doit configurer correctement les paramètres de communication dans Modbus Poll.

Le logiciel met à disposition une boîte de dialogue dédiée, accessible de deux manières: en appuyant sur la touche F3 du clavier, ou en sélectionnant le menu **Connection** → **Connecte** dans la barre supérieure.

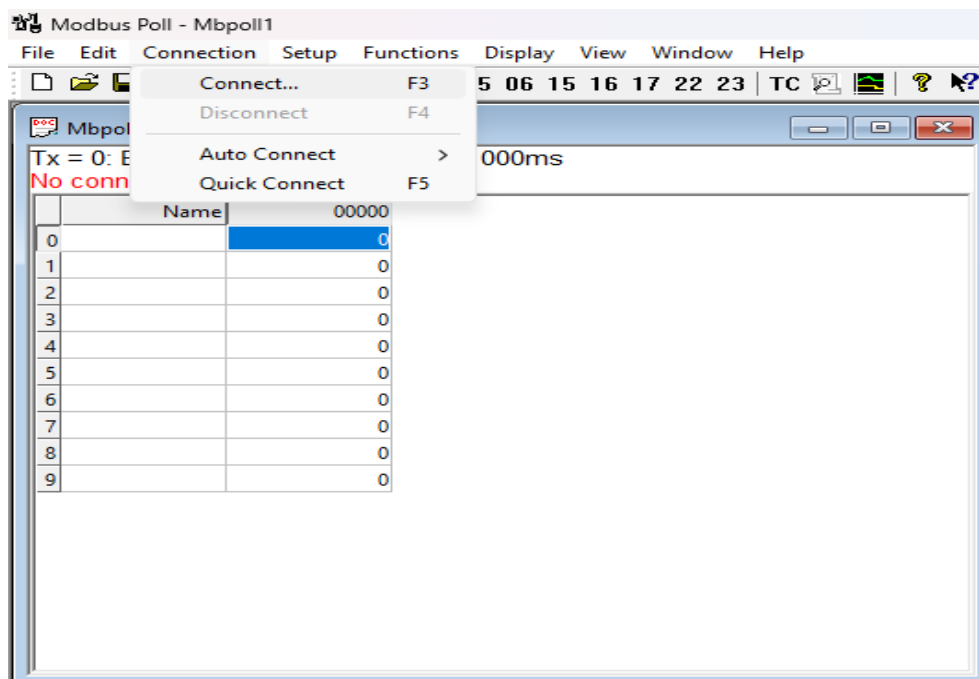


Figure 65: Menu de connexion dans Modbus Poll

Cette interface permet de paramétrer tous les aspects de la communication, que ce soit via un port série (RS232/RS485) ou via un réseau Ethernet (TCP/IP).

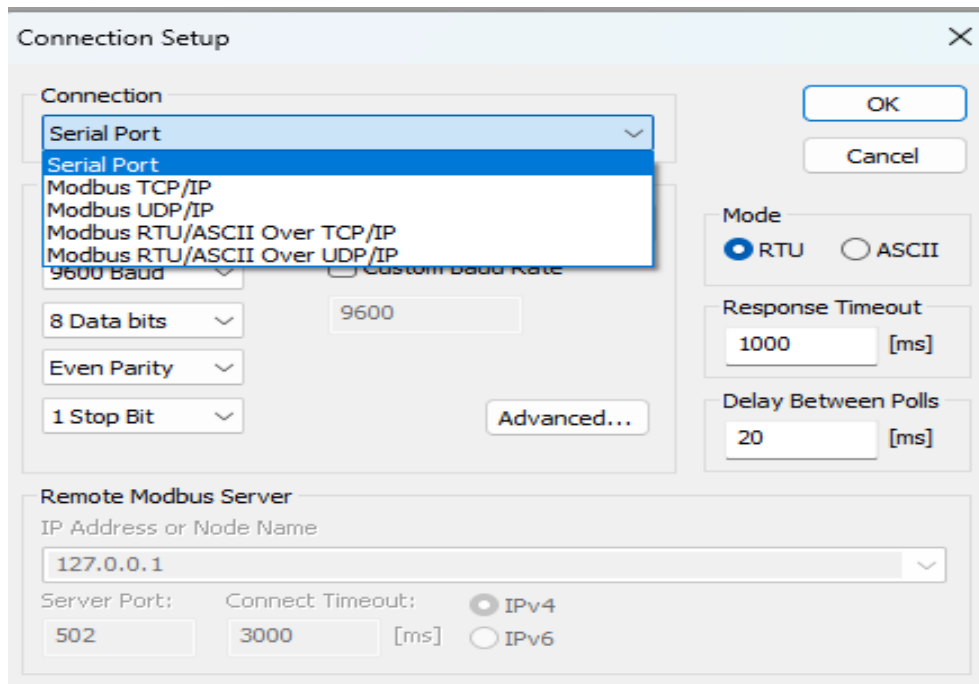


Figure 66: Configuration de la communication dans Modbus Poll

Selon le mode de communication sélectionné, la boîte de dialogue affiche différents champs à renseigner, propres à chaque type de connexion:

1. **Communication série (RS232/RS485):** Lorsque l'utilisateur opte pour une liaison série, plusieurs paramètres doivent être configurés avec précision afin d'assurer une transmission correcte:
 - **Port COM utilisé:** port série utilisé, physique ou virtuel.
 - **Débit en bauds (baudrate):** vitesse de transmission qui doit être identique côté maître et esclave.
 - **Parité:** mécanisme optionnel de détection d'erreur sur un bit. Trois options sont possibles:
 - Aucune (None): pas de contrôle d'erreur par parité.
 - Paire (Even): ajoute un bit de parité pour que le nombre total de bits à 1 soit pair.
 - Impaire (Odd): même principe, mais le nombre de bits à 1 doit être impair.
 - **Bits de données:** la taille des données transmises fixés à 8 bits, standard pour transmettre une donnée.
 - **Bits de stop:** indiquent la fin de transmission d'un caractère avec une valeur typique de 1 ou 2 bits.

Le choix du mode de transmission est également requis :

- **RTU**: format binaire plus rapide (par défaut),
- **ASCII**: format textuel, plus lisible mais plus lent.

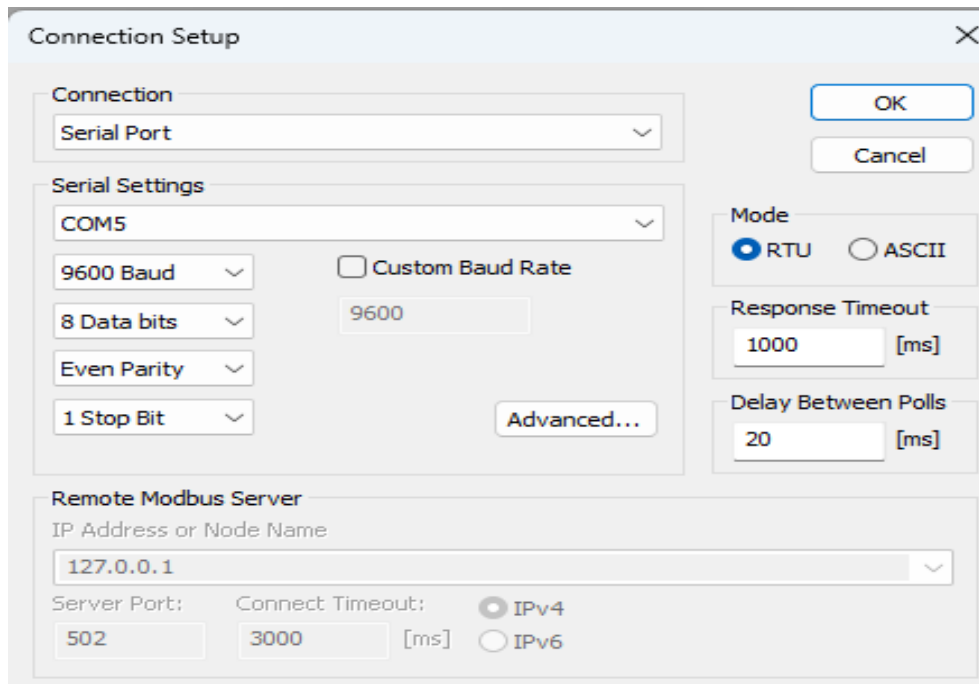


Figure 67: Communication série dans Modbus Poll

En cliquant sur le bouton Advanced, l'utilisateur peut accéder à des options spécifiques, notamment utiles dans les cas suivants :

