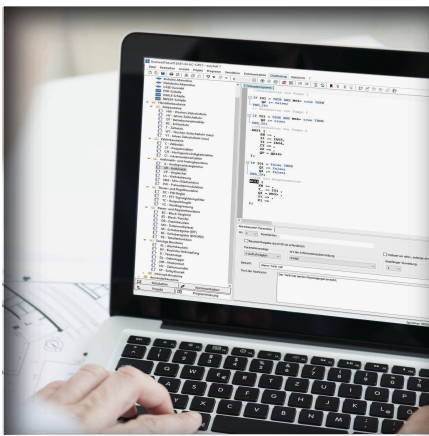


La nouvelle génération de modules logiques pour la mise en œuvre de solutions d'automatisation Build it in.

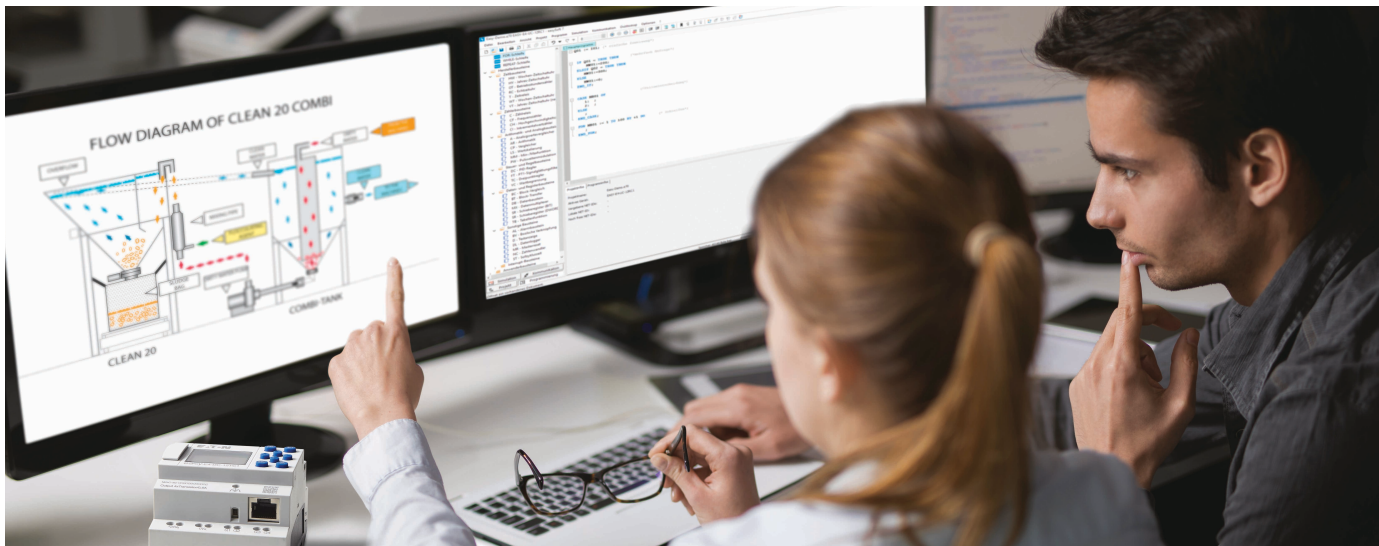


Ralf Cieslewicz

1^{ère} édition

EATON

Powering Business Worldwide



La nouvelle génération de modules logiques pour la mise en œuvre de solutions d'automatisation

L'embaras du choix : modules logiques ou automates programmables

Qu'il s'agisse de la conception d'une nouvelle machine, de la modernisation d'une machine existante ou de l'automatisation d'un bâtiment de pointe, il convient de choisir le type de solution d'automatisation adéquat. Plusieurs facteurs sont à prendre en compte : Quelle est la complexité de l'application ? Dans quelle mesure la solution est-elle évolutive ? De quelles qualifications les programmeurs, les ingénieurs de mise en service et le personnel de maintenance doivent-ils disposer ? Enfin, la solution doit également être économique - les coûts ne dépendent pas uniquement du matériel. Il faut également prendre en considération les investissements potentiels en matière de formation du personnel. Les modules logiques sont utilisés dans l'automatisation de bâtiments et d'applications industrielles de faible à moyenne complexité depuis plusieurs années. Néanmoins, les entreprises devant choisir une solution d'automatisation ont souvent opté pour les automates ; en effet, les solutions de modules logiques ont souvent été considérées comme peu évolutives. Est-ce toujours le cas aujourd'hui ?

L'époque où les modules logiques ne représentaient qu'une alternative économique aux relais temporisés est révolue depuis longtemps. Comparée aux automates souvent moins polyvalents, la dernière génération de modules logiques offre des fonctionnalités matérielles et de programmation d'une plus grande polyvalence ouvrant de toutes nouvelles possibilités d'applications dans le domaine des nano-commandes. Les modules logiques offrent désormais des fonctionnalités autrefois réservées aux automates traditionnels : langages de programmation avancés tels que le texte structuré (ST), possibilité d'interrompre le processus, création de blocs fonctionnels utilisateur, systèmes d'exploitation téléchargeables, enregistrement des données, horloge temps réel ou fonctions Watchdog.

De plus, le nombre d'E/S a également été augmenté pour répondre aux nouveaux défis. Aujourd'hui, les machines et les systèmes doivent fournir toujours plus d'informations, en particulier dans le contexte des applications IIoT (Internet Industriel des Objets). Si autrefois le simple message « moteur en marche » ou « moteur arrêté » suffisait, les utilisateurs veulent aujourd'hui connaître la vitesse exacte du moteur. Les constructeurs ont donc équipé leurs modules logiques d'entrées spéciales, pour codeurs incrémentaux, compteurs de fréquence, compteurs rapides, etc. Ils ont également adapté le nombre d'entrées/sorties analogiques et TOR aux futures demandes, en raison du nombre croissant d'actionneurs et de capteurs connectés nécessitant de recueillir leurs données.

