

Créé le 10 mai 2019 - Actualisé le 7 avril 2021

- ✓ Formation dispensée en présentiel
- ✓ Accessible aux handicapés

Réf. API_SCH_SOMACH_LANGAGE

Tarif forfaitaire

Prix inter-entreprises : nous consulter
Prix intra-entreprise : nous consulter

Objectifs - aptitudes et compétences

- Concevoir des programmes pour les API M221, M251, ... avec le logiciel SoMachine
- Savoir structurer et mettre au point l'application
- Etre capable d'utiliser la plupart des instructions dans les divers langages IEC 61131-3

Public concerné

- Toute personne devant développer ou maintenir une application d'automatisme utilisant le logiciel SoMachine.

Pré-requis

- Connaître les bases de la logique programmée et des automates programmables

Moyens pédagogiques

- Automate programmables Modicon M251
- PC portables
- Logiciel SoMachine
- Simulateurs

Documentation

- 1 documentation par stagiaire. La documentation fournie au stagiaire pendant sa formation est utilisable au quotidien dans l'entreprise au cours de son activité professionnelle.

Sanction

Un certificat de réalisation, mentionnant les objectifs, la nature et la durée de l'action, sera fourni. Les résultats de l'évaluation des acquis de la formation seront remis au donneur d'ordre à l'issue de la prestation.

Pédagogie / Qualité / Evaluation

La pédagogie est de type D.I.A. : Découverte, Intégration, Ancrage :

► La **découverte** a pour objectif de sensibiliser les acteurs aux concepts à l'origine des méthodes à appliquer sur le terrain, et de se situer par rapport aux exigences de rigueur nécessaires à leur mise en œuvre. Elle est réalisée par le biais d'exercices à caractère ludique et de réflexions/débats, relatifs à des expériences issues du site et menés en groupes.

► L'**intégration** est constituée par l'apport des connaissances relatives aux méthodes et outils définis dans le programme.

► L'**ancrage** consiste à traiter, au cours de la formation, des applications des méthodes et outils sur des sujets issus du terrain (principe de formation-action).

Suivi pédagogique : Le formateur évalue quotidiennement par des exercices pratiques la progression des stagiaires, tient compte des difficultés rencontrées et adapte son cours en conséquence.

Contrôle qualité : Toutes nos formations font l'objet d'une évaluation qualité à chaque fin de session.

Evaluation des objectifs de la formation : Nous réalisons à chaque fin de formation une évaluation à chaud sur la base des objectifs définis dans la fiche programme. Une attestation de formation est délivrée à l'issue de toutes les formations suivies dans le cadre du programme de la formation continue.

VALIDATION : attestations de présence

Copyright © avenirformation.com : Tous droits réservés. Le programme suivant est la propriété exclusive d'Avenir Formation SAS. Il est strictement interdit de copier, altérer ou modifier le contenu de celui-ci sans le consentement préalable d'Avenir Formation.

Agence d'Angers
Tél 02 41 35 00 35
angers@avenirformation.com

Agence de Douai
Tél 03 27 95 89 04 / 05
douai@avenirformation.com

Agence de Lyon
Tél 04 37 49 66 66
lyon@avenirformation.com

Agence de Mulhouse
Tél 03 89 45 26 26
mulhouse@avenirformation.com

Agence de Rouen
Tél 02 32 19 09 00
rouen@avenirformation.com

Siège social : 447 rue Jean Perrin - ZI Douai Dorignies - BP50315 - 59351 DOUAI CEDEX

SAS au capital de 200 000 € - RCS DOUAI B 388 557 118 - NAF 8559A - SIRET 388 557 118 00048 - Centre de formation déclaré n° 31 59 0216559 - TVA FR 5638857118

www.avenirformation.com

Programme

(70% pratique / 30% théorie)

- Présentation matérielle des API SCHNEIDER Modicon M251
 - Unité centrale
 - Moyens de communication
 - Cartes d'extension
- Le logiciel SoMachine
 - L'ergonomie du logiciel
 - Les onglets : Accueil, Propriétés, Configuration, Programme, Mise en service, Rapport
 - Les différents modes de création d'un projet
 - Paramétrage et configuration
 - La gestion des fenêtres et des menus de la partie programmation
- Structure d'un programme :
 - Les différents modules UOP : programme, Fonction, Bloc fonctionnel
 - Les tâches
- Les langages de programmation CEI et non CEI
 - Langage Ladder (LD)
 - Boîtes fonctionnelles (FBD)
 - Diagramme fonctionnel séquentiel (SFC)
 - Liste d'instruction (IL)
 - Texte structuré (ST)
 - Diagramme de fonction continu (CFC, non CEI)
 - Configuration des éditeurs,
- Les variables
 - Les types de données
 - Standard (BOOL, BYTE, WORD, ...)
 - Extensions à la norme (UNION, LTIME, WSTRING, REFERENCE, POINTER, ...)
 - Définis par l'utilisateur (ARRAY, STRUCT, Enumérations, ...)
- La déclaration des variables :
 - La syntaxe de déclaration
 - L'adressage des entrées/sorties
 - Les variables locales et globales
 - Les variables système
 - Les variables sauvegardées
- Les éléments de programmation
 - Opérateurs logiques, mathématiques, ... (Affectations, comparaison, ...)
 - Les fonctions et blocs fonctionnels CEI (Et, Ou, Temporisations, Compteurs, ...)
 - Aperçu du jeu d'instruction
 - Les bibliothèques constructeurs
 - Création et utilisation de bibliothèques utilisateur
- Les fonctions du logiciel
 - Les écrans de visualisation : création et utilisation pour la mise au point de l'application
 - La simulation sur PC
 - La compilation
 - Configuration de la communication avec l'API
 - Transfert du programme
 - Notion de Projet de démarrage et de code source
 - Fonctions en ligne
 - Démarrage, arrêt,
 - Mode Cycle par cycle, points d'arrêt, ...
 - Les différents modes de réinitialisation (Reset)
 - L'histogramme
 - Visualisation des variables et gestion des forçages (fenêtres de surveillance)
 - Visualisation de l'état de l'automate
- Bus CANopen
 - Principe
- Exercices pratiques de sauvegarde et restitution de programme
- Travaux pratiques : Réalisation de programmes à partir de petits cahier des charges proposés par l'animateur et destinés à mettre en œuvre les notions étudiées et les outils du logiciel