- **RTS Toggle**: C'est une ligne de signal utilisée dans les communications série RS232/RS485 pour indiquer que l'émetteur est prêt à envoyer des données.
RTS Toggle activé veut dire que cette ligne sera mise à l'état haut (1) pendant que le logiciel envoie des données, et remise à l'état bas (0) dès que l'envoi est terminé.
Une désactivation mal synchronisée du signal RTS peut entraîner soit la perte de données (si désactivé trop tôt), soit une mauvaise interprétation du message par l'esclave (si désactivé trop tard), d'où des erreurs de communication.
- **DSR (Data Set Ready)**: Signal physique émis par l'esclave pour indiquer qu'il est prêt à recevoir des données.
Si ce signal est désactivé, l'envoi est suspendu pour ne pas perdre les données.
- **CTS (Clear To Send)**: C'est un signal physique que le périphérique esclave envoie vers le maître aussi, mais plus direct que DSR.
- **Suppression de l'écho**: Certains convertisseurs ou dispositifs renvoient automatiquement les caractères qu'ils reçoivent (écho).

Cette option permet d'ignorer ces renvois pour éviter qu'ils ne soient interprétés à tort comme des réponses de l'esclave.

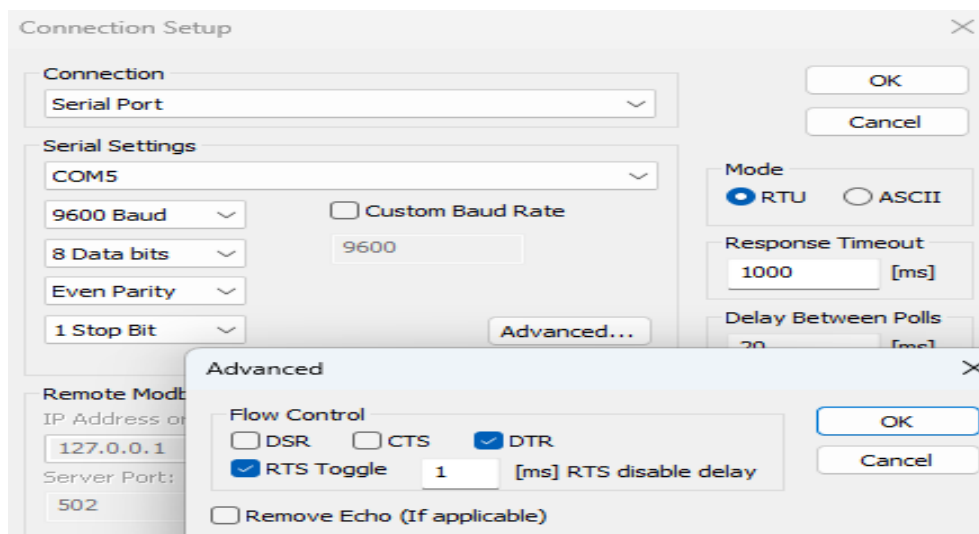


Figure 68: Paramètres avancés de la communication série dans Modbus Poll

2. **Communication Ethernet:** Pour une communication via un réseau, l'utilisateur sélectionne TCP/IP comme type de connexion. Les champs à configurer sont les suivants :

- **Adresse IP de l'esclave:** C'est l'identifiant unique de l'esclave sur le réseau.
- **Port TCP:** Il s'agit du point d'entrée sur l'appareil cible pour la communication Modbus.
- **Timeout de connexion:** Durée maximale (en millisecondes) pendant laquelle le logiciel attend une réponse après avoir tenté de se connecter.

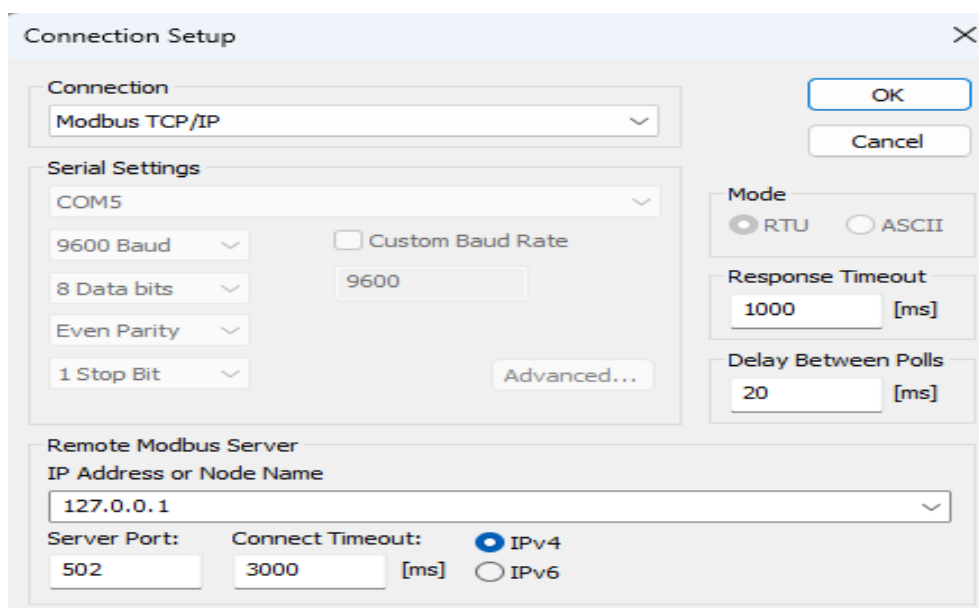


Figure 69: Communication Ethernet dans Modbus Poll

Certains réglages sont disponibles quel que soit le type de communication:

- **Timeout de réponse:**

Il détermine la durée maximale que le logiciel attend avant d'abandonner une requête si aucune réponse n'est reçue de l'esclave. La valeur par défaut est de 1000 millisecondes (ms). Un réglage est essentiel pour éviter les blocages du système en cas de perte de communication.

- **Délai minimum entre requêtes:**

Ce paramètre définit un temps d'attente entre deux requêtes successives pour éviter une surcharge du réseau. Une valeur minimale de 20 ms est conseillée en liaison RS485 multi-esclaves pour garantir un bon fonctionnement. Sur TCP/IP, cette contrainte est moins critique. Modbus Poll permet également de configurer des paramètres avancés:

Définition des échanges Modbus:

Une fois la communication établie, l'utilisateur doit définir les paramètres des échanges Modbus à réaliser dans la fenêtre active.

Cette configuration s'effectue via la boîte de dialogue "Read/Write Definition", accessible par le menu Setup → Read/Write Definition, par la touche F8, ou par l'icône correspondante dans la barre d'outils.

