Fatek plc programming manual Updated Version

Exercice avec solution sur GRAFCET : Comprendre et maîtriser la méthode

Exercice avec solution sur GRAFCET est une étape essentielle pour toute personne souhaitant maîtriser la programmation et l'automatisation des systèmes industriels. Le GRAFCET, ou Grafcet (GRAphe Fonctionnel de Commande Étape-Transition), est un langage graphique qui permet de modéliser le comportement des automates programmables industriels (API). Grâce à ces exercices, les étudiants et professionnels peuvent approfondir leur compréhension du fonctionnement des séquences d'automatisation, tout en se familiarisant avec la logique de commande associée.

Introduction au GRAFCET

Qu?est-ce que le GRAFCET ?

Le GRAFCET est un langage normalisé par l?IEC (International Electrotechnical Commission) qui sert à représenter de manière graphique la logique de contrôle d?un système automatisé. Il se compose principalement d?étapes, de transitions, d?actions, et de conditions de transition.

Les composants principaux du GRAFCET

- **Étapes** : représentent un état ou une phase du processus.
- **Transitions** : indiquent le passage d?une étape à une autre, sous condition.
- **Actions** : opérations ou consignes effectuées lors de l?activation d?une étape.
- **Conditions** : sont les critères qui doivent être remplis pour qu?une transition puisse s?effectuer.

Pourquoi pratiquer avec des exercices sur GRAFCET ?

Les exercices avec solutions permettent d?acquérir une compréhension pratique du langage GRAFCET. Ils aident à :

- Maîtriser la lecture et l?interprétation des diagrammes GRAFCET.
- Savoir concevoir des séquences d?automatisation simples ou complexes.
- Vérifier la compréhension des règles de conception et de sécurité.

- Se préparer à la programmation effective sur automate industriel.

Exemple d?exercice avec solution sur GRAFCET : la commande d?un système de porte automatique

Énoncé de l?exercice

Concevez un GRAFCET pour un système de porte automatique qui s?ouvre lorsque quelqu?un s?approche, reste ouverte pendant 10 secondes, puis se ferme automatiquement. La porte doit également pouvoir être fermée manuellement via un bouton de fermeture. Utilisez les composants suivants :

- Capteur de proximité (S) : active lorsque quelqu?un est proche.
- Bouton de fermeture (F) : action manuelle pour fermer la porte.
- Actuateur de porte (O) : ouvre ou ferme la porte.
- Temporisateur (T) : pour maintenir la porte ouverte pendant 10 secondes.

Conditions initiales

- La porte est fermée au départ.
- Le capteur de proximité active lorsque quelqu?un s?approche.
- Le bouton F peut être activé à tout moment pour fermer la porte.

Objectifs de l?exercice

- Modéliser la séquence d?ouverture de la porte lorsqu?une personne s?approche.
- Gérer la temporisation pour maintenir la porte ouverte pendant 10 secondes.
- Permettre une fermeture manuelle via le bouton F.
- Assurer la fermeture automatique après la temporisation ou manuellement.

Solution détaillée : conception du GRAFCET

Étapes de la conception

Voici la démarche pour élaborer le GRAFCET correspondant :

- Identifier les états principaux : porte fermée, porte ouverte, porte en cours d?ouverture, porte en

cours de fermeture.

- Définir les transitions entre ces états en fonction des conditions (capteur, bouton, temporisateur).
- Attribuer les actions à chaque étape (activer l?actuateur, démarrer le temporisateur, etc.).

Diagramme GRAFCET proposé

Voici une représentation simplifiée du GRAFCET pour ce système :

- **Étape 1 : Porte fermée (Initiale)**
- **Transition 1 : Capteur S activé** (personne s?approche) ? *Vers*
- **Étape 2 : Porte en ouverture**
- **Actions :** Activer l?actuateur O pour ouvrir la porte, démarrer le temporisateur T pour 10 secondes.
- **Transition 2 : Temporisateur T terminé** ? *Vers* étape 3
- **Étape 3 : Porte ouverte**
- **Actions :** Maintenir la porte ouverte, attendre une nouvelle commande (F ou S). Si F activé ? transition vers étape 4.
- **Transition 3 : F bouton activé** ? *Vers* étape 5
- **Étape 4 : Porte en fermeture automatique**
- **Actions :** Activer l?actuateur O pour fermer la porte.
- **Transition 4 : Porte fermée** ? *Vers* étape 1
- **Étape 5 : Fermeture manuelle**
- **Actions :** Fermer la porte via l?actuateur, puis revenir à l?état initial.

