I° PARTIE DU CHANTIER : PRÉPARATION

Durée : 1h

Objectif : Identifier les différents éléments Software et Hardware permettant de programmer l'Automate Programmable Industriel API de la porte de garage industriel

I. Identifier les différents composants de l'installation

I.1) API - Automate Programmable Industriel C11

Identifier les différents composants de l'installation : → p.15/25	

a) ***Identifie*** *les différentes caractéristiques de l'Automate Programmable Industriel (API) :*

- *Marque, Modèle, Référence ;*
- *Les tensions d'alimentation de l'API et des Entrées de l'API*
- *Le nombre d'Entrées et de Sorties*

II. Identifier les différents composants de l'installation : C11

II.1) Entrées

Entrées → p.15/25	

a) ***Recherche*** *et* ***note*** *à quelles entrées de l'API sont raccordés les différents boutons et capteurs de l'installation*

II.2) Sorties

Sorties → p.15/25	

a) ***Recherche*** *et* ***note*** *quels voyants et bobines de contacteurs alimentent les différentes sorties de l'API*



FAIRE VÉRIFIER PAR LE PROFESSEUR

Signature professeur

II° PARTIE DU CHANTIER : SIMULATION DE FONCTIONNEMENT

Pour cette partie, tu vas utiliser l'application de schématisation et simulation ScElSim :

https://scelsim.osec.fr/

I. Identification des composants : C10

Identification des composants : → p.16/25

a) ***Connecte-toi*** *avec tes identifiants dédiés ;*

b) ***Clique*** *sur Fichier → Bibliothèque de schémas ;*

c) ***Choisis*** *le schéma « Porte de garage v02 »*

d) ***Clique*** *sur « Popup glossaire au survol »*

e) ***Survole*** *chaque composant et* ***identifie*** *le nom et le rôle de chacun d'eux, tu seras interrogé oralement pour en faire une synthèse*



FAIRE VÉRIFIER PAR LE PROFESSEUR

Signature professeur

II. Simulation de fonctionnement : C8 - C10

a) ***Clique*** *sur « Simuler »*

b) *Tu peux Afficher les potentiels si tu veux*

c) ***Ferme*** *tous les 4 sectionneurs et disjoncteurs*

d) ***Passe*** *au Folio N°2*

e) ***Clique*** *sur « GRAFCET », **agrandis** et **déplace** la fenêtre du GRAFCET de façon à ce qu'elle ne gêne pas l'affichage*

Phénomènes observés X0 → X1 → p.16/25

Passage de l'étape X0 à l'étape X1

Il faut que les entrées de la condition %I0.0 et %I0.4 et (%I0.1 ou %I0.2) soit activées pour que la porte s'ouvre, c'est à dire : RUN et Porte fermée et (Demande extérieure ou Demande extérieure).

f) ***Mets** les différents boutons et capteurs dans les positions nécessaires pour répondre à cette condition*

g) ***Indique** ce que tu constates, tu peux observer le rendu sur le Folio 1 aussi*

Phénomènes observés X1 → X2 et X2 à X3 → p.16/25

Passage de l'étape X1 à l'étape X2 et passage de l'étape X2 à l'étape X3

La porte s'ouvre, donc le capteur « Porte fermée » ne doit plus être actif

h) ***Désactive** le capteur « Porte fermée »*

i) ***Active** le capteur « Porte ouverte »*

j) ***Observe** et **note** ce que produit l'activation du capteur « Porte Ouverte »*

Phénomènes observés X3 → X0 → p.17/25

Passage de l'étape X3 à l'étape X0

k) *Quel capteur permet de passer l'étape X3 à l'étape X0 ?*

l) ***Active** ce capteur ;*

m) ***Observe** et note ce que tu observes*

Phénomènes observés X3 → X1 → p.17/25

Passage de l'étape X3 à l'étape X1

n) ***Indique** quelle condition permet de passer de l'étape X3 à l'étape X1*

o) ***Relance** un cycle et lorsque l'étape X3 est active, **actionne** ce capteur et indique les effets constatés*



FAIRE VÉRIFIER PAR LE PROFESSEUR

Signature professeur

III° PARTIE DU CHANTIER : RENSEIGNER LES DIFFÉRENTS PHASES DE FONCTIONNEMENT DE LA PORTE SELON LES EXIGENCES DE TON CLIENT

La construction du programme va s'effectuer phase par phase :

I. Phase 1 – Ouverture – Demande bp extérieur S1

Durée : 30'