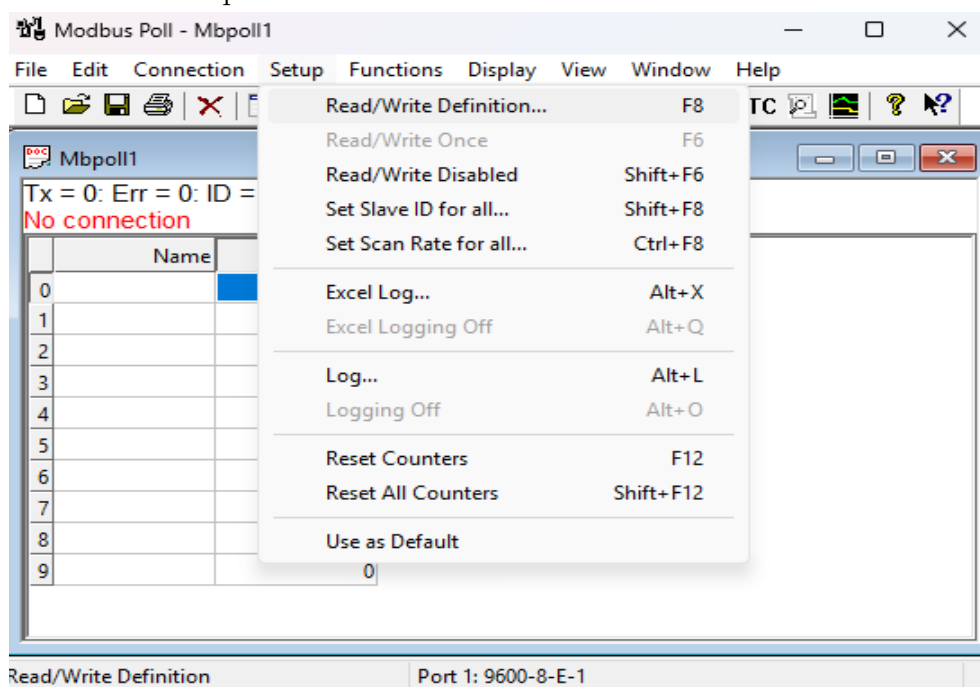


Figure 70: Accès à la fenêtre Read/Write Definition dans Modbus Poll

Après l'accès à cette fenêtre, plusieurs paramètres doivent être configurés:

- **Identifiant de l'esclave (Slave ID):** valeur comprise entre 1 et 255, correspondant à l'adresse de l'équipement cible. La valeur 0 peut être utilisée pour certains dispositifs TCP/UDP.
- **Code de fonction Modbus:** parmi les fonctions les plus courantes.

-
- **Adresse:** Modbus Poll prend en charge deux formats d'adressage :
 - **Adresse de protocole:** format numérique direct (0 à 65535), réellement envoyé sur le réseau. C'est le format utilisé par défaut dans Modbus Poll.
 - **Adresse symbolique (type PLC):** utilise un préfixe pour indiquer le type de donnée

Le logiciel permet aussi l'utilisation d'adresses à 5 ou 6 chiffres selon le matériel utilisé.

- **Quantité:** nombre d'éléments à traiter dans l'opération.
- **Fréquence de balayage (Scan rate):** elle détermine la fréquence à laquelle les requêtes sont envoyées. Elle peut aller de 1 à 60000 ms.
- **Désactivation de la communication:**
 - **Lecture/Écriture désactivée:** une fois cochée, plus aucune requête n'est envoyée et l'indication « (Disabled) » apparaît dans le titre de la fenêtre.
 - **Désactiver en cas d'erreur:** interrompt automatiquement la communication si une erreur est détectée.
- **Options d'affichage:**
 - **Lignes:** définit le nombre de lignes visibles dans le tableau de données.
 - **Masquer les colonnes de noms:** masque la colonne des alias pour économiser de l'espace si ces noms ne sont pas utilisés.
 - **Adresse dans les cellules:** Affiche l'adresse de chaque registre directement dans la cellule.
 - **Adresses PLC (Base 1):** Affiche les adresses des registres en commençant à 1, conformément à la convention de certains automates programmables industriels.
 - **Mode Enron/Daniel:** Adapte l'affichage à des dispositifs utilisant un format d'adresse spécifique non standard.

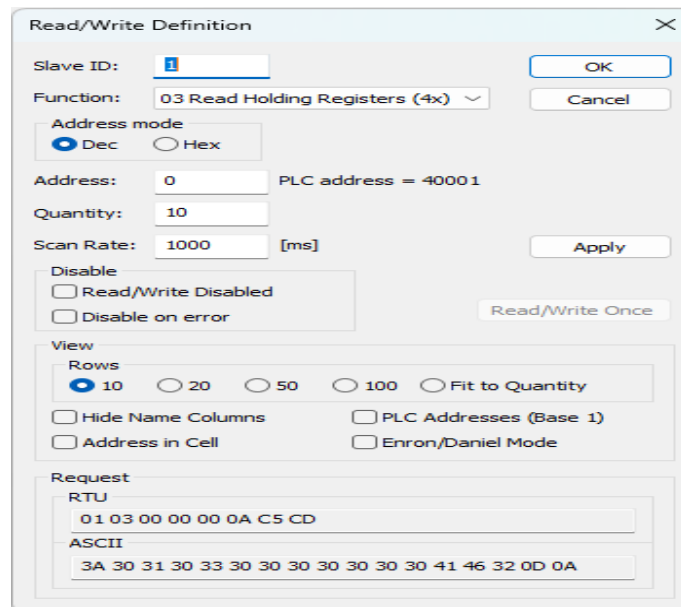


Figure 71: Fenêtre Read/Write Definition dans Modbus Poll

Surveillance et visualisation:

Modbus Poll intègre une fonctionnalité de surveillance du trafic Modbus, permettant d'afficher en temps réel les trames échangées entre le maître et l'esclave.

Ces informations sont visibles dans une fenêtre dédiée, accessible via le menu Affichage → Communication.

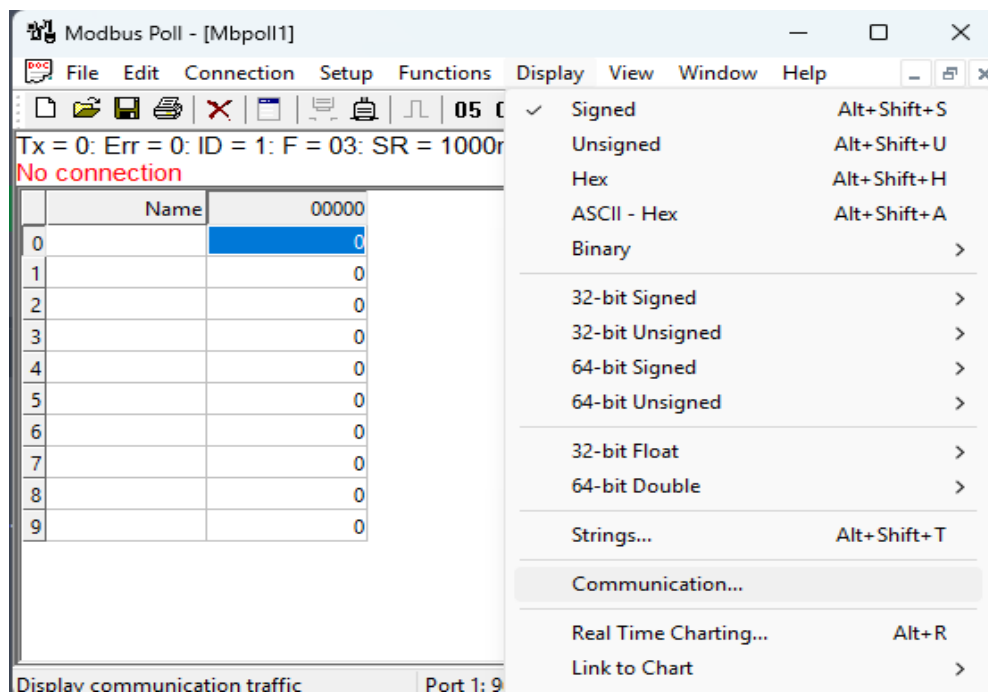


Figure 72: Accès à la fenêtre Communication Traffic dans Modbus Poll

Cette interface permet de suivre en temps réel les échanges entre le maître Modbus et les dispositifs esclaves, qu'ils soient connectés en série ou via TCP/IP. Elle affiche

plusieurs informations pour chaque trame:

Table 14: Exemple d'enregistrement des trames dans le journal de communication Modbus Poll

No.	Time	Length	Tx/Rx	Packet
Numéro d'ordre du message.	Horodatage de la trame.	Longueur de la trame.	Transmission (Tx) ou Réception (Rx).	Contenu de la trame en hexadécimal.