Une programmation plus polyvalente

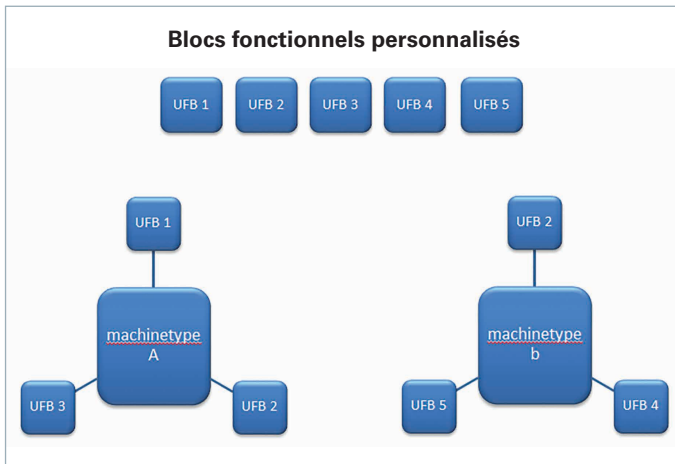
Auparavant, choisir un module logique par constructeur revenait souvent à choisir un langage de programmation. Les outils logiciels de pointe destinés à la programmation des modules logiques sont aujourd'hui beaucoup plus polyvalents.

Pour réaliser leurs projets, les utilisateurs peuvent désormais choisir entre le langage de programmation habituel et spécifique aux versions précédentes ou un langage standard comme le schéma à contacts (LD), le schéma fonctionnel (FBD) ou le texte structuré (ST). L'option de programmation directe sur l'appareil reste une possibilité pour réaliser de petits projets même si l'on peut douter de l'utilité. Les fonctionnalités avancées des modules logiques de la nouvelle génération, devenus désormais incontournables, requièrent l'utilisation d'outils logiciels.

Contrairement aux environnements logiciels complexes des automates programmables, les coûts d'acquisition de ces outils logiciels sont extrêmement faibles et n'influent ainsi en rien sur la prise de décision.

Les blocs fonctionnels intégrés, dont la configuration est très simple, constituent l'un des principaux avantages des modules logiques. Alors qu'à l'origine, ces modules étaient principalement destinés à l'automatisation des bâtiments, comme les minuteriers hebdomadaires/annuelles ou les minuteriers d'escalier, ils peuvent dorénavant être utilisés pour des tâches complexes de commande et de régulation. Les blocs arithmétiques et de modulation de largeur d'impulsion ainsi que les contrôleurs à trois positions et les régulateurs PID sont dorénavant des éléments standards.

En plus de la vaste bibliothèque de blocs fonctionnels prédéfinis, les nouveaux outils de programmation permettent de créer des blocs fonctionnels définis par l'utilisateur (UF), intégrables à tout type de projet. Une fois les UFs programmés et testés, l'ajout de différents types de machines est d'une simplicité inégalée à toute les phases d'expansion de l'installation. Comme les erreurs de programmation sont réduites au minimum, les tests et la mise en service requièrent moins de temps.



Communication – Mise en réseau – IIoT

Auparavant, les constructeurs de modules logiques s'appuyaient souvent sur leurs propres protocoles de communication et de réseau : cette tendance est en train de changer également. Le protocole Ethernet standard assure désormais la communication avec l'appareil, si bien que l'infrastructure Ethernet existante, notamment les routeurs, commutateurs et adaptateurs WLAN, peut être utilisée pour communiquer avec l'appareil. Les serveurs Web intégrés génèrent des pages Web au format HTML5, pouvant être affichées sur tous les navigateurs standards ainsi que sur des appareils mobiles (smartphones ou tablettes), ce qui favorise de nouveaux concepts opérationnels.

De plus, la mise en réseau des modules logiques n'est plus assurée par des protocoles série, mais par des protocoles de bus de terrain basés sur Ethernet. Le transfert des données vers le système de contrôle est ainsi plus sûr et plus rapide.

Les données peuvent également être transférées vers le cloud pour la surveillance d'état de l'appareil ou la maintenance prédictive, soit indirectement via des systèmes de commande intermédiaires, soit directement en utilisant les passerelles IIoT existantes dans le réseau.

Conclusion

Dans un avenir prévisible, les modules logiques ne remplaceront pas les automates traditionnels dont la programmation est basée sur la norme internationale CEI 61131. Cependant, pour les applications de faible et moyenne complexité, ils offrent une alternative moderne, évolutive et surtout économique aux solutions d'automatisation traditionnelles.

Dans les paragraphes suivants, vous trouverez une analyse détaillée des fonctionnalités de la nouvelle génération de modules logiques, à l'instar du nouveau easyE4 d'Eaton.

1. De l'armoire aux solutions cloud computing

La connectivité IIoT revêt une importance croissante, en particulier lorsqu'il s'agit d'assurer l'évolutivité d'un système. La question est de savoir si les modules logiques sont en mesure d'affronter la concurrence dans ce domaine.

1.1 Modularité au niveau local

Au niveau le plus élémentaire, l'IIoT a pour objet le recueil de données. Aujourd'hui, les machines et les systèmes sont équipés de plus en plus de capteurs et d'actionneurs afin d'obtenir des informations de plus en plus détaillées sur leur état actuel.

La conception du easyE4 prend évidemment en compte ces éléments. Il est possible d'étendre chaque appareil de base avec un maximum de 11 modules d'E/S analogiques et TOR. Ainsi, les signaux d'entrée, le traitement et l'écriture de données jusqu'à 96 entrées TOR + 92 sorties TOR ou 48 entrées analogique + 22 sorties analogique (12 bits) s'ajoutent à la configuration. Ces modules peuvent être combinés en termes de fonction et de tension : 12 V DC, 24 V AC, 24 V DC ou 230 V AC.