Représentation graphique simplifiée

Un diagramme GRAFCET pourrait ressembler à ceci :

Réalisation du GRAFCET étape par étape

Étape 1 : Porte fermée

- Action : La porte est fermée.
- Condition de transition : Activation du capteur S ? passage à étape 2.

Étape 2 : Ouverture de la porte

- Actions : Allumer l?actuateur O pour ouvrir, démarrer T pour temporiser 10s.
- Transition : Fin du temporisateur T ? étape 3.

Étape 3 : Porte ouverte

- Actions : La porte reste ouverte, en attente d?une nouvelle commande.
- Transitions :

- F activé ? étape 4 (fermeture manuelle).
- Pas d?action ? reste à cette étape jusqu?à F ou autre événement.

Étape 4 : Fermeture automatique

- Actions : Activer l?actuateur O pour fermer la porte.
- Transition : La porte est fermée ? étape 1.

Étape 5 : Fermeture manuelle

- Actions : Fermer la porte manuellement.
- Transition : Retour à l?état initial, porte fermée.

Vérification et simulation

Une fois le GRAFCET conçu, il est crucial de procéder à une vérification pour s?assurer de la cohérence et de la sécurité du système. La simulation permet de tester le comportement du diagramme dans différentes situations :

- Approche d?une personne
- Activation du bouton F en cours d?ouverture
- Arrêt du temporisateur
- Fermeture manuelle

Outils pour réaliser des exercices GRAFCET

Plusieurs logiciels permettent de créer, simuler, et analyser des diagrammes GRAFCET :

- Grafcet Designer
- Siemens LOGO! Soft Comfort
- Fatek GRAFCET Tool
- OpenGRAFCET

Exercice avec Solution sur GRAFCET : Un Guide Complet pour Maîtriser la Programmation des Automates

Le GRAFCET (Graphe Fonctionnel de Commande Étape-Transition) est un langage graphique qui joue un rôle central dans la conception et la programmation des systèmes automatisés. Utilisé dans l'automatisation industrielle, il permet de modéliser de manière claire et structurée le comportement d?un système, facilitant ainsi la programmation, la maintenance et la dépannage des automates programmables industriels (API). Dans cet article, nous vous proposons une exploration approfondie du GRAFCET à travers un exercice détaillé, accompagné de sa solution, pour vous aider à maîtriser cette méthode essentielle.

Introduction au GRAFCET

Qu?est-ce que le GRAFCET ?

Le GRAFCET est un langage de modélisation graphique basé sur une logique de transitions et d?états, permettant de représenter en un seul coup d??il le comportement séquentiel d?un système. Il est souvent utilisé dans la conception de machines automatisées, telles que convoyeurs, presses, robots, etc.

Concrètement, un GRAFCET se compose :

- Étapes : représentant les états ou phases du système.
- Transitions : indiquant les conditions nécessaires pour passer d?une étape à une autre.
- Actions : associées à chaque étape, décrivant ce qui doit être réalisé lorsque l?étape est active.
- Conditions de transition : les événements ou signaux qui déclenchent la progression vers la prochaine étape.

Intérêt et avantages du GRAFCET

- Clarté visuelle : sa représentation graphique facilite la compréhension.
- Facilité de modification : les modifications sont simples à intégrer.
- Standardisation : norme internationale (norme IEC 60848).
- Intégration dans la programmation : permet une traduction directe en langage de programmation pour automate.

Exercice pratique : Modélisation d?un système de convoyeur automatique

Supposons que vous deviez concevoir un système de convoyeur simple avec deux étapes principales :

- Étape 1 : Le convoyeur est arrêté.
- Étape 2 : Le convoyeur tourne pour déplacer un produit.

Le système doit respecter les règles suivantes :

- Lorsqu?un produit est détecté à l?entrée (capteur S1), le système doit démarrer le convoyeur.
- Le convoyeur doit continuer de tourner jusqu?à ce que le produit atteigne la sortie (capteur S2).
- Lorsqu?un produit atteint la sortie (S2), le convoyeur doit s?arrêter.
- Le système doit pouvoir être arrêté manuellement via un bouton d?arrêt (SB).