Des boutons en haut de l'interface permettent de:

- **Stop** : interrompre temporairement l'affichage pour analyser une trame,
- **Clear** : effacer l'historique visible,
- **Copy** : copier les trames sélectionnées,
- **Log** : lancer ou arrêter l'enregistrement dans un fichier,
- **Stop on Error** : figer l'affichage automatiquement dès qu'une erreur est détectée.

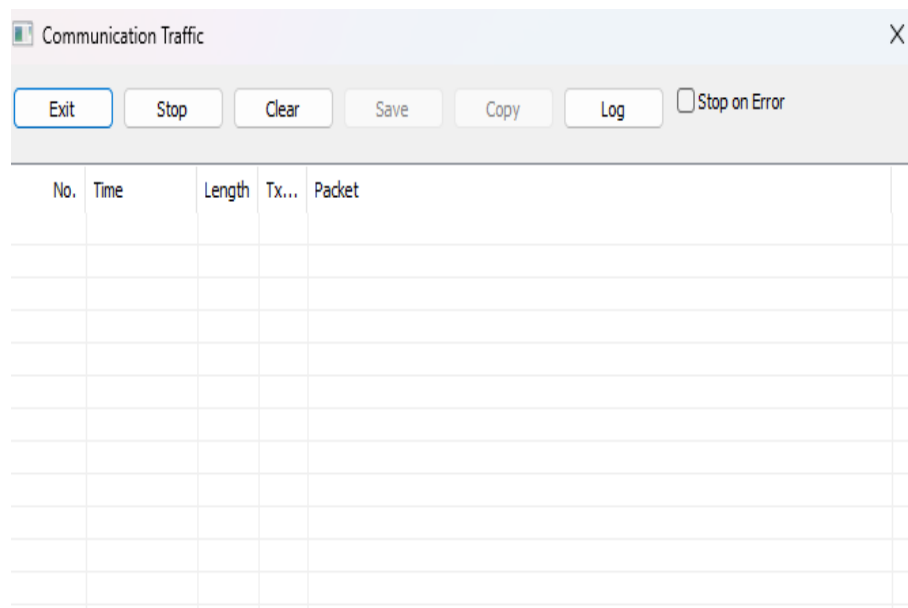


Figure 73: Fenêtre **Communication Traffic** dans Modbus Poll

Enregistrement des données:

Modbus Poll permet de journaliser automatiquement les échanges Modbus, ce qui facilite la traçabilité, l'analyse postérieure des mesures, ainsi que la détection d'éventuelles anomalies. Deux modes d'enregistrement sont proposés:

-
1. **Enregistrement des données:** L'accès à cette fonction se fait via le menu **Setup** → **Log**, ou par le raccourci clavier **Alt + L**.

L'utilisateur peut choisir le mode de déclenchement (à chaque requête ou à intervalles fixes), définir le format de sortie (largeur fixe, séparateur virgule ou tabulation), et activer des options d'optimisation telles que l'enregistrement uniquement en cas de modification, l'ajout de l'horodatage, ou encore l'inclusion des erreurs de communication (CRC, timeout...). L'écriture peut se faire ligne par ligne sans mise en cache, et il est possible de cumuler les données dans un même fichier.

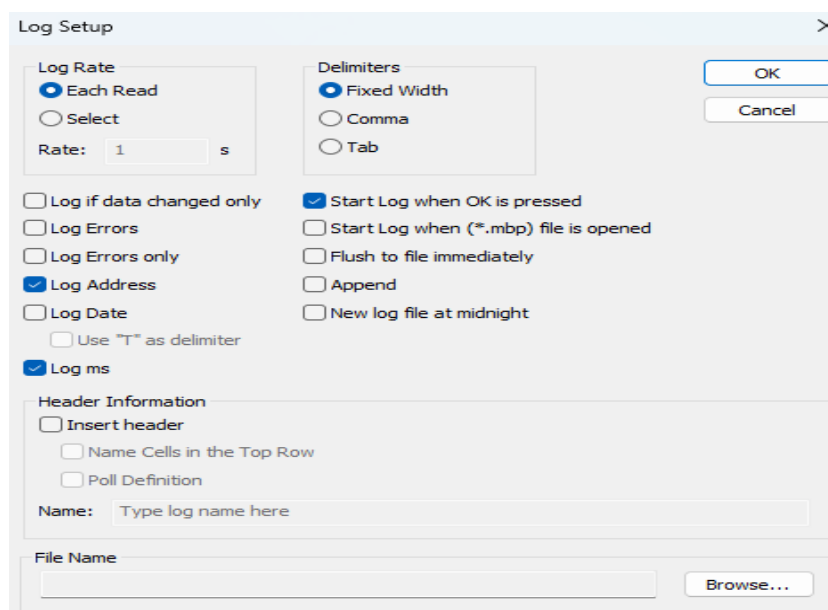


Figure 74: Fenêtre **Log Setup** dans Modbus Poll

2. **Enregistrement dans Microsoft Excel:** Accessible via le menu **Setup** → **Excel Log**, ou par le raccourci clavier **Alt + X**, cette option nécessite une instance active d'Excel.

L'utilisateur peut configurer le taux d'enregistrement (à chaque lecture ou à intervalle défini), fixer une limite de lignes à enregistrer, et choisir d'insérer un en-tête contenant le nom du journal, la définition du poll (ID, fonction, etc.), ainsi que les alias des cellules. Cette fonctionnalité facilite l'analyse des données et leur exploitation ultérieure. Il est recommandé de ne pas interagir avec Excel pendant la journalisation, sous peine d'interruption du processus.

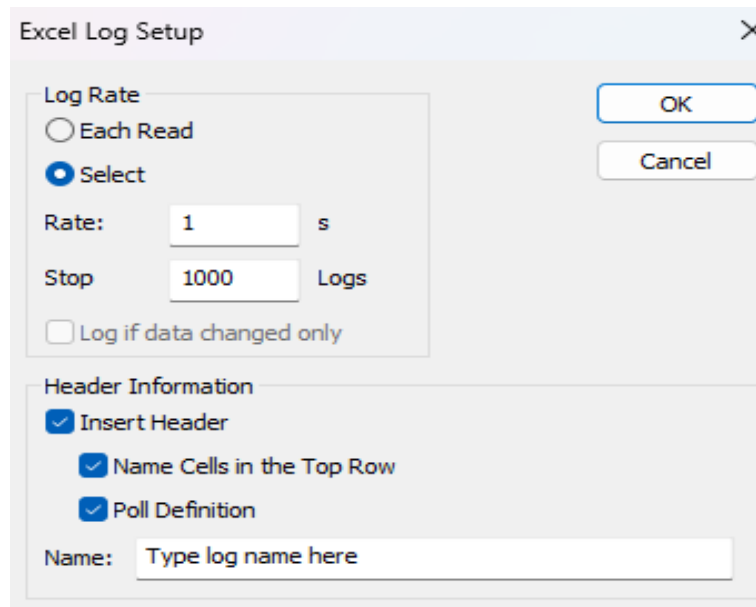


Figure 75: Fenêtre Excel Log Setup dans Modbus Poll

EasyModbus

EasyModbus est une bibliothèque logicielle qui implémente les protocoles Modbus TCP, UDP et RTU. Elle permet de développer facilement des applications cliente (maitre) ou serveur (esclave) pour tester, contrôler ou simuler des équipements Modbus. L'interface principale d'EasyModbus se présente sous la forme d'une fenêtre à onglets:

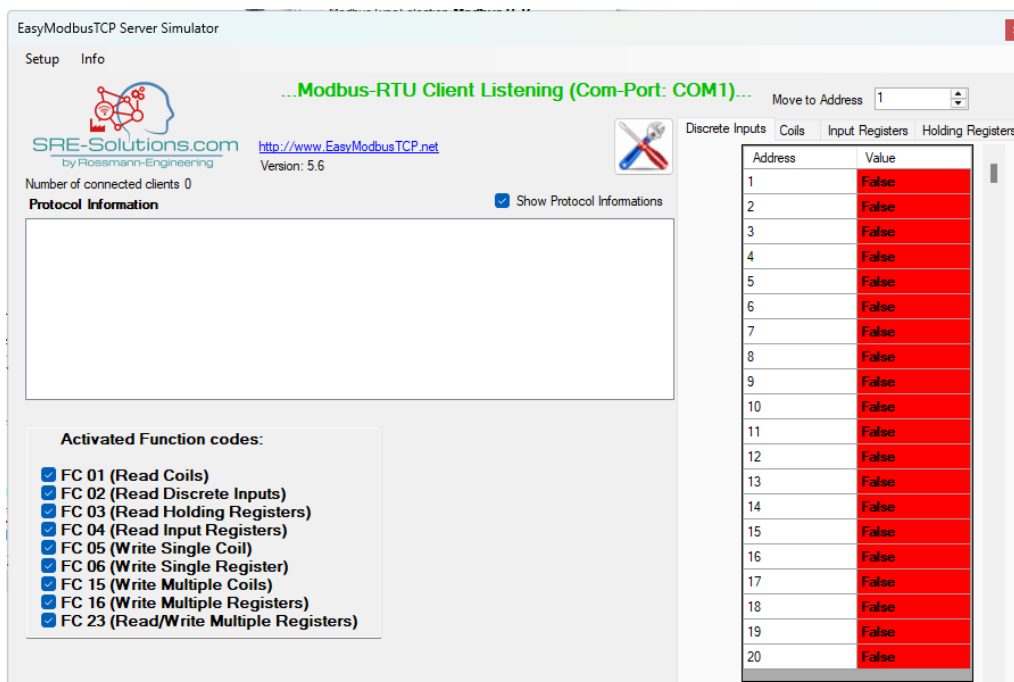


Figure 76: Interface du logiciel EasyModbus

Configuration de la communication:

Accessible via le bouton **Setup**, cette option ouvre la boîte de dialogue **Properties**, dans laquelle on sélectionne le mode RTU (port série COM) ou TCP/IP (port 502 par défaut). Les boutons **Discard** et **Accept** permettent respectivement d'annuler ou de valider les modifications. Le mode actif s'affiche en vert en haut, au centre de la fenêtre.

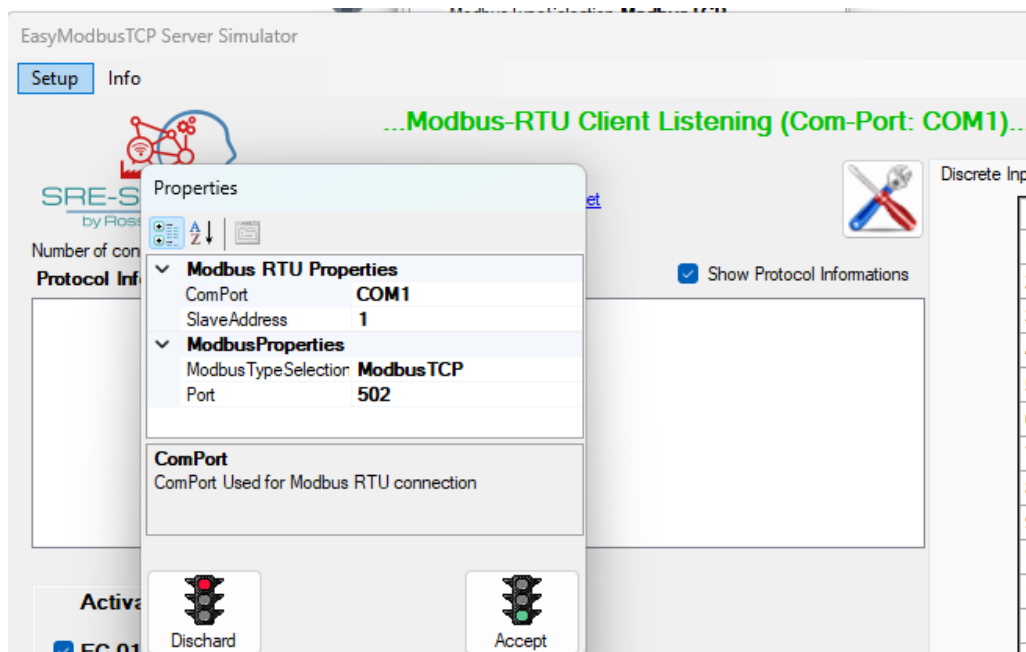


Figure 77: Configuration de la communication dans EasyModbus

Métadonnées du simulateur:

Accessible via le bouton **Info**, cette section ouvre la boîte de dialogue correspondante et affiche les informations clés du serveur EasyModbus: l'URL du projet, le numéro de version du logiciel et le nom du fournisseur.



Figure 78: Métadonnées du simulateur EasyModbus Server

Fonctions supportées:

Affiche la liste des codes de fonction actifs.

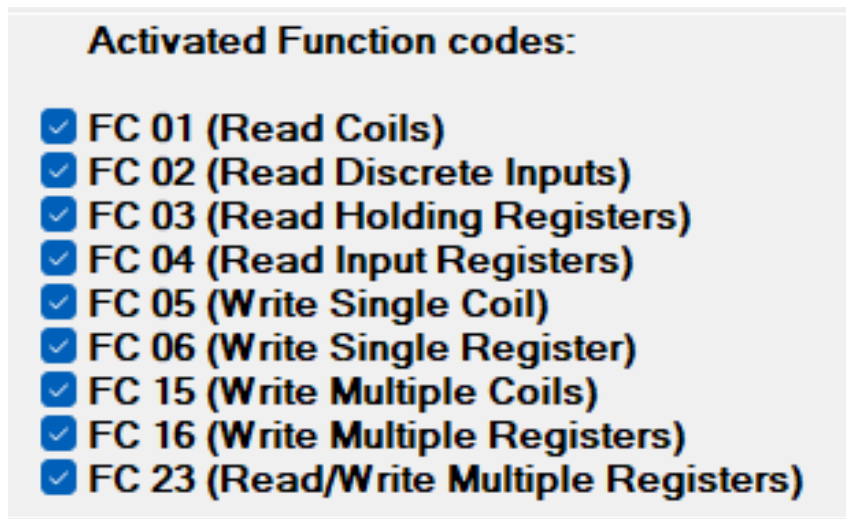


Figure 79: Fonctions supportées dans EasyModbus

Surveillance et interaction:

- **Mises à jour en temps réel:** les tableaux Discrete Inputs, Coils, Input Registers et Holding Registers se rafraîchissent dès qu'une requête client est reçue ou qu'une valeur est modifiée manuellement.
- **Édition directe:** double-clic sur une cellule pour changer son état (booléen) ou sa valeur (16bits).
- **Navigation rapide:** Move to Address est utilisée pour changer le début de l'adressage des différentes données.