Le niveau des E/S peut ainsi être configuré séparément en fonction des exigences spécifiques de l'application.

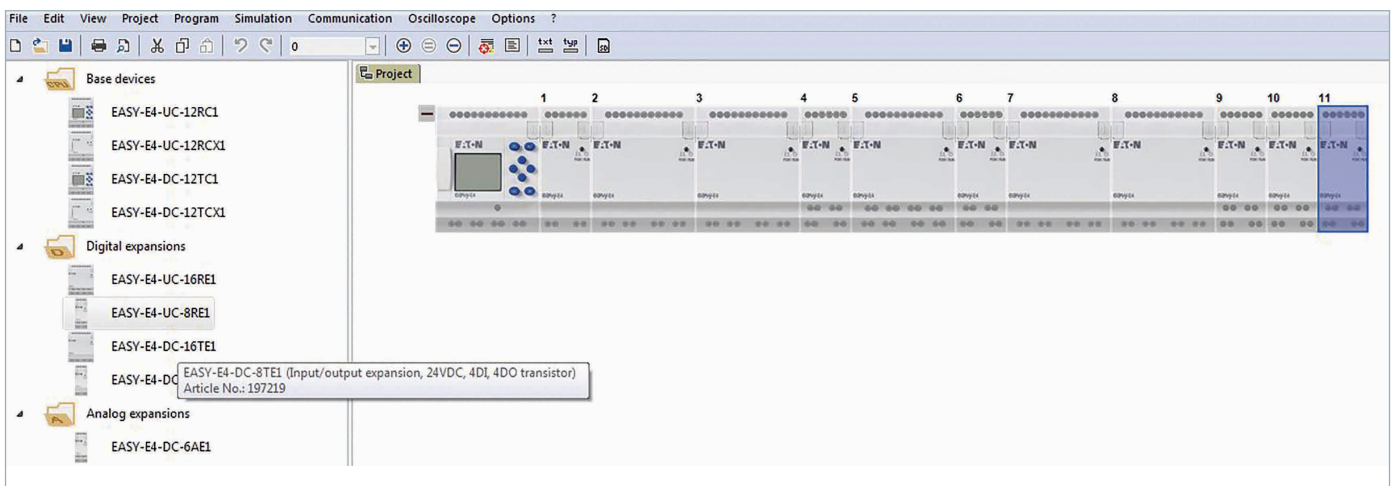


Fig. 1: Appareil de base easyE4 avec 11 extensions

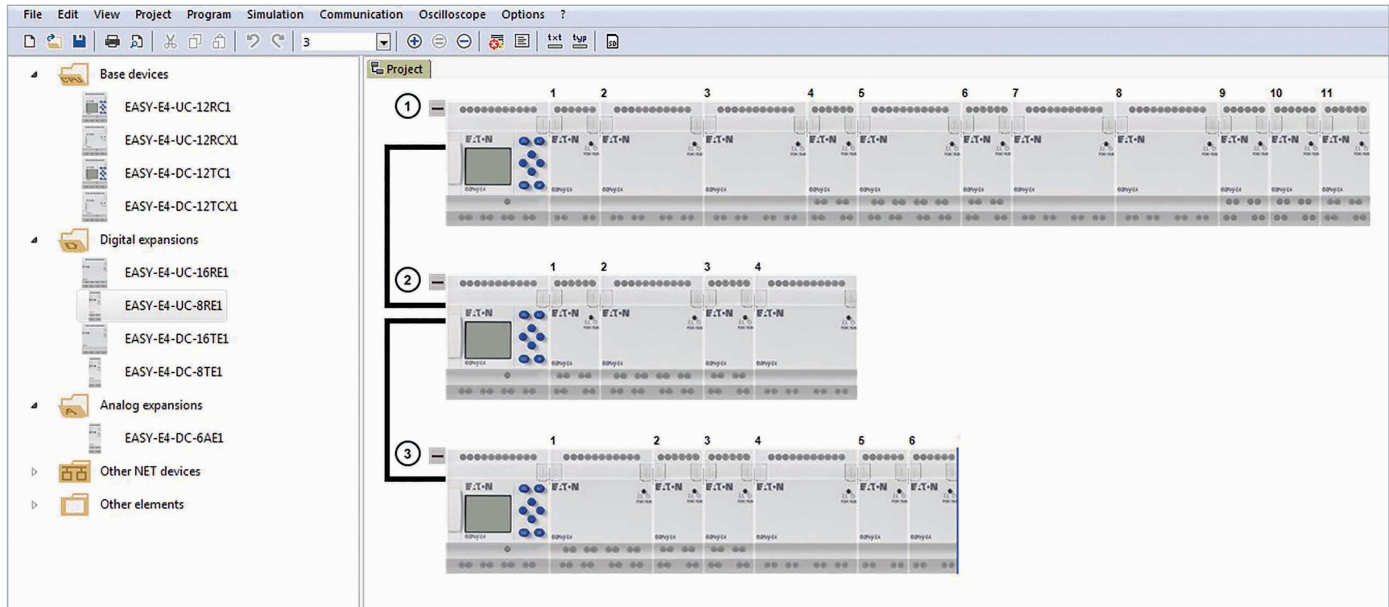


Fig. 2 : Trois appareils de base easyE4 mis en réseau

1.2 Structures en réseau local

Si un système comprend plus d'une application autonome, les structures mises en réseau jouent un rôle déterminant. Les programmes adaptés à un élément spécifique du système fonctionnent indépendamment des autres unités fonctionnelles et seules les informations pertinentes pour les autres modules en réseau sont partagées via Ethernet. Cette répartition des tâches facilite la configuration des différents éléments du système, ce qui permet d'augmenter la vue d'ensemble dans les applications plus importantes.