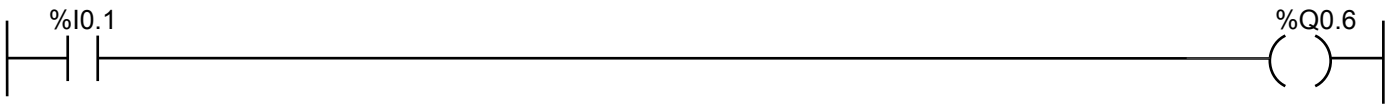
Objectif : Rédiger et tester le premier extrait de programme d'un Automate Programmable Industriel (API) permettant d'ouvrir la porte de garage

Entrées (Conditions) – Sorties (Action) :

Condition(s) *(Détection, appui bouton poussoir,...)*	Action(s) (Mouvement, témoin,...)
Appui S1 : %I0.1	Ouverture Porte :%Q0.6

Extrait du langage LADDER :

RUNG 1



I.1) Procédure et saisie d'un RUNG C3

Procédure et saisie d'un RUNG → p.17/25	

a) ***Décris*** *sommairement la procédure permettant d'accéder à la saisie du RUNG*

b) ***Saisis*** *le programme*

FAIRE VÉRIFIER PAR LE PROFESSEUR	Signature professeur

I.2) Transfert et mise en RUN de l'API C5 - C7

Transfert et mise en RUN de l'API → p.17/25	

a) ***Indique*** *sommairement la méthode de transfert du programme et de mise en RUN de l'API*

b) ***Transfère*** *le programme*

c) ***Mets*** *l'API en RUN «* ***Obligatoirement en présence du professeur »***

d) ***Effectue*** *les essais de fonctionnement*

e) ***Note*** *les défauts constatés et les modifications apportées le cas échéant*



FAIRE VÉRIFIER PAR LE PROFESSEUR

Signature professeur

II. Phase 2 – Ouverture – Demande bp extérieur S1 ou bp intérieur S2

Durée : 30'

Objectif : Modifier le programme pour prendre en compte une 2° commande d'ouverture de porte

II.1) Entrées – Sorties et RUNG 1 C3 - C11

Entrées – Sorties → p.18/25 et Modification RUNG 1 → p.18/25	

Le RUNG 1 modifié doit permettre d'ouvrir la porte si on appuie sur Demande « bp extérieur S1 » **ou** « bp intérieur S2 »

En t'aidant du RUNG 1 et de la partie Identifier les différents composants de l'installation : p.15/25

a) ***Complète*** *le tableau précisant les conditions (Entrées) et Actions (Sorties)*

b) ***Modifie*** *le RUNG 1*

II.2) Transfert et mise en RUN de l'API C5 - C7

Transfert et mise en RUN de l'API → p.18/25	

a) ***Transfère*** *le programme*

b) ***Mets*** *l'API en RUN « **Obligatoirement en présence du professeur** »*

c) ***Effectue*** *les essais de fonctionnement*

d) ***Note*** *les défauts constatés et les modifications apportées le cas échéant*



FAIRE VÉRIFIER PAR LE PROFESSEUR

Signature professeur

III. Phase 3 - Fermeture dès que la porte est ouverte

Durée : 30'

Objectif : Modifier le programme pour piloter la fermeture de la porte

III.1) Entrées - Sorties et RUNG 2 C3 - C11

Entrées - Sorties → p.18/25 et RUNG 2 → p.18/25

Le RUNG 2 doit provoquer la fermeture dès que la porte est détectée ouverte

En t'aidant du RUNG1 et de la partie Identifier les différents composants de l'installation : p.15/25

*a) **Complète** le tableau précisant les conditions (Entrées) et Actions (Sorties)*

*b) **Complète** le RUNG 2*

III.2) Transfert et mise en RUN de l'API C5 - C7

Transfert et mise en RUN de l'API → p.18/25

*a) **Transfère** le programme*

*b) **Mets** l'API en RUN « **Obligatoirement en présence du professeur** »*

*c) **Effectue** les essais de fonctionnement*

*d) **Note** les défauts constatés et les modifications apportées le cas échéant*



FAIRE VÉRIFIER PAR LE PROFESSEUR

Signature professeur

IV. Phase 4 - Sécurité arrêts moteur porte ouverte et porte fermée

Durée : 30'

Objectif : Modifier le programme pour sécuriser les fins de courses Porte ouverte et Porte fermée

IV.1) Entrées - Sorties et RUNG 1 et 2 C3 - C11

Entrées - Sorties et RUNG 1 et 2 → p.20/25 et Modifications RUNG 1 et 2 → p.20/25	

Les RUNG 1 et 2 modifiés doivent permettre de sécuriser l'installation en provoquant l'arrêt de la porte lors des détections de « Porte ouverte » et « Porte fermée »