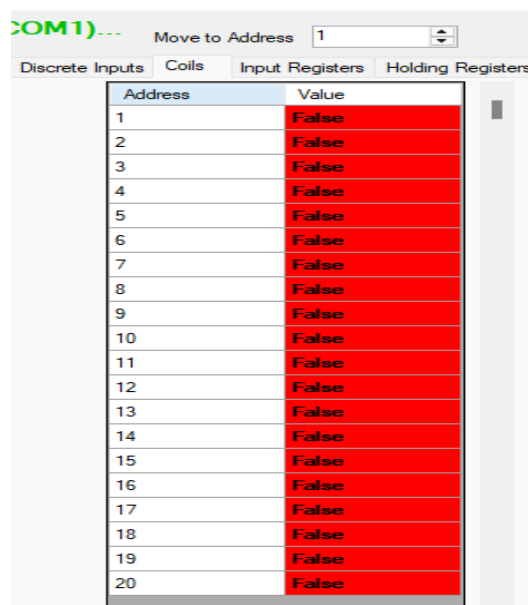


Figure 80: Types des données dans EasyModbus

Logs et diagnostics:

- **Afficher Protocol Informations:** affiche les trames Modbus brutes (hexadécimal) échangées, utile pour déboguer les échanges sur RS-485 ou TCP/IP.
- **Nombre de clients connectés:** indicateur en temps réel du nombre de masters ou automates simultanément attachés au serveur EasyModbus.

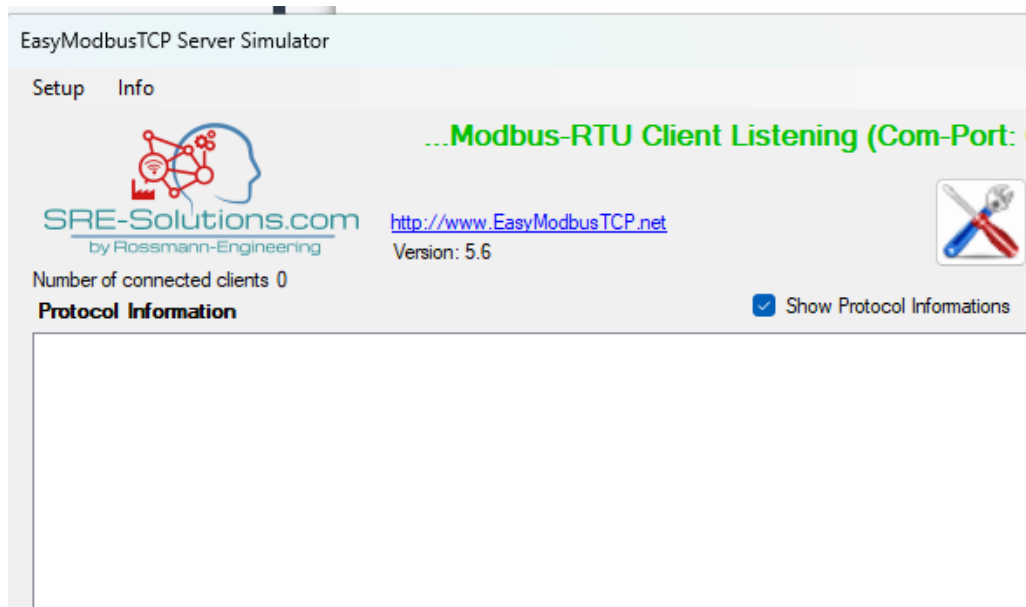


Figure 81: Protocole d'information dans EasyModbus

VSPE

VSPE est un émulateur de ports série virtuels développé par Eterlogic, destiné aux ingénieurs et développeurs souhaitant tester, déboguer ou simuler des communications série sans matériel réel.

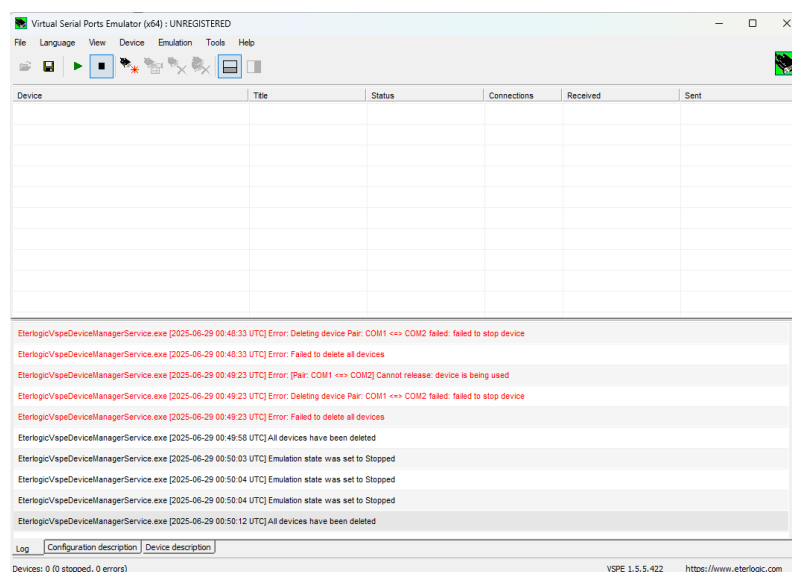


Figure 82: Interface de l'émulateur VSPE

Configuration des ports

Il s'installe comme un pilote Windows, expose une interface de gestion des périphériques:

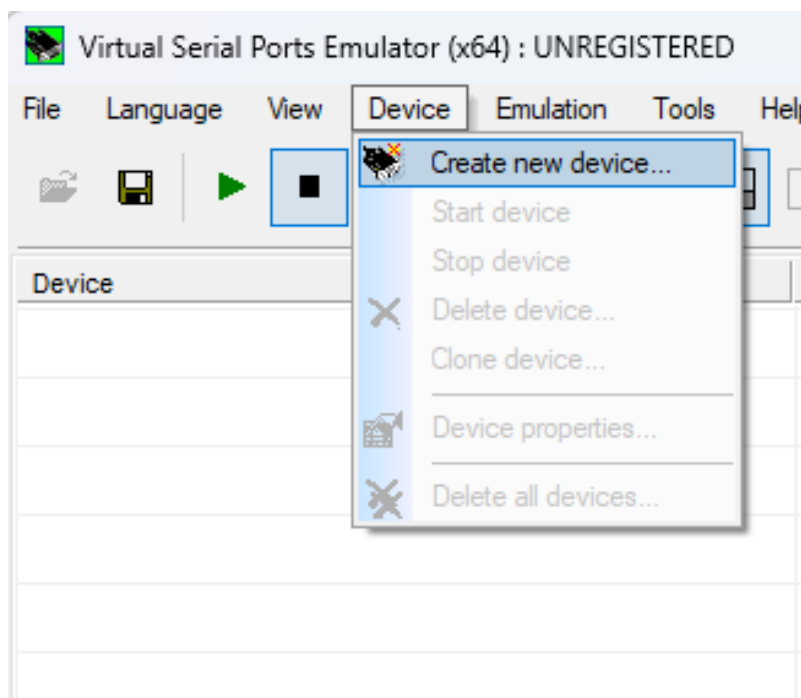


Figure 83: Création d'un périphérique dans VSPE

Après la création de nouveau périphérique, on choisit le type de périphériques à utiliser pour établir la communication:

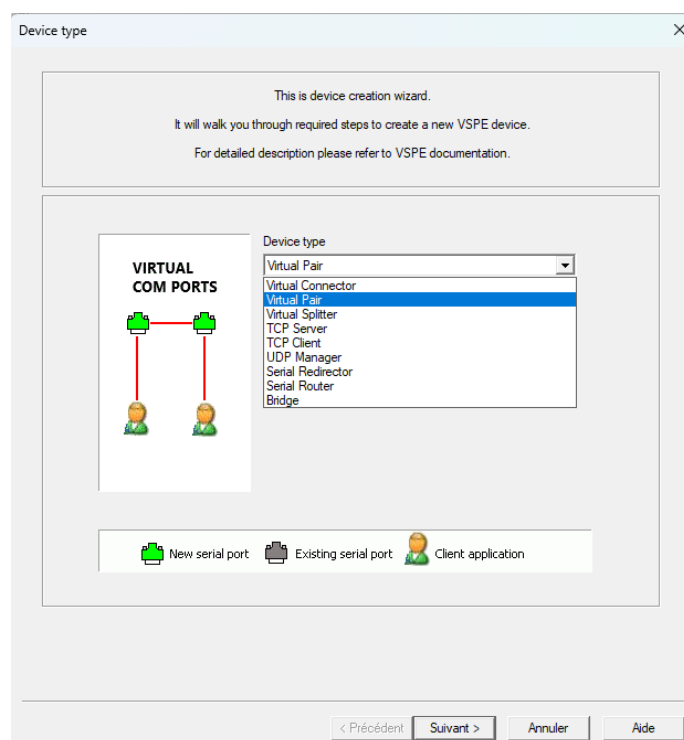


Figure 84: Type des périphériques dans VSPE

Il fournit un outil de surveillance en temps réel du trafic hexadécimal entre ports virtuels. Idéal pour l'architecture de simulation SCADA, VSPE vous permet de relier côté maître et côté esclave deux applications Modbus via un lien série bidirectionnel entièrement logiciel, tout en capturant et diagnostiquant chaque trame pour valider la configuration et détecter les erreurs de transmission.