Analyse détaillée de l?exercice

Définition des éléments

- Étapes :

- `E1` : État initial, convoyeur arrêté.
- `E2` : Convoyeur en marche.

- Transitions :

- `T1` : Activation du démarrage par détection du produit à S1.
- `T2` : Arrêt du convoyeur lorsque le produit est détecté par S2.
- `T3` : Arrêt manuel par SB.

- Actions :

- Sur `E2` : commande pour faire tourner le moteur du convoyeur.

- Variables de sortie :

- `MOTEUR` : action de démarrage/arrêt du moteur.

- Conditions de détection :

- `S1` : capteur détectant un produit à l?entrée.
- `S2` : capteur détectant un produit à la sortie.
- `SB` : bouton d?arrêt manuel.

Création du GRAFCET étape par étape

- Définir l?état initial

L?état initial est que le convoyeur est arrêté. Cela correspond à l?étape `E1` activée. La sortie `MOTEUR` est désactivée.

- Définir la transition de `E1` à `E2`

Pour passer de `E1` à `E2`, deux conditions doivent être remplies :

- La détection d?un produit à S1 (`S1 = 1`).
- Aucun arrêt manuel (`SB = 0`).

La transition `T1` s?active dès que ces conditions sont remplies.

- Définir l?étape `E2`

Lorsque le système est en `E2`, le moteur doit tourner. La sortie `MOTEUR` est activée. La transition vers `E1` se fait lorsque le produit atteint S2 (`S2 = 1`) ou si l?arrêt manuel est activé (`SB = 1`).

- Transitions de `E2` à `E1`

- Si `S2 = 1`, cela indique que le produit est arrivé à la sortie, il faut arrêter le convoyeur.
- Si `SB = 1`, l?opérateur arrête manuellement la machine.

Représentation graphique du GRAFCET

Voici une description textuelle de la structure :

```plaintext

[ E1 ] --(S1=1 AND SB=0)--> [ E2 ] --(S2=1 OR SB=1)--> [ E1 ]

```

- Lorsque `E1` est actif, `MOTEUR` est OFF.
- Lorsqu?on passe à `E2`, `MOTEUR` devient ON.
- Le retour à `E1` se fait lorsque `S2=1` ou `SB=1`.

Solution détaillée sous forme de GRAFCET

Étape 1 : Initialisation

- Étape : `E1` (Convoyeur arrêté)
- Action : `MOTEUR=0`

Transition 1 : De `E1` à `E2`

- Condition : `S1=1` ET `SB=0`
- Action : Passage à `E2`, moteur ON.

Étape 2 : En marche

- Étape : `E2`
- Action : `MOTEUR=1`

Transition 2 : Retour à `E1`

- Condition : `S2=1` OU `SB=1`
- Action : Arrêt du moteur, retour à `E1`.

Traduction GRAFCET en logique programmable

Ce modèle peut être traduit en langage de programmation pour automate ou en diagramme de logique combinatoire. Voici une proposition simplifiée en pseudo-code :

```pseudo
SI (E1 active ET S1=1 ET SB=0) ALORS
E1 := FAUX
E2 := VRAI
MOTEUR := 1
FIN SI
SI (E2 active ET (S2=1 OU SB=1)) ALORS
E2 := FAUX
E1 := VRAI
MOTEUR := 0
FIN SI
```

Ce code illustre la logique de transition entre états en fonction des capteurs et commandes.

Conseils pour la mise en pratique et l?apprentissage

- Pratique régulière : Le GRAFCET se maîtrise par la pratique, en modélisant différents systèmes

automatisés.

- Utilisez des outils logiciels : Des logiciels comme WinGRAFCET ou Siemens S7-GRAPH facilitent la création et la simulation.
- Analysez des exemples concrets : Étudiez des cas réels d?automatisation pour comprendre les subtilités.
- Vérifiez la cohérence : Toujours vérifier la logique pour éviter les blocages ou comportements inattendus.

Conclusion

Maîtriser le GRAFCET à travers des exercices avec solution constitue une étape essentielle pour tout professionnel ou étudiant en automatisme industriel. La modélisation graphique simplifie la compréhension et la programmation des systèmes automatisés complexes. En intégrant cet outil dans vos projets, vous gagnerez en efficacité, en fiabilité et en capacité d?analyse.