En t'aidant des RUNG précédents et de la partie Identifier les différents composants de l'installation : p.15/25

a) ***Complète*** *le tableau précisant les conditions (Entrées) et Actions (Sorties)*

b) ***Modifie*** *les RUNG 1 et 2*

IV.2) Transfert et mise en RUN de l'API C5 - C7

Transfert et mise en RUN de l'API → p.20/25	

a) ***Transfère*** *le programme*

b) ***Mets*** *l'API en RUN «* ***Obligatoirement en présence du professeur »***

c) ***Effectue*** *les essais de fonctionnement*

d) ***Note*** *les défauts constatés et les modifications apportées le cas échéant*



FAIRE VÉRIFIER PAR LE PROFESSEUR

Signature professeur

V. Phase 5 – Auto-maintien d'ouverture et de fermeture

Durée : 15'

Objectif : Modifier le programme pour automatiser le maintien des commandes d'ouverture et fermeture de la porte

V.1) Entrées – Sorties et RUNG 1 et 2 C3 - C11

Entrées – Sorties → p.21/25 et Modification RUNG 1 et RUNG 2 → p.21/25	

Les RUNG 1 et 2 modifiés doivent permettre de rendre automatique l'ouverture et la fermeture de la porte sans être obligé de conserver le bp de demande ouverture appuyé

En t'aidant des RUNG précédents et de la partie Identifier les différents composants de l'installation : p.15/25

a) ***Complète*** *le tableau précisant les conditions (Entrées) et Actions (Sorties)*

b) ***Modifie*** *les RUNG 1 et 2*

V.2) Transfert et mise en RUN de l'API C5 - C7

Transfert et mise en RUN de l'API → p.21/25	

a) ***Transfère*** *le programme*

b) ***Mets*** *l'API en RUN «* ***Obligatoirement en présence du professeur »***

c) ***Effectue*** *les essais de fonctionnement*

d) ***Note*** *les défauts constatés et les modifications apportées le cas échéant*



FAIRE VÉRIFIER PAR LE PROFESSEUR

Signature professeur

VI. Phase 6 – Temporisation avant descente

Durée : 30'

Objectif : Modifier le programme pour ajouter une temporisation avant fermeture de la porte

VI.1) Le bloc temporisateur C6 - C11

Le bloc temporisateur → p.22/25

Le RUNG 2 modifié doit permettre d'ajouter un temps d'attente de quelques secondes une fois que la porte est en ouverture avant de donner l'ordre de fermeture en ajoutant une temporisation

a) ***Complète*** *les renseignements relatifs à la temporisation*
   - *Base de temps, Présélection, Valeur courante*

b) ***Explique*** *son fonctionnement*

VI.2) Entrées – Sorties et RUNG 2 C3 - C11

Entrées – Sorties 22 → p./25 et
Modification du RUNG 2 → p.22/25

a) ***Complète*** *le tableau précisant les conditions (Entrées) et Actions (Sorties)*

b) ***Modif*** *le RUNG 2*

VI.3) Transfert et mise en RUN de l'API C5 - C6 - C7

Réglages du bloc temporisateur → p.23/25

a) ***Note*** *et* ***Règle*** *le bloc temporisateur la base de temps pour les valeurs données par le client*

b) ***Transfère*** *le programme*

c) ***Mets*** *l'API en RUN «* ***Obligatoirement en présence du professeur »***

d) ***Effectue*** *les essais de fonctionnement*

e) ***Note*** *les défauts constatés et les modifications apportées le cas échéant*



FAIRE VÉRIFIER PAR LE PROFESSEUR

Signature professeur

IV° PARTIE DU CHANTIER : LIVRAISON

Durée : 15'

Objectif : Expliquer le fonctionnement au client

I. Communication client : C13

Communication client : → p.24/25	

*a) **Explique** le fonctionnement et les détails techniques au client*

*b) Tu **chercheras** à connaître son niveau de satisfaction et de compréhension*

V° PARTIE DU CHANTIER : FIN DE CHANTIER

Durée : 15'

I. Mémoires PC et API

Communication client : → p.24/25	

*a) **Vide** la mémoire de l'automate*

*b) **Supprime** le fichier de travail*

*c) **Supprime** la configuration API*



FAIRE VÉRIFIER PAR LE PROFESSEUR

Signature professeur

Communication client : → p.24/25	

*d) **Débranche** la maquette de la porte de garage*

*e) **Éteins** le PC*

*f) **Range** le poste de travail*

*g) **Effectue** ton auto-évaluation CPro*

*h) **Remets** ton document au professeur*

Dossier N°1 : Dossier technique

I. Dossier technique de la Porte de Garage

https://osec.fr/tp_ressources/systemes_doc/41_porte_de_garage.pdf

II. Utilisation du logiciel :

- Ouvre le logiciel "EcoStruxure Machine Expert Basic"
- Clique dans "Programmation"
- Crée un nouveau projet
- Sélectionne la bonne référence de l'automate
- Dans l'onglet PROGRAMME : Nouveau Programme
- Ajoute une section
- Commence à programmer

III. Introduction Ladder :

Ladder Diagram (LD) ou **Langage Ladder** ou **Schéma à contacts** est un langage graphique très populaire auprès des automaticiens pour programmer les automates programmables industriels. Il ressemble un peu aux schémas électriques, et est facilement compréhensible.