1.3 Architectures de systèmes avec communication en cloud

Selon les critères décrits aux points 1.1 et 1.2, les données ne sont disponibles que localement. Pour le développement d'applications de pointe, cela ne suffit plus. A l'avenir, les processus de planification de la production, de contrôle de la qualité et de maintenance exigeront une plus grande disponibilité des données du système. La capacité du easyE4 de fonctionner en tant que serveur Modbus TCP permet de créer un système ouvert pour envoyer des données de processus. Les clients Modbus TCP, notamment l'automate Eaton XC300 ou l'écran tactile du terminal-automate XV300, peuvent ensuite recueillir les données fournies par un groupe easyE4 pour un traitement ultérieur. Les passerelles/routeurs IoT s'intègrent facilement dans cette architecture Ethernet, permettant de transférer les données vers le cloud et donc de les récupérer à tout moment et en tout lieu.

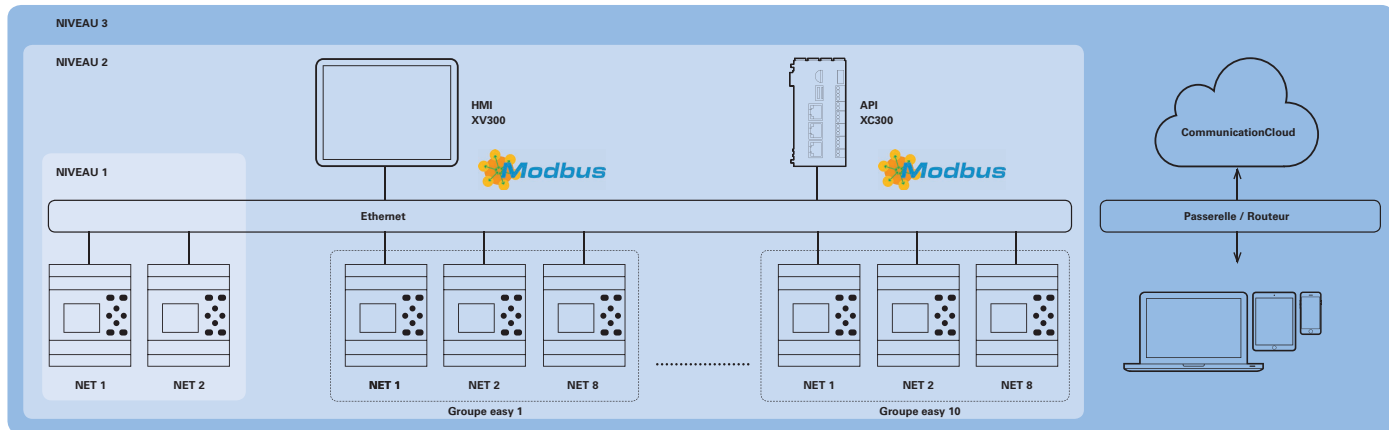


Fig. 3 : Un exemple d'architecture de système avec easyE4

2. Programmation

2.1 Importer des programmes existants

Le développement de nouvelles gammes de modules logiques et l'amélioration continue des appareils existants représentent un défi majeur pour les constructeurs. D'une part, le marché exige de nouvelles possibilités de programmation et d'autre part, des milliers de clients à prendre en compte sont habitués aux langages de programmation actuels.

Garder cette compatibilité avec les modèles et les outils de programmation existants est donc une priorité absolue. Est-ce suffisant ? Après tout, les mêmes restrictions s'appliquent en cas de transfert d'un programme à partir d'une version existante. Cela limite les possibilités d'intégrer de nouvelles fonctions et rend donc difficile toute amélioration et tout développement ultérieurs du programme.

Comment concilier le meilleur des deux mondes ? La nouvelle version easySoft V7 d'Eaton possède tous les atouts.

Lors de l'ouverture d'un programme développé avec une ancienne version d'easySoft, le logiciel est en mesure de le reconnaître et permet à l'utilisateur de choisir le langage de programmation qu'il préfère.

Les utilisateurs peuvent continuer à travailler dans un environnement de programmation familier. Ils n'ont pas besoin d'apprendre un nouveau langage de programmation, et cette forme d'importation est suffisante pour de nombreuses applications commerciales éprouvées.

Un autre cas de figure se présente si l'application existante a atteint ses limites et doit être étendue. Les limitations du langage de programmation habituelle, notamment les 256 lignes de code avec maximum de 4 contacts série, rendent tout développement ultérieur impossible.

A cet effet, l'ancien schéma peut être converti en un schéma à contacts (LD). Ainsi, les limitations de la version précédente sont éliminées, et de nouvelles fonctions sont disponibles, comme la possibilité de créer des blocs fonctionnels personnalisés.

Les utilisateurs n'ont pas à renoncer au confort de la gamme Easy, mais ils doivent se familiariser avec un nouveau langage de programmation dans le cadre des extensions des fonctionnalités.