Architecture de simulation

L'objectif de cette architecture est de simuler une communication Modbus RTU en boucle locale, sans matériel physique, en utilisant exclusivement des outils logiciels. La liaison maître-esclave est établie entre Modbus Poll (maître) et EasyModbus RTU Server (esclave) via des ports série virtuels créés avec VSPE.

Étape 1: Création de ports virtuels avec VSPE:

1. Lancer l'application Virtual Serial Ports Emulator (VSPE).
2. Créer une paire de ports série virtuels.
3. Cette paire permet de simuler un câble RS-232/RS-485 en connectant deux applications.

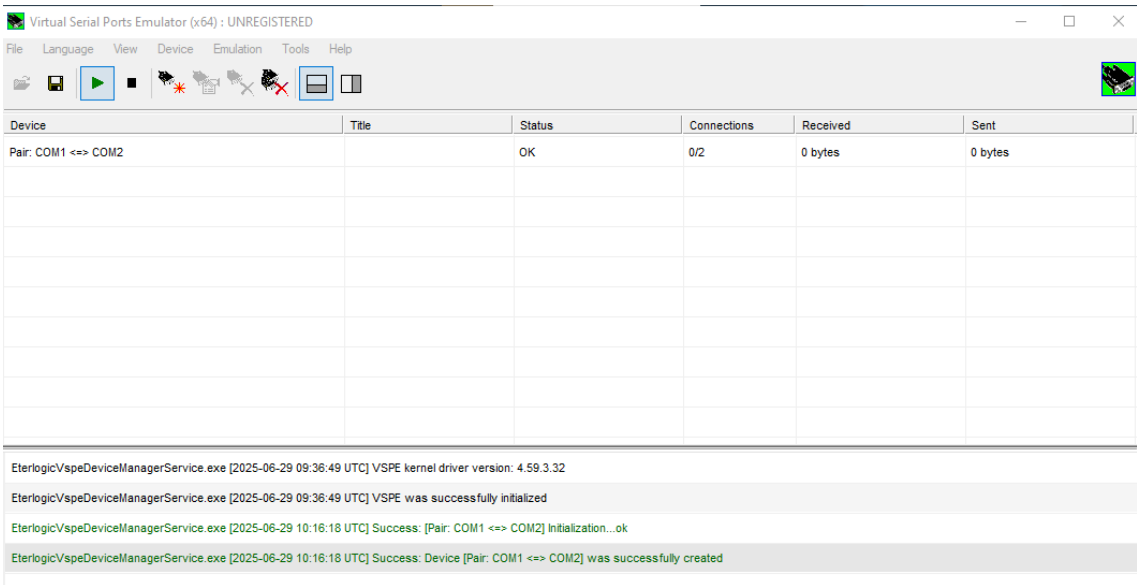


Figure 85: Création de ports virtuels avec VSPE

Étape 2: Lancement d'EasyModbus RTU Server:

1. Ouvrir le logiciel EasyModbusTCP/RTU Server Simulator.
2. Accéder au menu Setup, puis à Modbus Properties.
3. Sélectionner le mode RTU, et choisir COM1 comme port de communication.
4. Valider avec le bouton Accept pour activer le serveur esclave.

5. L'interface permet de visualiser et modifier les valeurs des registres.

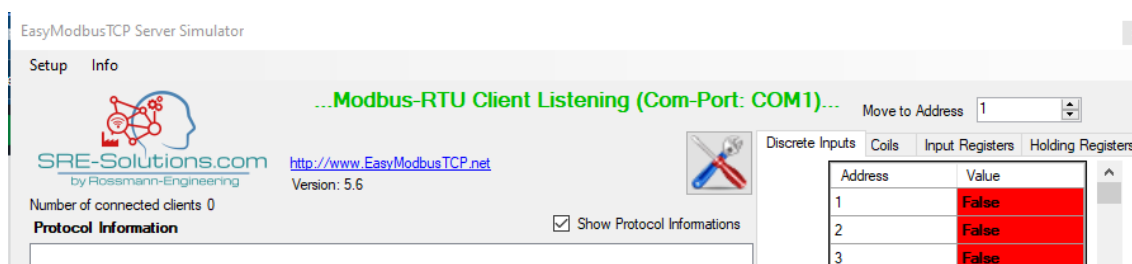


Figure 86: Lancement d'EasyModbus RTU Server

Étape 3 – Configuration de Modbus Poll Maître

1. Lancer Modbus Poll.
2. Configurer les paramètres de communication série selon le mode RTU standard.
3. Définir les requêtes Modbus en choisissant le type de fonction à exécuter et les adresses concernées.

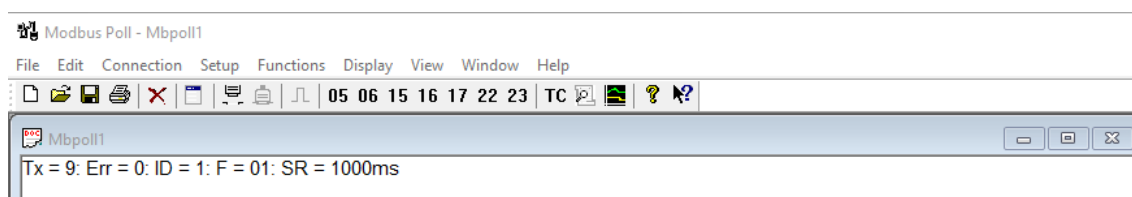


Figure 87: Configuration de Modbus Poll Maître

La figure suivante présente la configuration complète de la communication Modbus RTU en environnement simulé:

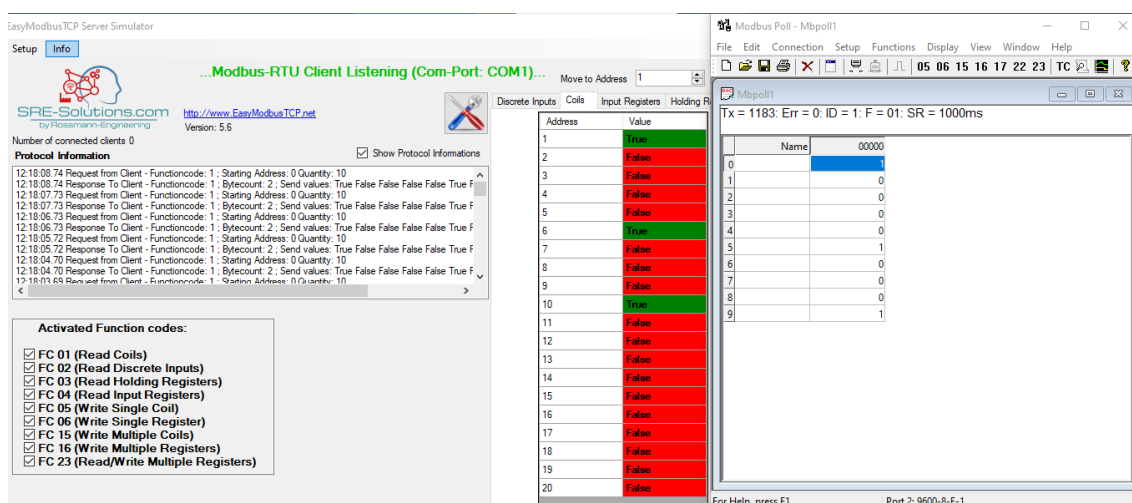


Figure 88: établissement de la communication série RTU

Fonctions testées

Dans le cadre de l'expérimentation, deux fonctions fondamentales du protocole Modbus RTU ont été testées:

1. Fonction 03 – Lecture de registres (Read Holding Registers):

Les valeurs lues s'affichent dans Modbus Poll et reflètent en temps réel les contenus des registres définis dans EasyModbus Server.