Ladder est le mot anglais pour échelle.

III.1) Principe :

Un programme Ladder se lit de haut en bas et l'évaluation des valeurs se fait de gauche à droite. Les valeurs correspondent en fait, si on le compare à un schéma électrique, à la présence ou non d'un potentiel électrique à chaque nœud de connexion.

En effet, le Ladder est basé sur le principe d'une alimentation en tension représentée par deux traits verticaux reliés horizontalement par des bobines, des contacts et des blocs fonctionnels, d'où le nom 'Ladder' (échelle).

C'est un langage volontairement simple et graphique pour être compréhensible. Il a permis dans les années 1990 d'être utilisé sans lourde formation par un électricien. Il est aujourd'hui un peu dépassé. La puissance de calcul des CPU actuelle permet de travailler directement en langage objets avec des notions de classe et d'héritage.

III.2) Les composants du langage

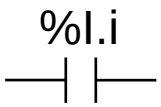
Il existe 3 types d'élément de langage :

- les entrées (ou contact), qui permettent de lire la valeur d'une variable booléenne ;
- les sorties (ou bobines) qui permettent d'écrire la valeur d'une variable booléenne ;
- les blocs fonctionnels qui permettent de réaliser des fonctions avancées.

IV. Les entrées (ou contacts) :

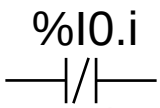
a) Il existe deux types de contact :

- Le contact normalement ouvert (NO) (en : NO Normally Open) :



Ce contact est fermé lorsque la variable booléenne associée (%I0.i) est vraie, sinon, il est ouvert.

- Le contact normalement fermé (NF) (en : NC Normally Closed) :

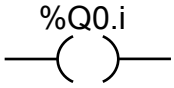


Ce contact est ouvert lorsque la variable booléenne associée (%I0.i) est vraie, sinon il est fermé.

V. Les sorties (ou bobines) :

Il existe, deux types de bobines :

- la bobine **monostable** est active uniquement lorsqu'elle est pilotée et se désactive dès qu'elle ne l'est plus :

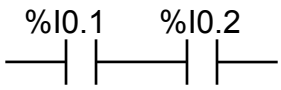
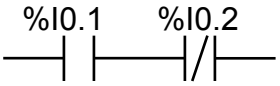
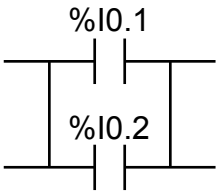


- la bobine **bistable** est activée est reste active grâce à la commande **S** (Set) et est désactivée grâce à la commande **R** (Reset) :

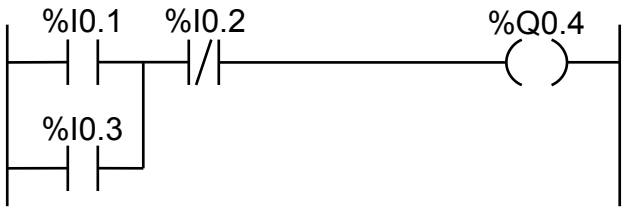


V.1) Réalisation de fonction logique :

Comme dit précédemment, les fonctions logiques sont dérivées de leurs réalisations électriques. Donc chaque fonction logique (AND, OR, XOR, NAND, NOR, NOT) a une représentation qui correspond à son équivalent électrique.

 Équivaut à %I0.1 AND %I0.2 Équivaut à %I0.1 . %I0.2	 Équivaut à NOT(%I0.1) AND %I0.2 Équivaut à $\overline{\%I0.1}$ AND %I0.2 Équivaut à $\overline{\%I0.1}$. %I0.2	 Équivaut à %I0.1 OR %I0.2 Équivaut à %I0.1 + %I0.2

V.2) Exemple de lecture :



Dans ce réseau, si %I0.1 OU %I0.3 est actionné ET si %I0.2 n'est pas actionné, la sortie %Q0.4 est active, soit :

%Q0.4 = (%I0.1 + %I0.3) . $\overline{\%I0.2}$

Cela signifie que $\overline{\%I0.2}$ est le « complément » de %I0.2.

II. Sécurité au travail et protection de l'environnement

CONSIGNES DE SÉCURITÉ SUR POSTE DE TRAVAIL

Porte de Garage