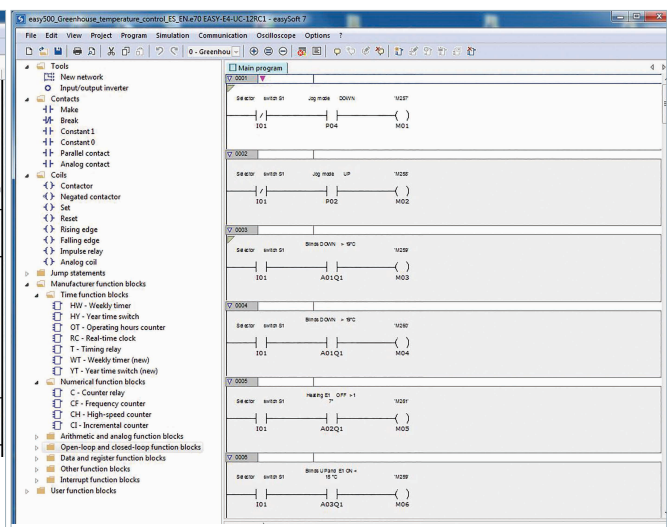
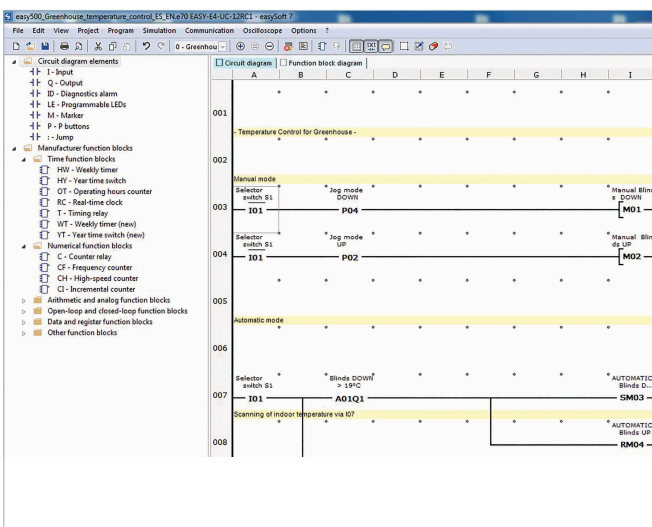
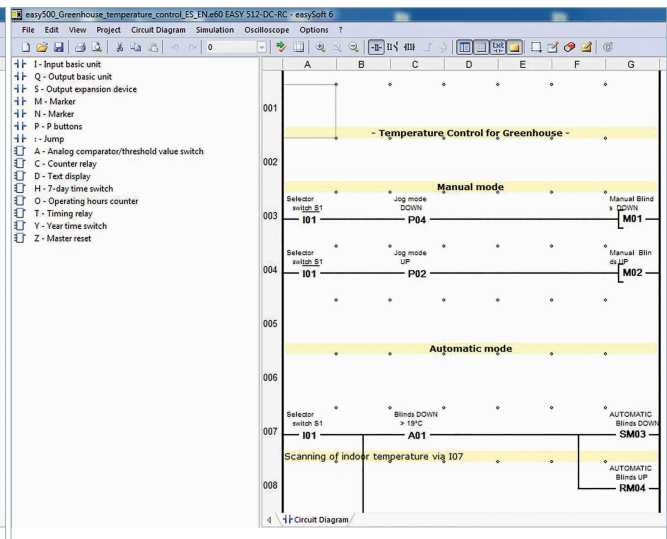
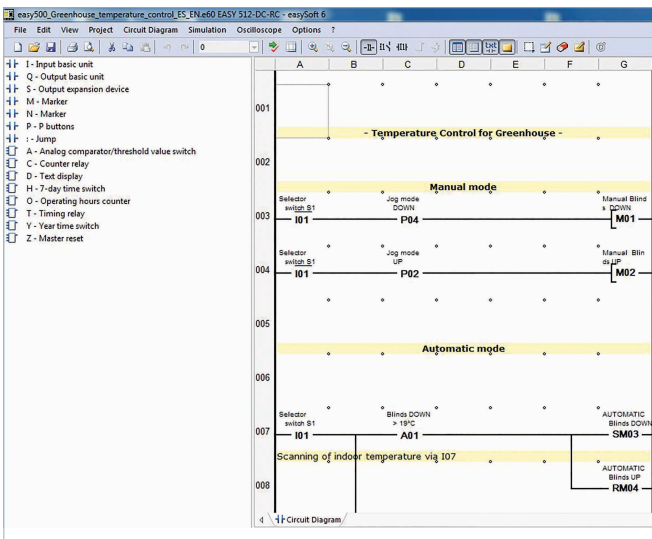
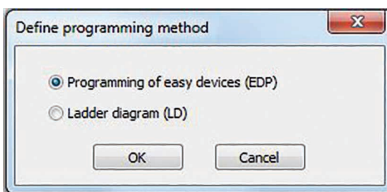
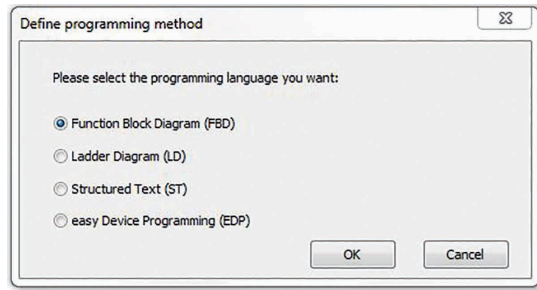


Fig. 4 : Comparaison de la programmation EDP d'easySoft V6 / easySoft V7

2.2 Langages de programmation pour les nouveaux projets

easySoft V7 contient quatre langages de programmation pour créer un nouveau projet.



La décision quant au langage de programmation à utiliser doit être bien réfléchi, car il est impossible de le changer ultérieurement. Quels sont les facteurs à prendre en compte ? L'aspect le plus important est que toutes les personnes impliquées dans le projet soient capables de maîtriser et de comprendre le langage de programmation en question.

Même si la mise en œuvre de certaines procédures dans un texte structuré peut être beaucoup plus facile que dans le schéma fonctionnel ou le schéma à contacts, cet avantage se perd rapidement si l'utilisateur ne connaît pas ce type de langage de programmation. D'autre part, certains spécialistes sont plus à l'aise dans les environnements de programmation basés sur la norme CEI 61131, tels que CODESYS. L'easyE4 leur donne la possibilité de réaliser des projets moins volumineux à l'aide d'un module logique sans avoir à renoncer à la programmation ST.