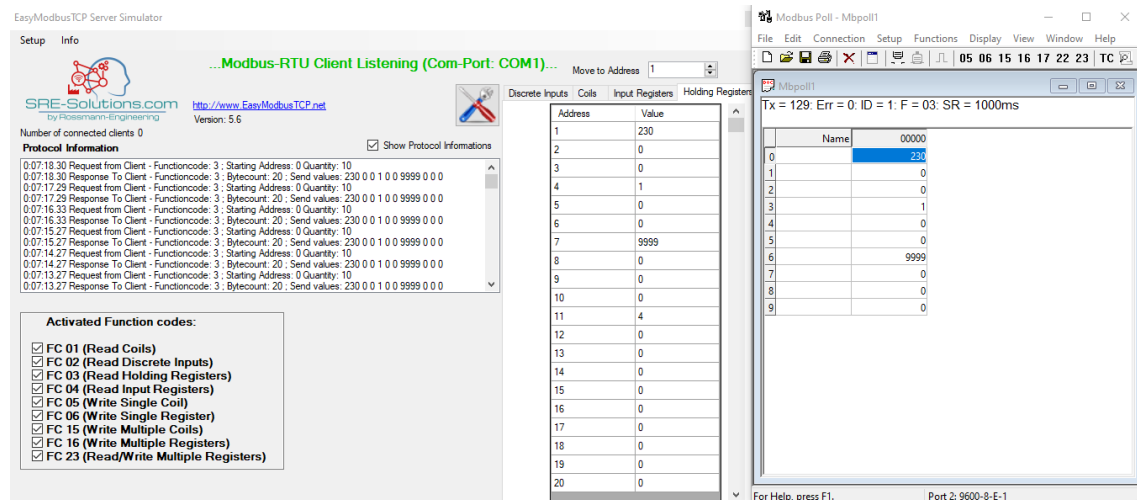


Figure 89: Communication Modbus RTU utilisant la fonction 03

2. Fonction 16 – Écriture multiple dans des registres (Write Multiple Holding Registers):

Les registres correspondants dans EasyModbus Server sont mis à jour instantanément à chaque écriture, avec confirmation dans la fenêtre des trames Modbus.

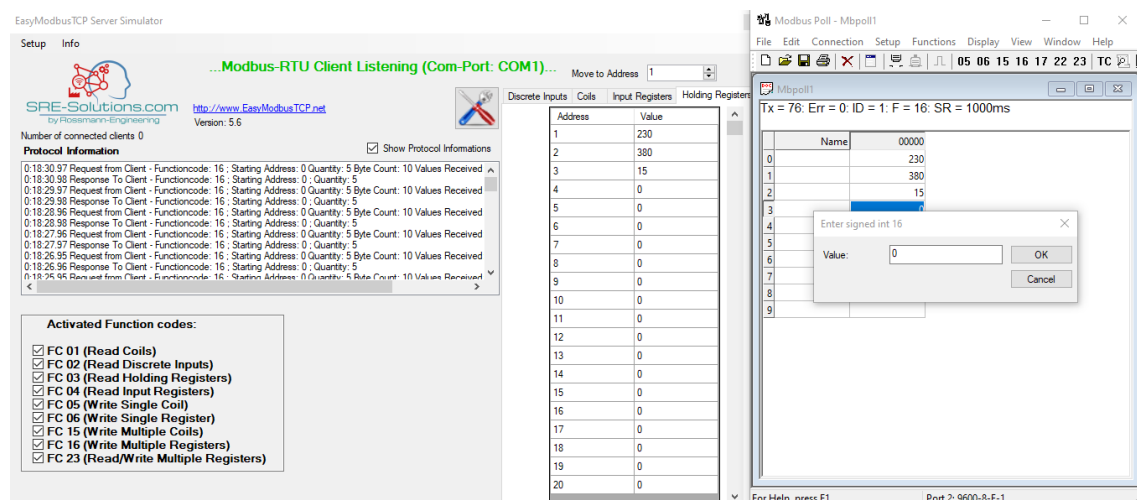


Figure 90: Communication Modbus RTU utilisant la fonction 16

Observations et analyse des trames Modbus

L'objectif de cette étape est de valider la conformité des échanges entre Modbus Poll et EasyModbus au protocole Modbus RTU. L'analyse s'appuie sur l'observation des trames émises Tx et reçues Rx, en vérifiant la structure, les champs d'adresse, le code fonction, la longueur des données, le CRC ainsi que le respect des délais configurés.

Les trames peuvent être consultées en temps réel via l'outil *Communication Traffic* dans Modbus Poll, et la fenêtre *Protocol Information* dans EasyModbus. L'exemple suivant illustre un échange cyclique utilisant la fonction 03, déclenché toutes les 1000 ms conformément au *scan rate* configuré.

No.	Time	Length	Tx...	Packet
0	00:56:42.763	8	Tx	01 03 00 00 00 0A C5 CD
1	00:56:42.794	25	Rx	01 03 14 00 E6 01 7C 00 0F 00 00 00 00 00 00 00 00 00 00 00 00 00 00 62 19
2	00:56:43.765	8	Tx	01 03 00 00 00 0A C5 CD
3	00:56:43.797	25	Rx	01 03 14 00 E6 01 7C 00 0F 00 00 00 00 00 00 00 00 00 00 00 00 00 00 62 19
4	00:56:44.766	8	Tx	01 03 00 00 00 0A C5 CD
5	00:56:44.798	25	Rx	01 03 14 00 E6 01 7C 00 0F 00 00 00 00 00 00 00 00 00 00 00 00 00 00 62 19
6	00:56:45.771	8	Tx	01 03 00 00 00 0A C5 CD
7	00:56:45.803	25	Rx	01 03 14 00 E6 01 7C 00 0F 00 00 00 00 00 00 00 00 00 00 00 00 00 00 62 19
8	00:56:46.777	8	Tx	01 03 00 00 00 0A C5 CD
9	00:56:46.808	25	Rx	01 03 14 00 E6 01 7C 00 0F 00 00 00 00 00 00 00 00 00 00 00 00 00 00 62 19
10	00:56:47.781	8	Tx	01 03 00 00 00 0A C5 CD
11	00:56:47.813	25	Rx	01 03 14 00 E6 01 7C 00 0F 00 00 00 00 00 00 00 00 00 00 00 00 00 00 62 19

Figure 91: Échange de trames Modbus – Fonction 03

Analyse des trames:

Table 15: Trame envoyée (Tx) – Fonction 03

01	03	00 00	00 0A	C5 CD
ID esclave	Code fonction	Adresse de départ	Nombre de registres à lire (10)	CRC

Table 16: Trame reçue (Rx) – Fonction 03

01	03	14	00 E6	01 7C	00 0F	00 00 (×7)	62 19
ID esclave	Code fonction	Nb. d'octets (20)	230	380	15	Registres 4 à 10 = 0	CRC

L'analyse confirme que:

- La structure de la réponse respecte les spécifications de la fonction 03.
- Le CRC est valide, garantissant l'intégrité des données.

-
- Les valeurs lues sont cohérentes avec celles affichées dans les fenêtres *Holding Registers* des deux logiciels.
 - L'intervalle entre les requêtes est conforme au *scan rate* défini (1000 ms).