Que vous soyez débutant ou expérimenté, la pratique régulière, accompagnée d?études de cas concrets, vous permettra d?acquérir une expertise solide en GRAFCET. N?oubliez pas que chaque exercice est une occasion d?apprendre et d?affiner votre maîtrise de l?automatisation industrielle.

En résumé :

- Le GRAFCET facilite la conception séquentielle.
- La compréhension de ses éléments (étapes, transitions, actions) est fondamentale.
- La pratique régulière d?exercices, comme celui présenté, permet d?acquérir une compétence opérationnelle.
- La maîtrise du GRAFCET ouvre la porte à des systèmes automatisés plus complexes et performants.

N?hésitez pas à expérimenter avec différents scénarios pour enrichir votre compréhension et à utiliser des outils de simulation pour valider vos modèles. La clé du succès réside dans la pratique et la réflexion structurée.

QuestionAnswer Qu'est-ce qu'un GRAFCET et à quoi sert-il dans l'automatisme industriel? Le GRAFCET est un langage graphique utilisé pour représenter le fonctionnement d'un automate industriel. Il sert à concevoir, analyser et programmer des systèmes automatisés en décrivant les étapes, transitions et actions nécessaires au processus. Comment représenter une étape et une transition dans un GRAFCET? Une étape est représentée par un rectangle ou un cercle rempli, indiquant une situation particulière. Une transition est représentée par un trait horizontal ou vertical entre deux étapes, souvent avec une condition de déclenchement associée, indiquant le passage d'une étape à une autre. Donnez un exemple simple d'exercice avec solution sur un GRAFCET pour

contrôler une lampe avec un bouton poussoir. Exercice: Créer un GRAFCET pour allumer une lampe quand le bouton est pressé, et l'éteindre quand il est relâché. Solution: Le GRAFCET comporte deux étapes: 'Lampe éteinte' et 'Lampe allumée'. La transition de 'éteinte' à 'allumée' est activée lorsque le bouton est pressé. La transition inverse se produit lorsque le bouton est relâché. Quels sont les principaux éléments d'un GRAFCET à maîtriser lors d'un exercice? Les principaux éléments sont les étapes, les transitions, les actions associées, les conditions de transition, et les règles de synchronisation ou de parallélisme si nécessaire. Comment réaliser un exercice pour gérer un système de convoyeur avec GRAFCET? L'exercice consiste à définir les étapes pour démarrer, arrêter, et déplacer le convoyeur, puis à représenter ces étapes et transitions en utilisant le langage GRAFCET, en intégrant les conditions de détection de présence, de commande de marche/arrêt, etc. Quels sont les erreurs courantes à éviter lors de la réalisation d'un exercice sur GRAFCET? Les erreurs courantes incluent l'omission de certaines transitions, l'utilisation incorrecte des conditions, la mauvaise représentation des actions, ou la confusion entre étapes parallèles et séquentielles. Il faut aussi vérifier la cohérence logique du diagramme. Comment tester la validité d'un GRAFCET dans un exercice pratique? On peut effectuer une simulation en suivant les différentes séquences possibles, vérifier que chaque transition se produit dans le bon ordre, et s'assurer que le comportement du système correspond aux spécifications initiales. Peut-on combiner plusieurs GRAFCET dans un même exercice, et comment faire? Oui, il est courant de combiner plusieurs GRAFCET pour modéliser des sous-systèmes ou des processus complexes. Il faut alors définir clairement les interfaces et les interactions entre eux, en utilisant des transitions conditionnelles ou des événements communs. Comment aborder un exercice d'élaboration de GRAFCET pour la gestion d'un processus multi-étapes? Il faut d'abord décomposer le processus en étapes principales, définir les conditions de transition, puis représenter chaque étape et transition dans le GRAFCET, en vérifiant la cohérence logique et le respect des contraintes du système. Quelles ressources ou outils peuvent aider à résoudre des exercices avec solution sur GRAFCET? Des logiciels de modélisation comme WinGRAFCET, EasyGRAFCET, ou des outils en ligne peuvent aider à créer et simuler des GRAFCET. De plus, des tutoriels, manuels techniques et cours en ligne sont utiles pour apprendre la méthode et la résolution d'exercices.

Related keywords: exercice grafcet, solution grafcet, diagramme de transition, automatisme industriel, programmation grafcet, étape grafcet, transition grafcet, automatisme programmable, exemple grafcet, cours grafcet