Mais qu'en est-il du langage de programmation EDP de l'appareil Easy ? EDP ne devrait plus constituer le premier choix pour les nouveaux projets. EDP et le schéma à contacts (LD) sont très similaires, ce qui rend la programmation schéma à contacts (LD) facile et rapide à apprendre et permet le déploiement de nouvelles fonctions.

Le schéma montre les différences de programmation avec FBD, LD et ST en utilisant l'exemple d'une fonction simple d'automatisme.

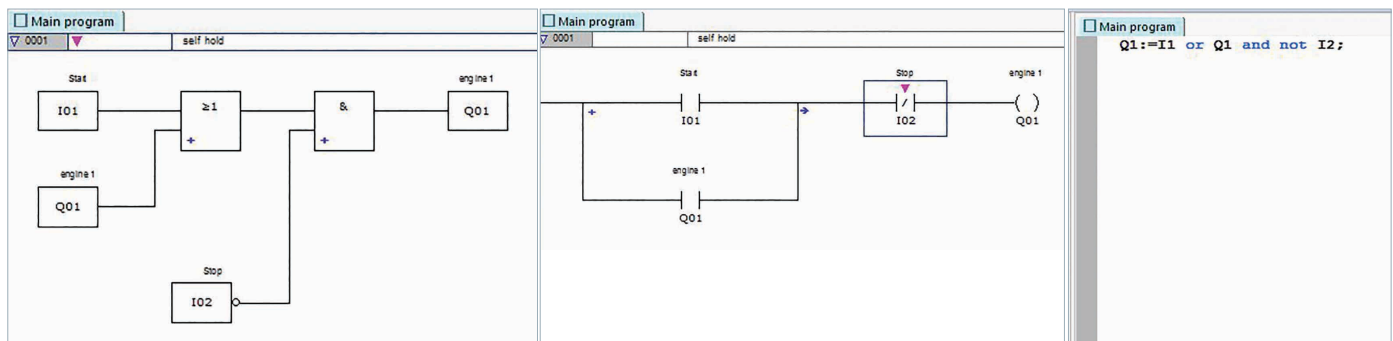


Fig. 5 : Représentation d'une fonction d'automatisme en FBD, LD, et ST

2.3 Blocs fonctionnels personnalisés/prédéfinis

Les modules logiques disposent généralement de diverses fonctions intégrées. Il s'agit notamment de fonctions élémentaires, telles que :

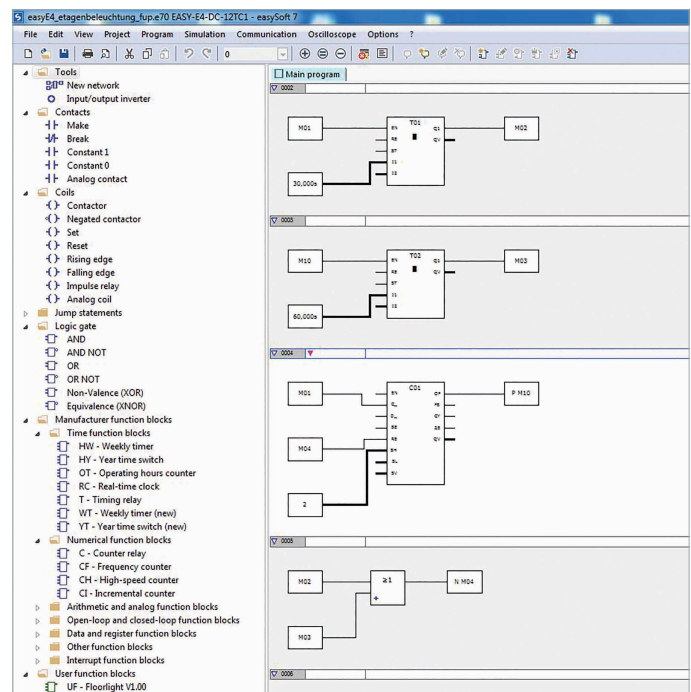
- Horloges hebdomadaire et annuelle
- Relais temporisés
- Comparateurs
- Compteurs

Mais aussi de fonctions plus sophistiquées, telles que :

- Modules arithmétiques
- Modulation de largeur d'impulsion
- Registres à décalage
- Enregistrement de données (data logging)
- Contrôleurs à trois positions
- Régulateurs PID

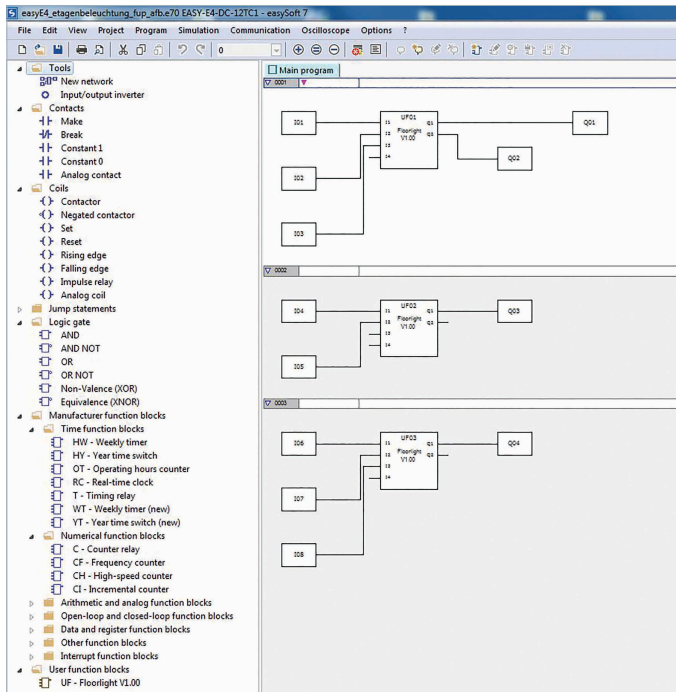
Cependant, il existe des applications pour lesquelles le module correct peut faire défaut, ce qui oblige l'utilisateur à reprogrammer plusieurs fois certains programmes, bien qu'en principe seuls les contacts d'entrée et de sortie respectifs doivent être réaffectés.

Cette procédure est non seulement laborieuse, mais elle consomme aussi beaucoup plus de ressources que les modules prédéfinis qui peuvent être tout simplement enfichés à répétition.



Le schéma indique comment programmer le système d'éclairage pour une partie d'un bâtiment en langage de schéma fonctionnel. Il y a jusqu'à quatre interrupteurs et deux luminaires. Un temps de fermeture a été configuré, et les lumières commencent à clignoter peu avant la fin du temps d'éclairage. La période d'éclairage peut être prolongée en appuyant une deuxième fois.

La séquence de programme constitue uniquement la partie d'un étage. En cas de deux, trois étages ou plus, la séquence doit être copiée aussi souvent que nécessaire tout en adaptant les entrées et sorties individuellement à chaque étage. Il en résulte un programme aussi long que confus.



Il est possible de suivre une autre approche avec les blocs fonctionnels personnalisés. L'utilisateur peut paramétrer et tester en toute simplicité la fonction pour un seul étage, puis créer un bloc fonctionnel sous un nom spécifique, par exemple « éclairage du couloir ». Le langage de programmation du module personnalisé ne doit pas nécessairement être identique à celui du programme principal. Par exemple, si le programme principal est écrit en FBD, le module personnalisé peut être programmé si besoin en ST.

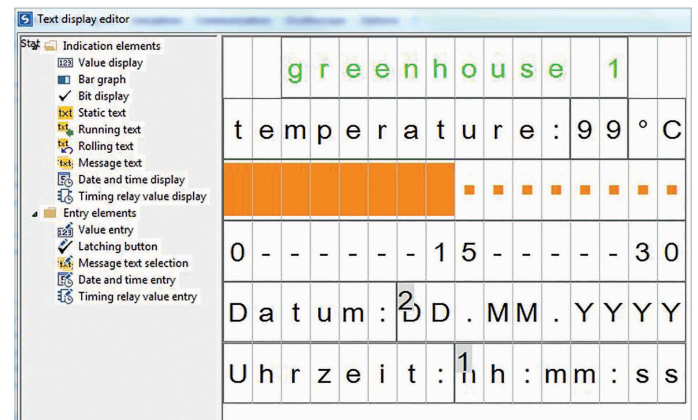
Pour appliquer le bloc fonctionnel à chaque étage, il suffit de l'intégrer dans le programme principal et de le relier aux entrées et sorties correspondantes pour les interrupteurs et les luminaires. Le programme peut être créé en un rien de temps, avec une interface claire et compréhensible. Une fois le module fonctionnel programmé et testé, les erreurs potentielles de programmation sont également réduites à un minimum.

De plus, le nouveau module peut être aussi bien intégré dans le projet en cours qu'utilisé dans tous les nouveaux projets. Au fil du temps, les utilisateurs peuvent ainsi constituer leur propre bibliothèque de blocs fonctionnels, ce qui réduit considérablement le temps de planification, de test et de mise en service d'un projet similaire.

3. Visualisation

3.1 Affichage local

Les modules logiques sont dotés d'un écran qui permet non seulement de configurer et de programmer les appareils, mais aussi de configurer les paramètres des différents modules (par ex. les horloges) en cours de fonctionnement ou de visualiser les valeurs actuelles de la machine.

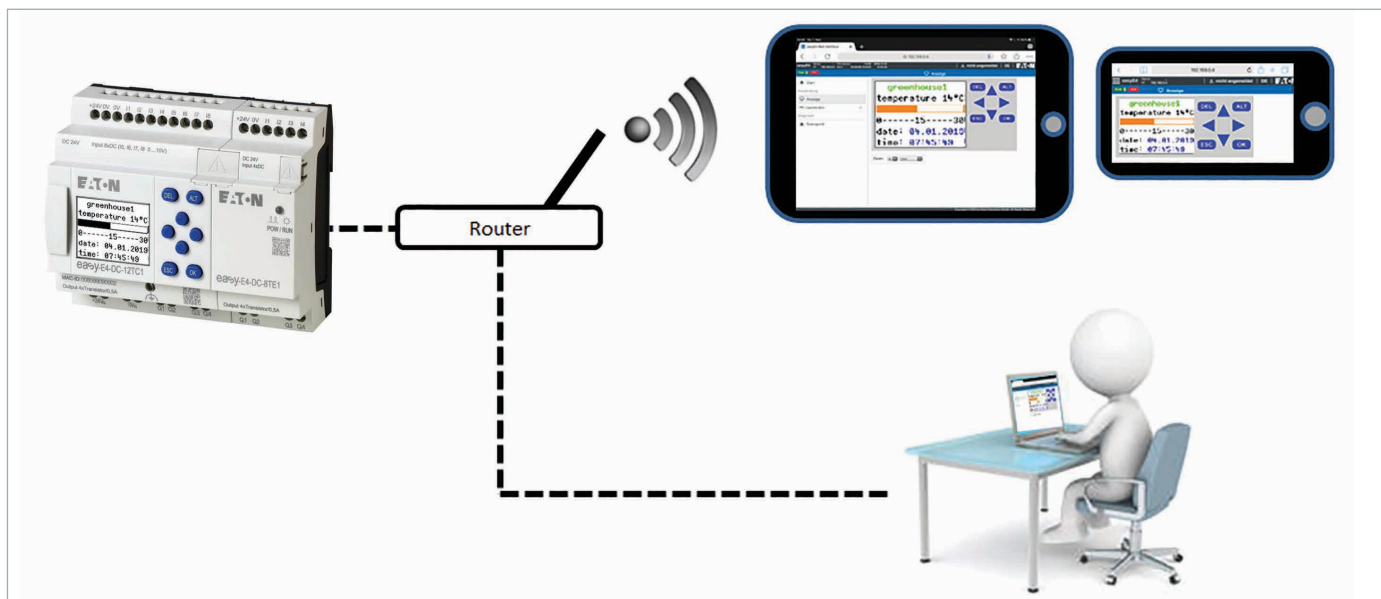


Le logiciel de programmation permet de créer un bloc fonctionnel qui indique les valeurs à afficher, telles que les niveaux de remplissage, les températures, les quantités, les heures de service ou les messages. Il s'agit d'une alternative beaucoup moins coûteuse que les IHMs fréquemment utilisées dans les applications API.

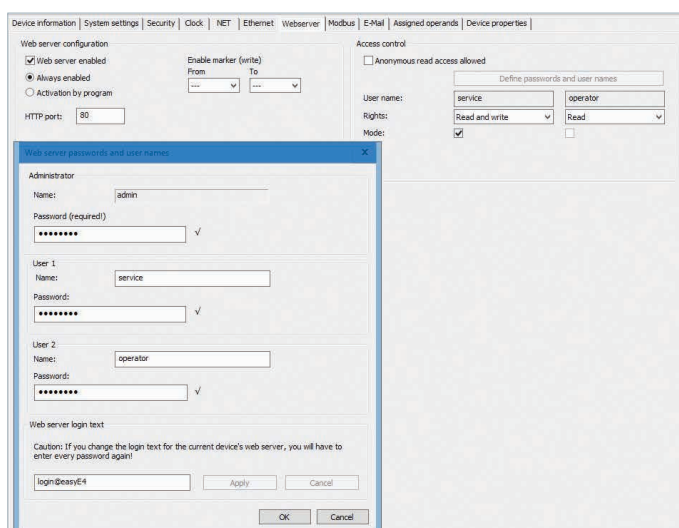
Cependant, il faut également préciser que les fonctionnalités offertes par ces deux solutions ne sont pas entièrement identiques.

3.2 Visualisation Web

L'afficheur intégré de l'easyE4 peut être un inconvénient dans la mesure où les modules logiques sont en général situés derrière la porte de l'armoire. Cependant, le serveur web intégré et la connexion des appareils via des réseaux LAN/WLAN standard constituent une alternative. En activant le serveur Web, le contenu à afficher est automatiquement converti en pages Web HTML5 qui peuvent être visualisées avec n'importe quel navigateur standard. Cela permet également aux utilisateurs d'accéder au module logique via un smartphone ou une tablette. En outre, les appareils peuvent également être commandés à l'aide des touches de fonction de l'écran.



Il va sans dire que l'accès du serveur web au module logique est protégé par un mot de passe.



En plus des droits d'administrateur, l'easyE4 offre également la possibilité de configurer des niveaux d'autorisation supplémentaires avec différents mots de passe et droits de lecture/écriture.

4. Conclusion

Revenons à la question soulevée au début de ce document. La nouvelle génération de modules logiques est-elle une alternative économique et évolutive aux automates programmables dans les applications de faible à moyenne complexité ?

La réponse est clairement oui.

- Plus il y a des possibilités d'extension au niveau des E/S, moins le risque de devoir passer ultérieurement à un automate est élevé.
- Les langages de programmation avancés, notamment le texte structuré, permettent de créer de nouvelles approches qui répondent également aux exigences des spécialistes de l'automatisation.
- Avec la vaste bibliothèque de modules prédéfinis, la possibilité de créer des blocs fonctionnels réutilisables et sécurisés permet d'assembler et de structurer le programme principal à partir d'un instrument existant. Cela permet d'éviter les erreurs et de réduire considérablement le temps nécessaire à la programmation.
- De plus, les systèmes de bus de terrain basés sur Ethernet tels que Modbus TCP facilitent la connexion au cloud et les communications entre les solutions du système.

Les modules logiques sont également accessibles à partir d'un smartphone ou d'une tablette via le serveur web intégré.

Des fonctions supplémentaires telles que la possibilité d'interrompre le processus, les modules d'alarme, la configuration des temps de cycle ou l'enregistrement des données permettent d'élargir les possibilités d'application.

Les utilisateurs ont donc vraiment l'embarras du choix. Comme le montre l'exemple de l'easyE4, les modules logiques jouissent de capacités tellement étendues que les utilisateurs les utiliseront de plus en plus à l'avenir.

Chez Eaton, nous recherchons des solutions pour fournir de l'énergie à un monde de plus en plus exigeant. Avec plus de 100 ans d'expertise dans le domaine de la gestion de l'énergie, nous sommes préparés pour l'avenir. Les industries clés du monde entier comptent sur Eaton, nos produits innovants, nos solutions globales et nos services d'ingénierie.

Les entreprises savent qu'elles peuvent nous faire confiance grâce à nos solutions pour la gestion de l'énergie qui sont fiables, efficaces et sûres. Notre service et notre assistance personnalisés, ainsi que notre culture anticipative font partie de ces solutions qui répondent aux besoins de demain, dès aujourd'hui. Le futur est en marche. Consultez **eaton.fr**



Eaton Industries (France) S.A.S
Bâtiment Axe Étoile
103-105 rue des Trois Fontanot
92000 Nanterre

Service commercial France:
N°Vert: 0 800 336 858
www.eaton.f

© 2019 Eaton Corporation
Tous droits réservés
Désignation : WP050001FR / CSSC-GL-5078
Avril 2019

Eaton est une marque déposée
d'Eaton Corporation.

Toutes les autres marques appartiennent
à leurs propriétaires respectifs.

Suivez-nous sur les réseaux sociaux pour
connaître les toutes dernières informations sur
les produits et le support.

